1420-7001

웹 프로그래밍

Web Programming

By

Dr. Abdul Majeed (조교수)

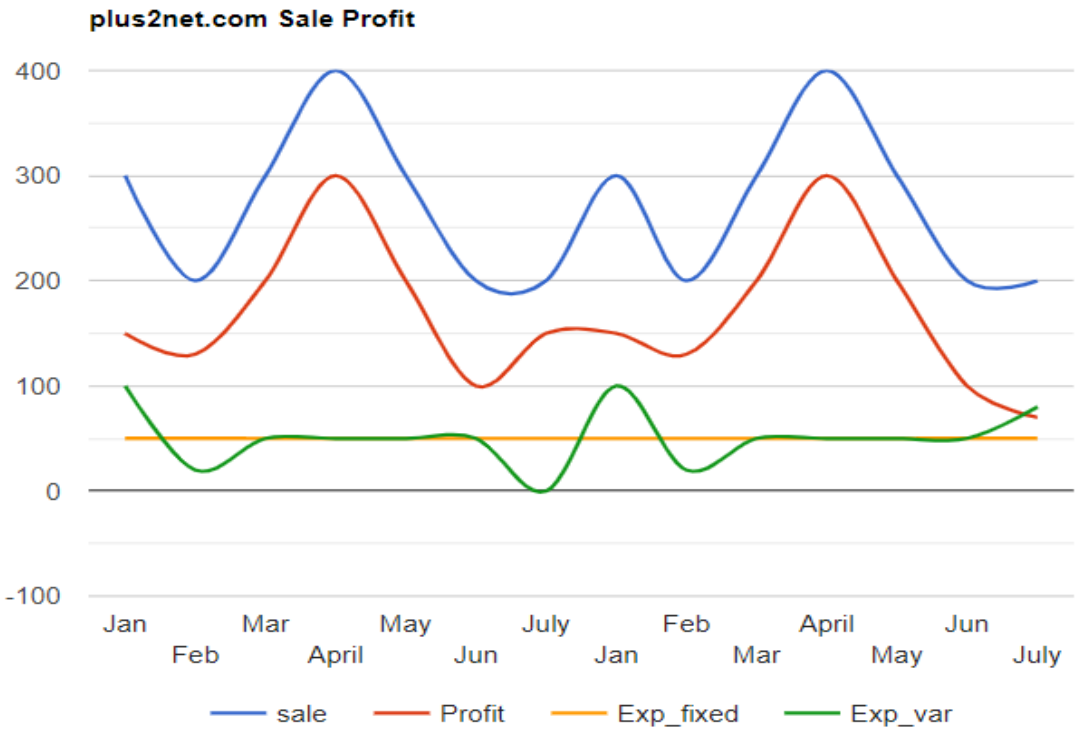
1st Semester, 2025

가천대학교
Gachon University

IT융합대학 컴퓨터공학부(컴퓨터공학전공)

Summary of the Today's Lesson

- Dynamic data visualizations
  - Database setup
  - PHP code setup
  - Front-end code setup
  - Display a bar graph in the Browser
- Dynamic data visualizations
  - Area graph
  - Line graph
  - Tools & libraries
  - Four examples
- Data visualization by reading data from a file
- Form data insertion to the DB [Next Class]

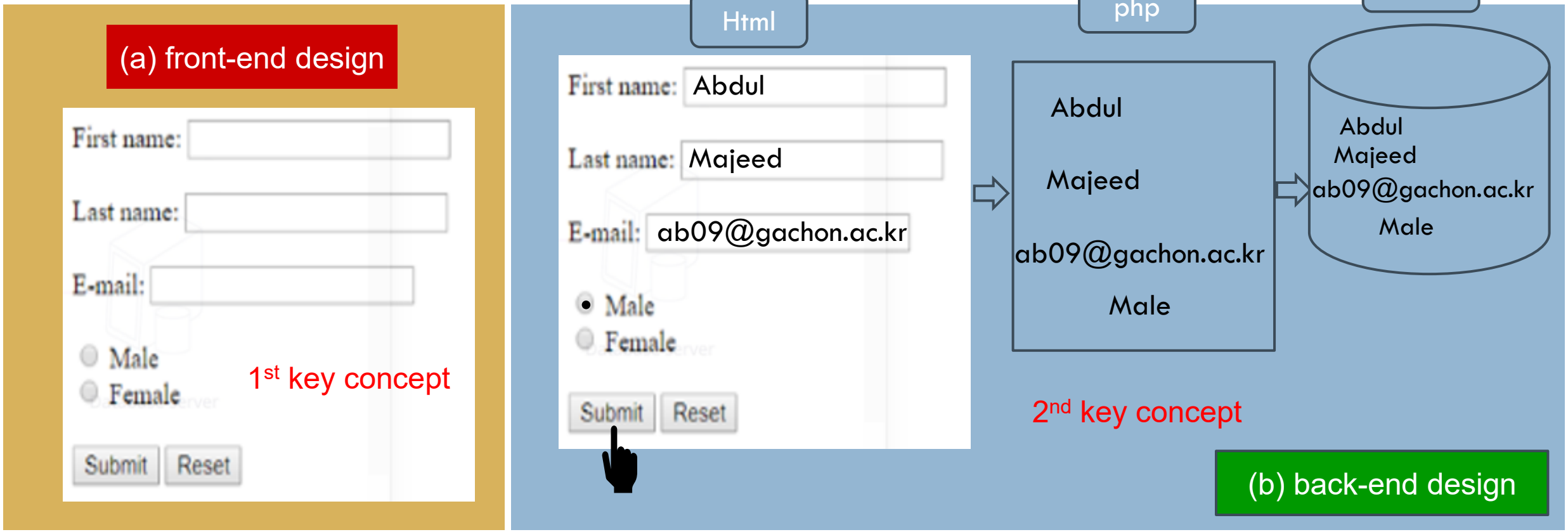


Part-I

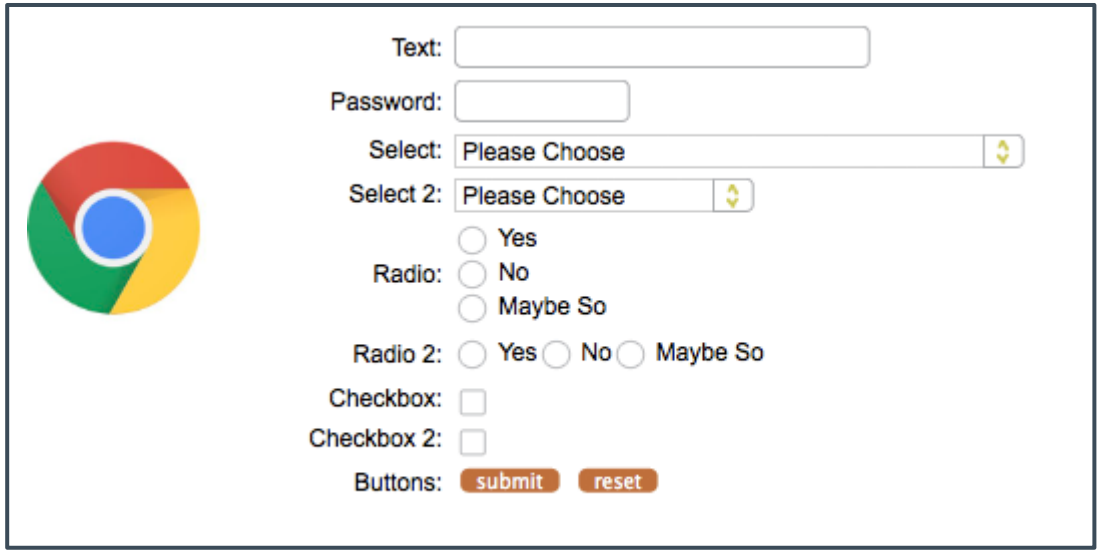
Inserting Data into DB

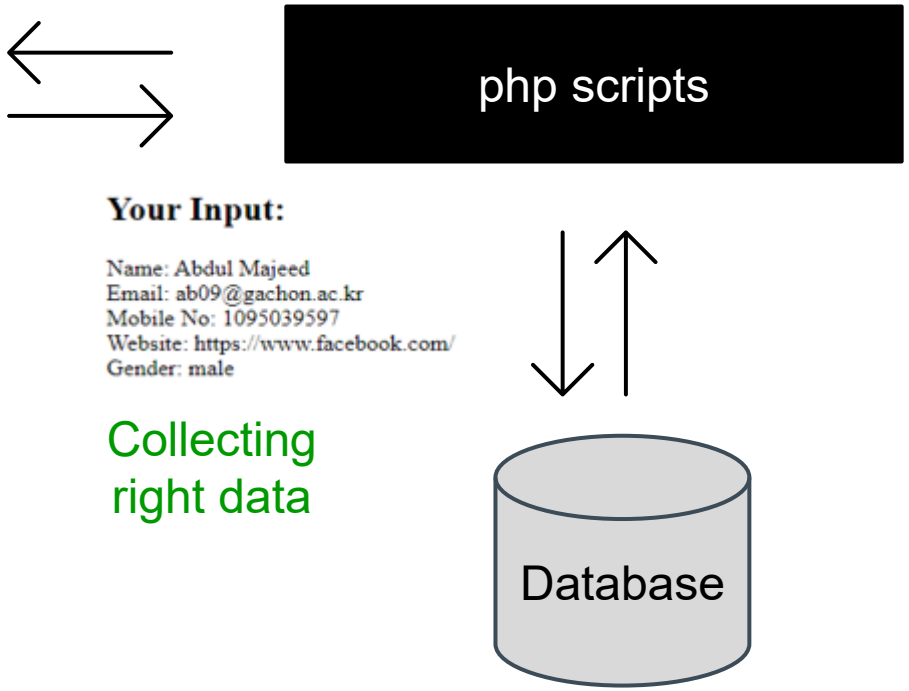
Focus of the Today's Class-Overall Goal

- 2nd major objective of this course
- Designing the front side of data collection
- Backend side will be covered in last session of the class

