

# ALL COMMANDS FOR EXECUTING LAMP STACK WEB APPLICATION ( EXAMPLE WEBSITE )

IAM-> USERS-> roles->permission/policy->boundaries

project - 1 : Build a LAMP stack web app

---

# EC2 creation:

-> state ( N.Virginia ) ->web server name -> Linux OS -> Amazon Linux 2023 -> Instance type ( t2 micro )

-> Create new key pair -> Network settings edit -> default vpc -> create subnet ( select a VPC , subnet name,

no preference, IPv4 subnet CIDR block (select suggested one), create subnet -> enable -> Create

security group -> Inbound Security Group Rules custom to anywhere, Allow HTTP (port 80), Allow HTTPS

(port 443), Allow SSH (port 22) for access and description (allows HTTPS traffic from anywhere ) for these

3 -> save rules -> launch instance.

---

# Powershell:

-> ssh -i C:\Users\DELL\Downloads\sapna.pem ec2-user@98.84.23.45

-> sudo yum update -y

-> sudo yum install httpd

-> sudo systemctl start httpd

-> sudo systemctl enable httpd

-> systemctl status httpd

check 34.229.65.88 it (working or not)

get better result do remaining steps -

-> echo '<h1>Hello, World!</h1>' | sudo tee /var/www/html/index2.html

-> sudo chown -R apache:apache /var/www/html

-> sudo chmod -R 755 /var/www/html

If problem occur : (check syntax ok or not) and restart server:

-> sudo apachectl configtest

-> sudo systemctl restart httpd

<http://98.84.23.45/index2.html> [ output ]

---

# php configuration :

-> sudo yum install php php-mysqlnd

for check php installed successfully or not do this steps -

-> cd /var/www/html

-> sudo vi phpinformation.php

press i for write something inside it

<?php

phpinfo();

?>

esc button + :wq + enter

<http://98.84.23.45/phpinformation.php> [ output ]

---

**# MariaDB installation :**

**-> cd**

**-> curl -Ls -O https://downloads.mariadb.com/MariaDB/mariadb\_repo\_setup**

**-> ls -lrt [ output total 36 ]**

**-> sudo yum search mariadb**

**-> sudo yum install -y mariadb105-server [latest version 10.5]**

**-> sudo mysql\_secure\_installation**

**[ root password : ds421517, y, y, y, n, y, y ]**

**-> sudo yum install MariaDB-server MariaDB-client -y [ problem ]**

**so stop it .**

**Final Option: Use Docker**

**If installation via YUM continues to fail, consider running MariaDB-  
in a Docker container as an alternative.**

**\_\_\_\_\_ Solution \_\_\_\_\_**

**-> sudo yum install docker -y**

**-> sudo systemctl start docker**

**-> sudo systemctl enable docker**

**-> sudo systemctl status docker**

**-> sudo lsof -i :3306**

**-> sudo docker rm -f mariadb**

**-> sudo docker run --name mariadb -e MYSQL\_ROOT\_PASSWORD=my-secret-pw -d -p 3307:3306  
mariadb:latest**

-> sudo docker ps [ checking purpose ]

-> mysql -u root -p -h 127.0.0.1 -P 3307

password: my-secret-pw

-> CREATE DATABASE sapnadbatabase;

-> USE sapnadbatabase;

-> show databases;

-> CREATE TABLE user5 (

id INT AUTO\_INCREMENT PRIMARY KEY,

username VARCHAR(50) NOT NULL,

password VARCHAR(255) NOT NULL

);

-> show tables;

-> INSERT INTO user5 (username, password) VALUES ('shupta', 'shupta');

-> SELECT \* FROM user5;

-> exit

# MariaDB working good or not for check use it -

-> sudo nano /var/www/html/in.php

<?php

echo '<h1>Welcome to My LAMP Stack Application! how can i help you dear ?</h1>';

?>

http://98.84.23.45/in.php [ output ]

---

\* # Experiment by shupta:

**Aim :**

**Modify your PHP application to connect to the database you created, using the appropriate credentials.**

**i need to connect a database with a login page. first logging page will be showed login ,password, forgot**

**password option will be visible, then we give user id and password then we entered in that database page. HTML, CSS, PHP .**

---

**If mariadb entering process failed use these commands for recreate the password:**

**-> mysql -u root -p**

**-> sudo systemctl stop mariadb**

**-> sudo mysqld\_safe --skip-grant-tables &**

**-> mysql -u root**

**-> FLUSH PRIVILEGES; -- Ensure the privilege tables are loaded**

**-> ALTER USER 'root'@'localhost' IDENTIFIED BY 'sapnapass'; -- Set your new password**

**-> exit**

**-> sudo systemctl stop mariadb**

**-> sudo systemctl start mariadb**

**Try Logging in Again:**

**-> mysql -u root -p [ password : sapnapass ]**

---

**Database creation:**

**-> SHOW DATABASES;**

**-> CREATE DATABASE sapnadatabase;**

-> USE sapnadbatabase;

CREATE TABLE users (

id INT(11) AUTO\_INCREMENT PRIMARY KEY,

username VARCHAR(255) NOT NULL UNIQUE,

password VARCHAR(255) NOT NULL,

address VARCHAR(255),

email VARCHAR(255) NOT NULL UNIQUE,

phone\_number VARCHAR(15),

course VARCHAR(100),

branch VARCHAR(100)

);

-> SHOW DATABASES; [sapnadbatabase]

-> SHOW TABLES; [users]

-> DESCRIBE users;

-> exit

1. Frist make a register page for student's registration :

-> sudo nano /var/www/html/register.php

<?php

session\_start();

error\_reporting(E\_ALL);

ini\_set('display\_errors', 1);

\$servername = "localhost";

```
$username = "root"; // Your database username

$password = "sapnapass"; // Your database password

$dbname = "sapnadatabase"; // Your database name


$conn = new mysqli($servername, $username, $password, $dbname);


if ($conn->connect_error) {

    die("Connection failed: " . $conn->connect_error);

}


$error = "";


if ($_SERVER['REQUEST_METHOD'] == 'POST') {

    $user = $_POST['username'];

    $pass = $_POST['password'];

    $address = $_POST['address'];

    $email = $_POST['email'];

    $phone_number = $_POST['phone_number'];

    $course = $_POST['course'];

    $branch = $_POST['branch'];


    // Hash the password

    $hashed_password = password_hash($pass, PASSWORD_DEFAULT);


    // Prepare a statement to insert into the database
```

```
$stmt = $conn->prepare("INSERT INTO users (username, password, address, email, phone_number,
course, branch) VALUES (?, ?, ?, ?, ?, ?, ?)");
```

```
$stmt->bind_param("ssssss", $user, $hashed_password, $address, $email, $phone_number,
$course, $branch);
```

```
if ($stmt->execute()) {
```

```
    header("Location: login.php"); // Redirect to login page after registration
```

```
    exit();
```

```
} else {
```

```
    $error = "Error: " . $stmt->error;
```

```
}
```

```
$stmt->close();
```

```
}
```

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

```
?>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <title>Register</title>
```

```
    <link rel="stylesheet" type="text/css" href="styles.css">
```

```
    <style>
```

```
        input[type="text"], input[type="password"], input[type="email"] {
```

```
            width: calc(100% - 20px); /* Adjust width to be slightly smaller */
```

```
            padding: 10px;
```



```
margin: 10px 0;

border: 2px solid rgba(255, 255, 255, 0.5); /* Lighter border color */

border-radius: 5px;

background: rgba(255, 255, 255, 0.2); /* Light transparent background */

color: #fff; /* Text color */

}
```

```
button {

    background-color: #39e75f; /* Button color */

    color: #fff; /* Button text color */

    padding: 10px;

    border: none;

    border-radius: 5px;

    cursor: pointer;

    width: 100%; /* Full width */

    font-size: 16px;

    transition: background 0.3s; /* Smooth transition */

}
```

```
button:hover {

    background-color: #34c754; /* Change color on hover */

}
```

```
</style>
```

```
</head>
```

```
<body>
```

