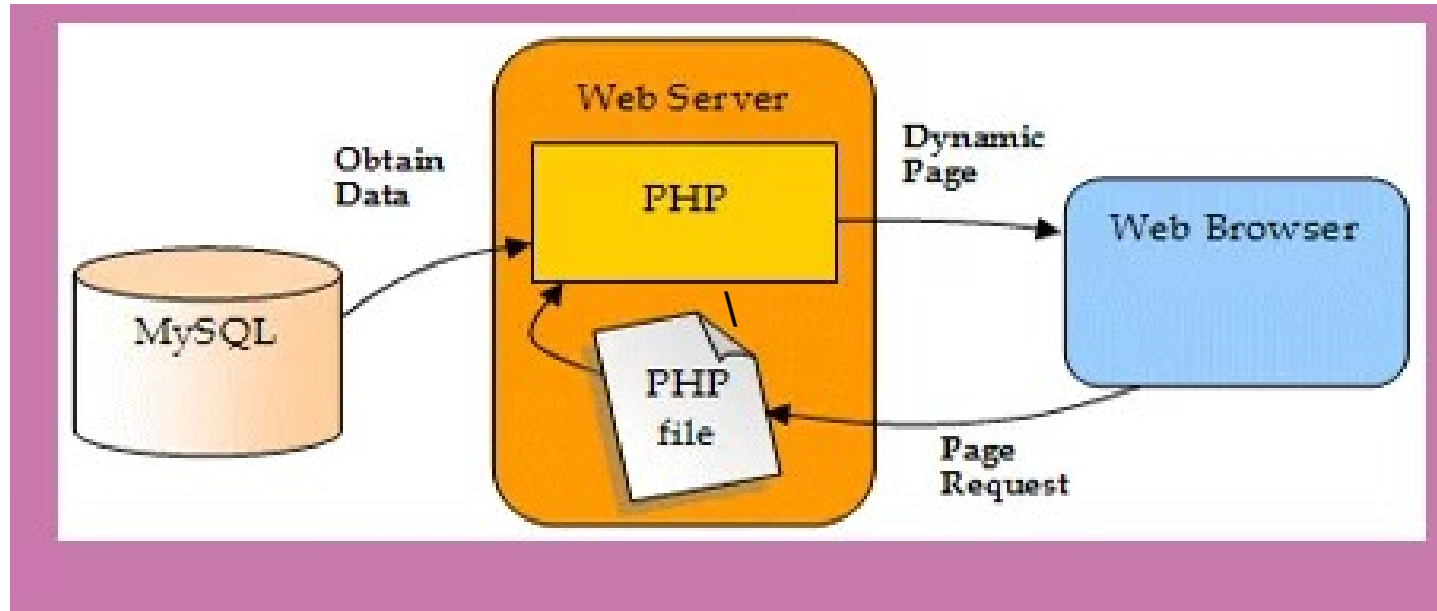


# Working with Database



# Working with Database



# Working with Database



- PHP seamlessly connects and manipulates databases for web development.
- MySQL is the go-to database system, widely used with PHP.
- PHP and MySQL together enable diverse database operations for dynamic web applications.
- Compatibility and extensive documentation enhance the efficiency of PHP and MySQL integration.
- PHP supports various database systems, providing flexibility beyond just MySQL like PostgreSQL.

# Working with Database



- MySQL, a web database system, stores data in tables with columns and rows.
- Suited for small to large applications, it operates on servers, offering speed, reliability, and user-friendliness.
- MySQL follows standard SQL, is freely available, and is developed by Oracle Corporation.
- The name "MySQL" is a personal touch, named after co-founder Monty Widenius's daughter, My.

# Working with Database



- A query is a question or a request.
- We can query a database for specific information and have a record returned.
- Look at the following standard SQL query:
  - **SELECT** Name **FROM** Students
- The query above selects all the data in the "Name" column from the "Students" table.