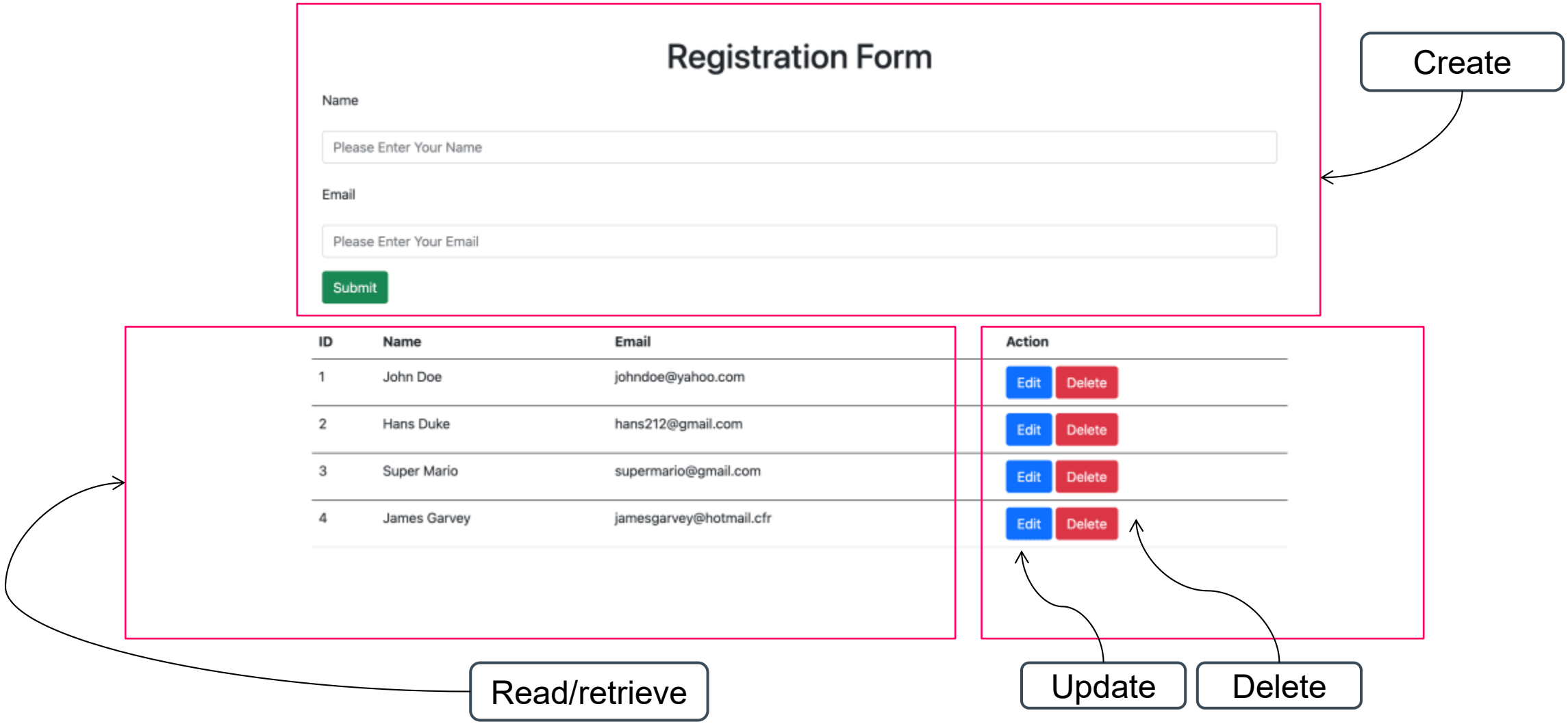


2nd Major Goal-{Data Storage}





Data Insertion- Easy Case [one item at a time]



Example 1

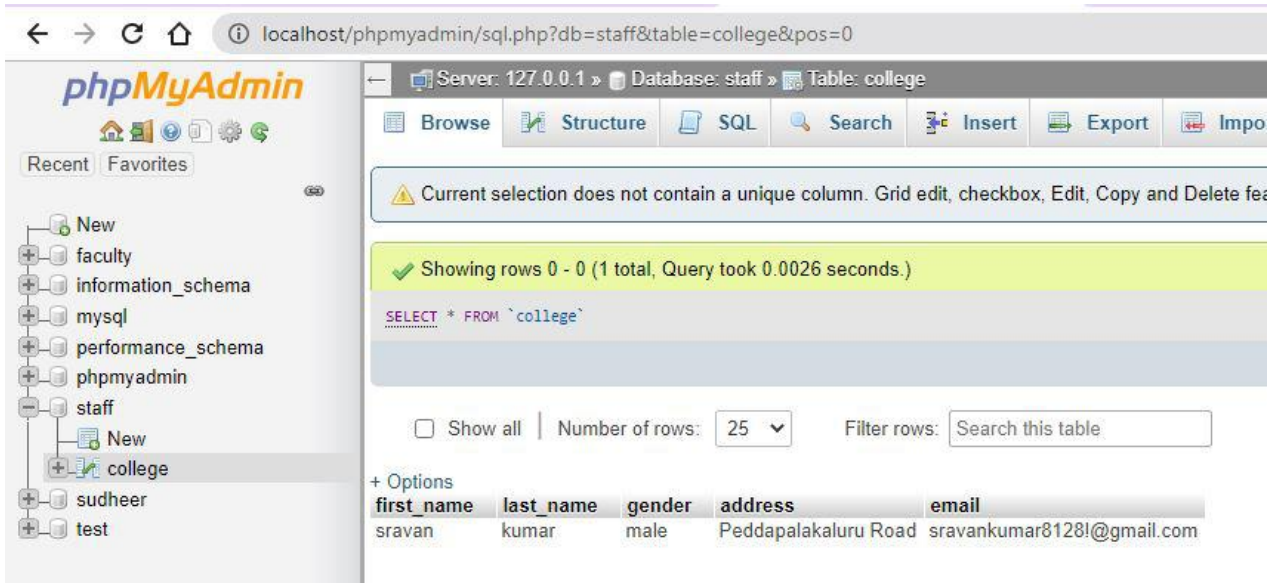
Form Data Insertion to XAMP Database

- Setting up database
  - Database creation
  - Table creation
    - # of columns
    - Label and datatypes of columns
- Designing server-side code
- Designing client-side code

https://www.geeksforgeeks.org/how-to-insert-form-data-into-database-using-php/

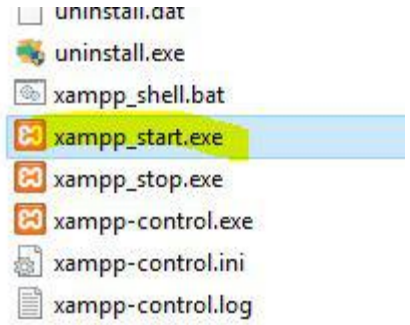
Overall Process of data Insertion

Data in destination



Data insertion to DB- Database Setup

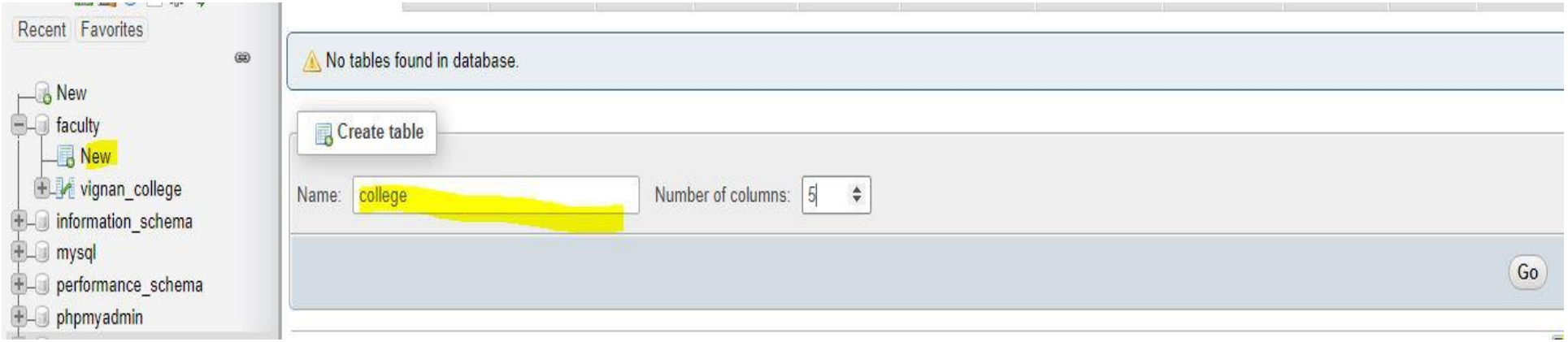
⌘ Open XAMPP to start the XAMPP server and click Start XAMPP.



⌘ Open ***localhost/phpmyadmin in your web browser,*** name the database **staff** and click 'Create'.

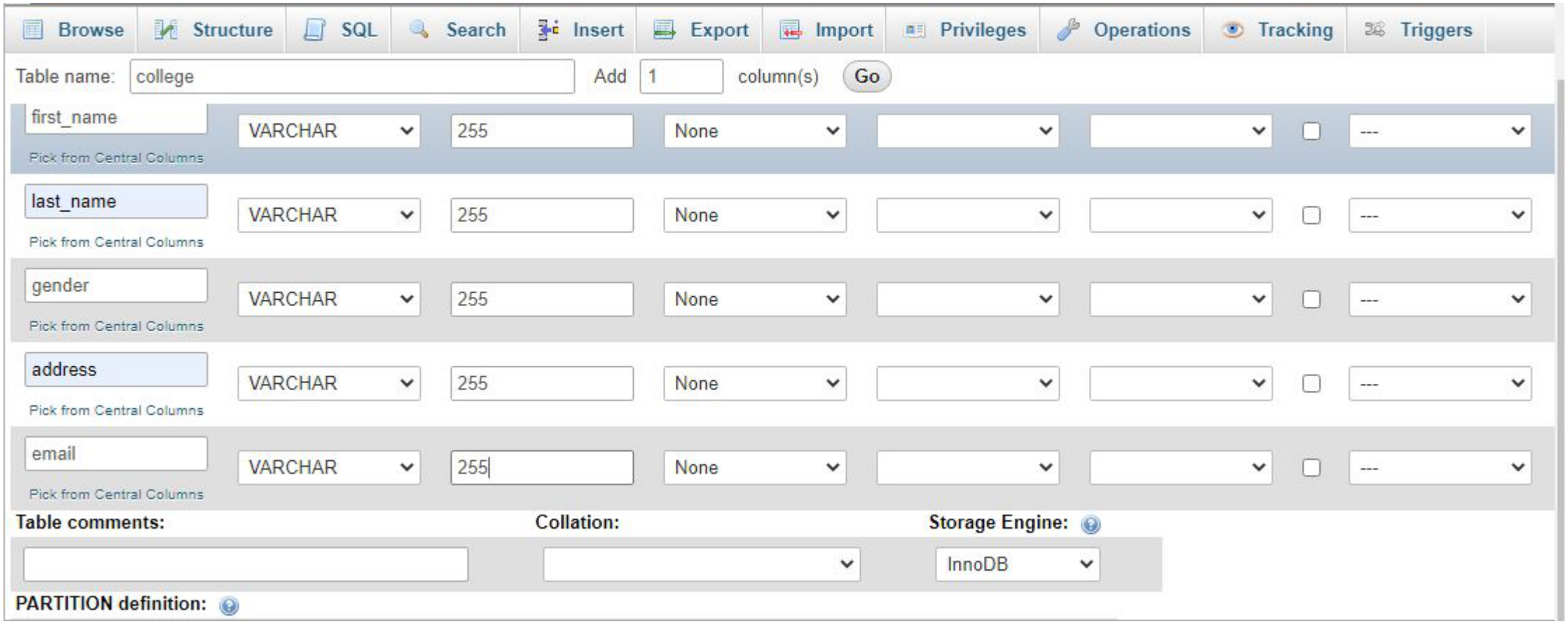
Data insertion to DB- Database Setup

- Then create a table named college .



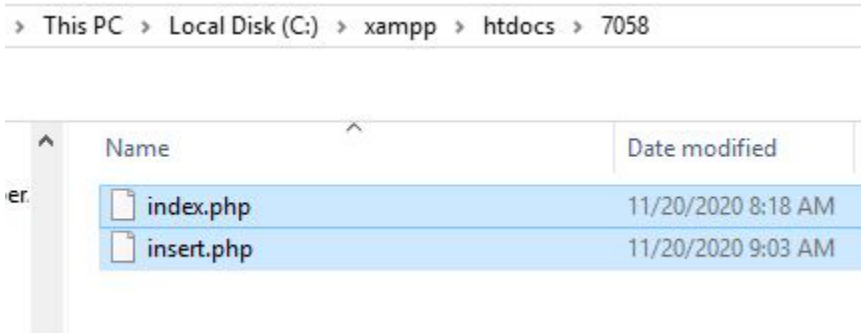
Data insertion to DB- Database Setup

⌘ Enter the columns and click Save.



Data insertion to DB- Client Side & Server-Side Code

- Now open a notepad and write your PHP code and save it as index.php, then open another notepad and save it as insert.php. Save both files in a folder named htdocs .



