

بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِیْمِ

## طراحی صفحات وب (پیشرفته)

شاخه: کاردانش

زمینه: خدمات

گروه تحصیلی: کامپیوتر

زیرگروه: کامپیوتر

رشته مهارتی: طراحی صفحات وب

شماره رشته مهارتی: ۳ ۱۷ ۱۰۱ ۳۱۲

کد رایانه‌ای رشته مهارتی: ۶۱۳۷

نام استاندارد مهارت مبنا: طراحی صفحات وب (پیشرفته)

کد استاندارد متولی: ۸۷ ۱۵/۴/۲ ف، هـ

شماره درس: نظری: ۴۷۸، عملی: ۴۷۹

عنوان و نام پدیدآور: طراحی صفحات وب ... [کتاب‌های درسی] کد استاندارد متولی ۸۷-۱۵/۴/۲- ف، هـ / مؤلف زهرا سلیمی‌زاده: [برای] وزارت آموزش و پرورش، سازمان پژوهش و برنامه‌ریزی آموزشی؛ برنامه‌ریزی محتوا و نظارت بر تألیف، دفتر تألیف کتاب‌های درسی فنی و حرفه‌ای و کاردانش.

مشخصات ظاهری: ج۲: مصور.

شابک: ۹۷۸-۹۶۴-۰۵-۲۱۸۹-۲

وضعیت فهرست‌نویسی: فیا

یادداشت: ج. ۲ (چاپ اول: ۱۳۸۹) (فیا).

مندرجات: ج. ۱. مقدماتی. - ج. ۲. پیشرفته.

موضوع: وب -- سایت‌ها -- طراحی

شناسه افزوده: سلیمی‌زاده، زهرا، ۱۳۵۹ -

شناسه افزوده: سازمان پژوهش و برنامه‌ریزی آموزشی. دفتر تألیف کتاب‌های درسی فنی و حرفه‌ای و کاردانش

شناسه افزوده: سازمان پژوهش و برنامه‌ریزی آموزشی

رده‌بندی کنگره: ۴۳ ۱۳۸۹ ط/۵۱۰۵/۸۸۸ TK

رده‌بندی دیویی: ۳۷۳ / ک ۸۷-۱۵/۴/۲

شماره کتابشناسی ملی: ۲ ۷ ۵۶۵

همکاران محترم و دانش آموزان عزیز :

پیشنهادات و نظرات خود را درباره محتوای این کتاب به نشانی  
تهران - صندوق پستی شماره ۴۸۷۴/۱۵ دفتر تألیف کتاب‌های درسی فنی و  
حرفه‌ای و کاردانش، ارسال فرمایند.

[tvoccd@roshd.ir](mailto:tvoccd@roshd.ir)

پیام نگار (ایمیل)

[www.tvoccd.medu.ir](http://www.tvoccd.medu.ir)

وب‌گاه (وب‌سایت)

محتوای این کتاب در کمیسیون جلسه مورخ ۸۹/۳/۲۳ کمیسیون تخصصی رشته  
کامپیوتر دفتر تألیف کتاب‌های درسی فنی و حرفه‌ای و کاردانش با عضویت: بتول  
عطاران، محمدرضا شکرریز، محمدعباسی، الهه اسماعیل‌زاده، نادیا ماجدی و اعظم  
امیدوار تأیید شده است.

### وزارت آموزش و پرورش سازمان پژوهش و برنامه‌ریزی آموزشی

برنامه‌ریزی محتوا و نظارت بر تألیف: دفتر تألیف کتاب‌های درسی فنی و حرفه‌ای و کاردانش

نام کتاب: طراحی صفحات وب (پیشرفته) - ۶۱۲/۱۰

مؤلف: مهندس زهرا سلیمی‌زاده

ویراستار ادبی: فاطمه پورعبدل

نظارت بر چاپ و توزیع: اداره کل نظارت بر نشر و توزیع مواد آموزشی

نشانی: تهران - خیابان ایرانشهر شمالی - ساختمان شماره ۴ آموزش و پرورش (شهید موسوی)

تلفن: ۸۸۸۳۱۱۶۱-۹، دورنگار: ۸۸۳۰۹۲۶۶، کدپستی: ۱۵۸۴۷۴۷۳۵۹

وب‌سایت: [www.chap.sch.ir](http://www.chap.sch.ir)

صفحه‌آرا: آمنه درویش

طراح جلد: بیتا اشرفی مقدم

ناشر: شرکت چاپ و نشر کتاب‌های درسی ایران: تهران - کیلومتر ۱۷ جاده مخصوص کرج - خیابان ۶۱ (داروپخش)

تلفن: ۴۴۹۸۵۱۶۱-۵، دورنگار: ۴۴۹۸۵۱۶۰، صندوق پستی: ۳۷۵۱۵-۱۳۹

چاپخانه: شرکت افست «سهامی عام»

نوبت و سال چاپ: چاپ اول برای سازمان ۱۳۹۲

حق چاپ محفوظ است.

## مجموعه کتاب‌های درسی رشته کامپیوتر شاخه کاردانش

(استاندارد وزارت فرهنگ و ارشاد اسلامی)

رشته تصویرسازی	رشته طراحی صفحات وب	رشته تولید چندرسانه‌ای
مفاهیم پایه فناوری اطلاعات	مفاهیم پایه فناوری اطلاعات	مفاهیم پایه فناوری اطلاعات
سیستم‌عامل مقدماتی	سیستم‌عامل مقدماتی	سیستم‌عامل مقدماتی
اطلاعات و ارتباطات	اطلاعات و ارتباطات	اطلاعات و ارتباطات
واژه‌پرداز Word 2007	واژه‌پرداز Word 2007	واژه‌پرداز Word 2007
صفحه گسترده Excel 2007	صفحه گسترده Excel 2007	صفحه گسترده Excel 2007
ارایه مطالب PowerPoint 2007	ارایه مطالب PowerPoint 2007	ارایه مطالب PowerPoint 2007
نرم‌افزارهای اداری تکمیلی	نرم‌افزارهای اداری تکمیلی	نرم‌افزارهای اداری تکمیلی
بانک اطلاعاتی Access 2007	بانک اطلاعاتی Access 2007	بانک اطلاعاتی Access 2007
سیستم‌عامل پیشرفته	سیستم‌عامل پیشرفته	سیستم‌عامل پیشرفته
برنامه‌نویسی مقدماتی	برنامه‌نویسی مقدماتی	برنامه‌نویسی مقدماتی
طراحی امور گرافیکی با رایانه	طراحی امور گرافیکی با رایانه	طراحی امور گرافیکی با رایانه
کاربر FreeHand	کاربر Flash	کاربر Flash
کاربر CorelDraw	طراحی صفحات وب مقدماتی	کاربر Director
	طراحی صفحات وب پیشرفته	میکس رایانه‌ای

## مجموعه کتاب‌های درسی رشته کامپیوتر شاخه کاردانش

(استاندارد وزارت کار و امور اجتماعی)

رشته تصویرسازی	رشته طراحی صفحات وب	رشته تولید چندرسانه‌ای	رشته برنامه‌نویسی پایگاه داده
مفاهیم پایه فناوری اطلاعات	مفاهیم پایه فناوری اطلاعات	مفاهیم پایه فناوری اطلاعات	مفاهیم پایه فناوری اطلاعات
سیستم‌عامل مقدماتی	سیستم‌عامل مقدماتی	سیستم‌عامل مقدماتی	سیستم‌عامل مقدماتی
اطلاعات و ارتباطات	اطلاعات و ارتباطات	اطلاعات و ارتباطات	اطلاعات و ارتباطات
سیستم‌عامل پیشرفته	سیستم‌عامل پیشرفته	سیستم‌عامل پیشرفته	سیستم‌عامل پیشرفته
واژه‌پرداز Word 2007	واژه‌پرداز Word 2007	واژه‌پرداز Word 2007	واژه‌پرداز Word 2007
صفحه گسترده Excel 2007	صفحه گسترده Excel 2007	صفحه گسترده Excel 2007	صفحه گسترده Excel 2007
ارایه مطالب PowerPoint 2007	ارایه مطالب PowerPoint 2007	ارایه مطالب PowerPoint 2007	ارایه مطالب PowerPoint 2007
نرم‌افزارهای اداری تکمیلی	نرم‌افزارهای اداری تکمیلی	نرم‌افزارهای اداری تکمیلی	نرم‌افزارهای اداری تکمیلی
طراح گرافیک رایانه‌ای	طراح گرافیک رایانه‌ای	طراح گرافیک رایانه‌ای	برنامه‌نویسی مقدماتی
شهروند الکترونیکی	نرم‌افزار گرافیکی Flash Mx	شهروند الکترونیکی	برنامه‌نویسی ویژوال بیسیک پیشرفته (جلد اول)
نرم‌افزار گرافیکی FreeHand	طراحی مقدماتی صفحات وب	نرم‌افزار گرافیکی Director	برنامه‌نویسی ویژوال بیسیک پیشرفته (جلد دوم)
نرم‌افزار گرافیکی CorelDraw	رایانه کار Interdev	تدوین فیلم و صدا SSP	مدیریت پایگاه داده
نرم‌افزار گرافیکی Flash Mx	رایانه کار Dreamweaver	نرم‌افزار گرافیکی Flash Mx	مهارت عمومی برنامه‌نویسی
	رایانه کار CIW	نرم‌افزار گرافیکی Authorware	

# فهرست مطالب

مقدمه

## واحد کار اول: نصب و راه اندازی MySQL و کار با آن

۲	۱-۱ آشنایی با MySQL
۴	۱-۲ نصب MySQL
۱۲	واژه نامه
۱۲	خلاصه مطالب
۱۳	آزمون نظری
۱۳	آزمون عملی

## واحد کار دوم: پایگاه داده

۱۶	مقدمه
۱۶	۲-۱ مدل های پایگاه داده
۲۳	۲-۲ انواع ماهیت ارتباط
۲۴	۲-۳ محیط عملیاتی تولیدکنندگان قطعات صنعتی
۲۷	۲-۴ نرمال سازی بانک های اطلاعاتی
۳۴	واژه نامه
۳۴	خلاصه مطالب
۳۶	آزمون نظری

## واحد کار سوم: آشنایی با دستورات کار با پایگاه داده

۳۸	مقدمه
۳۸	۳-۱ آشنایی با SQL
۴۰	۳-۲ دسترسی به MySQL از طریق خط فرمان
۴۱	۳-۳ دستور create
۴۵	۳-۴ دستور select
۴۷	۳-۵ دستور where
۴۹	۳-۶ دستور insert
۵۱	۳-۷ دستور update

۵۳	۳-۸ دستور delete
۵۴	۳-۹ دستور order by
۵۶	۳-۱۰ دستور and و or
۵۸	۳-۱۱ دستور in
۵۸	۳-۱۲ دستور between ... and
۶۰	۳-۱۳ ویژگی alias
۶۱	۳-۱۴ پیوند جدول‌ها
۶۷	۳-۱۵ دستور join
۶۸	۳-۱۶ Select های تودرتو
۷۰	۳-۱۷ دستور union
۷۲	۳-۱۸ دستور drop
۷۳	۳-۱۹ دستور alter
۷۴	۳-۲۰ توابع درون ساخته SQL
۸۵	واژه‌نامه
۸۵	خلاصه مطالب
۸۶	آزمون نظری
۸۸	آزمون عملی

## واحد کار چهارم: نصب APACHE

۹۰	مقدمه
۹۰	۴-۱ نصب Apache
۹۶	واژه‌نامه
۹۶	خلاصه مطالب
۹۷	آزمون نظری
۹۸	آزمون عملی

## واحد کار پنجم: نصب PHP

۱۰۰	مقدمه
۱۰۱	۱-۵ آشنایی با زبان PHP
۱۰۷	۲-۵ نصب PHP در لینوکس
۱۰۸	۳-۵ فایل php.ini در ویندوز
۱۰۹	۴-۵ نصب PHP بدون استفاده از سرویس‌دهنده‌ها
۱۱۲	۵-۵ پیکربندی Apache در ویندوز
۱۱۲	۶-۵ پیکربندی Apache در لینوکس
۱۱۲	۷-۵ بررسی صحت عملکرد PHP
۱۱۴	واژه‌نامه
۱۱۴	خلاصه مطالب
۱۱۵	آزمون نظری

## واحد کار ششم: کار با متغیرها، عملگرها و ساختارهای کنترلی

۱۱۸	مقدمه
۱۱۸	۱-۶ شروع برنامه‌نویسی در PHP
۱۲۲	۲-۶ متغیرها
۱۲۷	۳-۶ عملگرها
۱۳۰	۴-۶ کار با ساختارهای کنترلی
۱۳۵	۵-۶ حلقه‌ها
۱۳۹	۶-۶ دستورات break و continue
۱۴۰	۷-۶ متغیرهای سراسری
۱۴۱	واژه‌نامه
۱۴۱	خلاصه مطالب
۱۴۳	آزمون نظری
۱۴۴	آزمون عملی

## واحد کار هفتم: برقراری ارتباط با کاربر

۱۴۶	مقدمه
۱۴۶	۷-۱ دریافت اطلاعات از فرم
۱۴۹	۷-۲ اعتبارسنجی داده‌های ورودی
۱۵۰	۷-۳ ارسال نامه الکترونیکی از طریق PHP
۱۵۳	۷-۴ بارگذاری (Upload) فایل‌ها از طریق فرم
۱۶۰	۷-۵ طراحی فرم بازخورد (Feedback)
۱۶۲	واژه‌نامه
۱۶۲	خلاصه مطالب
۱۶۳	آزمون نظری
۱۶۳	آزمون عملی

## واحد کار هشتم: مدیریت فایل‌ها و پوشه‌ها

۱۶۶	مقدمه
۱۶۶	۸-۱ کار با فایل‌ها
۱۶۸	۸-۲ کار با پوشه‌ها
۱۷۰	واژه‌نامه
۱۷۰	خلاصه مطالب
۱۷۱	آزمون نظری
۱۷۱	آزمون عملی

## واحد کار نهم: کار با توابع

۱۷۴	مقدمه
۱۷۴	۹-۱ توابع رشته‌ای
۱۷۸	۹-۲ توابع تاریخ و زمان
۱۸۱	۹-۳ توابع عددی
۱۸۴	۹-۴ تعریف توابع جدید



۱۸۸	واژه‌نامه
۱۸۸	خلاصه مطالب
۱۸۹	آزمون نظری
۱۹۰	آزمون عملی

## واحد کار دهم: کار با پایگاه داده

۱۹۲	مقدمه
۱۹۲	۱۰-۱ مجوزهای کاربری
۱۹۶	۱۰-۲ لیست پایگاه‌های داده
۱۹۸	۱۰-۳ مدیریت پایگاه داده
۱۹۹	۱۰-۴ ساخت جدول phpMyAdmin
۲۰۱	۱۰-۵ ارتباط با پایگاه داده در Dreamweaver
۲۰۶	۱۰-۶ مشاهده اطلاعات پایگاه داده
۲۰۸	۱۰-۷ درج اطلاعات در پایگاه داده
۲۰۹	۱۰-۸ حذف رکورد از پایگاه داده
۲۱۳	۱۰-۹ ویرایش اطلاعات پایگاه داده
۲۱۶	مطالعه آزاد
۲۱۶	۱۰-۱۰ ایجاد ارتباط با سیستم مدیریت پایگاه داده MySQL
۲۱۸	۱۰-۱۱ ایجاد یک پایگاه داده و اجزای آن
۲۲۵	۱۰-۱۲ انتخاب و نمایش اطلاعات جدول
۲۲۸	۱۰-۱۳ به روزرسانی اطلاعات جدول
۲۲۹	۱۰-۱۴ حذف رکورد
۲۳۱	واژه‌نامه
۲۳۱	خلاصه مطالب
۲۳۲	آزمون نظری
۲۳۳	آزمون عملی

## واحد کار یازدهم: به‌کارگیری کوکی‌ها و Session‌ها

۲۳۶	مقدمه
۲۳۷	۱-۱۱ ایجاد کوکی
۲۴۱	۲-۱۱ جلسه Session
۲۴۴	واژه‌نامه
۲۴۴	خلاصه مطالب
۲۴۵	آزمون نظری
۲۴۶	آزمون عملی

## واحد کار دوازدهم: ارزیابی وب‌سایت و صفحات وب

۲۴۸	مقدمه
۲۴۸	۱-۱۲ ارزیابی صفحات وب
۲۵۳	واژه‌نامه
۲۵۳	خلاصه مطالب
۲۵۴	آزمون نظری
۲۵۴	آزمون عملی
۲۵۵	پیوست
۲۹۰	پاسخنامه
۲۹۲	فهرست منابع

## مقدمه

امروزه با پیشرفت علم و فناوری، افزایش سطح دانش و مهارت‌های ارتباطی به عنوان دو جزء لاینفک از یکدیگر قلمداد می‌شوند، در راستای ایجاد ارتباط در عصر اطلاعات، اینترنت مناسب‌ترین و در عین حال کم‌هزینه‌ترین راه ایجاد ارتباط تعاملی به شمار می‌رود، وب به عنوان اصلی‌ترین رابط اینترنت به منظور تبادل اطلاعات در دنیای مجازی به کار می‌رود.

لذا برای بهره‌گیری مؤثر از اینترنت و ارتباط با آن بهتر است خود را به علم طراحی صفحات وب و وبسایت مجهز نماییم.

در این کتاب سعی شده است مراحل طراحی یک وبسایت به طور خودآموز و با بیان بسیار ساده آموزش داده شود، اما مسلماً از اشتباهات احتمالی مبرا نخواهد بود، لذا از خوانندگان گرامی این کتاب خواهشمندم اینجانب را از نظرات ارزشمند خود بهره‌مند نمایند تا در چاپ بعدی کتاب نسبت به اصلاح آن اقدام شود.

مؤلف

## نصب و راه‌اندازی MySQL و کار با آن

### واحد کار اول

پس از مطالعه این واحد کار از فراگیر انتظار می‌رود که:

- ۱ - با MySQL آشنا باشد و نگارش‌های مختلف آن را بشناسد.
- ۲ - بتواند MySQL را با نرم‌افزارهای مشابه مقایسه کند.
- ۳ - امکانات لازم برای نصب و استفاده از MySQL را بشناسد.
- ۴ - با اصول نصب MySQL در ویندوز آشنا باشد.
- ۵ - با اصول نصب MySQL در لینوکس آشنا باشد.
- ۶ - ساختار پوشه‌های MySQL را بشناسد.
- ۷ - فایل‌های ذخیره‌سازی MySQL را بشناسد.
- ۸ - پوشه داده‌ها در MySQL را بشناسد.

زمان (ساعت)	
عملی	نظری
۸	۴

## ۱ - ۱ آشنایی با MySQL

MySQL یکی از سیستم‌های مدیریت پایگاه داده متن باز<sup>۱</sup> است که برای ایجاد و مدیریت بانک‌های اطلاعاتی در سطح وسیعی استفاده می‌شود. پایگاه داده، شامل مجموعه‌ای از داده‌هاست که به صورت ساخت یافته و در قالب جداول در کنار یکدیگر قرار گرفته‌اند، در پایگاه داده برای عملیاتی مانند حذف، اضافه و ویرایش داده‌ها باید از یک سیستم مدیریت پایگاه داده‌ها مانند MySQL استفاده کرد. اگر حجم اطلاعات و داده‌ها در پایگاه داده زیاد باشد، عملیات پردازشی و محاسباتی نقش زیادی در انتخاب پایگاه داده دارند، برخی از پایگاه‌های داده مانند Access نسبت به نمونه‌های مشابه سرعت کمی در عملیات پردازشی و محاسباتی دارد، لذا به کار بردن آن در بانک‌های اطلاعاتی با حجم بالا توصیه نمی‌شود، اما بانک اطلاعاتی Oracle به علت سرعت بسیار بالا غالباً در بانک‌های اطلاعاتی بسیار بزرگ استفاده می‌شود، بانک اطلاعاتی MySQL نیز در بانک‌های اطلاعاتی نسبتاً بزرگ و تا حجم چند میلیون داده عملکرد مناسبی ارائه می‌دهد، به همین دلیل در بین طراحان وب از محبوبیت بالایی برخوردار بوده و به دلیل راحت بودن محیط طراحی بانک اطلاعاتی و مدیریت آن همچنان مورد توجه طراحان بانک اطلاعاتی است.

### ۱-۱-۱ ویژگی‌های اصلی MySQL

- یکی از ویژگی‌های اصلی MySQL این است که به صورت متن باز بوده و آن را می‌توان به طور رایگان از اینترنت دریافت کرد.
- این سیستم مدیریت پایگاه داده می‌تواند در محیط‌های سیستم‌های عامل مختلف از جمله ویندوز و لینوکس به خوبی کار کند و به دلیل وجود API‌های (Application Programming Interface) متعدد، می‌تواند توسط زبان‌های برنامه‌نویسی مختلف از جمله Perl، PHP، java، C++ و C استفاده شود.
- MySQL می‌تواند در سیستم‌هایی که چند پردازنده دارند استفاده شود و حتی می‌تواند از چندین CPU برای اجرا استفاده کند.

- به دلیل استفاده از تکنیک‌های سریع دسترسی به حافظه می‌تواند با سرعت بالایی اطلاعات مورد نیاز کاربران را در اختیار آن‌ها قرار دهد.
- از امنیت نسبتاً بالایی برخوردار بوده و می‌تواند تا حد زیادی مانع دسترسی هکرها به اطلاعات درون بانک اطلاعاتی شود.
- نسخه‌های قبل از MySQL 4.1.3 از زبان فارسی پشتیبانی نمی‌کردند و برای ذخیره، نمایش و مرتب‌سازی داده‌های فارسی از قالب مخصوص زبان عربی استفاده می‌شد، به همین دلیل برای نمایش داده‌های فارسی که دارای حروف "گ، چ، پ، ژ" بودند مشکل پیش می‌آمد اما در نسخه‌های پس از آن، امکانات فارسی با استاندارد UTF8 به MySQL وارد شده و مشکلات زبان فارسی نیز حل شد

## ۲-۱-۱ نسخه‌های MySQL

- MySQL دارای نسخه‌های متعددی است که تفاوت‌هایی نیز با یکدیگر دارند، به‌طور کلی گروهی از نسخه‌ها فقط برای توسعه استفاده می‌شوند و گروهی دیگر می‌تواند به عنوان یک سیستم مدیریت پایگاه داده دائمی برای طراحی سایت به کار روند.
- اگر به عنوان نسخه‌های MySQL دقت کنید می‌بینید که در نامگذاری آن‌ها یک الگوی مشخص رعایت شده است، به عنوان مثال MySQL 5.1.25.
- عدد اول (5):** نسخه اصلی و توصیف‌کننده قالب فایل است. این به آن معناست که تمام نسخه‌های MySQL... دارای یک توصیف‌کننده قالب فایل یکسان هستند و الزاماً اینگونه نیست که MySQL4... قدیمی‌تر از MySQL5... باشد.
- عدد دوم (1):** میزان انتشار را معین می‌کند، شماره اول و دوم با همدیگر شماره سری انتشار را تشکیل می‌دهند.
- عدد سوم (25):** شماره ویرایش در سری انتشار است. یعنی هر بار که ویرایشی روی هر نسخه MySQL صورت می‌گیرد، به این شماره یکی افزوده می‌شود.
- پس از گروه اعداد فوق از یک پسوند هم در نامگذاری نسخه‌های MySQL استفاده می‌شود که عبارتند از:
- alpha:** نسخه‌ای که از این نوع باشد، نشان‌دهنده این است که به آن کدهای جدیدی افزوده شده است اما هنوز به‌طور کامل عیب‌یابی نشده‌اند و هنوز در دوران تست و عیب‌یابی به سر می‌برند.

**beta:** از لحاظ خصوصیات و ویژگی‌ها با نسخه alpha از همان نرم‌افزار تفاوت ندارد و فقط دوران تست را پشت سر گذاشته و به عنوان نسخه نهایی عرضه شده است.  
**gama:** همان نسخه alpha است که به منظور صحت عملکرد بررسی و تعمیر شده است.

## ۲-۱ نصب MySQL

برای نصب و راه‌اندازی MySQL نیاز به وجود قطعات سخت‌افزاری پیشرفته نیست و این سیستم مدیریت پایگاه داده می‌تواند روی تمام سخت‌افزارهای امروزی نصب شده و به راحتی مورد استفاده قرار بگیرد.

### ۱-۲-۱ نصب در لینوکس

برای نصب MySQL در لینوکس بهتر است از بسته‌های RPM استفاده کنید، این بسته‌ها برای پایگاه‌های داده طراحی شده‌اند و فقط روی برخی از نسخه‌های لینوکس قابل اجرا هستند. در حال حاضر برای برخی نسخه‌های لینوکس از جمله Suse7.3 بهتر کار می‌کند. هنگام نصب MySQL در لینوکس باید یک گروه کاری و یک کاربر جدید برای کار با MySQL ایجاد کنید.

### ۲-۲-۲ نصب در ویندوز

اگر قبلاً روی سیستم خود MySQL را نصب نکرده‌اید مراحل زیر را برای نصب آن انجام دهید:

در مسیر فایل‌های نصب MySQL روی آیکن مربوط به نصب دو بار کلیک کنید، پس از مدت کوتاهی کادر محاوره شکل ۱-۱ ظاهر می‌شود.  
روی دکمه Next کلیک کنید.

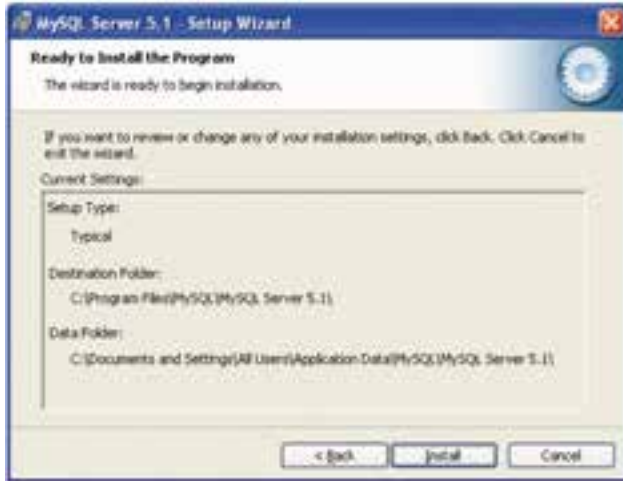


شکل ۱-۱



شکل ۱-۲

دکمه رادیویی Typical را انتخاب کرده، روی دکمه Next کلیک کنید.



شکل ۱-۳

در کادر محاوره شکل ۱-۳ روی دکمه Install کلیک کنید.

در کادر محاوره شکل ۱-۴ روی دکمه Next کلیک کنید.

در کادر محاوره شکل ۱-۵ دقت کنید که حتماً کادر علامت

Configure the MySQL Server now انتخاب شده باشد.

روی دکمه Finish کلیک کنید.





شکل ۱-۴

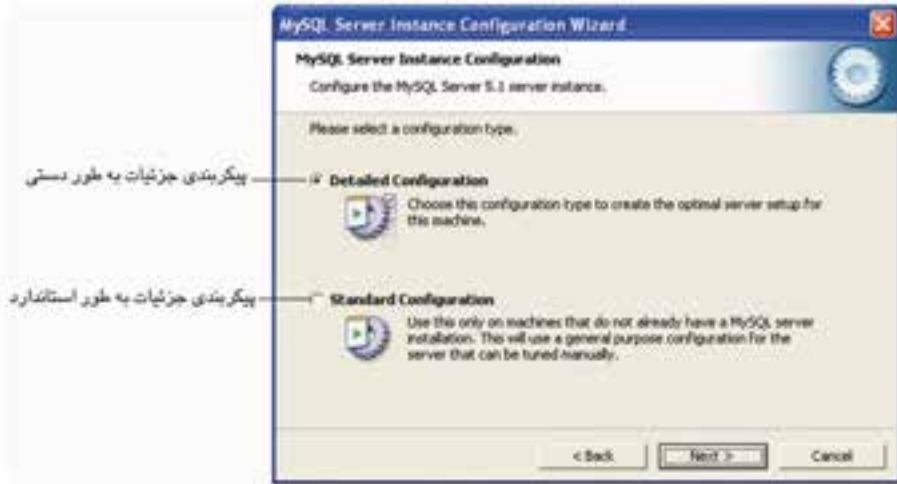


شکل ۱-۵



شکل ۱-۶

در شکل ۶-۱ روی دکمه Next کلیک کنید.  
دقت کنید که دکمه رادیویی Detailed Configuration انتخاب شده باشد، سپس روی دکمه Next کلیک کنید.

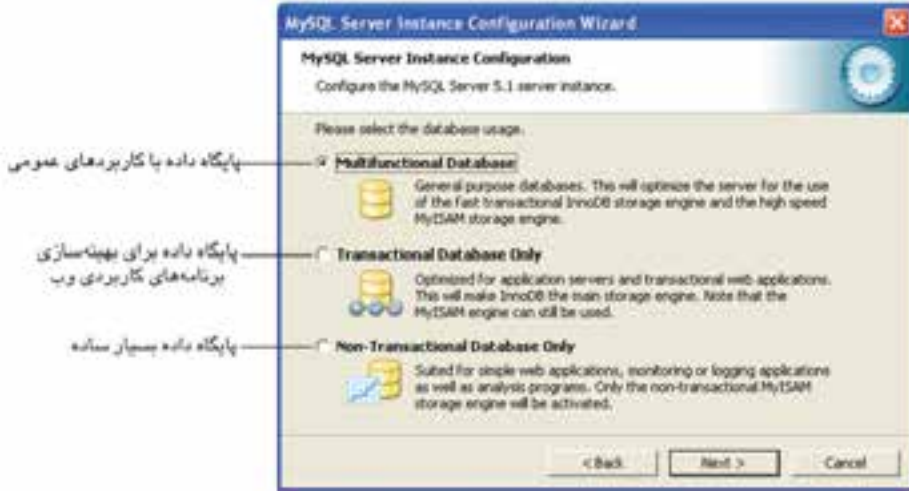


شکل ۷-۱

پس از انتخاب دکمه رادیویی Developer Machine در شکل ۸-۱، روی دکمه Next کلیک کنید.  
در کادرمحاوره شکل ۹-۱ دکمه رادیویی Multifunctional Database را انتخاب کرده، روی دکمه Next کلیک کنید.



شکل ۸-۱

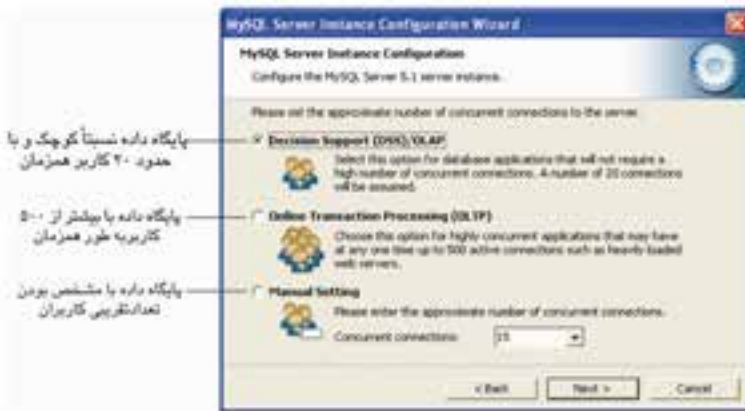


شکل ۹-۱



شکل ۱۰-۱

در کادر محاوره شکل ۱۰-۱ روی دکمه Next کلیک کنید.  
 دقت کنید دکمه رادیویی Decision Support (DSS)/OLAP در شکل ۱۱-۱ انتخاب شده باشد، سپس روی دکمه Next کلیک کنید.  
 در شکل ۱۲-۱ در حالتی که هر دو کادر علامت انتخاب شده باشند، روی دکمه Next کلیک کنید.



شکل ۱-۱۱



شکل ۱-۱۲



شکل ۱-۱۳

در شکل ۱۳-۱ برای پشتیبانی از حداکثر زبان‌ها، دکمه رادیویی Best Support For Multilingualism را انتخاب کرده، سپس روی دکمه Next کلیک کنید.



شکل ۱۴-۱

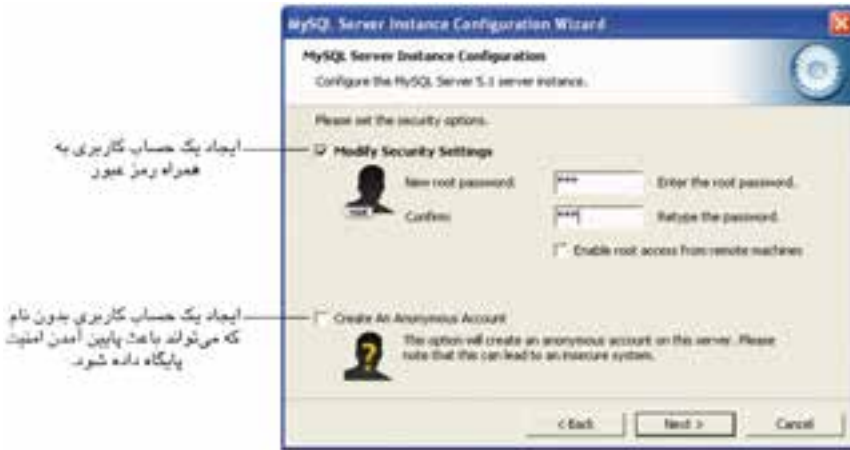
در شکل ۱۴-۱ هر سه کادر علامت موجود را انتخاب کرده، سپس روی دکمه Next کلیک کنید.

در شکل ۱۵-۱ دقت کنید کادر علامت Modify Security Settings در حالت انتخاب باشد، سپس در هر دو کادر متنی زیر آن، کلمه رمز را وارد کنید.



شکل ۱۵-۱

روی دکمه Next کلیک کنید.



شکل ۱-۱۶

در کادرمحاوره شکل ۱-۱۶ پس از تعیین رمز عبور، روی دکمه Next کلیک کنید.



شکل ۱-۱۷

و در نهایت در کادرمحاوره شکل ۱-۱۷ روی دکمه Finish کلیک کنید.

## واژه‌نامه

Configure	پیکربندی
Details	جزئیات
Developer	توسعه‌دهنده
Interface	رابط کاربری
Multifunctional	چندکاره
Typical	نمونه

## خلاصه مطالب

- MySQL یکی از سیستم‌های مدیریت پایگاه داده متن باز است که برای ایجاد و مدیریت بانک‌های اطلاعاتی استفاده می‌شود.
- در انتخاب پایگاه داده، حجم اطلاعات و داده‌ها مستقیماً دخالت دارد.
- عنوان نسخه‌های MySQL براساس یک الگوی مشخص تعیین می‌شود.
- عدد اول، نسخه اصلی و توصیف‌کننده قالب فایل است.
- عدد دوم، سطح انتشار را تعیین می‌کند.
- عدد سوم، شماره ویرایش در سری انتشار است.

## آزمون نظری

- ۱ - کدام یک از گزینه‌های زیر در انتخاب پایگاه داده مؤثرتر است؟
- الف- متن باز بودن  
ب- داشتن امکان نصب آسان  
ج- وجود API  
د- عملیات پردازشی و محاسباتی
- ۲ - MySQL به چه دلیل توسط زبان‌های برنامه‌نویسی می‌تواند مورد استفاده قرار گیرد؟
- الف- متن باز بودن  
ب- وجود APIهای متعدد  
ج- عملیات پردازشی و محاسباتی  
د- امکان نصب سریع
- ۳ - در MySQL 4.1.5، عدد ۴ نشان‌دهنده چیست؟
- الف- سطح انتشار  
ب- شماره ویرایش  
ج- زمان انتشار  
د- توصیف‌کننده قالب فایل
- ۴ - روی کدام یک از انواع نسخه‌های زیر از MySQL علاوه بر تست، تعمیر نیز صورت گرفته است؟
- الف - alpha      ب- beta      ج- gama      د- 5

## آزمون عملی

- ۱ - MySQL را روی سیستم خود نصب و راه‌اندازی کنید.





## پایگاه داده

### واحد کار دوم

- پس از مطالعه این واحد کار از فراگیر انتظار می‌رود که:
- ۱ - انواع مدل‌های پایگاه داده را بشناسد.
  - ۲ - مفهوم پایگاه‌های داده رابطه‌ای را بداند.
  - ۳ - انواع رابطه‌ها را بشناسد.
  - ۴ - اصول طراحی پایگاه داده را بداند.
  - ۵ - با مفهوم نرمال‌سازی آشنا باشد.
  - ۶ - انواع مدل‌های نرمال اول، دوم و سوم را بشناسد.

زمان (ساعت)	
عملی	نظری
۵	۸

## مقدمه

با مفاهیم کلی بانک اطلاعاتی، موجودیت، رکورد، فیلد و ... در درس بانک اطلاعاتی سال گذشته آشنا شده‌اید.


در این درس ضمن یادآوری مباحث گذشته، با مفاهیم پیشرفته‌تر بانک اطلاعاتی نیز آشنا خواهید شد. در حقیقت سیستم بانک اطلاعاتی یا پایگاه داده‌ها، سیستم کامپیوتری نگهداری اطلاعات است. پایگاه داده را می‌توان به یک قفسه الکترونیکی تشبیه کرد که اطلاعات را به‌طور منظم درون خود نگهداری می‌کند.

کاربران سیستم بانک اطلاعاتی می‌توانند اعمال مختلفی روی آن انجام دهند که نمونه‌هایی از این اعمال عبارتند از:

- افزودن فایل‌های جدید و خالی به پایگاه داده
- افزودن داده درون فایل‌هایی که از قبل ایجاد شده‌اند.
- بازیابی داده‌ها از فایل‌های موجود
- تغییر در داده‌های فایل‌های موجود
- حذف داده‌ها از فایل‌های موجود
- حذف فایل‌های پایگاه داده

## ۱-۲ مدل‌های پایگاه داده

مدل‌های پایگاه داده‌ها یک طرح و الگوی کاری برای کاربران پایگاه داده‌ها را در سطح منطقی مشخص می‌کنند و معمولاً سیستم‌های مدیریت پایگاه داده براساس مدل‌هایی که استفاده می‌کنند تقسیم‌بندی می‌شوند.

**نکته**  به برنامه کامپیوتری که برای مدیریت داده‌ها و پاسخگویی به کاربران از طریق پایگاه داده به‌کار برده می‌شود سیستم مدیریت پایگاه داده (DBMS) گفته می‌شود.

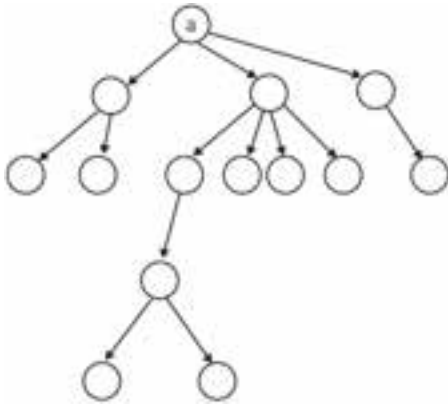
مهم‌ترین انواع مدل‌های پایگاه داده عبارتند از:

- پایگاه داده سلسله مراتبی
- پایگاه داده رابطه‌ای

- پایگاه داده شبکه‌ای
- پایگاه داده شیء‌گرا

### ۲-۱-۱ مدل سلسله مراتبی

این ساختار یکی از قدیمی‌ترین مدل‌های طراحی بانک اطلاعاتی است. در این مدل



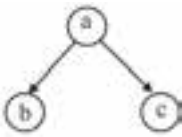
شکل ۲-۱

داده‌ها و ارتباط بین آن‌ها توسط اشاره‌گرهایی با یکدیگر مرتبط می‌شوند. شکل ۲-۱ یک نمونه مدل سلسله مراتبی است. مدل سلسله مراتبی که مشابه ساختار درخت پیاده می‌شود، دارای یک گره ریشه  $a$  و  $N$  گره فرزند است.

ارتباط بین گره‌ها از بالا به پایین صورت می‌گیرد و مسیری از گره‌های پایین‌تر به گره‌های بالاتر وجود ندارد.

هر گره پدری می‌تواند چند گره فرزند داشته باشد، اما هر گره فرزند، تنها یک گره

پدر دارد.



شکل ۲-۲

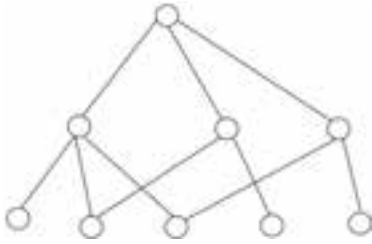
در شکل ۲-۲،  $a$  گره پدر و  $b$  و  $c$  گره‌های فرزند هستند.

در عمل پیاده‌سازی ساختار سلسله مراتبی بسیار مشکل است، به همین دلیل پس از مدتی جای خود را به مدل‌های دیگر پایگاه داده داد و عملاً منسوخ شده است.

### ۲-۱-۲ پایگاه داده شبکه‌ای

این مدل در سال ۱۹۶۹ و براساس دو مفهوم مهم مجموعه‌ها و رکوردها ارائه شد.

پیاده‌سازی این مدل توسط گراف انجام می‌شود.



شکل ۲-۳

مزیت این روش این است که مدل‌های ارتباطی طبیعی بیشتری را بین موجودیت‌ها فراهم می‌کند و در عین حال پیچیدگی کمتری خواهد داشت.

اگر بخواهیم این مدل را توسط ساختار درختی پیاده کنیم، می‌توانیم بگوییم که هر گره فرزند می‌تواند بیش از یک گره پدر داشته باشد.

این مدل پیچیده‌تر از مدل سلسله مراتبی است و در نتیجه عملیات ذخیره و بازیابی پیچیده‌تری نیز دارد.

### ۲-۱-۳ مدل رابطه‌ای

پایگاه داده رابطه‌ای از دید کاربران شامل یک مجموعه جدول است که هر جدول دارای سطرها و ستون‌هایی است. اطلاعات مربوط به هر ستون، فیلدهای جدول و اطلاعات هر سطر مربوط به یکی از رکوردهای جدول است.

جدول ۲-۱

شماره دانش‌آموزی	نام خانوادگی	نام
۱۳۴۵۷۹۳۲	اقبالی	علی
۳۲۴۳۳۵۹۲	احمدی	مریم
۲۴۱۲۳۷۱۴	بهرامی	میثم
۱۲۳۹۲۳۴۳	فدایی	زهرا

در جدول ۲-۱، سه فیلد با عناوین نام، نام خانوادگی و شماره دانش‌آموزی وجود دارد و همچنین هر سطر اطلاعات مربوط به یکی از رکوردهای جدول را نمایش داده است. **فیلد:** کوچک‌ترین واحد داده ذخیره شده در بانک اطلاعاتی است. جدول ۲-۱ شامل سه فیلد ایجاد شده با عناوین «نام»، «نام خانوادگی» و «شماره دانش‌آموزی» است.

**رکورد:** مجموعه‌ای از فیلدهای ذخیره شده مرتبط به هم است. به عنوان مثال در جدول ۲-۱ نام خانوادگی فدایی مربوط به فردی با نام زهرا و شماره دانش‌آموزی ۱۲۳۹۲۳۴۳ است و چنانچه این فرد از جدول حذف شود، تمام ویژگی‌های مربوط به او نیز از جدول حذف

می‌شوند و تغییر در مشخصات این فرد تأثیری در مشخصات افراد دیگر نخواهد داشت لذا هر سطر از جدول ۱-۲ به عنوان یک رکورد در نظر گرفته می‌شود.

مدل‌های رابطه‌ای چند ویژگی مهم دارند:

- ردیف‌های تکراری در آن‌ها وجود ندارد.
- ترتیب ردیف‌ها اهمیتی در ساختار مدل ندارد.
- ترتیب ستون‌ها اهمیتی در ساختار مدل ندارد.
- مقادیر هر فیلد غیرقابل تجزیه است.

### مفهوم کلید

• کلید در مدل رابطه‌ای، صفت یا ستونی است که برای هر کدام از رکوردها (سطرها) مقدار منحصر به فردی دارد. به عنوان مثال در جدول ۱-۲، ستون شماره دانش‌آموزی می‌تواند به عنوان فیلد کلید در نظر گرفته شود زیرا یقین داریم که برای هر رکورد منحصر به فرد است و مقدار این فیلد برای دو رکورد متفاوت از جدول یکسان نخواهد بود.

### محیط عملیاتی

سازمان، مؤسسه یا نهادی را در نظر بگیرید که قصد داریم برای آن یک بانک اطلاعاتی ایجاد کنیم. در این صورت به این سازمان، مؤسسه یا نهاد، محیط عملیاتی گفته می‌شود. مثال‌های زیر نمونه‌هایی از محیط عملیاتی هستند:

۴ - یک اداره یا سازمان دولتی

۵ - یک بانک

۶ - یک کتابخانه

۱ - یک مدرسه

۲ - یک دانشگاه

۳ - یک بیمارستان



تمرین:

چند محیط عملیاتی نام ببرید.

در هر محیط عملیاتی با توجه به نیاز، تعدادی از اجزای آن به عنوان موجودیت انتخاب می‌شوند، به عنوان مثال در محیط مدرسه موجودیت‌هایی با عنوان دانش‌آموز، معلم و درس داریم. در محیط عملیاتی بیمارستان موجودیت‌های بیمار، پزشک، اتاق، بخش و ... وجود دارد.

### تمرین:

چه موجودیت‌هایی برای محیط‌های عملیاتی بانک و کتابخانه می‌توان انتخاب کرد؟

محیط عملیاتی مدرسه را در نظر بگیرید، هر کدام از موجودیت‌های محیط عملیاتی ویژگی‌ها و صفات متعددی دارند که فقط برخی از آن‌ها را می‌توان به عنوان فیلد در نظر گرفت، به عنوان مثال یک دانش‌آموز می‌تواند صفات و ویژگی‌های زیر را داشته باشد:

- نام
- نام خانوادگی
- نام پدر
- نام مادر
- رنگ چشم
- قد
- وزن
- تاریخ تولد
- و ...

اما همان‌طور که مشاهده می‌کنید تمام این صفات در محیط عملیاتی مدرسه کاربردی نیستند، به عنوان مثال، رنگ چشم، قد، وزن، نام مادر و ... در محیط عملیاتی مدرسه به کار نمی‌روند. علاوه بر این در یک باشگاه ورزشی، یک بیمارستان یا هر محیط عملیاتی دیگر، با توجه به نیازها، صفات و ویژگی‌های متفاوتی در نظر گرفته می‌شوند.

به‌طور کلی در محیط عملیاتی مدرسه می‌توان ویژگی‌های زیر را برای هر کدام از موجودیت‌ها در نظر گرفت.

**ویژگی‌های موجودیت دانش‌آموز**

۱ - نام (name)

۲ - نام خانوادگی (family)

۳ - رشته تحصیلی (major)

۴ - تاریخ تولد (date)

۵ - محل تولد (area)

۶ - نمره (grade)

### ویژگی‌های موجودیت معلم

۱ - نام (tname)

۲ - نام خانوادگی (tfamily)

۳ - آدرس (taddress)

۴ - تلفن (ttel)

### ویژگی‌های موجودیت درس

۱ - کد درس (cnumber)

۲ - نام درس (cname)

۳ - تعداد ساعات در طول هفته (hcourse)

هر کدام از این موجودیت‌ها، یک جدول در محیط عملیاتی مدرسه و هر کدام از ویژگی‌ها و صفات موجودیت‌ها، یک فیلد در جدول‌های بانک اطلاعاتی خواهند بود. همان‌طور که پیش از این اشاره شد در مدل‌های رابطه‌ای نباید ردیف‌های تکراری وجود داشته باشد، برای روشن شدن مطلب کلاس خود را در نظر بگیرید، معلم شما برای صدا کردن دانش‌آموزان از کدام صفت دانش‌آموزان استفاده می‌کند، ممکن است پاسخ دهید نام خانوادگی، استفاده از نام خانوادگی برای صدا کردن دانش‌آموزان به این دلیل است که احتمال تکراری بودن کمتری دارد، تا اینجا می‌توان گفت نام خانوادگی به عنوان فیلد کلید در نظر گرفته شده است، اما اگر حتی یک مورد تکراری وجود داشته باشد و دو دانش‌آموز دارای نام خانوادگی مشابه باشند، به دلیل احتمال بروز اشتباه دیگر نمی‌توان از نام خانوادگی به تنهایی استفاده کرد، در این صورت معمولاً نام به دنبال نام خانوادگی استفاده می‌شود.

یعنی دو فیلد نام و نام خانوادگی به عنوان کلید استفاده می‌شوند.



• اگر کلید انتخاب شده شامل یک فیلد باشد، آن را فیلد کلید ساده می‌گویند.  
• اگر کلید انتخاب شده شامل دو یا چند فیلد باشد، آن را فیلد کلید مرکب می‌گویند.  
اما اگر قدری دقیق‌تر شویم، می‌بینیم که باز هم احتمال تکراری بودن نام و نام خانوادگی و این‌که دو نفر دارای نام و نام خانوادگی یکسان باشند نیز وجود دارد. اگر بخواهیم محیط عملیاتی مدرسه را طوری طراحی کنیم که احتمال تکراری بودن فیلدها را به صفر برسانیم بهتر است به جای ترکیب فیلدها یک فیلد جدید با نام شماره دانش‌آموزی به ویژگی‌های موجودیت دانش‌آموز اضافه کنیم، در این صورت اطمینان حاصل خواهد شد که هیچ دو دانش‌آموزی اطلاعات کاملاً یکسان ندارند و در نتیجه ایجاد سطر تکراری در جدول نخواهیم داشت.

برای تهیه جدول‌ها نکات زیر را در نظر بگیرید:

۱ - هر موجودیت مستقل را به عنوان یک جدول در نظر بگیرید.

• معلم

• درس

• دانش‌آموز

۲ - صفت‌های موجودیت‌ها به عنوان فیلدهای جدول‌ها در نظر گرفته می‌شوند.

### جدول مربوط به دانش‌آموز

جدول ۲-۲ فیلدهای جدول دانش‌آموز (Student)

شماره دانش‌آموزی*	نام	نام خانوادگی	تاریخ تولد	محل تولد	نمره
(id)	(name)	(family)	(date)	(area)	(grade)

### جدول مربوط به معلم

جدول ۲-۳ فیلدهای جدول معلم (teacher)

کد معلم*	نام	نام خانوادگی	آدرس	تلفن
tcode	tname	tfamily	taddress	ttel

## جدول مربوط به درس

جدول ۴-۲ فیلدهای جدول درس (course)

تعداد ساعات در طول هفته	نام درس	کد درس*
hcourse	cname	cnumber

### نکته



فیلدهایی که کنار آن‌ها علامت \* درج شده است، فیلدهای کلیدی هستند.

در هر محیط عملیاتی موجودیت‌ها با یکدیگر مرتبط هستند. در طراحی و تحلیل بانک‌های اطلاعاتی، وجود این ارتباط و بررسی ماهیت آن، یکی از مراحل اصلی کار است. در محیط عملیاتی مدرسه، هر دانش‌آموز در سال تحصیلی در چند درس ثبت نام می‌کند. به کلمات «هر» و «چند» در ابتدای موجودیت‌های دانش‌آموز و درس دقت کنید، وجود این کلمات به این معناست که یک دانش‌آموز چند درس را انتخاب کرده است. ماهیت ارتباط دانش‌آموز-درس، «یک به چند» است.

## ۲-۲ انواع ماهیت ارتباط

با توجه به توضیحات بخش محیط عملیاتی، به تناظر بین تعداد موجودیت‌ها، ماهیت ارتباط گفته می‌شود و این تناظر می‌تواند انواع زیر را داشته باشد:

- ارتباط یک به یک (۱:۱)
- ارتباط یک به چند (۱:N)
- ارتباط چند به چند (N:N)

اگر در محیط عملیاتی مدرسه موجودیت‌های مدرسه و مدیر را نیز در نظر داشته باشیم، در ارتباط یک به یک، یک نمونه از موجودیت اول با یک نمونه از موجودیت دوم ارتباط دارد، به عنوان مثال هر مدرسه یک مدیر دارد.

در ارتباط یک به چند، یک نمونه از موجودیت اول با چند نمونه از موجودیت دوم ارتباط دارد و در مقابل هر نمونه از موجودیت دوم با یک نمونه از موجودیت اول ارتباط دارد، به عنوان مثال یک مدرسه چند معلم دارد و هر معلم در یک مدرسه تدریس می‌کند.

در ارتباط چند به چند، یک نمونه از موجودیت اول با چند نمونه از موجودیت دوم ارتباط دارد و همچنین یک نمونه از موجودیت دوم با چند نمونه از موجودیت اول مرتبط است.

**مثال:** در محیط عملیاتی مدرسه با در نظر گرفتن موجودیت‌های درس، معلم و دانش‌آموز، ماهیت‌های ارتباط و تناظر بین موجودیت‌ها به صورت زیر خواهد بود:

**۱ - درس - معلم:** هر معلم چند درس ارائه می‌دهد و هر درس توسط یک معلم ارائه می‌شود لذا نوع ارتباط بین درس و معلم یک به چند خواهد بود.

**۲ - درس - دانش‌آموز:** هر دانش‌آموز در هر دوره تحصیلی چند درس را انتخاب می‌کند و یک درس توسط چند دانش‌آموز اخذ می‌شود. لذا نوع ارتباط بین درس - دانش‌آموز چند به چند است.



**تمرین:**

نوع ارتباط معلم - دانش‌آموز را از لحاظ ماهیت ارتباط بررسی کنید.

### ۱-۲-۲ تأثیر ماهیت ارتباط بر بانک اطلاعات

ماهیت ارتباط بین موجودیت‌ها می‌تواند در طراحی جدول‌ها تأثیرگذار باشد و حتی ممکن است لازم شود، در انتخاب و طراحی جدول‌ها یک بازبینی کلی انجام شود.

برای ایجاد ساختار نهایی جدول‌های بانک اطلاعاتی باید نکات زیر را در نظر گرفت:

۱ - در ارتباط‌های ۱:۱ جدول‌ها در هم ادغام می‌شوند.

۲ - در ارتباط‌های ۱:N:۱ فیلد اصلی طرف ۱ به عنوان کلید خارجی به جدول طرف N افزوده می‌شود.

