DWA_12 Knowledge Check

To complete this Knowledge Check, ensure you have worked through all the lessons in **Module 12: Declarative Abstractions.**

To prepare for your session with your coach, please answer the following questions. Then download this document as a PDF and include it in the repository with your code.

---

1. What are the benefits of direct DOM mutations over replacing HTML?

Performance boost

1. Direct DOM mutations allow to make specific changes to the DOM without affecting the entire HTML structure. More efficient than replacing entire HTML.

2. Updating specific elements in DOM can lead to fewer redraws, compared to replacing an entire HTML, and can result in better performance and smoother experience.

3. Real time updating specific elements or attributes in response to user actions, without reloading the entire page.

4. Direct DOM manipulation allows for more control over event handling. Attaching event listeners provide a more interactive and responsive user experience.

5. In some frameworks and libraries, direct DOM manipulation is used in conjunction with two way data binding. This enables synchronization between data model and DOM, making it easier to manage and update data-driven apps.

6. Direct DOM manipulation provides flexibility to implement custom rendering logic.

---

2. What low-level noise do JavaScript frameworks abstract away?

Imperative updating of the DOM, keeping track of what elements need to change

1. Cross-browser compatibility. Frameworks handle differences in how various browsers implement the DOM.

2. Frameworks abstract away the complexities of handling events in different browsers and ensuring a consistent event model.

3. Frameworks provides simplified and consistent API's for making async requests, abstracting away the intricacies of handling XMLHttpRequest or Fetch API in environments.

4. Frameworks offer state management solutions, abstracting the details of updating and synchronizing the application state across components or views.

5. Two-way data binding in frameworks help to synchronize the data model and the view, reducing the need for manual DOM manipulation.

6. Frameworks provide routing mechanisms that abstract away complexities of updating the state based on changes in URL.

7. Some frameworks use a virtual DOM to update the actual DOM. This abstraction optimizes the process of comparing and updating the DOM elements.

8. Frameworks can simplify the handling of Cross-Origin Resource Sharing issues, when making requests to a different domain.

9. Frameworks have a component based architecture and provide lifecycle hooks that abstract away the details of initializing, updating and destroying components.

10. Frameworks handle module loading and dependency management, abstracting away the need for manual management of script tags and dependencies.

11. Frameworks abstract away the differences between developing for different platforms, allowing use of the same codebase for web, mobile and desktop apps.

---

3. What essence do JavaScript frameworks elevate?

1. Frameworks offer a structured approach to organizing code, promoting the use of modular components and enforcing best practices, which helps with the building of scalable and maintainable applications.

2. Frameworks come with a set of pre-built functionalities, libraries and tools that can speed up the development process, allowing developers to focus on application logic rather than low-level details.

3. Frameworks handle cross-browser compatibility issues, ensuring that web applications work consistently across different browsers.

4. Frameworks encourage the reuse of code through components and modules.

5. Frameworks have large and active communities, which means a wealth of resources, documentation and third party plugins and extensions.

6. Frameworks enable the development of Single Page Applications, which provides a smoother user experience by loading a single HTML page and dynamically updates content as the user interacts with the application.

7. Frameworks provide two-way data binding, where changes in user interface automatically update the underlying data.

8. Frameworks are designed to optimize performance, resulting in faster rendering and a smoother user experience.

9. Frameworks provide patterns and structures that make it easier to scale applications.

10. Frameworks elevate web development by providing a structured, efficient and standardized approach to bullying applications.

---

4. Very broadly speaking, how do most JS frameworks achieve abstraction?
They hide away the imperative DOM mutations

1. JS Frameworks achieve abstraction through various mechanisms that allow developers to work at a higher level of abstraction, focusing on application logic rather than low level details.

2. Frameworks follow a component-based architecture which allows for creation of self-contained, reusable components that encapsulate both the UI and the logic. This abstraction allows for modularization of the codebase.

3. Frameworks use a virtual DOM to abstract away direct manipulation of browsers DOM. The virtual DOM is an in-memory representation of the actual DOM, and changes are first made to the virtual DOM. Framework then efficiently updates the real DOM, reducing unnecessary reflows and repaints.

4. Two-way data binding is a mechanism where changes in the UI automatically update the underlying data model, which simplifies the synchronization of data between the model and view, reducing the need for DOM manipulation.

5. Frameworks provide routing mechanisms that abstract away complexity of handling different views or pages in a single-page application. Developers can define routes and associated components, and the framework takes care of updating the view based on the current route.

6. Frameworks provide abstraction over AJAX requests and HTTP communication.

7. Frameworks use dependency injection to abstract the instantiation and management of dependencies.

8. State management libraries provide abstractions for managing the state of an application. They offer a central store that can be updated in a predictable and controlled manner, reducing the need for scattered state management code.

9. Abstraction over event handling is achieved by using event listeners or directives in frameworks.

10. Frameworks include template engines that abstract away manual creation of HTML.

---

5. What is the most important part of learning a JS framework?

1. Understanding the fundamental concepts of the framework. Understanding components, state management, data binding, routing and other core features.
2. Exploring documentation, which provides insights into capabilities, usage patterns and best practices.
3. Learning how to optimize the performance of your applications. Understanding techniques such as code splitting, lazy loading and minimizing unnecessary re-renders.