# Working with Database



- Start xampp server
- Url: <http://localhost/phpmyadmin/>

The screenshot displays the phpMyAdmin web interface for a local MySQL/MariaDB server. The left sidebar shows a tree view of databases including 'information\_schema', 'mysql', 'performance\_schema', 'phpmyadmin', and 'test'. The main content area is titled 'Server: localhost' and features a top navigation bar with tabs for 'Databases', 'SQL', 'Status', 'User accounts', 'Export', 'Import', 'Settings', 'Replication', and 'Variables'. The 'Settings' tab is active, showing 'Server connection collation' set to 'utf8mb4\_unicode\_ci'. Below this, the 'Appearance settings' section allows users to change the 'Language' to 'English' and the 'Theme' to 'pmahomme'. On the right, a 'Server status' panel lists details such as 'Server: Localhost via UNIX socket', 'Server type: MariaDB', 'Server version: 10.4.28-MariaDB', and 'User: root@localhost'. A 'Web server' panel at the bottom right lists the installed software stack: 'Apache/2.4.56 (Unix) OpenSSL/1.1.1t PHP/8.2.4 mod\_perl/2.0.12 Perl/v5.34.1'.

# Working with Database



- Click on **New**

Server: localhost

Databases SQL Status User accounts Export Import Settings Replication Variables More

### Databases

Create database

Database name: utf8mb4\_general\_ci Create

☐ Check all

Search

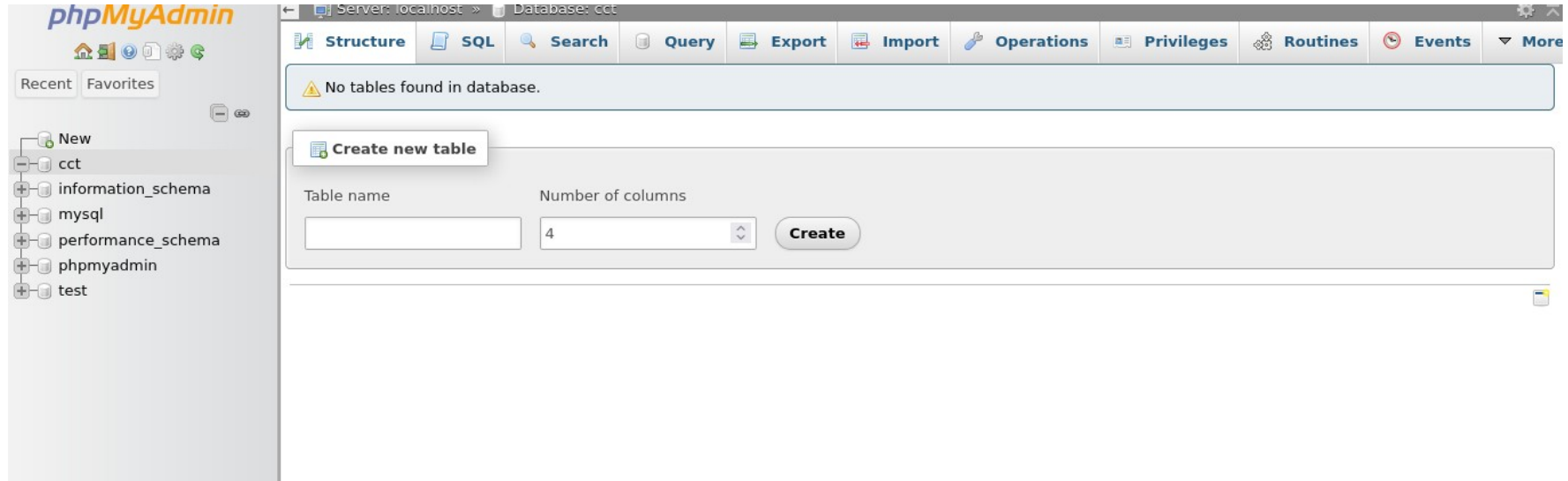
Database	Collation	Action
<input type="checkbox"/> information_schema	utf8_general_ci	<a href="#">Check privileges</a>
<input type="checkbox"/> mysql	utf8mb4_general_ci	<a href="#">Check privileges</a>
<input type="checkbox"/> performance_schema	utf8_general_ci	<a href="#">Check privileges</a>
<input type="checkbox"/> phpmyadmin	utf8_bin	<a href="#">Check privileges</a>
<input type="checkbox"/> test	utf8mb4_general_ci	<a href="#">Check privileges</a>