۳ - در ارتباط‌های N:N:۱ یک جدول جدید ساخته می‌شود، طوری که شامل کلیدهای هر دو طرف باشد.

### ۳-۲ محیط عملیاتی تولیدکنندگان قطعات صنعتی

به منظور ارائه مثال بیشتر، یک محیط عملیاتی جدید را در نظر می‌گیریم. مثلاً محیط عملیاتی تولیدکنندگان قطعات صنعتی در شهرهای مختلف. این محیط عملیاتی دارای دو

موجودیت اصلی «قطعه» و «تولیدکننده» است.

برای هر کدام از موجودیت‌ها می‌توان صفات (فیلدهای) زیر را در نظر گرفت:  
**موجودیت قطعه (جدول S)**

- ۱ - شماره قطعه (S#) \*
- ۲ - نام قطعه (Sname)
- ۳ - تعداد (Status)
- ۴ - شهر (City)

**موجودیت تولیدکننده (جدول P)**

- ۱ - شماره تولیدکننده (P#) \*
- ۲ - نام تولیدکننده (Pname)
- ۳ - رنگ (Color)
- ۴ - وزن (Weight)
- ۵ - شهر (City)

هر قطعه ممکن است توسط چند تولیدکننده تولید شود و همچنین هر تولیدکننده، چند قطعه تولید می‌کند، در نتیجه ماهیت ارتباط این دو موجودیت چند به چند است. همان‌طور که پیش از این اشاره شد، در ارتباط‌های چند به چند، باید یک جدول جدید ساخته شود که فیلدهای آن حداقل باید شامل فیلدهای کلید اصلی دو جدول اولیه باشند. در نتیجه یک جدول جدید شامل دو موجودیت شماره قطعه و شماره تولیدکننده خواهیم داشت:

**جدول (SP)**

- ۱ - شماره قطعه (S#)

- ۲ - شماره تولیدکننده (P#)

در این محیط عملیاتی تعداد قطعه تولید شده نیز می‌تواند به جدول SP افزوده شود. (Qty) با توجه به توضیحات فوق سه جدول مربوط به این محیط عملیاتی به این صورت ترسیم می‌شوند:

**جدول تولیدکننده (P)**

جدول ۲-۵

شماره تولیدکننده (P#)	نام تولیدکننده (Sname)	تعداد خط تولید (Status)	شهر (City)
-----------------------	------------------------	-------------------------	------------

جدول قطعه (S)

جدول ۲-۶

شماره قطعه (S#)	نام قطعه (Sname)	رنگ (Color)	وزن (Weight)	شهر (City)
-----------------	------------------	-------------	--------------	------------

جدول تولیدکننده - قطعه (SP)

جدول ۲-۷

شماره قطعه (S#)	شماره تولیدکننده (P #)	تعداد قطعه تولید شده (Qty)
-----------------	------------------------	----------------------------

جداول فوق را با داده‌های نمونه زیر در نظر می‌گیریم:

جدول ۲-۸ تولیدکننده

P	P#	Pname	Status	City
	P1	احمدی	۲۰	تهران
	P2	رضایی	۱۰	کرمانشاه
	P3	سلیمی	۳۰	کرمانشاه
	P4	مهدوی	۲۰	تهران
	P5	میثمی	۳۰	شیراز

جدول ۲-۹ قطعه

S	S#	Sname	Color	Weight	City
	S1	پیچ	قرمز	۱۲	تهران
	S2	شیر	سبز	۱۷	کرمانشاه
	S3	لوله	آبی	۱۷	بوشهر
	S4	لوله	قرمز	۱۴	تهران
	S5	پیچ گوشتی	آبی	۱۲	کرمانشاه
	S6	فازمتر	قرمز	۱۹	تهران

جدول ۲-۱۰ تولیدکننده - قطعه

SP	S#	P#	Qty
	S1	P1	۳۰۰
	S1	P2	۲۰۰
	S1	P3	۴۰۰
	S1	P4	۲۰۰
	S1	P5	۱۰۰
	S1	P6	۱۰۰
	S2	P1	۳۰۰
	S2	P2	۴۰۰
	S3	P2	۲۰۰
	S4	P2	۲۰۰
	S4	P4	۳۰۰
	S4	P5	۴۰۰

برای این که با مفهوم جدول SP بیشتر آشنا شوید، سطر اول را در نظر بگیرید، مفهوم سطر اول این است که:

قطعه S1 به میزان ۳۰۰ عدد توسط تولیدکننده P1، تولید شده است، یا این که تولیدکننده P1 از قطعه S1 به میزان ۳۰۰ عدد تولید کرده است.



تمرین:

مفهوم سطرهای دوم و آخر را بیان کنید.

## ۲-۴ نرمال سازی بانک های اطلاعاتی

در محیط عملیاتی تولیدکنندگان قطعات صنعتی سه جدول S، P و SP وجود دارد، کاملاً مشخص است که در جدول های فوق فیلهای مربوط به هر کدام، کاملاً ضروری هستند و به درستی انتخاب شده اند.

جدول ۲-۱۱

S#	City	P#	Qty
S1	تهران	P1	۳۰۰
S1	تهران	P2	۲۰۰
S1	تهران	P3	۴۰۰
S1	تهران	P4	۲۰۰
S1	تهران	P5	۱۰۰
S1	تهران	P6	۱۰۰
S2	کرمانشاه	P1	۳۰۰
S2	کرمانشاه	P2	۴۰۰
S3	کرمانشاه	P2	۲۰۰
S4	تهران	P2	۲۰۰
S4	تهران	P4	۳۰۰
S4	تهران	P5	۴۰۰

فرض کنید، به جای جدول SP در بانک اطلاعاتی، جدول SCP را با چهار فیلد S#، City، P# و Qty طراحی کرده باشیم، در این صورت مقادیر نمونه ای جدول به این صورت خواهد بود:

با دقت در جدول ۲-۱۱ مشاهده می شود که داده های موجود در فیلد City کاملاً تکراری هستند و از طریق جدول های P و S نیز قابل دسترسی هستند، به این عمل تکراری بودن داده ها که موجب اشغال فضای بیشتر در پایگاه داده ها و جدول ها می شود، عمل افزونگی داده ها گفته می شود.

برای طراحی یک پایگاه داده استاندارد

باید دقت کرد که عمل افزونگی داده‌ها اتفاق نیفتد.

در طراحی یک بانک اطلاعاتی پس از تحلیل و بررسی موجودیت‌ها و در نظر گرفتن صفات هر کدام از آن‌ها لازم است جداول به شکل نرمال دربیابند، نرمال‌سازی (Normalization)، فرایندی است در رابطه با بانک‌های اطلاعاتی که با دو هدف عمده زیر انجام می‌شود:

**کاهش افزونگی اطلاعات:** به این معنی که اطلاعات فقط در یک مکان (جدول) ذخیره شوند نه این‌که چند بار در جدول‌های مختلف تکرار شوند و در تمام بانک با استفاده از روابط (Relationship) تعریف شده قابل دسترسی باشند.

**حفظ یکپارچگی اطلاعات:** به این معنی که اعمال تغییرات روی اطلاعات (نظیر ایجاد، بهنگام‌سازی و حذف) در یک مکان انجام شود و به دنبال آن آثار تغییرات در تمام بانک مشاهده گردد. برای روشن شدن مفهوم یکپارچگی به مثال زیر توجه نمایید: فرض کنید یک بانک اطلاعاتی دارای دو موجودیت کتاب و نویسنده باشد. هر یک از موجودیت‌های فوق دارای صفت‌های (Attribute) مختص به خود می‌باشند. به عنوان نمونه موجودیت «کتاب» دارای صفت «نام نویسنده» و موجودیت «نویسنده» دارای صفت‌های متعددی نظیر «نام نویسنده»، «آدرس نویسنده» و ... باشد. در صورتی که در موجودیت «کتاب» یک رکورد ایجاد نماییم بدون این‌که نام نویسنده آن را در موجودیت «نویسنده» ایجاد کرده باشیم، مسلماً یک ناهنجاری به وجود خواهد آمد.

با توجه به اهداف فوق می‌توان گفت که فرایند نرمال‌سازی از ناهنجاری‌های به وجود آمده به دلیل بروز تغییرات در بانک اطلاعاتی جلوگیری خواهد نمود. با اعمال فرایند نرمال‌سازی، یک بانک اطلاعاتی کارا و مطمئن خواهیم داشت. فرایند نرمال‌سازی، فرم‌های متفاوتی دارد که انواع متداول آن به شرح ذیل می‌باشند:

- فرم اول نرمال (NF۱)
- فرم دوم نرمال (NF۲)
- فرم سوم نرمال (NF۳)

### ۱-۴-۲ فرم اول نرمال (NF۱)

موجودیت یا جدولی در فرم اول نرمال است که هیچ‌کدام از فیلدهای آن قابل تقسیم نباشد،

به عنوان مثال در محیط عملیاتی مدرسه موجودیت دانش‌آموز را در نظر بگیرید، چنانچه جدول مربوط به آن را به صورت زیر طراحی کرده باشید:

جدول ۱۲-۲

نام و نام خانوادگی	نام پدر	تاریخ تولد	محل تولد	معدل
--------------------	---------	------------	----------	------

حال جدول فوق با مقادیر نمونه‌ای به صورت زیر خواهد بود:

جدول ۱۳-۲

نام و نام خانوادگی	نام پدر	تاریخ تولد	محل تولد	معدل
مینا رضایی	احمد	۷۵/۳/۱	تهران	۱۹/۳۰
مریم توکلی	محمد رضا	۷۴/۷/۲	تهران	۱۹/۰۲
سارا بهرامی	مهدی	۷۵/۶/۳	مشهد	۱۸/۲۵
زهرا ساداتی	محمد سعید	۷۴/۱۲/۵	کرمانشاه	۱۸/۱۷
فاطمه خدابخشی	علی اکبر	۷۵/۶/۶	تهران	۱۹/۳۵

مقادیر فیلد اول جدول قابل تجزیه هستند به این معنا که اگر هر کدام از اسامی نام و نام خانوادگی را تقسیم کنیم، آنگاه دو بخش اطلاعاتی معنادار خواهیم داشت، به عنوان نمونه در رکورد اول، فیلد اول دارای مقدار «مینا رضایی» است، می‌توان در این فیلد دو مقدار «مینا» و «رضایی» را استخراج نمود طوری که هر کدام از این اطلاعات معنادار باشند (جزء اول نام و جزء دوم، نام خانوادگی رکورد اول است). در این حالت می‌توان گفت که جدول در فرم نرمال اول نیست، زیرا مقادیر فیلدها قابل تجزیه هستند.

**سؤال:** فیلد تاریخ تولد نیز به اجزای روز، ماه و سال قابل تجزیه است، آیا در این حالت نیز می‌توان گفت جدول در حالت نرمال نوع اول قرار ندارد؟

**پاسخ:** اصطلاح معنادار وابسته به نوع محیط عملیاتی است، در محیط عملیاتی مدرسه، فیلد تاریخ به عنوان یک فیلد مجزا و غیر قابل تجزیه در نظر گرفته می‌شود. همین مسأله را می‌توان در فیلد شماره تلفن نیز بررسی کرد، هر شماره تلفن از ترکیب چند عدد در کنار هم تشکیل شده است، اگر اعداد را تجزیه کنیم، معنایی در محیط عملیاتی نخواهند داشت. در نتیجه در جدول ۱۳-۲ فقط فیلد اول قابل تجزیه است.



## ۲-۴-۲ فرم دوم نرمال (NF ۲)

برای روشن شدن مفهوم فرم نرمال باید مفهوم وابستگی تابعی را بشناسید، مفهوم وابستگی تابعی در بانک‌های اطلاعاتی مشابه مفهوم تابع در ریاضی است.

**مفهوم تابع در ریاضی:** در یک رابطه با نام  $R$ ، متغیر  $X$  با  $Y$  وابستگی تابعی دارد، اگر به ازای هر مقدار  $X$  فقط و فقط یک مقدار  $Y$  وجود داشته باشد، در این صورت وابستگی تابعی  $X$  و  $Y$  به صورت  $X \rightarrow Y$  شان داده می‌شود.

جدول ۲-۱۴

X	Y	Z
x1	y1	z1
x2	y2	z3
x1	y1	z2
x3	y2	z2
x4	y3	z3

رابطه  $R$  را مطابق جدول ۲-۱۴ برای سه متغیر  $X$

و  $Y$  و  $Z$  در نظر بگیرید:

در رابطه  $R$ ، می‌خواهیم وابستگی تابعی  $X \rightarrow Y$  و  $Y \rightarrow Z$  را بررسی کنیم: وابستگی تابعی  $X \rightarrow Y$  برقرار است، زیرا به ازای مقادیر یکسان از  $x1$ ، مقادیر یکسانی در  $y1$  وجود دارد، به عنوان مثال ستون  $X$  را در نظر بگیرید، این ستون دارای دو مقدار تکراری  $x1$  است، که هر دو مقدار آن با  $y1$  از ستون  $Y$  متناظر است، در نتیجه به دلیل مساوی بودن طرف دوم رابطه، وابستگی تابعی  $X \rightarrow Y$  برقرار است. وابستگی تابعی  $Y \rightarrow Z$  برقرار نیست، ستون  $Y$  دارای دو مقدار تکراری  $y1$  و  $y2$  است، مقدار  $y1$  متناظر با  $z1$  و  $z2$  در ستون  $Z$  است، لذا به دلیل عدم تساوی این دو مقدار می‌توان گفت وابستگی تابعی  $Y \rightarrow Z$  برقرار نیست.

**نکته** برای بررسی برقراری وابستگی تابعی، حتی اگر یک نمونه یافت شود که وابستگی تابعی را نقض می‌کند، باید گفت که وابستگی تابعی برقرار نیست.



با توجه به مفهوم وابستگی تابعی می‌توان مفهوم فرم دوم نرمال را بررسی کرد: موجودیتی در فرم دوم نرمال است که اولاً در فرم نرمال اول باشد، ثانیاً تمام فیلدهای آن وابستگی تابعی را با کلید اصلی نقض نکنند.

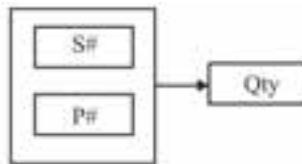
**نکته** اگر جدول دارای یک فیلد کلید ساده باشد، به دلیل ویژگی‌های فیلد کلید می‌توان گفت که جدول در فرم دوم نرمال قرار دارد، اما اگر یک جدول دارای فیلد کلید اصلی مرکب باشد، به این سادگی نمی‌توان نتیجه گرفت که جدول در حالت فرم دوم نرمال قرار دارد یا خیر.

جدول‌های مربوط به محیط عملیاتی تولیدکنندگان قطعات را در نظر بگیرید (جدول‌های ۲-۸، ۲-۹ و ۲-۱۰).

در جدول‌های S و P به دلیل نرمال نوع اول بودن و وجود کلید اصلی ساده می‌توان به راحتی نتیجه گرفت که رابطه مربوط به آن‌ها در حالت نرمال دوم قرار دارد.

اما در جدول SP، کلید اصلی شامل ترکیبی از فیلدهای S# و P# است، در نتیجه برای بررسی این‌که آیا در حالت نرمال دوم قرار دارد یا خیر، باید آن را از نظر وابستگی تابعی بررسی کرد.

نمودار وابستگی تابعی مربوط به جدول SP را می‌توان به صورت زیر نشان داد:



شکل ۲-۴

همان‌طور که می‌دانید جدول SP از روی جداول S و P استخراج شده است، با دقت در جدول SP (جدول ۲-۱۰) می‌بینیم که اجزای کلید اصلی (S# و P#) به‌طور مجزا با Qty وابستگی تابعی ندارند و وابستگی تابعی فقط توسط ترکیب S# و P# وجود دارد، لذا می‌توان گفت که رابطه جدول SP نیز در حالت نرمال دوم NF قرار دارد.

مثال: فرض کنید در محیط عملیاتی تولیدکنندگان قطعات جدول ۲-۱۵ را به جای

SP در نظر گرفته‌ایم:

جدول ۱۵-۲

S#	Status	City	P#	Qty
S1	۲۰	تهران	P1	۳۰۰
S1	۲۰	تهران	P2	۲۰۰
S1	۲۰	تهران	P3	۴۰۰
S1	۲۰	تهران	P4	۲۰۰
S1	۲۰	تهران	P5	۱۰۰
S1	۲۰	تهران	P6	۱۰۰
S2	۱۰	کرمانشاه	P1	۳۰۰
S2	۱۰	کرمانشاه	P2	۴۰۰
S3	۱۰	کرمانشاه	P2	۲۰۰
S4	۲۰	تهران	P2	۲۰۰
S4	۲۰	تهران	P4	۳۰۰
S4	۲۰	تهران	P5	۴۰۰

این جدول در فرم نرمال اول قرار دارد اما در فرم نرمال نوع دوم قرار ندارد. اگر فیلدهای P# و S# را به عنوان کلید اصلی مرکب در نظر بگیریم، بخش‌هایی از کلید اصلی با سایر فیلدها وابستگی تابعی دارد. S# به تنهایی با City وابستگی تابعی دارد که این امر با تعریف فرم دوم نرمال در تناقض است و نباید اجزای کلید اصلی در یک رابطه با سایر فیلدها وابستگی تابعی داشته باشد.

### ۳-۴-۲ فرم سوم نرمال (۳ NF)

موجودیت یا جدولی در فرم سوم نرمال است که اولاً در فرم دوم نرمال بوده، ثانیاً تمام صفت‌های غیرکلید آن وابستگی تابعی فقط به کلید اصلی داشته باشند نه به یک صفت غیرکلید، یا به عبارت دیگر صفات غیرکلید وابستگی تابعی با واسطه با کلید اصلی نداشته باشند به عنوان نمونه فرض کنید در محیط عملیاتی تولیدکنندگان قطعات، این جدول‌ها را در نظر گرفته‌ایم:

جدول ۲-۱۶

S#	City
S1	تهران
S2	کرمانشاه
S3	کرمانشاه
S4	تهران
S5	بوشهر

جدول ۲-۱۷

City	Status
بوشهر	۳۰
تهران	۲۰
کرمانشاه	۱۰
شیراز	۵۰

در این دو جدول می‌توان نمودار وابستگی تابعی را به صورت زیر ترسیم کرد:

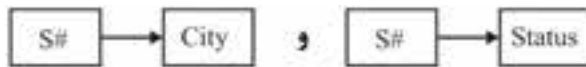


شکل ۲-۵

به دلیل وجود واسطه در وابستگی تابعی، جدول‌های ۲-۱۶ و ۲-۱۷ در فرم سوم نرمال نیستند.

(در این جدول‌ها کلید S# با City وابستگی تابعی دارد و City نیز با Status وابستگی تابعی دارد).

برای حذف این وابستگی تابعی می‌توان فیلد واسطه را در وابستگی تابعی حذف کرد:



شکل ۲-۶

در این صورت وابستگی تابعی با واسطه حذف شده است، بنابراین رابطه در حالت سوم نرمال قرار دارد.

## واژه‌نامه

Attribute	صفت
Entity	موجودیت
Hierarchical	سلسله مراتبی
Network	شبکه
Normalization	نرمال‌سازی
Producer	تولیدکننده
Relational	رابطه‌ای

## خلاصه مطالب

- پایگاه داده مجموعه‌ای سازمان‌یافته از اطلاعات است که می‌تواند به صورت رکوردهای ذخیره شده در جداول باشد و با یک روش اصولی و از طریق فرم‌ها، گزارش‌ها و ... می‌تواند به درخواست‌های کاربران پاسخ دهد.
- مدل‌های پایگاه داده طرح و الگوی کاری برای کاربران پایگاه داده‌ها را در سطح منطقی مشخص می‌کند.
- مدل سلسله مراتبی توسط گراف پیاده‌سازی می‌شود که ارتباط بین گره‌ها از بالا به پایین صورت می‌گیرد و مسیری از گره‌های پایین‌تر به گره‌های بالاتر وجود ندارد.
- مدل شبکه‌ای شباهت زیادی به سلسله مراتبی دارد، اما به دلیل وجود ارتباط از پایین به بالا پیاده‌سازی آن راحت‌تر است.
- مدل رابطه‌ای یکی از بهترین و سریع‌ترین مدل‌های پایگاه داده است و اطلاعات در قالب جدول‌های مجزا مشخص می‌شوند.
- مدل رابطه‌ای بهترین مدل پیاده‌سازی سیستم‌های بانک‌های اطلاعاتی است. در این مدل پایگاه داده از دید کاربران شامل یک مجموعه جدول است که هر جدول دارای سطرها و ستون‌هایی است.
- تعیین ماهیت ارتباط در مدل رابطه‌ای یکی از مهم‌ترین مراحل طراحی بانک اطلاعاتی است و پس از تحلیل موجودیت‌ها و بررسی ماهیت ارتباط آن‌ها، باید رابطه‌ها را به صورت

جدول پیاده‌سازی کرد.

- پس از تهیه جدول‌های بانک اطلاعاتی به منظور بهینه‌سازی جداول، کاهش افزونگی اطلاعات و حفظ یکپارچگی اطلاعات باید آن‌ها را نرمال‌سازی کرد.
- نرمال‌سازی مدل‌های مختلفی دارد که مهم‌ترین آن‌ها عبارتند از:
  - فرم نرمال اول
  - فرم نرمال دوم
  - فرم نرمال سوم

## آزمون نظری

- ۱ - به مجموعه‌ای سازمان یافته از اطلاعات ..... گفته می‌شود.
- الف - پایگاه داده - ب - جدول - ج - رکورد - د - فیلد
- ۲ - پیاده‌سازی کدام یک از مدل‌های پایگاه داده زیر مشکل‌تر از سایر مدل‌هاست؟
- الف - شبکه‌ای - ب - رابطه‌ای - ج - شیء‌گرا - د - سلسله مراتبی
- ۳ - مجموعه‌ها و رکوردها در کدام مدل بررسی می‌شود؟
- الف - شبکه‌ای - ب - رابطه‌ای - ج - شیء‌گرا - د - سلسله مراتبی
- ۴ - در سیستم عملیاتی مدرسه موجودیت‌های دانش آموز - درس چه نوع رابطه‌ای با هم دارند؟
- الف - یک به یک - ب - یک به چند
- ج - چند به چند - د - این دو موجودیت به‌طور مستقیم رابطه‌ای ندارند.
- ۵ - کدام یک از گزینه‌های زیر از اهداف نرمال‌سازی بانک اطلاعاتی است؟
- الف - کاهش افزونگی و حفظ یکپارچگی اطلاعات
- ب - افزایش افزونگی و افزایش یکپارچگی اطلاعات
- ج - کاهش تعداد جدول‌های بانک اطلاعاتی و حفظ یکپارچگی
- د - افزایش تعداد رابطه‌های بانک اطلاعاتی و افزایش یکپارچگی
- ۶ - اگر تمام صفت‌های غیر کلیدی یک موجودیت به تمام کلید اصلی وابستگی تابعی داشته باشند و همچنین تمام صفت‌های آن یکتا باشد، این موجودیت در کدام فرم نرمال است؟
- الف - فرم نرمال اول - ب - فرم نرمال دوم
- ج - فرم نرمال سوم - د - هیچ کدام

## آشنایی با دستورات کار با پایگاه داده

### واحد کارسوم

پس از مطالعه این واحد کار از فراگیر انتظار می‌رود که:

- ۱ - با اصول ایجاد پایگاه داده در MySQL آشنا باشد.
- ۲ - بتواند یک جدول جدید در MySQL ایجاد کند.
- ۳ - بتواند یک جدول را در MySQL حذف کند.
- ۴ - ساختار جداول را اصلاح کند.
- ۵ - بتواند جدول‌ها را تغییر نام دهد.
- ۶ - بتواند یک رکورد و داده‌های جدید به جدول اضافه کند.
- ۷ - بتواند اطلاعات را از جداول حذف کند.
- ۸ - بتواند اطلاعات جداول‌های موجود را ویرایش و به‌روز رسانی کند.
- ۹ - بتواند اطلاعات جدول‌ها را انتخاب کند.

زمان(ساعت)	
نظری	عملی
۱۷	۳۵



## مقدمه

همان طور که می‌دانید سیستم‌های مدیریت پایگاه داده مختلفی وجود دارند که با توجه به نوع نیاز، اندازه سیستم، بودجه و ... می‌توان یکی از آن‌ها را برای پیاده‌سازی سیستم بانک اطلاعاتی محیط عملیاتی استفاده کرد، سیستم‌های مدیریت پایگاه داده معمولاً از یک زبان ساخت‌یافته استاندارد با نام SQL به‌طور مشترک استفاده می‌کنند.

## ۳-۱ آشنایی با SQL

- SQL مخفف عبارت Structured Query Language (زبان جستجوی ساخت‌یافته) است و به کاربر امکان اتصال و دسترسی به اطلاعات موجود در یک پایگاه داده را می‌دهد.
- زبان SQL قادر است تا برای یک پایگاه داده عمل جستجو و گزینش اطلاعات را انجام دهد و همچنین اطلاعات ذخیره شده در یک پایگاه داده را بازیابی، حذف، ذخیره، اضافه یا به روزرسانی کند.
- زبان SQL یک استاندارد بین‌المللی است.

### Table ۳-۱-۱

با مفهوم جدول در واحدکار دوم آشنا شده‌اید. جدول‌ها مهم‌ترین عناصر پایگاه داده هستند که برای ذخیره و نگهداری سازمان‌یافته اطلاعات مورد استفاده قرار می‌گیرند.

### Query ۳-۱-۲

از Query برای ایجاد یک پرسش و جستجو در جدول و استخراج اطلاعات مورد نظر استفاده می‌شود. نتایج حاصل از یک Query در یک جدول موقت که دارای فیلدها و سطرهای واجد شرایط جستجو است، نمایش داده می‌شود.

مثال: به عنوان مثال می‌خواهیم، مشخصات دانش‌آموزانی را که در ترم دوم سال تحصیلی ۸۵-۸۶ موفق به گذراندن درس پایگاه داده شده‌اند از جدول دانش‌آموزان استخراج کنیم. برای این منظور از یک Query استفاده می‌کنیم. عمده دستورات SQL برای ایجاد و مدیریت

Query استفاده می‌شوند. در مباحث بعدی، کار با یک Query را بررسی می‌کنیم.

## ۳-۱-۳ SQL Data Manipulation Language (DML) یا زبان دستکاری

### اطلاعات SQL

این بخش از دستورات SQL، شامل دستوراتی برای به روز کردن، ورود اطلاعات یا حذف رکوردها در محیط پایگاه داده است. عمده دستورات این بخش عبارتند از:

- Select**: برای استخراج اطلاعات از یک جدول بانک اطلاعاتی مورد استفاده قرار می‌گیرد.
- Update**: برای به روز کردن اطلاعات یک جدول بانک اطلاعاتی مورد استفاده قرار می‌گیرد.
- Delete**: برای حذف اطلاعات از یک جدول بانک اطلاعاتی مورد استفاده قرار می‌گیرد.
- Insert**: برای ورود اطلاعات به یک جدول بانک اطلاعاتی مورد استفاده قرار می‌گیرد.

## ۳-۱-۴ SQL Data Definition Language (DDL) یا زبان تعریف اطلاعات SQL

از این قسمت می‌توان برای ایجاد یا حذف جداول از بانک اطلاعاتی، تعریف اندیس‌ها، کلیدواژه‌ها، ایجاد ارتباط و پیوند بین جدول‌ها یا ایجاد محدودیت‌های دسترسی استفاده کرد. عمده دستورات این بخش عبارتند از:

- Create Table**: برای ایجاد یک جدول جدید به کار می‌رود.
- Alter Table**: برای تغییر اطلاعات موجود در یک جدول به کار می‌رود.
- Drop Table**: برای حذف یک یا چند جدول از بانک اطلاعاتی به کار می‌رود.
- Create Index**: برای ایجاد اندیس یا کلید واژه در یک جدول به کار می‌رود.
- Drop Index**: برای حذف اندیس یا کلید واژه در یک جدول به کار می‌رود.

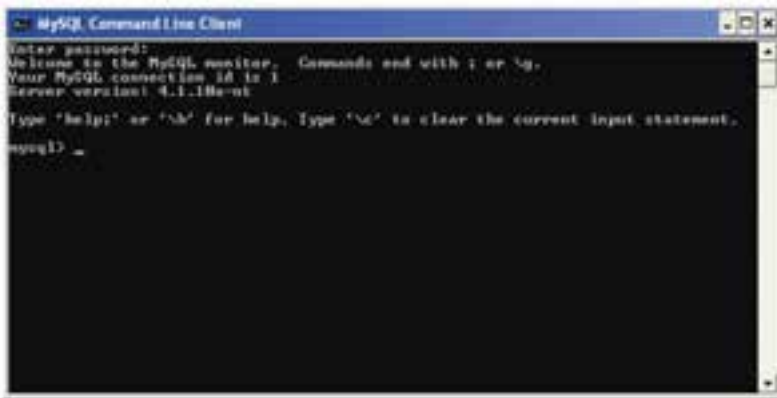
**اندیس (Index)**: اندیس عبارت است از یک شماره که به هر یک از فیلدها در سطرهای یک جدول اختصاص داده می‌شود. اندیس‌ها از دید کاربر کاملاً مخفی هستند. استفاده از اندیس‌ها باعث می‌شود تا برنامه بتواند مقادیر سطرهای مختلف را بر حسب مقدار یک فیلد و بر حسب شماره اندیس آن‌ها از کم به زیاد یا برعکس مرتب کند و در عملیات جستجو باعث بالا رفتن سرعت جستجو می‌شود.

برای اجرای دستورات SQL و مشاهده نتیجه آن باید از یک سیستم مدیریت پایگاه داده استفاده کرد. سیستم مدیریت پایگاه داده‌ای که در این کتاب استفاده شده است، MySQL می‌باشد.

**نکته** در انتهای هر دستور SQL باید از علامت ; استفاده شود.



ابتدا از نصب MySQL روی سیستم خود مطمئن شوید، با نصب MySQL در واحدکار اول آشنا شده‌اید. سپس تمام دستورات را از طریق command line اجرا کنید. برای ورود به محیط command line از منوی Start گزینه All Programs سپس MySQL، MySQL Server 5.1 و در نهایت MySQL Command Line Client را انتخاب نمایید.



شکل ۳-۱

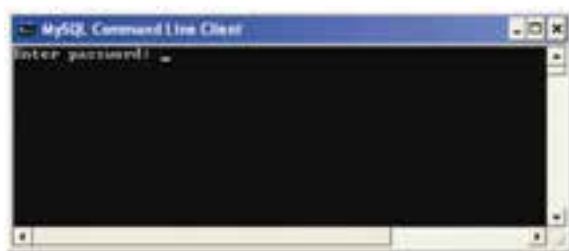
اگر رمز عبوری را هنگام نصب تعیین کرده‌اید، آن را وارد کنید، سپس کلید Enter را فشار دهید و اگر هیچ رمز ورودی تعیین نشده است، بدون وارد کردن هیچ کاراکتری، کلید Enter را فشار دهید.

## ۳-۲ دسترسی به MySQL از طریق خط فرمان

اگر قصد داشته باشید از طریق خط فرمان با محیط MySQL کار کنید، ابتدا باید MySQL را روی سیستم نصب کنید (با شیوه نصب MySQL در واحد کار اول آشنا شدید). از منوی Start و در قسمت All Programs گزینه MySQL، سپس MySQL Server و بعد از آن MySQL Command Line Client را انتخاب کنید.

در پنجره شکل ۳-۲ اگر کلمه رمزی را هنگام نصب MySQL تعیین کرده‌اید آن را وارد کنید، سپس کلید Enter را فشار دهید، در غیر این صورت بدون وارد کردن هیچ

کاراکتری کلید Enter را فشار دهید.



شکل ۳-۲

مقابل عبارت `MySQL>` علامت چشمک‌زن ظاهر می‌شود که به این معناست که به `MySQL` وارد شده‌اید و می‌توانید دستورات لازم را اجرا کنید. پس از ورود به محیط `MySQL` اگر بخواهید اعمال مختلف را روی یک بانک اطلاعاتی انجام دهید، ابتدا باید با استفاده از دستور `use` پایگاه داده مورد نظر را انتخاب کنید. ساختار این دستور به شکل زیر است:

نام پایگاه داده `use`

مثال: `use db1;`

## ۳-۳ دستور `create`

دستور `Create` برای ایجاد یک جزء جدید در سیستم‌های پایگاه داده مورد استفاده قرار می‌گیرد.

### ۳-۳-۱ ایجاد یک پایگاه داده (`Create a DataBase`)

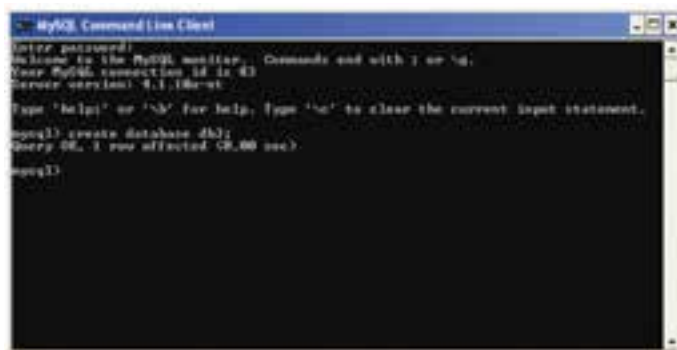
برای ایجاد یک پایگاه داده جدید از این دستور استفاده می‌شود:

نام پایگاه داده `create DataBase`

مثال: `create DataBase UniverArea ;`

مثال: `create DataBase db3 ;`

برای اجرای دستور فوق در خط فرمان، دستور مورد نظر را وارد کنید، سپس کلید `Enter` را فشار دهید.



شکل ۳-۳

پیغام Query OK به این معناست که درخواست با موفقیت انجام پذیرفته است.

### ۳-۳-۲ ایجاد یک جدول (create Table)

برای ایجاد یک جدول ۳ عامل اصلی را باید در نظر گرفت:

تعیین یک نام منحصر به فرد برای جدول

تعیین تعداد ستون‌های (فیلدها) جدول و نام آن‌ها

تعیین نوع داده‌ای ستون‌های جدول و اندازه آن‌ها در صورت نیاز

شکل کلی دستور ایجاد جدول در پایگاه داده به صورت زیر است:

```
create Table Students  
(  
StudentID Numeric (8) ,  
Name Char (15) ,  
Family Char (15) ,  
... ;  
)
```

**نکته** نوع داده انتخابی برای ستون‌ها، باید یکی از انواع داده‌های مجاز در SQL باشد. برای فیلدهایی که می‌خواهید حداکثر طول مجاز را تعیین کنید، می‌توانید مقدار آن را در پرانتز جلوی نام نوع داده فیلد ذکر کنید.



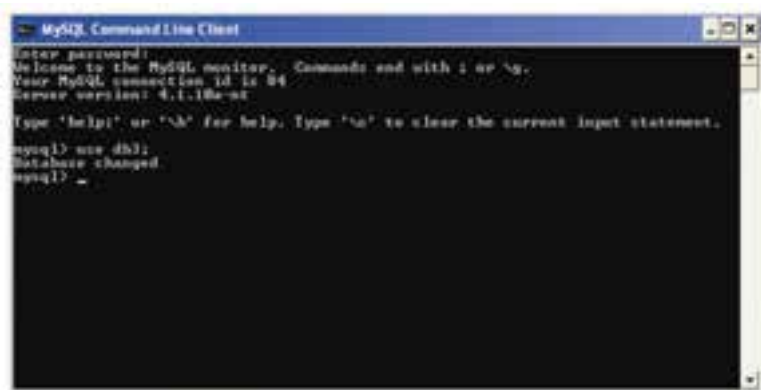
مثال: نوع داده‌ای از نوع کاراکتر با حداکثر طول ۱۵:

Name Char (15) ;

برای ایجاد جدول توسط خط فرمان، در صورتی که پایگاه داده را قبلاً ایجاد کرده باشید، ابتدا باید با استفاده از دستور

use نام پایگاه داده ;

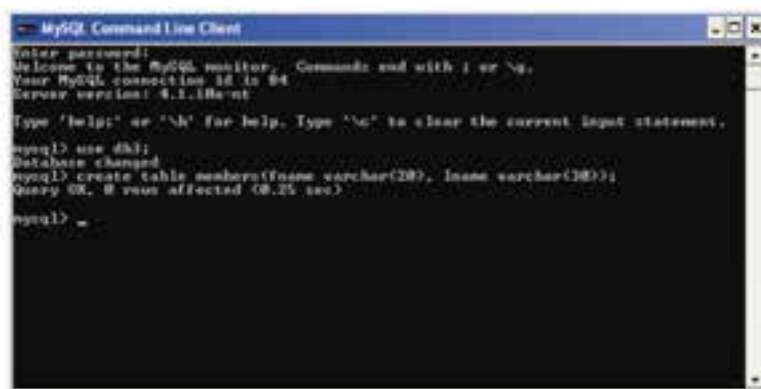
مشخص نمایید که می‌خواهید روی کدام پایگاه داده کار کنید.



شکل ۳-۴

حال برای ایجاد جدول باید دستور ایجاد جدول به همراه فیلدها و نوع هر کدام را به صورت زیر درج کنید:

(... و نوع متغیر دوم فیلد دوم و نوع متغیر اول فیلد اول) نام جدول create table



شکل ۳-۵

سایر دستورات SQL را نیز میتوانید از طریق خط فرمان اجرا کنید.

### ۳-۳-۳ ایجاد یک اندیس (create Index)

از این دستور برای ایجاد اندیس در ستون‌های جدول استفاده می‌شود. در یک جدول می‌توان برای یک یا چند ستون جدول، اندیس ایجاد کرد که این اندیس‌ها باعث بالا رفتن سرعت جستجو در رکوردهای جدول می‌شوند. اندیس یک شماره است که به هر سطر جدول اختصاص داده می‌شود و معمولاً از صفر شروع می‌شود. اندیس‌ها از دید کاربر مخفی هستند و هر اندیس یک نام منحصر به فرد دارد. شکل کلی ایجاد یک اندیس به صورت زیر است:

نام اندیس create Index

(نام فیلد مورد نظر) نام جدول on

مثال: create Index Rowup

on Students (StudentID) ;

**نکته** آوردن واژه Unique قبل از واژه Index در دستور قبل، باعث ایجاد یک شماره منحصر به فرد برای هر رکورد در ستون جدول می‌شود، به این معنی که هیچ دو رکوردی دارای اندیس‌های یکسان نخواهند بود.

- اندیس‌ها به صورت پیش‌فرض صعودی (از کم به زیاد) هستند، برای ایجاد یک اندیس با ترتیب برعکس (از زیاد به کم) باید بعد از نام فیلد در پرانتز، عبارت DESC ذکر شود.
- به منظور ایجاد یک اندیس برای بیش از یک فیلد، باید نام فیلدهای مورد نظر را به ترتیب در پرانتز بعد از نام جدول وارد کرد.

**مثال:** برای ایجاد اندیس روی فیلدهای StudentID و Name و Family، در جدول ۳-۱،

کد SQL را می‌توان به صورت زیر نوشت:

create Index Rowup

on Students (StudentID , Name , Family) ;

جدول ۳-۱

Student Table					
StudentID	Name	Family	Major	Area	Grade
41252214	Ahmad	Rezaee	Computer	Tehran	۱۸
10724113	Ehsan	Amiri	Electronic	Karaj	۱۴
10254861	Zahra	Hoseini	Computer	Tehran	۱۷
27365187	Sahar	Ahmadi	Electronic	Bam	۱۶
35654415	Hesam	Razavi	Electronic	Tehran	۱۹



تمرین:

جدول ۳-۱ را در MySQL ایجاد کنید.

## ۳-۴ دستور select

دستور Select برای انتخاب و استخراج اطلاعات مورد نظر از یک یا چند جدول و سپس مشاهده نتایج در یک جدول موقت استفاده می‌شود. شکل کلی این دستور به شکل زیر است:

نام ستون یا ستون‌های مورد نظر select

نام جدول from

مثال: برای مشاهده ستون‌های Name و Family از جدول ۳-۱، دستور SQL به صورت

زیر نوشته می‌شود:

کد:


```
select Name,Family
```

```
from Student
```

خروجی

Name	Family
Ahamd	Rezaee
Ehsan	Amiri
Zahra	Hoseini
Sahar	Ahmadi



**نکته**  برای انتخاب تمام ستون‌های جدول می‌توان از نشانگر \* در قسمت نام ستون استفاده کرد. در این صورت تمام ستون‌های جدول در خروجی نمایش داده خواهند شد:

```
select *  
from Student;
```

### تمرین

۱ - یک دستور SQL بنویسید که فیلدهای نام، نام خانوادگی و نمره جدول ۳-۱ را نشان دهد.

۲: جدول‌های ۲-۸، ۲-۹ و ۲-۱۰ (واحدکار دوم) را در محیط MySQL ایجاد کنید، سپس تمام فیلدهای این سه جدول را در خروجی نمایش دهید.

**واژه کلیدی DISTINCT:** چنانچه در ستون‌های مورد جستجو، موارد تکراری وجود داشته باشد در نتیجه خروجی نمایش داده خواهند شد. برای جلوگیری از چنین موردی و عدم نمایش موارد تکراری، پس از دستور Select عبارت DISTINCT نوشته می‌شود:

جدول ۳-۲

Orders Table	
Order Number	Company
1012	Dell
1014	Sega
1016	Dell
1018	Sony

مثال ۱:

کد:

```
select DISTINCT Company  
from Orders;
```

خروجی

Company
Dell
Sega
Sony

مثال ۲:

کد:

```
select Company
from Orders;
```

خروجی

Company
Dell
Sega
Dell
Sony

## ۵-۳ دستور where

دستور where برای افزودن شرط یا شرط‌هایی برای محدود کردن نتایج جستجو و استخراج نتایج دقیق‌تر استفاده می‌شود. این دستور باید پس از دستور select و تعیین ستون‌ها از جدول مورد نظر به کار رود. با استفاده از عملگرهای and، or و پرانتز می‌توان چندین شرط را با هم ترکیب کرد. برنامه فقط سطرهایی از جدول را که با شرط‌های ارائه شده سازگاری دارند نمایش خواهد داد. این دستور با عملگرهای زیر می‌تواند به کار رود:

جدول ۳-۳

عملگر	مفهوم	عملگر	مفهوم
=	برابر با، مساوی	!=	نا برابر با، نامساوی
<	کوچک‌تر یا کمتر	<=	کوچک‌تر مساوی
>	بزرگ‌تر یا بیشتر	>=	بزرگ‌تر مساوی
Between	بین دو مقدار	like	تعیین یک قالب برای جستجو

مثال: انتخاب نام و نام خانوادگی دانش‌آموزانی که شهر محل تولد آن‌ها تهران است از جدول ۳-۱ (Student) به این صورت انجام می‌شود:

کد:

```
select Name , Family
from Student
where Area = "Tehran ";
```

خروجی

Name	Family
Ahmad	Rezaee
Zahra	Hoseini
Hesam	Razavi

### ۵-۳ اعمال بیش از یک شرط و ادغام شرطها با عملگر and

مثال: مشخصات نام و نام خانوادگی دانش‌آموزانی که شهر آنها تهران بوده و نمره آنها بیش از پانزده است:  
کد:

select Name , Family  
from Student  
where Area = " Tehran"  
and Grade > 15;

خروجی

Name	Family
Ahmad	Rezaee
Hesam	Razavi

**نکته** در هنگام اعمال شرط جستجو، مقادیر رشته‌ای (متن) باید بین دو علامت " " قرار بگیرند، ولی در مورد موارد عددی این کار لازم نیست.

**تمرین:** 

در جدول S که آنرا در تمرین ۲ ایجاد کردید، اطلاعات رکوردهایی را نمایش دهید که در شهر تهران تولید شده‌اند.

### ۵-۳ عبارت like

این عبارت برای تعیین یک قالب خاص برای جستجو به کار می‌رود:  
مثال: نام دانش‌آموزانی که شهر محل تولد آنها با حرف K شروع می‌شود:

select Name from Student  
where area like "K%";

مثال: نام دانش‌آموزانی که نام شهر محل تولد آنها به an ختم می‌شود:  
select Name from Student

where Area like "%an";

مثال: نام دانش‌آموزانی که شهر محل تولد آنها دارای حروف st است:

Select Name From Student

Where Area like "%st%";

مثال: نام دانش‌آموزانی که شهر محل تولد آن‌ها ۶ حرفی بوده و با حرف T شروع

می‌شود:

```
select Name from Student
```

```
where Area like "T-----" ;
```

نتیجه‌گیری: علامت % به معنای هر چند کاراکتر یا حرف دلخواه در قالب ارائه شده

و علامت - نمایانگر یک کاراکتر دلخواه است.

## ۳-۶ دستور insert

این دستور برای وارد کردن و ایجاد یک رکورد جدید در جدول استفاده می‌شود.

شکل کلی این دستور به این شرح است:

```
insert into نام جدول
```

```
values (...و مقدار ۲ و مقدار ۱) ;
```

می‌توان تعیین کرد که مقادیر مورد نظر به ترتیب به کدام ستون‌های جدول وارد شوند.

برای این منظور ابتدا نام ستون‌های مورد نظر را به ترتیب جلوی نام جدول در یک پرانتز

وارد کرده و با کاما از هم جدا می‌کنیم. سپس مقادیر متناظر را به همان ترتیب پس از واژه

values در پرانتز وارد می‌کنیم.

**نکته** چنانچه برای فیلد یا فیلدهایی مقداری در نظر گرفته نشود، جای آن‌ها در

جدول خالی می‌ماند. فقط باید به ترتیب نام ستون‌ها و مقادیر دقت کرد.

در شکل ۳-۶ درج داده در جدول members را مشاهده می‌کنید.

حال برای اینکه اطلاعات درج شده را مشاهده کنید، می‌توانید از دستور زیر استفاده

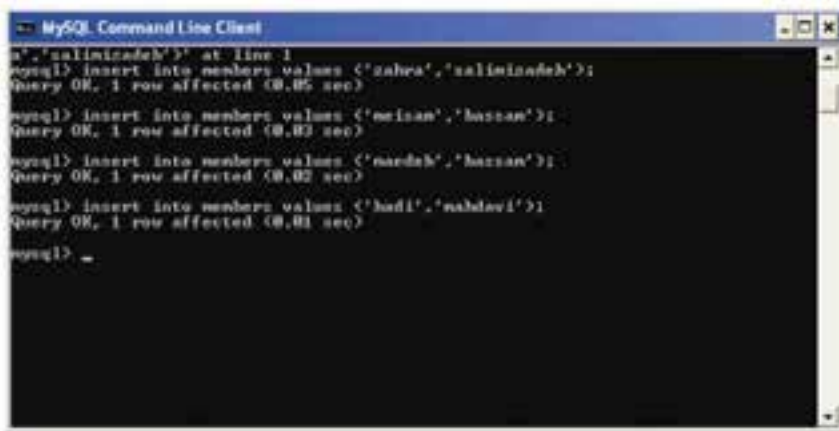
نمایید:

```
SELECT
```

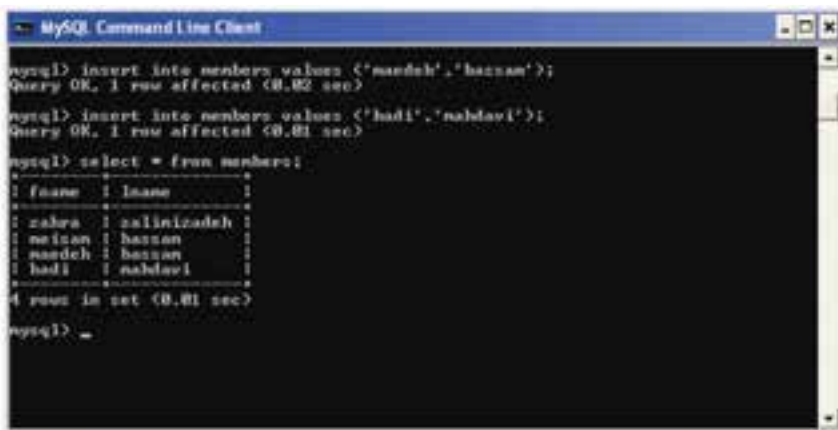
```
* from members;
```

این دستور را نوشته، سپس کلید Enter را فشار دهید. مشاهده می‌کنید اطلاعاتی که قبلاً

در جدول وارد شده بود مطابق شکل ۳-۷ ظاهر شده‌اند.



شکل ۳-۶



شکل ۳-۷

مثال: مشخصات نام و نام خانوادگی را برای یک دانش آموز به نام Omid Safavi در جدول ۳-۱ وارد کنید.

شرح دستور وارد کردن نام و وضعیت جدید جدول:  
کد:

```
insert into Student ( Name , Family )
values ( "Omid" , "Safavi" );
```

### خروجی

Student Table					
StudentID	Name	Family	Major	Area	Grade
41252214	Ahmad	Rezaee	Computer	Tehran	18
10724113	Ehsan	Amiri	Electronic	Karaj	14
10254861	Zahra	Hoseini	Computer	Tehran	17
27365187	Sahar	Ahmadi	Electronic	Bam	16
35654415	Hesam	Razavi	Electronic	Tehran	19
	Omid	Safavi			

## ۷-۳ دستور update

این دستور برای تغییر اطلاعات موجود در یک فیلد جدول و جایگزینی آن با یک مقدار جدید به کار می‌رود.

شکل کلی این دستور به صورت زیر است:

نام جدول update

مقدار جدید = نام فیلد set

؛ مقدار قبلی = نام فیلد where

مثال: نام دانش‌آموز احمد رضایی را به علی تغییر دهید:

```
update Student
```

```
set Name = "Ali"
```

```
where Name = "Ahmad" and Family = "Rezaee";
```

• همچنین می‌توان برای تعیین رکوردی که می‌خواهیم تغییر دهیم، از مقدار یک فیلد دیگر به غیر از فیلدی که می‌خواهیم تغییر کند استفاده کنیم. در این حالت از فیلد مذکور، برای آدرس‌دهی رکورد مورد نظر استفاده می‌شود.

مثال: نام دانش‌آموز با شماره دانش‌آموزی ۴۱۲۵۲۲۱۴ را به علی در جدول ۱-۳ تغییر دهید:

• اعمال دستور تغییر و وضعیت جدید جدول:

کد:

```
update Student set Name = "Ali "
```

```
where StudentID = "41252214 ";
```

### خروجی

Student Table					
StudentID	Name	Family	Major	Area	Grade
41252214	Ali	Rezaee	Computer	Tehran	18
10724113	Ehsan	Amiri	Electronic	Karaj	14
10254861	Zahra	Hoseini	Computer	Tehran	17
27365187	Sahar	Ahmadi	Electronic	Bam	16
35654415	Hesam	Razavi	Electronic	Tehran	19

### تغییر اطلاعات در بیش از یک فیلد در رکورد

در این حالت تمام مقادیر جدیدی که می‌خواهیم تغییر دهیم، جلوی واژه کلیدی Set به ترتیب نوشته و با کاما از هم جدا می‌کنیم. سپس به همان ترتیب، فیلدها با مقادیر قبلی را در جلوی واژه where می‌نویسیم. شکل کلی آن به صورت زیر است:

نام جدول update

مقدار = نام فیلد ۲ ، مقدار = نام فیلد ۱ set

where ... و مقدار قبلی = نام فیلد ۲ و مقدار قبلی = نام فیلد ۱

• همچنین می‌توان فقط از یک فیلد در قسمت شرط برای آدرس دهی رکورد مورد نظر استفاده کرد.

### تمرین:

در جدول ۹-۲ از واحد کار دوم، مشخصات تولیدکننده P6 را تغییر دهید، طوری که وزن قطعه‌های تولید شده به ۲۵ و رنگ قطعه به زرد تبدیل شود.

مثال: در جدول ۱-۳ نام، نام خانوادگی و رشته تحصیلی دانش‌آموز به شماره دانش‌آموزی ۴۱۲۵۲۲۱۴ را به ترتیب به Ali، Saeedi و Electronic تغییر دهید:

• اعمال دستور تغییر و وضعیت جدید جدول:

کد:

```
update Student
```

```
set Name = "Ali" , Family = "Saeedi" , Major = "Electronic"
```

```
where StudentID = 41252214;
```

### خروجی

Student Table					
StudentID	Name	Family	Major	Area	Grade
41252214	Ali	Saeedi	Electronic	Tehran	18
10724113	Ehsan	Amiri	Electronic	Karaj	14
10254861	Zahra	Hoseini	Computer	Tehran	17
27365187	Sahar	Ahmadi	Electronic	Bam	16
35654415	Hesam	Razavi	Electronic	Tehran	19

### ۸-۳ دستور delete

از این دستور برای حذف اطلاعات یک رکورد در جدول استفاده می‌شود.  
 شکل کلی این دستور به شکل زیر است:

نام جدول delete from

مقدار فیلد از رکورد موردنظر = نام فیلد where

مثال: در جدول ۱-۳، رکورد اطلاعات دانش‌آموز به شماره دانش‌آموزی ۱۰۷۲۴۱۱۳ را حذف کنید.

اعمال دستور حذف و وضعیت جدید جدول:  
 کد:

delete from Student

where StudentID = "10724113";

### خروجی

Student Table					
StudentID	Name	Family	Major	Area	Grade
41252214	Ahmad	Rezaee	Computer	Tehran	18
10254861	Zahra	Hoseini	Computer	Tehran	17
27365187	Sahar	Ahmadi	Electronic	Bam	16
35654415	Hesam	Razavi	Electronic	Tehran	19

### حذف تمام رکوردهای یک جدول

این امکان وجود دارد که بدون حذف یک جدول، تمام رکوردهای درون آن را پاک کرد. این به منزله خالی کردن جدول از اطلاعات است، یعنی ساختار، اندیس‌ها و ویژگی‌های جدول دست نخورده باقی مانده و فقط مقادیر درون آن از بین می‌رود.



