

1. Create a responsive web page which shows the ecommerce/college/ exam admin dashboard with sidebar and statistics in cards using HTML, CSS and Bootstrap.

```
<!doctype html>
<html lang=en>
<meta charset=UTF-8>
<meta content="width=device-width,initial-scale=1" name=viewport>
<title>Dashboard</title>
<link
href=https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.
css rel=stylesheet>
<style>
  body {
    font-family: Arial, sans-serif
  }
  .sidebar {
    height: 100%;
    width: 250px;
    position: fixed;
    top: 0;
    left: 0;
    background-color: #343a40;
    padding-top: 20px
  }
  .card {
    margin-bottom: 20px
  }
  .sidebar a {
    padding: 10px 15px;
    text-decoration: none;
    display: block;
    color: #f8f9fa
  }
  .sidebar a:hover {
    background-color: #495057
  }
  .content {
    margin-left: 250px;
    padding: 20px
  }
</style>
<div class=sidebar><a href=#>Dashboard</a><a href=#>Products</a><a
href=#>Orders</a><a href=#>Customers</a></div>
```

```
<div class=content>
  <div class=container>
    <h2>Dashboard</h2>
    <div class=row>
      <div class=col-md-4>
        <div class=card>
          <div class=card-body>
            <h5 class=card-title>Total Sales</h5>
            <p class=card-text>500
          </div>
        </div>
      </div>
      <div class=col-md-4>
        <div class=card>
          <div class=card-body>
            <h5 class=card-title>Total Products</h5>
            <p class=card-text>50
          </div>
        </div>
      </div>
      <div class=col-md-4>
        <div class=card>
          <div class=card-body>
            <h5 class=card-title>Total Orders</h5>
            <p class=card-text>100
          </div>
        </div>
      </div>
    </div>
    <div class="row mt-4">
      <div class=col-md-6>
        <div class=card>
          <div class=card-body>
            <h5 class=card-title>Top Selling Products</h5>
            <ul>
              <li>Product A - 50 units
              <li>Product B - 40 units
              <li>Product C - 30 units
            </ul>
          </div>
        </div>
      </div>
      <div class=col-md-6>
```

```
<div class=card>
  <div class=card-body>
    <h5 class=card-title>Recent Orders</h5>
    <ul>
      <li>Order #001 - ₹100
      <li>Order #002 - ₹150
      <li>Order #003 - ₹200
    </ul>
  </div>
</div>
</div>
</div>
```

2. Write a JavaScript Program to get the user registration data and push to Array/local storage with AJAX POST method and data list in the new page.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>User Registration</title>
</head>
<body>
  <div class="container mt-5">
    <h2>User Registration</h2>
    <form id="registrationForm">
      <div>
        <label for="name">Name:</label>
        <input type="text" class="form-control" id="name"
name="name" required>
      </div>
      <div>
        <label for="age">Age:</label>
        <input type="number" class="form-control" id="age"
name="age" required>
      </div>
      <div>
        <label for="phone">Phone:</label>
        <input type="text" class="form-control" id="phone"
name="phone" required>
      </div>
      <button type="submit" class="btn btn-primary"
id="btn">Register</button>
      <button type="button" class="btn btn-secondary"
id="clearBtn">Clear</button>
    </form>
    <h2 class="mt-5">Registered Users</h2>
    <div class="table-wrapper mt-3">
      <table class="table table-striped table-bordered">
        <thead class="thead-dark">
          <tr>
            <th>#</th>
            <th>Name</th>
            <th>Age</th>

```

```

        <th>Phone</th>
    </tr>
</thead>
<tbody id="tbody">
</tbody>
</table>
</div>
</div>
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js">
</script>
<script>
    $(document).ready(function () {
        let storedUser = JSON.parse(localStorage.getItem("users"))
|| [];

        let displayData = () => {
            let tbody = document.getElementById("tbody");
            tbody.innerHTML = "";
            storedUser.forEach((user, index) => {
                tbody.innerHTML += `
<tr>
    <td>${index + 1}</td>
    <td>${user.name}</td>
    <td>${user.age}</td>
    <td>${user.phone}</td>
</tr>`;
            });
        };

        $("#registrationForm").submit(function (event) {
            event.preventDefault();
            const name = $("#name").val();
            const age = $("#age").val();
            const phone = $("#phone").val();
            if (name && age && phone) {
                let postObject = {
                    name: name,
                    age: age,
                    phone: phone,
                };
                storedUser.unshift(postObject);
                localStorage.setItem("users",
JSON.stringify(storedUser));
                displayData();
            }
        });
    });

```

```
        } else {  
            alert("Please fill in all the required fields.");  
        }  
    });  
    $("#clearBtn").click(function () {  
        $("input[type='text'], input[type='number']").val("");  
    });  
    displayData();  
});  
</script>  
</body>  
</html>
```

3. Create a version control account on GitHub and use Git commands to create a repository and push your code to GitHub.

