

# MySQL

## فصل هفتم

هدف‌های رفتاری:

پس از آموزش این فصل هنرجو می‌تواند:

- ۱- نرم افزار WAMP را برای طراحی و مدیریت پایگاه داده، نصب و راه‌اندازی کند.
- ۲- با سیستم مدیریت پایگاه داده MySQL در برنامه PHP، ارتباط برقرار کند.
- ۳- در برنامه PHP، یک پایگاه داده جدید بسازد.
- ۴- اجزای مختلف پایگاه داده را ایجاد کند.
- ۵- اطلاعات مورد نظر را در جدول درج کند.
- ۶- اطلاعات درج شده را ویرایش و به روزرسانی کند.
- ۷- اطلاعات درج شده را نمایش دهد.
- ۸- رکورد یا رکوردهای مورد نظر را از جدول پایگاه داده، حذف کند.

## مقدمه

MySQL یک پایگاه داده (بانک اطلاعاتی) است و مانند سایر پایگاه‌های داده، اطلاعات را به صورت جدول ذخیره می‌کند.

همانطور که می‌دانید، جدول مجموعه‌ای از داده‌های مرتبط با هم است و سطرها و ستون‌های مختلفی تشکیل شده است.

طراحی پایگاه داده، به دلیل اینکه اطلاعات را به صورت گروه بندی شده ذخیره می‌کند، بسیار مفید است و کار بر روی داده‌ها را راحت می‌کند.

به طور مثال، پایگاه داده یک شرکت تجاری می‌تواند دارای جداول زیر باشد:

- کارمندان (Employees)

- محصولات (Products)

- مشتریان (Customers)

- سفارشات مشتریان (Orders)

شکل زیر یک مثال برای جدول مشتریان است که دارای چهار فیلد نام، نام خانوادگی، آدرس و شهر است.

جدول ۷-۱

City	Address	FirstName	LastName
تهران	خیابان بهار	احمد	ایمانی
همدان	میدان فردوسی	حسین	کریمی
کرمان	خیابان ولیعصر	لیلا	فراهانی

در جدول ۷-۱ سه رکورد مربوط به سه مشتری درج شده است.

پرس و جو (Query) نیز یک پرسش یا درخواست از پایگاه داده است که برای برگرداندن اطلاعات مشخص مجموعه‌ای از رکوردها استفاده می‌شود.

به طور مثال برای برگرداندن نام خانوادگی مشتریان در جدول ۷-۱ می‌توان پرس و جوی زیر را ایجاد نمود:

```
SELECT LastName FROM customers
```

با اجرای پرس و جوی فوق، نتیجه به صورت زیر برگردانده می‌شود:

جدول ۷-۲

LastName
ایمانی
کریمی
فراهانی

## ۷-۱ ایجاد پایگاه داده

ایجاد پایگاه داده در سرویس دهنده WAMP بسیار ساده است و نیاز به دانش بالایی در زمینه کار با زبان SQL و بانک اطلاعاتی ندارد. اما اگر از رابط متنی و با اجرای پرس و جوها به طور مستقیم در MySQL استفاده می کنید، بهتر است بر تمام دستورات ایجاد و مدیریت پایگاه داده و جداول تسلط کافی داشته باشید. برای ایجاد بانک اطلاعاتی در سرویس دهنده WAMP روی آیکن این برنامه بر روی ناحیه تذکر ویندوز کلیک کرده، سپس گزینه phpmyadmin را انتخاب کنید.



شکل ۷-۱

پنجره مربوط به ایجاد و مدیریت پایگاه داده به صورت شکل ۷-۲ باز می شود.



شکل ۲-۷

یکی از امکانات phpMyAdmin قابلیت پشتیبانی از زبان فارسی است و چنانچه در کادر باز شو Language که در شکل ۲-۷ با عنوان تغییر زبان مشخص شده است، گزینه Persian را انتخاب کنید، عبارات اصلی به زبان فارسی نشان داده می‌شوند.



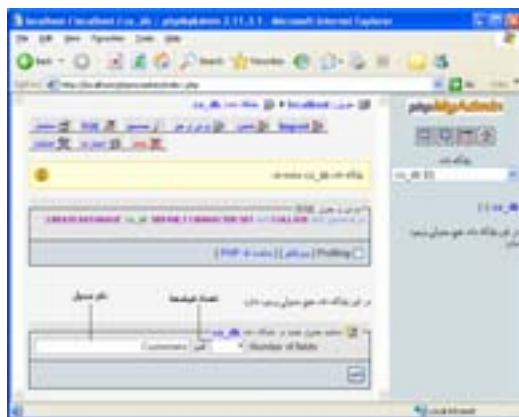
شکل ۳-۷

در شکل ۳-۷ بخش مربوط به ساخت پایگاه داده جدید مشخص شده است. در کادر متن «ساخت پایگاه داده جدید»، نام پایگاه داده‌ای که می‌خواهید ایجاد کنید را وارد نمایید، اگر قصد داشته باشید اطلاعات جداول را به صورت فارسی وارد کنید، باید در کادر باز شو Collation، کنار کادر متن نام پایگاه داده، گزینه `Utf8_general_ci` یا `Utf8_persian_ci` را انتخاب کنید.

