



**UTM**  
UNIVERSITI TEKNOLOGI MALAYSIA

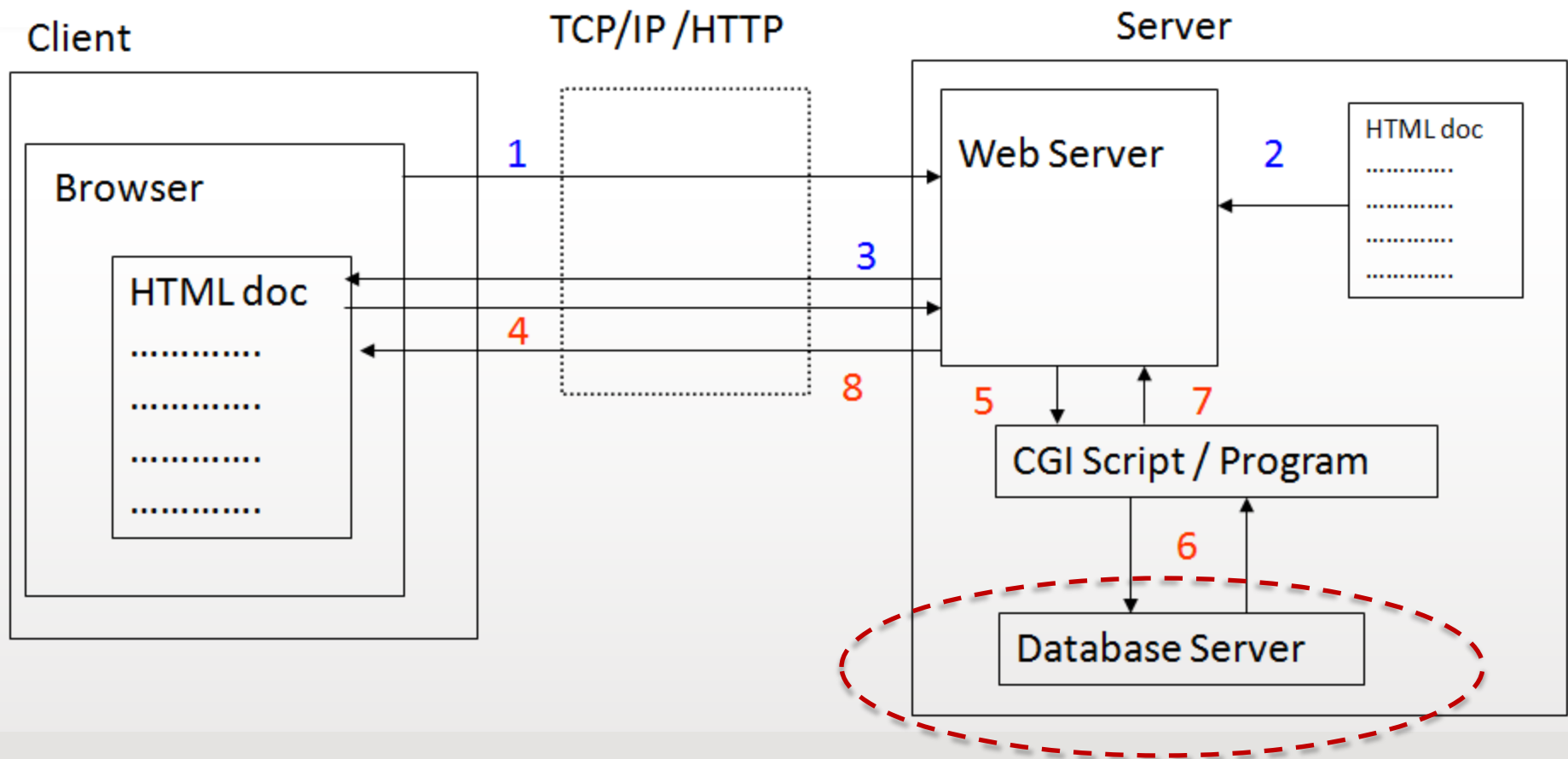


**Online  
Learning**

# PHP - Hypertext Preprocessor (Part 3)

**Sarina Sulaiman**  
**Faculty of Computing**

# Web Architecture: Database (MySQL)



# PHP MySQL – Connect/Disconnect

## Syntax

```
mysql_connect(servername,username,password);
```

Parameter	Description
servername	Optional. Specifies the server to connect to. Default value is "localhost:3306"
username	Optional. Specifies the username to log in with. Default value is the name of the user that owns the server process
password	Optional. Specifies the password to log in with. Default is ""

