PHP/MySQL Tutorial
Introduction to Database

MySQL Database Simple Operation

- **USE Databasename: This will be used to select a database in the MySQL** workarea.
- **SHOW DATABASES: Lists out the databases that are accessible by the MySQL** DBMS.
- **SHOW TABLES: Shows the tables in the database once a database has been** selected with the use command.
- **SHOW COLUMNS FROM tablename: Shows the attributes, types of attributes,** key information, whether NULL is permitted, defaults, and other information for a table.
- **SHOW INDEX FROM tablename: Presents the details of all indexes on the table,** including the PRIMARY KEY.

Difference between mysqli_connect and mysql_connect?

- Mysql**i**_connect is the newer version of mysql library.
- The mysqli extension, or as it is sometimes known, the MySQL improved extension, was developed to take advantage of new features found in MySQL systems versions 4.1.3 and newer.
- MySQL and MySQLi are two separate PHP extensions, MySQLi being the newer one.
- The *mysqli* extension has a number of benefits, the key enhancements over the *mysql* extension being:
  - Object-oriented interface
  - Support for Prepared Statements
  - Support for Multiple Statements, Mysql does not support
  - Support for Transactions
  - Enhanced debugging capabilities
  - Embedded server support

MySQL Connectivity

- **`mysqli_connect()`**
  - □ The **`mysqli_connect()`** function opens a non-persistent MySQL connection.
  - □ This function returns the connection on success, or FALSE and an error on failure. You can hide the error output by adding an '@' in front of the function name.
- Syntax
  - □ **`mysqli_connect(server,user,pwd,newlink,clientflag)`**

MySQL Connectivity

Parameter	Description
server	Specifies the server to connect to
user	Specifies the username to log in with.
pwd	Specifies the password to log in with.
newlink	If a second call is made to mysql_connect() with the same arguments, no new connection will be established; instead, the identifier of the already opened connection will be returned
clientflag	•MYSQL_CLIENT_SSL - Use SSL encryption •MYSQL_CLIENT_COMPRESS - Use compression protocol •MYSQL_CLIENT_IGNORE_SPACE - Allow space after function names •MYSQL_CLIENT_INTERACTIVE - Allow interactive timeout seconds of inactivity before closing the connection

Example: 1.dbconnect.php

```
<?php
/* Attempt MySQL server connection. Assuming you are running MySQL
server with default setting (user 'root' with no password) */
$dbhost = 'localhost';
$dbuser = 'root';
$dbpass = '';
$conn = mysqli_connect($dbhost, $dbuser, $dbpass);
 // Check connection
if($conn === false){
    die("ERROR: Could not connect. " . mysqli_connect_error());
}
// Print host information
echo "Connect Successfully. Host info: " . mysqli_get_host_info($conn);
 // Close connection
mysqli_close($conn);
?>
```

Create a Database Using PHP Script

- PHP uses **mysqli_query function to create or delete a MySQL database.** This function takes two parameters and returns TRUE on success or FALSE on failure.
- **Syntax:** bool mysql_query(sql, connection)

Parameter	Description
Sql	Required – SQL query to create or delete a MySQL database.
connection	Optional – if not specified, then the last opened connection by mysql_connect will be used.

Example: 2. Creating MySQL Database Using PHP.php

```
<?php
/* Attempt MySQL server connection. Assuming you are running MySQL server with default setting
      (user 'root' with no password) */
$dbhost = 'localhost';
$dbuser = 'root';
$dbpass = '';
$conn = mysqli_connect($dbhost, $dbuser, $dbpass);

// Check connection
if($conn === false){
    die("ERROR: Could not connect. " . mysqli_connect_error());
}

// Attempt create database query execution
$sql = "CREATE DATABASE demo";
if(mysqli_query($conn, $sql)){
    echo "Database created successfully";
} else{
    echo "ERROR: Could not able to execute $sql. " . mysqli_error($conn);
}

// Close connection
mysqli_close($conn);
?>
```

Example: 3. Drop MySQL Database Using PHP.php

```
<?php
/* Attempt MySQL server connection. Assuming you are running MySQL server with default setting
      (user 'root' with no password) */
$dbhost = 'localhost';
$dbuser = 'root';
$dbpass = '';
$conn = mysqli_connect($dbhost, $dbuser, $dbpass);

// Check connection
if($conn === false){
    die("ERROR: Could not connect. " . mysqli_connect_error());
}

// Attempt drop database query execution
$sql = “drop DATABASE bp";
if(mysqli_query($conn, $sql)){
    echo "Database created successfully";
} else{
    echo "ERROR: Could not able to execute $sql. " . mysqli_error($conn);
}

// Close connection
mysqli_close($conn);
?>
```

