

Web Development Workshop: Assessment Questions & Answers

Week 1: Foundations of Web Development & Tailwind CSS

1. What are the five main stages of the web development lifecycle?

- * Planning, Design, Development, Launch, Maintenance.

2. Explain the primary advantage of using a utility-first CSS framework like Tailwind CSS.

- * It allows for rapid UI development by applying pre-defined utility classes directly in your HTML, enabling you to build custom designs without writing custom CSS.

3. How do you include Tailwind CSS in a project using a CDN?

- * By adding the Tailwind CSS CDN script tag to the `<head>` of the HTML file.
- * Example: `<script src="https://cdn.tailwindcss.com"></script>`

4. Which Tailwind CSS utility class would you use to set a background color?

- * `bg-{color}` (e.g., `bg-blue-500`, `bg-gray-200`).

5. What is the purpose of the CSS Box Model?

- * The CSS Box Model is a fundamental concept that describes the rectangular boxes generated for elements, consisting of: content, padding, border, and margin. It defines how these parts combine to determine an element's total size and spacing.

6. Name the four components of the CSS Box Model.

- * Content, Padding, Border, Margin.

7. How do you add margin on all sides of an element using Tailwind?

- * `m-{size}` (e.g., `m-4`).

8. How do you add padding only to the top of an element using Tailwind?

- * `pt-{size}` (e.g., `pt-4`).

9. What is the main purpose of Flexbox in CSS?

- * To provide a efficient way to lay out, align, and distribute space among items in a container, even when their size is unknown or dynamic.

10. Which Tailwind class enables Flexbox on a container element?

- * `flex`.

11. What is the difference between `justify-content` and `align-items` in Flexbox?

- * `justify-content` aligns flex items along the main axis (horizontally, by default). `align-items` aligns flex items along the cross axis (vertically, by default).

12. In the context of the workshop's first week, what was the goal of the "Simple Web Page Structure" project?

- * To practice HTML layout and Tailwind CSS styling by building a basic multi-section webpage with a fixed navigation bar.

13. Why is a fixed navigation bar a common feature in web design?

- * It provides persistent access to the main navigation menu, improving user experience and site usability as the user scrolls.

14. In the "Box Model Exercise," how does the `border` property affect an element's total dimensions?

- * The `border` adds to the total width and height of an element. Total width = `width` + `padding-left` + `padding-right` + `border-left` + `border-right`.

15. What is the key difference between `margin` and `padding`?

- * `Padding` is the space between an element's content and its border. `Margin` is the space between an element's border and the surrounding elements.

Week 2: JavaScript Fundamentals & DOM Manipulation

16. What is the difference between `let`, `const`, and the older `var` for variable declaration?

- * `const` declares a variable that cannot be reassigned. `let` declares a block-scoped variable that can be reassigned. `var` is function-scoped and can be redeclared, which can lead to bugs.

17. When would you use `console.error()` instead of `console.log()`?

- * `console.error()` is used to log error messages to the console, which are often highlighted in red, making them easier to spot during debugging.

18. Write a simple `if-else` statement that checks if a variable `age` is 18 or over.

```
if (age >= 18) {  
  console.log("Adult");  
} else {  
  console.log("Minor");  
}
```

19. What is the DOM and why is it important for web interactivity?

- * The DOM (Document Object Model) is a programming interface for HTML and XML documents. It represents the page as a tree of objects, allowing JavaScript to dynamically access and update the content, structure, and style of a document.

20. What is the difference between `document.getElementById('id')` and `document.querySelector('#id')`?

* `getElementById` only selects an element by its ID. `querySelector` uses a CSS selector, so it can select by ID, class, tag, etc., and returns only the first matching element.

21. Write a `for` loop that prints numbers from 1 to 5 to the console.

```
for (let i = 1; i <= 5; i++) {  
  console.log(i);  
}
```

22. How does a `while` loop differ from a `for` loop?

* A `for` loop is typically used when the number of iterations is known. A `while` loop is used when the number of iterations is unknown and depends on a condition that must remain true to continue executing.

23. What is a JavaScript function?

* A function is a block of code designed to perform a particular task. It is executed when "called" or "invoked."

24. What is the key syntactic difference between a regular function declaration and an arrow function?

* A regular function uses the `function` keyword. An arrow function uses the `=>` arrow syntax and does not have its own `this` binding.

* Example: `function add(a, b) { return a + b; }` vs. `const add = (a, b) => a + b;`

25. What is an anonymous function?

* A function that does not have a name. It is often assigned to a variable or passed as an argument to another function.

26. In the "Simple ToDo App," what JavaScript event would you use to capture the action of a user clicking an "Add" button?

* The `click` event.

27. Describe the process of dynamically creating a new HTML element (like a list item) in JavaScript.

* You use `document.createElement('li')` to create the element, then set its properties (e.g., `textContent`), and finally append it to an existing parent element in the DOM using `parentElement.appendChild(newElement)`.