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

// some code

mysql_close($con);
?>
```

# PHP MySQL – `mysql_query()`

- **`mysql_query`** ( string `$query` [, resource `$link_identifier`] )
  - `$query`
    - A SQL query
    - The query string should not end with a semicolon.
  - `$link_identifier`
    - DB connection
    - If is not specified, the last link opened by [`mysql\_connect\(\)`](#) is assumed otherwise DB error

# PHP MySQL – Create (Db & Table)

## Syntax

```
CREATE DATABASE database_name
```

To get PHP to execute the statement above we must use the `mysql_query()` function. This function is used to send a query or command to a MySQL connection.

## Example

In the following example we create a database called "my\_db":

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$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);
?>
```

# PHP MySQL – Create (Db & Table)

## Create a Table

The CREATE TABLE statement is used to create a database table in MySQL.

### Syntax

```
CREATE TABLE table_name
(
column_name1 data_type,
column_name2 data_type,
column_name3 data_type,
.....
)
```

We must add the CREATE TABLE statement to the mysql\_query() function to execute the command.

# PHP MySQL – Create (Db & Table)

```
// Create table in my_db database
mysql_select_db("my_db", $con);
$sql = "CREATE TABLE Person
(
  FirstName varchar(15),
  LastName varchar(15),
  Age int
)";
mysql_query($sql,$con);

mysql_close($con);
?>
```

# PHP MySQL – Create (Db & Table)

Numeric Data Types	Description
int(size) smallint(size) tinyint(size) mediumint(size) bigint(size)	Hold integers only. The maximum number of digits can be specified in the size parameter
decimal(size,d) double(size,d) float(size,d)	Hold numbers with fractions. The maximum number of digits can be specified in the size parameter. The maximum number of digits to the right of the decimal is specified in the d parameter



# PHP MySQL – Create (Db & Table)

Textual Data Types	Description
char(size)	Holds a fixed length string (can contain letters, numbers, and special characters). The fixed size is specified in parenthesis
varchar(size)	Holds a variable length string (can contain letters, numbers, and special characters). The maximum size is specified in parenthesis
tinytext	Holds a variable string with a maximum length of 255 characters
text blob	Holds a variable string with a maximum length of 65535 characters
mediumtext mediumblob	Holds a variable string with a maximum length of 16777215 characters
longtext longblob	Holds a variable string with a maximum length of 4294967295 characters

# PHP MySQL – Create (Db & Table)

Date Data Types	Description
<code>date(yyyy-mm-dd)</code> <code>datetime(yyyy-mm-dd hh:mm:ss)</code> <code>timestamp(yyyymmddhhmmss)</code> <code>time(hh:mm:ss)</code>	Holds date and/or time

Misc. Data Types	Description
<code>enum(value1,value2,ect)</code>	ENUM is short for ENUMERATED list. Can store one of up to 65535 values listed within the ( ) brackets. If a value is inserted that is not in the list, a blank value will be inserted
<code>set</code>	SET is similar to ENUM. However, SET can have up to 64 list items and can store more than one choice

# PHP MySQL – Create (Db & Table)

## Primary Keys and Auto Increment Fields

Each table should have a primary key field.

A primary key is used to uniquely identify the rows in a table. Each primary key value must be unique within the table. Furthermore, the primary key field cannot be null because the database engine requires a value to locate the record.

The primary key field is always indexed. There is no exception to this rule! You must index the primary key field so the database engine can quickly locate rows based on the key's value.

The following example sets the personID field as the primary key field. The primary key field is often an ID number, and is often used with the AUTO\_INCREMENT setting. AUTO\_INCREMENT automatically increases the value of the field by 1 each time a new record is added. To ensure that the primary key field cannot be null, we must add the NOT NULL setting to the field.

## Example

```
$sql = "CREATE TABLE Person
(
personID int NOT NULL AUTO_INCREMENT,
PRIMARY KEY(personID),
FirstName varchar(15),
LastName varchar(15),
Age int
)";

mysql_query($sql,$con);
```

# PHP MySQL – INSERT

The **INSERT INTO** statement is used to insert new records into a database table.

## Insert Data Into a Database Table

The **INSERT INTO** statement is used to add new records to a database table.

### Syntax

```
INSERT INTO table_name  
VALUES (value1, value2,....)
```

You can also specify the columns where you want to insert the data:

```
INSERT INTO table_name (column1, column2,...)  
VALUES (value1, value2,....)
```

**Note:** SQL statements are not case sensitive. **INSERT INTO** is the same as **insert into**.

To get PHP to execute the statements above we must use the `mysql_query()` function. This function is used to send a query or command to a MySQL connection.