شکل کلی این دستور به صورت زیر است:

delete from جدول ;

یا

delete \* from جدول ;

مثال: حذف تمام رکوردهای جدول Student

delete from Student;

## ۳-۹ دستور order by

اطلاعات خروجی در دستور Select به طور کلی بی‌نظم است و مقادیر خروجی در ستون‌های جدول برحسب مقدار هیچ ستونی مرتب نیستند. این دستور برای مرتب کردن اطلاعات جدول بر اساس مقادیر یک یا چند ستون برحسب شاخص‌هایی مثل ترتیب حروف الفبا، بزرگ‌تر یا کوچک‌تر بودن اعداد و ... استفاده می‌شود.

شکل کلی این دستور به صورت زیر است:

نام جدول مورد نظر from نام فیلدهای مورد نظر select

نام فیلد مورد نظر order by

فیلدهای معرفی شده در دستور select از جدول مذکور نمایش داده شده و اطلاعات بر حسب ستون مقابل دستور order by مرتب می‌شوند.

**عبارت ASC:** به کار بردن این دستور پس از دستور order by باعث مرتب شدن اطلاعات درون جدول از کم به زیاد می‌شود که البته انتخاب پیش‌فرض SQL است و نیازی به درج آن نیست.

**عبارت DESC:** به کار بردن این واژه پس از دستور order by باعث مرتب شدن اطلاعات جدول از زیاد به کم در اعداد و در حروف از Z تا A می‌شود.

مثال: از جدول Student، مشخصات نام و نام خانوادگی دانش‌آموزان رشته Electronic را برحسب حروف الفبا نمایش دهید.  
کد:

```
select Name , Family from Student  
where Major ="Electronic"  
order by Family ;
```

خروجی

Name	Family
Sahar	Ahmadi
Ehsan	Amiri
Hesam	Razavi

**نکته** همچنین می‌توان یک جدول را بر حسب بیش از یک فیلد مرتب کرد، برای این منظور پس از دستور Order By نام تمام فیلدهای مورد نظر را نوشته و آن‌ها را با کاما از هم جدا می‌کنیم. اولویت مرتب‌سازی با فیلدهایی است که نام آن‌ها اول ذکر شده باشد.

**مثال:** جدول ۱-۳ را بر حسب فیلد نام خانوادگی (بر حسب حروف از A تا Z) و بر حسب نمره (از کم به زیاد) مرتب کنید:

**نکته** در جدول خروجی مثال فوق به ارتباط مقادیر ستون‌های Family و Grade توجه کنید. با وجود این که نمره درسی دانش‌آموز زهرا حسینی از احمد رضایی کمتر است، ولی به دلیل این که اولویت مرتب‌سازی با نام خانوادگی است، نام حسینی بالاتر از رضایی قرار گرفته است.

کد:

```
select Name , Family , Grade From Student  
order by Family ASC , Grade DESC ;
```

خروجی

Name	Family	Grade
Sahar	Ahmadi	16
Ehsan	Amiri	14
Zahra	Hoseini	17
Hesam	Razavi	19
Ahmad	Rezaee	18

**نکته** می‌توان در دستور order by به جای ذکر نام ستون‌ها از یک عدد به جای آن استفاده کرد. عدد استفاده شده، شماره قرارگیری ستون مورد نظر در دستور Select است. برای مثال اولین ستون دستور Select عدد ۱ و دومین ستون عدد ۲ و ... را خواهند داشت. بازنویسی دستور بالا با استفاده از این روش به صورت زیر خواهد بود:

کد:

```
select Name , Family , Grade from Student  
order by 2 ASC , 3DESC;
```

## ۱۰-۳ دستور and و or

از and و or برای ترکیب شرطها در دستور where استفاده می‌شود. گاهی اوقات اطلاعات موردنیاز ما در دستور select دارای چندین شرط مختلف است. به طور مثال ما مشخصات دانش‌آموزان ترم دوم رشته Computer که معدل آن‌ها بیش از ۱۵ است را می‌خواهیم. در این حالت باید هر یک از شرطهای مسئله را به تنهایی تعریف کرده و سپس آن‌ها را با هم ترکیب کنیم. برنامه به ارزیابی هر یک از شرطها پرداخته، سپس نتایج حاصله را با هم ترکیب کرده و خروجی را تعیین می‌کند. عملگر and برای اجرای دستور نیاز دارد تا تمام شرطهای تعیین شده برای آن، درست باشد. در حالی که عملگر or فقط نیاز دارد که حداقل یکی از شرطهای آن درست باشد. در جداول زیر چگونگی برقراری نتیجه نهایی در هنگام استفاده از این دو عملگر را مشاهده می‌کنیم:

and		
شرط ۱	شرط ۲	نتیجه
درست	درست	درست
درست	غلط	غلط
غلط	درست	غلط
غلط	غلط	غلط

or		
شرط ۱	شرط ۲	نتیجه
درست	درست	درست
درست	غلط	درست
غلط	درست	درست
غلط	غلط	غلط

چگونگی حالت‌های فوق را در مثال‌های زیر بررسی می‌کنیم:  
**مثال ۱:** مشخصات دانش‌آموزانی را از جدول ۱-۳ (Student) ارائه دهید که نمره آن‌ها بیش از ۱۵ و رشته تحصیلی آن‌ها Electronic باشد:  
 کد:

```
select * from Student
where Major ='Electronic' and Grade > 15 ;
```

### خروجی

StudentID	Name	Family	Major	Area	Grade
27365187	Sahar	Ahmadi	Electronic	Bam	16
35654415	Hesam	Razavi	Electronic	Tehran	19

مثال ۲: مشخصات دانش‌آموزانی را از جدول Student ارائه دهید که نمره آن‌ها بیش از ۱۷ بوده یا رشته تحصیلی آن‌ها Computer باشد:  
کد:

```
select * from Student  
where Major = 'Computer' OR Grade > 17 ;
```

### خروجی

StudentID	Name	Family	Major	Area	Grade
41252214	Ahmad	Rezaee	Computer	Tehran	18
35654415	Hesam	Razavi	Electronic	Tehran	19

**نکته** همچنین می‌توان با استفاده از عملگر پراتنز چندین شرط مختلف را به صورت دسته‌های مجزا تعریف کرده و سپس همه آن‌ها را با هم ترکیب کرد. در این حالت شرط‌های هر گروه ابتدا داخل پراتنز با هم ترکیب شده، سپس نتیجه حاصله به عنوان نتیجه نهایی آن گروه شرط به دستور Where ارسال می‌شود و در نهایت نتایج تمام گروه‌ها با هم ترکیب می‌شوند.

مثال: از جدول Student مشخصات دانش‌آموزانی را ارائه دهید که نمره آن‌ها ۱۸ و رشته تحصیلی آن‌ها Computer بوده یا نمره آن‌ها ۱۶ و رشته تحصیلی آن‌ها Electronic باشد:  
کد:

```
select * from Student  
where ( ( Grade = 18 and Major = 'Computer' ) or ( Garde = 16 and Major = 'Electronic' ) ) " ;
```

خروجی

StudentID	Name	Family	Major	Area	Grade
41252214	Ahmad	Rezaee	Computer	Tehran	18
27365187	Sahar	Ahmadi	Electronic	Bam	16

۱۱-۳ دستور in

این دستور برای مشاهده اطلاعات رکوردهایی از جدول به کار می‌رود که شما مقدار دقیق حداقل یکی از فیلدهای آن را می‌دانید.

شکل کلی این دستور به صورت زیر است:

نام جدول from نام فیلدهایی که می‌خواهید نمایش داده شود select  
 ; (مقدار ۲ و مقدار ۱) in نام فیلد where

مثال: از جدول Student مشخصات دانش‌آموزانی را ارائه دهید که نام خانوادگی آنها

احمدی یا رضایی است:

کد:

select \* from Student

where Family in ( "Ahmadi" , "Rezaee" );

خروجی

Student Table					
StudentID	Name	Family	Major	Area	Grade
41252214	Ahmad	Rezaee	Computer	Tehran	18
27365187	Sahar	Ahmadi	Electronic	Bam	16

۱۲-۳ دستور between ... and

این دستور برای انتخاب اطلاعات در یک محدوده خاص، در بین دو مقدار تعیین شده استفاده می‌شود. این مقادیر می‌تواند از نوع عددی، متن یا تاریخ باشد. (باید نوع مدنظر با نوع داده‌ای فیلدهای استفاده شده در دستور یکسان باشد).

شکل کلی این دستور به شکل زیر است:

نام جدول from نام فیلدهای مورد نظر برای نمایش select

نام فیلد مورد نظر where

مقدار ۲ and مقدار ۱ between

**نکته** در مورد داده‌های متنی، ترتیب قرارگیری حروف الفبا مد نظر برنامه است.

**مثال:** از جدول Student مشخصات دانش‌آموزانی را ارائه دهید که نام خانوادگی آنها بین دو مقدار احمدی و حسینی باشد. در ضمن خروجی بر حسب نام خانوادگی به ترتیب حروف الفبا مرتب باشد:

کد:

```
select * from Student
```

```
where Family between 'Ahmadi' and 'Hoseini'
```

```
order by family;
```

خروجی

StudentID	Name	Family	Major	Area	Grade
27365187	Sahar	Ahmadi	Electronic	Bam	16
10724113	Ehsan	Amiri	Electronic	Karaj	14
10254861	Zahra	Hoseini	Computer	Tehran	17

**نکته** قرار گرفتن رکوردهای ۲ مقدار تعیین شده در خروجی دستور select، به نوع برنامه پایگاه داده و مدیریت آن بستگی دارد. در برخی از پایگاه‌های داده رکوردهای هر دو مقدار ابتدا و انتها نمایش داده می‌شود و در برخی هم فقط یکی از آنها نمایش داده می‌شود.

برای نمایش اطلاعات خارج از محدوده تعیین شده از یک عملگر not قبل از دستور between ... and استفاده می‌شود.

**مثال:** از جدول ۱-۳ مشخصات دانش‌آموزانی را ارائه دهید که نمره آنها خارج از محدوده ۱۲ تا ۱۵ باشد. در ضمن مقادیر خروجی بر حسب نام خانوادگی به ترتیب حروف الفبا مرتب باشد:

کد:

```
select * from Student
where Grade not between 12 and 15
order by Family;
```

خروجی

StudentID	Name	Family	Major	Area	Grade
27365187	Sahar	Ahmadi	Electronic	Bam	16
10254861	Zahra	Hoseini	Computer	Tehran	17
35654415	Hesam	Razavi	Electronic	Tehran	19
41252214	Ahmad	Rezaee	Computer	Tehran	18

### ۱۳-۳ ویژگی alias

همانطور که در تمام مثال‌های ذکر شده از دستور select مشاهده کردید، نام یا عنوانی که در بالای هر ستون در خروجی نمایش داده می‌شود، همان نام فیلد مربوط به داده در جدول اصلی پایگاه داده است. گاهی اوقات تمایل بر این است که این نام را تغییر دهیم یا در مواردی که در بخش توابع SQL خواهیم دید، یک ستون جدید ایجاد می‌کنیم که باید نامی برای آن تعیین شود. از ویژگی alias برای در نظر گرفتن یک نام مستعار و موقت برای قرار گرفتن در بخش عنوان هر ستون در خروجی دستور select استفاده می‌شود. شکل کلی این دستور به صورت زیر است:

نام جدول from (alias) عنوان as نام فیلد select

مثال: از جدول Student نام و نام خانوادگی دانش‌آموزان رشته Electronic را بر حسب حروف الفبا در ۲ ستون تحت عنوان‌های "نام" و "نام خانوادگی" نمایش دهید:  
 کد:

```
select Name as نام , Family as نام خانوادگی from Student
where Major = 'Electronic'
order by Family ;
```

خروجی

نام	نام خانوادگی
Sahar	Ahmadi
Ehsan	Amiri
Hesam	Razavi



**نکته** می توان روی مقادیر ستون هایی که داده عددی دارند، عملیات ریاضی (مثل ضرب، تقسیم و ...) انجام داده و سپس نتایج محاسبات را در یک ستون جدید با عنوان دلخواه تعیین شده توسط ویژگی `alias` در خروجی دستور `select` نمایش داد. در این حالت ستون یا ستون ها به جدول خروجی اضافه می شود.

در این ارتباط مثالی را ذکر می کنیم:

**مثال:** فرض کنید جدولی داریم به نام Books که در آن مشخصات چند کتاب مختلف قرار دارد. در این جدول قیمت کتاب ها به دلار است و ما می خواهیم قیمت کتاب ها را در یک ستون جدید به تومان مشاهده کنیم پس به این روش عمل می کنیم:

جدول ۳-۴

Books Table			
Price	Publisher	Name	no
8	Microsoft	Learning Access	1
6	DeveloperStudio	Java Script Samples	2
7	DeveloperStudio	HTML Reference	3

کد

`select no , Name , Publisher , Price , Price * 840 as قیمت به تومان from Books ;`

خروجی

Books Table				
no	Name	Publisher	Price \$	قیمت به تومان
1	Learning Access	Microsoft	8	6720
2	Java Script Samples	DeveloperStudio	6	5040
3	HTML Reference	DeveloperStudio	7	5880

## ۱۴-۳ پیوند جدول ها

تا این قسمت تمام مثال ها و مسئله هایی که در SQL به آن ها پاسخ دادیم، مسئله هایی بودند که در آن ها اطلاعات ما فقط از یک جدول استخراج می شد. اما در برنامه نویسی واقعی پایگاه داده ها، ما مجبور هستیم که اطلاعات خود را از بیش از یک جدول استخراج کنیم.



در این حالت ابتدا باید جدول‌هایی که می‌خواهیم اطلاعات را از آن‌ها استخراج کنیم، با هم پیوند دهیم. هدف از ایجاد این ارتباط تلفیق اطلاعات در جدول‌ها و چاپ اطلاعات مورد نظر در خروجی است.

برای پیوند دادن جدول‌ها به عنوان یادآوری این نکات را در نظر داشته باشید:  
**کلید اصلی:** فیلد کلید اصلی در یک جدول، فیلدی است که شرایط زیر را داشته باشد:  
مقدار آن برای هر نمونه رکورد (سطر) منحصر به فرد و غیرتکراری باشد. به عبارت دیگر هیچ ۲ رکوردی در یک جدول در این فیلد مقدار یکسان نداشته باشند. کلید اصلی وجه تمایز ۲ نمونه رکورد مختلف در یک جدول است.  
طول مقادیر آن در حد امکان کوتاه باشد.

**نکته** یک جدول می‌تواند بیش از یک کلید اصلی داشته باشد.

**مثال:** فیلد شماره دانش‌آموزی در جدول Student، کلید اصلی است. هیچ دو دانش‌آموزی نمی‌توانند دارای شماره دانش‌آموزی یکسان باشند.

**کلید خارجی:** کلید خارجی، فیلدی است که در یک جدول کلید اصلی و در جدول دیگر به تنهایی کلید اصلی نباشد. از کلید خارجی برای ارتباط یک به چند ۲ جدول با هم استفاده می‌شود.

### شرط ارتباط دو جدول

برای ارتباط بین جدول‌ها باید شرط‌های زیر برقرار باشد. باید قبل از طراحی پایگاه داده و جدول‌های آن موارد زیر را جهت ارتباط جدول‌های مورد نظر رعایت کرد:

- وجود فیلد مشترک دقیقاً از یک نوع و یک اندازه.
- فیلد مشترک در یکی از جدول‌ها کلید اصلی و در جدول دیگر کلید خارجی باشد.

**نکته** لزومی ندارد مقادیر موجود در جدول‌های مبدأ و مقصد یکسان باشد.

### معرفی دو جدول دیگر

از این به بعد ما در مثال‌های خود از ۲ جدول دیگر به غیر از جدول Student، به

نام‌های Courses (درس‌ها) و Selection (انتخاب واحد) به شرح زیر استفاده می‌کنیم:

جدول ۳-۵

Courses Table			
CoType	Credit	CoTitle	CourseID
نوع درس	تعداد واحد	عنوان درس	(کلید اصلی) کد درس

جدول ۳-۶

Selection Table				
Grade	Year	Term	CourseID	StudentID
نمره	سال تحصیلی	ترم تحصیلی	کد درس (کلید اصلی خارجی)	شماره دانش آموزی (کلید اصلی خارجی)

**نکته** • در تمام مثال‌های قبلی، ما در دستور select فقط نام ستون‌ها را به تنهایی ذکر می‌کردیم، زیرا در آن زمان، اطلاعات ما فقط از یک جدول استخراج می‌شد. اما در هنگام پیوند دو جدول و استفاده از چند جدول در دستور select باید نام ستون را به همراه نام جدول مربوط به آن ذکر کرد. این کار ۲ دلیل اصلی دارد:

- باعث تمایز ستون‌های مشترک در جدول‌ها از یکدیگر می‌شود و مشخص می‌کند که هر ستون مربوط به کدام جدول است.
- باعث خوانایی و دقت بیشتر برنامه می‌شود.

شکل کلی این دستور به صورت زیر است:

نام ستون . نام جدول

مثال: انتخاب ستون StudentID از جدول Student:

Student.StudentID

### ۱-۱۴-۳ مثال‌های پیوند جدول‌ها

در این قسمت با ارائه چندین مثال، انواع حالت‌های مختلف پیوند جدول‌ها را بررسی می‌کنیم. از داده‌های موجود در جداول زیر برای مثال‌ها استفاده می‌کنیم:

**توجه:** جدول انتخاب واحد نشان‌دهنده این است که هر دانش‌آموز چه واحدهای درسی را در چه ترم و سالی و با چه نمره‌ای گذرانده است.

جدول ۳-۷

Courses Table			
CoType	Credit	CoTitle	CourseID
عملی	3	پایگاه داده	1011
عملی	3	سخت‌افزار	1012
نظری	2	زبان تخصصی	1013

جدول ۳-۸

Selection Table				
Grade	Year	Term	CourseID	StudentID
16	88 - 89	2	1011	41252214
14	88 - 89	2	1011	10724113
17	88 - 89	1	1012	41252214
11	88 - 89	1	1012	10724113
13	88 - 89	2	1013	10254861
8	87 - 88	2	1011	10254861
19	87 - 88	1	1012	27365187
16	87 - 88	1	1013	27365187
9	87 - 88	2	1011	35654415
17	87 - 88	2	1013	35654415

شکل کلی پیوند ۲ جدول برای استخراج اطلاعات به صورت زیر است:

نام رکورد یا رکوردهای مورد نظر برای نمایش select

نام جدول‌ها from

برابر قرار دادن فیلدهای مشترک ۲ جدول where

; بقیه شرط‌های مورد نظر and

در این حالت ابتدا در دستور select نام ستون‌هایی را که می‌خواهیم از ۲ جدول نمایش دهیم، تعیین می‌کنیم. سپس نام ۲ جدول را در مقابل دستور from نوشته و در اولین شرط دستور where، نام فیلد مشترک را از هر ۲ جدول نوشته و آن‌ها را برابر هم قرار می‌دهیم. این شرط، شرط برقراری پیوند و تلفیق اطلاعات ۲ جدول است. در ادامه هم می‌توان شرط‌های دیگری را برای استخراج اطلاعات تعیین کرد. در مثال‌های بعدی این مسئله را بررسی می‌کنیم:

مثال: نام و نام خانوادگی و سال ورود دانش‌آموزانی را ارائه دهید که در ترم ۱ سال تحصیلی ۸۸-۸۹، درسی را با کد ۱۰۱۲ انتخاب کرده‌اند:  
کد:

```
select Student.Name , Student.Family , Selection.Term , Selection.Year  
from Student , Selection  
where Student.StudentID = Selection.StudentID  
and CourseID = 1012 and Term = 1 and Year = '88 - 89'  
order by Student.Family;
```

خروجی

Name	Family	Term	Year
Ehsan	Amiri	1	88 - 89
Ahmad	Rezaee	1	88 - 89

مثال: نام دروسی را ارائه دهید که دانش‌آموز با شماره دانش‌آموزی ۱۰۲۵۴۸۶۱ انتخاب کرده است:  
کد:

```
select Courses.CourseID , Courses.Co Title  
from Courses , selection  
where Courses.CourseID = Selection.CourseID  
and Selection.StudentID = 10254861 ;
```

خروجی

CourseID	CourseTitle
1011	پایگاه داده
1013	زبان تخصصی

مثال: نام و نام خانوادگی دانش‌آموزانی را ارائه دهید که درس با کد ۱۰۱۳ در سال تحصیلی ۸۷-۸۸ را با نمره بالاتر از ۱۵ گذرانده‌اند:  
کد:

```
select Student.Name , Student.Family  
from Student , Selection  
where Student.Studentid = Selection.Studentid  
and Selection.Courseid = '1013' and Year = '87 - 88' and Grade > 15 ;
```

خروجی

Name	Family
Sahar	Ahamdi
Hesam	Razavi

## ۲-۱۴-۳ پیوند بیش از دو جدول به یکدیگر

گاهی اوقات لازم است که اطلاعات مورد نیاز ما از ۳ جدول یا بیشتر استخراج شود. در این حالت باید تمام جدول‌ها را به هم پیوند دهیم، به این صورت که معمولاً از یک جدول سوم برای پیوند ۲ جدول دیگر استفاده می‌شود و دو به دو جدول‌هایی که با هم فیلد مشترک دارند را با ذکر شرط پیوند در دستور where به هم پیوند می‌دهیم. سپس بقیه شروط دلخواه را نیز ذکر می‌کنیم.

شکل کلی این حالت به این صورت است:

```
نام ستون‌های مورد نظر از جدول‌ها select  
نام تمام جدول‌ها from  
برابر قرار دادن فیلد مشترک جدول‌های ۱ و ۲ where  
برابر قرار دادن فیلدهای مشترک جدول‌های ۲ و ۳ and  
and ... ;
```

مثال: نام و نام خانوادگی دانش‌آموزانی را ارائه دهید که حداقل یک درس از نوع نظری را انتخاب کرده باشند:

کد:

```
select Students.Name , Students.Family , Courses.CoTitle , Courses.CoType  
from Students , Courses , Selections  
where Student.StudentID = Selection.StudentID  
and Courses.CourseID = Selection.CourseID ;  
and Courses.CoType = 'نظری' ;
```

خروجی

Name	Family	CoTitle	CoType
Zahra	Hoseini	زبان تخصصی	نظری
Sahar	Ahamadi	زبان تخصصی	نظری
Hesam	Razavi	زبان تخصصی	نظری

با دقت در اطلاعات جدول‌های اصلی متوجه درست بودن نتایج خروجی خواهید شد.



تحصیلی دانش‌آموزانی را استخراج کنیم که در ترم ۱ سال تحصیلی ۸۸-۸۹، درسی با کد ۱۰۱۲ را انتخاب کرده‌اند:  
**کد:**

```
select Student.Name , Student.Family , Seleciton.Term , Selection.Year
from Student
inner join Seleciton
on Student.StudentID = Selection.StudentID
where Selection.CourseID = 1012 and Term = 1 and Year = ' 88 - 89 ' ;
```

خروجی

Name	Family	Term	Year
Ehsan	Amiri	1	88 - 89
Ahmad	Rezaee	1	88 - 89

**مثال:** نام و شماره درس‌هایی را ارائه دهید که دانش‌آموز با شماره دانش‌آموزی ۱۰۲۵۴۸۶۱ انتخاب کرده است:  
**کد:**

```
select Courses.CourseID , Courses.Co Title
from Courses
inner join Selection
on Courses.CourseID = Selection.CourseID
where Selection.StudentID = 10254861 ;
```

خروجی

CourseID	CourseTitle
1011	پایگاه داده
1013	زبان تخصصی

## ۱۶-۳ Select های تودرتو

می‌توان درون یک دستور select، دستور select دیگری را تعریف کرد. در این حالت ابتدا دستور Select درونی اجرا شده و نتایج مورد نظر خود را که سطرهایی از جدول مربوط به آن هستند استخراج کرده و آن را به select بیرونی انتقال می‌دهد. سپس select

بیرونی نیز اجرا شده و نتایج استخراجی خود را با استفاده از فیلد مشترک بین ۲ جدول با سطرهای select درونی پیوند زده و سطرهای مطابق با شرطهای تعیین شده را نمایش می‌دهد.

 **نکته** به تعداد دستورات select تودرتو، سطح آن می‌گویند.

به علت پیچیدگی selectهای ۳ سطحی، ۴ سطحی و ... در این واحد کار فقط دستور select دو سطحی را بررسی می‌کنیم.

### ۱-۱۶-۳ selectهای ۲ سطحی

در این حالت یک دستور select درون دستور select دیگری تعریف می‌شود. نتایج خروجی فقط می‌تواند شامل سطرهایی از جدول Select بیرونی باشد. شکل کلی select دو سطحی به صورت زیر است:

نام ستونهای مورد نظر برای نمایش از جدول بیرونی select

نام جدول ۱ from

نام فیلد مشترک دو جدول (Select) in نام فیلد مشترک دو جدول where

نام جدول ۲ From

(... شرطها Where);

مثالهای قسمت‌های قبل پیوند جدول‌ها را با استفاده از Selectهای تودرتو بازنویسی می‌کنیم

مثال: نام و نام خانوادگی دانش‌آموزانی را ارائه دهید که در ترم ۱ سال تحصیلی ۸۹-

۸۸، درس با کد ۱۰۱۲ را انتخاب کرده‌اند:

کد:

```
select Name , Family
```

```
from Student
```

```
where StudentID in (select StudentID
```

```
from Selection
```

```
where CourseID = 1012 and Term = 1 and Year = ' 88 - 89 ' );
```

خروجی

Name	Family
Ehsan	Amiri
Ahmad	Rezaee



**نکته** تفاوت این حالت با مثال‌های قبلی در این است که ستون‌های نمایش داده شده فقط از یک جدول (جدول select بیرونی) نمایش داده می‌شوند.

**مثال:** شماره و نام درس‌هایی را ارائه دهید که دانش آموز با شماره دانش‌آموزی ۱۰۲۵۴۸۶۱ انتخاب کرده است:  
کد:

```
select CourseID , Co Title
from Courses
where CourseID in ( Select CourseID
                    from Selection
                    where StudentID = 10254861 ) ;
```

خروجی

CourseID	CourseTitle
1011	پایگاه داده
1013	زبان تخصصی

## ۱۷-۳ دستور union

از دستور Union برای ترکیب و ادغام دو یا چند ستون مختلف از ۲ یا چند جدول استفاده می‌شود. دستور union نتیجه را در یک ستون مشترک نمایش می‌دهد. در این دستور، نوع داده ستون‌های انتخاب شده برای ادغام باید یکسان باشند. دستور union در هنگام ترکیب فیلدها، در صورت برخورد با مقادیر تکراری، آن‌ها را حذف کرده و از هر مقدار یک نمونه را نمایش می‌دهد. برای مشاهده تمام مقادیر، حتی مقادیر تکراری باید از دستور union all استفاده کرد. شکل کلی استفاده از این دستور به صورت زیر است:

( نام فیلد ۱ انتخابی ) SQL Statement 1

union

( نام فیلد ۲ انتخابی ) SQL Statement 1 ;

**مثال:** در جدول ۱-۳، شماره دانش آموزی دانش آموزانی را نمایش دهید که یا دانش آموز رشته Electronic بوده و درس سخت افزار با کد ۱۰۱۲ را انتخاب کرده اند یا در درس زبان تخصصی با کد ۱۰۱۳ ثبت نام کرده اند:

```
select StudentID from Student
where Major = 'Electronic' and CourseID = 1012
union
select StudentID from Selection ;
where CourseID = 1013 ;
```

خروجی

StudentID
10254861
10724113
27365187
35654415

**نکته** می توان در دستور union بیش از یک ستون مجزا را تعریف کرد. فقط باید تعداد ستون ها و ترتیب نوع داده ای ستون های تعریف شده یکسان باشند. برنامه مقادیر ستون ها را به ترتیب با هم پیوند می دهد.

**مثال:** شماره دانش آموزی، نام و نام خانوادگی دانش آموزانی را نمایش دهید که در رشته Electronic تحصیل کرده یا حداقل یک درس از نوع نظری را انتخاب کرده باشند:

```
select StudentID , Name , Family
from Student
where Major = 'Electronic'
union
select StudentID , Name , Family
from Student , Selection , Courses
where Student.StudentID = Selection.StudentID
and Selection.CourseID = Courses.CourseID
and CoType = 'نظری' ;
```

### خروجی

StudentID	Name	Family
10254861	Zahra	Hoseini
107241113	Ehsan	Amiri
27365187	Sahar	Ahmadi
35654415	Hesam	Razavi

## ۱۸-۳ دستور drop

از این دستور برای حذف یک جزء در محیط پایگاه داده استفاده می‌شود.

### ۱۸-۳-۱ حذف اندیس (drop index)

از این دستور برای حذف یک اندیس موجود در یک فیلد جدول استفاده می‌شود. شکل کلی این دستور به شرح زیر است:

نام جدول on نام اندیس drop index  
مثال: drop index Rowup on Student ;

### ۱۸-۳-۲ حذف جدول یا پایگاه داده (drop DataBase یا drop table)

از این دستور برای حذف یک جدول در پایگاه داده یا یک پایگاه داده به صورت کامل استفاده می‌شود.  
دستور حذف جدول:

نام جدول drop table ;


دستور حذف پایگاه داده:

مثال drop table Student ;

نام پایگاه داده drop DataBase ;

مثال: اگر پایگاه داده‌ای به نام School داشته باشیم، دستور حذف آن به شکل زیر است:

drop DataBase School ;

**نکته**  برای حذف تمام اطلاعات موجود در یک جدول بدون حذف ستون‌های

آن از دستور زیر استفاده می‌شود:

نام جدول truncate table ;

truncate table School ; مثال

این دستور باعث خالی شدن جدول از اطلاعات می شود، بدون این که خود جدول و ساختارش تغییری کند.

## ۱۹-۳ دستور alter

از دستور alter برای اضافه کردن یا حذف کامل یک ستون از یک جدول موجود در پایگاه داده استفاده می شود.

### ۱-۱۹-۳ اضافه کردن یک ستون

برای اضافه کردن یک ستون (فیلد) جدید به یک جدول موجود از دستور کلی زیر استفاده می شود:

نام جدول alter table

نوع داده ستون نام ستون add

مثال: اضافه کردن یک ستون جدید به نام Age به جدول Student:

فیلدهای جدول Student قبل از اجرای دستور

Student Table					
StudentID	Name	Family	Major	Area	Grade

کد:

```
alter table Student
```

```
add Age Int ;
```

فیلدهای جدول Student پس از اجرای دستور

Student Table						
StudentID	Name	Family	Major	Area	Grade	Age

### ۲-۱۹-۳ حذف ستون

برای حذف یک ستون از یک جدول موجود به روش زیر عمل می شود:

نام جدول alter table

نام ستون drop column

مثال: حذف فیلد Age از جدول Students :

جدول Student قبل از اجرای دستور

Student Table						
StudentID	Name	Family	Major	Area	Grade	Age

کد:

alter table Student  
 drop column Age ;

جدول Student پس از اجرای دستور

Student Table					
StudentID	Name	Family	Major	Area	Grade

## ۲۰-۳ توابع درون ساخته SQL

زبان SQL دارای یکسری توابع درون ساخته (آماده) برای محاسبات روی داده‌های جداول و ارائه خروجی مناسب برای اهداف خاص است. در هنگام استفاده از توابع SQL باید به نکات زیر دقت کرد: باید نوع داده هر ستون (فیلد) جدول متناسب با عملکرد تابع مورد استفاده باشد. برای مثال توابعی که اعمال ریاضی انجام می‌دهند، باید به فیلدهایی از نوع عددی اعمال شوند. می‌توان نتایج عملکرد هر تابع را با استفاده از ویژگی alias که قبلاً در مورد آن صحبت کردیم، به صورت یک ستون جدید با عنوان دلخواه نمایش داد. برای مثال می‌توان با استفاده از تابع avg، میانگین نمرات هر دانش‌آموز را محاسبه کرده و آن را در یک ستون جدید به نام average نمایش داد. این مورد را در مثال‌های ذکر شده بررسی می‌کنیم. برنامه، نتایج خروجی هر تابع را به صورت یک ستون جدید نمایش می‌دهد. شکل کلی استفاده از توابع درون ساخته به صورت زیر است:

نام جدول from ( نام فیلد ) \*function select

; نام تابع = function \*

مثال: محاسبه و نمایش معدل دانش‌آموزان با استفاده از تابع Avg و نمایش آن تحت

ستون جدیدی به نام دلخواه:

average :

select avg ( Grade ) as Average from Student ;

**نکته** در هنگام استفاده از توابع SQL، چنانچه شرطها برای دستور select تعریف شده باشد، تابع فقط مقادیر فیلدهایی از هر ستون را در محاسبه دخالت می دهد که با شروط ارائه شده سازگاری داشته باشند.

برای مثال وقتی می گوئیم که معدل دانش آموزان رشته Computer را در درس پایگاه داده محاسبه کنید، برنامه فقط میانگین نمرات درس پایگاه داده دانش آموزان را محاسبه می کند و به نمرات سایر دروس کاری ندارد.

**نکته** چنانچه توسط ویژگی alias، عنوانی برای ستون جدید ایجاد شده تعیین نکنید، خود برنامه به صورت خودکار یک نام پیش فرض به آن اختصاص می دهد.

### ۱-۲۰-۳ تابع avg

تابع avg برای محاسبه میانگین اعداد موجود در یک فیلد به کار می رود. این تابع اعداد را با هم جمع کرده و آنها را به طور اتوماتیک بر تعدادشان تقسیم می کند و نتیجه را به عنوان خروجی برمی گرداند.

**نکته** این تابع باید با فیلدهایی که داده آنها از نوع عددی است، اعمال شود.

**مثال:** معدل نمرات دانش آموزان را از جدول Student محاسبه کرده و در ستونی به نام معدل نمایش دهید:

کد:

```
select avg ( Students.Grade ) as معدل from Student
```

خروجی

معدل
16.8

**مثال:** معدل نمرات دانش آموزان رشته Computer را در درس پایگاه داده محاسبه کرده و در ستونی به نام معدل نمایش دهید:

کد:

```
select avg ( Selection.Grade ) as معدل
```

```
from Students,Selection,Courses
where Students.StudentID = Selection.StudentID
and Selection.CourseID = Courses.CourseID
and Major = 'Computer'
and CoTitle = 'پایگاه داده' ;
```

خروجی

معدل
11.5

### ۲-۲۰-۳ تابع sum

تابع sum، مجموع مقادیر اعداد در یک فیلد را محاسبه کرده و به عنوان خروجی برمی گرداند.

**نکته** این تابع باید با فیلدهایی که داده آنها از نوع عددی است، به کار رود.

**مثال:** از جدول انتخاب واحد (Selection) مجموع نمرات دانش آموزان را در درس پایگاه داده با کد ۱۰۱۱ محاسبه کنید:  
کد:

```
select Sum ( Selection.Grade ) as مجموع نمرات پایگاه داده
from Selection,Courses
and Selection.CourseID = Courses.CourseID
and Selection.CourseID = 1011 ;
```

خروجی

مجموع نمرات پایگاه داده
47

**مثال:** از جدول انتخاب واحد، مجموع نمرات دانش آموز حسام رضوی را محاسبه کنید:  
کد:

```
select Sum ( Selection.Grade ) as مجموع نمرات حسام رضوی
from Student,Selection
where Student.StudentID = Selection.StudentID
```

and Name = ' Hesam '

and Family = ' Razavi ' ;

خروجی

مجموع نمرات حسام رضوی

26

### ۳-۲۰-۳ تابع min

این تابع، کمترین مقدار موجود در بین مقادیر فیلدهای یک ستون را برمی گرداند.

**نکته** این تابع با ستون‌هایی که داده آن‌ها از نوع عددی یا متنی باشد، می‌تواند به کار رود. در فیلدها با مقادیر عددی تابع کوچک‌ترین عدد و در فیلدها با مقادیر متنی، تابع کلمه‌ای را که به ترتیب حروف الفبا از آ تا ی در فارسی و A تا Z در انگلیسی در بالاترین رده باشد برمی گرداند.

**مثال:** کوچک‌ترین نام خانوادگی از نظر حروف الفبایی را از جدول Student نمایش

دهید:

کد:

select min (Family) as Family from Student ;

خروجی

Family

Ahmadi

**مثال:** کمترین نمره دانش‌آموزان را از جدول Selection نمایش دهید:

کد:

select min ( Grade ) as کمترین نمره from Selection ;

خروجی

کمترین نمره

8

### ۳-۲۰-۴ تابع max

این تابع، بیشترین مقدار موجود در بین مقادیر فیلدهای یک ستون را برمی گرداند.

**نکته** این تابع با ستون‌هایی که داده آن‌ها از نوع عددی یا متنی باشد، می‌تواند به کار رود. در فیلدها با مقادیر عددی تابع بزرگ‌ترین عدد و در فیلدها با مقادیر متنی، تابع کلمه‌ای را که به ترتیب حروف الفبا از آ تا ی در فارسی و A تا Z در انگلیسی در آخرین رده باشد برمی گرداند.



مثال: آخرین نام خانوادگی دانش‌آموزان را از جدول Student نمایش دهید:

کد:

select max (Family) as Family from Student;

خروجی

Family

Rezaee

مثال: بیشترین نمره دانش‌آموزان را از جدول Selection نمایش دهید:

کد:

select max (Grade) as بیشترین نمره from Selection ;

خروجی

بیشترین نمره

19

### ۵-۲۰-۳ تابع first

این تابع مقدار اولین رکورد را در یک فیلد برمی‌گرداند. ترتیب در اینجا، همان ترتیب قرارگیری رکوردها در جدول است.

### ۶-۲۰-۳ تابع last

این تابع مقدار آخرین رکورد را در یک فیلد برمی‌گرداند. ترتیب در اینجا، همان ترتیب قرارگیری رکوردها در جدول است.

### ۷-۲۰-۳ تابع count

این تابع تعداد سطرهای موجود در یک فیلد را شمرده و مقدار آن را برمی‌گرداند. به این تابع، تابع سطر شمار هم می‌گویند.

مثال: این تابع فیلدهایی را که مقدار آن‌ها خالی یا تکراری باشد، نمی‌شمارد. برای شمارش تمام فیلدها (حتی تکراری) باید قبل از نام ستون در دستور count عبارت distinct ذکر شود.

مثال: تعداد درس‌هایی که دانش‌آموز با شماره دانش‌آموزی ۳۵۶۵۴۴۱۵ قبول شده را

ارائه دهید:

کد:

select count (CourseID) as تعداد دروس

from Selection

where StudentID = 35654415

and Grade > 10 ;

خروجی

تعداد دروس

1

مثال: تعداد دانش‌آموزانی که درس پایگاه داده را انتخاب کرده‌اند اعلام کنید:

کد:

select count (distinct Selection.CourseID) as تعداد دانش‌آموزان

from Selection , Courses

where Selection.CourseID = Courses.CourseID

and CoTitle = ' پایگاه داده ' ;

خروجی

تعداد دانش‌آموزان

4

### ۸-۲۰-۳ دستور group by

در هنگام استفاده از برخی توابع درون ساخته SQL که عمل محاسبه (مثل مجموع و میانگین) را روی داده‌ها انجام می‌دهند، این مشکل وجود دارد که این توابع قادر به جدا کردن و متمایز کردن اطلاعات موجود در دو ستون نسبت به هم نیستند و نتایج محاسبات را به صورت کلی برای همه آن‌ها در نظر می‌گیرند.

از دستور group by برای دسته‌بندی یک ستون بر حسب مقادیر مشابه فیلدهای یک ستون دیگر استفاده می‌شود.

شکل کلی استفاده از این دستور به صورت زیر است:

.... و نام فیلد ۲ و نام فیلد ۱ select

from نام جدول

; نام فیلد group by

مثال ۱: فرض کنید جدولی به صورت زیر برای نمرات ۲ دانش‌آموز داریم، می‌خواهیم

مجموع نمرات هر یک از این دو دانش‌آموز را نمایش دهیم:

جدول ۳-۹

Class Table	
Grade	Name
18	Ali Ahmadi
17	Reza Hoseini
20	Ali Ahmadi
13	Reza Hoseini

کد:

select sum (Grade) as مجموع  
 from Class ;

در صورت عدم استفاده از دستور group by در تابع Sum خروجی زیر را خواهیم داشت:  
 خروجی

مجموع نمرات
68

• مسلماً خروجی فوق مورد نظر نیست. می توان با استفاده از دستور group by، مجموع نمرات را برای دانش آموزان بر حسب ستون Name به صورت جدا محاسبه کرد و به این صورت نمایش داد:  
 کد:

select Name ,sum (Grade) as مجموع نمرات  
 from Class  
 group by Name ;

خروجی	
Name	مجموع نمرات
Ali Ahmadi	38
Reza Hoseini	30

مثال ۲: از جدول Student و Selection مجموع نمرات هر دانش آموز را به همراه نام و نام خانوادگی که بر حسب نام خانوادگی مرتب شده باشد نمایش دهید:

کد:

select Name , Family , sum (Selection.Grade) as مجموع نمرات

from Student , Selection  
 where Student.StudentID = Selection.StudentID  
 group by Name , Family  
 order by Family;

خروجی

Name	Family	مجموع نمرات
Sahar	Ahmadi	35
Ehsan	Amiri	25
Zahra	Hoseini	21
Hesam	Razavi	26
Ahmad	Rezaee	33

### ۹-۲۰-۳ دستور having

از این دستور برای افزودن شرط به توابع درون ساخته SQL استفاده می شود، زیرا از دستور where نمی توان برای کار با مقادیر خروجی توابع درون ساخته SQL استفاده کرد. به عبارت دیگر having برای اعمال شرط به ستون‌ها اعمال می شود و همان کاری را می کند که where در رکوردها انجام می دهد. با مثال‌های بعدی استفاده از دستور having را بررسی می کنیم:

**نکته** دستور having معمولاً با group by می آید.

**مثال:** از جدول Selection و Student مجموع نمرات و نام و نام خانوادگی دانش آموزانی را که مجموع نمرات آنها بیش از ۲۵ است نمایش دهید:  
 کد:

select Name , Family , sum (Selection.Grade) as مجموع نمرات  
 from Student , Selection  
 where Student.StudentID = Selection.StudentID  
 group by Name , Family  
 having Sum (Selection.Grade) > 25  
 order by Family;

خروجی

Name	Family	مجموع نمرات
Sahar	Ahmadi	۳۵
Ahmad	Rezaee	۳۳

**مثال:** از جدول Selection و Courses نام دروسی را ارائه دهید که ۴ بار توسط دانش‌آموزان انتخاب شده باشد:  
**کد:**


```
select CoTitle , Count (Selection.CourseID) as تعداد انتخاب
from Selection , Courses
where Selection.CourseID = Courses.CourseID
group by CoTitle
having Count (Selection.Grade) = 4 ;
```

خروجی

CoTitle	تعداد انتخاب
پایگاه داده	4

**select into دستور ۳-۲۰-۱۰**

- از دستور select into در موارد زیر استفاده می‌شود:
- ایجاد یک Backup (نسخه پشتیبان) از یک جدول
  - ایجاد یک آرشیو از رکوردهای یک جدول
  - قرار دادن برخی از رکوردها یا فیلدهای مورد نظر از یک جدول در یک جدول جدید
  - ایجاد یک نسخه پشتیبان از کل یک پایگاه داده در یک پایگاه داده جدید
- شکل کلی استفاده از این دستور به صورت زیر است:
- ```
select [نام یک پایگاه داده جدید in] نام جدول جدید into نام فیلدهای مورد نظر
from نام جدول مبدأ ;
```

**نکته**  ذکر نام یک پایگاه داده جدید در دستور select into اختیاری است. چنانچه نامی در این قسمت ذکر نشود، برنامه نسخه پشتیبان را در همان پایگاه داده موجود ایجاد می‌کند و چنانچه نامی ذکر شود، برنامه نسخه پشتیبان را در پایگاه داده ذکر شده ایجاد می‌کند.

**مثال:** ایجاد یک Backup از جدول Student در یک جدول جدید:

```
select * into Student_Backup
from Student ;
```

**مثال:** ایجاد یک Backup از جدول Student در یک جدول دیگر در یک پایگاه داده جدید به نام Backup.mdb:

```
select * into Student_Backup in «Backup.mdb»
```

```
from Student ;
```

توجه: در مثال‌های زیر از اطلاعات جداول Selection، Courses، و Student که در صفحات قبل معرفی کردیم، استفاده می‌کنیم.

**نکته** همچنین می‌توان فقط یک یا چند ستون یا رکورد را از یک جدول در یک جدول جدید وارد کرد. برای این کار باید نام آن فیلدها را پس از دستور select تعیین کرده و برای رکوردهای مورد نظر نیز شروط مورد نظر تعریف شود.

**مثال:** فیلدهای Name و Family را از جدول Student در یک جدول جدید به نام Names وارد کنید:  
کد:

```
select Name , Family into Names
```

```
from Student ;
```

بعد از اجرای دستور فوق جدول جدیدی به نام Names با مشخصات بعد ایجاد خواهد شد.

خروجی

| Name  | Family  |
|-------|---------|
| Ahmad | Rezaee  |
| Ehsan | Amiri   |
| Zahra | Hoseini |
| Sahar | Ahmadi  |
| Hesam | Razavi  |

**مثال:** از جدول Student، مشخصات دانش‌آموزانی که نمره آن‌ها بیش از ۱۷ است را در جدول جدیدی به نام Highscores وارد کنید:  
کد:

```
select * into Highscores
```

```
from Student
```

```
where Grade > 17 ;
```

بعد از اجرای دستور فوق جدول جدیدی به نام Highscores Table با مشخصات زیر ایجاد خواهد شد.

#### خروجی

| Highscores Table |       |        |            |        |       |
|------------------|-------|--------|------------|--------|-------|
| StudentID        | Name  | Family | Major      | Area   | Grade |
| 41252214         | Ahmad | Rezaee | Computer   | Tehran | 18    |
| 35654415         | Hesam | Razavi | Electronic | Tehran | 19    |

**مثال:** مشخصات نام، نام خانوادگی، نام درس و نمره دانش‌آموزانی که در جدول انتخاب واحد، نمره آن‌ها بیش از ۱۵ بوده را در یک جدول جدید به نام Highgrades وارد کنید:  
کد:

```
select Name , Family , CoTitle , Selection.Grade Into Highgrades
from Student , Selection , Courses
where Student.StudentID = Selection.StudentID
and Selection.CourseID = Courses.CourseID
and Grade > 15 ;
```

بعد از اجرای دستور فوق جدول جدیدی به نام Highgrades Table با مشخصات زیر ایجاد خواهد شد.

#### خروجی

| Highgrades Table |        |             |       |
|------------------|--------|-------------|-------|
| Name             | Family | CoTitle     | Grade |
| Ahmad            | Rezaee | پایگاه داده | 16    |
| Sahar            | Ahmadi | مباحث ویژه  | 19    |
| Ahmad            | Rezaee | مباحث ویژه  | 17    |
| Sahar            | Ahmadi | زبان تخصصی  | 16    |
| Hesam            | Razavi | زبان تخصصی  | 17    |

## واژه‌نامه

|         |          |
|---------|----------|
| Average | میانگین  |
| Count   | تعداد    |
| First   | اولین    |
| Group   | گروه     |
| Having  | داشتن    |
| Last    | آخرین    |
| Maximum | حداکثر   |
| Minimum | حداقل    |
| Sum     | حاصل جمع |

## خلاصه مطالب

- زبان SQL یک زبان ساخت یافته برای اتصال و ارتباط و کار با بانک‌های اطلاعاتی است.
- این زبانه با ارائه دستورات مختلف و کاربردی در تمام سیستم‌های مدیریت بانک‌های اطلاعاتی قابل دسترسی و استفاده است.
- برای اجرای دستورات SQL و مشاهده عملکرد آن‌ها، باید از یک سیستم مدیریت پایگاه داده‌ها استفاده شود.
- دستور create برای ایجاد یک جزء جدید در سیستم‌های پایگاه داده استفاده می‌شود.
- دستور select برای انتخاب و استخراج اطلاعات مورد نظر از یک یا چند جدول و مشاهده نتایج استفاده می‌شود.
- دستور insert برای وارد کردن و ایجاد یک رکورد جدید در جدول استفاده می‌شود.
- دستور update برای تغییر اطلاعات موجود در یک فیلد جدول استفاده می‌شود.
- دستور delete برای حذف اطلاعات یک رکورد در جدول استفاده می‌شود.



## آزمون نظری

۱ - برای ایجاد یک پرسش و استخراج اطلاعات جدول از ..... استفاده می شود.

الف-DML      ب- رکورد      ج- Query      د- DDL

۲- کدام یک از گزینه های زیر در عملیات جستجو، باعث بالا رفتن سرعت جستجو می شود؟

الف- Query      ب- Index      ج- MySQL      د- Create

۳ - دستور Create برای ایجاد کدام یک از گزینه های زیر به کار نمی رود؟

الف- پایگاه داده      ب- جدول      ج- اندیس      د- رکورد

۴- کدام یک از گزینه های زیر برای انتخاب تمام فیلدهای یک جدول با نام tbl1 به کار می رود؟

الف- select tbl1      ب- select fields from tbl1

ج- select \* from tbl1      د- select tbl1\*

۵ - در دستور Where برای ادغام چند شرط، از چه دستوری استفاده می شود؟

الف-add

ب-or

ج-like

د- استفاده از چند شرط با نوشتن چند دستور where امکان پذیر است.

۶ - برای جستجوی افرادی در جدول که قسمتی از نام آنها محمد است چه عبارتی

مقابل دستور Where نوشته می شود؟

الف- " محمد" like      ب- «محمد» like

ج- " محمد" like      د- "محمد" like

۷ - برای درج یک رکورد در جدول tbl1 با فیلدهای f1 و f2 کدام یک از گزینه های

زیر به کار می رود؟

الف- select f1 , f2 from tbl1

ب- ("مقدار فیلد دوم" و "مقدار فیلد اول") values

ج- select into tbl1 (f1 , f2)

د- ("مقدار فیلد دوم" و "مقدار فیلد اول") values

ج- insert into tbl1 (f1 , f2)

د- ("مقدار فیلد دوم" و "مقدار فیلد اول") values

د- insert f1 , f2 from tbl1

("مقدار فیلد دوم" و "مقدار فیلد اول") values

۸- برای جستجوی یک رکورد در جدول tbl1 که مقدار دقیق فیلد آن را می دانید از

چه دستوری استفاده می کنید؟

الف- select \* from tbl1

(... و "مقدار ۲" و "مقدار ۱") where f1 in

ب- select f1 from tbl1

(... و "مقدار ۲" و "مقدار ۱") where in

ج- search \* from tbl1

(... و "مقدار ۲" و "مقدار ۱") where f1 in

د- select f1 from tbl1

(... و "مقدار ۲" و "مقدار ۱") alias in

۹- برای انتخاب ستون Name از جدول Student کدام عبارت به کار می رود؟

الف- name.student      ب- student name

ج- student\*name      د- name from student

۱۰- برای ادغام دو یا چند ستون مختلف از ۲ یا چند جدول و نشان دادن آنها در

یک ستون مشترک از چه دستوری استفاده می شود؟

الف- select      ب- merge      ج- join      د- union

۱۱- برای حذف اجزای پایگاه داده از چه دستوری نمی توان استفاده کرد؟

الف- delete      ب- insert      ج- drop      د- alter

۱۲- تابع sum و avg به ترتیب برای محاسبه چه عملی در پایگاه داده به کار می روند؟

الف- مجموع- مجموع      ب- میانگین- مجموع

ج- مجموع- میانگین      د- میانگین- میانگین

## آزمون عملی

- ۱ - در Command Line یک پایگاه داده با عنوان Library ایجاد کنید.
- ۲ - برای موجودیت‌های کتاب و نویسنده، صفاتی که می‌توانند در محیط عملیاتی کتابخانه در نظر گرفته شوند را بررسی کنید و هر کدام از این دو موجودیت را به صورت جدول پیاده‌سازی کنید.
- ۳ - درون پایگاه داده Library که در تمرین ۱ ساخته‌اید دو جدول کتاب و نویسنده را با فیلدهای تعیین شده بسازید.
- ۴ - درون هر کدام از جدول‌های تمرین ۲ اطلاعات مربوط به چند رکورد را درج کنید.
- ۵ - تمام رکوردهای مربوط به جدول کتاب را نمایش دهید.
- ۶ - تمام رکوردهای مربوط به جدول نویسنده را که حرف اول نام نویسنده آن‌ها "H" است را نمایش دهید.
- ۷ - در جدول کتاب اسامی نویسندگان را نمایش دهید، طوری که اسامی تکراری حذف شوند.
- ۸ - در جدول کتاب اطلاعات مربوط به یک رکورد با عنوان "صفحات وب" را وارد کنید (سایر فیلدها را به دلخواه وارد نمایید)، سپس اطلاعات همان رکورد را تغییر دهید طوری که عنوان کتاب با مقدار جدید "طراحی صفحات وب مقدماتی" جایگزین شود.
- ۹ - در جدول نویسنده رکوردی که عنوان نویسنده آن با حرف "a" شروع می‌شود را حذف کنید.

## نصب Apache

### واحد کار چهارم

پس از مطالعه این واحد کار از فراگیر انتظار می‌رود که:

- ۱ - با Apache و کاربرد آن آشنا باشد.
- ۲ - امکانات لازم برای نصب و به‌کارگیری Apache را بشناسد.
- ۳ - با اصول نصب Apache و انجام تنظیمات آن آشنا باشد.
- ۴ - با اصول برقراری اتصال با Apache آشنا باشد.

| زمان (ساعت) |      |
|-------------|------|
| عملی        | نظری |
| ۸           | ۴    |

## مقدمه

همان‌طور که قبلاً در طراحی صفحات وب مقدماتی آموختید، Apache یک سرور دهنده صفحات وب مطمئن، کارا و متن‌باز است که علاوه بر ویندوز در لینوکس نیز به راحتی قابل استفاده است، البته در ابتدا این سرور دهنده مختص لینوکس طراحی شده بود که به مرور امکانات پشتیبانی در ویندوز نیز به آن افزوده شد. (این فصل به صورت مستقیم نرم‌افزار Apache را نصب می‌کند و کاربر باید تمام تنظیمات را به صورت دستی (Customize) انجام دهد. ولی به‌طور عمومی می‌توان با استفاده از نرم‌افزار WAMP که در فصل بعدی بیان شده تمام تنظیمات دستی توسط خود نرم‌افزار انجام گیرد.)