Data insertion to DB- Client Side

⌘ This is the client side code.

```
<!DOCTYPE html>
<html lang="en">
   <head>
      <title>GFG- Store Data</title>
   </head>
   <body>
      <center>
         <h1>Storing Form data in Database</h1>
         <form action="insert.php" method="post">

<p>
               <label for="firstName">First Name:</label>
               <input type="text" name="first_name" id="firstName">
            </p>


<p>
               <label for="lastName">Last Name:</label>
               <input type="text" name="last_name" id="lastName">
            </p>


<p>
               <label for="Gender">Gender:</label>
               <input type="text" name="gender" id="Gender">
            </p>


<p>
               <label for="Address">Address:</label>
               <input type="text" name="address" id="Address">
            </p>


<p>
               <label for="emailAddress">Email Address:</label>
               <input type="text" name="email" id="emailAddress">
            </p>

            <input type="submit" value="Submit">
         </form>
      </center>
   </body>
</html>
```

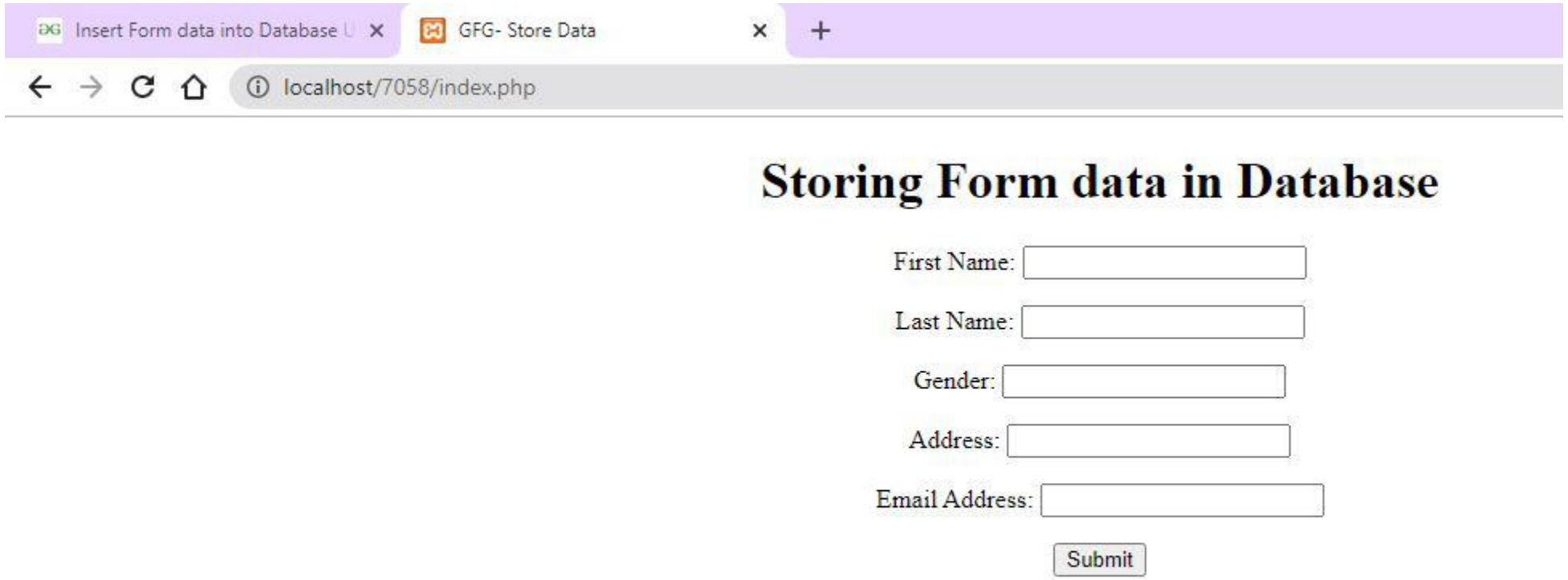
Data insertion to DB-Server Side

- This is the server-side code.

```php
<!DOCTYPE html>
<html>

<head>
    <title>Insert Page page</title>
</head>

<body>
    <center>
        <?php

        // servername => localhost
        // username => root
        // password => empty
        // database name => staff
        $conn = mysqli_connect("localhost", "root", "", "staff");

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

        // Taking all 5 values from the form data(input)
        $first_name =  $_REQUEST['first_name'];
        $last_name = $_REQUEST['last_name'];
        $gender =  $_REQUEST['gender'];
        $address = $_REQUEST['address'];
        $email = $_REQUEST['email'];

        // Performing insert query execution
        // here our table name is college
        $sql = "INSERT INTO college  VALUES ('$first_name',
            '$last_name','$gender','$address','$email')";

        if(mysqli_query($conn, $sql)){
            echo "<h3>data stored in a database successfully."
                . " Please browse your localhost php my admin"
                . " to view the updated data</h3>";

            echo nl2br("\n$first_name\n $last_name\n "
                . "$gender\n $address\n $email");
        } else{
            echo "ERROR: Hush! Sorry $sql. "
                . mysqli_error($conn);
        }

        // Close connection
        mysqli_close($conn);
        ?>
    </center>
</body>

</html>
```

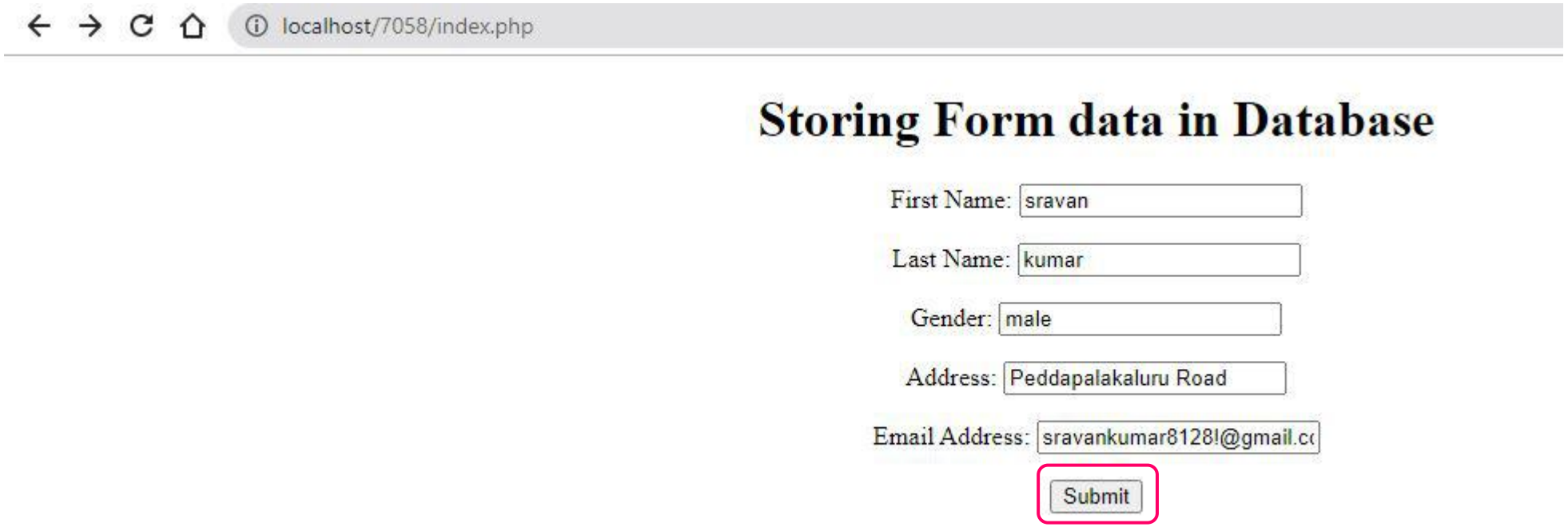
Data insertion to DB-Run Code

When you type ***로컬호스트/7058/index.php*** in your browser , the form will be displayed. After you submit the form, the form data will be submitted to the database.



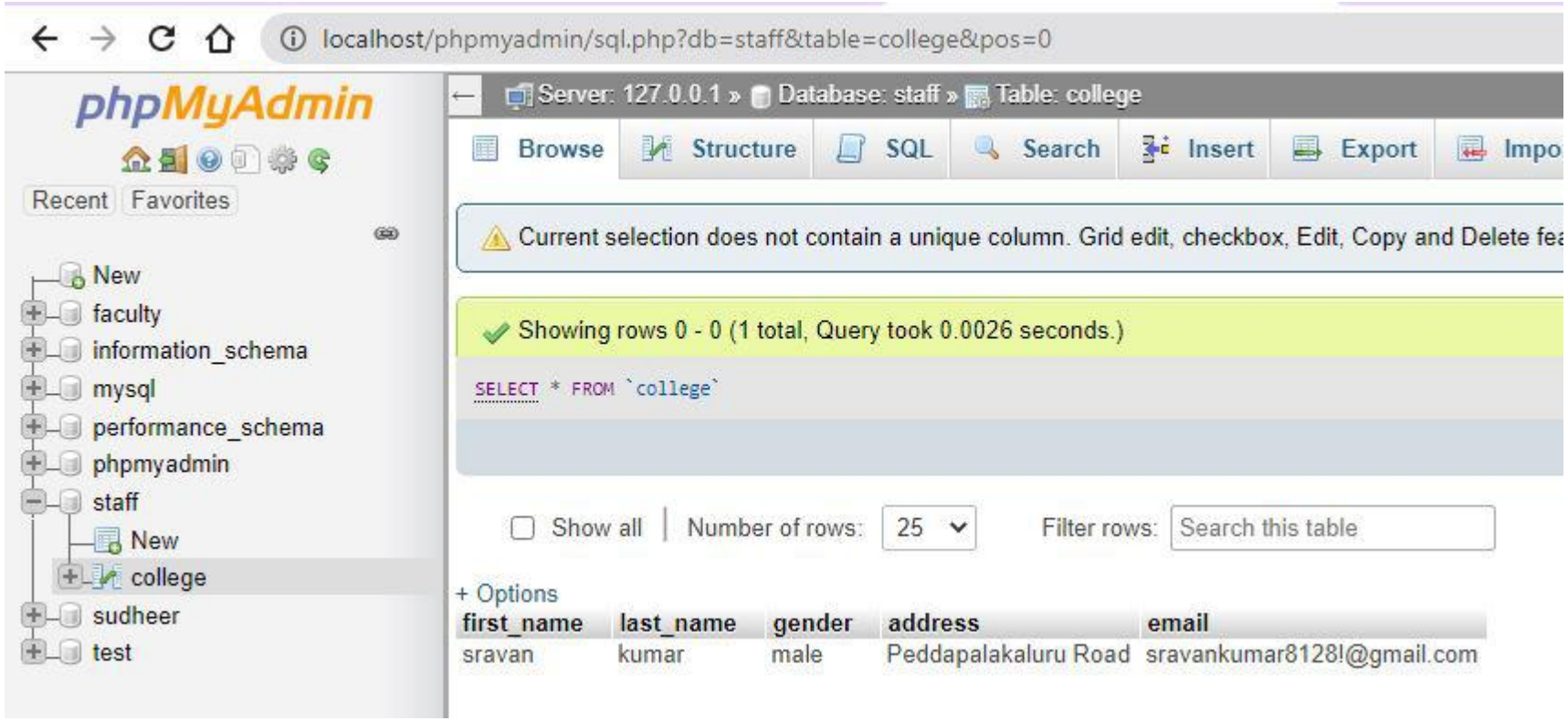
Data insertion to DB-Run Code

Fill the relevant fields with your data & click submit.



Data insertion to DB-Run Code

The data will be added into the xamp database.