سپس به منظور ایجاد پایگاه داده، روی دکمه "ساختن" کلیک کنید.

## ۷-۲ ساخت جداول پایگاه داده

پس از ساخت پایگاه داده، باید جدول های مورد نیاز را نیز طراحی کنید. برای ساخت جداول درون هر پایگاه داده مطابق شکل ۴-۷ و در بخش "ساختن جدول جدید" نام جدول و تعداد فیلدهای آن را در کادر متن مربوطه وارد کنید.



شکل ۴-۷

سپس برای ساخته شدن جدول روی دکمه تأیید کلیک کنید، پنجره phpMyAdmin نام فیلدها و نوع هر کدام به همراه سایر ویژگی ها را مطابق شکل ۵-۷ از شما دریافت خواهد کرد.



شکل ۵-۷

اگر بخش Collation را برای پایگاه داده تنظیم کرده باشید، قابلیت های زبان مورد نظر

شما در سراسر بانک در دسترس است و نیاز نیست در شکل ۵-۷ برای هر فیلد این قسمت را تنظیم کنید.  
پس از انجام تنظیمات، روی دکمه "ذخیره" کلیک کنید.

**تمرین:** بانک اطلاعاتی یک شرکت تجاری که در ابتدای همین فصل جداول آن نام برده شد را بسازید. پس از ساختن بانک اطلاعاتی، علاوه بر جدول customers فیلدهای سایر جداول را نیز طراحی کرده، سپس تمام آن‌ها را در phpMyAdmin ایجاد کنید.



### ۳-۷ 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 برنامه نویسان 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 شده‌ید، 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 چیز، 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 ۳-۷ بیان شده اند.

جدول ۳-۷

توضیحات	پارامتر
این پارامتر اختیاری است و نام سرورس دهنده را مشخص می کند، چنانچه این پارامتر مقدار دهی نشود، به طور پیش فرض مقدار «localhost:۳۳۰۶» برای آن تعیین می شود.	servername
این پارامتر نیز اختیاری است و شناسه کاربری سرورس دهنده را تعیین می کند. در سرورس 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 مقدار دهی نکنید، کاربری که پایگاه داده را ایجاد کرده است به طور پیش فرض تعیین می شود.	username
مقدار این پارامتر نیز اختیاری است و رمز ورود شناسه کاربری را دریافت می کند، چنانچه مقدار دهی نشود، یک رشته تهی را به طور پیش فرض در نظر می گیرد.	password

تابع `mysql_close()` نیز برای قطع ارتباط با پایگاه داده به کار می رود.

بهرتر است پس از خاتمه کار با پایگاه داده، توسط این تابع در برنامه اتصال به آن قطع شود.

اگر بخواهیم در PHP یک پیغام نمایش داده شود، سپس از اسکریپت خارج شود، از تابع `die()`

استفاده می شود، این تابع می تواند برای اطمینان از بروز یا عدم بروز خطا هنگام اجرای کدها استفاده شود.

مثال:



در مثال زیر اتصال به پایگاه داده در متغیر `$co` ذخیره می شود، سپس در برنامه از این متغیر در برنامه استفاده می شود. تابع `die()` در صورتی اجرا می شود که اتصال به پایگاه داده انجام نشده باشد.

پس از خاتمه عملیات نیز ارتباط با پایگاه داده قطع می شود.

```
<?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 که منجر به عدم اتصال به پایگاه داده شده است را نمایش می‌دهد.

## ۴-۷ ایجاد یک پایگاه داده و اجزای آن

همان‌طور که می‌دانید 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 با نحوه ایجاد پایگاه داده و هر یک از اجزای آن توسط PHP آشنا خواهید شد.

### ۴-۷-۱ ایجاد یک پایگاه داده

برای ایجاد پایگاه داده از دستور `CREATE DATABASE` استفاده می‌شود.

ساختار کلی این دستور به شکل زیر است:

نام پایگاه داده `CREATE DATABASE`

در PHP برای اجرای دستورات مربوط به SQL که بر روی پایگاه داده صورت می‌گیرد،

تابع `mysql_query()` به کار می‌رود. این تابع یک `query` برای ایجاد پایگاه داده به MySQL که اتصال به آن از قبل برقرار شده است ارسال می‌کند.

مثال:



مثال زیر پس از اتصال به MySQL، یک پایگاه داده با نام `my_db` می‌سازد:

```
<?php
```

```
$con = mysql_connect("localhost","root","");
```

```
if (!$con)
```



```

{
die('Could not connect: ' . mysql_error());
}

if (mysql_query("CREATE DATABASE my_db1",$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.