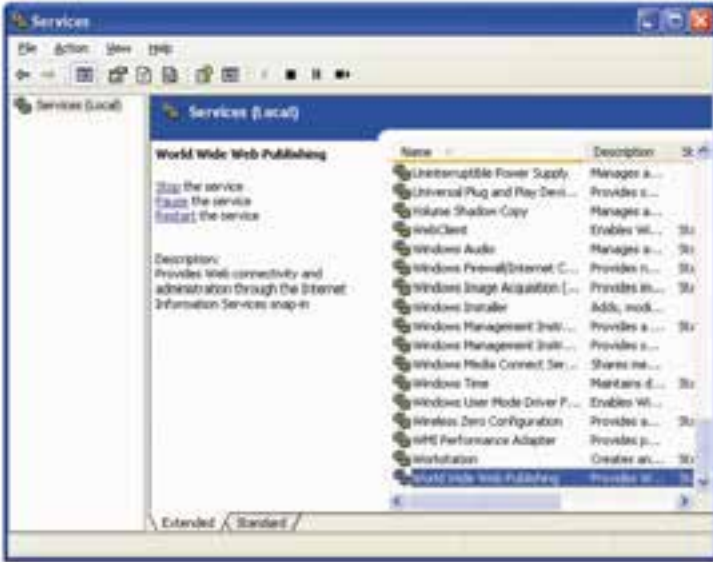
**تذکر:** در این فصل روش نصب و انجام تنظیمات سرور دهنده وب Apache بررسی شده است و کاربر برای استفاده از آن باید تمام تنظیمات را به‌طور دستی انجام دهد. اما به‌طور کلی می‌توانید با استفاده از نرم‌افزار WAMP که در واحد کار بعدی آموزش داده شده است، تمام تنظیمات را به‌طور خودکار توسط سیستم انجام دهید.

## ۱-۴ نصب Apache

قبل از شروع به نصب Apache باید موارد مهمی را در نظر بگیرید:  
اگر می‌خواهید Apache به عنوان سرور دهنده وب سایت شما باشد، باید سرورهای مشابه را حذف یا غیرفعال کنید، یکی از این سرورهای که قبلاً نیز با تنظیم آن آشنا شده‌اید، سرور IIS است، اگر این سرور را قبلاً فعال کرده‌اید، با استفاده از مراحل زیر آن را غیرفعال کنید، سپس مراحل نصب را ادامه دهید.  
از پنجره Control Panel وارد Administrative Tools شوید و پس از آن برنامه Services را اجرا کنید.

### 1-Open Source

متن باز به این معناست که دسترسی به منبع آزاد است و علاوه بر این، منبع به‌طور رایگان در اختیار کاربران قرار می‌گیرد.



شکل ۴-۱

گزینه World Wide Web Publishing را انتخاب کرده، سپس روی دکمه (Stop) ■ از نوار ابزار کلیک کنید تا سرویس IIS متوقف شود.  
CD نصب Apache را درون درایو گذاشته و فایل مربوط به نصب آن را اجرا کنید.



شکل ۴-۲

روی دکمه Next کلیک کنید.



شکل ۳-۴

سپس دکمه رادیویی I accept the terms in the license agreement را انتخاب کرده، روی دکمه Next کلیک کنید.



شکل ۴-۴

در کادر محاوره شکل ۴-۴ روی دکمه Next کلیک کنید.



شکل ۴-۵

در کادرمحاوره شکل ۴-۵ در کادر متنی Network Domain باید دامنه شبکه نوشته شود، اما چون در این بحث راه اندازی Apache را برای سیستم به طور محلی انجام می دهیم، مهم نیست که دقیقاً چه چیزی وارد شود. در کادر متنی Server Name، نام سرویس دهنده و در انتها آدرس ایمیل مدیر سایت وارد می شود. روی دکمه Next کلیک کنید.



شکل ۴-۶

در کادرمحاوره شکل ۴-۶ دکمه رادیویی Typical را انتخاب کرده، روی دکمه Next کلیک کنید.





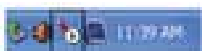
شکل ۴-۷

اگر تمایل دارید مسیر نصب Apache را از حالت پیش فرض تغییر دهید، روی دکمه Change کلیک کنید و مسیر موردنظر را انتخاب کنید. سپس روی دکمه Next کلیک نمایید.



شکل ۴-۸

در کادر محاوره شکل ۴-۸ برای انجام عملیات نصب روی سیستم، روی دکمه Install کلیک کنید.



شکل ۴-۹

پس از نصب Apache، آیکن این برنامه روی ناحیه تذکر (notification) از نوار وظیفه نمایان می شود.

آیکن فلش سبز رنگ به این معنا است که سرویس دهنده Apache در حال اجرا است. برای متوقف کردن سرویس Apache روی آیکن آن کلیک کنید، ماوس را روی عنوان Apache نگه داشته، سپس روی گزینه Stop کلیک کنید.



شکل ۱۰-۴

برای اجرای مجدد سرویس Apache در شکل ۱۰-۴ روی Start کلیک کنید.

### ۱-۱-۴ در لینوکس

برای نصب و راه اندازی Apache در لینوکس، باید فایل Apache CD مخصوص این سیستم عامل را تهیه کنید یا از اینترنت، فایل های مربوطه را دریافت نمایید و آن را روی سیستم خود نصب نمایید.

سرویس دهنده وب Apache پس از نصب در بیشتر نسخه های لینوکس از مسیر `/var/www/html/` برای نگهداری فایل های وب استفاده می کند، اما فایل های مربوط به پیکربندی سرویس دهنده Apache را در مسیر `/etc/httpd/conf/` ذخیره می نماید.

فایل اصلی پیکربندی Apache با نام `httpd.conf` نیز که در مسیر فوق ذخیره می شود، می تواند برای تنظیمات عملکرد Apache روی لینوکس به کار برده شود.

در این واحد کار و واحد کار اول با نصب و پیکربندی Apache و MySQL آشنا شدید. اما در عمل توصیه می شود برای شروع آموزش PHP سرویس دهنده WAMP را روی سیستم خود راه اندازی کنید، در واحد کار پنجم روش نصب و استفاده از این سرویس دهنده آموزش داده شده است. مزیت استفاده از WAMP این است که تمام امکانات لازم برای برنامه نویسی PHP و اجرای صفحات آن را به طور خودکار راه اندازی می کند و نیازی به نصب مجزای Apache، MySQL و PHP نیست.

## واژه‌نامه

Configuration

پیکربندی

Notification

تذکر

## خلاصه مطالب

Apache به عنوان سرویس‌دهنده صفحات وب می‌تواند در طراحی صفحات وب پویا مؤثر باشد، برای نصب و راه‌اندازی Apache باید سرویس‌های مشابه از قبیل IIS حذف یا غیرفعال شوند.

پس از نصب Apache، آیکن آن در ناحیه تذکر از نوار وظیفه ظاهر می‌شود. سرویس‌دهنده وب Apache در بیشتر نسخه‌های لینوکس از مسیر `/var/www/html/` استفاده می‌کند.

فایل‌های مربوط به پیکربندی سرویس‌دهنده Apache در مسیر `/etc/httpd/conf/` ذخیره و نگهداری می‌شوند.

## آزمون نظری

۱ - Apache به همراه کدام یک از گزینه‌های زیر نمی‌تواند به‌طور هم‌زمان نصب و

اجرا شود؟

الف - PHP  
ب - Internet Explorer

ج - IIS  
د - الف و ج

۲ - برای متوقف کردن سرویس IIS کدام گزینه در برنامه Services باید متوقف شود؟

الف - Internet Information Services  
ب - World Wide Web Publishing

ج - IIS  
د - World Wide Web Services

۳ - سرویس‌دهنده Apache در لینوکس فایل‌های وب را در چه مسیری ذخیره می‌کند؟

الف - var/www/web/  
ب - var/www/conf/

ج - var/www/html/  
د - etc/httpd/conf/

۴ - سرویس‌دهنده Apache در لینوکس از چه مسیری برای نگهداری فایل‌های

پیکربندی استفاده می‌کند؟

الف - etc/httpd/html/  
ب - var/conf/httpd/

ج - var/httpd/conf/  
د - etc/httpd/conf/

۵ - فایل اصلی پیکربندی Apache کدام است؟

الف - httpd.conf  
ب - conf httpd/

ج - etc.conf  
د - var.conf/

## آزمون عملی

- ۱ - سرویس دهنده IIS را روی سیستم خود متوقف کنید.
- ۲ - سرویس دهنده Apache را روی سیستم خود راه اندازی کنید.

## نصب PHP

### واحد کار پنجم

پس از مطالعه این واحد کار از فراگیر انتظار می‌رود که:

- ۱ - مفهوم زبانهای Client Side و Server Side را بداند.
- ۲ - با PHP آشنایی کلی داشته و کاربرد آن را بداند.
- ۳ - امکانات لازم برای نصب PHP را بشناسد.
- ۴ - بتواند PHP را در محیطهای ویندوز و لینوکس راهاندازی کند.
- ۵ - با پیکربندی PHP، MySQL و Apache از طریق فایل‌های php.ini و httpd.conf آشنا باشد.
- ۶ - بتواند برنامه PHP را تست و بررسی نماید.

| زمان (ساعت) |      |
|-------------|------|
| عملی        | نظری |
| ۸           | ۴    |

## مقدمه

در سیستم عامل ویندوز برنامه‌های متعددی وجود دارند که PHP را به همراه برنامه‌ها و نرم‌افزارهای جانبی مورد نیاز برای برنامه‌نویسان راه‌اندازی کرده و تمام سرویس‌های مربوط به آن‌ها را ارائه می‌دهند. چند مورد از پرستفاده‌ترین این برنامه‌ها عبارتند از: WAMP، EasyPHP، PHPtraid، و SQLware. هر کدام از این برنامه‌ها را نصب کنید، برنامه‌های PHP، Apache، و MySQL نیز نصب می‌شوند.

## ۱-۵ آشنایی با زبان PHP

PHP یک زبان اسکریپت‌نویسی است که برای محیط وب نوشته شده است، این زبان برای نوشته شدن نیاز به محیط خاصی ندارد و کدهای آنرا حتی می‌توانید در محیط ساده Notepad بنویسید.

زبانهای برنامه‌نویسی و اسکریپت‌نویسی وب به طور کلی به دو دسته اصلی تقسیم می‌شوند:

Client Side •

Server Side •

Client Side (سمت سرویس‌گیرنده) به برنامه‌هایی گفته می‌شود که روی کامپیوتر کاربر اجرا می‌شوند مانند HTML و جاوا اسکریپت. Server Side (سمت سرویس‌دهنده) برنامه‌هایی هستند که روی سرویس‌دهنده اجرا می‌شوند و سرویس‌گیرنده فقط نتایج اجرا را مشاهده می‌کند، مانند PHP، ASP و...

## ۱-۱-۵ نصب PHP در لینوکس

یکی از مهم‌ترین ویژگی‌های زبان PHP قابلیت اجرا روی سیستم‌های عامل مختلف است، برای استفاده از امکانات مختلف PHP نیز می‌توانید از برنامه‌های سرویس‌دهنده مختلف از قبیل WAMP استفاده نمایید، اما اگر بخواهید بدون استفاده از این‌گونه برنامه‌ها، از تمام امکانات PHP بهره بگیرید باید حداقل امکانات زیر را روی لینوکس نصب نمایید:

• Apache

• MySQL

برای نوشتن کدهای PHP در لینوکس نیز نیاز به محیط کدنویسی مخصوص نیست و حتی می‌توان از ویرایشگرهای متنی ساده استفاده کرد.

## ۲-۱-۵ نصب PHP در ویندوز

همان‌طور که پیش از این اشاره شد، برای شروع برنامه نویسی با PHP، می‌توانید از سرویس‌دهنده‌های مختلفی استفاده کنید، از بین این گونه برنامه‌ها، سرویس‌دهنده WAMP از محبوبیت بسیار زیادی بین برنامه‌نویسان PHP برخوردار است. در این واحد کار با روش نصب این برنامه آشنا می‌شوید.

CD مربوط به برنامه WAMP را در درایو مربوطه قرار دهید.

اگر CD این برنامه را در اختیار ندارید می‌توانید WAMP را از اینترنت دانلود نمایید، یکی از آدرس‌های دریافت این برنامه عبارت است از:

<http://www.en.wampserver.com/download.php>

روی آیکن مربوط به نصب WAMP دابل کلیک کنید.



شکل ۱-۵

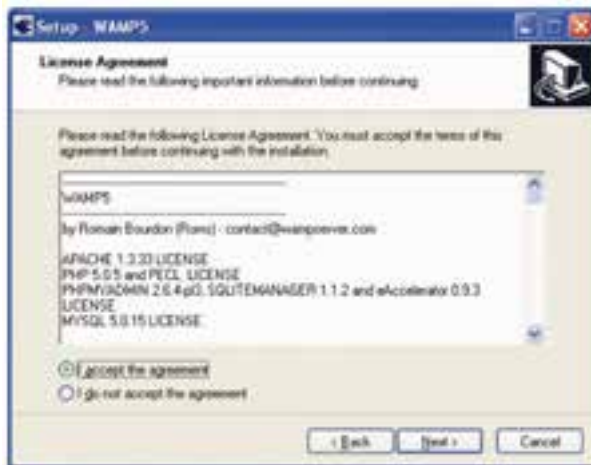
پنجره شکل ۲-۵ ظاهر خواهد شد.





شکل ۵-۲

در این پنجره روی دکمه Next کلیک کنید.



شکل ۵-۳

در پنجره شکل ۵-۳، دکمه رادیویی I accept the agreement را انتخاب کرده و روی دکمه Next کلیک کنید.

در کادر متنی شکل ۵-۴ مسیر نصب برنامه مشخص شده است که به طور پیش فرض پوشه wamp در درایو مربوط به ویندوز انتخاب شده است، چنانچه میخواهید این مسیر پیش فرض را تغییر دهید روی دکمه Browse کلیک کنید و مسیر دلخواه خود را برای نصب این برنامه تعیین کنید.



شکل ۴-۵

توصیه میشود مسیر نصب WAMP را یک مسیر ساده انتخاب کنید که از این پس برای دسترسی به مسیر آن دچار مشکل و سردرگمی نشوید.

شکل ۵-۵ نام پوشه‌های را نشان میدهد که WAMP پس از نصب در منوی All Programs درون آن قرار می‌گیرد.

در صورت دلخواه میتوانید این پوشه و نام آنرا با استفاده از دکمه Browse تغییر دهید. روی دکمه Next کلیک کنید.



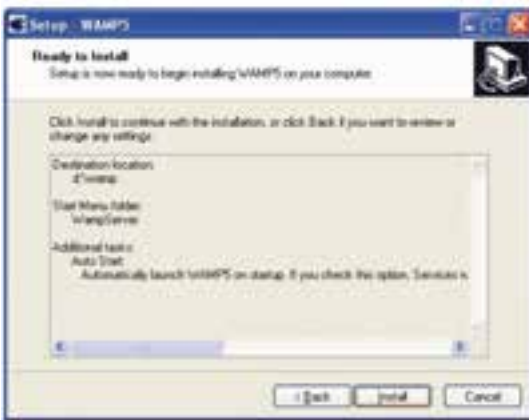
شکل ۵-۵



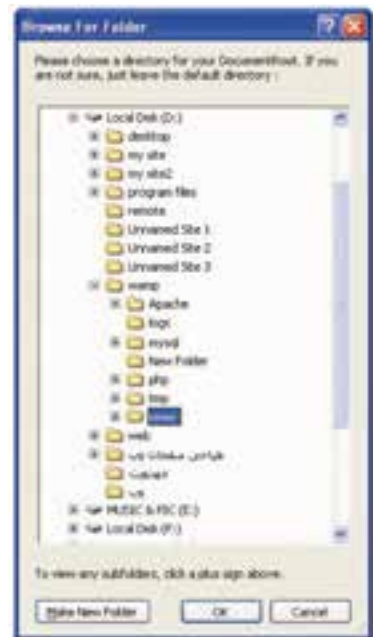
شکل ۵-۶

در پنجره شکل ۵-۶ یک کادر علامت وجود دارد که اگر آنرا انتخاب کنید، از این پس به محض روشن شدن سیستم و راهاندازی ویندوز، برنامه WAMP به طور خودکار راهاندازی و آماده استفاده میشود. در غیر این صورت قبل از این که بخواهید از آن استفاده کنید باید حتماً قبل از شروع به کار، WAMP را از مسیر برنامه‌های نصب شده روی سیستم از طریق منوی Start و زیر منوی Application، اجرا کنید.

در پنجره شکل ۵-۷ روی دکمه Install کلیک کنید. پس از چند ثانیه فایل‌های این برنامه روی رسانه دیسک سخت ذخیره می‌شوند و کادرمحاوره شکل ۵-۸ ظاهر می‌شود.



شکل ۵-۷

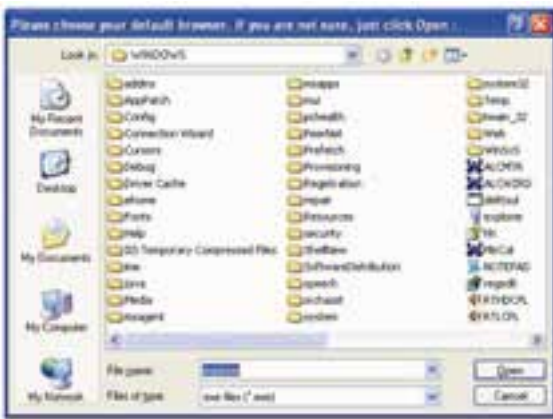


شکل ۵-۸

در این کادرمحاوره مسیر پیش فرض ریشه برای قرار دادن فایل‌های مربوط به سایت تعیین شده است، این مسیر به طور پیش فرض در مسیر نصب WAMP و در پوشه WWW انتخاب شده است، اگر مایل بودید می‌توانید این مسیر را عوض کرده و مسیر موردنظر خود را برای ذخیره فایلها و اطلاعات مربوط به سایت تعیین کنید.

**نکته** این مسیر را به خاطر بسپارید، زیرا از این پس حتماً باید فایل‌های مربوط به وب سایت را در این پوشه قرار دهید. در غیر این صورت نمی‌توانید از امکانات WAMP استفاده نمایید.

پس از انتخاب مسیر و کلیک روی دکمه OK کادرمحاوره شکل ۹-۵ ظاهر میشود. در شکل ۹-۵ باید مرورگر وب پیش فرض برای اجرای صفحات را تعیین کنید، این مرورگر به طور پیش فرض Internet Explorer است، اما در صورت دلخواه میتوانید مرورگر دیگری که آنرا قبلاً روی سیستم خود نصب نموده‌اید، از مسیر نصب انتخاب کنید و روی دکمه Open کلیک نمایید. در پنجره شکل ۱۰-۵ روی دکمه Finish کلیک کنید.



شکل ۹-۵



شکل ۱۰-۵



روی ناحیه تذکر از نوار وظیفه یک آیکن به شکل نیمدایره سفید ظاهر میشود. این نیمدایره ممکن است مواقعی به صورت زردرنگ یا قرمز رنگ ظاهر شود، این به آن معناست که سرویس‌ها به طور کامل در حال اجرا نیستند.



شکل ۱۱-۵



شکل ۱۲-۵

در اینصورت برای اجرای مجدد سرویسها روی آیکن نیمدایره کلیک کنید. در شکل ۱۳-۵ توجه کنید، باید کنار سرویس‌های Apache و MySQL علامت قرار گرفته باشد، وجود این علامت  به این معناست که سرویس در حال اجراست، در شکل ۱۳-۵ اگر علامتی به شکل  کنار هر یک از این سرویسها قرار گرفته بود به این معناست که آن سرویس متوقف شده است، در این صورت، ماوس را روی آن سرویس نگه دارید تا زیر منوی مربوطه باز شود.



شکل ۵-۱۴

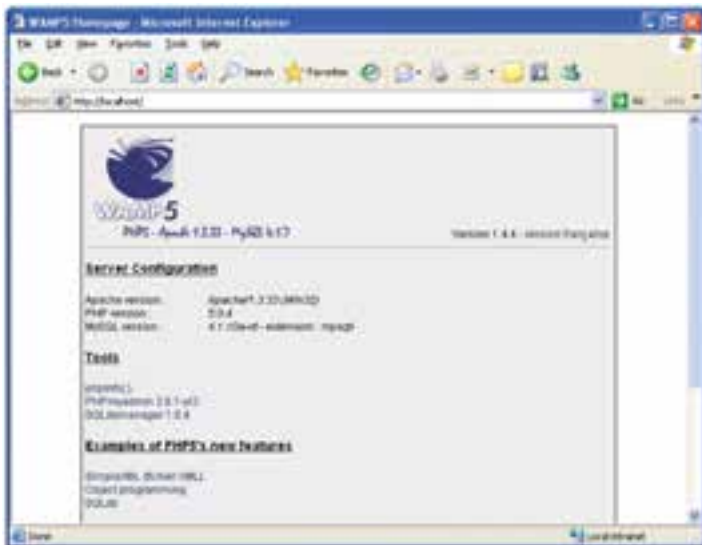
شکل ۵-۱۳

برای اجرای سرویس روی گزینه Start/Resume Service کلیک کنید (شکل ۵-۱۴).

برای اینکه اطمینان پیدا کنید سرویس‌دهنده به خوبی کار میکند در شکل ۵-۱۳ گزینه Localhost را انتخاب کنید.

اگر پنجره‌های مانند شکل ۵-۱۵ در مرورگر نشان داده شود، به معنای صحت عملکرد سرویس‌هاست. این سرویس‌ها مانند سایر سرویس‌های در حال اجرا بوده و می‌توانید آنها را در لیست سرویس‌های ویندوز مشاهده کنید.

برای مشاهده سرویس‌های ویندوز از مسیر Control Panel روی گزینه Administrative Tools دابل کلیک کنید و در آن برنامه Services را باز کنید.



شکل ۵-۱۵

در شکل ۵-۱۶ لیست سرویس‌های در حال اجرا روی ویندوز نشان داده شده است، همان طور که ملاحظه می‌کنید سرویس‌های wampapache و wampmysql نیز در حال اجرا هستند.



شکل ۱۶-۵

**نکته** پس از نصب برنامه WAMP دیگر نیازی به نصب Apache و MySQL نیست و می‌توانید بدون درگیر شدن با آن برنامه‌ها، کار با زبان PHP را شروع کنید.

## ۲-۵ نصب PHP در لینوکس

برای نصب و راه‌اندازی PHP در لینوکس ابتدا باید Apache را نصب کنید، پس از نصب Apache که در واحدهای کار قبل با آن آشنا شدید، می‌توانید کدهای مربوط به PHP را از سایت PHP به آدرس [www.PHP.net](http://www.PHP.net) دریافت کنید.

در سیستم عامل لینوکس برای فراهم شدن امکان اجرای کدهای PHP توصیه می‌شود ابتدا Apache را روی سیستم خود نصب نمایید. پس از اتمام مراحل نصب Apache، باید فایل‌های مربوط به نصب PHP در لینوکس را از اینترنت دریافت نمایید. پس از نصب PHP فایل `php.ini-dist` را به `php.ini` تغییرنام داده و آنرا در مسیر `/usr/Local/Lib/` کپی کنید.

در مرحله بعدی باید یکی از فایل‌های سرویس‌دهنده وب Apache را پیکربندی کنید. برای این منظور فایل `httpd.conf` را از مسیر `/usr/local/apache/` پیدا کنید، آنرا باز کنید و دو سطر زیر را به انتهای آن بیفزایید:

```
Load Module php5_module modules/libphp5.S0
```

```
Add Type application/X_httpd_php.php.phtml
```

سپس در همان فایل دنبال کلمه DirectoryIndex بگردید و آنرا به صورت زیر تغییر دهید:  
DirectoryIndex index.php index.htm

فایل را ذخیره کنید و سپس آنرا ببندید.

پس از طی مراحل فوق، سرویس Apache را از مسیری که نصب شده بود اجرا کنید.

### ۳-۵ فایل php.ini در ویندوز

PHP برای تنظیم عملیات مختلفی که روی سرویس دهنده انجام میدهد از دستورات عمل‌هایی که درون فایلی با نام php.ini قرار داده شده است استفاده میکند و برنامه‌نویس میتواند بسته به نیاز خود این فایل را تغییر دهد.

اگر از WAMP برای نصب و راه اندازی php استفاده کنید، فایل مذکور به طور خودکار در شاخه system32 ویندوز کپی میشود. اما در سایر روش‌ها باید توسط فردی که PHP را نصب می‌کند، به‌طور دستی از مسیر نصب PHP درون پوشه system32 کپی شود.

برای دسترسی به این فایل روی آیکن نیم‌دایره از برنامه WAMP کلیک کنید. روی گزینه config files و پس از آن در زیر منوی باز شده روی گزینه php.ini کلیک کنید.



شکل ۱۷-۵



شکل ۱۸-۵

**نکته** سعی کنید فقط در مواقع لزوم و مطابق دستورالعمل محتویات فایل

php.ini را دستکاری کنید.

## ۴-۵ نصب PHP بدون استفاده از سرویس دهنده‌ها

• CD نصب PHP5 را تهیه کنید. آنرا در درایو CD گذاشته و فایل مربوط به نصب را اجرا کنید. PHP را در یک مسیر مشخص نصب کنید و در مراحل نصب، روی دکمه Next کلیک کنید تا نصب خاتمه پیدا کند.

• فایل‌های php5ts.dll و php5isapi.dll را از مسیر نصب PHP پیدا کنید و پوشه system32 را از ویندوز کپی کنید.

• در مسیر نصب PHP، درون پوشه ext فایل php.ini-recommended را به php.ini تغییر نام دهید، آنرا با Notepad باز کنید.

از منوی Edit گزینه Find را انتخاب کنید، در کادر متنی Find what کلمه extinction\_dir را وارد کرده، با استفاده از دکمه Find Next محل آن را در متن پیدا کنید، سپس مقابل آن بین دو علامت " " مسیری که PHP را نصب کرده‌اید به همراه php/ext/ وارد کنید.

کلمات MySQL.default\_port، MySQL.default\_host و MySQL.default\_user را نیز به ترتیب پیدا کنید و مقابل آنها این مقادیر را وارد کنید:

MySQL.default\_port=3306

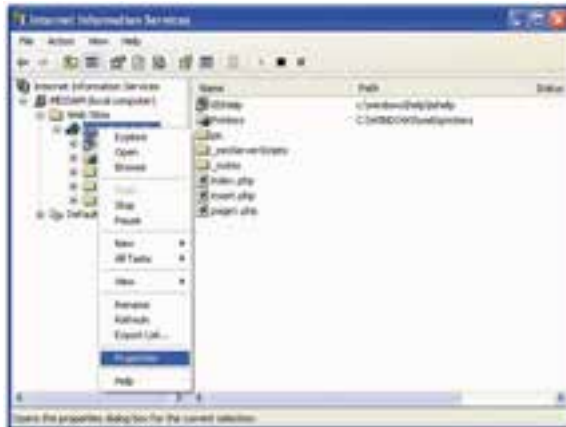
MySQL.default\_host=localhost

MySQL.default\_user=root

فایل php.ini را ذخیره کنید و سپس آنرا ببندید.

• این فایل تغییر یافته را در پوشه Windows کپی کنید.

• در پنجره Control Panel وارد Administrative Tools شوید و برنامه IIS را باز کنید. روی Default website کلیک راست کرده، گزینه Properties را انتخاب کنید (شکل ۱۹-۵).



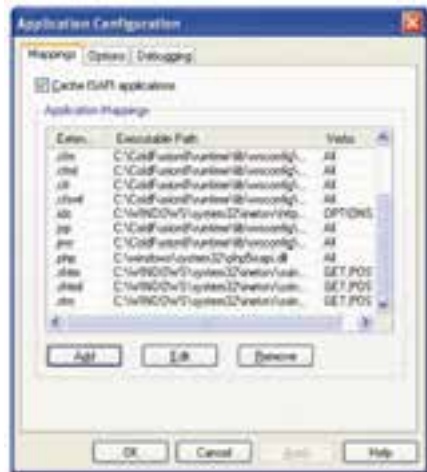
شکل ۱۹-۵



در زبانه Home Directory روی دکمه Configuration کلیک کنید.



شکل ۵-۲۰



شکل ۵-۲۱

در کادرمحاوره ظاهر شده روی دکمه Add کلیک کنید. در کادرمحاوره شکل ۵-۲۲ در کادر متنی Executable عبارت زیر را وارد کنید:

C:\windows\system32\php5isapi.dll

و در کادر متنی Filter Name عبارت php. را بنویسید.



شکل ۵-۲۲

روی دکمه OK کلیک کنید تا به کادرمحاوره شکل ۵-۲۱ بازگردید، مجدداً روی OK کلیک کنید. دوباره به کادرمحاوره شکل ۵-۲۰ برمیگردید، زبانه ISAPI Filters را انتخاب کنید (شکل ۵-۲۳).



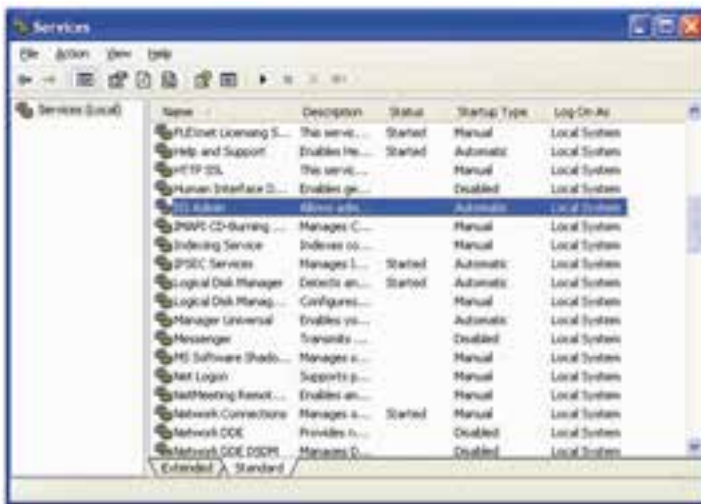
شکل ۵-۲۳

سپس روی دکمه Add کلیک کنید.

مسیری را که در قسمت قبل نوشته بودید در کادر متنی Executable وارد کنید و روی دکمه OK کلیک کنید. مجدداً به کادرمحاوره شکل ۲۳-۵ برمیگردید، دکمه OK را کلیک کنید و از برنامه IIS خارج شوید.

سرویس IIS را به صورت زیر ابتدا متوقف کرده و سپس مجدداً اجرا کنید:

• در Control Panel وارد Administrative Tools شوید و برنامه Services را باز کنید:



شکل ۲۴-۵

- سرویس IIS Admin را پیدا کنید، آنرا انتخاب نمایید و در نوار ابزار روی دکمه کلیک کنید. این عمل باعث میشود سرویس IIS متوقف شود.
- پس از چند ثانیه که عملیات متوقف سازی IIS انجام شد، روی دکمه کلیک کنید تا این سرویس مجدداً اجرا شود.

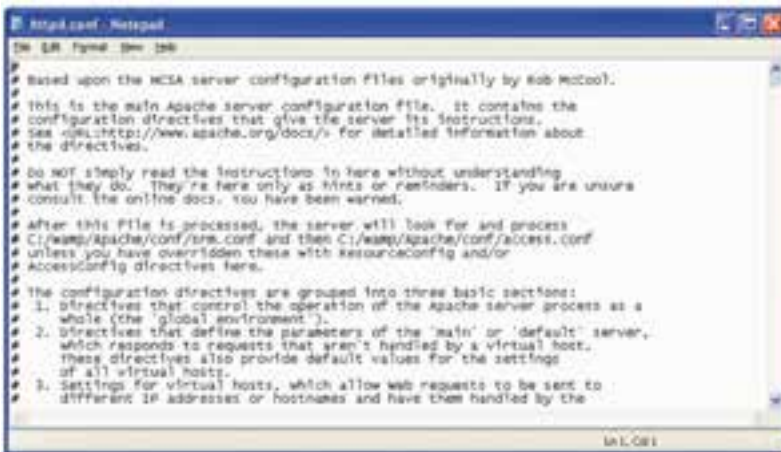
## ۱-۴-۵ ارتباط PHP با MySQL

برای ارتباط دادن PHP با MySQL فایل های Libmysql.dll و php\_mysql.dll را از مسیر

نصب PHP پیدا کنید و آنها را در پوشه system32 کپی کنید. سپس فایل Libmysql.dll را از مسیر نصب PHP پاک کنید.

## ۵-۵ پیکربندی Apache در ویندوز

برای پیکربندی سرویس دهنده وب Apache فایلی با نام httpd.conf در نظر گرفته شده است. این فایل دستورات مختلفی برای پیکربندی و مدیریت رفتارهای مختلف سرویس دهنده دارد. برای دسترسی به این فایل روی آیکن WAMP کلیک کنید و ماوس را روی گزینه config files نگه دارید و در زیرمنوی باز شده گزینه httpd.conf را انتخاب کنید. (شکل ۲۵-۵).



شکل ۲۵-۵

## ۵-۶ پیکربندی Apache در لینوکس

فایل پیکربندی Apache در لینوکس در مسیر `/etc/httpd/httpd.conf` ذخیره شده است، این فایل در محیط لینوکس نیز میتواند توسط یک ویرایشگر متنی ساده باز شود. سعی کنید تغییرات این فایل را نیز فقط در مواقع لزوم و مطابق دستورالعمل تغییر دهید.

## ۵-۷ بررسی صحت عملکرد PHP

پس از آن که PHP را نصب کردید، لازم است از صحت عملکرد و درست و کامل بودن عملیات نصب اطمینان حاصل کنید. به این منظور روی آیکن نیم دایره WAMP کلیک کرده، از پنل WAMP5 گزینه phpMyAdmin را انتخاب کنید.

در صفحه مرورگر برنامه phpMyAdmin باز میشود. یکی از ویژگی‌های سرویس‌دهنده WAMP قابلیت پشتیبانی از زبان فارسی است، اگر می‌خواهید این صفحه برای شما به زبان فارسی نشان داده شود از لیست بازشوی Language گزینه Persian (fa-utf-8) را انتخاب کنید.



شکل ۲۶-۵



شکل ۲۷-۵



شکل ۲۸-۵

در واحدهای کار آینده به نحوه کار با این برنامه خواهیم پرداخت.

## واژه نامه

Client Side  
Configuration  
Server Side

سمت سرویس گیرنده  
پیکربندی  
سمت سرویس دهنده

## خلاصه مطالب

- مهیا کردن سیستم برای شروع اسکریپت نویسی پویا با PHP تا حدودی کار در دسترس است. به این دلیل که علاوه بر نصب برنامه های MySQL، PHP و Apache باید در هر کدام تنظیمات ویژه ای انجام شود. اما نرم افزارهایی مانند WAMP، EasyPHP و ... تمام تنظیمات لازم را به طور خودکار انجام میدهند.
- زبان های برنامه نویسی و اسکریپت نویسی به طور کلی به دو دسته Server Side و Client Side تقسیم میشوند.
- هنگام نصب برنامه WAMP یک مسیر به عنوان مسیر سرویس دهنده محلی در نظر گرفته می شود و هر صفحه وبی که بخواهد از سرویس دهنده به طور مجازی استفاده کند باید در این مسیر ذخیره شود.
- اگر از برنامه های سرویس دهنده از قبیل WAMP استفاده نکنید، لازم است پیکربندی فایل php.ini را به طور دستی انجام دهید، تنظیم این فایل در ویندوز و لینوکس دستورالعمل مشخصی دارد.
- برای ایجاد ارتباط در PHP لازم است پیکربندی فایل httpd.conf نیز به طور دستی صورت گیرد.

## آزمون نظری

- ۱- کدام یک از گزینه های زیر در مورد انواع زبان های اسکریپت نویسی وب درست است؟  
الف- زبان های Server Side به زبان هایی گفته میشود که اطلاعات را از سرویس دهنده دریافت کرده، به سرویس گیرنده ارائه میدهند.  
ب- زبان های Client Side به زبان هایی گفته میشود که اطلاعات را از سرویس گیرنده دریافت کرده و به سرویس دهنده ارائه میدهند.  
ج- زبان های Server Side زبان هایی هستند که روی سیستم سرویس دهنده و زبان های Client Side زبان هایی هستند که روی سیستم سرویس گیرنده اجرا میشوند.  
د- تمام گزینه ها صحیح است.
- ۲- کدام یک از گزینه های زیر از برنامه های ارائه دهنده سرویس PHP و نرم افزارهای

جانبی نیستند؟

- |             |         |            |            |
|-------------|---------|------------|------------|
| الف- Apache | ب- WAMP | ج- EasyPHP | د- SQLware |
|-------------|---------|------------|------------|
- ۳- پیکربندی PHP توسط کدام یک از فایل های زیر صورت میگیرد؟  
الف- WAMP.ini  
ب- httpd.conf  
ج- php.ini  
د- php.exe
  - ۴- پیکربندی Apache توسط کدام یک از فایل های زیر صورت میگیرد؟  
الف- WAMP.ini  
ب- httpd.conf  
ج- php.ini  
د- php.exe
  - ۵- برای پیکربندی PHP فایل php5ts.dll باید در کجا کپی شود؟  
الف- در شاخه Windows  
ب- در شاخه system32  
ج- در شاخه root  
د- در شاخه phpMyAdmin
  - ۶- فایل پیکربندی Apache لینوکس در چه مسیری قرار دارد؟  
الف- /etc/Apache/  
ب- Apache/httpd-  
ج- etc/httpd  
د- lib/httpd -



## کار با متغیرها، عملگرها و ساختارهای کنترلی

### واحد کارششم

پس از مطالعه این واحد کار از فراگیر انتظار می‌رود که:

- ۱ - تگ‌های آغاز و پایان PHP را بشناسد.
- ۲ - اصول نحوه نوشتن دستورات در PHP را بشناسد.
- ۳ - بتواند کدهای HTML و PHP را در کنار یکدیگر استفاده کند.
- ۴ - با نحوه درج توضیحات درون کدهای PHP آشنا باشد.
- ۵ - انواع متغیرهای PHP را بشناسد.
- ۶ - بتواند متغیرهای از پیش تعریف شده را در برنامه به کار ببرد.
- ۷ - با ثابت‌ها و عملکرد آن‌ها آشنا باشد.
- ۸ - با اصول نامگذاری متغیرها آشنا باشد.
- ۹ - انواع عملگرهای PHP را بشناسد.
- ۱۰ - اصول کار با انواع ساختارهای کنترلی را بداند.

| زمان (ساعت) |      |
|-------------|------|
| عملی        | نظری |
| ۱۲          | ۶    |



## مقدمه

برای ایجاد فایل‌های PHP نیازی نیست حتماً برنامه خاصی روی سیستم نصب شود و کدهای PHP را میتوان در یک ویرایشگر ساده متنی مانند Notepad نوشت. نرم‌افزار Dreamweaver نیز می‌تواند فایل‌های PHP را پشتیبانی کرده و محیط مناسبی برای کدنویسی فراهم کند.

## ۱-۶ شروع برنامه‌نویسی در PHP

برای نوشتن دستورات PHP دو روش اصلی وجود دارد که در ادامه به توضیح آن‌ها خواهیم پرداخت. شما به صورت دلخواه می‌توانید هر کدام از این ساختارها را در برنامه، مورد استفاده قرار دهید.

• روش اول: دستورات و کدهای PHP درون علائم `<? ... ?>` قرار می‌گیرند.

`<?>`

دستورات و کدهای PHP

`?>`

• روش دوم: دستورات و کدهای PHP درون علائم `?php ... ?>` قرار می‌گیرند.

`<?php`

دستورات و کدهای PHP

`?>`

---

توصیه: به منظور تسهیل در کدنویسی و حداکثر سازگاری کدها در مرورگرهای مختلف، توصیه می‌شود برای نوشتن کدهای PHP از روش دوم استفاده نمایید.

---

برای ایجاد یک فایل اجرایی بسیار ساده در PHP متن `hello` را در مرورگر نمایش می‌دهیم. در انتهای هر خط از دستورات نیز باید از علامت «» استفاده شود. برای نمایش اطلاعات در مرورگر، دستورات مختلفی به کار می‌رود، دستور `Print` یکی از پرکاربردترین آن‌هاست. کد زیر میتواند متن `hello` را در مرورگر نمایش دهد.

```
<? PHP
```

```
Print ("hello!");
```

```
?>
```

این کد را در Notepad بنویسید، فایل را با پسوند PHP در مسیر نصب WAMP و در شاخه www ذخیره کنید.

سپس مرورگر را باز کرده و در نوار آدرس این متن را تایپ کنید: (به جای نام فایل، عنوانی را که برای ذخیره فایل وارد کردید درج نمایید).

http://localhost/فایل.php نام فایل



شکل ۶-۱

**نکته** اگر این فایل را با نام index.php یا default.php ذخیره کنید، لازم نیست در نوار آدرس نام فایل را ذکر نمایید، فقط کافی است عبارت http://localhost را تایپ کرده و کلید Enter را فشار دهید.

در تمام سایت‌ها، صفحه اصلی (home page) سایت را با نام index یا default ذخیره میکنند، هنگام درج آدرس یک سایت، همیشه صفحه‌ای که در مسیر اصلی سایت با نام index یا default باشد، به طور پیش فرض برای اجرا انتخاب میشود، سایر صفحات از طریق همین صفحه با یکدیگر مرتبط میشوند.

دستور اصلی برای چاپ اطلاعات در خروجی هر زبان PHP، دستور echo است. برنامه قبل را با این دستور می‌توانید به این صورت بنویسید:

```
<? Php
```

```
echo ("hello!");
```

```
?>
```

اگر بخواهید چند عبارت را توسط یک دستور echo چاپ کنید، باید هر عبارت را با  
"." از عبارت بعدی جدا کنید. مانند (echo ("hello!". " welcome"  
حال اگر بخواهید هر کدام از عبارات در یک سطر جدا چاپ شود باید بین هر عبارت  
از "<br>" استفاده کنید.

```
echo ("hello!". "<br>". "welcome")
```

مثال:

## ۱-۱-۶ استفاده از کدهای PHP و HTML در کنار یکدیگر

یکی از قابلیت‌های زبان PHP این است که میتواند درون کدهای HTML به کار برده شود.

```
<html>
```

```
<head>
```

```
<title>first program</title>
```

```
</head>
```

```
<body>
```

```
<?php
```

```
echo("hello!");
```

```
?>
```

```
</body>
```

```
</html>
```

همانطور که ملاحظه می‌کنید، برای درج کد PHP درون HTML فقط کافی است آن را  
درون تگ <body> ... </body> بنویسید، اما پس از نوشتن کدها، باید فایل با پسوند .php  
ذخیره شود و در صورت ذخیره فایل با پسوند htm یا html، کدهای PHP اجرا نخواهند شد.

## ۱-۱-۲ درج توضیحات درون کدهای PHP

توضیحات در زبان‌های برنامه‌نویسی به منظور راهنمایی برای برنامه‌نویس یا فردی

است که میخواهد از کدها استفاده کند، در PHP نیز مانند سایر زبانهای برنامه‌نویسی دستورات عمل‌هایی برای درج توضیحات وجود دارد. در PHP برای توضیحات یک خطی از علائم // و # و برای درج توضیحات چند خطی از علامت /\* در ابتدا و از علامت \*/ در انتهای توضیحات استفاده میشود.

```
<html>
<head>
<title>first program</title>
</head>
<body>

<?php
/* Line number1: These 3 lines will not either!
Line number2
Line number3: End of comment */

echo("hello!");

// This line will not be parsed in PHP

echo("welcome!");

# This line will not be parsed in PHP like the line above

?>

</body>
</html>
```

## ۲-۶ متغیرها

یکی از مهمترین ابزارهایی که برای تبدیل صفحات ساده و ثابت به صفحات پویا لازم است، این است که سرویس دهنده بتواند اطلاعات را از کاربر سرویس گیرنده دریافت کند و با توجه به اطلاعات پاسخ مناسبی به کاربر بدهد. متغیرها ابزاری هستند که با استفاده از آنها میتوان اطلاعات را از کاربر دریافت کرد. در واقع متغیرها به منزله ظرفهایی هستند که اطلاعات کاربر را به طور موقت ذخیره کرده و در مواقع لزوم آنرا برای سرویس دهنده ارسال میکنند.

برای تعریف متغیر در PHP باید از حرف \$ استفاده کرد.

\$a;

مثال:

\$b=10;

نوع متغیر در PHP اهمیتی ندارد و لازم نیست هنگام تعریف متغیر، نوع آن ذکر شود. متغیرهای PHP نسبت به حروف کوچک و بزرگ حساس هستند. به عنوان مثال متغیرهای  $a$  و  $A$  با یکدیگر متفاوت بوده و دو متغیر جدا محسوب می شوند. در نامگذاری متغیرها علاوه بر حروف کوچک و بزرگ می توان از علائمی مانند "\_" یا خط فاصله پایین (Underline) استفاده کرد. استفاده از اعداد نیز در نامگذاری متغیرهای PHP مجاز است در صورتی که اولین حرف از متغیر نباشد.

مثال: متغیرهای با نامهای  $\$abc$  و  $\$ab_1$  مجاز هستند.

مثال: متغیرهای با نامهای  $\$1a$  و  $\$1ab$  غیرمجاز هستند.

**نکته** نامگذاری متغیرها به طوری که "\_" (Underline) در ابتدای آن‌ها باشد، مجاز است اما توصیه می شود از نامگذاری متغیرها به این شیوه استفاده نکنید و حتی الامکان از "\_" در ابتدای نام متغیرها استفاده نکنید.

در متغیرها نشان دادن نوع توسط مقداردهی تعیین می شود. دستورالعمل تعیین نوع برای هر متغیر به صورت زیر است:

- اعداد صحیح به طور ساده نوشته می شوند.

\$a=15;

مثال:

• اعداد اعشاری توسط علامت "." تعیین می شوند.

مثال:


\$b=7.0;

\$C=1.295;

• اعداد کسری توسط علامت "/" تعیین می شوند.

\$f=1/2;

مثال:

**نکته**  اعداد کسری به صورت اعشاری ذخیره می شوند، در مثال فوق متغیر f با 0.5 مقداردهی می شود.

رشته ها درون علامت "" یا '' درج می شوند.

\$Str = "Hello" ;

مثال:

\$H= "2914" ;

\$M = 'Welcome' ;

اگر یک عدد بین علامت "" قرار گیرد، مقدار رشته ای محسوب می شود.

مثال: می خواهیم ابتدا دو متغیر را با اعداد ۳۵ و ۷۹ مقداردهی کنیم، سپس در خروجی

حاصل جمع این دو عدد را چاپ کنیم:

```
<html>
```

```
<head>
```

```
<title>first program</title>
```

```
</head>
```

```
<body>
```

```
<?php
```

```
$num1 = 35;
```

```
$num2 = 79;
```

```
$sum = $num1 + $num2;  
echo("sum is: ");  
echo("$sum");  
  
?>  
  
</body>  
</html>
```



شکل ۲-۶

## ۱-۲-۶ آرایه‌ها

همان‌طور که می‌دانید آرایه‌ها از لحاظ ساختاری پیچیده‌تر از انواع داده‌های دیگر هستند، برخلاف انواع داده‌های رشته‌ای و عددی که فقط می‌توانند یک مقدار را بپذیرند، آرایه‌ها می‌توانند شامل لیستی از مقادیر مختلف باشند.

در زبان PHP آرایه‌ها را می‌توان به دو دسته اندیس‌دار<sup>۱</sup> و انجمنی<sup>۲</sup> تقسیم کرد، روش دسترسی به عناصر در این دو روش متفاوت است.

در آرایه‌های اندیس‌دار، دسترسی به هر یک از عناصر توسط اندیس آن صورت می‌گیرد.

اولین اندیس آرایه نیز مانند زبان جاوا اسکریپت «۰» است.

ساختار کلی آرایه اندیس‌دار به صورت زیر است:

مثال: "مقدار عنصر آرایه" = [ ] نام متغیر آرایه‌ای;

```
$name[]="Maryam";
```

```
$name[]="zahra";
```

```
$name[]="Meisam";
```

روش دوم برای معرفی آرایه‌ها استفاده از شناسه array است.

... و "مقدار عنصر دوم" و "مقدار عنصر اول" array = نام متغیر آرایه‌ای

مثال:

```
$name = array ("Maryam" و "Zahra" و "Meisam");
```

حال می‌توانیم با استفاده از دستورات برنامه به هر کدام از عناصر آرایه دسترسی داشته باشیم. اندیس عناصر آرایه name به این صورت برای PHP شناسایی می‌شوند:

```
$name [0] = "Maryam"
```

```
$name [1] = "Zahra"
```

```
$name [2] = "Meisam"
```

```
<html>
```

```
<head>
```

```
<title>first program</title>
```

```
</head>
```

```
<body>
```

```
<?php
```



شکل ۳-۶

```
$name= array("maryam","zahra","meisam");
```

```
echo(" $name[0] ." $name[1] ." $name[2] ");
```

```
echo("<br>."<br>." $name[0] ."<br>." $name[1] ."<br>." $name[2] ");
```

```
?>
```

```
</body>
```

```
</html>
```

### تمرین:

یک آرایه عددی با ۴ عنصر تعریف کنید، عناصر را با اعداد ۹ تا ۵ مقداردهی کرده، سپس با استفاده از اندیس آرایه‌ها، آن‌ها را با هم جمع کرده، نتیجه حاصل جمع را در خروجی نمایش دهید.



## ۲-۲-۶ متغیرهای از پیش تعریف شده

متغیرهای از پیش تعریف شده متغیرهایی هستند که در یکی از برنامه‌های کاربردی سرویس‌دهنده وب استفاده شده‌اند و برنامه‌نویس مجاز نیست از نام آنها برای متغیرهای خود استفاده کند.

مهم‌ترین متغیرهای از پیش تعریف شده عبارتند از:

**OSTYPE**: نوع سیستم عاملی را که روی سرویس‌دهنده در حال اجراست در بر می‌گیرد.

**HOSTNAME**: نام سرویس‌دهنده را در بر می‌گیرد.

**PHP-SELF**: نام صفحه جاری را در بر می‌گیرد.

## ۲-۲-۳ ثابت‌ها

همان‌گونه که از نام ثابت‌ها پیداست، مقادیری هستند که در طول اجرای برنامه ثابت بوده و مقدار آنها تغییر نمی‌کند.

ثابت‌ها به‌طور سراسری و در همه جای برنامه قابل استفاده و در دسترس هستند.

برای تعریف ثابت‌ها از تابع `define` استفاده می‌شود و نیاز به درج علامت `$` ندارند.

هنگام استفاده از ثابت‌ها در طول برنامه نیز به علامت `$` نیاز نداشته و تنها نوشتن نام متغیر کافی است.

ساختار کلی تعریف ثابت‌ها به‌صورت زیر است:

مثال:

```
define ("مقدار ثابت","نام ثابت");
```

```
define("PI","۳.۱۴");
```

مثال زیر عدد  $\pi$  را در متغیری به نام `pi` به‌طور ثابت قرار می‌دهد، شعاع دایره را

متغیری صحیح و با عدد ۵ مقداردهی می‌کند، سپس محیط دایره را محاسبه کرده و در

خروجی نشان می‌دهد.

```
<?php
```

```
define ("pi","3.14");
```

```
$r=5;
```

```
$e=pi*2*$r";
```

```
echo ("environment : ".$e")
```

```
?>
```



شکل ۴-۶

## ۳-۶ عملگرها

### ۱-۳-۶ عملگرهای محاسباتی

عملگرها در تمام زبان‌ها برای اجرای محاسبات روی داده‌ها به کار می‌روند.

جدول ۱-۶

مثال	عملگر	
$a+b$	جمع	+
$a-b$	تفریق	-
$a*b$	ضرب	*
$a/b$	تقسیم	/
$a \% b$	باقیمانده تقسیم	%
$a++$ یا $++a$	افزایش	++
$--a$ یا $a--$	کاهش	--

## ۲-۳-۶ عملگرهای مقایسه‌ای

جدول ۶-۲

مثال	عملگر	
$a == b$	تساوی	<code>==</code>
$a === b$	تساوی	<code>===</code>
$a != b$	نامساوی	<code>!=</code>
$a <> b$	نامساوی	<code>&lt;&gt;</code>
$a < b$	کوچک‌تر از	<code>&lt;</code>
$a > b$	بزرگ‌تر از	<code>&gt;</code>
$a <= b$	کوچک‌تر یا مساوی	<code>&lt;=</code>
$a >= b$	بزرگ‌تر یا مساوی	<code>&gt;=</code>

اگر بخواهید مشخص کنید که آیا دو متغیر  $a$  و  $b$  با هم مساوی هستند یا خیر، می‌توانید از عملگر `==` استفاده کنید.

مثال ۱:

- $a = 5$ ;
- $b = 5$ ;
- $c = 10$ ;
- $a == b$ ;
- $b == c$ ;

در دستور سطر ۴ چون  $a$  با  $b$  برابر هستند مقدار `true` برگردانده می‌شود. اما در سطر ۵ به دلیل این که  $c$  و  $b$  با هم مساوی نیستند مقدار `false` برگردانده می‌شود.

مثال ۲:

- $a = 3$ ;  
 $b = "3"$ ;  
 $a == b$ ;

در دستور سوم ارزش `true` برگردانده می‌شود زیرا ابتدا متغیر  $b$  نوع رشته‌ای به نوع عددی تبدیل می‌شود و سپس مقایسه صورت می‌گیرد.

اگر بخواهیم تبدیل نوع صورت نگیرد و مقایسه با انواع یکسان صورت گیرد، باید از عملگر `===` استفاده کنیم:

`$a = 7;`

`$b = "7";`

`$a === $b;`

دستور سوم مقدار `false` را برمی گرداند، زیرا `$a` و `$b` همنوع نیستند.