Total: 5

# Working with Database



- Provide Database name and click on create





# Working with Database



- Provide Table name, No of Columns and click on create

phpmyAdmin

Recent Favorites

New  
cct  
information\_schema  
mysql  
performance\_schema  
phpmyadmin  
test

Structure SQL Search Query Export Import Operations Privileges Routines Events More

Table name: student Add 1 column(s) Go

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index	
<input type="text"/>	INT	<input type="text"/>	None			<input type="checkbox"/>	---	
<a href="#">Pick from Central Columns</a>								
<input type="text"/>	INT	<input type="text"/>	None			<input type="checkbox"/>	---	
<a href="#">Pick from Central Columns</a>								
<input type="text"/>	INT	<input type="text"/>	None			<input type="checkbox"/>	---	
<a href="#">Pick from Central Columns</a>								
<input type="text"/>	INT	<input type="text"/>	None			<input type="checkbox"/>	---	
<a href="#">Pick from Central Columns</a>								

# Working with Database



- Fill the table student and click on save

phpMyAdmin

Server: localhost » Database: cct

Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Triggers

Table name: student Add 1 column(s) Go

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index	A_1	Comments
sid	INT	2	None			<input type="checkbox"/>	PRIMARY	<input type="checkbox"/>	
<a href="#">Pick from Central Columns</a>									
sname	VARCHAR	50	None			<input type="checkbox"/>		<input type="checkbox"/>	
<a href="#">Pick from Central Columns</a>									
faculty	VARCHAR	50	None			<input type="checkbox"/>		<input type="checkbox"/>	
<a href="#">Pick from Central Columns</a>									
phone	INT	10	None			<input type="checkbox"/>		<input type="checkbox"/>	
<a href="#">Pick from Central Columns</a>									

Table comments:

Collation:

Storage Engine: InnoDB

PARTITION definition:

Partition by:  ( Expression or column list )

Partitions:

Preview SQL Save

# Working with Database



- Then we see

phpMyAdmin

Recent Favorites

Server: localhost » Database: cct » Table: student

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking More

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 <b>sid</b>	int(2)			No	None			Change  Drop  More
<input type="checkbox"/>	2 <b>sname</b>	varchar(50)	utf8mb4_general_ci		No	None			Change  Drop  More
<input type="checkbox"/>	3 <b>faculty</b>	varchar(50)	utf8mb4_general_ci		No	None			Change  Drop  More
<input type="checkbox"/>	4 <b>phone</b>	int(10)			No	None			Change  Drop  More

☐ Check all With selected: Browse Change Drop Primary Unique Index Spatial Fulltext

Add to central columns Remove from central columns

Print Propose table structure Track table Move columns Normalize

Add  column(s)