Data insertion to DB-Putting everything together

```html
<!DOCTYPE html>
<html lang="en">
   <head>
      <title>GFG- Store Data</title>
   </head>
   <body>
      <center>
         <h1>Storing Form data in Database</h1>
         <form action="insert.php" method="post">

<p>
               <label for="firstName">First Name:</label>
               <input type="text" name="first_name" id="firstName">
            </p>


<p>
               <label for="lastName">Last Name:</label>
               <input type="text" name="last_name" id="lastName">
            </p>


<p>
               <label for="Gender">Gender:</label>
               <input type="text" name="gender" id="Gender">
            </p>


<p>
               <label for="Address">Address:</label>
               <input type="text" name="address" id="Address">
            </p>


<p>
               <label for="emailAddress">Email Address:</label>
               <input type="text" name="email" id="emailAddress">
            </p>

            <input type="submit" value="Submit">
         </form>
      </center>
   </body>
</html>
```

```php
<!DOCTYPE html>
<html>

<head>
    <title>Insert Page page</title>
</head>

<body>
    <center>
        <?php

        // servername => localhost
        // username => root
        // password => empty
        // database name => staff
        $conn = mysqli_connect("localhost", "root", "", "staff");

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

        // Taking all 5 values from the form data(input)
        $first_name =  $_REQUEST['first_name'];
        $last_name = $_REQUEST['last_name'];
        $gender =  $_REQUEST['gender'];
        $address = $_REQUEST['address'];
        $email = $_REQUEST['email'];

        // Performing insert query execution
        // here our table name is college
        $sql = "INSERT INTO college  VALUES ('$first_name',
            '$last_name','$gender','$address','$email')";