## ۲-۴-۷ ایجاد یک جدول در پایگاه داده

ایجاد جدول در پایگاه داده ای 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 شکل زیر 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.

**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);
?>
```

### ۳-۴-۷ فیلد کلیدی و خاصیت افزایش خودکار

هر جدول می تواند یک فیلد کلیدی داشته باشد که برای اطمینان از تکراری بودن در جدول استفاده می شود. دستور PRIMARY KEY برای تعریف فیلد کلیدی هنگام ساخت جدول به کار می رود. علاوه بر این فیلد کلیدی نمی تواند بدون مقدار باشد. جلوگیری از بدون مقدار بودن یک فیلد توسط دستور NOT Null صورت می گیرد.

در بسیاری از جدول ها به عنوان فیلد کلیدی شناسه ای غیر از مشخصه های اصلی موجودیت، انتخاب می شود. در این گونه مواقع معمولاً لازم است خاصیت افزایش خودکار این فیلد نیز انجام در بسیاری از جدول ها به عنوان فیلد کلیدی شناسه ای غیر از مشخصه های اصلی موجودیت، انتخاب می شود. در این گونه مواقع معمولاً لازم است خاصیت افزایش خودکار این فیلد نیز انجام شود. برای فعال کردن این خاصیت می توان 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 شده اند.

 **تمرین:** بانک اطلاعاتی مربوط به شرکت تجاری که در ابتدای این فصل معرفی شد را به همراه جداول و فیلدهای هر کدام با استفاده از دستورات PHP بسازید.

### ۴-۴-۷ درج اطلاعات در جدول

با استفاده از دستور INSERT INTO می توان 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,...)
```

 **نکته:** از این بعد در تمامی مثال‌ها فرض بر این است که پایگاه داده از قبل ساخته شده است، اما اگر از نام پایگاه داده‌ای که تاکنون ایجاد نشده است استفاده شود، عملیات کار با پایگاه داده ناموفق خواهد بود.  
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 آن را با نام person 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);
$query1= ("INSERT INTO Persons (FirstName, LastName, Age)
=mysql_query ($query1)
VALUES ('Mahdi', 'Razavi', '15')");
mysql_query($query1);
VALUES ('Hadi', 'Salimi', '13')");
mysql_close($con);
?>
```

همان طور که اشاره شد تابع `mysql_query()` برای اجرای دستورات مربوط به SQL به کار می رود، در مثال فوق، این تابع `query` مربوط به درج اطلاعات معادل فیلدهای `LastName`، `FirstName` و `Age` برای دو رکورد جدید را اجرا می کند.

### ۵-۴-۷ درج اطلاعات در جدول از طریق فرم

همانطور که می دانید، فرم ها به منظور تعامل با کاربران در صفحات وب سایت، استفاده می شوند. از فرم ها می توان برای دریافت اطلاعات از کاربر و درج در پایگاه داده، استفاده نمود. در مثال زیر یک فرم با سه فیلد برای ارتباط با جدول `person` طراحی شده است، با دریافت اطلاعات از کاربر و ارجاع به صفحه `insert.php`، عمل درج در پایگاه داده، انجام می شود.

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 می‌کند.

```
<?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 می‌شود هنگام خواندن آن‌ها، کدهای نامشخصی نمایش داده شود.



## ۷-۵ انتخاب و نمایش اطلاعات جدول

پس از ایجاد جدول و وارد کردن اطلاعات در آن می‌توانید با استفاده از دستور SELECT آن‌ها را انتخاب نمایید.

ساختار کلی این دستور به صورت زیر است:

```
SELECT نام فیلد یا فیلدها
FROM نام جدول
```

**مثال:**

در این مثال تمام اطلاعات جدول person 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 ۶-۷ نتیجه اجرای اسکریپت

همانطور که در شکل فوق مشاهده می‌کنید، تمام اطلاعات جدول person نمایش داده شده است.

در این مثال تابع `mysql_query()` با اجرای کد SQL تمام اطلاعات جدول را در متغیر `$result` می‌ریزد، سپس تابع `mysql_fetch_array()` اولین سطر از مجموعه رکوردها را که به صورت آرایه در نظر گرفته می‌شوند، درون متغیر `$row` می‌ریزد، به همین دلیل `$row` هم یک آرایه خواهد بود. با هر بار اجرای حلقه، و فراخوانی تابع `mysql_fetch_array()`، سطر بعدی فراخوانی می‌شود.