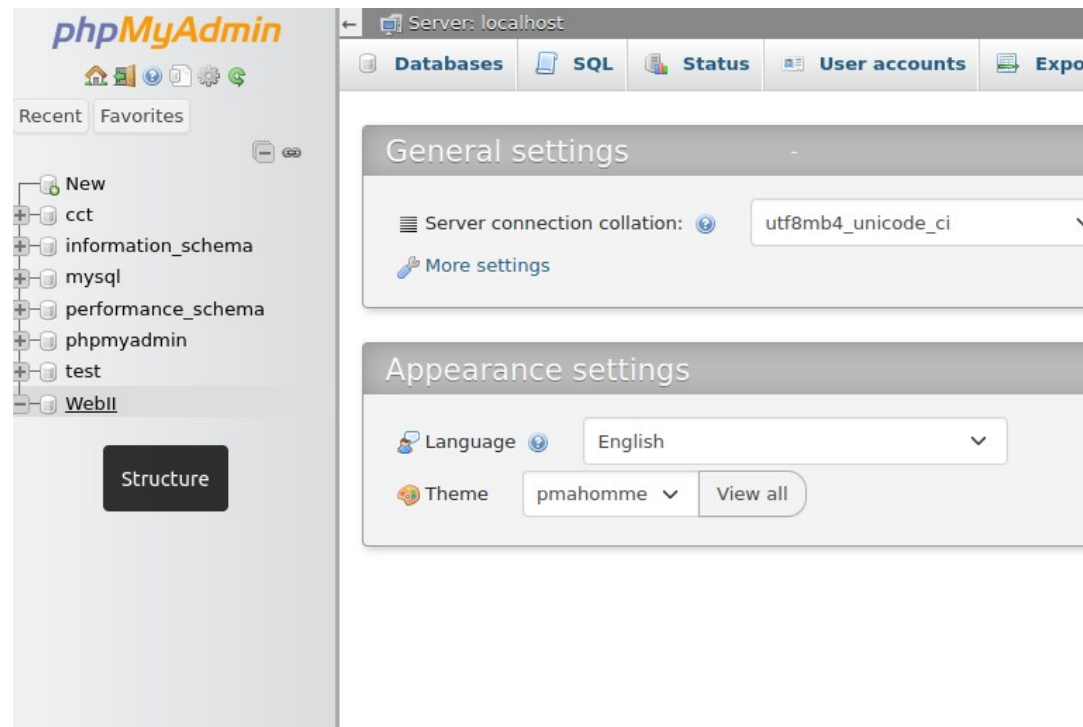
Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit  Rename  Drop	PRIMARY	BTREE	Yes	No	sid	0	A	No	

# Working with Database: Create



```
1  <?php
2  $servername = "localhost";
3  $username = "root";
4  $password = "";
5  // Create connection
6  $conn = new mysqli($servername, $username, $password);
7  // Check connection
8  if ($conn->connect_error) {
9      die("Connection failed: " . $conn->connect_error);
10 }
11 // Create database named WebII
12 $sql = "CREATE DATABASE WebII";
13 if ($conn->query($sql) === TRUE) {
14     echo "Database created successfully";
15 } else {
16     echo "Error creating database: " . $conn->error;
17 }
18 $conn->close();
19 ?>
```





# Working with Database: Create

```
1 <?php
2 $servername = "localhost";
3 $username = "root";
4 $password = "";
5 $dbname = "WebII";
6 // Create connection
7 $conn = new mysqli($servername, $username, $password, $dbname);
8 // Check connection
9 if ($conn->connect_error) {
10     die("Connection failed: " . $conn->connect_error);
11 }
12 // sql to create table
13 $sql = "CREATE TABLE Student (
14     sid INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
15     firstname VARCHAR(30) NOT NULL,
16     lastname VARCHAR(30) NOT NULL,
17     email VARCHAR(50),
18     phone VARCHAR(10)
19 );";
20 if ($conn->query($sql) === TRUE) {
21     echo "Table Student created successfully";
22 } else {
23     echo "Error creating table: " . $conn->error;
24 }
25 $conn->close();
26 ?>
```

The screenshot shows the phpMyAdmin interface. On the left, the database structure tree is visible, with 'WebII' selected and 'Student' highlighted. The main panel displays the 'Table structure' view for the 'Student' table. The table has five columns: 'sid' (int(6), UNSIGNED, AUTO\_INCREMENT, PRIMARY KEY), 'firstname' (varchar(30), utf8mb4\_general\_ci), 'lastname' (varchar(30), utf8mb4\_general\_ci), 'email' (varchar(50), utf8mb4\_general\_ci), and 'phone' (varchar(10), utf8mb4\_general\_ci). Below the table structure, there are options to 'Check all', 'With selected', 'Browse', 'Change', 'Drop', 'Primary', 'Unique', 'Index', 'Spatial', and 'Fulltext'. At the bottom, there is a 'Print' button and a 'Propose table structure' button. The 'Indexes' tab is active at the bottom.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	sid	int(6)		UNSIGNED	No	None		AUTO_INCREMENT	<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
2	firstname	varchar(30)	utf8mb4_general_ci		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
3	lastname	varchar(30)	utf8mb4_general_ci		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
4	email	varchar(50)	utf8mb4_general_ci		Yes	NULL			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
5	phone	varchar(10)	utf8mb4_general_ci		Yes	NULL			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>