        if(mysqli_query($conn, $sql)){
            echo "<h3>data stored in a database successfully."
                . " Please browse your localhost php my admin"
                . " to view the updated data</h3>";

            echo nl2br("\n$first_name\n $last_name\n "
                . "$gender\n $address\n $email");
        } else{
            echo "ERROR: Hush! Sorry $sql. "
                . mysqli_error($conn);
        }

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

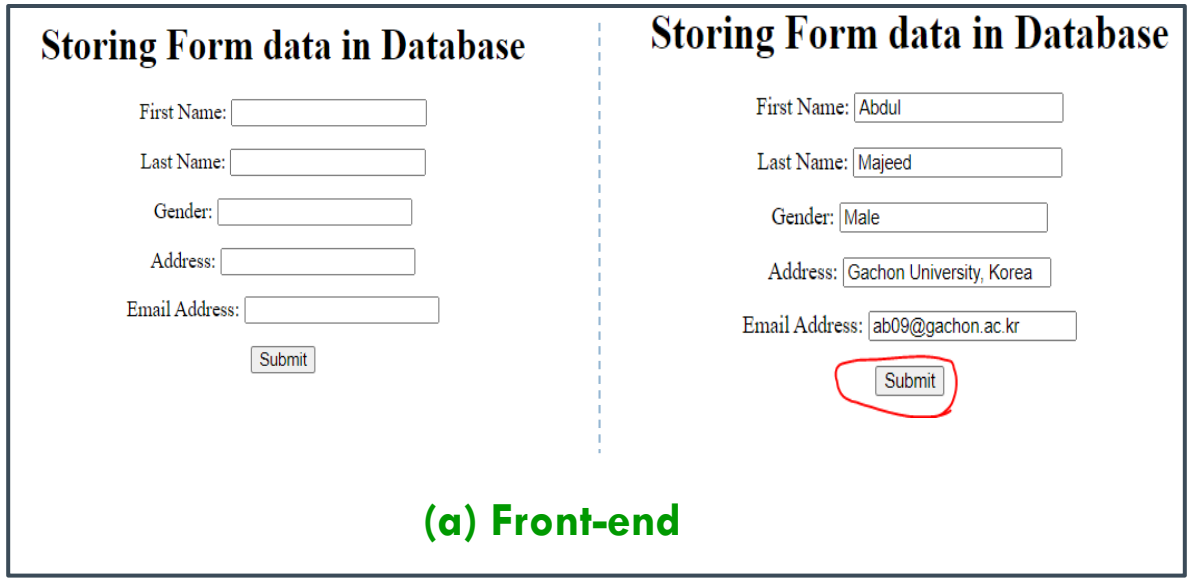
Front-end

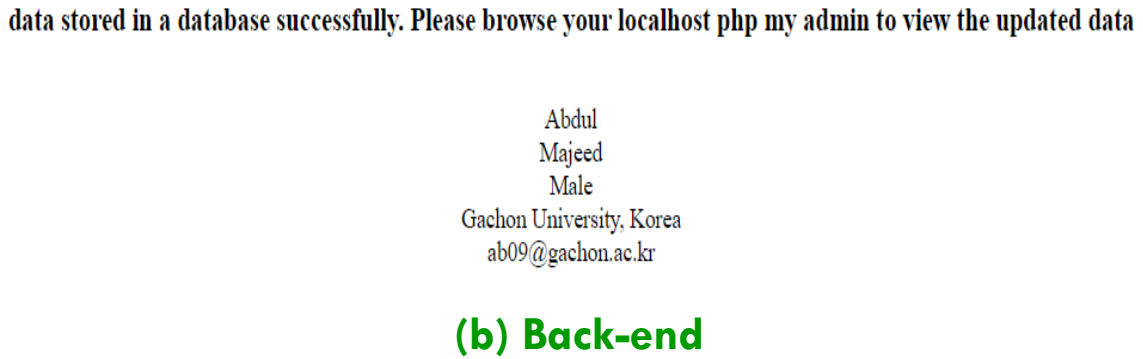
Back-end

Database

first_name	last_name	gender	address	email
sravan	kumar	male	Peddapalakaluru Road	sravankumar8128l@gmail.com

Data insertion to DB-Putting everything together



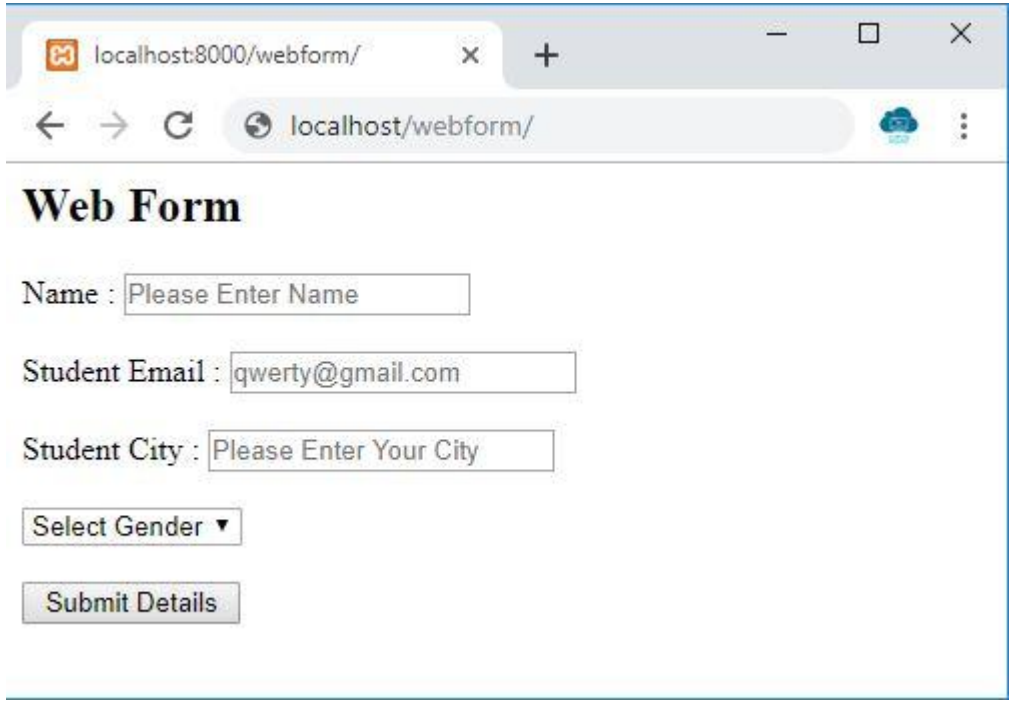


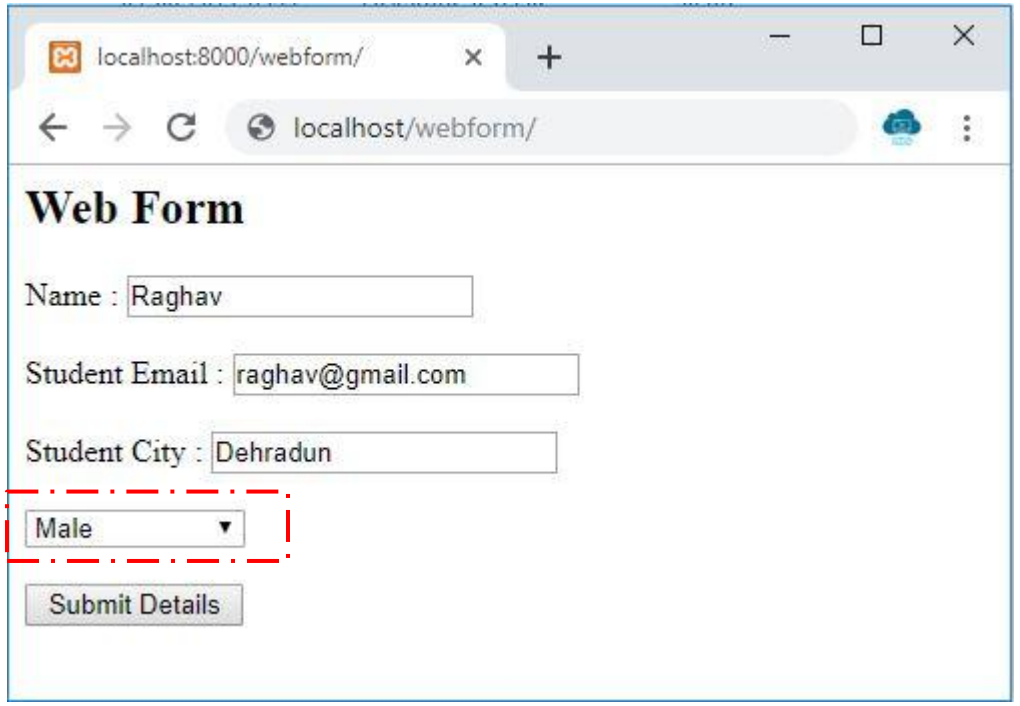
(c) Database

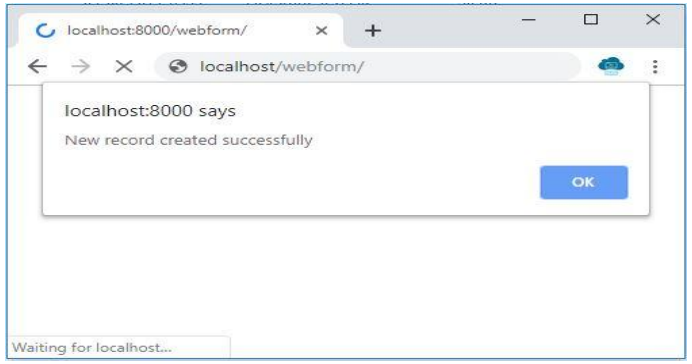
first_name	last_name	gender	address	email
Abdul	Majeed	Male	Gachon University, Korea	ab09@gachon.ac.kr

Example 2

Data Insertion- Complex Case

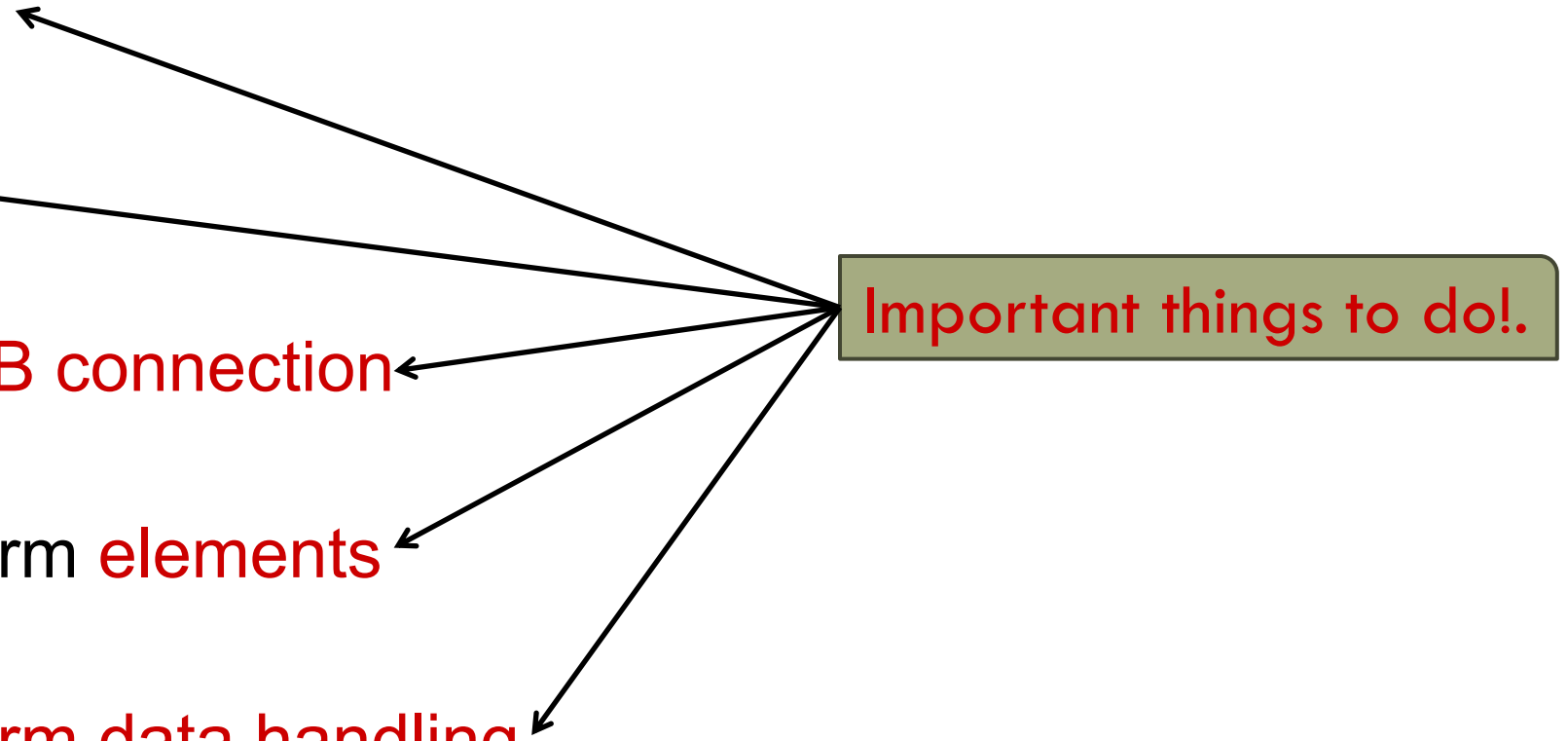






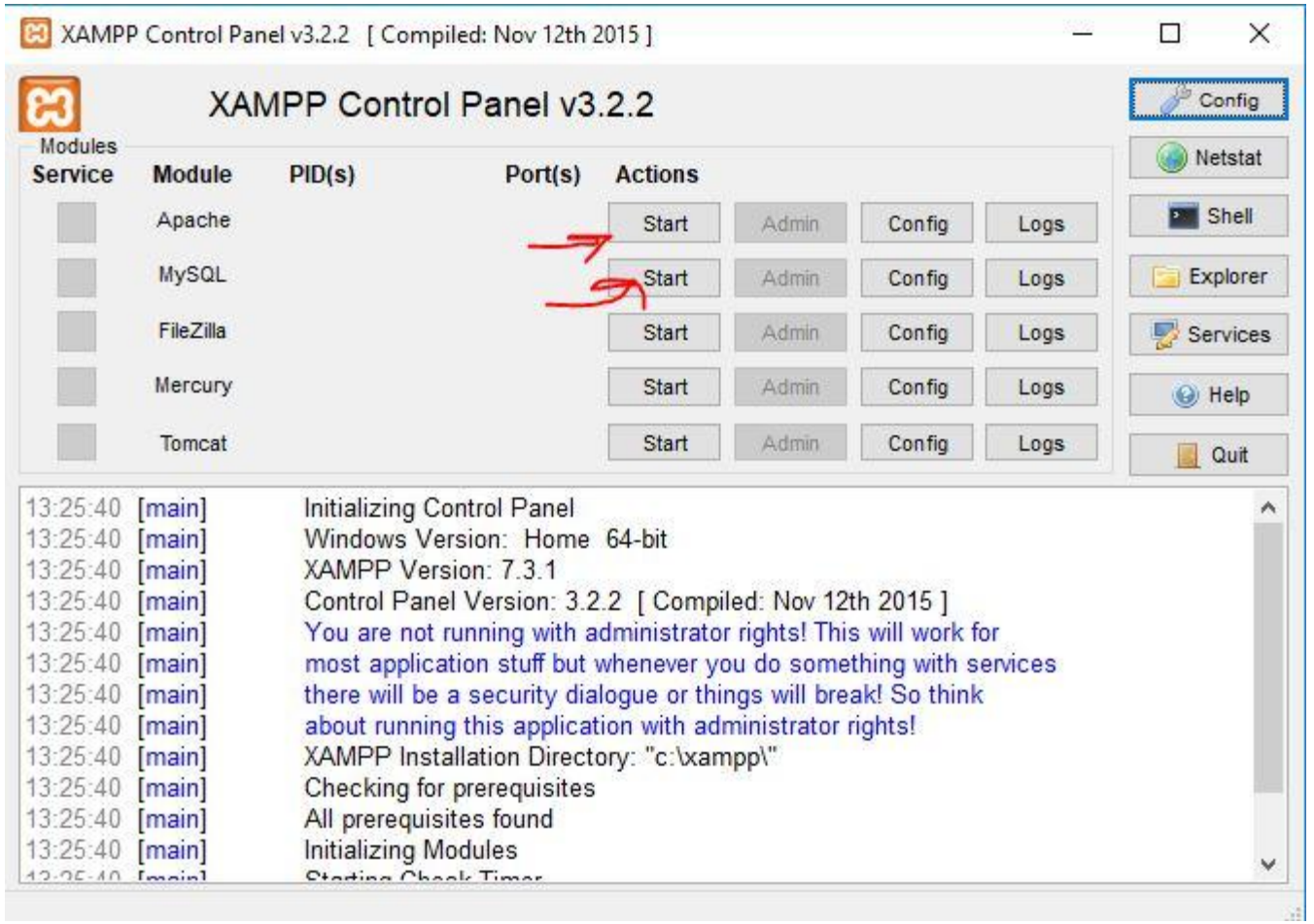
Steps to Perform Data Insertion

1) Create database
2) Create table
3) Write code for DB connection
4) Write code for form elements
5) Write code for form data handling



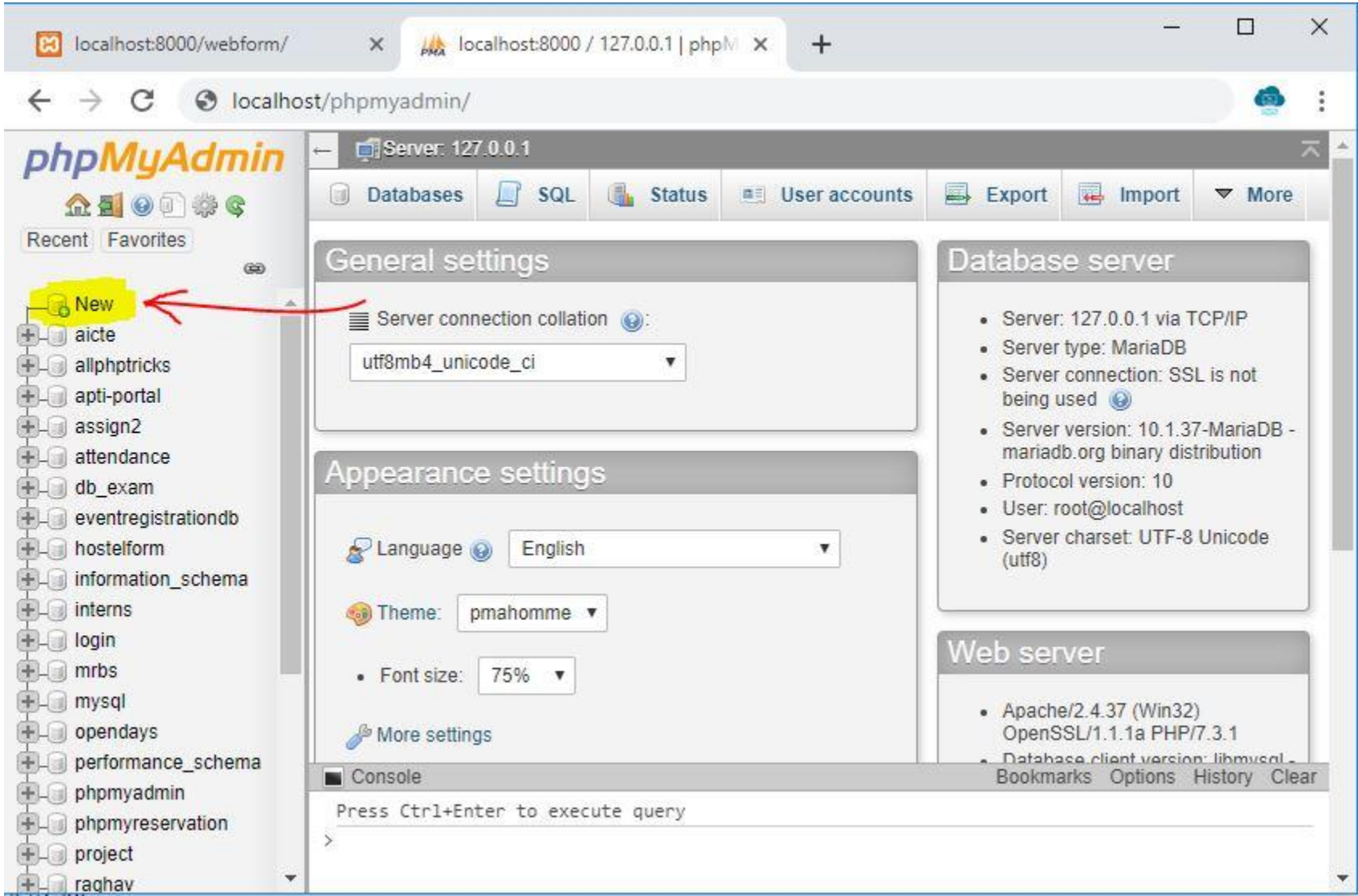
Step 1: Creating Database

- ✓ Start the Apache and MySQL servers in XAMPP.



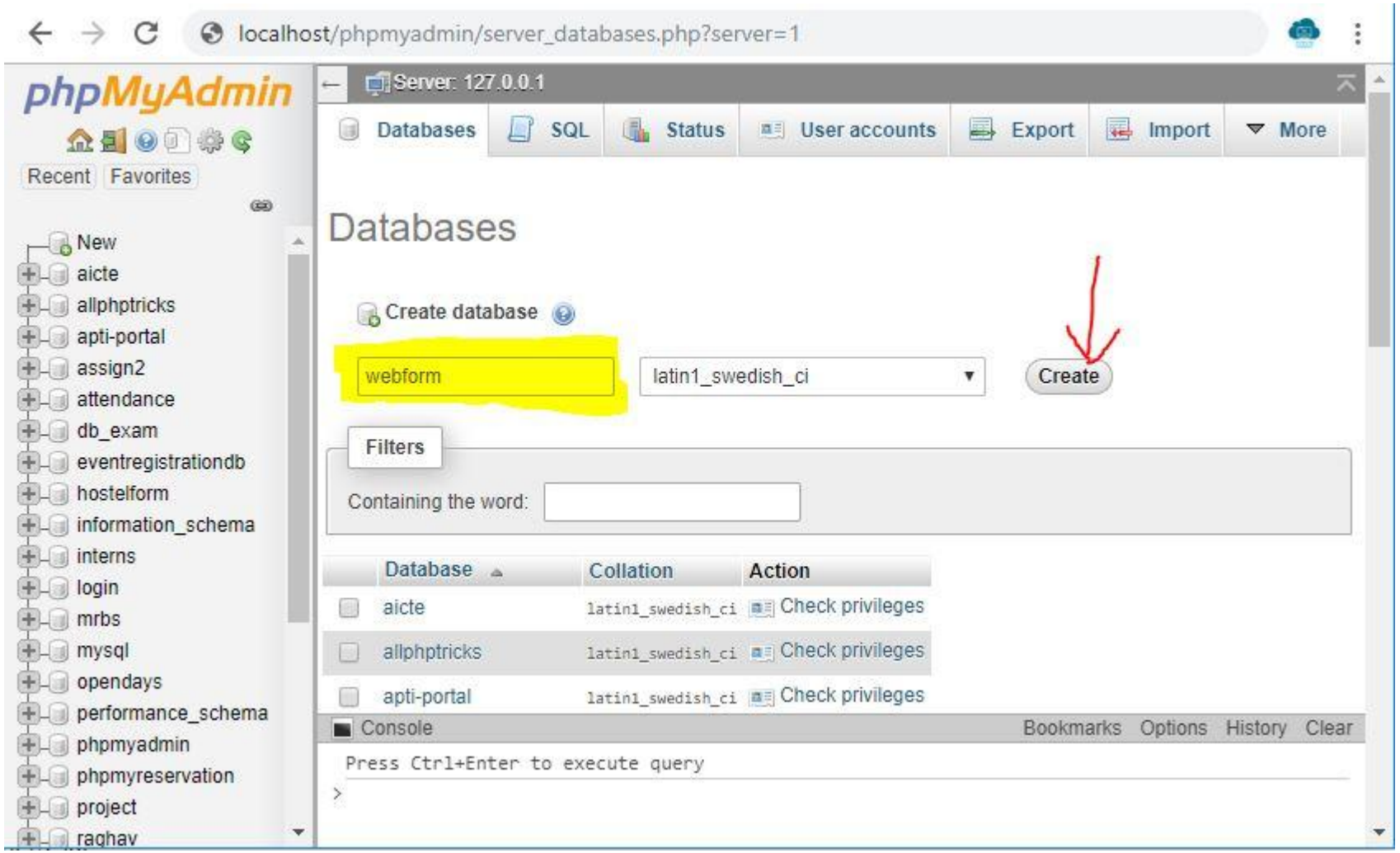
Step 1: Creating Database

- Now, go to your browser and open http://localhost/phpmyadmin
- In the left, go to New



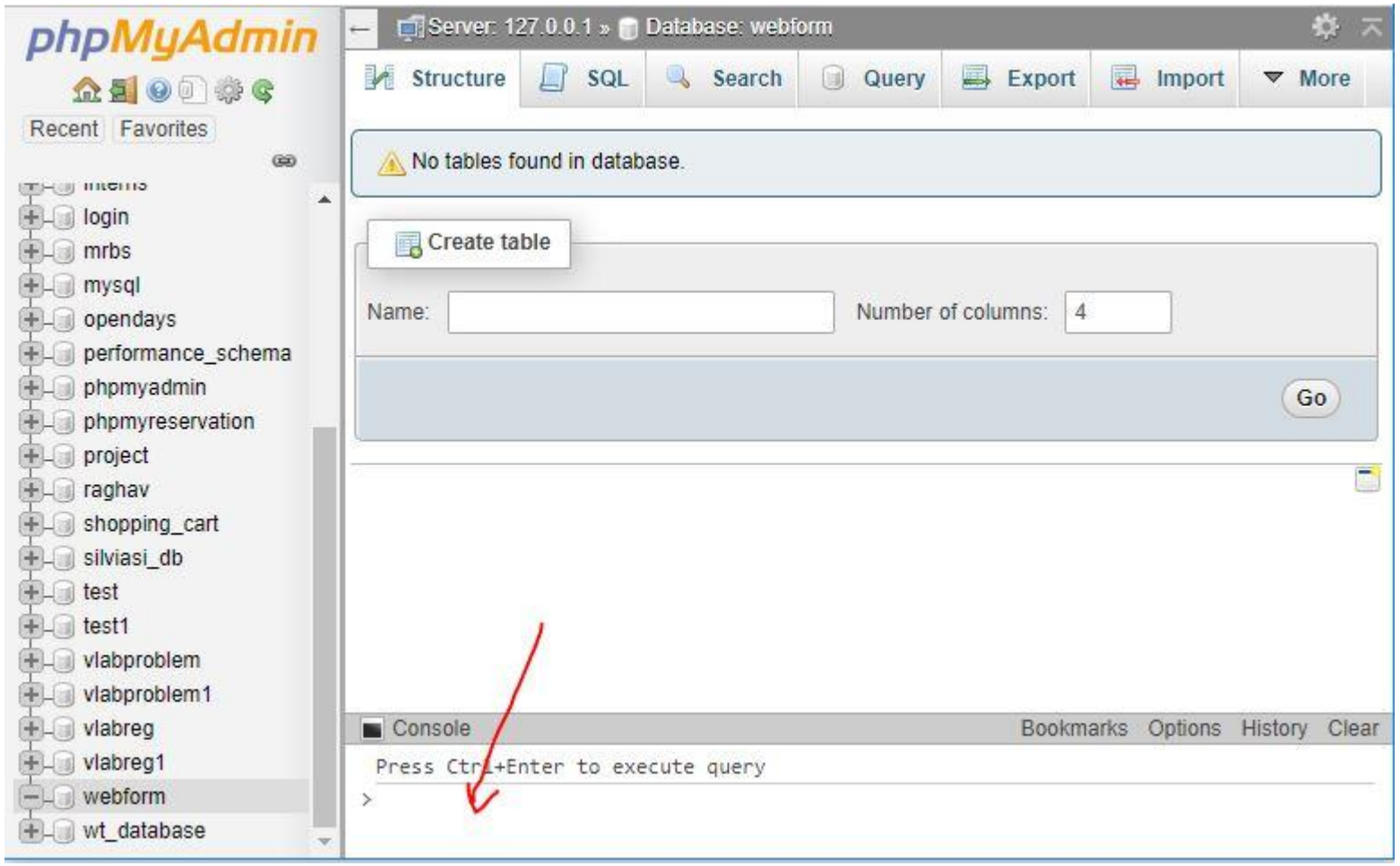
Step 1: Creating Database

- Name the database **webform.**



Step 1: Creating Database

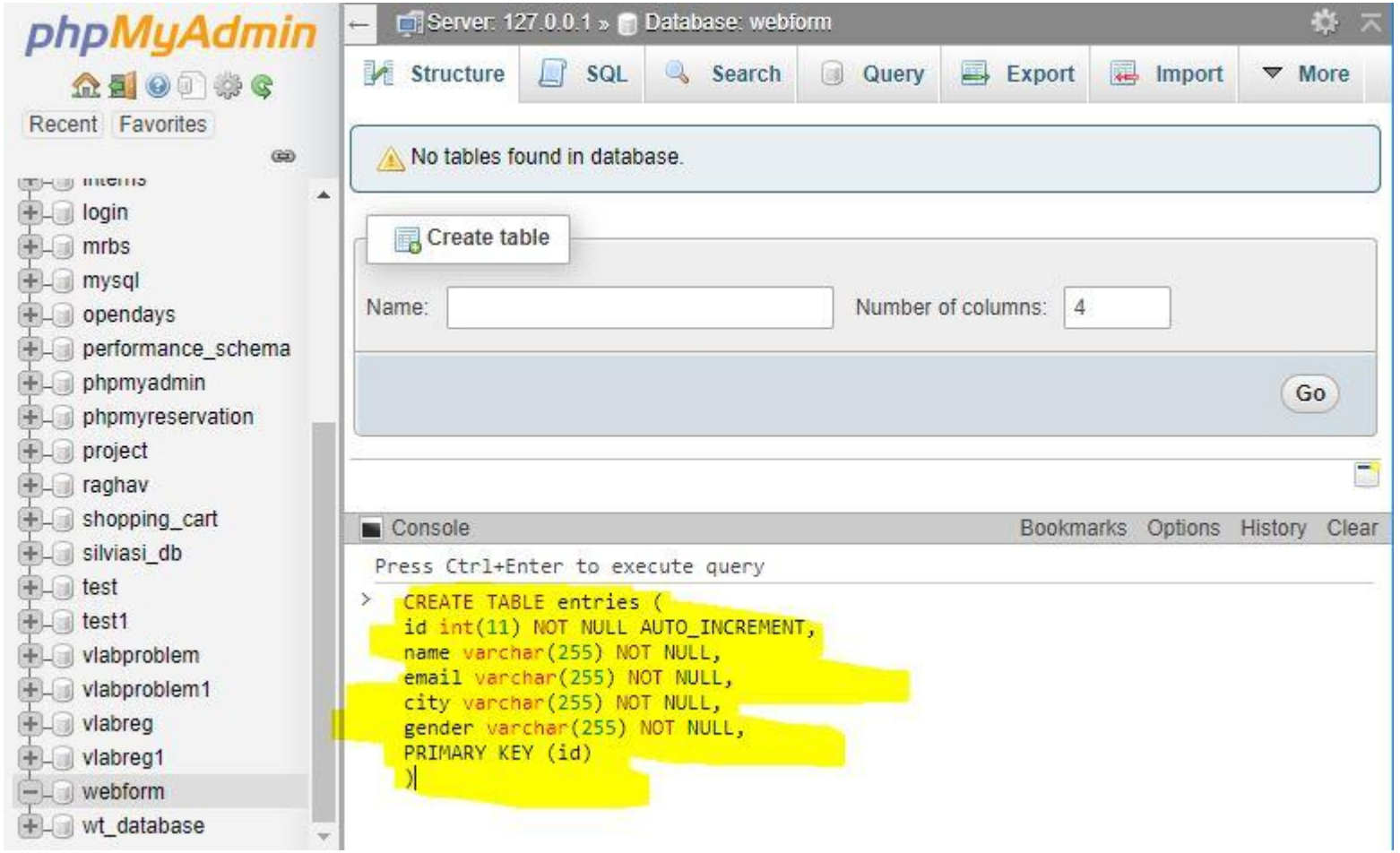
•You can see the **webform database now in your XAMP.**



Step 2: Creating Table in a Database

- Create the table with names entries in XAMP.