MySQL – Datatypes

- MySQL uses many different data types broken into three categories:
  1. a. Numeric
  2. b. Date and Time
  3. c. String Types

Numeric Data Types

- **INT –** Can be signed or unsigned. Signed range 2147483648 to 2147483647. Unsigned range 0 to 4294967295.
- TINYINT – Can be signed or unsigned. Signed range -128 to 127. Unsigned range from 0 to 255.
- **SMALLINT** – Can be signed or unsigned. Signed range -32768 to 32767. Unsigned range 0 to 65535.
- **MEDIUMINT** – Can be signed or unsigned. Signed range -8388608 to 8388607. Unsigned range from 0 to 16777215.
- **BIGINT** – A large integer that can be signed or unsigned.
- **FLOAT(M,D)** – Floating-point number that cannot be unsigned. You can define the display length (M) and the number of decimals (D)
- **DOUBLE(M,D)**
- **DECIMAL(M,D)**

Date and Time Types

- **DATE –** A date in YYYY-MM-DD format, between 1000-01-01 and 9999-12-31. Example, December 30th 1973 would be stored as 1973-12-30.
- **DATETIME –** Date and time combination YYYY-MM-DD HH:MM:SS format, between 1000-01-01 00:00:00 and 9999-12-31 23:59:59. Exp, 3:30 afternoon December 30th 1973 stored as 1973-12-30 15:30:00.
- **TIMESTAMP –** Timestamp between midnight, January 1st 1970 and sometime in 2037. Looks like previous DATETIME format, but without the hyphens between numbers; 3:30 afternoon December 30th 1973 stored as 19731230153000 (YYYYMMDDHHMMSS).
- **TIME –** Stores the time in a HH:MM:SS format.
- **YEAR(M) –** Stores a year in a 2-digit or a 4-digit format. If the length is specified as 2 (for example YEAR(2)), YEAR can be between 1970 to 2069 (70 to 69). If the length is specified as 4, then YEAR can be 1901 to 2155. The default length is 4.

String Types

- **CHAR(M) –** A fixed-length string between 1 and 255 characters in length (for example CHAR(5))
- **VARCHAR(M)** – A variable-length string between 1 and 255 characters in length. For example, VARCHAR(25).
- **BLOB or TEXT** – Maximum length of 65535 characters. BLOBs are "Binary Large Objects" and are used to store large amounts of binary data, such as images or other types of files. Fields defined as TEXT also hold large amounts of data. You do not specify a length with BLOB or TEXT.

  The difference between the two is that sorts and comparisons on stored data are case sensitive on BLOBs and are not case sensitive in TEXT fields
- **TINYBLOB or TINYTEXT – M**aximum length of 255 characters
- **MEDIUMBLOB or MEDIUMTEXT** – Maximum length 16777215 characters
- **LONGBLOB or LONGTEXT –M**aximum length of 4294967295 characters.
- **ENUM** – An enumeration, which is a fancy term for list. When defining an ENUM, you are creating a list of items from which the value must be selected (or it can be NULL). For example, if you wanted your field to contain "A" or "B" or "C", you would define your ENUM as ENUM ('A', 'B', 'C') and only those values (or NULL) could ever populate that field

Creating Tables from MySQL

```
CREATE TABLE tutorials_tbl(
tutorial_id INT NOT NULL AUTO_INCREMENT,
tutorial_title VARCHAR(100) NOT NULL,
tutorial_author VARCHAR(40) NOT NULL,
submission_date DATE,
PRIMARY KEY ( tutorial_id )
);
```

Example: 4. Creating Tables inside MySQL Database Using PHP.php

- To create new table in any existing database you would need to use PHP function **mysqli_query().**

```
<?php
/* Attempt MySQL server connection. Assuming you are running MySQL
server with default setting (user 'root' with no password) */
$link = mysqli_connect("localhost", "root", "", "demo");
 // Check connection
if($link === false){
    die("ERROR: Could not connect. " . mysqli_connect_error());
}
 // Attempt create table query execution
$sql = "CREATE TABLE persons(
    id INT NOT NULL PRIMARY KEY AUTO_INCREMENT,
    first_name VARCHAR(30) NOT NULL,
    last_name VARCHAR(30) NOT NULL,
    email VARCHAR(70) NOT NULL UNIQUE
)";
if(mysqli_query($link, $sql)){
    echo "Table created successfully.";
} else{
    echo "ERROR: Could not able to execute $sql. " . mysqli_error($link);
}
 // Close connection
mysqli_close($link);
?>
```