### ۳-۳-۶ عملگرهای منطقی

عملگرهای منطقی روی عبارات منطقی عمل می کنند. به عنوان مثال اگر بخواهیم تشخیص دهیم مقدار متغیر `$a` بین ۱ و ۱۰۰ قرار دارد باید از عملگرهای منطقی استفاده کنیم. این عملگرها را در جدول زیر مشاهده می نمایید:

جدول ۳-۶

عملگر	نام	مثال	نتیجه
	نقیض (NOT)	<code>\$b</code>	اگر <code>\$b</code> نادرست باشد، نتیجه درست است و برعکس.
&&	و (AND)	<code>\$a &amp;&amp; \$b</code>	اگر <code>\$a</code> و <code>\$b</code> هر دو درست باشند نتیجه درست است در غیر این صورت نادرست است.
	یا (  )	<code>\$a    \$b</code>	اگر حداقل یکی از متغیرهای <code>\$a</code> یا <code>\$b</code> درست باشند، نتیجه درست است و در غیر این صورت نادرست است.

### ۴-۳-۶ عملگرهای ترکیبی

تعدادی از عملگرها در PHP وجود دارند که ترکیبی از دو عملگر دیگر هستند و به نام عملگرهای ترکیبی خوانده می شوند. این عملگرها در جدول زیر توضیح داده شده اند:

جدول ۴-۶

عملگر	نام	مثال	عبارت معادل
<code>+ =</code>	انتساب جمع	<code>\$a += \$b</code>	<code>\$a = \$a + \$b</code>
<code>- =</code>	انتساب تفریق	<code>\$a -= \$b</code>	<code>\$a = \$a - \$b</code>
<code>* =</code>	انتساب ضرب	<code>\$a *= \$b</code>	<code>\$a = \$a * \$b</code>
<code>/ =</code>	انتساب تقسیم	<code>\$a /= \$b</code>	<code>\$a = \$a / \$b</code>
<code>% =</code>	انتساب باقیمانده تقسیم	<code>\$a %= \$b</code>	<code>\$a = \$a % \$b</code>

## ۵-۳-۶ عملگرهای رشته‌ای

یکی از پرستفاده‌ترین عملیاتی که روی رشته‌ها انجام می‌گیرد، عمل الحاق کردن رشته‌هاست. به عمل چسباندن دو رشته به یکدیگر عمل الحاق رشته‌ها گفته می‌شود. به عنوان مثال دو رشته زیر را در نظر بگیرید.

```
$a="web";
```

```
$b="design";
```

برای الحاق رشته‌ها در PHP از عملگر . استفاده می‌شود:

```
$c=$a . $b;
```

به این ترتیب حاصل رشته `$c`، "webdesign" خواهد بود.

در PHP دو نوع رشته وجود دارد:

رشته‌هایی که بین دو علامت "" قرار می‌گیرند. مثل: "webdesign"

رشته‌هایی که بین دو علامت ' ' قرار می‌گیرند. مثل: 'webdesign'

اگر رشته‌ای در نقل قول دوتایی قرار گیرد، PHP سعی می‌کند آن را مثل یک عبارت ارزیابی کند، ولی اگر رشته در نقل قول یکانی قرار گیرد، ارزیابی نمی‌شود. برای مثال به دستورات زیر نگاه کنید:

```
1.$a=10;
```

```
2.echo "value of $a";
```

```
3.echo 'value of $a';
```

دستور سطر اول مقدار ۱۰ را در متغیر \$a قرار می‌دهد. خروجی دستور سطر دوم به صورت زیر است:

```
Value of 10
```

در حالی که دستور سطر سوم، به صورت زیر است:

```
Value of $a
```

## ۴-۶ کار با ساختارهای کنترلی

در مبحث جاوا اسکریپت با ساختارهای کنترلی آشنا شدید، در این قسمت ضمن بیان نکات مختصری جهت یادآوری مطالب، به بررسی ساختار هر کدام از دستورات در زبان

PHP می‌پردازیم:

### ۱-۴-۶ دستور if

یک عبارت را بررسی می‌کند، در صورت داشتن پاسخ true وارد بلوک کد مربوط به if می‌شود، در غیر این صورت از بلوک if می‌گذرد. ساختار دستور if به صورت زیر است:

if(عبارت شرطی)

{

در پرانتز مقابل دستور if باید یک عبارت شرطی گذاشته شود. دستورات بدنه

}

مثال:

```
<html>
<head>
<title>Listing 5.1</title>
</head>
<body>
<?php
$a = "hello";
if ( $a == "hello" )
{
echo "hello... welcome";
}

?>
</body>
</html>
```

اگر بخواهیم دستور if را طوری بنویسیم که در صورت برقرار نبودن شرط اول، شرط دوم، سوم و ... بررسی شود و در صورت صحیح بودن هر کدام کدهای مخصوص آن

اجرا شود می‌توانیم از دستور `elseif` و `else` استفاده کنیم.  
ساختار کلی این دستورات عبارتند از:

```
if(شرط اول)
{
    دستورات بدنه
}
elseif(شرط دوم)
{
    دستورات بدنه
}
elseif(شرط سوم)
{
    دستورات بدنه
    .....
}
else
{
    دستورات بدنه
}
```

در ساختار فوق ابتدا شرط مقابل `if` چک می‌شود، اگر درست بود، دستورات بدنه آن اجرا می‌شود، اما اگر نادرست بود شرط مقابل `elseif` اول اجرا می‌شود، در صورت صحیح بودن، کدهای آن اجرا شده، اما اگر صحیح نباشد، دستورات `elseif` بعدی را بررسی می‌کند و این عمل به تعداد دستورات `elseif` ادامه پیدا می‌کند و در نهایت اگر هیچ‌کدام از شرط‌های `if` و `elseif` صحیح نبودند، دستورات مربوط به بدنه `else` اجرا می‌گردد.

**مثال:**

```
<html>
<head>
<title>Listing 5.3</title>
</head>
```

```

<body>
<?php
$mood = "sad";
if ( $mood == "happy" )
{
echo "Hooray, I'm in a good mood";
}
elseif ( $mood == "sad" )
{
echo " I'm in a bad mood!";
}
else
{
echo "Neither happy nor sad but $mood";
}
?>
</body>
</html>
    
```

پس از اجرای کد مورد نظر مشاهده می‌کنید که به دلیل درست بودن شرط مقابل `elseif`، دستورات مربوط به این بلوک اجرا می‌شود.



شکل ۵-۶



## تمرین:

دستورات فوق را طوری تغییر دهید که بلوک مربوط به else اجرا شود.

### ۲-۴-۶ دستور Switch

ساختار کلی این دستور مانند جاوا اسکریپت است:

Switch (نام متغیر)

```
{
```

Case مقدار اول:

دستورات بدنه

```
break;
```

Case مقدار دوم:

دستورات بدنه

```
break;
```

```
⋮
```

```
default
```

دستورات بدنه

```
}
```

در ساختار Switch مقدار متغیر مقابل Switch با مقادیر مقابل Caseها مقایسه می‌شود، اگر با مقدار مقابل Case اول برابر باشد، دستورات بدنه آن اجرا می‌شود و سپس از بلوک Switch خارج می‌شود، اگر برابر با مقدار مقابل Case دوم باشد، دستورات مربوط به آن اجرا شده و از بلوک Switch خارج می‌شود.

به همین ترتیب چک کردن مقادیر ادامه پیدا می‌کند و اگر هیچ‌کدام از مقادیر برابر با متغیر مقابل Switch نبودند، دستور بدنه default اجرا می‌شود.

مثال قبل مربوط به شرط if ... elseif را با استفاده از دستور Switch بازنویسی می‌کنیم:

```
<html>
```

```
<head>
```

```
<title>Listing 5.4</title>
```

```

</head>
<body>
<?php
$mood = "sad";
switch ( $mood )
{
case "happy":
echo "Hooray, I'm in a good mood";
break;
case "sad":
echo "I'm in a bad mood!";
break;
default:
echo "Neither happy nor sad but $mood";
}
?>
</body>
</html>

```

پس از اجرای کد فوق خروجی این برنامه نیز مشابه شکل ۵-۵ خواهد بود.

## ۵-۶ حلقه‌ها

حلقه‌ها یکی دیگر از امکانات زبان‌های برنامه‌نویسی هستند که اجازه می‌دهند دستور یا دستوراتی چند بار اجرا شوند.

### ۱-۵-۶ دستور for

ساختار کلی دستور for به صورت زیر است:

for (میزان افزایش یا کاهش متغیر ; عبارت شرطی ; مقدار اولیه حلقه)

```

{
بدنه حلقه
}

```

ساختار حلقه‌ها در PHP نیز شباهت زیادی به جاوا اسکریپت دارد. مقدار اولیه حلقه، مقداری است که برای متغیر شمارنده تعداد دفعات اجرا تعیین می‌شود.

عبارت دوم، عبارت کنترلی است که آخرین بار اجرای حلقه را معین می‌کند. عبارت سوم مقدار اضافه یا کم شدن متغیر اولیه در هر بار اجرای حلقه را تعیین می‌کند. مثال: می‌خواهیم اعداد ۱ تا ۱۰ را در خروجی چاپ کنیم.

```
<html>
<head>
<title>for</title>
</head>
<body>
<?php
for ( $i=1; $i<=10; $i++ )
{
echo"$i". "<br>";
}
?>
</body>
</html>
```



شکل ۶-۶

### تمرین:



با استفاده از حلقه for برنامه‌ای بنویسید که اعداد ۱۰۰ تا ۵۰ را به صورت نزولی در خروجی چاپ کند.

در ساختار حلقه for مانند جاوا اسکریپت می‌توان مقدار اولیه حلقه را قبل از مقداردهی کرد و از نوشتن آن مقابل for صرف نظر کرد، هم‌چنین از میزان افزایش یا کاهش متغیر که معمولاً در پرانتز مقابل دستور for انجام می‌شود می‌توان صرف نظر کرده و عمل کاهش یا افزایش متغیر را درون حلقه for انجام داد.

### مثال:

```
<html>
<head>
<title>for</title>
</head>
<body>
<?php
$i=1;
for ( ; $i<=10; )
{
echo "$i".<br>";
$i++;
}
?>
</body>
</html>
```

### ۲-۵-۶ حلقه while

ساختار کلی این حلقه به صورت زیر است:

while (شرط خاتمه)

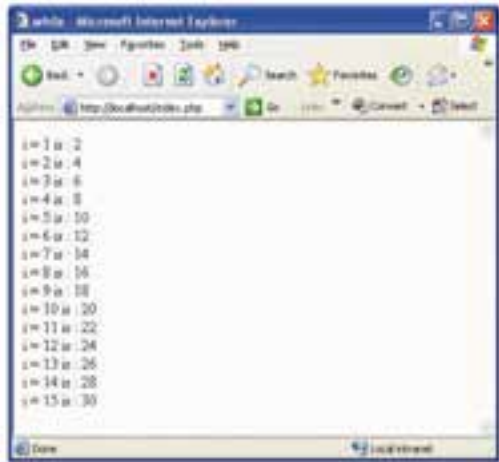
```
{
دستورات بدنه
}
```

مثال: می خواهیم توسط دستور while مضارب ۲ مربوط به اعداد کوچک تر از ۱۵ را

در خروجی نمایش دهیم.

```
<html>
<head>
<title>while</title>
</head>
<body>
```

```
<?php
$i = 1;
while ( $i <= 15 )
{
echo "i = ".$i is : " . ($i*2) . "<br>";
$i=$i+1;
}
?>
</body>
</html>
```



شکل ۶-۷

### ۳-۵-۶ حلقه do ... while

ساختار این حلقه نیز کاملاً شبیه به do ... while در جاوا اسکریپت است. این حلقه در تمام زبان‌ها حداقل یک بار اجرا می‌شود، چون شرط بررسی ادامه در انتهای حلقه نوشته شده است.

```
do {
    دستورات بدنه
}
while (عبارت شرطی);
```

مثال:

```
<html>
<head>
<title>do...while</title>
</head>
<body>
<?php
$n1 = 8;
do
```

```

{
print "number: " . $n1 . "<br>";
$n1++;
}
while ( $n1 > 10 && $n1 < 15 );
?>
</body>
</html>

```



شکل ۸-۶

در کد قبلی متغیر اولیه  $n1$  با عدد ۸ مقداردهی اولیه شده است و در عبارت شرطی مقابل `while`، بررسی شده است که تا زمانی که متغیر  $n1$  بزرگ‌تر از ۱۰ و کوچک‌تر از ۱۵ است حلقه ادامه یابد، اما در شکل ۸-۶ ملاحظه می‌کنید با این که این شرط هرگز صحیح نیست، یک‌بار اجرا شده است.

## ۶-۶ دستورات `break` و `continue`

عملکرد این دستورات مانند جاوا اسکریپت است. دستور `break` برای خروجی از بلوک جاری و ادامه برنامه استفاده می‌شود. دستور `continue` برای خروجی از حلقه با پارامتر فعلی و ادامه بلوک با مقدار پارامتر بعدی انجام می‌شود.

## ۶-۷ متغیرهای سراسری

اگر در برنامه بخواهید یک متغیر را بین تمام قسمت‌ها به‌طور یکسان استفاده کنید، می‌توانید آن را از نوع سراسری (global) تعریف نمایید:  
شیوه تعریف یک متغیر سراسری به فرم‌های زیر است:

نام متغیر \$global

مقدار متغیر = نام متغیر \$global

## واژه‌نامه

Association  
 Break  
 Continue  
 Default  
 Define  
 HomePage  
 Indexed

انجمن  
 شکستن  
 ادامه دادن  
 پیش فرض  
 مقداردهی  
 صفحه خانگی  
 دارای زیرنویس

## خلاصه مطالب

- برای ایجاد فایل‌های PHP نیازی به محیط مخصوص این زبان نیست، حتی در محیط متنی ساده Notepad نیز می‌توان کدهای آن را نوشت.
- فایل HomePage صفحات سایت باید با نام index یا default ذخیره شود.
- مهم‌ترین ابزار تولید صفحات پویا استفاده از متغیرهاست.
- متغیرهای PHP همه منظوره هستند و نیازی نیست هنگام معرفی، نوع متغیر تعیین شود.
- هنگام مقداردهی متغیرهای PHP، نوع آن‌ها به‌طور ضمنی تعیین می‌شود.
- آرایه‌های PHP دارای دو نوع اصلی اندیس‌دار و انجمنی هستند.
- اندیس مقادیر آرایه‌های PHP از صفر شروع می‌شود یعنی اندیس اولین عنصر آرایه صفر و اندیس آخرین عنصر آرایه  $n-1$  است.
- در PHP متغیرهایی وجود دارند که به‌طور از پیش تعریف شده برای اهداف خاصی در نظر گرفته شده‌اند، در نامگذاری متغیرها باید دقت کنید، متغیری که تعریف می‌کنید، هم‌نام با متغیرهای از پیش تعریف شده نباشد.
- عملگرهای زبان PHP نیز مانند سایر زبان‌ها برای انجام عملیات محاسباتی، مقایسه‌ای، منطقی و ... به‌کار می‌روند.
- در تمام زبان‌های برنامه‌نویسی از جمله PHP، برای کنترل روند اجرای برنامه از



- ساختارهای کنترلی استفاده می‌شود که مهم‌ترین آن‌ها دستورات `if`، `if ... elseif` و `Case ... Switch` هستند.
- اگر در نظر داشته باشیم یک یا چند دستورالعمل را چند بار اجرا کنیم بهتر است از دستورات حلقه استفاده نماییم.
  - مهم‌ترین دستورات حلقه عبارتند از: `for`، `while` و `do ... while`



۱۰ - برای تعریف ثابت‌ها از چه تابعی استفاده می‌شود؟

الف - const      ب - global      ج - array      د - define

۱۱ - اگر بخواهیم یک برنامه را با ساختار if پیاده کنیم در حالتی که ۵ شرط مختلف

وجود داشته باشد، مجموع تعداد if، elseif و else در کد برنامه چند عدد خواهد بود؟

الف - ۳      ب - ۴      ج - ۵      د - ۶

۱۲ - در دستور for نوشتن کدام یک از اجزا در مقابل for اجباری است؟

الف - میزان افزایش یا کاهش متغیر      ب - عبارت شرطی

ج - مقدار اولیه حلقه      د - الف و ج

۱۳ - برای تعریف یک متغیر سراسری باید از کدام گزینه قبل از نام متغیر استفاده

شود؟

الف - global      ب - Local      ج - define      د - const

## آزمون عملی

۱ - یک آرایه برای معرفی روزهای هفته بنویسید.

۲ - در سؤال اول بدون استفاده از حلقه‌ها، هر کدام از عناصر آرایه را در یک سطر

جدا نمایش دهید.

۳ - سؤال دوم را با استفاده از حلقه for و while پیاده‌سازی کنید.

۴ - اعداد کوچک‌تر از ۲۰۰ را که مضرب ۵ هستند در خروجی نمایش دهید.

۵ - محیط و مساحت یک مربع با قطر ۲۰ را محاسبه و چاپ کنید.

## برقراری ارتباط با کاربر

### واحد کار هفتم

پس از مطالعه این واحد کار از فراگیر انتظار می‌رود که:

- ۱ - اصول دریافت اطلاعات از فرم‌های HTML را بداند.
- ۲ - اصول استفاده از متدهای GET و POST را در فرم بشناسد.
- ۳ - بتواند داده‌های ورودی را اعتبارسنجی کند.
- ۴ - بتواند از طریق PHP ایمیل ارسال کند.
- ۵ - بتواند یک فایل را از طریق فرم، Upload کند.

زمان(ساعت)	
عملی	نظری
۸	۴

## مقدمه

با مفهوم فرم در مباحث طراحی صفحات وب مقدماتی آشنا شدید، فرم‌ها اساساً برای ارسال اطلاعات از سرویس گیرنده به سرویس دهنده طراحی شده‌اند و به دلیل این که PHP یک زبان سمت سرویس دهنده است، باید بتوانید فرم‌ها را در PHP مدیریت کنید.

## ۱-۷ دریافت اطلاعات از فرم

همان طور که می‌دانید فرم‌ها برای ارسال اطلاعات از دو متد GET و POST استفاده می‌کنند.

**سؤال:** تفاوت متدهای GET و POST چیست؟

کد زیر یک فرم ایجاد می‌کند:

```
<html>
<head>
<title>form</title>
</head>
<body>

<form method="POST" action="Page1.php" name="Form1">
<input type="text" name="txt1" size="20">
<input type="submit" value="send" name="B1">
<input type="reset" value="cancel" name="B2">
</form>

</body>
</html>
```

کد فوق را در یک فایل نوشته و آن را با نام index و با پسوند htm در مسیر سرویس دهنده وب محلی ذخیره کنید. در این کد ۳ شیء ورودی وجود دارد، یک شیء کادر متنی و یک دکمه از نوع Submit برای ارسال اطلاعات به صفحه‌ای که در خاصیت action از فرم تعیین شده است (Page1.php) و یک دکمه از نوع reset برای پاک کردن محتویات فرم.

حال می‌خواهیم در صفحه Page1.php متغیر ارسال شده از سوی فایل قبل را دریافت کرده و آن را ۳ بار در خروجی چاپ کنیم.  
کد فایل Page1.php به صورت زیر نوشته می‌شود:

```
<?php
$str1 = $_POST['txt1'];
for($i = 0 ; $i < 3 ; $i++)
{
echo ($str1."<br>");
}
?>
```



شکل ۷-۱

حال فایل index.htm را اجرا کنید.  
یک کلمه یا عبارت را به دلخواه درون کادر متنی شکل ۷-۲ بنویسید و روی دکمه Send کلیک کنید.



شکل ۷-۲



شکل ۷-۳

به نوار Address شکل ۳-۷ توجه کنید، صفحه Page1.php در حال نمایش است، در این صفحه کلمه وارد شده در کادر متنی شکل ۲-۷، سه بار نوشته شده است. به کد فایل Page1.php دقت کنید:

متغیری که در کادر متنی فرم با عنوان txt1 نامگذاری شده بود توسط متد POST برای Page1.php ارسال شده است.

این متغیر ارسال شده در Page1.php با عنوان \$\_POST['txt1'] قابل شناسایی است و چون یک متغیر است باید ابتدای آن علامت \$ درج شود.

برای دسترسی راحت تر به این مقدار، بهتر است در یک متغیر مشخص قرار داده شود.

```
$str1 = $_POST['txt1'];
```

حال در Page1.php می توان با آن مانند یک متغیر عادی برخورد کرد که در اینجا توسط حلقه for آن را ۳ بار چاپ کرده ایم.

**نکته** اگر متغیرهای فایل مبدأ توسط متد GET ارسال شوند، در صفحه مقصد باید به صورت ['نام فیلد'] GET شناسایی شوند.

## ۱-۱-۷ تابع REQUEST\_\$

PHP یک تابع درون ساخته مهم دارد که می تواند صرف نظر از روش استفاده شده برای ارسال اطلاعات توسط فرم به مقادیر \$\_GET و \$\_POST دسترسی داشته باشد. این تابع می تواند در متدهای ارسال get و post استفاده شود. اگر مثال قبل با استفاده از تابع \$\_REQUEST نوشته شود، کد فایل به صورت زیر خواهد بود:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">  
<html>  
<head>  
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
```

</head>

<body>

Welcome <?php echo \$\_REQUEST["txt1"]; ?>!

<br />

</body>

</html>

## ۲-۷ اعتبارسنجی داده‌های ورودی

تعیین صحت و درستی داده‌های ورودی از طرف کاربران یکی از مهم‌ترین جنبه‌های امنیت سایت است. داده‌ای که به‌طور اشتباه از طرف کاربر وارد می‌شود، می‌تواند بر اثر اشتباه تایپی باشد، یا این‌که کاربر یک هکر بوده و قصد نفوذ یا آسیب رساندن به سایت را داشته باشد، لذا داده حتماً باید پس از نوشتن توسط کاربر اعتبارسنجی شود.


فرض کنید در یک فرم طراحی شده از کاربر درخواست شده است اطلاعات مربوط به خودش را وارد کند، اگر این اطلاعات مانند نام و نام‌خانوادگی باشد، فقط هر ترکیبی از حروف الفبا که بیش از یک حرف داشته باشد قابل قبول است اما اگر کاربر یک عدد یا تنها یک حرف الفبا وارد کند، نشان‌دهنده اشتباه بودن اطلاعات ورودی کاربر است یا به عنوان مثال اگر از کاربر بخواهیم رشته تحصیلی خود را وارد کند و کاربر abcd را وارد کند، نشان‌دهنده اشتباه بودن اطلاعات ورودی کاربر است. در این‌گونه مواقع می‌توان با در نظر گرفتن حالات مختلفی هنگام طراحی تا حدود زیادی از وارد کردن اطلاعات نادرست از طرف کاربر جلوگیری کرد. مثلاً برای ورودی‌هایی مانند نام و نام‌خانوادگی بررسی کرد که عبارت ورودی از طرف کاربر ترکیبی از چند حرف الفبا باشد، برای شماره تلفن ترکیبی از اعداد پذیرفته شود و برای ورودی‌هایی مانند رشته تحصیلی یا محل زندگی به دلیل محدود و مشخص بودن رشته‌های تحصیلی و اسامی شهرها، از لیست‌های انتخاب استفاده کرد که کاربر یکی از موارد موجود در لیست را که با اطلاعات او تطابق دارد، انتخاب کند.



به این ترتیب می‌توان تا حد زیادی مطمئن بود اطلاعات ورودی در پایگاه داده خطرناک نخواهند بود. یکی از راه‌هایی که هکرها و افرادی که قصد تخریب اطلاعات سایت را دارند، انتخاب می‌کنند این است که سعی کنند به اطلاعات ارسال شده فرم از طریق متد POST دسترسی پیدا کرده و آن‌ها را تغییر دهند. پس از تغییر اطلاعات توسط هکرها اغلب شرایطی پیش می‌آید که داده‌ها به شکل نامعتبری درمی‌آیند، لذا با بررسی اعتبار داده‌ها می‌توان از راه‌یابی داده‌های مشکوک به بانک اطلاعاتی جلوگیری کرد.

### ۳-۷ ارسال نامه الکترونیکی از طریق PHP

یکی از قابلیت‌های PHP ارسال نامه الکترونیکی است. برای این‌که این قابلیت در اختیار برنامه‌نویس قرار بگیرد باید پیکربندی فایل `php.ini` انجام گیرد.

**نکته**  در صورتی‌که از سرویس‌دهنده WAMP استفاده می‌کنید، نیاز به پیکربندی نیست و خود WAMP به‌طور خودکار تمام سرویس‌ها را آماده می‌کند.

برای پیکربندی فایل `php.ini` روی آیکن نیم‌دایره سرویس‌دهنده WAMP کلیک کنید، ماوس را روی گزینه `config files` نگه داشته و از زیر منوی باز شده روی گزینه `php.ini` کلیک کنید.

با کلیک روی گزینه Edit از نوار منو و انتخاب گزینه Find، کادر محاوره Find ظاهر می‌شود. در کادر متنی Find what عبارت `mail function` را وارد کرده و با کلیک روی دکمه Find Next، این عبارت را در کل فایل جستجو کنید.

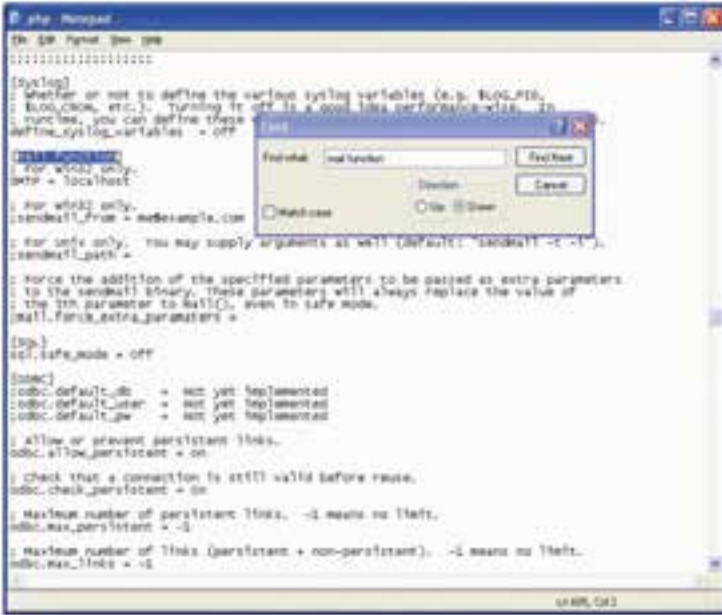
پس از عبارت [mail function] سه گزینه با عناوین `SMTP`، `sendmail_from` و `sendmail_path` وجود دارد.

برای پیکربندی در ویندوز مقابل گزینه SMTP باید آدرس سرویس‌دهنده پست الکترونیکی، برای `sendmail_path` مسیر برنامه ارسال پست الکترونیکی و هم‌چنین برای `sendmail_from` آدرس پست الکترونیکی خود را مشخص نمایید.

[mail function]

SMTP = SMTP.my.isp.net

sendmail\_from = myaccount@yahoo.com



شکل ۷-۴

### ۱-۳-۷ به کارگیری توابع ارسال ایمیل

پس از اطمینان از پیکربندی فایل `php.ini`، برای ارسال ایمیل می‌توانید از دستور `mail` استفاده کنید.

ساختار کلی این تابع به صورت زیر است:

`bool mail (string to , string object , string message);`

در صورتی که ایمیل به درستی و بدون اشکال ارسال شود، این تابع مقدار `True` را برمی‌گرداند و در غیر این صورت مقدار برگشتی تابع، `False` خواهد بود.

#### پارامترهای تابع `mail`

- نکته** • برنامه ارسال نامه الکترونیکی روی سیستم به صورت محلی اجرا نمی‌شود و این کدها حتماً باید روی سرویس‌دهنده وب واقعی اجرا شوند.
- در اغلب سرویس‌دهنده‌های PHP امکان ارسال نامه الکترونیکی به طور رایگان امکان‌پذیر است اما برای اطمینان بیشتر قبل از اینکه این تابع را در برنامه استفاده کنید، از نصب `mail server` روی سرویس‌دهنده خود مطمئن شوید.

پارامترهای این تابع در جدول زیر توضیح داده شده‌اند.

عملکرد	پارامتر
توسط این پارامتر آدرس پست الکترونیکی دریافت کننده نامه تعیین می‌شود. این پارامتر می‌تواند آدرس پست الکترونیکی چند نفر را دریافت کند. اگر بخواهید یک ایمیل را برای چند نفر ارسال کنید، باید آدرس‌های ایمیل با کاراکتر ، از یکدیگر جدا شوند.	to
مثال: "user1@yahoo.com" مثال: "user1@yahoo.com , user2@yahoo.com"	
این پارامتر عنوان نامه را تعیین می‌کند.	subject
متن اصلی نامه را معین می‌کند و می‌تواند حداکثر تا ۷۰ کاراکتر را دریافت کند.	message
این پارامتر نیز اختیاری است و چنانچه بخواهید غیر از پارامترهای فوق، پارامتر دیگری به نامه الکترونیکی اضافه کنید، می‌توانید در این بخش درج نمایید.	parameters

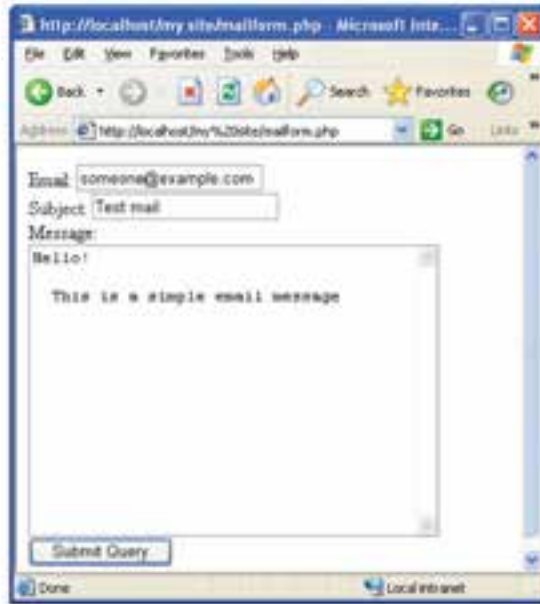
### مثال:

مثال زیر یک نامه الکترونیکی ساده را در صفحه وب ارسال می‌کند:

```
<?php
$to = "someone@example.com";
$subject = "Test mail";
$message = "Hello! This is a simple email message.";
$from = "someone@example.com";

mail($to,$subject,$message);
echo "Mail Sent.";
?>
```

در یک صفحه وب بهتر است فرم مخصوصی مانند شکل ۵-۷ به منظور دریافت پارامترهای نامه الکترونیکی و سپس ارسال آن طراحی شود.



شکل ۷-۵ نمونه فرم ارسال نامه الکترونیکی

**نکته** چنانچه در صورت نصب بودن mail server توسط برنامه، نامه الکترونیکی ارسال نشود، در فایل php.ini از ابتدای عبارت sendmail\_path علامت ; را بردارید و مسیر mail server را مقابل آن نوشته و هم‌چنین مقدار smtp\_port را برابر با ۲۵ قرار دهید و تغییرات را ذخیره کنید.

## ۴-۷ بارگذاری (Upload) فایل‌ها از طریق فرم

کاربر می‌تواند توسط تابع ftp\_put یک فایل موجود روی سیستم خود را توسط فرم upload کند.

شکل کلی این دستور به صورت زیر است:

```
bool ftp_put (resource $ftp_stream , string $remote_file , string $Local_file , int $mode);
```

پارامترهای تابع ftp\_put

**ftp\_stream**: یک اتصال FTP برقرار می‌کند.

**remote\_file**: مسیر و نام فایل راه دور را تعیین می‌کند.

**Local\_file**: مسیر و نام فایل محلی را تعیین می‌کند.  
**mode**: شیوه انتقال را تعیین می‌کند که می‌تواند یکی از مقادیر FTP\_ASCII یا FTP\_BINARY را بپذیرد.  
اگر بارگذاری فایل با موفقیت انجام شود، تابع مقدار True را برمی‌گرداند، در غیر این صورت مقدار برگشت داده شده توسط این تابع False خواهد بود.

مطالعه آزاد

### ایجاد فرم بارگذاری فایل

برای بارگذاری یک فایل در اینترنت نیز باید آن را از کاربر دریافت نمود، دریافت فایل از کاربر و ارسال آن از طریق فرم انجام می‌شود.  
به منظور انتخاب فایل برای بارگذاری، باید نوع عنصر input را از نوع فایل قرار دهید:

```
<input type="file" name="file" id="file" />
```

در این صورت دکمه Browse نیز روی فرم ظاهر می‌شود که امکان انتخاب فایل از طریق کادر محاوره Choose file را فراهم می‌کند.  
هنگام طراحی فرم بارگذاری فایل، باید enctype را از نوع multipart/form-data قرار دهید، این نوع کد گذاری برای ارسال اطلاعات باینری مانند محتویات فایل مناسب است.

**مثال:** کد زیر یک فرم انتخاب فایل برای بارگذاری را ایجاد می‌کند:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">  
<html>  
<head>  
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>  
</head>  
<body>  
  
<form action="upload_file.php" method="post"
```

```
enctype="multipart/form-data">
<label for="file">Filename:</label>
<input type="file" name="file" id="file" />
<br />
<input type="submit" name="submit" value="Submit" />
</form>

</body>
</html>
```

با اجرای کد فوق، مرورگر مشابه شکل ۲-۵ یک فایل را از کاربر دریافت می‌کند.



شکل ۶-۷

در این کد، فایل دریافت شده، برای اسکریپت `upload_file.php` ارسال می‌شود. به منظور پیاده‌سازی اسکریپت `upload_file.php` می‌توانید کد بعد را تایپ کنید و آن را با نام `upload_file.php` ذخیره نمایید.

```
<?php
if ($_FILES["file"]["error"] > 0)
{
echo "Error: " . $_FILES["file"]["error"] . "<br />";
}
else
{
```

```
echo "Upload: " . $_FILES["file"]["name"] . "<br />";  
echo "Type: " . $_FILES["file"]["type"] . "<br />";  
echo "Size: " . ($_FILES["file"]["size"] / 1024) . " Kb<br />";  
echo "Stored in: " . $_FILES["file"]["tmp_name"];  
}  
?>
```

اگر به طور مثال اسم فایل بارگذاری شده test1.htm باشد، نتیجه اجرای اسکریپت upload\_file.php به صورت زیر خواهد بود:

Upload: test1 htm

Type: text/html

Size: 0.14609375 Kb

Stored in: C:\wamp\tmp\php19.tmp

فرم بارگذاری فایل، چند آرگومان را به اسکریپت PHP ارسال می کند. برای شناسایی هر کدام از این آرگومان ها می توان از تابع مشخصی استفاده نمود. توابع مهم و کاربردی فرم بارگذاری عبارتند از:

• **\$\_FILES["file"]["error"]**: نتیجه خطای احتمالی، هنگام بارگذاری فایل را مشخص می کند.

اگر هنگام بارگذاری فایل هیچ خطایی رخ نداده باشد، این تابع مقدار صفر را برمی گرداند.

• **\$\_FILES["file"]["name"]**: نام فایل بارگذاری شده را مشخص می کند.

• **\$\_FILES["file"]["type"]**: نوع و پسوند فایل بارگذاری شده را مشخص می کند. به

عنوان مثال برای فایل های تصویری می تواند image/ jpg، image/tiff، image/ gif و ... باشد.

• **\$\_FILES["file"]["size"]**: اندازه فایل بارگذاری شده را برحسب بایت مشخص

می کند.

• **\$\_FILES["file"]["tmp\_name"]**: مسیر ریشه فایل های ذخیره شده مربوط به

سرویس دهنده را مشخص می کند. تا زمانی که فایل ها توسط سرویس دهنده WAMP اجرا

می شوند، این فایل ها در مسیر نصب سرویس دهنده WAMP ذخیره می شوند.



**نکته** فراهم کردن امکان بارگذاری فایل توسط کاربر در وب سایت به دلیل احتمال وجود تهدیدات امنیتی می تواند خطرات جدی در پی داشته باشد، بنابراین فقط باید در صورت ضرورت از این قابلیت در صفحات وب استفاده کنید.

مثال قبل شکل بسیار ساده ای از بارگذاری فایل است، شما به عنوان طراح وب سایت می توانید محدودیت هایی را برای فایل انتخاب شده از طرف کاربر تعیین کنید. به طور مثال اگر در صفحه وب از کاربر خواسته باشید عکس خودش را بارگذاری کند، می توانید از او بخواهید تصویر ارسالی خود را از نوع gif، bmp یا jpg با اندازه محدود انتخاب کند، سپس در کد اسکریپت upload\_file.php نوع و اندازه آن را بررسی کنید و در صورت تناقض به فایل اجازه بارگذاری روی سرور ندهید. در این صورت می توانید مشابه کد زیر را برای اسکریپت upload\_file.php بنویسید:

```
<?php
if ((($_FILES["file"]["type"] == "image/gif")
|| ($_FILES["file"]["type"] == "image/jpg")
|| ($_FILES["file"]["type"] == "image/bmp")))
&& ($_FILES["file"]["size"] < 20000))
{
if ($_FILES["file"]["error"] > 0)
{
echo "Error: " . $_FILES["file"]["error"] . "<br />";
}
else
{
echo "Upload: " . $_FILES["file"]["name"] . "<br />";
echo "Type: " . $_FILES["file"]["type"] . "<br />";
echo "Size: " . ($_FILES["file"]["size"] / 1024) . " Kb<br />";
echo "Stored in: " . $_FILES["file"]["tmp_name"];
}
}
```



```
}  
else  
{  
    echo "Invalid file";  
}  
?>
```

### تمرین

از کاربر بخواهید در یک صفحه وب رزومه کاری خود را در قالب فایل متنی با پسوند .doc یا .rtf و اندازه حداکثر ۱۰۰ کیلوبایت، ارسال کند و اگر فایل انتخاب شده کاربر این ویژگی‌ها را نداشت، پیغام مناسبی نمایش داده شود.

### ذخیره فایل بارگذاری شده

فایل‌های کپی شده در مسیر ریشه، پس از خاتمه اسکریپت، حذف می‌شوند، به همین دلیل اگر بخواهید فایل بارگذاری شده از طرف سرویس‌گیرنده را در سرویس‌دهنده استفاده کنید، باید عمل انتقال فایل و ذخیره آن را در مسیری غیر از مسیر ریشه سرویس‌دهنده نیز انجام دهید.

برای انتقال فایل‌ها از ریشه، تابع `move_uploaded_file` مورد استفاده قرار می‌گیرد. برای این کار می‌توانید کد زیر را به انتهای اسکریپت `upload_file.php` اضافه کنید:

```
move_uploaded_file($_FILES["file"]["tmp_name"],  
"upload/" . $_FILES["file"]["name"]);  
echo "Stored in: " . "upload/" . $_FILES["file"]["name"];
```

هنگام کپی فایل روی سرویس‌گیرنده ممکن است فایل هم نام و هم نوع با فایل کاربر وجود داشته باشد، در نتیجه کپی فایل کاربر باعث از بین رفتن فایل هم نامی که از قبل وجود داشته، می‌شود، برای پیشگیری از این مشکل می‌توانید از تابع `file_exists` برای بررسی تکراری نبودن فایل استفاده کنید.

**توصیه:** قبل از انجام عمل کپی و نوشتن اسکریپت مربوط به آن در مسیر ذخیره وب سایت خود، پوشه‌ای را برای ذخیره فایل‌های بارگذاری شده ایجاد کنید.

### مثال:

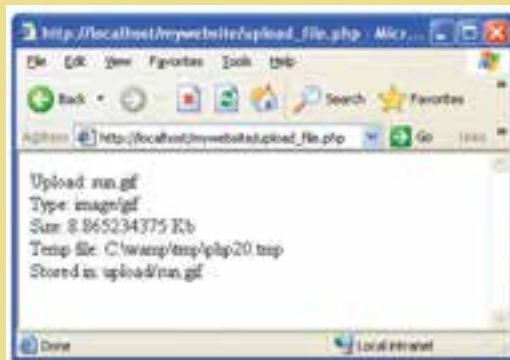
کد زیر در صورت عدم وجود خطا، مشخصات فایل ارسال شده توسط کاربر را در خروجی نشان می‌دهد، سپس اگر فایل تکراری بود، پیغام already exists به همراه نام فایل ظاهر می‌شود، در غیر این صورت عمل انتقال آن به پوشه upload که آن را در مسیر سرویس‌دهنده ایجاد کرده‌ایم، انجام می‌شود.

```
<?php
if ((($_FILES["file"]["type"] == "image/gif")
|| ($_FILES["file"]["type"] == "image/jpg")
|| ($_FILES["file"]["type"] == "image/bmp")))
&& ($_FILES["file"]["size"] < 20000))
{
if ($_FILES["file"]["error"] > 0)
{
echo "Return Code: " . $_FILES["file"]["error"] . "<br />";
}
else
{
echo "Upload: " . $_FILES["file"]["name"] . "<br />";
echo "Type: " . $_FILES["file"]["type"] . "<br />";
echo "Size: " . ($_FILES["file"]["size"] / 1024) . " Kb<br />";
echo "Temp file: " . $_FILES["file"]["tmp_name"] . "<br />";

if (file_exists("upload/" . $_FILES["file"]["name"]))
{
echo $_FILES["file"]["name"] . " already exists. ";
}
}
```

```
else
{
    move_uploaded_file($_FILES["file"]["tmp_name"],
"upload/" . $_FILES["file"]["name"]);
    echo "Stored in: " . "upload/" . $_FILES["file"]["name"];
}
}
}
else
{
    echo "Invalid file";
}
?>
```

شکل زیر نتیجه اجرای کد فوق را هنگام بارگذاری فایل تصویری sun.gif نشان می‌دهد.



شکل ۷-۷

## ۵-۷ طراحی فرم بازخورد (Feedback)

یکی از قابلیت‌هایی که در بسیاری از سایت‌ها استفاده می‌شود، ارائه فرم بازخورد است، این گونه فرم‌ها بیشتر به منظور دریافت نظرات کاربران در مورد یک موضوع خاص به کار می‌رود.

به عنوان مثال می توان به سایت های صدا و سیما اشاره کرد که پس از پخش یک فیلم سینمایی یا یک سریال تلویزیونی فرم هایی را روی سایت خود به منظور بررسی تأثیرات آن بر مخاطب، قرار می دهند.

این فرم ها حاوی سؤالاتی در مورد فیلم یا سریال هستند که کاربران با پر کردن آن ها و ارسال فرم پر شده به سرویس دهنده نظرات خود را در اختیار مدیر سایت قرار می دهند.

در طراحی یک فرم بازخورد نکات متعددی را باید در نظر گرفت:

- از طراحی سؤالات با تعداد زیاد خودداری کنید.
  - سؤالات طرح شده دارای جواب های کوتاه باشند.
  - کاربر بتواند پس از ارسال نظر خود، آمارسنجی نظرات سایر کاربران را ببیند.
  - کاربر بتواند انعکاس نظرات خود را به طور عملی ببیند.
- شکل های ۷-۸ و ۷-۹ دو نمونه فرم بازخورد را به شما نشان می دهند.



شکل ۷-۸



شکل ۷-۹

## واژه‌نامه

Feedback

بازخورد

## خلاصه مطالب

- برای دریافت اطلاعات از سروریس گیرنده و ارسال آن‌ها به سروریس دهنده از فرم‌ها استفاده می‌شود.
- ارسال اطلاعات در فرم به دو شیوه GET و POST صورت می‌گیرد.
- برای شناسایی متغیرهای ارسالی از طرف فایل مبدأ در فایل مقصد از توابع GET و POST استفاده می‌شود.
- اعتبارسنجی داده‌های ارسال شده یکی از مهم‌ترین جنبه‌های امنیت سایت است، زیرا هکرها با دسترسی به اطلاعات ارسالی و تخریب آن‌ها غالباً کدهای نامعتبری تولید می‌کنند که با بررسی صحت و اعتبار آن‌ها تا حدود زیادی می‌توان از حمله افراد هکر و ... جلوگیری کرد.
- برای ارسال نامه الکترونیکی از طریق ایمیل باید تنظیماتی روی فایل php.ini انجام شود، اما اگر از WAMP استفاده می‌کنید، این تنظیمات به‌طور خودکار انجام می‌شوند.
- ارسال ایمیل از طریق تابع Mail صورت می‌گیرد.

## آزمون نظری

- ۱ - اگر در ارسال اطلاعات از فایل a.php به فایل b.php از متد POST استفاده شده باشد متغیرهای دریافتی توسط چه عبارتی شناخته می‌شوند؟
- الف - POST  
ب - \_POST  
ج - GET  
د - \_GET
- ۲ - در پیکربندی فایل php.ini برای ارسال ایمیل، آدرس سرویس دهنده پست الکترونیکی مقابل کدام گزینه نوشته می‌شود؟
- الف - sendmail\_path  
ب - sendmail\_from  
ج - SMTP  
د - function
- ۳ - تابع Mail در صورتی که با موفقیت نامه را ارسال کند چه مقداری را به سرویس دهنده می‌فرستد؟
- الف - True  
ب - False  
ج - متن نامه الکترونیکی  
د - موضوع نامه الکترونیکی
- ۴ - در تابع بارگذاری فایل‌ها از طریق فرم، پارامتر mode چه مقادیری را می‌تواند بپذیرد؟
- الف - FTP\_ASCII  
ب - FTP\_BINARY  
ج - True  
د - الف و ب

## آزمون عملی

- ۱ - در یک صفحه وب نام، نام خانوادگی، شماره شناسنامه و ایمیل کاربر را دریافت کنید و اطلاعات دریافتی را در فایل دیگر توسط مرورگر نمایش دهید.
- ۲ - تابعی بنویسید که اطلاعات دریافت شده از کاربر در سؤال اول را به ایمیل او ارسال کند.
- ۳ - تابع سؤال دوم را طوری تغییر دهید که اطلاعات برای چند نفر به طور گروهی ارسال شود (آدرس‌های ایمیل را به طور اختیاری وارد کنید).

## مدیریت فایل‌ها و پوشه‌ها

### واحد کار هشتم

پس از مطالعه این واحد کار از فراگیر انتظار می‌رود که:

- ۱ - اصول کار با فایل‌ها را بشناسد.
- ۲ - بتواند یک فایل را باز کند.
- ۳ - بتواند یک فایل را ببندد.
- ۴ - بتواند یک فایل را نمایش دهد.
- ۵ - بتواند اطلاعات را از فایل بخواند یا در آن بنویسد.
- ۶ - اصول کار با پوشه‌ها را بداند.
- ۷ - بتواند یک پوشه جدید ایجاد کند.
- ۸ - بتواند یک پوشه را حذف کند.
- ۹ - بتواند فایل‌ها را از طرف سرویس‌گیرنده بارگذاری نماید.

زمان(ساعت)	
عملی	نظری
۴	۲



## مقدمه

در اغلب زبان‌های برنامه‌نویسی امکان انجام مدیریت روی فایل‌ها هنگام اجرا وجود دارد، در زبان PHP نیز امکانات مختلفی برای دسترسی به فایل‌ها وجود دارد. مهم‌ترین این عملیات عبارتند از: ایجاد یک فایل، به‌روزرسانی یک فایل، حذف یک فایل و ...

## ۱-۸ کار با فایل‌ها

### ۱-۸-۱ نوشتن محتویات فایل در مرورگر

یکی از پرکاربردترین اعمالی که روی فایل انجام می‌شود، عمل نوشتن اطلاعات فایل در مرورگر است.

برای این که قادر باشیم اطلاعات درون فایل را دریافت کنیم، باید ابتدا فایل را باز کنیم. ساختار دستور باز کردن فایل به صورت زیر است:

؛ ( "r" و "psوند فایل. نام فایل" ) fopen = متغیر نام فایل \$

برای خواندن اطلاعات، پس از باز کردن فایل باید از دستور (متغیر نام فایل \$) fgets استفاده کنیم.

در هر فایل یک شناسه برای مشخص شدن انتهای آن وجود دارد، در PHP، رسیدن به انتهای فایل توسط دستور (متغیر نام فایل \$) feof تعیین می‌شود.

حال با استفاده از سه دستور فوق می‌خواهیم فایل "Newest.txt" را باز کرده و محتویات آن را در مرورگر نمایش دهیم.

رسیدن به انتهای فایل توسط مرورگر قابل تشخیص نیست، لذا باید از شرط (نام فایل \$) feof استفاده شود. به این ترتیب که چک کنیم تا زمانی که به انتهای فایل نرسیده است عمل خواندن اطلاعات انجام شود و اطلاعات خوانده شده را توسط دستور echo در خروجی چاپ کنیم.

تابع (نام فایل \$) feof یک مقدار boolean را برمی‌گرداند یعنی دارای دو مقدار True یا False است، اگر به انتهای فایل رسیده باشیم، مقدار آن True و در غیر این صورت مقدار

آن False خواهد بود، ما قصد داریم برنامه را طوری بنویسیم که تا زمانی که به انتهای فایل نرسیده است عملیات را انجام دهد.

برای پیاده کردن این عمل باید از حلقه‌ها استفاده کنیم و چون حلقه‌ها زمانی اجرا می‌شوند که شرط مقابل آن‌ها True باشد و شرط بررسی و ادامه این برنامه توسط تابع False، feof است، لذا باید از یک علامت نقیض (!) در ابتدای نام تابع استفاده کنیم، کد این برنامه به این صورت خواهد بود:

```
<?php
$file=fopen("Newest.txt","r");
while(!feof($file))
{
    echo (fgets($file));
}
?>
```

دقت کنید که در مسیر سرویس دهنده محلی شما فایل Newest.txt وجود داشته باشد، اگر این فایل در مسیر فوق موجود نیست، آن را ایجاد کرده و چند جمله داخل آن بنویسید، فایل را ذخیره کرده، از آن خارج شوید.

پس از اجرای برنامه فوق در مرورگر خواهید دید که محتویات فایل متنی Newest.txt در مرورگر ظاهر شده است.

**نکته** در دستور fopen () اگر فایلی که به عنوان پارامتر تعیین می‌شود از قبل وجود نداشته باشد، با استفاده از این دستور ساخته خواهد شد.

یکی دیگر از دستورات خواندن از فایل، دستور fread است. تفاوت دستور fread با دستور fgets در این است که دستور fgets یک پارامتر دارد که مشخص‌کننده نام فایل است و تمام محتویات فایل مشخص شده را می‌خواند اما توسط دستور fread می‌توان تعیین کرد که تعداد مشخصی کاراکتر از فایل خوانده شود.

```
<?php
$file=fopen("Newest.txt","r");
```

```
while(!feof($file))
{
    echo (fread( $file, 10 ));
}
?>
```

## ۲-۱-۸ نوشتن در فایل

به منظور نوشتن یک عبارت در فایل از دستور ( fputs ) استفاده می‌شود. این دستور عبارت موردنظر را به محتوای فایل اضافه می‌کند.

**نکته** قبل از اقدام به هرگونه عملیات خواندن یا نوشتن فایل باید با استفاده از دستور fopen، فایل موردنظر باز شده و در انتها نیز با استفاده از دستور fclose بسته شود.

```
<?php
$file=fopen("Newest.txt","w");
$str="This is same file with different text";
fputs($file,$str);
$file=fopen("Newest.txt","r");
while(!feof($file))
{
    echo fgets($file)."<br/>";
}
?>
```

دستور دیگری نیز برای نوشتن در فایل وجود دارد. این دستور fwrite است. دستور fwrite و fputs دو پارامتر ورودی دارند: پارامتر اول متغیری است که نام فایل را مشخص می‌کند و پارامتر دوم، عبارتی است که باید در فایل نوشته شود.

## ۲-۸ کار با پوشه‌ها

### ۱-۲-۸ ایجاد پوشه جدید

دستور mkdir ( ) یک پوشه جدید را در مسیر ریشه سرویس دهنده محلی ایجاد می‌کند.

مثال زیر در مسیر ریشه سرویس‌دهنده یک پوشه با نام dir1 را ایجاد می‌کند.  
مثال:

```
<?php  
mkdir (dir1);  
?>
```

## ۲-۲-۸ حذف پوشه

دستور `rmdir ()` پوشه مشخص شده در پارامتر ورودی تابع را حذف می‌کند.  
مثال زیر پوشه ایجاد شده در مثال قبل را حذف می‌کند.  
مثال:

```
<?php  
rmdir (dir1);  
?>
```

## واژه‌نامه

Make  
Remove

ساختن  
حذف

## خلاصه مطالب

- در زبان PHP نیز مانند سایر زبان‌های برنامه‌نویسی امکان انجام اعمال مدیریتی توسط کد برنامه وجود دارد، این اعمال مدیریتی عبارتند از: ایجاد فایل یا پوشه جدید، به‌روز رسانی یک فایل، حذف یک فایل یا پوشه و ...
- از دستور (fputs) برای نوشتن محتویات فایل در مرورگر استفاده می‌شود (خواندن از فایل و نوشتن در مرورگر).
- تابع feof یک مقدار Boolean را برای اینکه بدانیم به انتهای فایل رسیده‌ایم یا خیر برمی‌گرداند.
- برای نوشتن در فایل از دستورات fwrite و fputs استفاده می‌شود.
- برای ایجاد پوشه جدید از دستور mkdir و برای حذف آن از دستور rmdir استفاده می‌شود.

## آزمون نظری

- ۱- اگر بخواهیم اطلاعات را از یک فایل بخوانیم، اولین دستوری که باید به کار ببریم چیست؟  
الف- fgetc () - ب- fopen () - ج- feof () - د- fputs ()
- ۲- پس از دریافت محتویات فایل برای نوشتن در خروجی از چه دستوری استفاده می‌شود؟  
الف- fgetc () - ب- fputs () - ج- echo - د- fwrite ()
- ۳- برای نوشتن یک عبارت در فایل از چه دستوری استفاده می‌شود؟  
الف- fgetc () - ب- fputs () - ج- echo - د- print
- ۴- ایجاد یک فایل جدید توسط چه دستوری انجام می‌شود؟  
الف- fopen - ب- fgetc () - ج- make () - د- fputs ()
- ۵- برای ایجاد پوشه جدید از چه دستوری استفاده می‌شود؟  
الف- mkdir () - ب- makedir () - ج- diropen () - د- mkdir ()
- ۶- اگر توسط کد برنامه‌نویسی یک فایل یا پوشه جدید ایجاد شود این فایل یا پوشه در چه مسیری ساخته می‌شود؟  
الف- My Documents - ب- میز کار
- ج- ریشه سرویس دهنده محلی - د- توسط پارامترهای تابع تعیین می‌شود.