```
CREATE TABLE entries
( id int(11) NOT NULL
AUTO_INCREMENT,
 name varchar(255) NOT NULL,
 email varchar(255) NOT NULL,
city varchar(255) NOT NULL,
gender varchar(255) NOT NULL,
PRIMARY KEY (id) )
```



Step 3: Writing Code to link application with DB

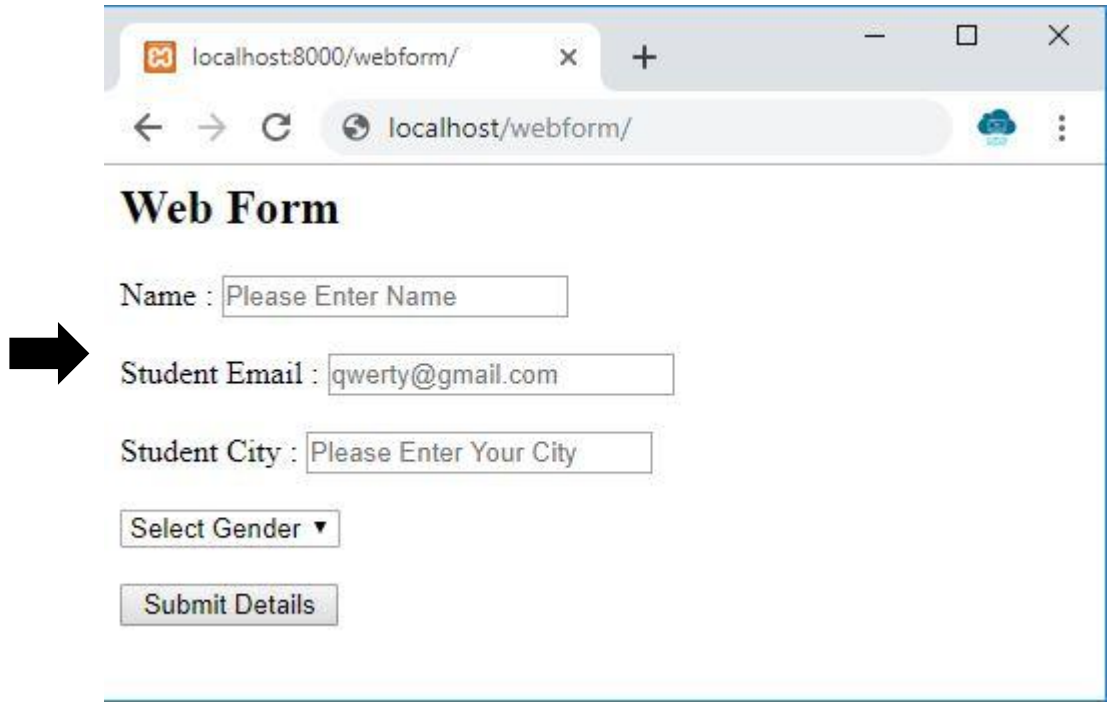
- Configuration file to connect with the database.