# Working with Database: Insert data



- Here are some syntax rules to follow:
  - PHP requires SQL queries to be enclosed in quotes.
  - String values within the SQL query must be surrounded by quotes.
  - Numeric values should not be enclosed in quotes.
  - The keyword "NULL" must not be quoted.
- The INSERT INTO statement is employed to insert new records into a MySQL table.
- It follows the format:
  - ***INSERT INTO table\_name (column1, column2, column3, ...)***
  - ***VALUES (value1, value2, value3, ...);***

# Working with Database: Insert Data



```
1 <?php
2 $servername = "localhost";
3 $username = "root";
4 $password = "";
5 $dbname = "WebII";
6 // Create connection
7 $conn = new mysqli($servername, $username, $password, $dbname);
8 // Check connection
9 if ($conn->connect_error) {
10     die("Connection failed: " . $conn->connect_error);
11 }
12 $sql = "INSERT INTO Student (firstname, lastname, email, phone)
13 VALUES ('Prakash', 'Neupane', 'praksh@gmail.com', 9842512345)";
14 if ($conn->query($sql) === TRUE) {
15     echo "New record created successfully";
16 } else {
17     echo "Error: " . $sql . "<br>" . $conn->error;
18 }
19 $conn->close();
20 ?>
```

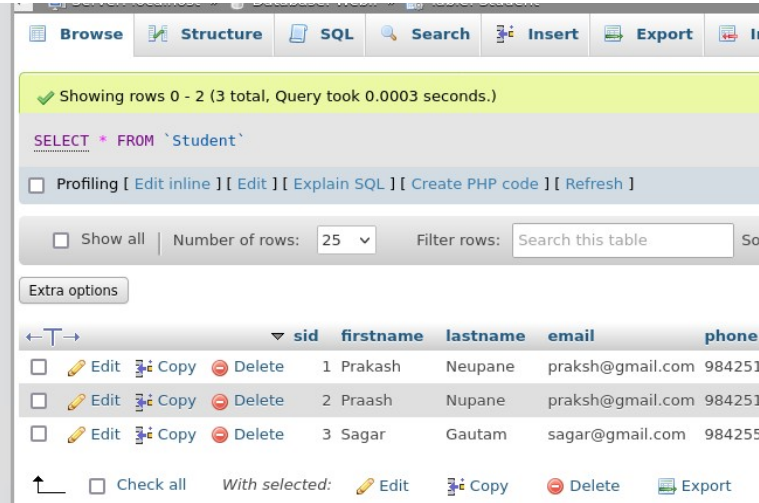
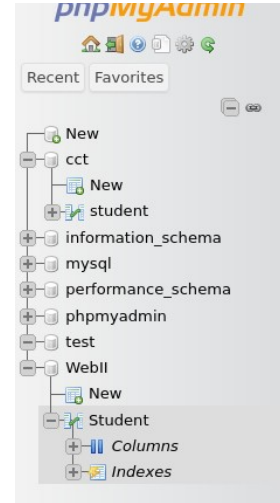
The screenshot shows the phpMyAdmin interface. On the left, the database structure is visible, with 'WebII' selected. The 'Student' table is highlighted. The main panel shows the 'Table: Student' view. A green message at the top indicates 'Showing rows 0 - 0 (1 total, Query took 0.0003 seconds.)'. Below this, the SQL query 'SELECT \* FROM `Student`' is displayed. The 'Query results operations' section shows a table with one row of data:

	sid	firstname	lastname	email	phone
1	1	Prakash	Neupane	praksh@gmail.com	9842512345