## آزمون عملی

- ۱ - در مسیر سرویس‌دهنده محلی یک پوشه با نام خودتان بسازید.
- ۲ - یک فایل متنی ایجاد کرده، سپس مشخصات خود را در آن درج کنید.
- ۳ - محتویات فایل سؤال دوم را در مرورگر نشان دهید.

## کار با توابع

### واحد کار نهم

پس از مطالعه این واحد کار از فراگیر انتظار می‌رود که:

۱ - اصول کار با توابع رشته‌ای زیر را بداند:

- strlen
- strtolower
- strtoupper
- ucfirst
- ucwords
- addslashes

۲ - اصول کار با توابع تاریخ و زمان را بشناسد.

۳ - بتواند تابع جدیدی ایجاد کند.

۴ - بتواند اعداد و متغیرها را در توابع به‌کار ببرد.

زمان(ساعت)	
عملی	نظری
۱۰	۵



## مقدمه

در طراحی یک سایت پویا اطلاعات ورودی کاربران نقش اصلی روی تصمیمات سرویس دهنده دارند، لذا زبان‌هایی که به منظور طراحی سایت‌های پویا به کار برده می‌شوند باید بتوانند پردازش‌های لازم را روی اطلاعات ورودی انجام دهند، اغلب پردازش‌ها توسط توابع از پیش تعریف شده‌ای قابل انجام هستند.

زبان PHP نیز دستورات متعددی برای کار با رشته‌ها و دیگر متغیرها دارد که در ادامه این مبحث با مهم‌ترین آن‌ها آشنا خواهیم شد.

## ۹-۱ توابع رشته‌ای

### ۹-۱-۱ تابع strlen

این تابع طول رشته‌ای را که به عنوان پارامتر ورودی به آن داده می‌شود برمی‌گرداند، در نتیجه مقدار برگشت داده شده آن یک عدد صحیح است.

ساختار این تابع به صورت زیر است:

strlen («پارامتر ورودی»)

```
<?php
```

```
$str1 = 'abcdef';
```

```
echo "str1."---length is : ".strlen($str1)."<br>";
```

```
$str2 = ' ab cd ';
```

```
echo "str2."---length is : ".strlen($str2)."<br>";
```

```
?>
```



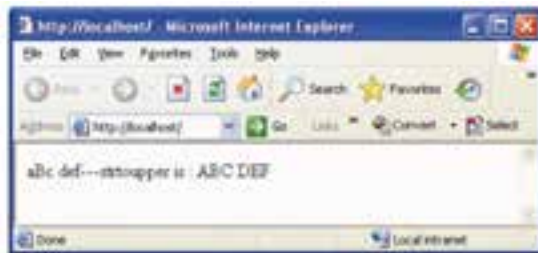
شکل ۹-۱

همان‌طور که در نتیجه اجرای کد قبل ملاحظه می‌کنید تابع `strlen` توانایی شمارش کاراکتر Space را نیز دارد، به همین دلیل رشته 'ab cd' که شامل ۴ حرف و ۳ فضای خالی است به عنوان پارامتر ورودی تابع، نتیجه ۷ را نشان می‌دهد.

## ۹-۱-۲ تابع `strtoupper()`

این تابع رشته‌ای را که به عنوان پارامتر ورودی می‌گیرد به حروف بزرگ الفبای انگلیسی تبدیل می‌کند.

```
<?php  
$str1 = 'aBc def';  
echo "$str1". "---strtoupper is : ".strtoupper($str1). "<br>";  
?>
```



شکل ۹-۲

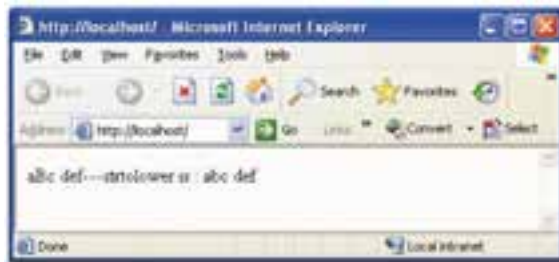
در کد فوق توسط این تابع تمام حروف مربوط به پارامتر ورودی "aBc def" به حروف بزرگ تبدیل شده‌اند. فاصله‌ها (Space) در خروجی این تابع تأثیر ندارند و باقی می‌مانند

## ۹-۱-۳ تابع `strtolower()`

این تابع عکس تابع `strtoupper` عمل می‌کند و تمام حروف پارامترهای ورودی را به حروف کوچک الفبای انگلیسی تبدیل می‌کند.  
مثال:

```
<?php  
$str1 = 'aBc def';
```

```
echo "$str1". "---strtolower is : ".strtolower($str1). "<br>";  
?>
```



شکل ۳-۹

### تابع ucfirst () ۱-۱-۴

این تابع حرف اول پارامتر ورودی را به حرف الفبایی بزرگ انگلیسی تبدیل می‌کند. این تابع جز اولین حرف از رشته، هیچ‌کدام از حروف دیگر را تحت تأثیر قرار نمی‌دهد و بدون تغییر برمی‌گرداند.

مثال:

```
<?php  
$foo = 'hello world!';  
echo "$foo". "---ucfirst is : ".ucfirst($foo). "<br>";  
$bar = 'HELLO WORLD!';  
echo "$bar". "---ucfirst is : ".ucfirst($bar). "<br>";  
  
echo "$bar". "---ucfirst(strtolower) is : ".ucfirst(strtolower($bar)). "<br>";  
?>
```



شکل ۴-۹

## ۵-۱-۹ تابع (ucword)

این تابع حرف اول تمام کلمات موجود در پارامتر ورودی را به حرف الفبای بزرگ انگلیسی تبدیل می‌کند. این تابع از کاراکتر Space برای تشخیص کلمه بعدی استفاده می‌کند، یعنی هر جا به این کاراکتر برسد، کلمه قبلی را خاتمه یافته فرض می‌کند و حرف بعد از کاراکتر Space را اولین حرف از کلمه بعدی می‌داند.

```
<?php  
$foo = 'hello world!';  
echo "$foo". "---ucword is : ".ucwords($foo). "<br>";  
  
$bar = 'HELLO WORLD!';  
echo "$bar". "---ucword is : ".ucwords($bar). "<br>";  
echo "$bar". "---ucword is : ".ucwords(strtolower($bar)). "<br>";  
?>
```



شکل ۵-۹

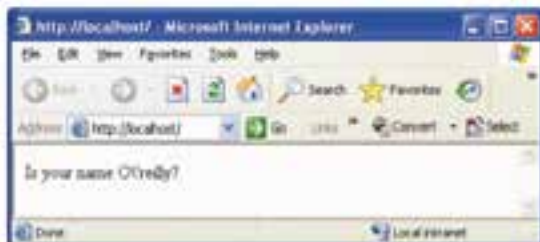
### تمرین:

برنامه‌ای بنویسید که در آن یک متغیر با نام `str1` تعریف شده است که با مقادیر نام، نام خانوادگی و نام مدرسه شما به دنبال یکدیگر (با رعایت فاصله) مقداردهی شده است، حال متغیر `str1` را طوری چاپ کنید که حرف اول، نام، نام خانوادگی و نام مدرسه به صورت حروف بزرگ الفبای انگلیسی و سایر حروف این کلمات با حروف کوچک الفبا، چاپ شود.

## ۹-۱-۶ تابع addslashes ()

همان‌طور که می‌دانید کاراکتر (') برای متغیرهای رشته‌ای استفاده می‌شود و وارد کردن این کاراکتر در پایگاه داده‌ها مشکل‌آفرین می‌شود. اما اگر در بعضی مواقع این کاراکتر جزئی از رشته باشد باید به نوعی بتوان آن را در پایگاه داده درج کرد. به عنوان مثال اسم (o'reilly) را در نظر بگیرید، کاراکتر (') جزئی از اسم است و تنها راه چاره برای درج آن در پایگاه داده استفاده از کاراکتر (\\) است. تابع addslashes () این کار را به‌طور خودکار انجام می‌دهد.

```
<?php  
$str = "Is your name O'reilly?";  
  
echo addslashes($str);  
?>
```



شکل ۹-۶

## ۹-۲ توابع تاریخ و زمان

### ۹-۲-۱ تابع date ()

یک پارامتر را به عنوان ورودی دریافت می‌کند و سپس یک رشته را به عنوان خروجی برمی‌گرداند. پارامتر ورودی یک قالب درخواستی برای برگرداندن تاریخ است. هر کدام از قالب‌ها یک شناسه استاندارد دارند که در جدول ۹-۱ چند نمونه از این شناسه‌ها بیان شده است.

جدول ۹-۱

شناسه	توضیحات	مثال (نمونه رشته خروجی)
d	شماره روز و ماه به صورت عدد (در صورتی که شماره تک رقمی باشد به ابتدای آن صفر را اضافه می کند).	01 to 31
j	شماره روز و ماه به صورت عدد (بدون درج صفر)	1 to 31
z	تعداد روزهای گذشته از سال	0 through 365
Y	شماره سال را به صورت یک عدد ۴ رقمی برمی گرداند.	1999 or 2003
y	شماره سال را به صورت یک عدد ۲ رقمی برمی گرداند.	99 or 03
a	فقط تعیین می کند که قبل از ظهر است یا بعد از ظهر (با حروف الفبای کوچک انگلیسی)	am or pm
A	فقط تعیین می کند که قبل از ظهر است یا بعد از ظهر (با حروف الفبای بزرگ انگلیسی)	AM or PM
P	ساعت و دقیقه را به وقت گرینویچ تعیین می کند.	+02:00
c	سال - روز - ماه - ساعت - دقیقه - ثانیه - صدم ثانیه	2010-02-12T15:19:21+00:00
r	سال - روز - ماه - ساعت - دقیقه - ثانیه	Thu, 21 Dec 2000 16:01:07 +0200

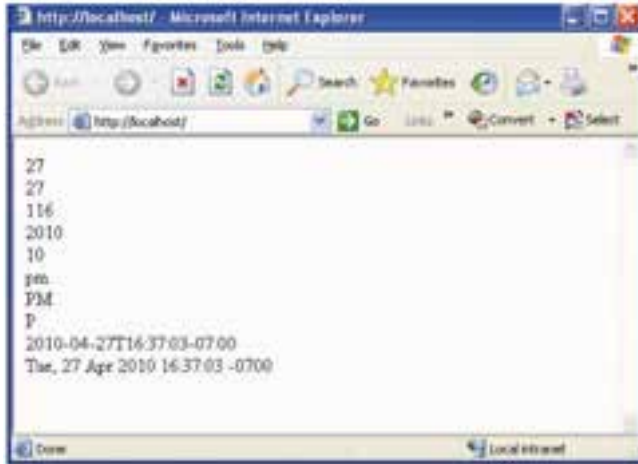
<?php

```

echo date("d")."<br>";
echo date("j")."<br>";
echo date("z")."<br>";
echo date("Y")."<br>";
echo date('y')."<br>";
echo date("a")."<br>";
echo date("A")."<br>";
echo date("P")."<br>";
echo date("c")."<br>";
echo date("r")."<br>";

```

?>



شکل ۷-۹

### تمرین

برنامه‌ای بنویسید که زمان جاری سیستم را بر حسب ساعت و دقیقه به وقت گرینویچ، چاپ کند.

### ۲-۲-۹ تابع time

این تابع در PHP لحظه جاری را به صورت یک عدد صحیح شامل ثانیه، دقیقه، ساعت، روز، ماه و سال به وقت گرینویچ نشان می‌دهد. کد زیر را اجرا کنید:

```
<?php
```

```
$Now = time();
```

```
echo $Now;
```

```
?>
```



شکل ۸-۹

تابع `time()` به تنهایی کاربرد کمی دارد اما می‌تواند با توابع دیگر به منظور تبدیل ساعت نواحی مختلف براساس اختلاف زمانی که با گرینویچ دارند، مورد استفاده قرار بگیرد.

## ۳-۹ توابع عددی

### ۳-۹-۱ تابع `abs()`

این تابع قدر مطلق پارامتر ورودی خود را محاسبه می‌کند.

```
<?php
$num1 = -243;
echo ("$num1."<br>");
$num2 = abs($num1);
echo "$num2."<br>";
?>
```



شکل ۹-۹

### ۳-۹-۲ تابع `floor()`

این تابع عددی را که به عنوان پارامتر ورودی دریافت می‌کند به عدد صحیح کوچک‌تر تبدیل کرده و مقدار اعشار آن را از بین می‌برد.

مثال زیر با استفاده از پارامترهای مختلف، رفتار این تابع را به خوبی نشان می‌دهد.

مثال:

```
<?php
echo(floor(0.60) . "<br />");
echo(floor(0.40) . "<br />");
echo(floor(5) . "<br />");
echo(floor(5.1) . "<br />");
echo(floor(-5.1) . "<br />");
echo(floor(-5.9))
?>
```



خروجی:

0  
0  
5  
5  
-6  
-6

### ۹-۳-۲ تابع fmod()

این تابع دو پارامتر را دریافت می‌کند و باقیمانده حاصل تقسیم پارامتر اول بر پارامتر دوم را بر می‌گرداند.

مثال:

```
<?php  
$r = fmod(15,2);  
echo $r  
?>
```

خروجی:

1

### ۹-۳-۴ تابع max()

این تابع دو پارامتر را به عنوان ورودی دریافت می‌کند، سپس عدد بزرگ‌تر را به عنوان خروجی برمی‌گرداند.

مثال:

```
<?php  
echo(max(5,7) . "<br />");  
echo(max(-3,5) . "<br />");  
echo(max(-3,-5) . "<br />");
```

```
echo(max(7.25,7.30))
```

```
?>
```

خروجی:

7

5

-3

7.3

### ۵-۳-۹ تابع min()

این تابع دو پارامتر را به عنوان ورودی دریافت می‌کند، سپس عدد کوچک‌تر را به عنوان خروجی برمی‌گرداند.

### ۶-۳-۹ تابع rand()

از تابع rand() برای تولید اعداد صحیح تصادفی استفاده می‌شود. اگر این تابع بدون پارامتر ورودی فراخوانی شود، عددی که تولید می‌شود مقداری بین صفر و بیشترین عدد صحیح روی سیستم را خواهد داشت، اما چنانچه بخواهید محدوده مشخصی برای تولید اعداد تصادفی در نظر بگیرید باید دو پارامتر ابتدا و انتهای محدوده را به عنوان ورودی تابع قرار دهید.

به طور مثال حاصل فراخوانی تابع به صورت rand() عدد تصادفی صحیح و بزرگ‌تر از صفر خواهد بود، اما با فراخوانی تابع به شکل rand(10,100) عددی تصادفی در محدوده بین ۱۰ و ۱۰۰ تولید خواهد کرد.

مثال:

```
<?php
```

```
echo(rand() . "<br />");
```

```
echo(rand() . "<br />");
```

```
echo(rand(10,100)) . "<br />");
```

```
echo(rand(100,300))
```

```
?>
```

خروجی:

```
2776  
37974  
35  
186
```

### ۷-۳-۹ تابع round()

این تابع اعداد اعشاری را به نزدیک‌ترین اعداد صحیح تبدیل می‌کند.  
مثال:

```
<?php  
echo(round(0.60) . "<br />");  
echo(round(0.50) . "<br />");  
echo(round(0.49) . "<br />");  
echo(round(-4.40) . "<br />");  
echo(round(-4.60))  
?>
```

خروجی:

```
1  
1  
0  
-4  
-5
```

### ۴-۹ تعریف توابع جدید

همان‌طور که ملاحظه کردید در زبان PHP توابعی وجود دارند که در روند برنامه‌نویسی به ما کمک می‌کنند، بسیاری از این توابع جزء ساختار اولیه زبان PHP نیستند و به دلیل متن باز بودن به ساختار آن اضافه شده‌اند.

فرض کنید می‌خواهید یک سری عملیات مشخص را به دفعات در برنامه خود استفاده کنید، می‌توانید بدون نیاز به درج کدهای تکراری برای خود یک تابع بسازید که عملیات کدنویسی را برای شما راحت‌تر کند.  
ساختار کلی تعریف تابع جدید به شکل زیر است:

(پارامترهای ورودی) نام تابع function

}

بدنه تابع

{

اگر تابع پارامتر ورودی نداشته باشد می‌توانید پرانتزهای جلوی نام تابع را خالی بگذارید.

برنامه زیر یک تابع ساده با نام hello می‌سازد که عبارت (hello,welcome!) را در خروجی چاپ می‌کند.

```
function hello()
```

```
}
```

```
echo ("hello , welcome!");
```

```
}
```

پس از نوشتن تابع، در برنامه هر جایی که نیاز به تابع داشتید کافی است نام تابع را بنویسید، به این عمل فراخوانی تابع گفته می‌شود.

```
<?php
```

```
function hello()
```

```
{
```

```
echo ("hello , welcome!");
```

```
}
```

```
hello();
```

```
?>
```

همان‌طور که اشاره شد تابع می‌تواند پارامتری یا پارامترهایی را به عنوان ورودی داشته باشد. فرض کنید می‌خواهیم در صفحه مرورگر با استفاده از یک تابع پیغام خوشامدگویی

به کاربر به همراه نام او ظاهر شود، در این صورت نام کاربر به عنوان پارامتر ورودی در اختیار تابع قرار می‌گیرد:

```
<?php
function hello($name)
{
echo ("hello , welcome!". "$name");
}
$myname="mina";
hello($myname);
?>
```

برخی از توابع هستند که عبارت خاصی را چاپ نمی‌کنند و فقط یک مقدار را به عنوان خروجی برمی‌گردانند. برای پیاده‌سازی این گونه توابع از دستور return استفاده می‌شود. ساختار کلی این گونه توابع به صورت زیر است:

```
function تابع (پارامترهای ورودی) نام تابع
{
بدنه تابع
return "متغیری که باید برگشت داده شود";
}
```

می‌خواهیم یک تابع پیاده‌سازی کنیم به طوری که دو عدد را دریافت کند و حاصل ضرب آن‌ها را برگرداند، سپس حاصل را در خارج از تابع چاپ کند:

```
<?php
function calc($n,$m)
{
$p = $n*$m;
return $p;
}
```

```
$n1=10;
```

```
$n2=43;  
$n3 = calc($n1,$n2);  
echo $n3;
```

?>

در کد فوق دو متغیر با نام‌های  $n1$  و  $n2$  تعریف شده است که به ترتیب با اعداد ۱۰ و ۴۳ مقداردهی شده‌اند، سپس تابع  $calc()$  با دو پارامتر ورودی (۴۳ و ۱۰) فراخوانی می‌شود.

درون تابع  $calc()$  حاصلضرب این دو پارامتر ورودی محاسبه شده، توسط دستور  $return$  برگردانده می‌شود.

درون برنامه اصلی مقدار برگردانده شده توسط دستور  $return$  در متغیر  $n3$  قرار داده می‌شود و در سطر آخر برنامه توسط دستور  $echo$  مقدار این متغیر در خروجی چاپ می‌شود.

## واژه‌نامه

Absolute  
Length  
Lowercase  
Uppercase

قدر مطلق  
طول  
حروف کوچک  
حروف بزرگ

## خلاصه مطالب

- به دلیل این که اغلب اطلاعات ورودی کاربران در سایت‌های پویا به صورت متنی وارد می‌شود، PHP باید بتواند پردازش‌های لازم را روی آن‌ها انجام دهد، اغلب این پردازش‌ها توسط توابع از پیش تعریف شده انجام می‌شوند.
- تابع رشته‌ای `strlen` به منظور محاسبه طول رشته‌ای به کار می‌رود که به عنوان پارامتر ورودی به تابع داده می‌شود.
- تابع رشته‌ای `strtoupper` را که به عنوان پارامتر ورودی می‌گیرد، به حروف بزرگ تبدیل می‌کند.
- تابع رشته‌ای `strtolower` را که به عنوان پارامتر ورودی می‌گیرد، به حروف کوچک تبدیل می‌کند.
- تابع `ucfirst()` حرف اول عبارتی را که به عنوان پارامتر ورودی می‌گیرد به حرف بزرگ تبدیل می‌کند.
- تابع `ucword()` حرف اول تمام کلمات عبارت ورودی را به حرف بزرگ تبدیل می‌کند.
- تابع `date()` یکی از توابع پرکاربرد در PHP است، این تابع با استفاده از پارامترهای دریافتی نحوه نمایش تاریخ و زمان را تعیین می‌کند.

## آزمون نظری

- ۱ - نوع داده‌ای که توسط تابع `strlen()` برگردانده می‌شود، چیست؟  
الف - رشته‌ای      ب - `boolean`      ج - عددی      د - منطقی
- ۲ - خروجی تابع `strlen("h-e-l-l-o")` چیست؟  
الف - ۹      ب - ۵      ج - `hello`      د - -
- ۳ - نتیجه تابع `strtoupper("Hello Word!")` چیست؟  
الف - `helloworld`      ب - `HELLOWORD`  
ج - `HELLO WORD`      د - `hello word`
- ۴ - کدام یک از گزینه‌های زیر نمی‌تواند خروجی تابع `date("d")` باشد؟  
الف - ۱۲      ب - ۰۱      ج - ۲۹      د - ۱
- ۵ - خروجی کدام یک از توابع زیر همیشه یک عدد مثبت است؟  
الف - `abs()`      ب - `date()`  
ج - `ucfirst()`      د - `strtoupper()`
- ۶ - به منظور محاسبه قدر مطلق یک عدد از چه تابعی استفاده می‌شود؟  
الف - `date()`      ب - `abs()`  
ج - `absolute()`      د - ب و ج
- ۷ - کدام یک از توابع زیر برای تولید اعداد تصادفی به کار می‌رود؟  
الف - `round()`      ب - `abs()`  
ج - `rand()`      د - `floor()`
- ۸ - کدام یک از مقادیر زیر توسط تابع `fmod(25,4)` برگردانده می‌شود؟  
الف - ۱      ب - ۲      ج - ۳      د - ۶



## آزمون عملی

- ۱ - برنامه‌ای بنویسید که یک رشته را به عنوان ورودی دریافت کند، سپس به تعداد طول رشته، همان رشته را در خروجی نمایش دهد.
- ۲ - برنامه سؤال اول را با تابع پیاده‌سازی کنید، طوری که پارامتر ورودی تابع یک رشته باشد.
- ۳ - تابعی بنویسید که هر رشته را که به عنوان پارامتر دریافت می‌کند به حروف کوچک تبدیل کند، سپس در برنامه با دو پارامتر ورودی مختلف آن را فراخوانی کنید.
- ۴ - زمان و تاریخ جاری سیستم را به‌طور کامل در صفحه مرورگر نشان دهید.

## کار با پایگاه داده

### واحد کار دهم

پس از مطالعه این واحد کار از فراگیر انتظار می‌رود که:

- ۱ - مجوزهای کاربری و عملکرد آنها را بررسی کند.
- ۲ - اصول اتصال به MySQL را بداند.
- ۳ - بتواند از پایگاه‌های داده سرویس‌دهنده فهرست تهیه کند.
- ۴ - اصول مدیریت پایگاه داده‌ها را بداند.
- ۵ - عملکرد دستورات INSERT، REPLACE، UPDATE و DELETE را بداند.
- ۶ - بتواند در پایگاه داده جستجو کند و نتایج را نمایش دهد.
- ۷ - اصول خطیابی و رفع اشکالات ناشی از ارسال Query را بداند.
- ۸ - اصول ارسال فایل‌های چند رسانه‌ای به پایگاه داده را بداند.

زمان(ساعت)

زمان(ساعت)	
عملی	نظری
۱۲	۶

## مقدمه

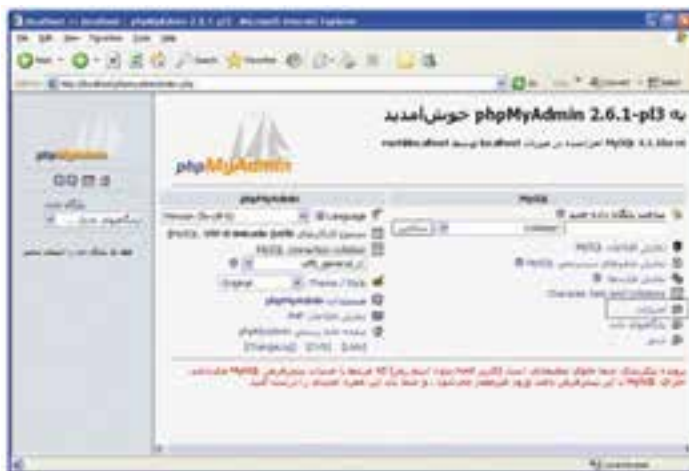
امروزه محافظت از بانک‌های اطلاعاتی در برابر تهاجم هکرها و افرادی که قصد سرقت اطلاعات را دارند یک امر بسیار مهم به شمار می‌رود. از طرفی سیستم مدیریت پایگاه داده MySQL به علت ارزان بودن و سرعت در اجرا، روز به روز کاربران بیشتری پیدا می‌کند و به همان نسبت حجم حملات به سایت‌هایی که با MySQL طراحی شده‌اند نیز بیشتر می‌شود.

## ۱-۱۰ مجوزهای کاربری

هنگامی که MySQL را روی سیستم نصب می‌کنید، به طور پیش فرض به کاربر اصلی (مدیر) یک شناسه کاربری با عنوان root داده می‌شود. این شناسه کاربری به طور پیش فرض کلمه عبور ندارد، لذا ضروری به نظر می‌رسد که مدیر سیستم به عنوان اولین گام نسبت به تنظیم کلمه عبور اقدام کند.

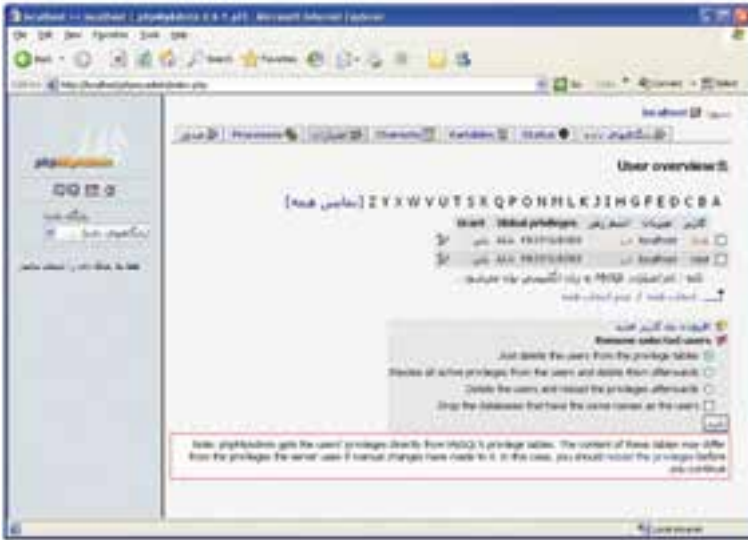
**تذکر:** برای شروع کار بررسی کنید که روی سیستم شما WAMP نصب شده باشد (نصب WAMP در واحد کار پنجم آموزش داده شده است).

روی آیکن نیم‌دایره برنامه WAMP کلیک کنید، سپس گزینه phpMyAdmin را انتخاب کنید. زبان این برنامه را به فارسی تغییر دهید (با روش فارسی کردن WAMP در واحد کار پنجم آشنا شده‌اید).




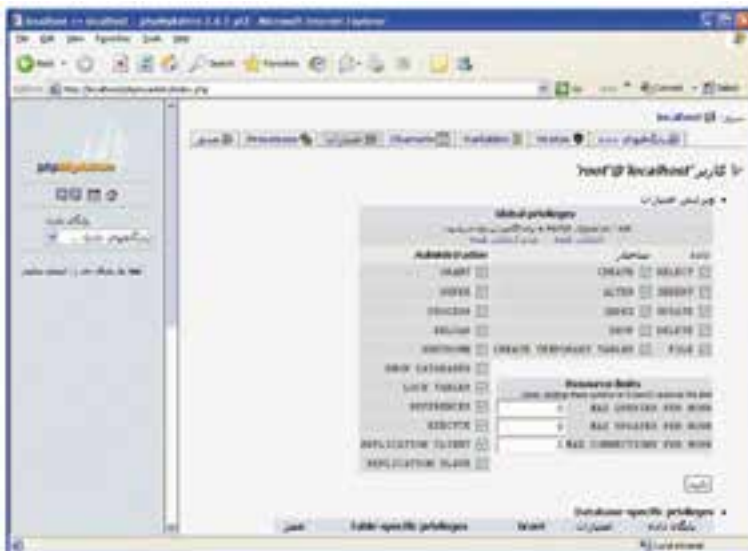
شکل ۱-۱۰

در شکل ۱۰-۱ روی امتیازات کلیک کنید.



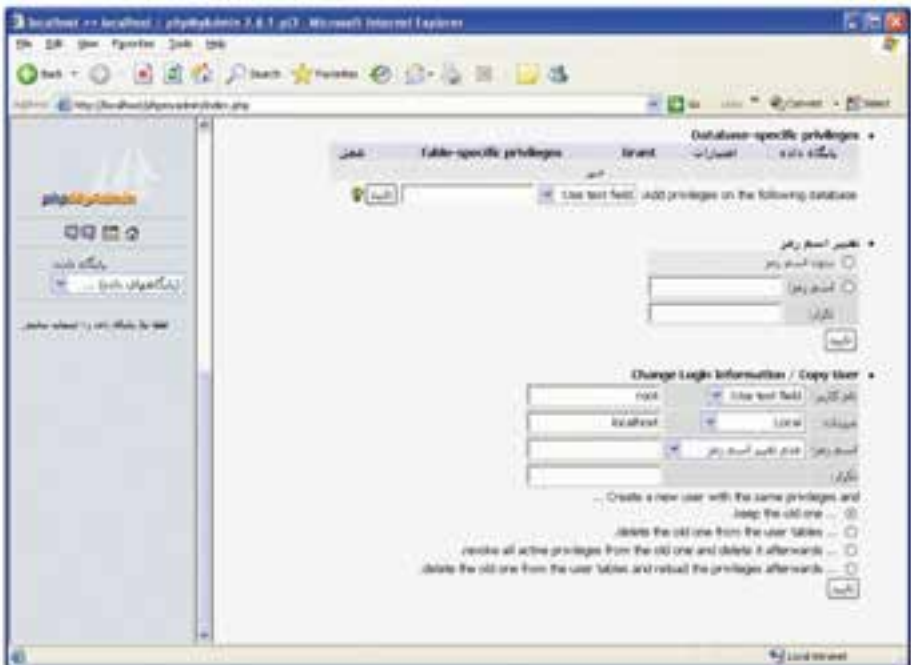
شکل ۱۰-۲

در قسمت User overview لیست کاربران به همراه مشخصات آن‌ها نشان داده شده است. در شکل ۱۰-۲ مشاهده می‌شود که هیچ‌یک از کاربران در حال حاضر (root) اسم رمز ندارند. برای تعیین اسم رمز روی آیکن  در انتهای سطر مربوط به کاربر root کلیک کنید.



شکل ۱۰-۳

در جدول شکل ۳-۱۰ امتیازاتی به کاربر داده شده است، این امتیازات برای اعمالی است که کاربر می تواند روی پایگاه داده و جدول های آن داشته باشد. برای گرفتن مجوز دسترسی از کاربر، کافی است آن را از حالت انتخاب خارج کنید. پس از تعیین امتیازات برای اعمال روی پایگاه داده باید روی دکمه تأیید کلیک کنید. نوار پیمایش شکل ۳-۱۰ را به طرف پایین صفحه بکشید.



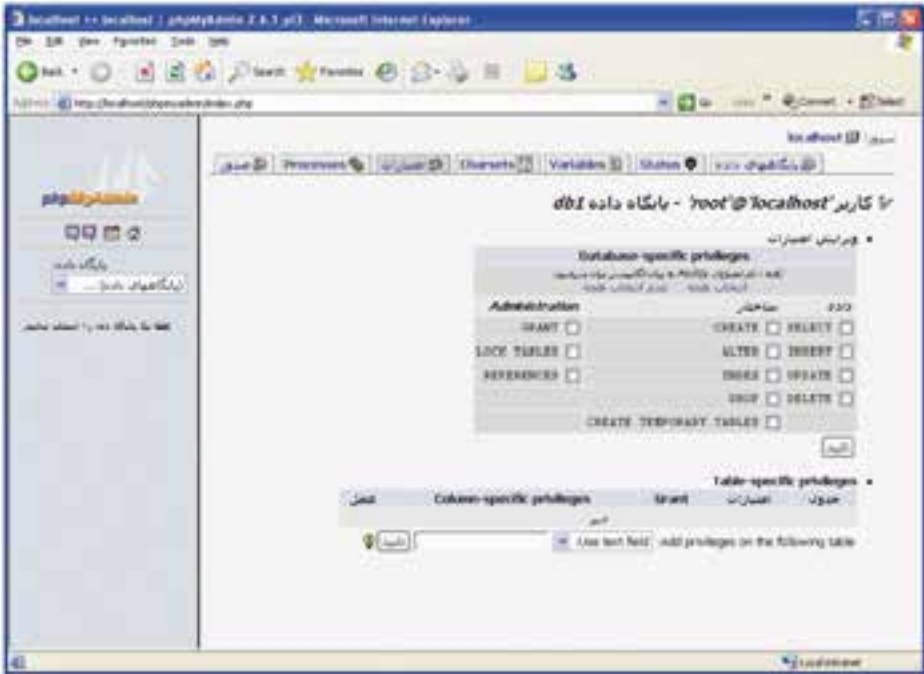
شکل ۴-۱۰

در شکل ۴-۱۰ در لیست بازشوی مقابل گزینه Add privileges on the following data - base می توانید یک پایگاه داده را انتخاب کرده، سپس مجوزهای دسترسی را فقط در آن پایگاه تنظیم کنید.

به عنوان مثال پایگاه داده db1 را از لیست بازشوی مقابل گزینه Add privileges on the following database انتخاب می کنیم، صفحه مرورگر به شکل ۵-۱۰ ظاهر می شود.

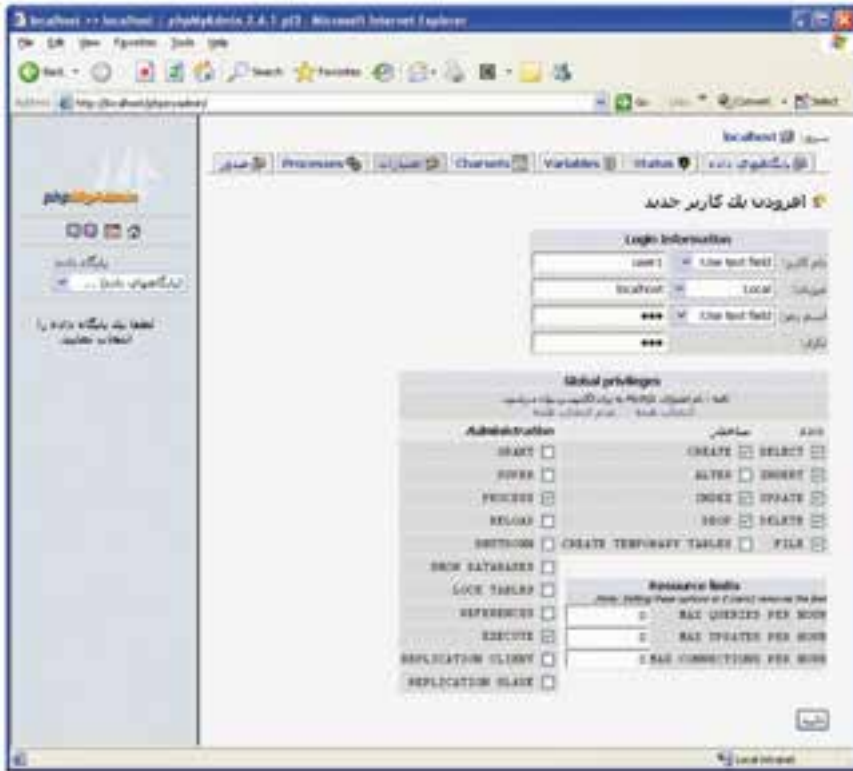
تمام مجوزها به طور پیش فرض غیرفعال هستند، هر کدام از مجوزها را که می خواهید کاربر root داشته باشد، انتخاب کنید، سپس روی دکمه تأیید کلیک کنید. برای تعیین یا تغییر اسم رمز به شکل ۴-۱۰ بازگردید.

برای تعیین اسم رمز در کادرهای متنی مقابل اسم رمز و تکرار، کلمه رمز را وارد کنید و روی دکمه تأیید مربوط به قسمت تغییر اسم رمز کلیک کنید.



شکل ۵-۱۰


یک پیغام در مرورگر ظاهر می‌شود که نشان‌دهنده این است که عمل تغییر یا تعیین کلمه رمز با موفقیت انجام شده است. مجدداً به بخش امتیازات (شکل ۲-۱۰) برگردید. برای تعیین کاربر جدید روی عنوان «افزودن یک کاربر جدید» کلیک کنید. در کادر متنی نام کاربر، یک نام مناسب وارد کنید. در شکل ۶-۱۰، در کادرهای متنی مقابل اسم رمز و تکرار، رمز مربوط به کاربر را وارد کنید. می‌توانید برای کاربر اسم رمز تعیین نکنید و کادر متنی آن را خالی بگذارید. در بخش Global privileges مجوزهای لازم را به کاربر بدهید و سپس روی دکمه تأیید واقع در پایین صفحه کلیک کنید. یک پیغام با عنوان "شما یک کاربر جدید اضافه کردید" نمایان می‌شود که نشان‌دهنده این است که عمل ساخت کاربر جدید با موفقیت به پایان رسیده است.

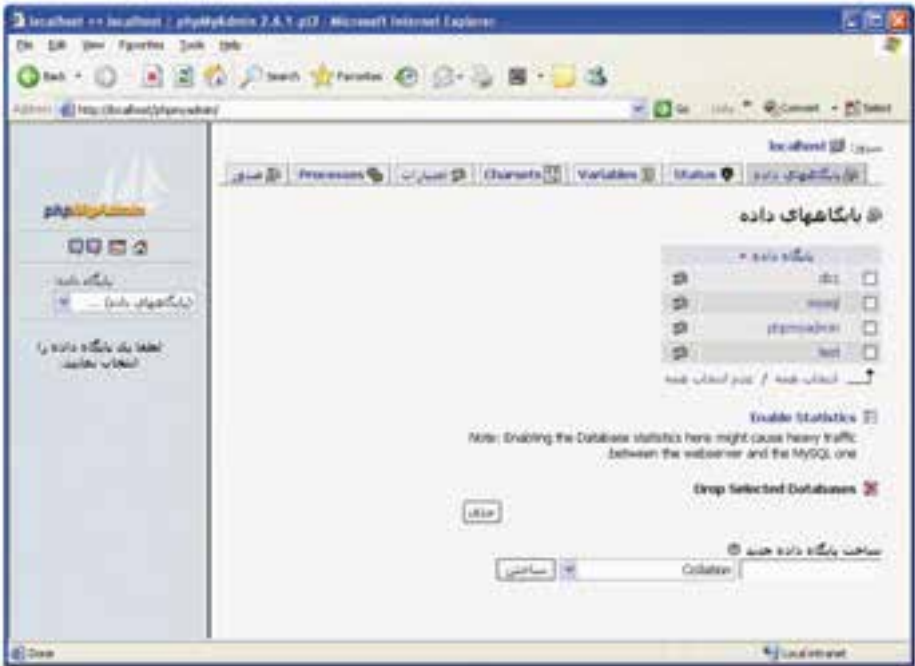


شکل ۱۰-۶

## ۲-۱۰ لیست پایگاه‌های داده

برای مشاهده پایگاه‌های داده‌ای که روی سیستم وجود دارد می‌توانید روی گزینه پایگاه‌های داده کلیک کنید.

در شکل ۷-۱۰ پایگاه‌های داده‌ای که قبلاً ایجاد شده‌اند نشان داده شده است، در این قسمت هم می‌توانید با کلیک روی آیکن  مقابل هر پایگاه داده، مجوزهای دسترسی را به دلخواه تغییر دهید.



شکل ۷-۱۰

### ۱-۲-۱۰ ایجاد پایگاه داده جدید

برای ساختن یک پایگاه داده جدید در بخش پایینی شکل ۷-۱۰، در کادر متنی مربوط به «ساخت پایگاه داده جدید» یک نام دلخواه و مناسب برای پایگاه داده وارد کنید.



شکل ۸-۱۰

در لیست بازشویی که با عنوان Collation مشخص شده است، برای پشتیبانی از زبان فارسی، گزینه utf8\_persian\_ci را انتخاب نمایید. سپس روی دکمه «ساختن» کلیک کنید.

### ۲-۲-۱۰ حذف پایگاه داده

برای حذف یک پایگاه داده در شکل ۷-۱۰ روی کادر علامت کنار عنوان پایگاه داده موردنظر کلیک کنید تا انتخاب شود، سپس روی دکمه حذف کلیک کنید.



یک پیغام ظاهر می شود که "آیا مطمئن هستید:" برای حذف روی دکمه «بلی» کلیک کنید.



شکل ۹-۱۰

### ۳-۱۰ مدیریت پایگاه داده

هنگام تنظیم مجوزها با استفاده از قسمت «امتيازات» با گزینه‌های متعددی مواجه شدید که با برخی از آنها در واحد کارهای قبل آشنا شدید.

مهم‌ترین این مجوزها به بخش مدیریت پایگاه داده اختصاص دارند. مهم‌ترین این مجوزها به همراه شیوه عملکرد عبارتند از:

**INSERT و REPLACE:** این قابلیت مربوط به بخش داده است و می‌تواند داده‌های جدید را در جدول درج کند یا این که داده‌ها را با مقادیر دیگر جایگزین کند. اگر می‌خواهید این قابلیت در اختیار یک کاربر قرار نگیرد و نتواند داده‌ای را درون جدول بریزد یا آن‌ها را تغییر دهد، باید این مجوز را به آن کاربر ندهید.

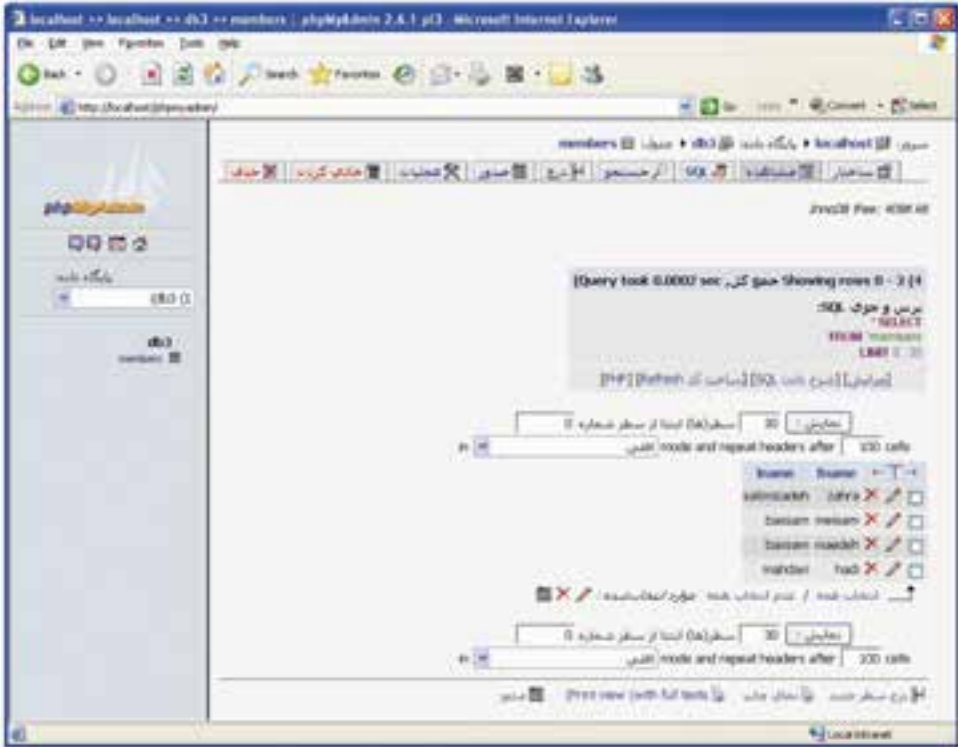
**UPDATE:** این امتیاز مربوط به تغییر و به‌روز رسانی داده‌هاست.

**DELETE:** این امتیاز مربوط به حذف داده‌های جدول‌هاست.

**FILE:** این مجوز اجازه می‌دهد کاربر داده‌ها را از یک فایل بخواند و در جدول‌ها درج کند و همچنین بتواند داده‌های جدول‌ها را به یک فایل مجزا ارسال کند.

برای مشاهده جدول در محیط phpMyAdmin، ابتدا در پنل سمت چپ از لیست بازشوی پایگاه داده نام پایگاه داده را انتخاب کنید، سپس جدول‌های مربوط به آن در پایین همان پنل نشان داده می‌شود.

روی عنوان جدول کلیک کنید، سپس در بخش سمت راست روی گزینه «مشاهده» کلیک کنید.

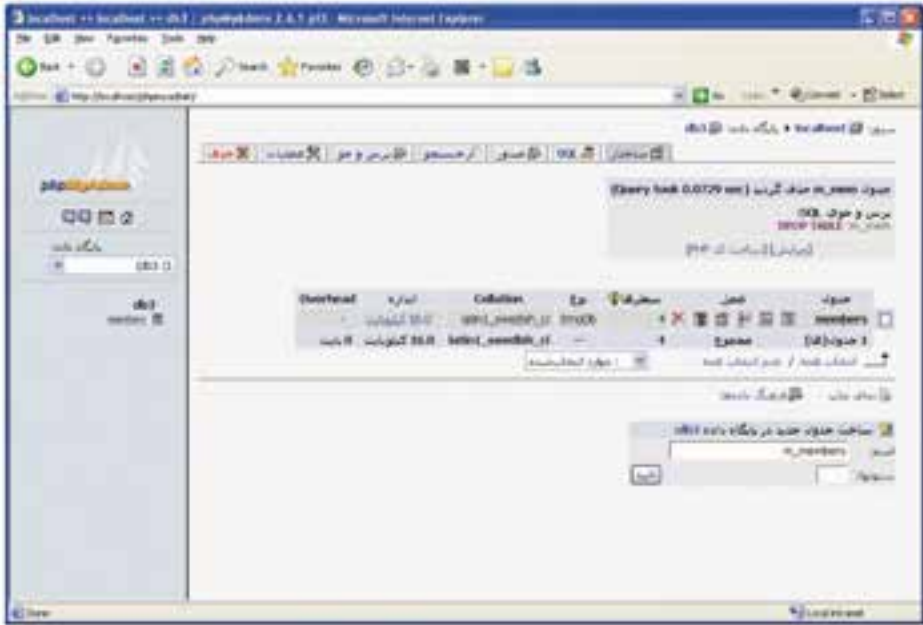


شکل ۱۰-۱۰

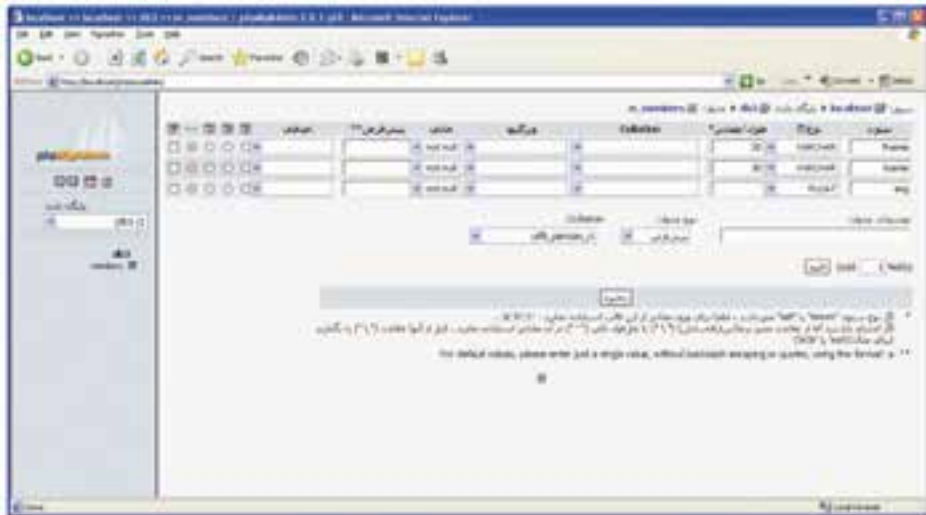
## ۴-۱۰ ساخت جدول phpMyAdmin

برای ساخت جدول در محیط phpMyAdmin ابتدا پایگاه داده مورد نظر را باز کنید. سپس در قسمت ساخت جدول جدید در پایگاه داده مطابق مراحل بعدی برای ساخت جدول جدید اقدام کنید:

در شکل ۱۱-۹ در کادرمتنی اسم یک نام مناسب برای جدول وارد کنید. در کادرمتنی ستون‌ها، تعداد فیلدهای جدول را وارد کنید. روی دکمه تأیید کلیک کنید.



شکل ۱۱-۱۰



شکل ۱۲-۱۰

در شکل ۱۲-۱۰ اطلاعات فیلدهای جدول را وارد کنید.  
 در بخش ستون، نام فیلد و در بخش نوع، نوع داده‌ای را که آن فیلد می‌تواند بپذیرد وارد کنید و سپس در بخش طول/مقادیر حداکثر تعداد حروف یا اعدادی که مجاز است

بپذیرد را وارد کنید.

در بخش Collation برای پشتیبانی از زبان فارسی از لیست بازشو، گزینه utf8\_per-

sian\_ci را انتخاب کنید.

سپس روی دکمه Add کلیک کنید.

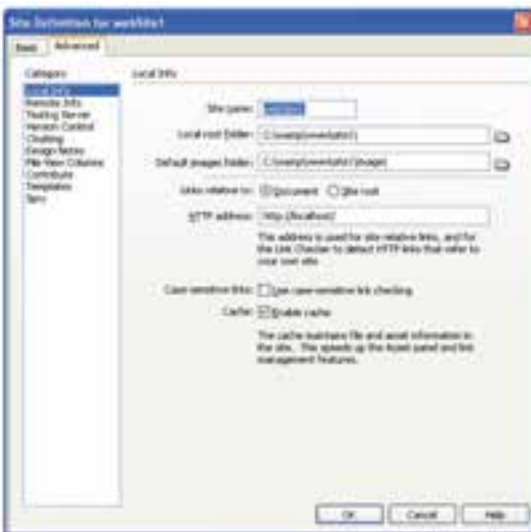
**نکته** اگر در بخش Collation پایین صفحه نوع utf8\_persian\_ci را انتخاب کرده باشید، لازم نیست برای تک تک فیلدها هم تنظیم شود.

اگر MySQL را به طور مجزا روی سیستم خود نصب نکرده باشید، از طریق WAMP نیز می‌توانید به خط فرمان MySQL دسترسی پیدا کنید. برای اجرای خط فرمان MySQL روی آیکن WAMP کلیک کرده و از زیر منوی MySQL گزینه MySQL Console را انتخاب کنید.

## ۵-۱۰ ارتباط با پایگاه داده در Dreamweaver

یک سایت جدید با تنظیماتی که قبلاً یاد گرفته‌اید با سرویس دهنده WAMP در برنامه phpMyAdmin در مسیر WAMP/www/ بسازید.

به دلیل اهمیت و حساسیت تنظیمات این قسمت، مجدداً به بیان آن می‌پردازیم.



پس از اقدام به ساخت وب سایت، کادری مشابه شکل ۱۳-۱۰ ظاهر می‌شود، همان‌طور که می‌دانید انجام تنظیمات گروه Local Info شامل تعیین نام وب سایت، مسیر ذخیره فایل‌های وب سایت، مسیر ذخیره تصاویر و ... است.

شکل ۱۳-۱۰ تنظیمات مسیر وب سایت

**نکته** انجام تنظیمات این مراحل بسیار حساس است و در صورت عدم انجام صحیح این تنظیمات، قادر به برقراری ارتباط با پایگاه داده یا سایر اجزای آن نخواهید بود.

در گروه‌بندی Testing Server مدل سرویس‌دهنده و نحوه دسترسی به آن را بر اساس شکل ۱۴-۱۰، انجام دهید.



شکل ۱۴-۱۰ تنظیمات سرویس‌دهنده مورد استفاده

در گروه Testing Server مطابق شکل ۱۴-۱۰، پس از انتخاب PHP MySQL برای Server model، نحوه دسترسی را از نوع Local/Network قرار دهید.

مسیر کامل ذخیره وب سایت در پوشه سرویس‌دهنده WAMP را مقابل کادر Testing server folder وارد نمایید.

مسیر http://localhost و در ادامه آن، پوشه‌ای را که درون www (در مسیر WAMP) برای ذخیره وب سایت ایجاد کرده‌اید در قسمت URL prefix برای دسترسی به مسیر سرویس‌دهنده وارد کنید.

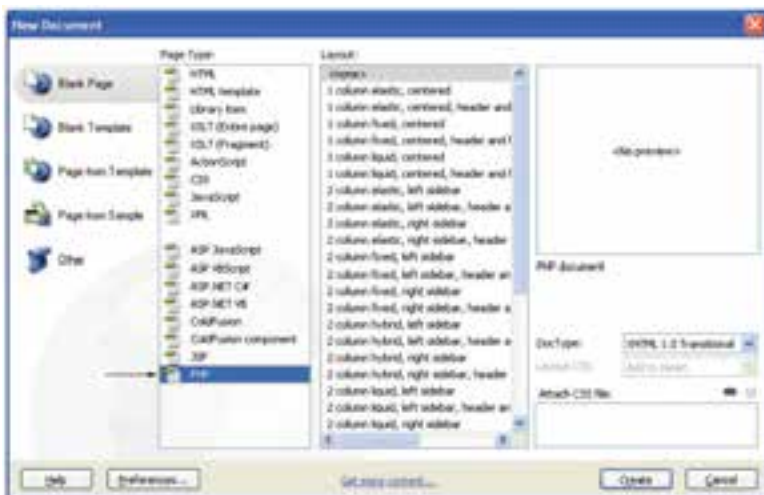
پس از ایجاد وب سایت جدید با تنظیمات فوق، اتصال به پایگاه داده و ارائه سایر قابلیت‌های مربوط به آن توسط پنل DATABASES فراهم می‌شود. اگر هر کدام از

پنل‌های مورد نیاز خود را مشاهده نمی‌کنید، از منوی Window آن را روی محیط کاری نمایان سازید. (شکل ۱۵-۱۰)




شکل ۱۵-۱۰

سپس یک صفحه وب جدید بسازید. هنگام ایجاد یک فایل جدید انواع مختلف صفحه به طور آماده وجود دارد، نوع PHP را انتخاب نمایید (شکل ۱۶-۱۰).