```php
//Connecting to the database
<?php
$servername = "localhost";
$username = "root"; //edit if you have set a username for MySQL
$password = ""; // edit if you have set a password
$dbname = "webform";

// Create connection syntax
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
?>
```

Step 4: Writing Code to link application with DB

- Creating form to acquire data from the users.

```
<div id="webform">
<h2>Web Form</h2>
<form action="" method="post">
<label>Name :</label>
<input type="text" name="name" required placeholder="Please Enter Name"/><br><br>
<label>Student Email :</label>
<input type="email" name="email" required placeholder="qwerty@gmail.com"/><br><br>
<label>Student City :</label>
<input type="text" name="city" required placeholder="Please Enter Your City"/><br><br>
<select name="gender" required>
    <option value=""> Select Gender </option>
    <option value="M"> Male </option>
    <option value="F"> Female </option>
</select>
<br><br>
<input type="submit" value=" Submit Details " name="submit"/><br />
</form>
</div>
```



Step 5: Writing Code to Handle forms data

- Till now, we have written the code for the form. Now, we will come to the main part, which is handling the date entered into this form.

- In the above code, we have set the method of the form to **post,** and we have left the action attribute empty. By doing this, the form will post the data on the same page. If we had written the address to some other file/link in the action attribute, the form would have posted data to that particular address. I hope this makes sense.

- Now take a close look at the following code

```php
<?php
if(isset($_POST["submit"])){
include 'dbconfig.php';

$sql = "INSERT INTO entries (name, email, city, gender)
VALUES ('".$_POST["name"]."','".$_POST["email"]."','".$_POST["city"]."','".$_POST["gender"]."')";

if ($conn->query($sql) === TRUE) {
echo "
    <script type= 'text/javascript'>
        alert('New record created successfully');
    </script>";
}
else
{
    echo
    "<script type= 'text/javascript'>
        alert('Error: " . $sql . "<br>" . $conn->error."');
    </script>";
}

$conn->close();
}
?>
```

Step 5: SQL Query to Insert forms data into DB

```
$sql = "INSERT INTO entries (name, email, city, gender)
VALUES ('".$_POST["name"]."','".$_POST["email"]."','".$_POST["city"]."','".$_POST["gender"]."')";
```

Putting all things together

https://iq.opengenus.org/html-form-database-insertion-php-mysql/

Getting Right Data

Data Validation

Data Validation

- Required field will check whether the field is filled or not in the proper way. Most of cases we will use the * symbol for required field.

What is Validation ?

Validation means check the input submitted by the user.

- **Client-Side Validation** – Validation is performed on the client machine web browsers.
- **Server Side Validation** – After submitted by data, The data has sent to a server and perform validation checks in server machine.

Validation for Input Field

- Some of Validation rules for field are given below.

Field	Validation Rules
Name	Should required letters and white-spaces
Email	Should required @ and .
Website	Should required a valid URL
Radio	Must be selectable at least once
Check Box	Must be checkable at least once
Drop Down menu	Must be selectable at least once

Client-Side Validation-**Empty String**

- The code below checks that the field is not empty.
- If the user leaves the required field empty, it will show an error message.

```
if (emptyempty ($_POST["name"])) {
    $errMsg = "Error! You didn't enter the Name.";
            echo $errMsg;
} else {
    $name = $_POST["name"];
}
```

Client-Side Validation-**Validate String**

- The code below checks that the field will contain only alphabets and whitespace, for example - name.
- If the name field does not receive valid input from the user, then it will show an error message:

```
$name = $_POST ["Name"];
if (!preg_match ("/^[a-zA-z]*$/", $name) ) {
   $ErrMsg = "Only alphabets and whitespace are allowed.";
         echo $ErrMsg;
} else {
   echo $name;
}
```

Client-Side Validation-**Validate Number**

- The below code validates that the field will only contain a numeric value. For example - Mobile no.
- If the Mobile no field does not receive numeric data from the user, the code will display an error message:

```
$mobileno = $_POST ["Mobile_no"];
if (!preg_match ("/^[0-9]*$/", $mobileno) ){
    $ErrMsg = "Only numeric value is allowed.";
    echo $ErrMsg;
} else {
    echo $mobileno;
}
```

Client-Side Validation-**Validate Email**

- A valid email must contain @ and . symbols. PHP provides various methods to validate the email address.
- Here, we will use regular expressions to validate the email address.

```php
$email = $_POST ["Email"];
$pattern = "^[_a-z0-9-]+(\.[_a-z0-9-]+)*@[a-z0-9-]+(\.[a-z0-9-]+)*(\.[a-z]{2,3})$^";
if (!preg_match ($pattern, $email) ){
   $ErrMsg = "Email is not valid.";
        echo $ErrMsg;
} else {
   echo "Your valid email address is: " .$email;
}
```

Client-Side Validation-**Input Length Validation**

- The input length validation restricts the user to provide the value between the specified range, for Example - Mobile Number. A valid mobile number must have 10 digits.

The given code will help you to apply the length validation on user input

```
$mobileno = strlen ($_POST ["Mobile"]);
$length = strlen ($mobileno);

if ( $length < 10 && $length > 10) {
   $ErrMsg = "Mobile must have 10 digits.";
         echo $ErrMsg;
} else {
   echo "Your Mobile number is: " .$mobileno;
}
```

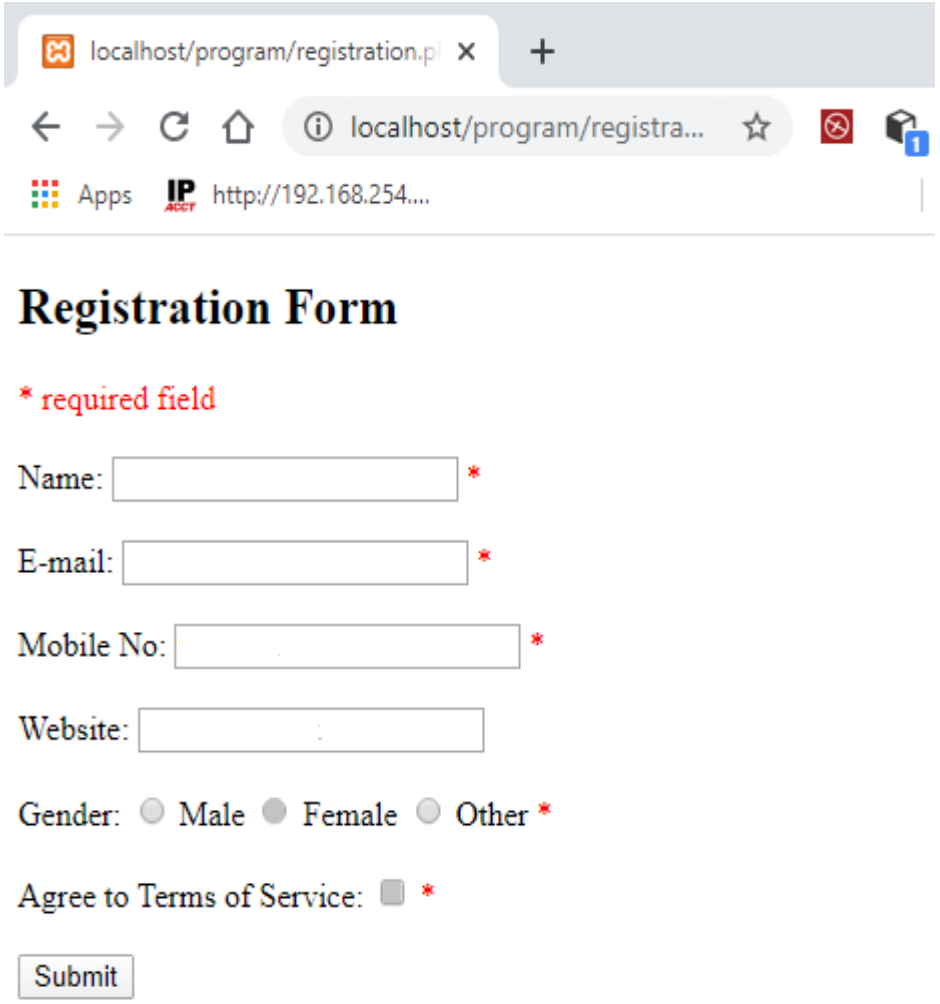
Client-Side Validation-**Button Click Validation**

- The below code validates that the user click on submit button and send the form data to the server one of the following method - get or post.