**<h2>Register</h2>**

**<form method="POST" action="">**

**<label for="username">Username:</label>**

**<input type="text" id="username" name="username" required>**

**<br>**

**<label for="password">Password:</label>**

**<input type="password" id="password" name="password" required>**

**<br>**

**<label for="address">Address:</label>**

**<input type="text" id="address" name="address" required>**

**<br>**

**<label for="email">Email:</label>**

**<input type="email" id="email" name="email" required>**

**<br>**

**<label for="phone\_number">Phone Number:</label>**

**<input type="text" id="phone\_number" name="phone\_number">**

**<br>**

**<label for="course">Course:</label>**

**<input type="text" id="course" name="course" required>**

**<br>**

```
<label for="branch">Branch:</label>

<input type="text" id="branch" name="branch" required>

<br>

<button type="submit">Register</button>

<br>

<?php if (!empty($error)) echo "<p>$error</p>"; ?>

</form>

</body>

</html>
```

<http://98.84.23.45/register.php> [ output ]

---

2. Second make a login page for student's registration :

-> `sudo nano /var/www/html/login.php`

```
<?php

session_start();

error_reporting(E_ALL);

ini_set('display_errors', 1);


// Database connection settings

$servername = "localhost";
```

```
$username = "root"; // Your database username

$password = "sapnapass"; // Your database password

$dbname = "sapnadatabase"; // Your database name


// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);


// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

$error = "";

$userData = [];


if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $user = $_POST['username'];
    $pass = $_POST['password'];


    // Escape user inputs for security
    $user = $conn->real_escape_string($user);


    // Prepare a statement to avoid SQL injection
    $stmt = $conn->prepare("SELECT password, address, email, phone_number, course, branch FROM
users WHERE username=?");

    $stmt->bind_param("s", $user);
```

```
$stmt->execute();

$stmt->store_result();

if ($stmt->num_rows > 0) {

    // User exists, fetch the hashed password and additional fields

    $stmt->bind_result($hashed_password, $address, $email, $phone_number, $course, $branch);

    $stmt->fetch();


    // Verify the password

    if (password_verify($pass, $hashed_password)) {

        // Successful login

        $_SESSION['username'] = $user;

        // Store additional user data in session

        $_SESSION['address'] = $address;

        $_SESSION['email'] = $email;

        $_SESSION['phone_number'] = $phone_number;

        $_SESSION['course'] = $course;

        $_SESSION['branch'] = $branch;


        header("Location: dashboard.php"); // Redirect to a dashboard or welcome page

        exit();

    } else {

        $error = "Invalid username or password!";

    }

} else {
```

```
    $error = "Invalid username or password!";  
}
```

```
$stmt->close();  
}
```

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

```
<!DOCTYPE html>  
  
<html>  
  
<head>  
  
    <title>Login</title>  
  
    <link rel="stylesheet" type="text/css" href="styles.css">  
  
</head>  
  
<body>  
  
    <h2>Login</h2>  
  
    <form method="POST" action="">  
  
        <label for="username">Username:</label>  
  
        <input type="text" id="username" name="username" required>  
  
        <br>  
  
        <label for="password">Password:</label>  
  
        <input type="password" id="password" name="password" required>  
  
        <br>  
  
        <button type="submit">Login</button>
```

```
<br>

<?php if (!empty($error)) echo "<p>$error</p>"; ?>

</form>

</body>

</html>
```

<http://98.84.23.45/login.php> [ output ]