شکل ۱۶-۱۰

این فایل را با نامی متناسب که نشان دهنده درج اطلاعات در جدول باشد، ذخیره نمایید. (به عنوان مثال فایل را با نام insert.php ذخیره می‌کنیم.)  
فرم قبل را طوری تنظیم کنید که با کلیک روی دکمه "ارسال" اطلاعات برای صفحه insert.php ارسال شود.  
برای اتصال به پایگاه داده در پنل DATABASES روی علامت  کلیک کنید و گزینه MySQL Connection را انتخاب کنید.



شکل ۱۷-۱۰ ایجاد اتصال به پایگاه داده

به منظور انتخاب پایگاه داده مورد نظر می‌توانید نام آن را مقابل Database وارد کنید یا این‌که با کلیک روی دکمه Select آن را از لیست پایگاه‌های داده موجود، انتخاب نمایید.



شکل ۱۸-۱۰ انتخاب پایگاه داده

در این مرحله از کار و ایجاد ارتباط با پایگاه داده، ممکن است با پیغام خطاهای مختلفی مواجه شوید، در این صورت قبل از هر کاری با مطالعه پیغام خطا سعی کنید علت را پیدا کنید. این پیغام خطاها می‌تواند دلایل متفاوتی داشته باشد که مهم‌ترین آن‌ها عدم اجرای WAMP یا بخشی از سرویس‌های آن است.



شکل ۱۹-۱۰ پنل DATABASES

پس از انجام اتصالات به پایگاه داده و برقراری ارتباط، پنل DATABASES جداول موجود در پایگاه داده‌ای که اتصال با آن برقرار شده است را به همراه فیلدهای هر کدام نشان می‌دهد.

علاوه بر این در صورتی که ارتباط با پایگاه داده به درستی برقرار شده باشد، در مسیر ذخیره وب سایت، دو پوشه با نام‌های Connections و mmServerScripts ساخته می‌شود که تنظیمات مربوط به این ارتباط را نگهداری می‌کنند.

### ۱-۵-۱۰ ایجاد Recordset


همان‌طور که می‌دانید، صفحات وب قادر نیستند به طور مستقیم به اطلاعات ذخیره شده در بانک‌های اطلاعاتی دسترسی پیدا کنند و برای برقراری این ارتباط از Recordset استفاده می‌نمایند.

یک Recordset مجموعه‌ای از اطلاعات مربوط به رکوردهاست که توسط Query ایجاد می‌شود.

Query ها ابزارهای اصلی کار با پایگاه داده و اجزای آن هستند و اغلب، عملیات مدیریتی روی پایگاه داده نیازمند ایجاد یک Query است.

فرض کنید می‌خواهیم اطلاعات درج شده در جدول persons را از پایگاه داده‌ای که در مبحث قبل ایجاد کرده‌ایم نمایش دهیم.

ابتدا باید برای فیلدهای مورد نظر در جدول، برای نمایش در صفحه وب، یک Recordset ایجاد شود.

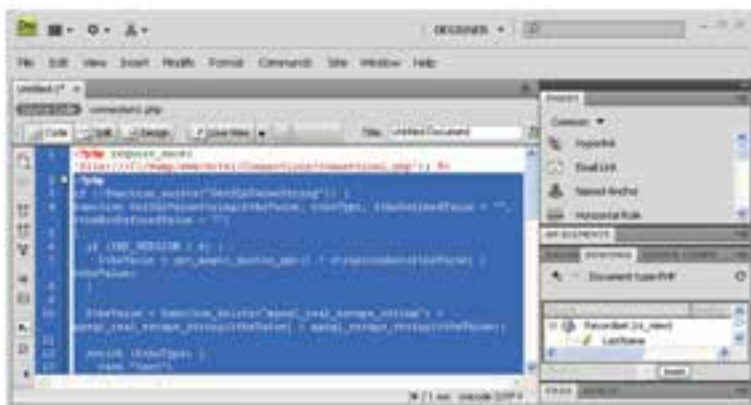
برای ساخت یک Recordset جدید در پنل BINDINGS روی علامت  کلیک کنید و از منوی باز شده، گزینه (Query) Recordset را انتخاب نمایید.





شکل ۲۰-۱۰ ایجاد Recordset

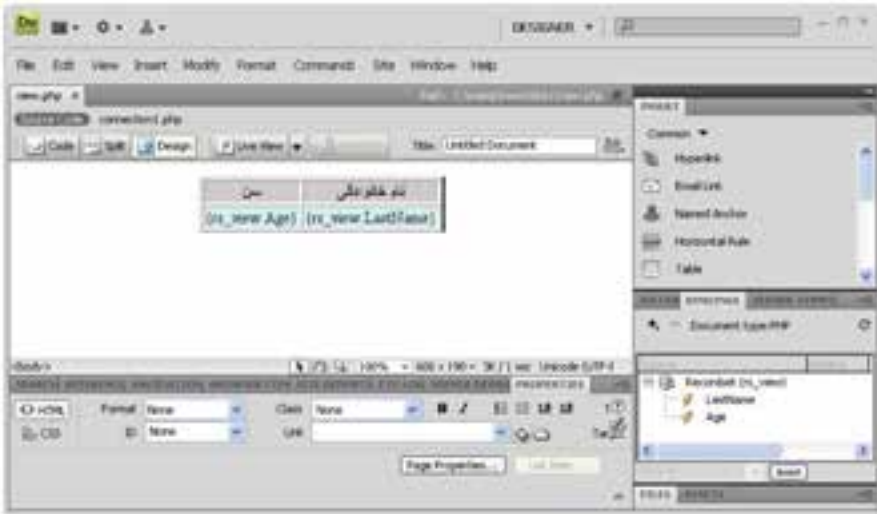
در کادر محاوره شکل ۲۰-۱۰ تنظیمات لازم برای Recordset را اعمال کرده، روی دکمه OK کلیک کنید.  
با باز کردن نمای کد در محیط Dreamweaver، کدهای درج شده برای Recordset و سایر اجزا را مشاهده خواهید کرد.



شکل ۲۱-۱۰ کدهای درج شده به طور خودکار

## ۶-۱۰ مشاهده اطلاعات پایگاه داده

حال، برای استفاده از Recordset به منظور مشاهده رکوردهای موجود در پایگاه داده، در صفحه وب، باید هر کدام از فیلدها را به محل مورد نظر انتساب دهید.  
جدولی به تعداد ستون فیلدهای مورد نظر درج کنید و در هر یک از سطرهای آن، رکورد مربوطه را درگ کنید.



شکل ۱۰-۲۲ انتساب فیلدها به جدول

اگر این صفحه را از طریق localhost در مرورگر مشاهده کنید، خواهید دید که اولین رکورد جدول پایگاه داده را نشان می‌دهد.



شکل ۱۰-۲۳ مشاهده فیلدهای انتخاب شده از اولین رکورد جدول

برای مشاهده تمام رکوردهای جدول، باید از رفتارهای سرویس‌دهنده استفاده شود. این رفتارها از طریق پنل SERVER BEHAVIORS قابل استفاده هستند. سطری که در نرم‌افزار Dreamweaver فیلدهای نام خانوادگی و سن را در آن درگ کردید، انتخاب نمایید و در پنل SERVER BEHAVIORS روی علامت **+** کلیک کنید و در منوی مربوط به آن، گزینه Repeat Region را انتخاب نمایید. چنانچه از انتخاب دقیق این سطر اطمینان ندارید، می‌توانید با مراجعه به نمای کد صفحه، بخشی را که درون برچسب <tr> قرار دارد انتخاب کنید.

```
<tr>
  <td><?php echo $row_rs_view['Age']; ?></td>
  <td><?php echo $row_rs_view['LastName']; ?></td>
</tr>
```

پس از اعمال این ویژگی، مشاهده خواهید کرد که کد فوق، درون یک حلقه while به صورت زیر قرار گرفته است:

```
<?php do { ?>
  <tr>
    <td><?php echo $row_rs_view['Age']; ?></td>
    <td><?php echo $row_rs_view['LastName']; ?></td>
  </tr>
<?php } while ($row_rs_view = mysql_fetch_assoc($rs_view)); ?>
```

**نکته** تابع `mysql_fetch_assoc()` یک سطر از جدولی را که توسط recordset مربوطه، تعیین شده است به صورت آرایه برمی گرداند و به طور خودکار، در هر بار اجرای آن، سطر بعدی جدول برگردانده می شود.

## ۷-۱۰ درج اطلاعات در پایگاه داده

همان طور که می دانید برای درج اطلاعات در پایگاه داده، از فرم استفاده می شود. ابتدا فرم مورد نظر را برای ارتباط با پایگاه داده، طراحی کنید.



شکل ۲۴-۱۰

اگر بخواهید اطلاعات را در همان جدولی از پایگاه داده که از قبل اتصال با آن را برقرار کردید، درج کنید، دیگر نیازی به ساختن مجدد اتصال نیست.

پس از ساخت فرم از پنل SERVER BEHAVIORS روی علامت **+** کلیک کنید و در منوی مربوط به آن، گزینه Insert Record را انتخاب نمایید.



شکل ۲۵-۱۰ درج رکورد

در کادر محاوره شکل ۲۵-۱۰، هر کدام از ستون‌های جدول را به مقدار متناظر در فرم نسبت دهید (با انتخاب هر گزینه در بخش Columns گزینه مناسب برای آن را در مقابل کادر بازسوی Value انتخاب نمایید).  
در گزینه مقابل After inserting, go to، صفحه‌ای را انتخاب کنید که قصد دارید پس از درج اطلاعات در پایگاه داده، کاربر به آن جا هدایت شود.

## ۸-۱۰ حذف رکورد از پایگاه داده

برای حذف اطلاعات از پایگاه داده، ابتدا باید رکوردهای معینی نمایش داده شوند، سپس با کلیک روی دکمه یا پیوند ویژه‌ای، پیغامی به کاربر نشان داده شود که پس از تأیید آن، موفق به حذف رکورد مورد نظرش از پایگاه داده شود.  
ابتدا باید در صفحه وب، فرمی را برای تأیید حذف سطر مورد نظر کاربر طراحی کنید و آن را با نام مشخصی ذخیره کنید (به طور مثال با نام deleteConfirm.php) کدهای مربوط به حذف اطلاعات، باید در همین صفحه قرار گیرند. روی این فرم، یک فیلد مخفی و یک دکمه درج کنید.  
فیلد مخفی یکی از عناصر مورد استفاده در فرم است که برای ارسال یک پارامتر به طور مخفی به کار می‌رود و برای درج آن می‌توانید در نوار Insert از زبانه Form گزینه Hidden Field را انتخاب کنید.

برای نمایش رکوردها می‌توانید مانند بخش قبل، جدولی برای نمایش تمام اطلاعات جدول بانک اطلاعاتی، طراحی کنید و در کنار اطلاعات جدول، یک ستون اضافی برای حذف، درج کرده، آن را به صفحه deleteConfirm.php پیوند دهید و به انتهای آدرس URL آن عبارت زیر را اضافه کنید:

```
?recordID=<?php echo $row_recordsetName['fieldName']; ?>
```

در عبارت فوق، به جای recordsetName نام Recordset و به جای fieldName نام فیلدی را که می‌خواهید حذف رکورد بر اساس آن انجام شود وارد کنید. recordID نیز نام یک متغیر است که به دلخواه می‌توانید آن را تغییر دهید. علامت ؟ در ابتدای عبارت فوق، یک پارامتر را برای URL تعیین می‌کند. با توجه به توضیحات فوق، اگر بخواهید در جدول persons حذف اطلاعات بر اساس فیلد نام خانوادگی انجام شود و در متغیر record\_id ریخته شود، باید کد زیر را درج نمایید:

```
deleteConfirm.php?record_In=<?php echo $row_recordset1['LastName']; ?>
```



شکل ۲۶-۱۰

در Dreamweaver حالت Live Data را برای نمایش اطلاعات جدول، فعال کنید. برای فعال کردن این حالت، در منوی View، گزینه Live Data را در حالت انتخاب شده قرار دهید. در مرحله بعد، باید ابتدا رکورد مورد نظر برای حذف، نمایش داده شود، سپس از کاربر خواسته شود که در صورت اطمینان روی دکمه حذف کلیک کند. بنابراین، در صفحه deleteConfirm.php علاوه بر فیلد مخفی و دکمه، کادرهای متنی برای نمایش

اطلاعات مربوط به رکورد انتخاب شده برای حذف طراحی کرده، سپس یک Recordset جدید ایجاد نمایید.

برای ایجاد Recordset مانند مباحث قبل در پنل BINDINGS روی علامت **+** کلیک کنید و از منوی باز شده، گزینه Recordset(Query) را انتخاب نمایید. اگر کادر مربوط به ایجاد Recordset در حالت Advanced (مانند شکل ۱۰-۲۷) باز شده است، روی دکمه Simple کلیک کنید تا کادر آن مطابق شکل ۱۰-۲۸ باز شود.



شکل ۱۰-۲۷





شکل ۱۰-۲۸

در کادر محاوره شکل ۱۰-۲۸، تنظیمات را مطابق شکل انجام دهید. در این کادر، در کادر بازشوی Filter فیلدی را که حذف بر اساس آن انجام می‌شود و توسط صفحه

قبل ارسال شده است انتخاب نمایید و در مقابل URL Parameter نام متغیری را که هنگام ارسال اطلاعات توسط فایل deleteConfirm.php تعیین کرده بودید وارد کنید (به کوچک و بزرگ بودن حروف دقت کنید).  
به هر کدام از کادرهای متنی فرم، فیلد مربوطه را از Recordset ای که ساخته‌اید، انتساب دهید.

شکل ۲۹-۱۰

برای فیلد مخفی نیز، پس از انتخاب آیکن زرد رنگ، در پنل PROPERTIES با کلیک روی علامت  فیلدی را که برای حذف در نظر گرفته شده بود انتساب دهید.  
در این مرحله، می‌توانید از رفتارهای سرویس دهنده استفاده نمایید.  
در پنل SERVER BEHAVIORS روی علامت  کلیک کنید و در منوی مربوط به آن، گزینه Delete Record را انتخاب نمایید.



شکل ۳۰-۱۰

این کادر محاوره را نیز مطابق شکل ۳۰-۱۰ تنظیم کنید. در مقابل Primary key column فیلدی را که برای حذف انتخاب شده بود تعیین کنید و در کادر باز شوی Primary key value،

گزینه Form Variable را انتخاب و در مقابل آن، عنوان فیلد مخفی را وارد کنید. در کادر متنی After deleting, go to صفحه وبی را وارد کنید که می خواهید پس از حذف اطلاعات رکورد انتخابی، کاربر به آنجا منتقل شود. حال با اجرای فایل اولیه حذف اطلاعات، (در این جا فایل del.php) می توانید با مشاهده بخشی از اطلاعات مربوط به رکوردهای جدول، پس از انتخاب ستون حذف مربوط به آن و تأیید عملیات، رکورد مورد نظر خود را از بانک اطلاعاتی حذف نمایید.

## ۹-۱۰ ویرایش اطلاعات پایگاه داده

برای ویرایش اطلاعات موجود در پایگاه داده، بهتر است ابتدا رکورد مورد نظر برای ویرایش انتخاب شود، سپس در یک فرم اطلاعات موجود نمایش یابد و کاربر بتواند اطلاعات جدید را وارد کرده، پس از کلیک روی دکمه مخصوص، عمل ویرایش اطلاعات در بانک اطلاعاتی انجام شود.

روش انتخاب رکورد به سلیقه طراح وابسته است، در این جا به طور نمونه در کنار دکمه حذف که پیش از این برای حذف رکورد درج کردیم، یک پیوند برای ویرایش اطلاعات رکورد اضافه می نمایم.

برای طراحی عملیات ویرایش، ابتدا فرمی را برای نمایش اطلاعات مربوط به رکورد مورد نظر کاربر طراحی کنید و آن را با نام مشخصی ذخیره کنید (به طور مثال با نام update.php).

همانند مرحله حذف اطلاعات، روی این فرم، یک فیلد مخفی، کادرهای متنی نمایش اطلاعات رکورد و یک دکمه نیز درج نمایید.

برای نمایش رکوردها، می توانید مانند بخش حذف، در جدول نمایش اطلاعات، یک ستون اضافی برای ویرایش نیز درج کنید و آن را به صفحه update.php پیوند دهید و با توجه به توضیحات بخش حذف اطلاعات، اگر بخواهید در جدول persons حذف اطلاعات بر اساس فیلد نام خانوادگی انجام شود، باید به جای آدرس URL آن کد زیر را درج نمایید:

```
update.php?record_in=<?php echo $row_recordset1['LastName']; ?>
```





نام خانوادگی	سن	تعداد رکورد	ویرایش رکورد
Karaband	19	1	ویرایش اطلاعات
Arbana	23	2	ویرایش اطلاعات

شکل ۱۰-۳۱

در Dreamweaver حالت Live Data را برای نمایش اطلاعات جدول، فعال کنید. در مرحله بعد، باید ابتدا اطلاعات رکورد مورد نظر برای ویرایش، نمایش داده شود. بنابراین، در صفحه update.php علاوه بر فیلد مخفی و دکمه، کادرهای متنی برای نمایش اطلاعات مربوط به رکورد انتخاب شده برای ویرایش را طراحی کرده، سپس یک Recordset جدید ایجاد نمایید. مانند مرحله حذف، در حالت Simple اطلاعات Recordset را مطابق شکل بعد وارد نمایید.





نام خانوادگی	سن	تعداد رکورد	ویرایش رکورد
Karaband	19	1	ویرایش اطلاعات
Arbana	23	2	ویرایش اطلاعات

شکل ۱۰-۳۲

در صفحه update.php به هر کدام از کادرهای متنی فرم، فیلد مربوطه را از Recordset ای که ساخته‌اید، انتساب دهید و بخش Filter را مطابق توضیحات ارائه شده در بخش حذف اطلاعات، تنظیم کنید.



شکل ۱۰-۳۳

برای فیلد مخفی نیز، پس از انتخاب آیکن زرد رنگ، در پنل PROPERTIES با کلیک روی علامت  فیلدی را که برای ویرایش در نظر گرفته شده بود انتساب دهید. در این مرحله از کار، می‌توانید از رفتارهای سرویس دهنده استفاده نمایید. در پنل SERVER BEHAVIORS روی علامت  کلیک کنید و در منوی مربوط به آن، گزینه Update Record را انتخاب نمایید.



شکل ۱۰-۳۴

این کادر را نیز مطابق شکل ۱۰-۳۴ تنظیم کنید. در کادر متنی After updating, go to صفحه وبی را وارد کنید که می‌خواهید، پس از ویرایش اطلاعات رکورد، کاربر به آنجا منتقل شود.

## مطالعه آزاد

در واحد کار دوم این کتاب با انواع دستورات SQL آشنا شدید. در واقع تمام زبان‌های Server Side از این دستورات به عنوان دستورات پایه‌ای، برای ارتباط با پایگاه داده استفاده می‌کنند. به دلیل اینکه کار با این کدها، اندکی پیچیده‌تر از روش کار با نرم‌افزار Dreamweaver است، در این قسمت به عنوان مطالعه آزاد به بیان روش ایجاد کدهای مربوط به کار با پایگاه داده می‌پردازیم. مزیت این روش در این است که اگر نرم‌افزار Dreamweaver در دسترس نبود نیز می‌توانید به راحتی، حتی در ویرایش‌گرهای متنی ساده، تمام اعمال مربوط به کدنویسی را انجام دهید.

## ۱۰-۱۰ ایجاد ارتباط با سیستم مدیریت پایگاه داده MySQL

همانطور که پیش از این اشاره شد، مهم‌ترین ویژگی وب سایت‌های پویا، توانایی ارتباط با پایگاه داده و کار با آن است. با روش ساخت پایگاه داده و جداول آن در WAMP آشنا شدید. برخی از برنامه‌نویسان و طراحان وب سایت‌های پویا، تمام اعمال مدیریتی از جمله ساخت پایگاه داده و جداول مورد نیاز را توسط اسکریپت‌های برنامه انجام می‌دهند. به همین دلیل باید با تمام دستورات مربوط به این اعمال آشنا باشید. برای ایجاد و مدیریت اجزای بانک اطلاعاتی در برنامه، از دستورات SQL استفاده می‌شود که در درس پایگاه داده با آن‌ها به طور مفصل آشنا شده‌اید، به همین دلیل در این فصل به معرفی مختصر دستورات مورد نیاز می‌پردازیم. سایر دستورات SQL نیز در MySQL قابل دسترسی هستند و در صورت نیاز می‌توانید از آن‌ها استفاده کنید. برای ارتباط با MySQL در صفحات وب، باید قبل از هر چیزی، اتصال با پایگاه داده ایجاد شود.

اتصال به پایگاه داده در PHP توسط تابع `mysql_connect()` انجام می‌شود. ساختار این تابع به صورت زیر است:

```
mysql_connect(servername,username,password);
```

تابع `mysql_connect` سه پارامتر دارد که همراه با توضیحات در جدول ۱-۱۰ بیان شده‌اند.

جدول ۱-۱۰

پارامتر	توضیحات
servername	این پارامتر اختیاری است و نام سرویس دهنده را مشخص می‌کند، چنانچه این پارامتر مقداردهی نشود به طور پیش فرض مقدار "localhost:3306" برای آن تعیین می‌شود.
username	این پارامتر نیز اختیاری است و شناسه کاربری سرویس دهنده را تعیین می‌کند. در سرویس دهنده WAMP، این شناسه، همان کلمه کاربری است که برای کار با WAMP، تنظیم شده است. اگر شناسه کاربری را تنظیم نکرده باشید، مقدار آن را root در نظر بگیرید و اگر این پارامتر را مقداردهی نکنید، کاربری که پایگاه داده را ایجاد کرده است به طور پیش فرض تعیین می‌شود.
password	مقدار این پارامتر نیز اختیاری است و رمز ورود شناسه کاربری را دریافت می‌کند، چنانچه مقداردهی نشود، یک رشته تهی را به طور پیش فرض در نظر می‌گیرد.

تابع `mysql_close()` نیز برای قطع ارتباط با پایگاه داده به کار می‌رود. بهتر است پس از خاتمه کار با پایگاه داده، توسط این تابع در برنامه اتصال به آن قطع شود.

**مثال:** در مثال زیر اتصال به پایگاه داده در متغیر `$co` ذخیره می‌شود، سپس از این متغیر در برنامه استفاده می‌شود. تابع `die` در صورتی اجرا می‌شود که اتصال به پایگاه داده انجام نشده باشد. پس از خاتمه عملیات نیز ارتباط با پایگاه داده قطع می‌شود.

```
<?php
$con = mysql_connect("localhost","root","");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

// some code
mysql_close($con);
?>
```

در مثال فوق تابع `mysql_error()` خطایی را که منجر به عدم اتصال به پایگاه داده شده است نمایش می دهد.

**نکته**  تابع `die()` پس از نمایش یک پیغام از اسکرپت جاری خارج می شود.

## ۱۰-۱۱ ایجاد یک پایگاه داده و اجزای آن

همانطور که می دانید در یک بانک اطلاعاتی یک یا چند جدول وجود دارد و هر جدول نیز از رکوردهای متعددی تشکیل شده است. هر رکورد هم شامل چند فیلد مرتبط با هم و مربوط به یک موجودیت خاص است. در این بخش بانحوه ایجاد پایگاه داده و هر یک از اجزای آن توسط PHP آشنا خواهید شد.

### ۱۰-۱۱-۱ ایجاد یک پایگاه داده

برای ایجاد پایگاه داده از دستور `CREATE DATABASE` استفاده می شود. ساختار کلی این دستور به شکل زیر است:

نام پایگاه داده `CREATE DATABASE`

برای نوشتن کد PHP به منظور اجرا و ایجاد پایگاه داده، تابع `mysql_query()` به کار می رود. این تابع یک `query` برای ایجاد پایگاه داده به MySQL که اتصال به آن از قبل برقرار شده است ارسال می کند.

**مثال:** مثال زیر پس از اتصال به MySQL، یک پایگاه داده با نام "my\_db" می سازد:

```
<?php
$con = mysql_connect("localhost","root","");
if (!$con)
{
die('Could not connect: ' . mysql_error());
}
```

```
if (mysql_query("CREATE DATABASE my_db",$con))
{
echo "Database created";
}
else
{
echo "Error creating database: " . mysql_error();
}

mysql_close($con);
?>
```

**نکته** در مثال فوق، اتصال به سیستم مدیریت پایگاه داده‌ای برقرار شده است که کلمه کاربری آن root و رمز ورودی برای آن تعیین نشده است، چنانچه روی سیستم شما تنظیماتی غیر از این انجام شده، این دو پارامتر را با توجه به آن تنظیمات وارد کنید.

## ۲-۱۱-۱۰ ایجاد یک جدول در پایگاه داده

ایجاد جدول در پایگاه داده‌ای که از قبل ایجاد شده است، توسط دستور CREATE TABLE انجام می‌شود.

این دستور نیز توسط تابع mysql\_query() اجرا می‌شود.  
ساختار کلی دستور CREATE TABLE به این شکل است:

```
CREATE TABLE نام جدول
(
    نام فیلد(ستون) اول , نوع داده آن
    نام فیلد(ستون) دوم , نوع داده آن
    نام فیلد(ستون) سوم , نوع داده آن
    ....
)
```

هنگام تعریف هر فیلد باید نوع داده آن مشخص شود. مهم‌ترین انواع داده MySQL عبارتند از:

**Int**: برای تعریف اعداد صحیح بدون علامت به کار می‌رود.

**VARCHAR(size)**: برای تعریف رشته‌ای با طول مشخص به کار می‌رود.

**TEXT**: متغیرهای رشته‌ای با حداکثر تعداد ۶۵۵۳۵ کاراکتر را می‌توان توسط این نوع

داده تعریف کرد و لازم نیست طول متغیر مربوط به آن مشخص شود.

انواع داده‌ای که در MySQL قابل استفاده هستند در پیوست انتهای کتاب درج شده‌اند.

**مثال:** در مثال زیر یک جدول با نام "Persons" که دارای سه فیلد با عناوین "FirstName" "LastName" ، و "Age" است، ساخته می‌شود:

```
<?php
$con = mysql_connect("localhost","root","");
if (!$con)
{
die('Could not connect: ' . mysql_error());
}

// Create database
if (mysql_query("CREATE DATABASE my_db",$con))
{
echo "Database created";
}
else
{
echo "Error creating database: " . mysql_error();
}

// Create table
mysql_select_db("my_db", $con);
```

```
$sql = "CREATE TABLE Persons  
(  
  FirstName varchar(15),  
  LastName varchar(15),  
  Age int  
);"  
  
// Execute query  
mysql_query($sql,$con);  
  
mysql_close($con);  
?>
```

### ۳-۱۱-۱۰ فیلد کلیدی و خاصیت افزایش خودکار

هر جدول می‌تواند یک فیلد کلیدی داشته باشد که برای اطمینان از تکراری بودن در جدول استفاده می‌شود. دستور PRIMARY KEY برای تعریف فیلد کلیدی هنگام ساخت جدول به کار می‌رود.

علاوه بر این، فیلد کلیدی نمی‌تواند بدون مقدار باشد. جلوگیری از بدون مقدار بودن یک فیلد توسط دستور NOT NULL صورت می‌گیرد.

در بسیاری از جدول‌ها به عنوان فیلد کلیدی شناسه‌ای غیر از مشخصه‌های اصلی موجودیت، انتخاب می‌شود. در اینگونه مواقع معمولاً لازم است خاصیت افزایش خودکار این فیلد نیز انجام شود. برای فعال کردن این خاصیت می‌توان از دستور AUTO\_INCREMENT استفاده نمود.

**مثال:** در این مثال دستورات ساخت جدول مثال قبل طوری بازنویسی شده است که علاوه بر تعریف کلید، خاصیت افزایش خودکار و خاصیت جلوگیری از بی مقدار بودن آن نیز بررسی شود:

```
$sql = "CREATE TABLE Persons  
(  
  personID int NOT NULL AUTO_INCREMENT,
```



```
PRIMARY KEY(personID),  
FirstName varchar(15),  
LastName varchar(15),  
Age int  
)";
```

```
mysql_query($sql,$con);
```

پس از اجرای موفقیت آمیز هر کدام از کدهای مربوط به پایگاه داده در مرورگر، با مراجعه به محیط سرویس دهنده WAMP، پایگاه داده، جدول و فیلدهای مربوط به آن را مشاهده خواهید کرد که توسط کدهای PHP ساخته شده‌اند.



تمرین:

بانک اطلاعاتی مربوط به شرکت تجاری را که در ابتدای این فصل معرفی شد به همراه جداول و فیلدهای هر کدام با استفاده از دستورات PHP بسازید.

## ۴-۱۱-۱۰ درج اطلاعات در جدول

با استفاده از دستور INSERT INTO می‌توان اطلاعات را در یک جدول درج نمود. این دستور به یکی از دو شکل کلی زیر به کار می‌رود:

شکل اول:

```
INSERT INTO table_name  
VALUES (value1, value2, value3,...)
```

شکل دوم:

```
INSERT INTO table_name (column1, column2, column3,...)  
VALUES (value1, value2, value3,...)
```

مثال: این مثال اطلاعات مربوط به دو نفر را در جدولی که از قبل با نام persons

طراحی شده است، درج می‌کند:

```
<?php
$con = mysql_connect("localhost","root","");
if (!$con)
{
die('Could not connect: ' . mysql_error());
}

mysql_select_db("my_db", $con);

mysql_query("INSERT INTO Persons (FirstName, LastName, Age)
VALUES ('Mahdi', 'Razavi', '15')");

mysql_query("INSERT INTO Persons (FirstName, LastName, Age)
VALUES ('Hadi', 'Salimi', '13')");

mysql_close($con);
?>
```

## ۵-۱۱-۱۰ درج اطلاعات در جدول از طریق فرم

همانطور که می‌دانید، فرم‌ها به منظور تعامل با کاربران در صفحات وب سایت، استفاده می‌شوند. از فرم‌ها می‌توان برای دریافت اطلاعات از کاربر و درج در پایگاه داده، استفاده نمود.

در مثال زیر یک فرم با سه فیلد برای ارتباط با جدول persons طراحی شده است، با دریافت اطلاعات از کاربر و ارجاع به صفحه insert.php، عمل درج در پایگاه داده، انجام می‌شود.

مثال: فرم دریافت اطلاعات:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html>
<head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
</head>
<body>

<form action="insert.php" method="post">
Firstname: <input type="text" name="firstname" />
Lastname: <input type="text" name="lastname" />
Age: <input type="text" name="age" />
<input type="submit" />
</form>

</body>
</html>
```

فایل insert.php که اطلاعات دریافت شده از فرم را در جدول درج می کند:

```
<?php
$con = mysql_connect("localhost","root","");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

mysql_select_db("my_db", $con);

$sql="INSERT INTO Persons (FirstName, LastName, Age)
VALUES
($_POST[firstname],$_POST[lastname],$_POST[age])";

if (!mysql_query($sql,$con))
{
```

```
die('Error: ' . mysql_error());  
}  
echo "1 record added";  
  
mysql_close($con)  
?>
```

**نکته** برای درج اطلاعات فارسی در جدول، حتماً باید از طریق فرم، اقدام به ورود فیلدها نمایید و درج اطلاعات فارسی به طور مستقیم در جدول موجب می شود هنگام خواندن آن‌ها، کدهای نامشخصی نمایش داده شود.

## ۱۰-۱۲ انتخاب و نمایش اطلاعات جدول

پس از ایجاد جدول و وارد کردن اطلاعات در آن می‌توانید با استفاده از دستور SE-LECT آن‌ها را انتخاب نمایید. ساختار کلی این دستور به صورت زیر است:

نام فیلد یا فیلدها SELECT  
نام جدول FROM

مثال: این مثال تمام اطلاعات جدول persons را نمایش می‌دهد.

```
<?php  
$con = mysql_connect("localhost","root","");  
if (!$con)  
{  
die('Could not connect: ' . mysql_error());  
}  
  
mysql_select_db("my_db", $con);  
$result = mysql_query("SELECT * FROM Persons");
```

```
while($row = mysql_fetch_array($result))  
{  
echo $row['FirstName'] . " " . $row['LastName'];  
echo "<br />";  
}  
  
mysql_close($con);  
?>
```



شکل ۳۵-۱۰ نتیجه اجرای اسکریپت

همانطور که در شکل ۳۵-۱۰ مشاهده می کنید، تمام اطلاعات جدول persons نمایش داده شده است.

در این مثال تابع `mysql_query()` با اجرای کد SQL تمام اطلاعات جدول را در متغیر `$result` می ریزد، سپس تابع `mysql_fetch_array()` اولین سطر از مجموعه رکوردها را که به صورت آرایه در نظر گرفته می شوند، درون متغیر `$row` می ریزد، به همین دلیل `$row` هم یک آرایه خواهد بود. با هر بار اجرای حلقه و فراخوانی تابع `mysql_fetch_array()` سطر بعدی فراخوانی می شود.

## ۱-۱۲-۱۰ نمایش اطلاعات پایگاه داده در جدول

با توجه به این که یکی از کاربردهای جدولها در HTML، سازماندهی اطلاعات و نحوه نمایش آنهاست، بنابراین بهتر است اطلاعات فراخوانی شده از جدولهای بانک اطلاعاتی را در قالب جدول نمایش دهیم.

مثال قبل را طوری بازنویسی می کنیم که اطلاعات را در جدول نشان دهد:

مثال:

```
<?php
```

```
$con = mysql_connect("localhost","root","");  
if (!$con)  
{  
die('Could not connect: ' . mysql_error());  
}  
  
mysql_select_db("my_db", $con);  
  
$result = mysql_query("SELECT * FROM Persons");  
  
echo "<table border='1'>  
<tr>  
<th>Firstname</th>  
<th>Lastname</th>  
</tr>";  
  
while($row = mysql_fetch_array($result))  
{  
echo "<tr>";  
echo "<td>" . $row['FirstName'] . "</td>";  
echo "<td>" . $row['LastName'] . "</td>";  
echo "</tr>";  
}  
echo "</table>";  
  
mysql_close($con);  
?>
```



شکل ۱۰-۳۶ نمایش اطلاعات در جدول



اسکرپتی بنویسید که اطلاعات جدول را به صورت مرتب شده (صعودی) بر اساس "نام خانوادگی" نمایش دهد. (راهنمایی: با استفاده از دستور مرتب‌سازی در SQL)

## ۱۰-۱۳ به روزسانی اطلاعات جدول

یکی از مهم‌ترین عملیاتی که در پایگاه‌های داده روی اطلاعاتی که از قبل وارد شده‌اند، انجام می‌شود امکان ویرایش آن‌هاست. به منظور تغییر اطلاعات موجود در جدول از دستور UPDATE استفاده می‌شود.

ساختار کلی این دستور به شکل زیر است:

نام جدول UPDATE

...،فیلد اول=مقدار، فیلد دوم=مقدار SET

شرط WHERE

مثال: جدول persons مثال‌های قبل را در نظر بگیرید، فرض کنید فیلد مربوط به

Age برای فردی به نام Mahdi Razavi اشتباه درج شده باشد و بخواهیم مقدار آن را به ۱۶ تغییر دهیم.

جدول ۱۰-۲

FirstName	LastName	Age
Mahdi	Razavi	15
Hadi	Salimi	13

در این صورت کد اسکرپت مورد نظر به شکل زیر نوشته می‌شود:

```
<?php
$con = mysql_connect("localhost","root","");
if (!$con)
{
die('Could not connect: ' . mysql_error());
}
```

```
mysql_select_db("my_db", $con);
```

```
mysql_query("UPDATE Persons SET سن = '16'  
WHERE نام = 'Mahdi' AND نام خانوادگی = 'Razavi'");
```

```
mysql_close($con);
```

```
?>
```

جدول ۳-۱۰ به روزرسانی جدول persons

FirstName	LastName	Age
Mahdi	Razavi	<u>16</u>
Hadi	Salimi	13

## ۱۴-۱۰ حذف رکورد

دستور DELETE FROM برای حذف رکورد از جدول به کار می‌رود.  
ساختار کلی این دستور به شکل زیر است:

نام جدول DELETE FROM

شرط WHERE

توسط دستور شرطی WHERE رکورد یا رکوردهایی که باید حذف شوند، تعیین می‌شود.

مثال: فرض کنید در جدول persons بخواهیم رکورد فردی را که نام خانوادگی او

Razavi است حذف کنیم، در این صورت اسکریپت اجرای رکورد مورد نظر به صورت

زیر خواهد بود:

```
<?php  
$con = mysql_connect("localhost","root","");  
if (!$con)  
{  
die('Could not connect: ' . mysql_error());  
}
```



```
mysql_select_db("my_db", $con);
```

```
mysql_query("DELETE FROM Persons WHERE نام خانوادگی='Razavi'");
```

```
mysql_close($con);
```

```
?>
```

پس از اجرای کد فوق، جدول persons به شکل زیر خواهد بود:

جدول ۴-۱۰

FirstName	LastName	Age
Hadi	Salimi	13

## واژه نامه

Collation	تطبیق
Privileges	امتيازات
Recordset	مجموعه رکوردها
Region	ناحیه
Repeat	تکرار
Root	ریشه

## خلاصه مطالب

- کلمه کاربری که به طور پیش فرض برای کار روی بانک اطلاعاتی MySQL ساخته می شود با عنوان root است. این کلمه کاربری در ابتدا معمولاً بدون اسم رمز ساخته می شود و برای تأمین امنیت در برابر نفوذ هکرها و افراد مزاحم باید هرچه سریع تر نسبت به تعیین اسم رمز برای آن اقدام نمود.
- توسط برنامه WAMP می توان در یک محیط گرافیکی کاربران جدید تعریف کرده و حق دسترسی را برای هر کدام از آنها به دلخواه محدود کرد.
- هنگام ایجاد یک پایگاه داده جدید در WAMP برای پشتیبانی از زبان فارسی نوع Collation را برای آن به utf8\_persian\_ci تغییر دهید.
- اصلی ترین اعمال مدیریتی در پایگاه داده توسط دستورات INSERT، REPLACE، UPDATE و DELETE صورت می گیرد، مدیر سیستم می تواند هر کدام از این اعمال را به کاربران بدهد یا از آنها بگیرد.
- انجام عملیات مدیریتی و درج و حذف و ویرایش داده ها از طریق Query ها صورت می گیرد و هر Query شامل یک یا چند دستور SQL است.
- هر Query به صورت یک Recordset ایجاد می شود.

## آزمون نظری

- ۱ - عنوان کلمه کاربری مدیر سیستم در MySQL به طور پیش فرض چیست؟  
الف - Admin      ب - Administrator      ج - user1      د - root
- ۲ - برای این که در یک جدول بانک اطلاعاتی بتوانیم از اطلاعات فارسی استفاده کنیم باید کدام قسمت را تغییر دهیم؟  
الف - Language      ب - Collation      ج - utf8      د - Persian
- ۳ - برای تنظیم اطلاعات فارسی برای جدول باید آن را از چه نوعی انتخاب کنیم؟  
الف - Persian      ب - utf8-collation-ci      ج - utf8\_persian\_ci      د - utf8
- ۴ - برای ورود به پایگاه داده در خط فرمان از چه دستوری استفاده می شود؟  
الف - use      ب - insert      ج - Enter      د - create
- ۵ - انجام عملیات مدیریتی روی پایگاه داده، از قبیل درج، حذف و ویرایش از طریق کدام گزینه انجام می شود؟  
الف - رکوردها      ب - جدولها      ج - Query      د - Database
- ۶ - هر Query به صورت یک ..... ایجاد می شود.  
الف - پرس و جو      ب - Recordset      ج - جدول      د - Database
- ۷ - برای این که تمامی رکوردها در جدول نشان داده شوند، باید سطرهای جدول از کدام رفتار سرویس دهنده استفاده کند؟  
الف - Recordset      ب - Binding      ج - Repeat Region      د - Server\_Region

## آزمون عملی

- ۱ - در محیط WAMP یک کاربر جدید با نام `user_Guest` بسازید.
- ۲ - از کاربر `user_Guest` که در سؤال اول ساخته‌اید، امکان درج و حذف داده‌ها را بگیرید.
- ۳ - یک جدول شامل فیلدهای نام، نام خانوادگی و شماره شناسنامه توسط خط فرمان MySQL ایجاد کنید.
- ۴ - یک فرم در Dreamweaver طراحی کنید که کاربر بتواند از طریق صفحه وب اطلاعات خود را در جدول سؤال دوم درج کند.
- ۵ - یک فرم در Dreamweaver طراحی کنید که کاربر بتواند در یک صفحه وب لیست افراد موجود در جدول را ببیند.



## به کارگیری کوکی ها و Session ها

### واحد کار یازدهم


پس از مطالعه این واحد کار از فراگیر انتظار می رود که:

- ۱ - مفهوم و کاربرد کوکی ها را بداند.
- ۲ - با اصول تنظیم کوکی ها آشنا باشد.
- ۳ - با اصول تعیین تاریخ انقضا برای کوکی ها آشنا باشد.
- ۴ - اصول استفاده از کوکی ها برای اعتبارسنجی را بداند.
- ۵ - مفهوم و کاربرد Session ها را بشناسد.
- ۶ - متغیرهای Session را بشناسد.
- ۷ - با اصول فعال سازی شماره دسترسی برای کاربر آشنا باشد.
- ۸ - با اصول نگهداری تنظیمات شخصی کاربر آشنا باشد.

زمان (ساعت)	
عملی	نظری
۱۰	۶

## مقدمه

کوکی‌ها (Cookie) فایل‌های کوچک متنی هستند که از طرف سرویس‌دهنده‌های اینترنت و توسط مرورگر روی سیستم کاربر ایجاد می‌شوند. کوکی‌ها می‌توانند در موارد مختلفی به سرویس‌دهنده وب و سرویس‌دهنده برنامه کاربردی کمک کنند. به عنوان مثال می‌توانند به سرویس‌دهنده وب کمک کنند تا کاربرانی را که قبلاً به سایت آن‌ها مراجعه کرده‌اند بشناسد، یا این‌که به یادآوری رمز عبور کاربر در مراجعات بعدی کمک کنند. یک کاربر می‌تواند سیستم خود را طوری تنظیم کند که سرویس‌دهنده اجازه تولید کوکی را نداشته باشد و یا این‌که کاربر را هنگام تولید کوکی مطلع کند. کوکی‌ها می‌توانند اطلاعات مهمی را درون خود نگهداری کنند و می‌توانند توسط سرویس‌دهندگان غیرمجاز اطلاعات مهمی از هویت شما را فاش کنند. به عنوان مثال اگر در یک سایت عضو شده باشید، چنانچه یک کوکی برای آن روی هارد سیستم شما ایجاد شده باشد، یک سایت غیرمجاز می‌تواند با اهداف مخرب از طریق خواندن کوکی به رمز عبور شما دسترسی پیدا کند.

 **نکته** کوکی‌ها به تنهایی نمی‌توانند برای سیستم مشکلات امنیتی ایجاد کنند اما می‌توانند در آسیب رساندن به امنیت سیستم شما دخیل باشند.

حداکثر اندازه کوکی‌ها ۴ kB (کیلوبایت) است. یکی از ویژگی‌های کوکی‌ها این است که پس از آن‌که توسط یک مرورگر روی یک سیستم ایجاد شدند، فقط روی همان سیستم می‌توانند خوانده شوند. هر کوکی یک آدرس URL منحصر به فرد دارد و سرویس‌گیرنده از روی این آدرس تعیین می‌کند که با کدام یک کار کند. استفاده از کوکی‌ها در سرویس‌گیرنده‌ها و مرورگرهای مختلف محدودیت‌هایی دارد. به عنوان مثال حداکثر کوکی‌های مرورگر Internet Explorer، ۳۰۰ فایل است.

## ۱-۱۱ ایجاد کوکی

ایجاد کوکی در فایل ها و اسناد PHP توسط دستور Setcookie انجام می شود.  
ساختار کلی این دستور به شکل زیر است:  
( "پارامتر امن", "اطلاعات دامنه", "تاریخ انقضای کوکی", "مقدار کوکی", "نام کوکی" ) Setcookie  
مثال:

Setcookie ("Webdesign", "#FE593", "01.12.2011, 23:5:35" )

توسط تاریخ انقضا می توان اعتبار یا عدم اعتبار کوکی را سنجید، اگر تاریخ انقضای یک کوکی تمام شده باشد، مرورگر دیگر آن را برای سرویس دهنده ارسال نمی کند (تاریخ انقضا با قالب استاندارد GMT<sup>۱</sup> تعیین می شود).

پارامترهای کوکی ها توسط سرویس گیرنده تعیین می شود و می تواند علاوه بر پارامترهای ساختار فوق، پارامترهای دیگری را نیز شامل شود، علاوه بر این ممکن است یک کوکی فقط شامل نام و یک پارامتر دیگر باشد.

اطلاعات دامنه تعیین می کند که کوکی توسط چه دامنه هایی قابل دسترسی است.  
اگر پارامتر امن نیز در خصوصیات کوکی تنظیم شده باشد، کوکی فقط مجاز است از طریق کانال های امن مانند https منتقل شود.

### ۱-۱-۱۱ مهم ترین پارامترهای کوکی

**path:** برای تعیین مسیر به کار می رود و مقدار پیش فرض آن تمام پوشه های سرویس دهنده است.

**domain:** دامنه مجاز را تعیین می کند که به طور پیش فرض همان دامنه سرویس دهنده ای است که آن را ایجاد کرده است.

**expire:** تاریخ انقضای کوکی را مشخص می کند.

**secure:** این گزینه سطح دسترسی به کوکی را تعیین می کند که به صورت پیش فرض غیرفعال است.



ساختار کلی ارسال کوکی:

HTTP/1.1 200 ok

Date : sun , 20 june 2011 22:44:50 GMT

Server : WAMP

Set-Cookie : Color = red ; expires = sun , 25 june 2011 22:30:00 GMT

Connection : Close

Connect-Type : text/html

## ۲-۱-۱۱ ایجاد کوکی

کوکی‌ها توسط یکی از دو تابع `Setcookie()` و `header()` ایجاد می‌شوند.

کوکی‌های ارسال شده از طرف هر سرویس دهنده در یک آرایه به نام `$_COOKIE` ذخیره شده‌اند و دسترسی به آن‌ها مانند سایر متغیرهاست.

به عنوان مثال اگر یک کوکی با نام `username` داشته باشیم می‌توانیم توسط `$_COOKIE` `«username»` به محتوای آن دسترسی پیدا کنیم.

به عنوان مثال می‌خواهیم یک کوکی ایجاد کنیم که تعداد بازدیدهای کاربر را از سایت تعیین کند.

می‌توانیم یک متغیر تعریف کنیم که با هر بار ملاقات توسط کاربر، به مقدار آن یک واحد افزوده شود.

```
<?php
```

```
$count=$_COOKIE["count"];
```

```
$count++;
```

```
Setcookie("count",$count,time()+36000);
```


```
?>
```

به ساختار کد فوق دقت کنید:

در سطر دوم یک متغیر جدید تعریف شده است که در صورتی که اولین بار باشد که

کوکی اجرا می‌شود این متغیر با عدد صفر مقداردهی می‌شود و در سطر سوم به آن یک

واحد اضافه می شود و در خط بعد کوکی ایجاد شده و مقدار متغیر count در آن ذخیره می شود. حال اگر بار دومی باشد که این برنامه اجرا می شود، مقدار count یک واحد دیگر افزوده شده و مقدار ۲ به خود می گیرد و به همین ترتیب در هر بار ملاقات صفحه توسط کاربر، یک واحد به این شمارنده افزوده می شود. در مثال فوق برای تعیین زمان انقضای کوکی از تابع () time استفاده شده است. تابع () time زمان جاری سیستم را نشان می دهد، مقدار تابع time با عدد ۳۶۰۰۰ جمع شده است، ۳۶۰۰۰ بر حسب ثانیه بوده و تعیین کننده ۱۰ ساعت است و این به آن معنی است که کوکی به مدت ۱۰ ساعت انقضا دارد.

 **نکته** • در یک کوکی اگر تاریخ انقضا مشخص نشود، کوکی تا زمانی اعتبار دارد که پنجره IE باز باشد و به محض بستن این پنجره تاریخ انقضای کوکی نیز به پایان خواهد رسید.

• هر تابع یا دستوری که قبل از تابع Setcookie در برنامه نوشته شود، اجرا نمی شود، لذا دستورات چاپ و ... را هرگز قبل از این دستور قرار ندهید و همیشه ساختار کوکی را در ابتدای صفحه پیاده سازی کنید.

### ۳-۱-۱۱ دسترسی به اطلاعات cookie

cookie های ارسال شده از طرف هر سرویس دهنده در یک آرایه به نام \$\_COOKIE ذخیره شده اند و دسترسی به آن ها مانند سایر متغیرهاست. به عنوان مثال اگر یک cookie با نام username داشته باشیم می توانیم توسط \$\_COOKIE["username"] به محتوای آن دسترسی پیدا کنیم.

**مثال:** می خواهیم یک cookie ایجاد کنیم که تعداد بازدیدهای کاربر را از سایت تعیین کند. برای این کار ابتدا یک متغیر تعریف می کنیم که با هر بار ملاقات توسط کاربر به مقدار آن یک واحد افزوده شود.

```
<?php
```

```
$count=$_COOKIE["count"];
```

```
$count++;  
Setcookie("count",$count,time()+36000);
```

?>

به ساختار کد فوق دقت کنید:

در سطر دوم یک متغیر جدید تعریف شده است که اگر اولین بار باشد که cookie اجرا می شود این متغیر با عدد صفر مقداردهی می شود و در سطر سوم به آن یک واحد اضافه می شود و در خط بعد cookie ایجاد شده و مقدار متغیر count در آن ذخیره می شود. حال اگر بار اول اجرای این برنامه نباشد، به مقدار قبلی count یک واحد افزوده شده و به همین ترتیب در هر بار ملاقات صفحه توسط کاربر، یک واحد به شمارنده افزوده می شود.

**سؤال:** تاریخ انقضای cookie در مثال فوق چه مدت است؟

**نکته** • در یک cookie اگر تاریخ انقضا مشخص نشود، cookie تا زمانی زنده است که پنجره IE باز باشد و به محض بستن این پنجره زمان انقضای cookie نیز به پایان خواهد رسید.

• هر تابع یا دستوری که قبل از تابع Setcookie در برنامه نوشته شود، اجرا نمی شود، لذا دستورات چاپ و... را هرگز قبل از این دستور قرار ندهید و همیشه ساختار cookie را در ابتدای صفحه پیاده سازی کنید.

## ۴-۱-۱۱ تابع isset()

یکی از توابع مهم و کاربردی در PHP تابع isset() است، توسط این تابع می توان از معرفی شدن متغیرها اطمینان حاصل کرد.

اگر متغیر مورد نظر قبلاً تعریف شده باشد، مقدار true و در غیر این صورت مقدار false را بر می گرداند.

فرض کنید در برنامه ای به مقدار یک متغیر نیاز داریم، اما نمی دانیم که این متغیر قبلاً معرفی و مقداردهی شده است یا خیر، در این صورت می توانیم از تابع isset() برای بررسی این مسأله استفاده کنیم.

مثال:

```
if (isset($first_name))  
{  
print '$first_name is set';  
}
```

در PHP علاوه بر دستور echo می توان از دستور print برای نمایش در خروجی استفاده نمود.

تمرین 

با استفاده از تابع `isset()` تعداد بازدیدهای کاربر از سایت را به دست آورید.

## ۵-۱-۱۱ حذف یک کوکی

برای حذف یک کوکی فقط کافی است تابع `Setcookie` را با نام فایل فراخوانی کنیم:  
`Setcookie ("نام کوکی")`;

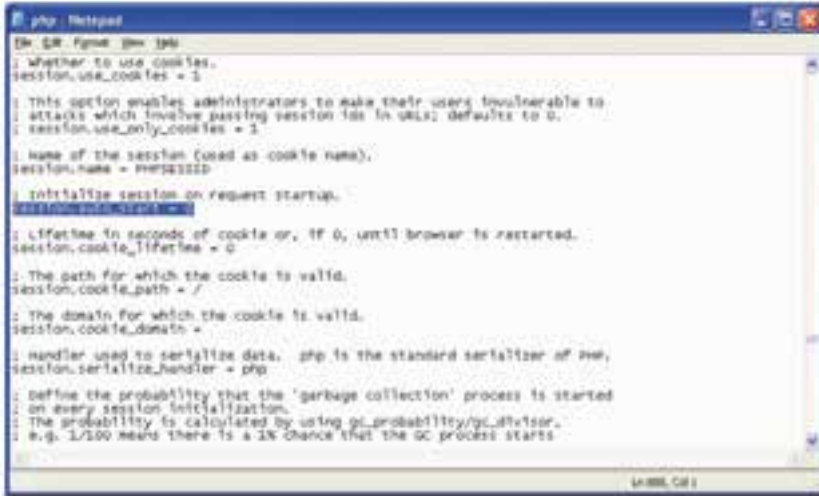
## ۲-۱۱ جلسه Session

Session یکی از مباحث اصلی شبکه و اینترنت است و توسط آن می توان اطلاعات کاربران را نگهداری کرد. هر کاربر که به سایتی وارد می شود با اجرای اسکریپت PHP یک جلسه ایجاد می شود و در صورتی که اولین بار باشد که وارد آن سایت می شود یک شناسه منحصر به فرد به او اختصاص داده می شود.

جلسه با دستور `Session_Start ()` شروع و ایجاد می شود.

شناسه جلسه توسط دستور `Session_id ()` تعیین می شود.

در فایل `PHP.ini` یک شناسه با عنوان `Session.auto_Start` وجود دارد، این شناسه به طور پیش فرض مقدار صفر دارد، اگر این مقدار را به ۱ تغییر دهید، برای هر سند PHP به طور خودکار یک جلسه ایجاد می شود اما اگر مقدار آن صفر باشد باید با استفاده از دستورات برنامه نویسی برای ایجاد جلسه اقدام شود.



شکل ۱-۱۱

زبان PHP اطلاعات جلسه را در فایل‌ها ذخیره می‌کند، مسیر ذخیره این فایل‌ها از طریق دستور `Session.Save_Path` تعیین می‌شود. پس از ایجاد یک جلسه می‌توان شناسه آن را به دست آورد.

