

# Assignment 1 : Create a responsive web page - registration page

In web development, **responsiveness** refers to a website or web application's ability to **adapt its layout and design to different screen sizes and devices**, such as desktops, tablets, and smartphones.

## Key points:

- A **responsive website** looks good and works well on all screen sizes.
- It adjusts content like images, text, and navigation menus automatically based on the device's **screen resolution and orientation**.
- It uses techniques like:
  - **Media queries** in CSS
  - **Flexible grid layouts**
  - **Relative units** (% , em, rem) instead of fixed pixels
  - **Responsive images**

## Example:

On a large screen, a website might show a horizontal navigation bar. On a mobile phone, the same site may switch to a hamburger menu to save space.

### ✅ 3. What is the purpose of setting `display: flex` and `justify-content: center` in the `body`?

Answer:

This centers the `.container` horizontally and vertically on the page. It's a modern way to create a vertically and horizontally centered layout using Flexbox.

### ✅ 5. What does the media query do in your code?

Answer:

The media query applies styles specifically for screens smaller than or equal to 480px (like mobile devices). It reduces padding to ensure the form fits well on small screens, enhancing usability.

### ✓ 1. What is a registration page and why is it important?

Answer:

A registration page allows users to create an account by providing personal information like name, email, password, etc. It is important because it helps identify users, provide personalized experiences, and store user-related data securely.

### ✓ 2. What does "responsive design" mean?

Answer:

Responsive design means that the layout and elements of a web page adjust automatically based on the screen size and device (mobile, tablet, desktop) to ensure a good user experience.

### ✓ 3. How did you make your registration page responsive?

Answer:

I used CSS media queries to adjust the layout based on screen width. I also used flexible units like percentages instead of fixed pixels, and made sure form elements stack vertically on smaller screens.

### ✓ 13. Why is it important to separate HTML, CSS, and JavaScript files?

Answer:

Separating these files improves code organization, readability, easier debugging, and better maintenance. It also supports reusability and faster page load with caching.

### ✓ 14. How can you test if your page is truly responsive?

Answer:

I tested the page on different screen sizes using browser developer tools, resized the window manually, and also checked on real devices like smartphones and tablets.

## ✓ 1. What are Media Queries? Why are they used for responsiveness?

Answer:

Media queries are a feature of CSS that allow you to apply different styles depending on the device's characteristics, like screen width, height, or resolution. They are crucial for creating responsive designs because they enable a webpage to adjust its layout and styling based on the size of the viewport (the visible area of the page).

For example, you can change the layout from a multi-column view on desktop to a single-column view on mobile using media queries. This ensures that the page is user-friendly on various devices, enhancing the overall user experience.

Example:

```
css Copy Edit

@media screen and (max-width: 480px) {
  .container {
    padding: 10px;
  }

  .form {
    padding: 15px;
  }
}
```

In this example, when the screen width is 480px or less, the `.container` and `.form` elements will get smaller padding, making them fit better on smaller screens.

## ✓ 3. What is Bootstrap?

Answer:

Bootstrap is a popular open-source CSS framework that provides pre-designed components, grid systems, and utilities for building responsive web pages. It comes with a set of predefined styles for common UI elements (like buttons, forms, navbars, and alerts) and a grid layout system that adapts to different screen sizes.

## ✓ 4. What is Tailwind CSS?

Answer:

Tailwind CSS is a utility-first CSS framework that provides low-level utility classes to style elements directly in HTML. Unlike Bootstrap, which offers pre-designed components, Tailwind focuses on providing utility classes (like `text-center`, `p-4`, `bg-blue-500`) to style individual elements.

## Summary:

Feature	Bootstrap	Tailwind CSS
Approach	Component-based	Utility-first
Customizability	Less flexible (pre-designed components)	Highly customizable (build from scratch)
File Size	Larger, includes components and JavaScript	Smaller, purgable with PurgeCSS
Design Philosophy	Predefined, standardized look	Custom and unique designs
Learning Curve	Easy for beginners (ready-to-use components)	Higher learning curve (requires understanding of CSS)

## Assignment 2 : Create a responsive web page - Login page

### ✓ 1. What is a login page, and why is it essential?

Answer:

A login page is a user interface that allows users to authenticate their identity by providing credentials such as a username (or email) and password. It is essential because it helps secure access to personal or sensitive information by ensuring that only authorized users can log in to a system.

### ✓ 2. What does "responsive design" mean?

Answer:

Responsive design refers to the practice of creating web pages that adjust their layout and content based on the screen size and resolution of the device. It ensures that the website is easy to use on a variety of devices such as desktops, tablets, and smartphones, providing an optimal viewing experience.

### ✓ 3. What is the purpose of the `input` field in a login form?

Answer:

The `input` field in a login form is used to collect user data, such as the username and password. It allows users to enter their credentials and submit them to the server for authentication. The type of input (e.g., `email`, `password`) defines the kind of data the field accepts and how it is displayed.