### ۱-۵-۷ نمایش اطلاعات پایگاه داده در جدول

با توجه به اینکه یکی از کاربردهای جدول‌ها در HTML، سازماندهی اطلاعات و نحوه نمایش آن‌هاست، بنابراین بهتر است اطلاعات فراخوانی شده از جداول های بانک اطلاعاتی را در قالب جدول نمایش دهیم.

مثال:



مثال قبل را طوری بازنویسی می‌کنیم که اطلاعات را در جدول نشان دهد:



```
<?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 ۷-۷ نمایش اطلاعات در جدول

**پژوهش:** اسکریپتی بنویسید که اطلاعات جدول را به صورت مرتب شده (صعودی) بر اساس "نام خانوادگی" نمایش دهد. (راهنمایی: با استفاده از دستور مرتب سازی در SQL)



## ۶-۷ به روز رسانی اطلاعات جدول

یکی از مهمترین عملیاتی که در پایگاه‌های داده روی اطلاعاتی که از قبل وارد شده‌اند، امکان ویرایش آن‌ها است. به منظور تغییر اطلاعات موجود در جدول از دستور UPDATE در استفاده می‌شود.

ساختار کلی این دستور به شکل زیر است:

نام جدول UPDATE  
 مقدار = فیلد دوم, مقدار = فیلد اول SET  
 شرط WHERE

مثال:



جدول person مثال‌های قبل را در نظر بگیرید، فرض کنید فیلد مربوط به Age برای فردی به نام Razavi Mahdi اشتباه درج شده باشد و بخواهیم مقدار آن را به ۱۶ تغییر دهیم.

جدول ۴-۷

FirstName	LastName	Age
Mahdi	Razavi	15
Hadi	Salimi	13

در این صورت کد اسکریپت مورد نظر به شکل زیر نوشته می شود:

```
<?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 Age = '16'
WHERE FirstName='Mahdi' AND LastName='Razavi'");
mysql_close($con);
?>
```

جدول ۵-۷ به روز رسانی جدول person

FirstName	LastName	Age
Mahdi	Razavi	16
Hadi	Salimi	13

## ۷-۷ حذف رکورد

دستور DELETE FROM برای حذف رکورد از جدول به کار می رود.

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 می شود.

مثال:



فرض کنید در جدول person بخواهیم رکورد فردی را که نام خانوادگی او Razavi است را 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 Last Name = 'Razavi'");

mysql_close($con);
?>
```

پس از اجرای کد فوق، جدول person به شکل زیر خواهد بود:

جدول ۶-۷

FirstName	LastName	Age
Hadi	Salimi	13

## چکیده فصل :

جدول مجموعه‌ای از داده‌های مرتبط با هم است و سطرها و ستون‌های مختلفی تشکیل شده است.

پرس و جو (Query) یک پرسش یا درخواست از پایگاه داده است که برای برگرداندن اطلاعات مشخص مجموعه ای از رکوردها استفاده می‌شود.

اتصال به پایگاه داده در PHP توسط تابع `mysql_connect()` انجام می‌شود.

تابع `mysql_close()` نیز برای قطع ارتباط با پایگاه داده به کار می‌رود.

تابع `mysql_error()` خطایی که منجر به عدم اتصال به پایگاه داده شده است را نمایش می‌دهد.

برای ایجاد پایگاه داده از دستور `CREATE DATABASE` استفاده می‌شود.

ایجاد جدول در پایگاه داده‌ای که از قبل ایجاد شده است، توسط دستور `CREATE`

`TABLE` انجام می‌شود.

پرس و جوها توسط دستور نیز توسط تابع `mysql_query()` اجرا می‌شوند.

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.

با استفاده از دستور `INSERT INTO` می‌توان اطلاعات را در یک جدول درج نمود.

به منظور تغییر اطلاعات موجود در جدول از دستور `update` در استفاده می‌شود.

دستور `DELETE FROM` برای حذف رکورد از جدول به کار می‌رود.

## خودآزمایی:

۱. توسط دستورات SQL، یک پایگاه داده برای کلاس خود ایجاد کنید.
۲. توسط دستورات SQL، در پایگاه داده سوال اول، جدولی برای نگهداری مشخصات همکلاسی‌های خود ایجاد کنید.
۳. توسط دستورات SQL، مشخصات خودتان را به همراه چند نفر از دوستان خود درون جدول سوال دوم، درج کنید.
۴. توسط دستورات SQL، تمام رکوردهای موجود در جدول را نمایش دهید.
۵. در جدولی که ایجاد نمودید، مشخصات خودتان را تغییر دهید.