# Modify the following code to insert multiple records in to a table at once (output should be like in the given figure)



```
1 <?php
2 $servername = "localhost";
3 $username = "root";
4 $password = "";
5 $dbname = "WebII";
6 // Create connection
7 $conn = new mysqli($servername, $username, $password, $dbname);
8 // Check connection
9 if ($conn->connect_error) {
10     die("Connection failed: " . $conn->connect_error);
11 }
12 $sql = "INSERT INTO Student (firstname, lastname, email, phone)
13 VALUES ('Prakash', 'Neupane', 'praksh@gmail.com', 9842512345)";
14 if ($conn->query($sql) === TRUE) {
15     echo "New record created successfully";
16 } else {
17     echo "Error: " . $sql . "<br>" . $conn->error;
18 }
19 $conn->close();
20 ?>
```





# Use the concept of Prepared Statements and Bound Parameters for inserting records in database



- Prepared statements are very useful against SQL injections.

## Select/ retrieve Data from Database



- `SELECT * FROM table_name`
- `SELECT column_name(s) FROM table_name WHERE column_name operator value`
- `SELECT column_name(s) FROM table_name ORDER BY column_name(s) ASC|DESC`

# Select/ retrieve Data from Database



```
1 <?php
2 $servername = "localhost";
3 $username = "root";
4 $password = "";
5 $dbname = "WebII";
6 // Create connection
7 $conn = new mysqli($servername, $username, $password, $dbname);
8 // Check connection
9 if ($conn->connect_error) {
10     die("Connection failed: " . $conn->connect_error);
11 }
12 $sql = "SELECT sid, firstname, lastname, email, phone FROM Student";
13 $result = $conn->query($sql);
14 if ($result->num_rows > 0) {
15     // output data of each row
16     while($row = $result->fetch_assoc()) {
17         echo "Student_id: " . $row["sid"]. " Name: " . $row["firstname"]. " " . $row["lastname"]. " Email: " . $row["email"]. " Phone number: " . $row["phone"]. "<br>";
18     }
19 } else {
20     echo "0 results";
21 }
22 $conn->close();
23 ?>
```

Student\_id: 1 Name: Prakash Neupane Email: praksh@gmail.com Phone number: 9842512345  
Student\_id: 2 Name: Praash Nupane Email: praksh@gmail.com Phone number: 9842512345  
Student\_id: 3 Name: Sagar Gautam Email: sagar@gmail.com Phone number: 9842554321

## Select/ retrieve Data from Database



- Use the concepts of Where and order By clause to filter the records while retrieving records from table using php.

# Delete Data from Database



- The DELETE statement is used to delete records from a table:
  - DELETE FROM table\_name
  - WHERE some\_column = some\_value
- ***Notice the WHERE clause in the DELETE syntax: The WHERE clause specifies which record or records that should be deleted. If you omit the WHERE clause, all records will be deleted!***

# Delete Data from Database



```
1 <?php
2 $servername = "localhost";
3 $username = "root";
4 $password = "";
5 $dbname = "WebII";
6 // Create connection
7 $conn = new mysqli($servername, $username, $password, $dbname);
8 // Check connection
9 if ($conn->connect_error) {
10     die("Connection failed: " . $conn->connect_error);
11 }
12 // sql to delete a record
13 $sql = "DELETE FROM Student WHERE sid=3";
14 if ($conn->query($sql) === TRUE) {
15     echo "Record deleted successfully";
16 } else {
17     echo "Error deleting record: " . $conn->error;
18 }
19 $conn->close();
20 ?>
```

sid	firstname	lastname	email	phone
1	Prakash	Neupane	praksh@gmail.com	9842512345
2	Praash	Nupane	praksh@gmail.com	9842512345
3	Sagar	Gautam	sagar@gmail.com	9842554321

sid	firstname	lastname	email	phone
1	Prakash	Neupane	praksh@gmail.com	9842512345
2	Praash	Nupane	praksh@gmail.com	9842512345

# Update Data in Database



- The UPDATE statement is used to update existing records in a table:
  - UPDATE table\_name
  - SET column1=value, column2=value2,...
  - WHERE some\_column=some\_value