✅ **14. What are some best practices for the user experience (UX) on a login page?**

Answer:

- Keep the form fields minimal (only the essential fields like username/email and password).
- Provide visual feedback (e.g., showing a loading spinner when logging in).
- Use appropriate placeholders or input labels for clarity.
- Make error messages clear and helpful, guiding users to fix mistakes.
- Ensure the page is accessible (e.g., screen reader support and keyboard navigation).
- Ensure fast load times to reduce user frustration.

## Assignment 5 : Create HTTP server using Node JS

### Theory Questions.

**1. What is Node.js, and why is it useful for building HTTP servers?**

- **Answer:** Node.js is a JavaScript runtime built on Chrome's V8 engine. It allows JavaScript to be run on the server side, outside of a browser. It is especially useful for building HTTP servers because it is non-blocking, event-driven, and can handle many connections concurrently, making it efficient for real-time applications and scalable systems.

**2. What is the purpose of the `http` module in Node.js?**

- **Answer:** The `http` module in Node.js is used to create HTTP servers and clients. It allows us to handle incoming HTTP requests, send HTTP responses, and define routing logic. It is a core module in Node.js, and it's essential for building web servers.

**3. Explain the concept of request and response in an HTTP server.**

- **Answer:** In an HTTP server, the client sends an HTTP request to the server, which contains details like the HTTP method (GET, POST, etc.), headers, and the request URL. The server processes the request and sends an HTTP response back to the client. The response includes a status code (e.g., 200 for success or 404 for not found), headers, and a body (e.g., content like text, HTML, JSON).

4. What does `res.setHeader('Content-Type', 'text/plain')` do?

- Answer: `res.setHeader('Content-Type', 'text/plain')` sets the response's `Content-Type` header to `text/plain`, indicating that the response body will be plain text. This helps the client (browser) interpret the response appropriately.

5. What are the possible HTTP status codes and their meanings?

- Answer:
  - **200 OK:** The request was successful, and the server returns the requested data.
  - **404 Not Found:** The server cannot find the requested resource.
  - **500 Internal Server Error:** The server encountered an unexpected condition that prevented it from fulfilling the request.
  - **301 Moved Permanently:** The requested resource has been permanently moved to a new URL.

6. What is the significance of `req.url` and `res.statusCode` in your server code?

- Answer:
  - `req.url` gives the requested URL, which helps determine which page or resource the client wants to access.
  - `res.statusCode` specifies the HTTP status code for the response, indicating the result of processing the request (e.g., 200 for OK, 404 for Not Found).

2. Modify the code to respond with JSON data for the `/about` route instead of plain text.

Answer:

```
js                                                                    Copy Edit

const http = require('http');
const PORT = 3000;

const server = http.createServer((req, res) => {
  res.setHeader('Content-Type', 'application/json');

  if (req.url === '/') {
    res.statusCode = 200;
    res.end('Welcome to the Home Page');
  } else if (req.url === '/about') {
    res.statusCode = 200;
    const aboutData = {
      page: 'About',
      description: 'This page provides information about our website.'
    };
    res.end(JSON.stringify(aboutData));
  } else {
    res.statusCode = 404;
    res.end('Page Not Found');
  }
});

server.listen(PORT, () => {
  console.log(`Server running at http://localhost:${PORT}/`);
});
```

## Assignment 6 : Create Authentication server using Node JS

### 1. What is Express.js and why is it used?

- Answer: Express.js is a minimal and flexible Node.js web application framework. It simplifies the process of building web servers and APIs by providing robust features like routing, middleware support, request/response handling, and more.

### 2. What does `express.urlencoded({ extended: false })` do?

- Answer: It is middleware that parses incoming request bodies in `application/x-www-form-urlencoded` format. Setting `extended: false` means the `querystring` library is used to parse the data (no nested objects).

### 3. How is form data sent and received in this app?

- Answer:
  - The form in the HTML uses `method="POST"` and `action="/login"` to send the username and password.
  - The server uses `app.use(express.urlencoded())` to parse that data.
  - `req.body` contains the submitted values.

### 4. What is the difference between GET and POST methods?

- Answer:
  - GET sends data in the URL and is used for retrieving data (e.g., loading the login form).
  - POST sends data in the request body and is used for actions like submitting forms or sensitive data like passwords.

## Assignment 7 : Create static website Node JS application

### ◆ Static Website using Node.js:

- The content **doesn't change** for different users.
- Pages are **pre-built** (like HTML files) and served as-is.
- Node.js just acts like a **server** to send those files.
- Example: A portfolio website with fixed info.

Think of it like a **printed book** — everyone sees the same thing.

### ◆ Dynamic Website using Node.js:

- The content **changes based on user actions**, data, or input.
- Pages are **generated on the fly** using code (Node.js + database).
- Node.js handles **logic**, **user input**, **database**, etc.
- Example: A login system, blog, or shopping cart.

Think of it like a **restaurant menu** — different people can order different things.

### 1. What does `express.static(__dirname)` do?

- Answer: It tells Express to serve static files (HTML, CSS, JS, images, etc.) from the current directory (`__dirname`). So files like `style.css`, `image.png`, etc., can be accessed directly via URL.