Example: 5. DropTables inside MySQL Database Using PHP.php

- To create new table in any existing database you would need to use PHP function **mysqli_query().**

```
<?php
$dbhost = 'localhost';
$dbuser = 'root';
$dbpass = '';
$conn = mysql_connect($dbhost, $dbuser, $dbpass);
if(! $conn )
{
die('Could not connect: ' . mysql_error());
}
echo 'Connected successfully<br />';
$sql = "DROP TABLE persons";
mysql_select_db( 'demo' );
$retval = mysql_query( $sql, $conn );
if(! $retval )
{
die('Could not delete table: ' . mysql_error());
}
echo "Table deleted successfully\n";
mysql_close($conn);
?>
```

Example: 6. Inserting Data into a MySQL Database Table.php

```
<?php
/* Attempt MySQL server connection. Assuming you are running MySQL
server with default setting (user 'root' with no password) */
$link = mysqli_connect("localhost", "root", "", "demo");

// Check connection
if($link === false){
    die("ERROR: Could not connect. " . mysqli_connect_error());
}

// Attempt insert query execution
$sql = "INSERT INTO persons (first_name, last_name, email) VALUES ('Peter', 'Parker',
       'peterparker@mail.com')";
if(mysqli_query($link, $sql)){
    echo "Records inserted successfully.";
} else{
    echo "ERROR: Could not able to execute $sql. " . mysqli_error($link);
}

// Close connection
mysqli_close($link);
?>
```

Example: 7.Inserting Multiple Rows into a Table.php

```
<?php
/* Attempt MySQL server connection. Assuming you are running MySQL
server with default setting (user 'root' with no password) */
$link = mysqli_connect("localhost", "root", "", "demo");

// Check connection
if($link === false){
    die("ERROR: Could not connect. " . mysqli_connect_error());
}
 // Attempt insert query execution
$sql = "INSERT INTO persons (first_name, last_name, email) VALUES
            (‘John', 'Rambo', 'johnrambo@mail.com'),
            (‘Clark', 'Kent', 'clarkkent@mail.com'),
            (‘John', 'Carter', 'johncarter@mail.com'),
            (‘Harry', 'Potter', 'harrypotter@mail.com')";
if(mysqli_query($link, $sql)){
    echo "Records added successfully.";
} else{
    echo "ERROR: Could not able to execute $sql. " . mysqli_error($link);
}
// Close connection
mysqli_close($link);
?>
```

Example: 8. Selecting Data From Database Tables.php

```
<?php
$link = mysqli_connect("localhost", "root", "", "demo");
if($link === false){
    die("ERROR: Could not connect. " . mysqli_connect_error());
}
$sql = "SELECT * FROM persons"; // Attempt select query execution
if($result = mysqli_query($link, $sql)){
          if(mysqli_num_rows($result) > 0)
          {  echo "<table>";
             echo "<tr>";
             echo "<th>id</th>";
             echo "<th>first_name</th>";
             echo "<th>last_name</th>";
             echo "<th>email</th>";
             echo "</tr>";
                        while($row = mysqli_fetch_array($result))
                         {          echo "<tr>";
                                    echo "<td>" . $row['id'] . "</td>";
                                    echo "<td>" . $row['first_name'] . "</td>";
                                    echo "<td>" . $row['last_name'] . "</td>";
                                    echo "<td>" . $row['email'] . "</td>";
                                    echo "</tr>";
                           }
                echo "</table>";
               mysqli_free_result($result);     // Free result set. mysqli_free_result() function frees the memory associated with the result.
              }
              else{   echo "No records matching your query were found.";  }
}
else{  echo "ERROR: Could not able to execute $sql. " . mysqli_error($link);  }
mysqli_close($link);
?>
```

Example: 9.Filtering the Records firstname=john.php

```
<?php
$link = mysqli_connect("localhost", "root", "", "demo");
if($link === false){
    die("ERROR: Could not connect. " . mysqli_connect_error());
}
$sql = "SELECT * FROM persons WHERE first_name='john'";
if($result = mysqli_query($link, $sql)){
    if(mysqli_num_rows($result) > 0){
        echo "<table>";
            echo "<tr>";
                echo "<th>id</th>";
                echo "<th>first_name</th>";
                echo "<th>last_name</th>";
                echo "<th>email</th>";
            echo "</tr>";
        while($row = mysqli_fetch_array($result)){
            echo "<tr>";
                echo "<td>" . $row['id'] . "</td>";
                echo "<td>" . $row['first_name'] . "</td>";
                echo "<td>" . $row['last_name'] . "</td>";
                echo "<td>" . $row['email'] . "</td>";
            echo "</tr>";
        }
        echo "</table>";
        mysqli_free_result($result);
    } else{
        echo "No records matching your query were found.";
    }
} else{
    echo "ERROR: Could not able to execute $sql. " . mysqli_error($link);
}
mysqli_close($link);
?>
```