# Update Data in Database



- The UPDATE statement is used to update existing records in a table:
  - UPDATE table\_name
  - SET column1=value, column2=value2,...
  - WHERE some\_column=some\_value
- ***Notice the WHERE clause in the UPDATE syntax: The WHERE clause specifies which record or records that should be updated. If you omit the WHERE clause, all records will be updated!***



# Update Data in Database

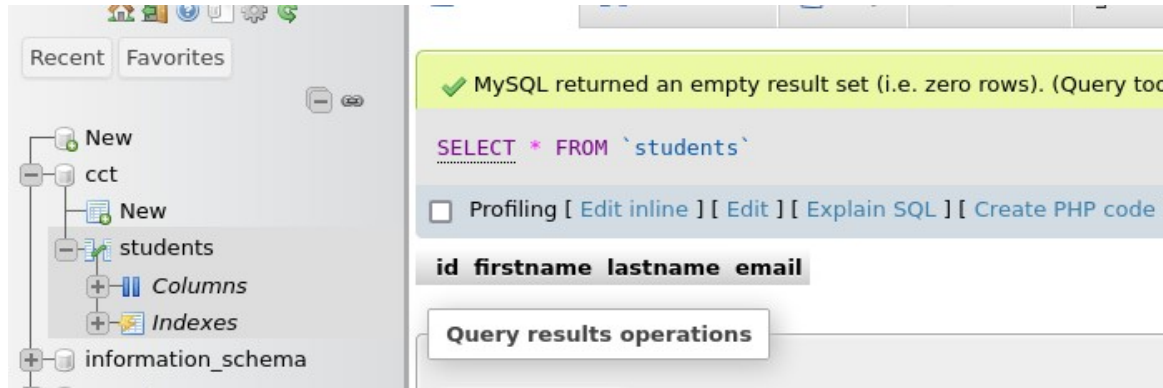


```
1 <?php
2 $servername = "localhost";
3 $username = "root";
4 $password = "";
5 $dbname = "WebII";
6 // Create connection
7 $conn = new mysqli($servername, $username, $password, $dbname);
8 // Check connection
9 if ($conn->connect_error) {
10     die("Connection failed: " . $conn->connect_error);
11 }
12 $sql = "UPDATE Student SET firstname ='Sagar', lastname='Gautam' WHERE sid=2";
13 if ($conn->query($sql) === TRUE) {
14     echo "Record updated successfully";
15 } else {
16     echo "Error updating record: " . $conn->error;
17 }
18 $conn->close();
19 ?>
```

sid	firstname	lastname	email	phone
1	Prakash	Neupane	praksh@gmail.com	9842512345
2	Praash	Nupane	praksh@gmail.com	9842512345

▼ sid	firstname	lastname	email	phone
te 1	Prakash	Neupane	praksh@gmail.com	9842512345
te 2	Sagar	Gautam	praksh@gmail.com	9842512345

# Create/Insert data into table using form

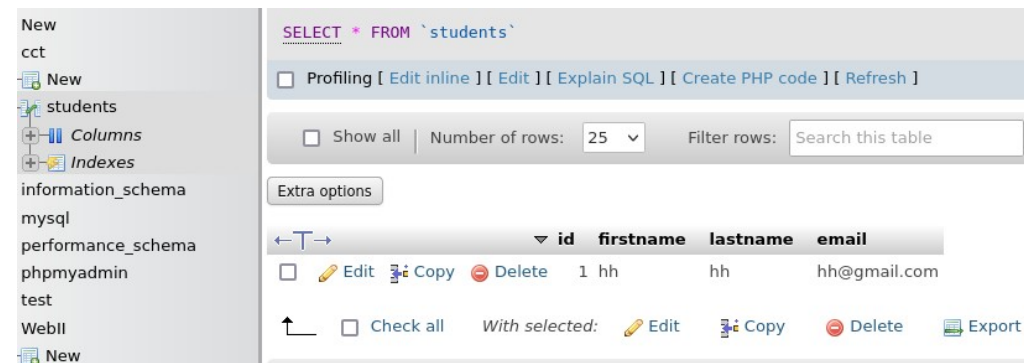


## Create Student Record

First Name:

