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**: **Making direct changes to the specific elements in the DOM** can be **faster and improves the responsiveness** of the web pages rather than directly replacing the HTML.

- **Flexibility**: When you **directly manipulate the DOM**, you have the freedom to apply your **own custom logic and make changes to the DOM elements**. It gives you the **flexibility to customize how elements behave and look on your webpage.**

- **Efficient updates**: When your webpage has content that frequently changes, **directly manipulating the DOM helps you only update the specific parts** of the page that you want to be changed **without affecting the rest of the page** and it **avoids unnecessary updates to other elements.**

- **Integration of external libraries**: When you use direct DOM mutations, it becomes **easier to integrate these libraries into your code**. These **libraries are specifically designed to work directly with the DOM.**

---

2. Wha**t low-level noise** do **JavaScript frameworks abstract away**?

- **DOM manipulation**: JavaScript frameworks simplify manipulating the DOM by **hiding the complex details**.

- **Event handling**: Frameworks simplify event handling by **easily attaching event listeners to handle events** without having to worry about **compatibility issues** across different browsers.

- **State management**: Frameworks abstract away the **complexity of managing application state across components or modules**.

- **Routing**: Frameworks handle the complexities of managing different page URLs and make it easier for developers to create navigable web apps.

---

3. What **essence** do **JavaScript frameworks elevate**?
Frameworks **elevate several essential aspects** of web development allowing developers to create **robust, scalable and interactive apps**.

- **Productivity**
- **Organization**
- **Abstraction**
- **Performance**
- **Maintainability**
- **Reusability**

---

4. Very broadly speaking, how do most **JS frameworks achieve abstraction**?
JS frameworks achieve abstraction by **simplifying tasks** and **hiding the implementation details using higher-level APIs, conventions and patterns.**They **encourage modularization through encapsulation** which allows developers to concentrate on **component functionality without being concerned about internal workings.**

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

The most important part of learning a JS framework is to **understand its core concepts and principles**. Although **each framework** has its **own unique features and syntax**, having a **solid understanding** of the **basic concepts** will allow you to work more **efficiently** with the framework and make the most of its capabilities. **Key aspects** to **focus on when learning a JS framework**:

- **Understanding** the frameworks **architecture and design patterns**.
- **Understanding** the concepts of **component-based development**.
- Learning about the **different options** for **state management**.
- **Mastering routing** and **navigation** within the framework.
- **Engaging** with the framework's **community** and utilizing **documentation**.

Learning JS frameworks is an ongoing journey that involves continuous practice, project-building and experimentation. By focusing on understanding the core concepts, you will develop the skills to adapt and effectively work with the framework and allow you to solve real-world problems and apply your knowledge in practical scenarios.