### 2. Why is `res.sendFile()` used in the `/` route?

- Answer: It explicitly sends the `index.html` file when the user accesses the root path `/`. This is needed if we want to serve `index.html` as the default file when visiting the homepage.

### 3. What types of files are typically served in a static website?

- Answer:
  - `.html` – web page content
  - `.css` – stylesheets for design
  - `.js` – frontend scripts
  - `.png`, `.jpg`, `.svg` – images
  - Fonts, PDFs, etc.



## Assignment 8 : Create a program of react JS using React hooks

### ✔ What is React?

React is an open-source **JavaScript library** used for **building user interfaces**, especially for **single-page applications (SPAs)**. It allows developers to create **reusable UI components** and efficiently update the UI when data changes.

### ◆ Key Features of React:

1. **Component-Based:** UI is built from small, independent components.
2. **Declarative:** You describe what the UI should look like, and React takes care of rendering it.
3. **Virtual DOM:** React uses a virtual copy of the DOM to update the UI efficiently.
4. **Reusable Code:** Components can be reused across different parts of the application.
5. **Hooks:** Enable using state and lifecycle methods in functional components.

Reason	Benefit
Component-based	Reusability and modular code
Virtual DOM	High performance UI updates
Declarative syntax	Easier to read and debug
Hooks	Powerful state management in functional code
Strong ecosystem	Rich libraries and tools
SPAs support	Fast, dynamic user experience

## 1. What is a React Hook?

- Answer: Hooks are special functions in React that let you use state and other React features in functional components. Examples include `useState`, `useEffect`, `useContext`, etc.

## 2. What is the purpose of `useState`?

- Answer: `useState` allows you to declare and manage state variables in functional components.

Example:

```
js                                                                    Copy Edit
const [count, setCount] = useState(0);
```

## 3. What does `useEffect` do in React?

- Answer: `useEffect` runs side effects in functional components — like data fetching, timers, subscriptions, etc.

Example:

```
js                                                                    Copy Edit
useEffect(() => {
  console.log("Component mounted or updated");
}, []);
```

## ✅ Simple Difference Between React and Angular

Feature	React	Angular
Type	Library (for UI)	Full Framework (includes everything)
Language	JavaScript + JSX	TypeScript
Developed By	Facebook (Meta)	Google
Learning	Easier, more flexible	Steeper learning curve
Usage	You choose tools (like routing)	Everything is built-in
Rendering	Fast (uses Virtual DOM)	Fast (uses Real DOM + change detection)
Components	Function-based (with Hooks)	Class-based (with decorators)

### 🧠 Think of it like this:

- React = Just the engine (you add your own wheels, body, seats).
- Angular = A full car (engine + wheels + everything ready-made).

## Assignment 9 : Simple program to display Hello message on browser using React JS

### 1. What is React JS?

**Answer:** React is a JavaScript library developed by Facebook for building user interfaces, especially single-page applications using reusable components.

### 2. What is a component in React?

**Answer:** A component is a self-contained piece of UI (like a function) that can return JSX to render part of the interface.

### 3. What is JSX?

**Answer:** JSX (JavaScript XML) is a syntax extension that lets you write HTML-like code in JavaScript, which React then transforms into actual DOM elements.

## Assignment 12 : Design Simple web page using jQuery mobile

### ◆ 1. What is jQuery Mobile?

Answer:

jQuery Mobile is a **touch-optimized web framework** (based on jQuery) used to build responsive and accessible web applications that work on smartphones, tablets, and desktops with a single codebase.

### ◆ 2. What is the use of `data-role="page"` ?

Answer:

`data-role="page"` defines a **separate page or screen** within a jQuery Mobile application. It allows single HTML documents to have multiple pages.

### ◆ 3. How is the header created in jQuery Mobile?

Answer:

The header is created using `<div data-role="header">`. It automatically styles the section as a **header bar**, often used for titles or navigation.

### ◆ 4. What is the purpose of the `meta` tag with `viewport` in this code?

Answer:

It ensures the webpage is **mobile responsive** by controlling layout scaling on mobile devices.

Example:

html

Copy

Edit

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

### ◆ 5. What does the `ui-btn` class do?

Answer:

The `ui-btn` class styles the element as a **jQuery Mobile button**, making it touch-friendly and visually enhanced.

### ◆ 7. What are the roles of jQuery and jQuery Mobile libraries in this project?

Answer:

- **jQuery:** Provides core JavaScript functionality (DOM manipulation, events, etc.)
- **jQuery Mobile:** Adds mobile UI components and page transitions using `data-*` attributes.

## Assignment : Update DOM using JQuery

### ✓ What is jQuery?

jQuery is a **fast, small, and feature-rich JavaScript library** that simplifies HTML document traversal, event handling, animation, and Ajax interactions for rapid web development.

### ◆ 1. What is the DOM?

Answer:

DOM (Document Object Model) is a programming interface for HTML and XML documents. It represents the page structure as a **tree of objects** which can be modified using JavaScript or jQuery.

### ◆ 7. Why is jQuery used instead of plain JavaScript?

Answer:

jQuery simplifies common DOM tasks like element selection, event handling, and animations with **less code** and **better cross-browser support**.