Last Name:

Email:



# Create/Insert data into table using form



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Create Student Record</title>
7 </head>
8 <body>
9   <h2>Create Student Record</h2>
10  <form method="post" action="insert_cct_students.php">
11    <label for="firstname">First Name:</label>
12    <input type="text" name="firstname" required><br>
13    <label for="lastname">Last Name:</label>
14    <input type="text" name="lastname" required><br>
15    <label for="email">Email:</label>
16    <input type="email" name="email" required><br>
17    <input type="submit" value="Create/Insert Record">
18  </form>
19 </body>
20 </html>
21
```

```
1 //insert_cct_students.php
2 <?php
3 $servername = "localhost";
4 $username = "root";
5 $password = "";
6 $dbname = "cct";
7 $conn = new mysqli($servername, $username, $password, $dbname);
8 if ($conn->connect_error) {
9   die("Connection failed: " . $conn->connect_error);
10 }
11 if ($_SERVER["REQUEST_METHOD"] == "POST") {
12   $firstname = $_POST["firstname"];
13   $lastname = $_POST["lastname"];
14   $email = $_POST["email"];
15   $sql = "INSERT INTO students (firstname, lastname, email) VALUES ('$firstname',
16     '$lastname', '$email')";
17   if ($conn->query($sql) === TRUE) {
18     echo "Record created successfully";
19   } else {
20     echo "Error: " . $sql . "<br>" . $conn->error;
21   }
22 $conn->close();
23 ?>
```

# Retrieve data from database in tabular form



Left sidebar (Database Structure):

- New
- cct
- students
  - Columns
  - Indexes
- information\_schema
- mysql
- performance\_schema
- phpmyadmin
- test
- Webli
- New

SQL Query:

```
SELECT * FROM `students`
```

Buttons: ☐ Profiling [ [Edit inline](#) ] [ [Edit](#) ] [ [Explain SQL](#) ] [ [Create PHP code](#) ] [ [Refresh](#) ]

Controls:

- ☐ Show all
- Number of rows: 25
- Filter rows:

Extra options

	id	firstname	lastname	email
<input type="checkbox"/>	1	hh	hh	hh@gmail.com

Actions:

- ☐ Check all
- With selected: [Edit](#) [Copy](#) [Delete](#) [Export](#)

## Student Records

ID	First Name	Last Name	Email
1	hh	hh	hh@gmail.com

# Retrieve data from database in tabular form



```
1 <?php
2 $servername = "localhost";
3 $username = "root";
4 $password = "";
5 $dbname = "cct";
6 $conn = new mysqli($servername, $username,
7 $password, $dbname);
8 if ($conn->connect_error) {
9     die("Connection failed: " . $conn->connect_error);
10 }
11 $sql = "SELECT * FROM students";
12 $result = $conn->query($sql);
13 ?>
```

```
1 <h2>Student Records</h2>
2 <table border="1">
3     <tr>
4         <th>ID</th>
5         <th>First Name</th>
6         <th>Last Name</th>
7         <th>Email</th>
8     </tr>
9 <?php
10 if ($result->num_rows > 0) {
11     while($row = $result->fetch_assoc()) {
12         echo "<tr>";
13         echo "<td>" . $row["id"] . "</td>";
14         echo "<td>" . $row["firstname"] . "</td>";
15         echo "<td>" . $row["lastname"] . "</td>";
16         echo "<td>" . $row["email"] . "</td>";
17         echo "</tr>";
18     }
19 } else {
20     echo "<tr><td colspan='4'>No records found</td></tr>";
21 }
22 ?>
23 </table>
```

## Update and delete data using form



Refer: [https://github.com/neupaneprakash/webTech\\_II/blob/main/Crud.zip](https://github.com/neupaneprakash/webTech_II/blob/main/Crud.zip)

## Update and delete data using form



Refer: [https://github.com/neupaneprakash/webTech\\_II/blob/main/Crud.zip](https://github.com/neupaneprakash/webTech_II/blob/main/Crud.zip)