

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?

Some advantages of using direct DOM mutations include:

- Direct DOM mutations allow you to modify specific elements or attributes within the DOM without the need to recreate the entire HTML structure.
- With direct DOM mutations, you can target specific elements or attributes and modify them individually, providing granular control over the changes.
- By directly manipulating the DOM, you can achieve seamless interactivity within your web applications. You can dynamically update content, respond to user actions, and reflect changes in real-time without having to reload or replace the entire HTML structure.
- With direct DOM mutations, you can preserve the current state of the page.
- Since direct DOM mutations only require transmitting and applying incremental changes to the DOM, they can significantly reduce bandwidth consumption compared to sending entire HTML documents back and forth.
- Integration with other JavaScript libraries or frameworks.

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

Here are some common low-level noises that JavaScript frameworks handle:

- Frameworks handle cross-browser compatibility issues and provide fallback mechanisms or polyfills to bridge the gaps in browser support.
- JavaScript frameworks abstract away the low-level event handling mechanisms by providing unified event models and abstractions. This allows developers to attach event listeners, respond to user actions, and handle events in a consistent and simplified manner.
- JavaScript frameworks provide abstractions for DOM manipulation, offering high-level APIs that make it easier to create, update, and remove DOM elements. These abstractions hide the low-level complexities of working with the DOM API and provide a more intuitive and declarative way to interact with the HTML structure.
- JavaScript frameworks offer built-in state management solutions that abstract away the low-level details of managing state.

3. What essence do JavaScript frameworks elevate?

Here are some essential aspects that JavaScript frameworks elevate:

- JavaScript frameworks offer high-level APIs, component-based architectures, and reusable code patterns that streamline development and reduce boilerplate code.
- Frameworks promote code organization and maintainability by enforcing structured architectures and best practices. They often provide guidelines and patterns for organizing code into modules, components, or services, promoting separation of concerns and modularity.
- JavaScript frameworks excel at optimizing the rendering of user interfaces.
- JavaScript frameworks offer built-in state management solutions or integrate well with dedicated state management libraries.
- JavaScript frameworks help address cross-platform compatibility by providing abstractions and tools for building applications that work across various devices and platforms.

4. Very broadly speaking, how do most JS frameworks achieve abstraction?

- JavaScript frameworks often adopt a component-based architecture, where applications are built by composing reusable and self-contained components
- Frameworks abstract away the low-level details of directly manipulating the Document Object Model (DOM).
- Many frameworks promote a declarative programming approach, where developers describe the desired outcome rather than specifying the step-by-step procedural logic.
- JavaScript frameworks often offer data binding capabilities, which establish a connection between the application state and the UI elements. They provide mechanisms to automatically update the UI when the underlying data changes and vice versa. This abstraction eliminates the need for manual DOM updates and synchronization, as the framework handles the reactive updates transparently.

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

- The most important part of learning a JavaScript (JS) framework is understanding its underlying concepts and principles such as Core Concepts, Documentation, Hands-on Practice, Real-World Projects, Community Engagement & Continuous Learning.