```
<?php
session_start();
?>
<html>
<head>
<title> start a session</title>
</head>
<body>
<?php
echo " your session id is : ".session_id();
</body>
</html>
```

### ۱-۲-۱۱ متغیرهای جلسه

هر جلسه می‌تواند تعداد زیادی متغیر داشته باشد و توسط آن‌ها می‌توان اطلاعات کاربران را در صفحات مختلف مدیریت کرد.

این متغیرها نیز در یک آرایه فوق سراسری نگهداری می شوند.  
تعریف متغیرهای جلسه به صورت زیر است:

مقدار متغیر [= 'نام متغیر'] \$ SESSION

مثال: \$ SESSION ['number'] = 1239;

مثال: \$ SESSION ['name'] = "Zahra";

پس از تعریف متغیرهای جلسه در یک صفحه، در صفحات دیگر می توان به آنها دسترسی پیدا کرد، دسترسی به متغیرهای جلسه با همان نامی صورت می گیرد که هنگام تعریف متغیرها به کار رفته بود.

به عنوان مثال متغیرهای جلسه فوق در فایل دیگری از همان سایت از طریق دستورات زیر می توانند در خروجی چاپ شوند.

```
echo "$_SESSION ['number']";
```

```
echo "$_SESSION ['name']";
```

## ۱۱-۲-۲ حذف جلسه

حذف جلسه از طریق کد () Session\_destroy صورت می گیرد.

اگر این تابع در یک جلسه استفاده شود، تمام متغیرها و شناسه های جلسه نیز حذف می شوند.

```
<?php
```

```
Session_start ();
```

```
Session_destroy ();
```

```
?>
```

## ۱۱-۲-۳ نگهداری تنظیمات شخصی کاربر

همان طور که گفته شد جلسه ها می توانند متغیرهای زیادی در صفحه ایجاد کنند، این متغیرها می توانند حتی شامل اطلاعات محرمانه کاربر باشند و به دلیل این که اطلاعات جلسه به صورت یک آرایه فوق سراسری ذخیره می شود، می تواند مورد سوء استفاده هکرها قرار گیرد به همین دلیل باید به نوعی به دنبال مخفی کردن این اطلاعات بود.

رمزگذاری اطلاعات جلسه روشی است که توسط PHP قابل پشتیبانی است، این عمل توسط دستور () session\_encode صورت می گیرد، در مقابل اگر اطلاعات session بخواید مورد استفاده فایل PHP قرار گیرد باید از حالت رمز خارج شود و این عمل توسط دستور () session\_decode صورت می گیرد.

## واژه‌نامه

Expire	تاریخ انقضا
Path	مسیر
Secure	ایمن
Session	جلسه

## خلاصه مطالب

- کوکی‌ها فایل‌های متنی هستند که از طرف سرویس‌دهنده‌های اینترنت و توسط مرورگر روی سیستم کاربر ایجاد می‌شوند.
- حداکثر اندازه فایل‌های کوکی ۴ کیلوبایت است.
- کوکی‌ها به تنهایی نمی‌توانند برای سیستم مشکلات امنیتی پیش بیاورند، اما می‌توانند در آسیب رساندن به امنیت سیستم دخیل باشند.
- کوکی‌ها پس از این‌که توسط مرورگر روی یک سیستم ایجاد شدند، فقط روی همان سیستم می‌توانند خوانده شوند.
- در PHP برای ایجاد کوکی از دستور Setcookie استفاده می‌شود.
- یکی از مهم‌ترین ویژگی‌های کوکی‌ها، تاریخ انقضا است. توسط تاریخ انقضا می‌توان اعتبار یا عدم اعتبار کوکی را سنجید. اگر تاریخ انقضای یک کوکی تمام شده باشد، مرورگر دیگر آن را برای سرویس‌دهنده ارسال نمی‌کند.
- جلسه یکی از مباحث اصلی شبکه و اینترنت است، توسط جلسه می‌توان اطلاعات کاربران را نگهداری کرد، در یک صفحه وب که به زبان PHP نوشته شده است، به محض اجرای اسکریپت `<?php` یک جلسه ایجاد می‌شود.
- جلسه‌بادستور `Session_Start()` ایجاد و شناسه آن توسط دستور `Session_id()` تعیین می‌شود.

## آزمون نظری

- ۱ - حداکثر اندازه فایل های کوکی چقدر است؟  
الف- kB1      ب- kB4      ج- MB1      د- MB4
- ۲ - حداکثر کوکی های مرورگر IE چند عدد است؟  
الف- ۳۰      ب- ۳۰۰      ج- ۱۰۰      د- نامحدود
- ۳ - کدام یک از دستورات زیر برای ساخت کوکی به کار می رود؟  
الف- Setcookie      ب- Pathcookie  
ج- Cookie      د- Savecookie
- ۴ - برای حذف یک کوکی باید کدام یک از دستورات زیر استفاده شود؟  
الف- Delete      ب- DeleteCookie  
ج- Setcookie      د- Removecookie
- ۵ - پارامتر domain به طور پیش فرض کدام دامنه مجازی را برای کوکی در نظر می گیرد؟  
الف- com      ب- org      ج- net      د- دامنه سرویس دهنده
- ۶ - دستور حذف یک کوکی چند پارامتر ورودی دارد؟  
الف- ۱      ب- ۲      ج- ۳      د- پارامتر ورودی ندارد.
- ۷ - پس از تعریف متغیرهای جلسه در یک صفحه، دسترسی به آن ها در صفحات دیگر با چه نامی صورت می گیرد؟  
الف- با نام صفحه جدید  
ب- با همان نام متغیر جلسه  
ج- با نام صفحه ای که در آن تعریف شده است.  
د- دسترسی به متغیرهای جلسه در صفحات دیگر امکان پذیر نیست.
- ۸ - برای حذف جلسه از چه دستوری استفاده می شود؟  
الف- Sessiondestroy ()      ب- Sessionkill  
ج- Session\_destroy ()      د- Session\_kill ()



## آزمون عملی

- ۱ - یک کوکی جدید با عنوان Color با مقدار #ffffff با تاریخ انقضای به مدت ۱ ساعت بسازید.
- ۲ - تاریخ انقضای کوکی سؤال اول را طوری تنظیم کنید که به محض بستن مرورگر، کوکی نیز از بین برود.
- ۳ - تنظیمات سیستم را طوری انجام دهید که برای هر سند به طور خودکار یک جلسه ساخته شود.
- ۴ - در یک فایل با نام Page1 یک جلسه ایجاد کنید.
- ۵ - در یک فایل با نام Page2 پارامترهای جلسه Page1 را چاپ کنید.

## ارزیابی وبسایت و صفحات وب

### واحد کاردوازدهم

- پس از مطالعه این واحد کار از فراگیر انتظار می‌رود که:
- ۱ - معیارهای ارزیابی سایت را بشناسد.
  - ۲ - با اصول ارزیابی سایت آشنا باشد.

زمان(ساعت)	
عملی	نظری
۸	۵

## مقدمه

امروزه اینترنت به عنوان یکی از سریع‌ترین و در دسترس‌ترین ابزارهای بازیابی و تأمین اطلاعات به شمار می‌رود و از سوی کاربران مورد توجه بسیار زیادی واقع شده است. کاربران اینترنت معمولاً بدون این‌که به معیارهای ارزیابی سایت توجه داشته باشند، به اطلاعات آن استناد کرده و به آن اعتماد دارند.

و این امر در حالی اتفاق می‌افتد که به علت این‌که امکان ایجاد و طراحی یک سایت به سهولت در اختیار همگان قرار می‌گیرد، ممکن است هر فردی اقدام به ایجاد یک سایت با هر گونه اطلاعاتی بنماید و حتی می‌تواند به راحتی به اشاعه افکار خود بپردازد، از سوی دیگر در جامعه کنونی که یک جامعه اطلاعاتی به شمار می‌رود نمی‌توان به کلی استفاده از اینترنت را کنار گذاشت.

جهانی شدن و گرایش به دانش از ویژگی‌های اصلی عصر اطلاعات است، از این رو باید هر فردی که می‌خواهد از اینترنت استفاده کند مهارت‌هایی در ارزیابی صفحات وب سایت و محتویات صفحات آن‌ها داشته باشد.

## ۱-۱۲ ارزیابی صفحات وب

به‌طور کلی به منظور افزایش معلومات مفید کاربران باید بتوان یک ارتباط مؤثر بین منابع اطلاعاتی و افرادی که به دنبال آن هستند به وجود آورد. اطلاعات درست، بدون در نظر گرفتن محیطی که روی آن قرار گرفته‌اند دارای ارزش فراوانی است. اما این نباید باعث شود جستجوی مطالب از منابع و بازیابی داده‌ها و اطلاعات آن‌ها بدون هیچ‌گونه ارزیابی صورت بپذیرد. سازمان‌دهی و استفاده مؤثر از اطلاعات به منظور بالا بردن سطح تفکر و اندیشه، مهارت‌های ویژه‌ای را نیاز دارد. مجموعه این مهارت‌ها در قالب مهارت‌های اطلاع‌یابی قابل تعریف هستند.

### ۱-۱-۱۲ مهارت اطلاع‌یابی (یافتن اطلاعات)

مهارت‌های اطلاع‌یابی شامل مجموعه موارد زیر می‌باشد:  
۱ - مهارت در بازیابی اطلاعات

## ۲ - مهارت در ارزیابی اطلاعات

### ۳ - مهارت در تبادل اطلاعات

مرحله اول این مهارت‌ها توسط منابع اطلاعاتی مختلف و هم‌چنین آموزش روش‌های جستجو در آن‌ها قابل دستیابی است، اما مرحله ارزیابی اطلاعات یکی از کلیدی‌ترین مراحل از مهارت‌های فوق به شمار می‌رود. اطلاعات بازیابی شده ممکن است در هر ساختار و قالبی و از طرف هر فردی ارائه شده باشد، لذا باید محتوای اطلاعات، متناسب با نیاز کاربر ارزیابی شوند.

لازمه ارزیابی دقیق اطلاعات بازیابی شده داشتن یک تفکر دقیق و انتقادی است. کاربران با تقویت این نوع تفکر باید خود را در جایگاهی قرار دهند که به یافته‌های خود در اینترنت با یک نگاه منتقدانه برخورد کنند و این لازمه کسب توان ارزیابی اطلاعات است.

هر انسانی برای رفع نیازهای اطلاعاتی خود به دنبال کسب اطلاعات است و از اطلاعات کسب شده برای تصمیم‌گیری‌ها و بالا بردن دانش شخصی بهره می‌برد، لذا این تصمیم‌گیری‌ها زمانی ارزشمند و مفید خواهد بود که بر مبنای اطلاعات درست و صحیح صورت گرفته باشد.

از طرفی جامعه اطلاعاتی عصر حاضر ضرورت‌های دیگری را نیز دارد که عبارتند از:

- **جهانی شدن و افزایش ارزش اطلاعات:** یکی از نتایجی که با جهانی شدن به دست می‌آید، افزایش ارزش اطلاعات است و اطلاعات در این‌گونه جوامع به عنوان معیار توسعه به شمار می‌رود.

- **رشد اطلاعات در کنار آلودگی اطلاعات:** در کنار رشد سریع شبکه‌های اطلاعاتی و سهولت دسترسی به آن‌ها توسط کاربران، آلودگی اطلاعات نیز روز به روز در حال افزایش است.

- **کاربر نهایی و مهارت‌های یافتن اطلاعات:** از طرفی به علت فراگیر شدن شبکه‌های اطلاعاتی در سطح جوامع، خود افراد و بدون کمک دیگران اقدام به یافتن اطلاعات می‌کنند، به همین دلیل باید تمام افراد تا حدودی بتوانند اطلاعات مفید و معتبر را از غیر آن تشخیص دهند.

اطلاعاتی می‌تواند به‌طور مفید قابل استفاده کاربران قرار گیرد که مشخصات زیر را داشته باشد:

- ۱ - صحت
- ۲ - قابلیت اطمینان
- ۳ - به‌روز بودن
- ۴ - واقعیت داشتن
- ۵ - روشن و واضح بودن
- ۶ - اعتبار علمی
- ۷ - مؤثر بودن
- ۸ - کامل بودن
- ۹ - کم هزینه بودن و ...

## ۲-۱-۱۲ معیارهای ارزیابی اطلاعات

ارزیابی محتویات صفحات وب و اطلاعات از دو جنبه کمی و کیفی باید مورد توجه قرار بگیرد و فقط پرداختن به جنبه‌های کمی برای ارزیابی اطلاعات کافی نیست. برای ارزیابی صفحات وب می‌توان موارد زیر را مدنظر قرار داد:

### ۱ - ارزیابی هدف

هر چیزی که در اینترنت انتشار پیدا می‌کند، با هدف مشخصی صورت می‌گیرد، به عنوان مثال برخی از سایت‌ها صرفاً هدف تجاری دارند و برای رسیدن به این اهداف ممکن است هرگونه اطلاعاتی را درون سایت خود ارائه دهند.

به‌طور کلی نوع دامنه سایت‌ها یکی از مهم‌ترین مواردی است که می‌تواند ملاک تعیین اعتبار سایت باشد.

سایت‌هایی که با دامنه‌های gov, mil, net, edu, org هستند از اعتبار بیشتری نسبت به سایت‌های با دامنه‌های دیگر از قبیل com برخوردارند.

دامنه com به سایت‌های تجاری و آزاد داده می‌شود و نمی‌توان بدون بررسی و ارزیابی دقیق به محتویات آن‌ها اطمینان حاصل کرد.

هر کدام از دامنه‌های مذکور با اهداف مشخصی ثبت می‌شوند که کاربران باید این نکته را نیز مدنظر قرار دهند.

البته این معیار به‌طور صد در صد قابل اطمینان نیست و تنها تا حدودی می‌تواند مورد

استفاده قرار بگیرد.

## ۲ - ارزیابی شهرت و اعتبار

کاربران باید بتوانند ناشر یا ارائه‌دهندگان مطالب مندرج در صفحات سایت‌ها را شناسایی کنند، به عنوان مثال اگر یک مقاله در مورد مبحثی در یک صفحه وب قرار گرفته است، چنانچه نام نویسنده و آدرس ایمیل و متنی که مشخص‌کننده تخصص نویسنده باشد، درج شده باشد، مسلماً اعتبار بیشتری می‌تواند داشته باشد نسبت به سایتی که مطالب را بدون درج مشخصات ارائه‌دهنده در اختیار کاربران قرار می‌دهد. هم‌چنین یک وبسایت که دارای بخش ارتباط با ما یا درباره ما (About us) است و به کاربران اجازه شناسایی پدیدآورندگان وبسایت را می‌دهد می‌تواند یکی از ملاک‌های ارزیابی باشد.

## ۳ - ارزیابی صحت

فرض کنید در یک سایت پزشکی مطلبی در مورد شیوه علاج معده درد بیان شده است و کاربر بدون توجه به اعتبار مطلب و بدون ارزیابی آن، دستور ارائه شده را به کار می‌بندد، اما نه تنها بهبود پیدا نمی‌کند بلکه برای او مشکلات بدتری به وجود می‌آید. این مورد به وضوح لزوم ارزیابی صحت اطلاعات را بیان می‌کند. تعیین میزان صحت اطلاعات بازایی شده مهم‌ترین بخش ارزیابی انتقادی اطلاعات است. اغلب کاربران از روی ظاهر سایت و گرافیک جذاب آن به ارزیابی سایت می‌پردازند، در حالی که این امر دلیل مناسبی برای صحت اطلاعات مندرج در وبسایت نیست. محتوای مقالات و مباحث علمی سایتی که دارای بخش منابع نباشد را نمی‌توان چندان معتبر دانست. یک سایت که پس از ارائه مقالات علمی پیوندهای معتبر مختلفی را برای دستیابی کاربران به منابع در اختیار آن‌ها قرار داده است، در واقع با این عمل به اعتبار سایت خود افزوده است.

## ۴ - به روز بودن اطلاعات

با توجه به این‌که جامعه اطلاعاتی به سرعت در حال پیشرفت است، ممکن است اعتبار زمانی اطلاعات نیز به سرعت تغییر کند، لذا هنگام استفاده از اطلاعات مندرج در یک صفحه وب به تاریخ انتشار آن توجه داشته باشید. البته این امر به تنهایی نمی‌تواند دلیل محکمی بر به روز بودن اطلاعات باشد، فعال نبودن پیوندهای سایت، مخصوصاً پیوندهایی که به سایت‌های دیگر دارد، دلیلی بر به روز نبودن وبسایت است. برای


تعیین اعتبار زمانی باید به خود مطلب نیز توجه نمود، به عنوان مثال علم کامپیوتر روز به روز در حال پیشرفت است، لذا مطالب مرتبط با این علم که در سال ۱۹۹۸ درج شده‌اند، اغلب نمی‌تواند مورد استفاده قرار بگیرد، اما در مورد مطالبی که مربوط به دستور پخت غذاها هستند، اعتبار زمانی چندان مورد توجه نیست.

### ۵ - در دسترس بودن

دسترسی به منابع و اطلاعات مختلف در اینترنت ممکن است به روش‌های مختلفی محدود شود، به عنوان مثال ممکن است برخی سایت‌ها اطلاعات خود را فقط در اختیار کاربران عضو قرار دهند، یا این که برخی دیگر در ازای اخذ یک مبلغ مشخص به کاربران اجازه استفاده از منابع را بدهند، از دید بسیاری از کاربران این موارد خوشایند نیست و اغلب به دنبال این هستند که بتوانند به راحتی و در حداقل زمان به اطلاعات مورد نیاز خود دست پیدا کنند و حاضر نیستند حتی زمانی را صرف عضویت در این گونه سایت‌ها بنمایند.

### ۶ - نحوه ارائه اطلاعات

در وب‌سایت نوع سازماندهی اطلاعات، تعریف پیوندها، امکان جستجو در سایت، میزان تعاملی بودن سایت، سرعت بارگذاری صفحات، داشتن ساختار مشخص به نحوی که کاربر را دچار سردرگمی نکند نیز در ارزیابی سایت‌ها از عوامل بسیار مؤثر است.

**نکته**  ممکن است در صفحات وب پیوندهای متعددی وجود داشته باشد، روی این پیوندها دقت کنید، ممکن است فعال نباشند، یا این که شما را به بن‌بست هدایت کنند و یا این که اتصال شما را با مطلبی برقرار کنند که در واقع ارتباطی با اصل موضوع نداشته باشد، وجود این موارد در یک سایت، از اعتبار آن تا حدود زیادی می‌کاهد.

### ۷ - ارزیابی قیمت

بسیاری از سایت‌ها اطلاعاتشان را به‌طور رایگان در اختیار کاربران قرار نمی‌دهند، به عنوان مثال چکیده یک مطلب را به‌طور رایگان ارائه می‌دهند، اما کاربر برای مشاهده کل مقاله باید هزینه‌ای را بپردازد، در نتیجه باید ارزش نسبی مطالب را با توجه به هزینه‌ای که قرار است بپردازید حدس بزنید، تناسب محتوای ارائه شده با قیمت آن می‌تواند از ملاک‌های ارزیابی یک وب‌سایت باشد.

## واژه‌نامه

Accuracy	صحت
Adequacy	کفایت، مناسب بودن
Authority	اعتبار
Authorship	اصالت
Currency	پول، قیمت
Effective	مؤثر
Fairness	روشنی، زیبایی
Objectivity	واقعیت داشتن، وجود خارجی داشتن
Organization	سازمان
Reliability	قابلیت اطمینان

## خلاصه مطالب

- به منظور افزایش ارتباط مؤثر و در راستای افزایش معلومات کاربران باید بتوان بین منابع اطلاعاتی و کاربران ارزیابی درستی انجام داد.
  - از طرفی جامعه امروزی به عنوان یک جامعه اطلاعاتی ضرورت‌هایی دارد که عبارتند از:
    - ⚙️ جهانی شدن و افزایش ارزش اطلاعات
    - ⚙️ رشد همزمان اطلاعات و آلودگی اطلاعات
    - ⚙️ کاربر نهایی و مهارت‌های یافتن اطلاعات
- ارزیابی محتویات صفحات وب و اطلاعات از دو جنبه کمی و کیفی باید مورد توجه قرار بگیرد و برای این ارزیابی موارد زیر را باید در نظر داشت:

✓ ارزیابی هدف	✓ ارزیابی شهرت و اعتبار
✓ ارزیابی صحت	✓ ارزیابی به روز بودن اطلاعات
✓ ارزیابی در دسترس بودن	✓ ارزیابی نحوه ارائه اطلاعات
✓ ارزیابی قیمت	



## آزمون نظری

- ۱ - کدام یک از موارد زیر از ویژگی‌های اصلی عصر اطلاعات است؟  
الف- جهانی شدن و افزایش صفحات وب      ب- افزایش صفحات وب  
ج- جهانی شدن و گرایش به دانش د- گرایش به دانش و ارزیابی صفحات وب
- ۲ - مهارت یافتن اطلاعات شامل کدام یک از گزینه‌های زیر نیست؟  
الف- بازیابی اطلاعات      ب- ارزیابی اطلاعات  
ج- تبادل اطلاعات      د- ایجاد اطلاعات
- ۳ - لازمه ارزیابی دقیق اطلاعات بازیابی شده ..... است.  
الف- تبادل اطلاعات صحیح      ب- تفکر انتقادی  
ج- تفکر خلاق      د- الف و ب
- ۴ - کدام یک از دامنه‌های زیر از اعتبار کمتری نسبت به سایر دامنه‌ها برخوردار است؟  
الف- com      ب- edu      ج- org      د- ir
- ۵ - کدام یک از دامنه‌های زیر به سایت‌های تجاری اختصاص دارد؟  
الف- org      ب- net      ج- com      د- gov
- ۶ - قرار دادن بخش «درباره ما» در سایت به دستیابی کدام مورد از پارامترهای ارزیابی وب سایت کمک می‌کند؟  
الف- هدف      ب- شهرت و اعتبار  
ج- در دسترس بودن      د- نحوه ارائه اطلاعات
- ۷ - کدام یک از موارد زیر جزء ملاک‌های ارزیابی نیست؟  
الف- هدف      ب- اعتبار      ج- صحت      د- گستردگی

## آزمون عملی

- ۱ - یک فرم طراحی کنید که تمام ملاک‌های ارزیابی سایت‌ها برای امتیازدهی به سایت‌ها در آن در نظر گرفته شده باشد.
- ۲ - در زمینه علوم کامپیوتر، با استفاده از موتورهای جستجو، ۲ سایت پیدا کنید، سپس با استفاده از فرمی که در سؤال اول طراحی کرده‌اید به این سایت‌ها به طور مجزا امتیازدهی کنید.

### پیوست

در این بخش کدهای اجرایی مربوط به وبسایت توضیح داده شده در واحدکار آخر آمده است، با مطالعه دقیق کدها می‌توانید اشکالات احتمالی اجرای برنامه‌های خود را برطرف نمایید.

• کد فایل `sabt.php` دریافت اطلاعات از کاربر و ارسال آن به فایل `insert.php`

```
<html dir="rtl"><head>
<meta http-equiv="Content-Language" content="fa">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>New Page 1</title>

<script Language="Javascript">
function checkform()
{
var f;
f = document.forms[1];
if (f.username_on.value== "")
{
alert(" لطفاً نام کاربری را وارد نمایید ");
f.username_on.focus();
return false;
}
f = document.forms[1];
if (f.password_on.value== "")
{
alert(" لطفاً رمز عبور دلخواه را وارد کنید");
f.password_on.focus();
return false;
}
}
```

```
f = document.forms[1];
if (f.mail_on.value== "")
{
alert("لطفاً ایمیل خود را وارد کنید.");
f.mail_on.focus();
return false;
}

return true
}
</script>
<style type="text/css">
<!--
body {

background-color: #91C8FF;
}
-->
</style></head>

<body>

<div align=center>
<form action="?pg=insert.php" method=post name="form1" onSubmit="return
checkform()">
<table border="0" width="60%" id="table1" height="30%">
<tr>
<td colspan="2" height="21">
برای عضویت در سایت، این فرم را پر کنید:
<p align="center"><strong>
```





```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>انجام ثبت نام</title>
</head>
<body>
<div align=center>
</a></font></b>بازگشت<b><font face="Tahoma"><a href="index.php">
<table border=0>
<tr>
<td>

<?php
include "DB.php";
$myDb = new db();
$myDb->connect('on');

$username_on=$_POST['username_on'];
$password_on=$_POST['password_on'];
$mail_on=$_POST['mail_on'];
$anj_on=$_POST['anj_on'];

$query="SELECT * From user_on ORDER BY id ASC";
$result=$myDb->query($query);
$numRec = $myDb -> printrec($result);
$send = count($numRec);
//echo $send;
$i=0; $flag=0;
```

```
while ($i <> $end){
    $id = $numRec[$i]['id'];
    //echo $username_on."&nbsp;&nbsp; ".$numRec[$i]['username_on']."<br>";
    //echo $password_on."&nbsp;&nbsp; ".$numRec[$i]['password_on']."<br>";

if (($username_on == $numRec[$i]['username_on']) && ($password_on ==
    $numRec[$i]['password_on']))
    {
    $flag = 1; /****** if has found inside the table *****/
    }
    $i=$i+1;
    }

if (($username_on != "") and ($password_on != "") AND ($flag != 1))
    {
$query="insert into user_on (username_on,password_on,mail_on,anj_on) val-
ues (".$username_on.", ".$password_on.", ".$mail_on.", ".$anj_on.)";
$result=mysql_query($query);

$query2="SELECT * From user_on ORDER BY id ASC";
$result2=$myDb->query($query2);
$numRec2 = $myDb -> printrec($result2);
$end2 = count($numRec2);
echo $numRec[$end2]['id'];
$userid = $numRec2[$end2-1]['id'];
//echo $end2;
$query3="insert into license_on (user_id,license) values ($userid,2)";
// echo $query3;
$result3=mysql_query($query3);
```

```

if($result)
{
if (basename($_FILES['userfile']['name']) != "")
{

$query="SELECT * From user_on ORDER BY id ASC";
$result=$myDb->query($query);
$numRec = $myDb -> printrec($result);
$end = count($numRec);
$rec_id = $numRec[$end-1]['id'];

$uploaddir = './upload/';
$uploadfile =basename($_FILES['userfile']['name']);
list($file_name,$type) = split ('.', $uploadfile);
$uploadfile=$rec_id.".".$type;
if (move_uploaded_file($_FILES['userfile']['tmp_name'],      $uploaddir
.$uploadfile))
{
$flag_upload=1;
} else{
$flag_upload=0;
}

} //if (basename($_FILES['userfile']['name']) != "")

echo '<b></img></b>';

```



```
} //if($result)

}else{ //((( $username_on != "" ) and ( $password_on != "" ) AND ( $flag <> 1 ))

    echo "<b><img border='0' src='img/tekrar.jpg' width='200' height='41'></img></b>";
}
mysql_close();
?>
</td>
</tr>
<</table
<p>&nbsp;</div>
</body>

<html/>
```

• کد فایل DB.php برای کار با پایگاه داده و تنظیم اتصالات به آن:

```
<?php
class db{
    var $link;

    //-----
    function connect($dbname){
        $this->link = mysql_connect('localhost', 'root', '')
        or die('Couldnot connect: ' . mysql_error());
        mysql_select_db($dbname) or die('Could not select database');
    }

    //-----
    function query($query){
```

```

$result = mysql_query($query) or die('Query failed: ' . mysql_error());
return $result;
    mysql_free_result($result);
}
//-----
function update($query){
    mysql_query($query);
}
//-----
function add($sql){
    mysql_query($sql);
}
//-----
--
function printrec($result){
    $find=array();
    $line= mysql_fetch_array($result, MYSQL_ASSOC);
    array_push($find,$line);
    if($line==NULL){
        //echo("NULL");
        return;
    }
    else
    {
        //print_r($line);
    }
while ($line= mysql_fetch_array($result, MYSQL_ASSOC)) {
    //print_r($line);
    array_push($find,$line);
}

```

```
        return $find;
    }
//-----
    function clos(){
        mysql_close($this->link);
    }
//-----
    }
?>
```

• کد فایل question.php برای دریافت پرسش‌های کاربران و درج در پایگاه داده:

```
<html dir="rtl">

<head>
<meta http-equiv="Content-Language" content="fa">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>صفحه سؤالات متداول کاربران</title>
</head>
<script Language="Javascript">
function checkform()
{
    var f;
    f = document.forms[1];
    if (f.user_question.value=="")
        {
            alert("لطفاً نام کاربری را وارد نمایید");
            f.user_question.focus();
        }
    return false;
}
```

```
f = document forms[1];
if (f.anj_question.value= "")
    {
        alert("لطفاً نام انجمن مربوطه را وارد نمایید");
        f.anj_question focus();
    }
return false;
}
```

```
f = document.forms[1];
if (f.ques_question.value= "")
    {
        alert("شما هیچ سؤالی را وارد نکرده‌اید");
        f.ques_question focus();
        return false;
    }
return true
}
</script>
```

```
<body>
<?php
include "DB.php";
$myDb = new db();
$myDb->connect('on');

if (($_POST['ques_question'] != "") && ($_POST['flag'] == 1))
```

```
{
$user_question = $_POST['user_question'];
$anj_question = $_POST['anj_question'];
$ques_question = $_POST['ques_question'];

$query => insert into questions_on VALUES (
,$anj_question,$user_question,$ques_question,");
//echo $query;
$result = mysql_query($query);
if($result)
{
echo 'سؤال شما برای مدیر انجمن ارسال شد';

}ELSE{
echo 'سؤال ثبت نشد';

}
?>
<form action="" name="form1" method="post" onSubmit="return checkform()">

<div align="center">
&nbsp;<table border="0" width="45%" id="table3" cellspacing="0" cellpadding="0">
<tr>
<td width="100" align="center"><b>
<fontface="Tahoma" size="2" color="#00008" <کاربر<"/font></b></td>
<td width="329">
```

```

<b><font size="2" face="Tahoma"color="#000080">
<span lang="en-us">&nbsp;&nbsp;&nbsp;</span>
<?php
$query="SELECT * From questions_on ORDER BY id_question ASC";
$result=$myDb->query($query);
$numRec = $myDb -> printrec($result);
$end = count($numRec);
echo $numRec[$end]['id_question']+2;

    echo "<input type='hidden' name='user_question' size='1' value='1'>";
?>
</font></b>
</td>
</tr>
<tr>
<td width="100" align="center"><b>
<font face="Tahoma" size="2" color="#000080">انجمن</font></b></td>
<td width="329">&nbsp;&nbsp;<select name="anj_question" size="1">
<option value="VB">VB</option>
<option value="PHP">PHP</option>
<option value=" net">.net</option>
<option value="Access">Access</option>
<option value="MySQL">MySQL</option>
<option value="Excel">Excel</option>
<option value="متفرقه">متفرقه</option>
</select></td>
</tr>

```

```
<tr>
```

```
<td width="100" align="center"><b>
```

```
<font face="Tahoma" size="2" color="#000080">
```

```
ارسال سؤال </font></b></td>
```

```
<td width="329">&nbsp;<textarea type="text" name="ques_question"
```

```
rows="6" cols="37"></textarea></td>
```

```
</tr>
```

```
<tr>
```

```
<td width="100" height="25">&nbsp;</td>
```

```
<td width="329" height="25">&nbsp;<input type="submit" value="
```

```
"ارسال">
```

```
<input type="reset" value="انصراف">
```

```
<input type="hidden" name="flag" size="1" value="1">
```

```
</td>
```

```
</tr>
```

```
</table>
```

```
</div>
```

```
</form>
```

```
</body>
```

```
</html>
```

• کد فایل viewuser.php برای مشاهده مشخصات کاربرانی که قبلاً ثبت نام کرده اند:

```
<html dir="rtl">
```

```
<head>
```

```

<meta http-equiv="Content-Language" content="fa">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
</title>صفحه سؤالات متداول کاربران</title>
</head>
<script Language="Javascript">
function checkform()
{
var f;
f = document.forms[1];
if (f.user_question.value== "")
    {
        alert("لطفاً نام کاربری را وارد نمایید");
        f.user_question focus();
        return false;
    }
f = document.forms[1];
if (f.anj_question.value== "")
    {
        alert("لطفاً نام انجمن مربوطه را وارد نمایید");
        f.anj_question focus();
        return false;
    }

f = document forms[1];
if (f.ques_question.value=="")
    {
        alert("شما هیچ سؤالی را وارد نکرده‌اید");
    }

```



```
f.ques_question focus();  
return false;  
}  
return true  
}  
</script>  
  
<body>  
<?php  
include "DB.php";  
$myDb = new db();  
$myDb->connect('on');  
  
if (($_POST['ques_question'] != "") && ($_POST['flag'] == 1))  
{  
$user_question=$_POST['user_question'];  
$anj_question=$_POST['anj_question'];  
$ques_question=$_POST['ques_question'];  
  
$query="insert into questions_on VALUES ("  
, '$anj_question', '$user_question', '$ques_question', ")";  
//echo $query;  
$result=mysql_query($query);  
if($result)  
{  
echo 'سؤال شما برای مدیر انجمن ارسال شد';  
}ELSE{
```

```
echo 'سؤال ثبت نشد';
```

```
}
```

```
}
```

```
?>
```

```
<form action="" name="form1" method=post onSubmit="return checkform()">
```

```
<div align=center>
```

```
&nbsp;  <table border="0" width="45%" id="table3" cellspacing="0" cellpadding="0">
```

```
<tr>
```

```
<td width="100" align="center"><b>
```

```
<font face="Tahoma" size="2" color="#000080">کاربر</font></b></td>
```

```
<td width="329">
```

```
<b><font size="2" face="Tahoma" color="#000080">
```

```
<span lang="en-us">&nbsp;&nbsp;&nbsp;</span>
```

```
<?php
```

```
$query="SELECT * From questions_on ORDER BY id_question ASC";
```

```
$Result=$myDb->query($query);
```

```
$numRec = $myDb -> printrec($Result);
```

```
$end = count($numRec);
```

```
echo $numRec[$end]['id_question']+2;
```

```
echo "<input type='hidden' name='user_question' size='1' value='1'>";
```

```
?>
```

```
</font></b>
```

```
</td>
```

```
</tr>
```

```
<tr>
```

```
<td width="100" align="center"><b>
```

```
<font face="Tahoma" size="2" color="#000080">انجمن</font></b></td>
<td width="329">&nbsp;<select name="anj_question" size="1">
<option value="VB">VB</option>
<option value="PHP">PHP</option>
<option value=".net"> net</option>
<option value="Access">Access</option>
<option value="MySQL">MySQL</option>
<option value="Excel">Excel</option>
<option value="متفرقه">متفرقه</option>
</select></td>
</tr>
<tr>
<td width="100" align="center"><b>
<font face="Tahoma" size="2" color="#000080">ارسال سؤال</font></b></td>
<td width="329">&nbsp;<textarea type="text" name="ques_question"
rows="6" cols="37"></textarea></td>
</tr>
<tr>
<td width="100" height="25">&nbsp;</td>
<td width="329" height="25">&nbsp;<input type="submit" value="ارسال">
<br>
<input type="reset" value="انصراف">
<input type="hidden" name="flag" size="1" value="1">
</td>
</tr>
</table>
```

```
<</div>
</form>

</body>
</html>
```

• کد فایل index htm دریافت نام کاربری و کلمه عبور اعضا برای ورود کاربران و ارسال آن به فایل Enterance.php


```
<html dir="rtl">
<<head
<meta http-equiv="Content-Language" content="fa">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>New Page 1</title>
</head>

<body>

<form action="Enterance.php" method="post">
<font color="#000080" size="1"
face="Tahoma">نام کاربری:</font><br>
<input type="text" name="username" align="right"
size=8><br>
<b><font color="#000080" face="Tahoma"
size="1">کلمه عبور:</font></b><font color="#000080"><br>
</font><input type="text" name="pass
size=8><br><input type="submit" value="تأیید" size=11><input type="reset
value="انصراف"></form>

</body>
</html>
```

## • کد فایل Enterance.php برای بررسی نام کاربری و کلمه عبور اعضا:

**نکته**  در این فایل بررسی می‌شود که نام کاربری و کلمه عبور، مربوط به اعضای عادی است یا مدیران سایت، اگر کاربر عادی باشد امکانات مختلف از جمله مشاهده جواب سؤالاتی که از قبل مطرح کرده است، امکان درج سؤال جدید و ... در اختیار او قرار می‌گیرد، اما برای مدیران سایت امکان مشاهده سؤالات مربوط در انجمن‌ها، امکان پاسخ به سؤالات و ... فراهم می‌شود.

متن سؤالات توسط فایل show\_ques\_admin.php برای مدیر سایت نمایش داده می‌شود که در ادامه بیان شده است:

```
<?php
session_start();
?>
<html dir="rtl">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<<meta http-equiv="Content-Language" content="fa"
<title></title>
</head>

<body>
<?php
include "DB.php";
$myDb = new db();
$myDb->connect('on');
$query="SELECT * From user_on where username_on = '$_POST['username']' AND
password_on = '$_POST['pass']'";
```

```

$result=$myDb->query($query);
$num = $myDb -> printrec($result);
$send = count($num);
if ($send == 0)
{
?>
<div align=center>
<table border="1" width="70%" id="table1" cellspacing="0" cellpadding="0">
<tr>
<td width="70%">
</td>
</tr>
<tr>
<td width="70%">&nbsp;&nbsp;&nbsp;</td>
</tr>
</table>
</div>
<?php
}else{
$user_id = $num[0]['id'];
$_SESSION['anjoman_name'] = $num[0]['anj_on'];
//echo $_SESSION['anjoman_name'];
$query_lice="SELECT * From license_on where user_id = $user_id";
//echo $query_lice;
$result_lice=$myDb->query($query_lice);
$num_lice = $myDb -> printrec($result_lice);
$send_lice = count($num_lice);

```

```
if ($send_lice > 0){

if ($num_lice[0]['license'] == 1)
{
?>

<script language=javascript>
window.location="index.php?pg=show_ques_admin.php";
</script>

<?php
}else if ($num_lice[0]['license'] == 2)
{
?>

<script language=javascript>
window.location="index.php?pg=show_ques.php";
</script>

<?php
}
}
}
?>
</body>

</html>

• کد فایل show_ques_admin.php برای مشاهده سؤالات مربوط به مدیر انجمن:
<?php
session_start();
?>
<html dir="rtl">
```

```

<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>New Page 1</title>
</head>

<body>
<?php
include "DB.php";
$myDb = new db();
$myDb->connect('on');
$query="SELECT * From questions_on where answer_question = " ORDER BY
id_question DESC";
//echo $query;
$result=$myDb->query($query);
$num = $myDb -> printrec($result);
$end = count($num);
if ($end > 0)
{
?>
<div align=center>
<p><font color="#E7CFB8"><b>سؤالات متداول</b></font></p>

<table border="1" width="500" id="table2" cellspacing="0" cellpadding="0"
bordercolor="#FFFFFF">
<tr>
<td width="7%" align="center" bgcolor="#E7CFB8" height="40">
<b>
<font color=#5A3D1F face="Tahoma" size="2">ردیف</font></b></td>

```



```
<td width="220" align="center" bgcolor="#E7CFB8" height="40">
<b>
<font face="Tahoma" size="2"
color="#5A3D1F"> انجمن </font></b></td>
<td width="12%" align="center" bgcolor="#E7CFB8" height="40">
<b>
<font face="Tahoma" size="2" color="#5A3D1F"> کاربرد </font></b></td>
<td align="left" width="34%" bgcolor="#E7CFB8" height="40">
<p align="center"><b><font face="Tahoma" size="2"
color="#5A3D1F"> سؤال </font></b></td>
<td align="left" width="22%" bgcolor="#E7CFB8" height="40">
<p align="center"><b><font face="Tahoma" size="2"
color="#5A3D1F"> پاسخ </font></b></td>
</tr>

<?php
for($i=0;$i<=5;$i++)
{
?>
<tr>
<td width="7%" align="center" bgcolor="#F7EEE6" height="40"><b>
<font color="#5A3D1F" style="font-size: 9pt" face="Tahoma">
<?PHP ECHO $i+1; ?>

</font>
</b></td>
<td width="7%" align="center" bgcolor="#F7EEE6" height="40" ><b>
```

```

<font color="#5A3D1F" style="font-size: 9pt" face="Tahoma">
<?PHP ECHO $num[$i]['anj_question']; ?>
</font>
</b></td>
<td width="7%" align="center" bgcolor="#F7EEE6" height="40"><b>
<font color="#5A3D1F" style="font-size: 9pt" face="Tahoma">
<?PHP ECHO $num[$i][<user_question>]; ?>
</font>
</b></td>
<td width="7%" align="center" bgcolor="#F7EEE6" height="40"><b>
<font color="#5A3D1F" style="font-size: 9pt" face="Tahoma">
<?PHP ECHO $num[$i]['ques_question']; ?>
</font>
</b></td>
<td width="7%" align="center" bgcolor="#F7EEE6" height="40"><b>
<font color="#5A3D1F" style="font-size: 9pt" face="Tahoma">
<?PHP
IF ($num[$i]['answer_question'] != "")
{
    ECHO $num[$i]['answer_question'];
}ELSE{
    ECHO "---";
} ?>
</font>
</b></td>
</tr>
<?PHP

```

```
}
mysql_close();
?>
</table>
</div>
<?php
}else{
?>
<div align=center>
<p><font color="#E7CFB8"><b>سؤالات متداول</b></font></p>

<table border="1" width="400" id="table2" cellspacing="0" cellpadding="0"
bordercolor="#FFFFFF">
<tr>
<td align="center" bgcolor="#E7CFB8" height="40">
<p align="center"><span lang="fa"><b>
<font face="Tahoma" size="2" color="#5A3D1F">
در حال حاضر هیچ سؤالی جهت نمایش موجود نیست
</font></b></span></td>
</tr>

</table>
</div>
<?php
}
?>
</body>
```

```
</html>
```

• کد فایل show\_ques.php برای مشاهده کل سؤالات مطرح شده:

```
<?php
```

```
session_start();
```

```
?>
```

```
<html dir="rtl">
```

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

```
<title>New Page 1</title>
```

```
</head>
```

```
<body>
```

```
<?php
```

```
echo $_SESSION['anjoman_name'];
```

```
include "DB.php";
```

```
$myDb = new db();
```

```
$myDb->connect('on');
```

```
$query="SELECT * From questions_on where ( anj_question = " . $_SESSION[anjoman_
name] ." AND answer_question = " ) ORDER BY id_question DESC";
```

```
$result=$myDb->query($query);
```

```
$num = $myDb -> printrec($result);
```

```
$send = count($num);
```

```
if ($send > 0)
```

```
{
```

```
?>
```

```
<div align=center>
```

```
<p><font color="#E7CFB8"><b>سؤالات متداول</b></font></p>
```

```
<table border="1" width="500" id="table2" cellspacing="0" cellpadding="0"
bordercolor="#FFFFFF">
  <tr>
    <td width="7%" align="center" bgcolor="#E7CFB8" height="40">
      <b>
        <font color=#5A3D1F face="Tahoma" size="2">ردیف</font></b></td>
    <td width="220" align="center" bgcolor="#E7CFB8" height="40">
      <b>
        <font
          face="Tahoma"
          size="2"
color="#5A3D1F">انجمن</font></b></td>
    <td width="12%" align="center" bgcolor="#E7CFB8" height="40">
      <b>
        <font face="Tahoma" size="2" color="#5A3D1F">کاربر</font></b></td>
    <td align="left" width="34%" bgcolor="#E7CFB8" height="40">
      <p align="center"><b><font
        face="Tahoma"
        size="2"
color="#5A3D1F">سؤال</font></b></td>
    <td align="left" width="22%" bgcolor="#E7CFB8" height="40">
      <p align="center"><b><font face="Tahoma" size="2"
color="#5A3D1F">پاسخ</font></b></td>
  </tr>

  <?php
for($i=0;$i<$end;$i++)
{
  ?>
  <tr>
```

```
<td width="7%" align="center" bgcolor="#F7EEE6" height="40"><b>
```

```
<font color="#5A3D1F" style="font-size: 9pt" face="Tahoma">
```

```
<?PHP ECHO $i+1; ?>
```

```
</font>
```

```
</b></td>
```

```
<td width="7%" align="center" bgcolor="#F7EEE6" height="40" ><b>
```

```
<font color="#5A3D1F" style="font-size: 9pt" face="Tahoma">
```

```
<?PHP ECHO $num[$i]['anj_question']; ?>
```

```
</font>
```

```
</b></td>
```

```
<td width="7%" align="center" bgcolor="#F7EEE6" height="40"><b>
```

```
<font color="#5A3D1F" style="font-size: 9pt" face="Tahoma">
```

```
<?PHP ECHO $num[$i]['user_question']; ?>
```

```
</font>
```

```
</b></td>
```

```
<td width="7%" align="center" bgcolor="#F7EEE6" height="40"><b>
```

```
<font color="#5A3D1F" style="font-size: 9pt" face="Tahoma">
```

```
<?PHP ECHO $num[$i]['ques_question']; ?>
```

```
</font>
```

```
</b></td>
```

```
<td width="7%" align="center" bgcolor="#F7EEE6" height="40"><b>
```

```
<font color="#800000" style="font-size: 9pt" face="Tahoma">
```

```
<?PHP
```

```
IF ($num[$i]['answer_question'] != "")
```

```
{
```

```
    ECHO $num[$i]['answer_question'];
```

```
}ELSE{
```

```
    $ques_id = $num[$i]['id_question'];
```

```
ECHO "<a href='index.php?pg=answer.php&ques_id=$ques_id'>
```

```
"برای پاسخ به این سؤال کلیک کنید"
```

```
 } ?>
```

```
</font>
```

```
</b></td>
```

```
</tr>
```

```
<?PHP
```

```
 }
```

```
mysql_close();
```

```
 ?>
```

```
</table>
```

```
</div>
```

```
<?php
```

```
 }else{
```

```
 ?>
```

```
<div align=center>
```

```
<p><font color="#E7CFB8"><b>سؤالات متداول</b></font></p>
```

```
<table border="1" width="400" id="table2" cellspacing="0" cellpadding="0"
```

```
bordercolor="#FFFFFF">
```

```
 <tr>
```

```
 <td align="center" bgcolor="#E7CFB8" height="40">
```

```
 <p align="center"><span lang="fa"><b>
```

```
<font face="Tahoma" size="2" color="#5A3D1F">
```

در حال حاضر هیچ سؤالی جهت نمایش موجود نیست

```
</font></b></span></td>
```

```
</tr>
```

```
</table>
```

```
</div>
```

```
<?php
```

```
}
```

```
?>
```

```
</body>
```

```
</html>
```

• کد فایل `answer.php` برای ارائه پاسخ به سؤالات توسط مدیر انجمن:

```
<?php
```

```
session_start();
```

```
?>
```

```
<html dir="rtl">
```

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

```
<title>New Page 1</title>
```

```
</head>
```

```
<body>
```

```
<?php
```

```
echo $_SESSION['anjoman_name'];
```

```
include "DB.php";
```

```
$myDb = new db();
```

```
$myDb->connect('on');
```

```
$query="SELECT * From questions_on where (anj_question = " .  
$_SESSION[anjoman_name] . "AND answer_question = " ) ORDER BY id_question  
DESC";
```



```
$result=$myDb->query($query);
$num = $myDb -> printrec($result);
$end = count($num);
if ($end > 0)
{
?>
<div align=center>
<p><font color="#E7CFB8"><b>سؤالات متداول</b></font></p>

<table border="1" width="500" id="table2" cellspacing="0" cellpadding="0"
bordercolor="#FFFFFF">
<tr>
<td width="7%" align="center" bgcolor="#E7CFB8" height="40">
<b>
<font color=#5A3D1F face="Tahoma" size="2">رديف</font></b></td>
<td width="220" align="center" bgcolor="#E7CFB8" height="40">
<b>
<font face="Tahoma" size="2"
color="#5A3D1F">انجمن</font></b></td>
<td width="12%" align="center" bgcolor="#E7CFB8" height="40">
<b>
<font face="Tahoma" size="2" color="#5A3D1F">كاربر</font></b></td>
<td align="left" width="34%" bgcolor="#E7CFB8" height="40">
<p align="center"><b><font face="Tahoma" size="2"
color="#5A3D1F">سؤال</font></b></p>
<td align="left" width="22%" bgcolor="#E7CFB8" height="40">
```

```

        <p align="center"><b><font face="Tahoma" size="2"
color="#5A3D1F">پاسخ</font></b></td>
    </tr>

```

```

<?php
for($i=0;$i<$end;$i++)
{
?>
<tr>
<td width="7%" align="center" bgcolor="#F7EEE6" height="40"><b>
<font color="#5A3D1F" style="font-size: 9pt" face="Tahoma">
<?PHP ECHO $i+1; ?>
</font>
</b></td>
<td width="7%" align="center" bgcolor="#F7EEE6" height="40" ><b>
<font color="#5A3D1F" style="font-size: 9pt" face="Tahoma">
<?PHP ECHO $num[$i]['anj_question']; ?>
</font>
</b></td>
<td width="7%" align="center" bgcolor="#F7EEE6" height="40"><b>
<font color="#5A3D1F" style="font-size: 9pt" face="Tahoma">
<?PHP ECHO $num[$i]['user_question']; ?>
</font>
</b></td>
<td width="7%" align="center" bgcolor="#F7EEE6" height="40"><b>
<font color="#5A3D1F" style="font-size: 9pt" face="Tahoma">
<?PHP ECHO $num[$i]['ques_question']; ?>
</font>

```

```
</b></td>
```

```
<td width="7%" align="center" bgcolor="#F7EEE6" height="40"><b>
```

```
<font color="#800000" style="font-size: 9pt" face="Tahoma">
```

```
<?PHP
```

```
IF ($num[$i]['answer_question'] != "")
```

```
{
```

```
    ECHO $num[$i]['answer_question'];
```

```
}ELSE{
```

```
    $ques_id = $num[$i]['id_question'];
```

```
    ECHO "<a href='index.php?pg=answer.php&ques_id=$ques_id'>
```

«برای پاسخ به این سؤال کلیک کنید»:

```
    } ?>
```

```
</font>
```

```
</b></td>
```

```
</tr>
```

```
<?PHP
```

```
}
```

```
mysql_close();
```

```
?>
```

```
</table>
```

```
</div>
```

```
<?php
```

```
}else{
```

```
?>
```

```
<div align=center>
```

```
<p><font color="#E7CFB8"><b>سؤالات متداول</b></font></p>
```

```
<table border="1" width="400" id="table2" cellspacing="0" cellpadding="0">
```

```
bordercolor="#FFFFFF">
```

```
<tr>
```

```
<td align="center" bgcolor="#E7CFB8" height="40">
```

```
<p align="center"><span lang="fa"><b>
```

```
<font face="Tahoma" size="2" color="#5A3D1F">
```

در حال حاضر سؤالی جهت نمایش موجود نیست

```
</font></b></span></td>
```

```
</tr>
```

```
</table>
```

```
</div>
```

```
<?php
```

```
}
```

```
?>
```

```
</body>
```

```
</html>
```

### پاسخنامه

#### واحدکار اول

(د-۱) (ب-۲) (د-۳) (ج-۴)

#### واحدکار دوم

(الف-۱) (الف-۲) (الف-۳) (ج-۴)

#### واحدکار سوم

(ج-۱) (ب-۲) (د-۳) (ج-۴)  
(ب-۵) (ج-۶) (ج-۷) (الف-۸)  
(ب-۹) (د-۱۰) (ب-۱۱) (ج-۱۲)

#### واحدکار چهارم

(ج-۱) (ب-۲) (ج-۳) (د-۴)  
(الف-۵)

#### واحدکار پنجم

(ج-۱) (ب-۲) (ج-۳) (ب-۴)  
(ب-۵) (ج-۶)

#### واحدکار ششم

(د-۱) (الف-۲) (د-۳) (ب-۴)  
(ب-۵) (ج-۶) (الف-۷) (ب-۸)  
(الف-۹) (الف-۱۰) (ج-۱۱) (ب-۱۲)  
(الف-۱۳)

#### واحدکار هفتم

(الف-۱) (ج-۲) (الف-۳) (د-۴)

واحد کار ہشتم

- |       |       |       |         |
|-------|-------|-------|---------|
| (۱-ب) | (۲-ج) | (۳-ب) | (۴-الف) |
| (۵-د) | (۶-ج) |       |         |

واحد کار نہم

- |         |         |       |       |
|---------|---------|-------|-------|
| (۱-ج)   | (۲-الف) | (۳-ج) | (۴-د) |
| (۵-الف) | (۶-ب)   |       |       |

واحد کار دہم

- |       |       |       |         |
|-------|-------|-------|---------|
| (۱-د) | (۲-ب) | (۳-ج) | (۴-الف) |
| (۵-ج) | (۶-ب) | (۷-ج) |         |

واحد کار یازدہم

- |       |         |         |       |
|-------|---------|---------|-------|
| (۱-ب) | (۲-ب)   | (۳-الف) | (۴-ج) |
| (۵-د) | (۶-الف) | (۷-ب)   | (۸-ج) |

واحد کار دوازدہم

- |       |       |       |         |
|-------|-------|-------|---------|
| (۱-ج) | (۲-د) | (۳-ب) | (۴-الف) |
| (۵-ج) | (۶-ب) | (۷-د) |         |

## فهرست منابع

- 1 - An Introduction to data base system. Data C.j.
- 2 - [www.PHP.net](http://www.PHP.net)
- 3 - [www.developerstudio.ir](http://www.developerstudio.ir)

