# 7.8 PHP Create MySQL Tables

A database table has its own unique name and consists of columns and rows.

##  Create a MySQL Table Using MySQLi and PDO

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

We will create a table named "MyGuests", with five columns: "id", "firstname", "lastname", "email" and "reg\_date":

CREATE TABLE MyGuests (
id INT(6) UNSIGNED AUTO\_INCREMENT PRIMARY KEY,
firstname VARCHAR(30) NOT NULL,
lastname VARCHAR(30) NOT NULL,
email VARCHAR(50),
reg\_date TIMESTAMP
)

* **Notes on the table above:**

The data type specifies what type of data the column can hold. For a complete reference of all the available data types, go to our [Data Types reference](https://www.w3schools.com/sql/sql_datatypes.asp).

After the data type, you can specify other optional attributes for each column:

* **NOT NULL** - Each row must contain a value for that column, null values are not allowed
* **DEFAULT value** - Set a default value that is added when no other value is passed
* **UNSIGNED** - Used for number types, limits the stored data to positive numbers and zero
* **AUTO INCREMENT** - MySQL automatically increases the value of the field by 1 each time a new record is added
* **PRIMARY KEY** - Used to uniquely identify the rows in a table. The column with **PRIMARY KEY** setting is often an ID number, and is often used with **AUTO\_INCREMENT**

Each table should have a primary key column (in this case: the "id" column). Its value must be unique for each record in the table.

The following examples shows how to create the table in PHP:

###  Example (MySQLi Object-oriented)

<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

**// Create connection**$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect\_error) {
    die("Connection failed: " . $conn->connect\_error);
}
**// sql to create table**$sql = "CREATE TABLE MyGuests (
id INT(6) UNSIGNED AUTO\_INCREMENT PRIMARY KEY,
firstname VARCHAR(30) NOT NULL,
lastname VARCHAR(30) NOT NULL,
email VARCHAR(50),
reg\_date TIMESTAMP
)";
if ($conn->query($sql) === TRUE) {
    echo "Table MyGuests created successfully";
} else {
    echo "Error creating table: " . $conn->error;}
$conn->close();
?>

###  Example (MySQLi Procedural)

<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

**// Create connection**$conn = mysqli\_connect($servername, $username, $password, $dbname);

**// Check connection**if (!$conn) {
    die("Connection failed: " . mysqli\_connect\_error());
}
**// sql to create table**$sql = "CREATE TABLE MyGuests (
id INT(6) UNSIGNED AUTO\_INCREMENT PRIMARY KEY,
firstname VARCHAR(30) NOT NULL,
lastname VARCHAR(30) NOT NULL,
email VARCHAR(50),
reg\_date TIMESTAMP
)";
if (mysqli\_query($conn, $sql)) {
    echo "Table MyGuests created successfully";
} else {
    echo "Error creating table: " . mysqli\_error($conn);
}
mysqli\_close($conn);
?>

###  Example (PDO)

<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDBPDO";
try {
    $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username, $password);

    **// set the PDO error mode to exception**    $conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

**// sql to create table**    $sql = "CREATE TABLE MyGuests (
    id INT(6) UNSIGNED AUTO\_INCREMENT PRIMARY KEY,
    firstname VARCHAR(30) NOT NULL,
    lastname VARCHAR(30) NOT NULL,
    email VARCHAR(50),
    reg\_date TIMESTAMP
    )";

    **// use exec() because no results are returned**
    $conn->exec($sql);
    echo "Table MyGuests created successfully";
    }
catch(PDOException $e)
    {
    echo $sql . "<br>" . $e->getMessage();
    }
$conn = null;
?>

# 7.10 PHP Insert Data Into MySQL

## Insert Data Into MySQL Using MySQLi and PDO

After a database and a table have been created, we can start adding data in them.

Here are some syntax rules to follow:

* The SQL query must be quoted in PHP
* String values inside the SQL query must be quoted
* Numeric values must not be quoted
* The word NULL must not be quoted

The INSERT INTO statement is used to add new records to a MySQL table:

INSERT INTO table\_name (column1, column2, column3,...)
VALUES (value1, value2, value3,...)

In the previous chapter we created an empty table named "MyGuests" with five columns: "id", "firstname", "lastname", "email" and "reg\_date". Now, let us fill the table with data.

**Note:** If a column is AUTO\_INCREMENT (like the "id" column) or TIMESTAMP (like the "reg\_date" column), it is no need to be specified in the SQL query; MySQL will automatically add the value.

The following examples add a new record to the "MyGuests" table:

### Example (MySQLi Object-oriented)

<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";
**// Create connection**$conn = new mysqli($servername, $username, $password, $dbname);

**// Check connection**if ($conn->connect\_error) {
    die("Connection failed: " . $conn->connect\_error);
}
$sql = "INSERT INTO MyGuests (firstname, lastname, email)
VALUES ('John', 'Doe', 'john@example.com')";

if ($conn->query($sql) === TRUE) {
    echo "New record created successfully";
} else {
    echo "Error: " . $sql . "<br>" . $conn->error;}
$conn->close();
?>

###  Example (MySQLi Procedural)

<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

**// Create connection**$conn = mysqli\_connect($servername, $username, $password, $dbname);

**// Check connection**if (!$conn) {
    die("Connection failed: " . mysqli\_connect\_error());
}
$sql = "INSERT INTO MyGuests (firstname, lastname, email)
VALUES ('John', 'Doe', 'john@example.com')";
if (mysqli\_query($conn, $sql)) {
    echo "New record created successfully";
} else {
    echo "Error: " . $sql . "<br>" . mysqli\_error($conn);}
mysqli\_close($conn);
?>

### 3- Example (PDO)

<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDBPDO";
try {
    $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username, $password);

    **// set the PDO error mode to exception**    $conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);
    $sql = "INSERT INTO MyGuests (firstname, lastname, email)
    VALUES ('John', 'Doe', 'john@example.com')";

**// use exec() because no results are returned**    $conn->exec($sql);
    echo "New record created successfully";     }
catch(PDOException $e)
    {
    echo $sql . "<br>" . $e->getMessage();    }
$conn = null;
?>