28. In the ToDo app, how would you implement the functionality to delete a task when a "Delete" button is clicked?

* Attach an event listener to each "Delete" button. When clicked, the event handler would find the parent list item of the button and remove it from the DOM using `element.remove()`.

29. Why is it important to prevent the default form submission behavior when handling a button click in a ToDo app?

* To stop the browser from refreshing the page, which is the default action for a form submission. This allows for a dynamic, single-page application experience without losing state.

30. What is "event delegation" and how could it be useful in a ToDo list application?

- * Event delegation is a technique where you attach a single event listener to a parent element to handle events on its child elements. It's useful for dynamically added elements (like new todo items) because you don't have to attach a listener to each new element individually.

Week 3: Asynchronous JavaScript & APIs

31. Why is JavaScript's default synchronous execution a problem when fetching data from an API?

- * Because the network request to the API can take a long time. If JavaScript waited for it to finish synchronously, the entire webpage would freeze, leading to a poor user experience.

32. What is a Promise in JavaScript?

- * A Promise is an object representing the eventual completion (or failure) of an asynchronous operation and its resulting value.

33. How does the `async` keyword change a function?

- * It makes the function always return a Promise. If the function returns a value, that value is wrapped in a resolved Promise.

34. How do you wait for a Promise to resolve inside an `async` function?

- * By using the `await` keyword before the Promise.

35. What is the primary purpose of the `fetch` API?

- * To make network requests (like GET, POST) to retrieve resources (usually data) from a server or an API.

36. What does the `fetch()` function return?

- * It returns a Promise that resolves to the `Response` object representing the response to the request.

37. Why do you need to call the `.json()` method on the response from a `fetch` request?

- * The `Response` object contains the entire HTTP response. The `.json()` method is used to extract the JSON data from the response body and parse it into a usable JavaScript object. This method itself returns a Promise.

38. What is the purpose of a `try...catch` block?

- * To handle errors gracefully. Code in the `try` block is executed, and if an error occurs, control is passed to the `catch` block to handle that error without crashing the entire script.

39. In the "Image Gallery" project, what two pieces of user input are used to fetch from the Picsum Photos API?

- * The page number and the limit (number of images per page).

40. Describe the steps to dynamically load an image from an API and display it on a webpage.

1. Use `fetch()` to get the image URL from the API.
2. Process the response (e.g., with `.json()`).
3. Create an `` element using `document.createElement('img')`.
4. Set the `src` attribute of the `` element to the URL received from the API.
5. Append the `` element to a container in the DOM.

41. How would you handle a scenario where the `fetch` request to the Picsum API fails (e.g., no internet connection)?

* By using a `try...catch` block around the `await fetch()` call. The error would be caught in the `catch` block, where you could display a user-friendly error message on the webpage.

42. What does the `await` keyword do?

* It pauses the execution of an `async` function until a Promise is settled (resolved or rejected). It can only be used inside an `async` function.

Week 4: Version Control, Deployment & Collaboration

43. What is version control and what problem does it solve?

* Version control is a system that records changes to files over time. It allows you to revert files or entire projects to previous states, track modifications, and compare changes. It solves the problem of losing work and managing collaborative development.

44. What is the primary purpose of Git?

* Git is a distributed version control system for tracking changes in source code during software development.

45. What is the difference between Git and GitHub?

* Git is the version control tool you use locally on your computer. GitHub is a cloud-based hosting service that uses Git for version control and provides a platform for collaboration.

46. Describe the function of the following Git commands: `git init`, `git add .`, `git commit -m "message"`.

- * `git init`: Initializes a new local Git repository in the current directory.
- * `git add .`: Stages all new and modified files in the current directory, preparing them for a commit.
- * `git commit -m "message"`: Creates a snapshot of the staged changes, saving them to the repository's history with a descriptive message.

47. After committing your code locally, what command do you use to send it to a remote repository on GitHub?

- * `git push origin main` (or `master`/your branch name).

48. If you want to contribute to an open-source project on GitHub that you don't have write access to, what is the process called?

- * Forking.

49. After forking a repository and making changes, how do you propose those changes be merged into the original project?

- * By creating a Pull Request (PR) from your forked repository to the original repository.

50. What is the purpose of "starring" a repository on GitHub?

- * To bookmark a repository you find interesting or useful, showing appreciation to the maintainer and making it easier to find in your own starred list.

51. Which deployment platform is covered in the workshop for deploying static websites?

- * Vercel.

52. Briefly describe the typical steps to deploy a project from your GitHub repository to Vercel.

- * Log in to Vercel with your GitHub account, click "Import Project," select the GitHub repository you want to deploy, configure settings (if needed), and click "Deploy." Vercel will automatically build and host your site.

53. After deploying to Vercel, what is the process of connecting a custom domain name to your project called?

- * Domain configuration or connecting a custom domain.