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

> **Faster Updates**: It's quicker because you only change what needs to be changed, not everything.
> **Quicker Responses**: Your website will react faster when people click or do things.
> Less Memory Used: It doesn't use as much computer memory because you're not creating lots of new things.
> **Better Control**: You can be more precise and make sure only the right things change.
> **Nice Animation**s: If you want things to move smoothly, it's better to change them directly.
> **Works Well with Other Stuff**: Sometimes, when you're using other tools to make websites, changing things directly works better.
> **Fewer Problems with How Things Look**: It helps to keep things looking good without making the computer work too hard.
> **Less Confusing Code**: You can write code that's easier to understand and less likely to have mistakes.

Changing only what you need in the web page can make your website faster and work better.

---

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

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

> **Making Things Look Nice**: Without a framework, you have to tell the computer exactly how to change the website's appearance when things happen, like a button click. Frameworks do this for you. You just say what you want, and the framework figures out how to make it happen.
>
> **Keeping Track of Changes**: When your website's information changes, like when you get a new message in a chat app, frameworks help you show that change on the screen without you having to remember which part of the screen to change. They do this tracking automatically.

Frameworks take away the hard work of making things look good and keeping track of changes, making it easier for you to build websites and apps.

---

3. What essence do JavaScript frameworks elevate?

> **Get More Done Faster**: They give you shortcuts and tools so you can make your website or app more quickly.
>
> **Keep Things Organized**: They help you keep your code neat and tidy, making it easier to understand and work on.
>
> **Grow Your Project**: You can add more stuff to your website or app without it becoming a mess.

Make It Work Well: They help your website or app run smoothly and quickly.

Work Everywhere: They make sure your website or app works the same way on different web browsers and devices.

Use What Others Made: You can borrow code and ideas from a big group of other people, making your work easier.

Stay Safe: They help protect your website or app from bad things happening, like hackers.

So, **JavaScript frameworks** make web development faster, tidier, and more reliable. They also let you tap into a big community of developers to learn from and share with.

---

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

Most **JavaScript frameworks** make it easier to build websites by **hiding the technical details** of how web pages work. They do this by using building blocks called components, keeping a copy of the web page in a computer's memory, and letting you describe how things should look and work in an easier way.

These frameworks also help you **connect the information on your web page** with what's happening behind the scenes, **like when you type something into a form or click a button**. They **make it simpler** to reuse pieces of your website, manage information, and make your site faster.

Overall, JavaScript frameworks make web development more straightforward by handling many complicated tasks for you.

---

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

The most important thing when learning a **JavaScript framework is to understand its main ideas and how it works**.

> **Components**: Know how to build and use pieces of your web application called "components." These are like building blocks for your website.
> **Data Handling**: Understand how the framework manages and updates data in your web app. Learn how to connect data to what you see on the screen.
> **Connecting Actions**: Learn how to make things happen when users click buttons or interact with your site. Make sure you can respond to what users do.
> **Page Navigation**: If your site has different pages or views, figure out how to move between them and send information back and forth.
> **Templates**: Learn how to create the visual part of your site using the framework's special way of writing things (like templates or JSX).
> **Best Ways to Do Things**: Find out the right and efficient ways to build with the framework by following guidelines and tips from experienced users.
> **Practice**: Work on real projects to put what you've learned into action. This helps you get better.
> **Stay Updated**: Keep an eye on updates and changes in the framework. It evolves, so you should keep learning.