# PHP MySQL – INSERT

```
mysql_select_db("my_db", $con);  
  
mysql_query("INSERT INTO person (FirstName, LastName, Age)  
VALUES ('Peter', 'Griffin', '35')");  
  
mysql_query("INSERT INTO person (FirstName, LastName, Age)  
VALUES ('Glenn', 'Quagmire', '33')");  
  
mysql_close($con);  
?>
```

# PHP MySQL – SELECT

The **SELECT** statement is used to select data from a database.

## Select Data From a Database Table

The **SELECT** statement is used to select data from a database.

### Syntax

```
SELECT column_name(s)  
FROM table_name
```

**Note:** SQL statements are not case sensitive. **SELECT** is the same as **select**.

To get PHP to execute the statement above we must use the `mysql_query()` function. This function is used to send a query or command to a MySQL connection.

# PHP MySQL – SELECT

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

mysql_select_db("my_db", $con);

$result = mysql_query("SELECT * FROM person");

while($row = mysql_fetch_array($result))
{
    echo $row['FirstName'] . " " . $row['LastName'];
    echo "<br />";
}

mysql_close($con);
?>
```

# PHP MySQL – SELECT

- The previous example stores the data returned by the **mysql\_query()** function in the \$result variable.
- Next, we use the **mysql\_fetch\_array()** function to return the first row from the recordset as an array.
- Each subsequent call to **mysql\_fetch\_array()** returns the next row in the recordset.
- The while loop loops through all the records in the recordset.
- To print the value of each row, we use the PHP \$row variable (**\$row['FirstName']** and **\$row['LastName']**).



# PHP MySQL – SELECT

## ■ WHERE clause

```
SELECT column FROM table  
WHERE column operator value
```

The following operators can be used with the WHERE clause:

Operator	Description
=	Equal
!=	Not equal
>	Greater than
<	Less than
>=	Greater than or equal
<=	Less than or equal
BETWEEN	Between an inclusive range
LIKE	Search for a pattern

```
$result = mysql_query("SELECT * FROM person  
WHERE FirstName='Peter'");
```

# PHP MySQL – SELECT

## ■ ORDER by – record sorting

### The ORDER BY Keyword

The ORDER BY keyword is used to sort the data in a recordset.

### Syntax

```
SELECT column_name(s)
FROM table_name
ORDER BY column_name
```

**Note:** SQL statements are not case sensitive. ORDER BY is the same as order by.

### Sort Ascending or Descending

If you use the ORDER BY keyword, the sort-order of the recordset is ascending by default (1 before 9 and "a" before "p").

Use the DESC keyword to specify a descending sort-order (9 before 1 and "p" before "a"):

```
SELECT column_name(s)
FROM table_name
ORDER BY column_name DESC
```

# PHP MySQL – UPDATE

## Update Data In a Database

The UPDATE statement is used to modify data in a database table.

### Syntax

```
UPDATE table_name  
SET column_name = new_value  
WHERE column_name = some_value
```

**Note:** SQL statements are not case sensitive. UPDATE is the same as update.

To get PHP to execute the statement above we must use the `mysql_query()` function. This function is used to send a query or command to a MySQL connection.

# PHP MySQL – DELETE

## Delete Data In a Database

The DELETE FROM statement is used to delete records from a database table.

### Syntax

<

```
DELETE FROM table_name  
WHERE column_name = some_value
```

**Note:** SQL statements are not case sensitive. DELETE FROM is the same as delete from.

To get PHP to execute the statement above we must use the `mysql_query()` function. This function is used to send a query or command to a MySQL connection.

```
mysql_query("DELETE FROM Person WHERE LastName='Griffin'");
```

# Database (MySQL)

## Activity 11

- Database Manipulation - MYSQL - [db import notes](#) - ([download source](#)) | ([db.php](#))
- **SQL INSERT**
  - HTML: [form insert](#)
  - PHP: [server-side code](#)
- **SQL SELECT**
  - key-in matriks from HTML form ([html](#) | [php-server-side-code](#))
  - user select matriks from HTML form - combo box ([form](#) | [php-server-side-code](#))
  - display rows of data not in table ([source](#))
  - display rows of data in table ([source](#))

# Database (MySQL)

- **SQL UPDATE**

- user select matriks from HTML form - combo box  
([form](#) | [process](#) | [view](#))
- display data in table with A tag  
([table](#) | [process](#) | [view](#))

- **SQL DELETE**

- user select matriks from HTML form - combo box  
([form](#) | [process](#))
- data in table with X ([table](#) | [process](#))



**UTM**  
UNIVERSITI TEKNOLOGI MALAYSIA



**Online  
Learning**

<http://gmm.fsksm.utm.my/~rosely/scv1223/php/>

# THANK YOU