

# 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?

1. Performance: Direct DOM mutations can be more performant compared to replacing HTML because they only modify specific elements or attributes instead of recreating the entire HTML structure. This can lead to better overall performance and responsiveness, especially when dealing with large or complex DOM structures.

2. Granular control: Direct DOM mutations allow you to have fine-grained control over individual elements or attributes. You can target specific elements, modify their properties, update their text content, or add/remove classes or event listeners. This level of control can be useful when you need to make precise changes or interact with specific elements in the DOM.

---

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

1. DOM Manipulation: Frameworks abstract away the manual manipulation of the Document Object Model (DOM) elements. Instead of directly interacting with DOM APIs like `getElementById` or `createElement`, frameworks provide abstractions that handle the rendering, updating, and event binding of components or UI elements. This

simplifies the process of creating and manipulating the UI, reducing the amount of code required and improving developer productivity.

2. Event Handling: Frameworks often provide abstractions for event handling. Instead of manually attaching event listeners to DOM elements, frameworks provide mechanisms for declaratively binding event handlers to specific UI components or elements. This abstraction simplifies event management and allows developers to define event-driven behavior more intuitively.

---

3. What essence do JavaScript frameworks elevate?

- Productivity
- Organization
- Abstraction
- Performance
- Maintainability
- Reuseability

---

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

Most JavaScript frameworks achieve abstraction by providing higher-level abstractions and APIs that encapsulate complex underlying functionality and implementation details. Here are some common techniques and mechanisms used by JavaScript frameworks to achieve abstraction

---

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

When learning a JavaScript framework, the most important aspect is to understand the underlying concepts and principles on which the framework is built. Here are some key points to focus on when learning a JS framework:

1. Core Concepts: Gain a deep understanding of the core concepts and principles that the framework is based on. This includes understanding the framework's architecture, component structure, data flow, and key abstractions. It's crucial to grasp how the framework handles DOM manipulation, state management, event handling, and other fundamental operations.

2. Documentation and Guides: Consult the official documentation and guides provided by the framework's creators. The documentation provides valuable insights into the framework's features, APIs, best practices, and usage patterns. It often includes tutorials, examples, and explanations of key concepts. Referencing the documentation throughout your learning journey will help you understand the framework more effectively.