```
$git init
$git add .
$git commit -m "Here goes message"

$git remote add origin {remote_link}
$git push -u origin HEAD:master

$git branch branch_name #create branch
$git checkout branch_name #switch to branch
$git branch #list all branches
$git branch -d abc #delete branch which is not
currently being used

$git show #displays commit history with changes
```

4. Create Docker Container Environment.

Create a Dockerfile: Inside your directory, create a file named **Dockerfile** (without any file extension). This file will contain instructions for building your Docker image.

Write Dockerfile: Open the Dockerfile in a text editor and add the following lines:

```
FROM alpine:latest  
CMD clear && echo "Hello, World!"
```

Build Docker Image: Open a terminal, navigate to your directory containing the Dockerfile, and run the following command to build your Docker image:

```
docker build -t hello-world .
```

Run Docker Container: After the image is built, you can run a container based on that image using the following command:

```
docker run hello-world
```


5. Create an Angular application which will do the following actions: Register User, Login User, Show User Data on Profile Component.

Create new Angular project using

```
$ng new project_name  
$code projectname #to open project in VSCode
```

Now edit *src/app/app.component.html* and replace everything with

```
Name: <input type="text" name="name" id="name" #name> <br>  
Email: <input type="email" name="email" id="email"  
#email><br>  
Password: <input type="password" name="password"  
id="password" #password><br>  
<button type="submit" (click)="getValue(name.value,  
email.value, password.value)">Submit</button>  
<h2>Registered users</h2>  
<p>Name: {{getName}}</p>
```

And *src/app/app.component.ts* with

```
import { Component } from '@angular/core';  
import { RouterOutlet } from '@angular/router';  
  
@Component({  
  selector: 'app-root',  
  standalone: true,  
  imports: [RouterOutlet],  
  templateUrl: './app.component.html',  
  styleUrls: ['./app.component.css']  
})  
export class AppComponent {  
  title = 'vayshi';  
  getName="";  
  getEmail="";  
  getPass="";  
}
```

```
getvalue(name:string,email:string,pass:string){  
    this.getEmail=email;  
    this.getName=name;  
    this.getPass=pass;  
}  
}
```

6. Create a Node.JS Application which serves a static website.

Create the file main.js with below code in it

```
const express = require('express');
const app = express();
app.use(express.static('public'));

app.listen(5000, ()=>{
  console.log(`server @ http://localhost:5000`);
});
```

As well create the folder named **public** and paste your static website files there.
then open the terminal and run below commands.

```
$npm install express
$node main.js
```

Done your server is now live at <http://localhost:5000>

7. Create a simple Mobile Website using jQuery Mobile

```
<!DOCTYPE html>
<html>
<head>
  <title>Simple jQuery Mobile Website</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css"
>
  <script src="https://code.jquery.com/jquery-1.11.1.min.js"></script>
  <script
src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.js">
</script>
</head>
<body>
<div data-role="page" id="homepage">
  <div data-role="header">
    <h1>Welcome to my site</h1>
  </div> <div role="main" class="ui-content">
    <p>
This is the homepage. Click the button below to go to the secondary
page.
</p>
    <a href="#secondarypage" class="ui-btn ui-btn-b ui-corner-all">
Go to Secondary Page</a>
  </div> <div data-role="footer">
    <h4>&copy; 2024 Simple Website</h4>
  </div></div>
<div data-role="page" id="secondarypage">
  <div data-role="header">
    <h1>Secondary Page</h1>
    <a href="#homepage" class="ui-btn ui-btn-left ui-icon-home
ui-btn-icon-left">Home</a>
  </div> <div role="main" class="ui-content">
    <p>This is the secondary page.</p>
  </div> <div data-role="footer">
    <h4>&copy; 2024 Simple Website</h4>
  </div></div>
</body>
</html>
```