---

3. Second make a dashboard page :

-> sudo nano /var/www/html/dashboard.php

```
<?php

session_start();

if (!isset($_SESSION['username'])) {

    header("Location: login.php");

    exit();

}

?>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Dashboard</title>

<link rel="stylesheet" type="text/css" href="styles.css">

</head>

<body>

<h1>Welcome, <?php echo htmlspecialchars($_SESSION['username']); ?>!/h1>


<div class="user-info-container">

    <div class="user-info-box">

        <p><strong>Address:</strong> <?php echo htmlspecialchars($_SESSION['address']); ?></p>

    </div>

    <div class="user-info-box">

        <p><strong>Email:</strong> <?php echo htmlspecialchars($_SESSION['email']); ?></p>

    </div>

    <div class="user-info-box">

        <p><strong>Phone Number:</strong> <?php echo
htmlspecialchars($_SESSION['phone_number']); ?></p>

    </div>

    <div class="user-info-box">

        <p><strong>Course:</strong> <?php echo htmlspecialchars($_SESSION['course']); ?></p>

    </div>

    <div class="user-info-box">

        <p><strong>Branch:</strong> <?php echo htmlspecialchars($_SESSION['branch']); ?></p>

    </div>

</div>
```



```
<a href="logout.php" class="logout-button">Logout</a>
</body>
</html>
```

<http://98.84.23.45/dashboard.php> [output ]

---

4. Last make a styles.css file :

-> sudo nano /var/www/html/styles.css

```
body {
    font-family: 'Arial', sans-serif;

    background: linear-gradient(135deg, #6a11cb 0%, #ffcccb 100%); /* Gradient background */
    color: #fff; /* Text color */

    margin: 0;

    padding: 40px; /* Increased padding for spacing */

    display: flex;

    flex-direction: column; /* Stack elements vertically */

    justify-content: center;

    align-items: center; /* Center content */

    min-height: 100vh; /* Full height */

    text-align: center;
}
```

```
form {  
  
    background-color: rgba(255, 255, 255, 0.1); /* Slightly transparent background */  
  
    border-radius: 10px;  
  
    padding: 20px;  
  
    box-shadow: 0 4px 15px rgba(0, 0, 0, 0.3);  
  
    width: 300px; /* Fixed width for form */  
  
    margin: 20px 0; /* Space between forms if needed */  
  
}
```

```
input[type="text"], input[type="password"] {  
  
    width: 100%; /* Full width */  
  
    padding: 10px;  
  
    margin: 10px 0;  
  
    border: 2px solid #fff; /* White border */  
  
    border-radius: 5px;  
  
    background: rgba(255, 255, 255, 0.2); /* Light transparent background */  
  
    color: #fff; /* Text color */  
  
    font-size: 16px; /* Slightly larger font size */  
  
}
```

```
input[type="text"]:focus, input[type="password"]:focus {  
  
    border-color: #6a11cb; /* Change border color on focus */  
  
    outline: none; /* Remove outline */  
  
}
```

```
button {  
  
    background-color: #2575fc; /* Button color */  
  
    color: #fff; /* Button text color */  
  
    padding: 10px;  
  
    border: none;  
  
    border-radius: 5px;  
  
    cursor: pointer;  
  
    width: 100%; /* Full width */  
  
    font-size: 16px;  
  
    transition: background 0.3s; /* Smooth transition */  
}
```

```
button:hover {  
  
    background-color: #6a11cb; /* Change color on hover */  
}
```

```
p {  
  
    margin-top: 10px;  
  
    color: #ffd700; /* Golden color for error messages */  
}
```

/\* Additional styles for user info boxes \*/

```
.user-info-box {  
  
    background-color: rgba(255, 255, 255, 0.2); /* Slightly transparent background */
```

```
border-radius: 10px;  
padding: 15px;  
margin: 10px 0; /* Space between boxes */  
width: 280px; /* Adjusted width for better fit */  
box-shadow: 0 4px 10px rgba(0, 0, 0, 0.3);  
}
```

---

**output :**

**<http://98.84.23.45/register.php>**

**Username:**

**sk**

**Password:**

**sk123**

**Address:**

**kkd**

**Email:**

**sk@gmail.com**

**Phone Number:**

**0236584129**

**Course:**

**b.tech**

**Branch:**

**CSE**

**Register**

**<http://98.84.23.45/login.php>**

**Username:**

**sk**

**Password:**

**sk123**

**<http://98.84.23.45/dashboard.php>**

**Welcome, sk!**

**Address: kkd**

**Email: sk@gmail.com**

**Phone Number: 0236584129**

**Course: b.tech**

**Branch: CSE**

**Logout**