```php
if (isset ($_POST['submit']) {
    echo "Submit button is clicked.";
    if ($_SERVER["REQUEST_METHOD"] == "POST") {
        echo "Data is sent using POST method ";
    }
} else {
    echo "Data is not submitted";
}
```

https://www.javatpoint.com/form-validation-in-php

Client-Side Validation-Putting Everything Together



Case I: Press submit without data entry

Registration Form

* required field

Name: * Name is required

E-mail: * Email is required

Mobile No: * Mobile no is required

Website:

Gender: Male Female Other * Gender is required

Agree to Terms of Service: * Accept terms of services before submit.

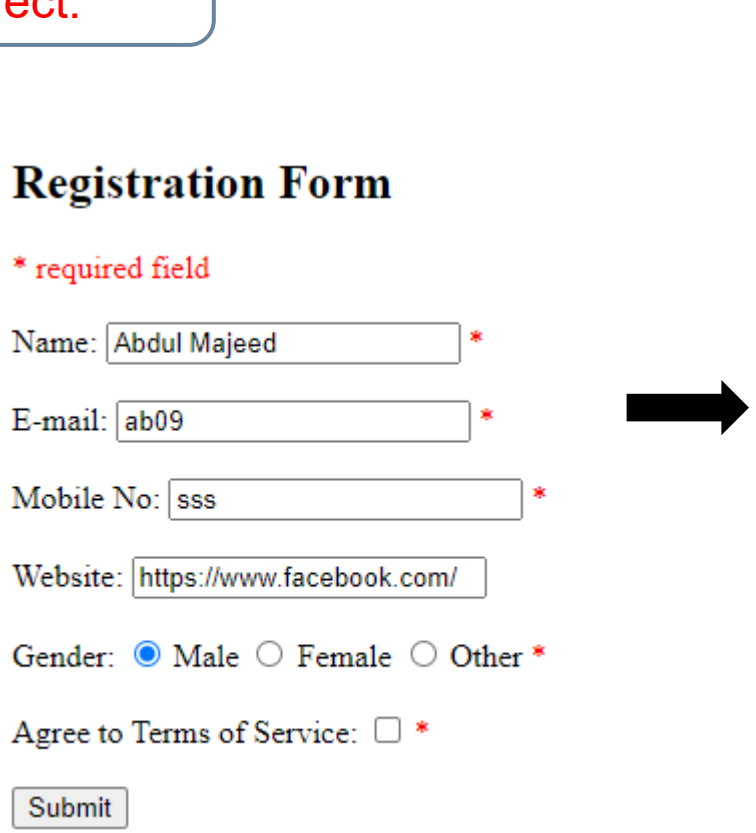
Submit

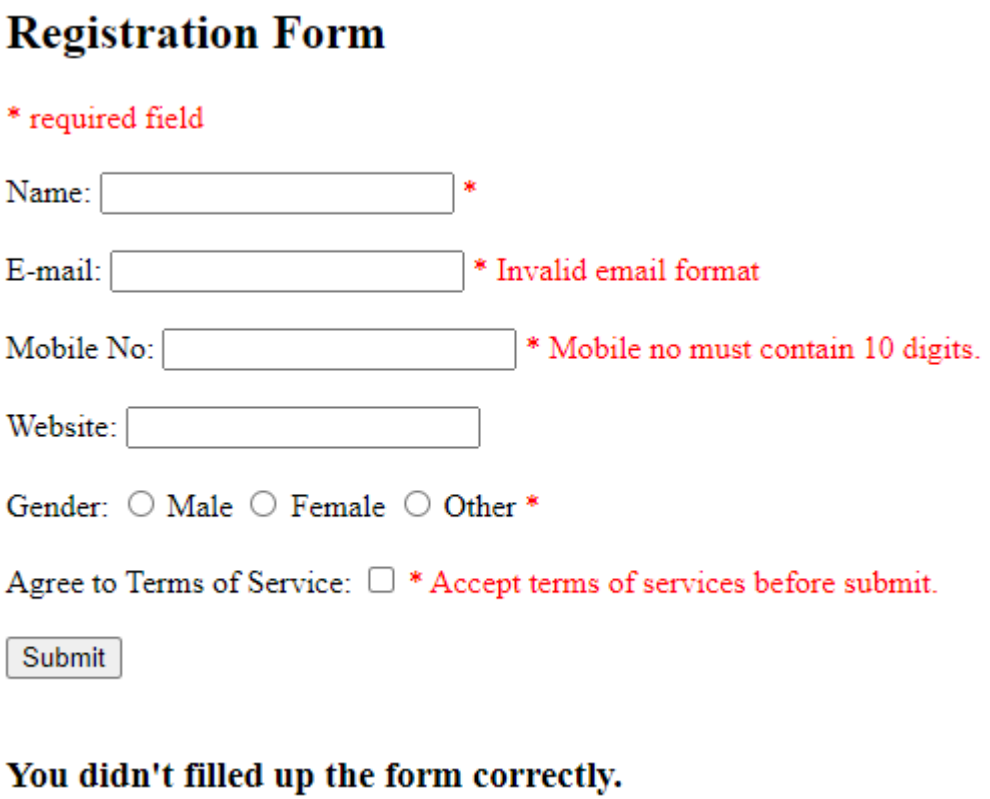
You didn't filled up the form correctly.

https://www.javatpoint.com/form-validation-in-php

Client-Side Validation-Putting Everything Together

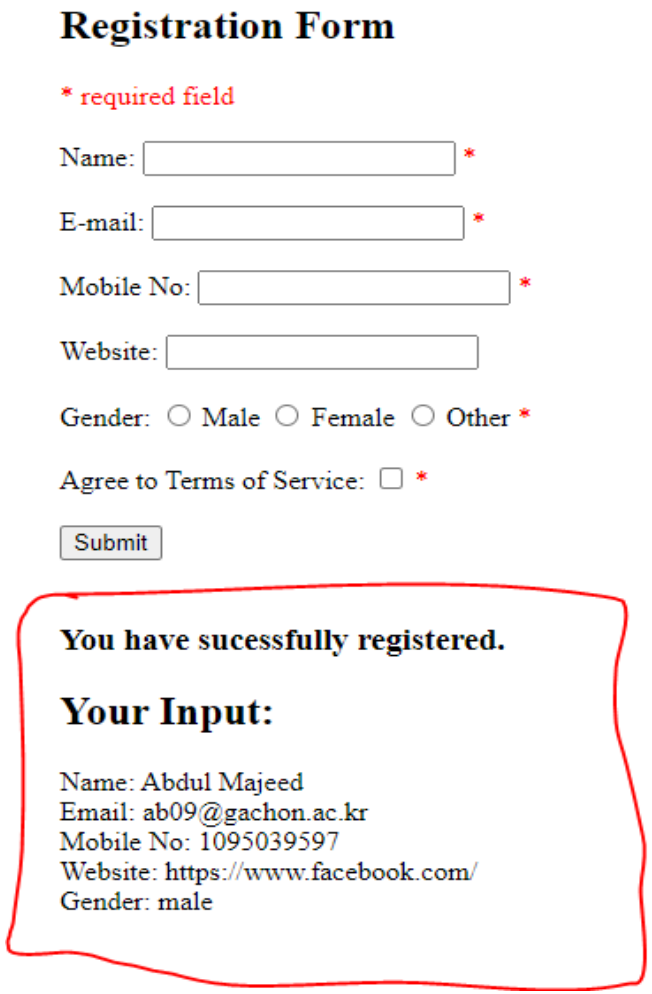
Case II: half data is correct & half is incorrect.





Client-Side Validation-Putting Everything Together

Case III: All data is correct.



Client-Side Validation-Putting Everything Together

```php
<?php
// define variables to empty values
$nameErr = $emailErr = $mobilenoErr = $genderErr = $websi
$name = $email = $mobileno = $gender = $website = $agree

//Input fields validation
if ($_SERVER["REQUEST_METHOD"] == "POST") {

//String Validation
    if (emptyempty($_POST["name"])) {
        $nameErr = "Name is required";
    } else {
        $name = input_data($_POST["name"]);
            // check if name only contains letters and wh
            if (!preg_match("/^[a-zA-Z ]*$/",$name)) {
                $nameErr = "Only alphabets and white spac
            }
    }

    //Email Validation
    if (emptyempty($_POST["email"])) {
            $emailErr = "Email is required";
    } else {
            $email = input_data($_POST["email"]);
            // check that the e-mail address is well-form
            if (!filter_var($email, FILTER_VALIDATE_EMAIL
                $emailErr = "Invalid email format";
            }
     }

    //Number Validation
    if (emptyempty($_POST["mobileno"])) {
            $mobilenoErr = "Mobile no is required";
    } else {
            $mobileno = input_data($_POST["mobileno"]);
            // check if mobile no is well-formed
            if (!preg_match ("/^[0-9]*$/", $mobileno) ) {
            $mobilenoErr = "Only numeric value is allowed
            }
        //check mobile no length should not be less and g
        if (strlen ($mobileno) != 10) {
            $mobilenoErr = "Mobile no must contain 10 dig
            }
    }
```

```php
<h2>Registration Form</h2>
<span class = "error">* required field </span>
<br><br>
<form method="post" action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>" >
    Name:
    <input type="text" name="name">
    <span class="error">* <?php echo $nameErr; ?> </span>
    <br><br>
    E-mail:
    <input type="text" name="email">
    <span class="error">* <?php echo $emailErr; ?> </span>
    <br><br>
    Mobile No:
    <input type="text" name="mobileno">
    <span class="error">* <?php echo $mobilenoErr; ?> </span>
    <br><br>
    Website:
    <input type="text" name="website">
    <span class="error"><?php echo $websiteErr; ?> </span>
    <br><br>
    Gender:
    <input type="radio" name="gender" value="male"> Male
    <input type="radio" name="gender" value="female"> Female
    <input type="radio" name="gender" value="other"> Other
    <span class="error">* <?php echo $genderErr; ?> </span>
    <br><br>
    Agree to Terms of Service:
    <input type="checkbox" name="agree">
    <span class="error">* <?php echo $agreeErr; ?> </span>
    <br><br>
    <input type="submit" name="submit" value="Submit">
    <br><br>
</form>
```

```php
<?php
    if(isset($_POST['submit'])) {
    if($nameErr == "" && $emailErr == "" && $mobilenoErr == "" && $genderE
        echo "<h3 color = #FF0001> <b>You have sucessfully registered.</b>
        echo "<h2>Your Input:</h2>";
        echo "Name: " .$name;
        echo "<br>";
        echo "Email: " .$email;
        echo "<br>";
        echo "Mobile No: " .$mobileno;
        echo "<br>";
        echo "Website: " .$website;
        echo "<br>";
        echo "Gender: " .$gender;
    } else {
        echo "<h3> <b>You didn't filled up the form correctly.</b> </h3>";
    }
    }
?>
```

Summary of the Today's Lesson

- Form data insertion to DB
  1) Create database
  2) Create table
  3) Write code for DB connection
  4) Write code for form elements
  5) Write code for form data handling
- Getting right data
  1) Empty string
  2) Email validation
  3) Other input validations

Summary/Coverage of this course-Web Programming

- Familiarity with six web development languages & Tools
  - Html
  - CSS
  - JavaScript
  - PHP
  - SQL
  - Databases
- Three-tier web app development
  - Front end, backend, databases
- Understanding of design patterns
  - MVC
- Data visualization
  - Static scenario
  - Dynamic scenario
- Examples of right data processing
  - Data validation
- Examples of data insertion/retravel in databases
  - Data validation- getting the correct data
- Real-world examples of top websites
  - Things board
  - Coupang
  - Gachon university

- Six Assignments
- Topics Presentations
- Project Presentations
- Discussions
- Demos
- Six Class Activities

Final Exam Schedule & Course Included

Code	Subject	Students	Professor/Tutor	Date (Day)	Time	Place
1420-7001	Web Prog.	31	MAJEED ABDUL	13. 06 (FRI)	13:30 ~	AI공학관-408

Please be on time.
Please thoroughly check all assignments,
PPT, and slides.
Off course, you need preparation as well.

Exam Syllabus
Lecture Notes 8~13 along with Lab Practices.
HomeWorks 5~7

Exam style: Multiple choice questions, and short questions (Same as mid term exams).