

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?

When you replace the HTML every time a new element is added or updated it can cause some serious loading time issues, on a small scale it might not be noticeable but if you have a very larger website and all that changes is a checkbox then in order to update it the whole HTML is changed in order to accommodate for that change, where with DOM mutation one just simply updates that mutates part of the HTML

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

It can abstract things like event listeners or form submissions, anything really that is used in order to change the state of the DOM. Frameworks can be used to account and create abstractions for that.

3. What essence do JavaScript frameworks elevate?

In my understanding I'd say functional programming and overall immutability and scalability

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

Well looking at the definition of abstraction which is defining a set of rules/actions or properties and using it where it is needed, frameworks achieve this by defining what is and what needs to be done and creates those modular abstractions which can then be used in order to achieve functionality.

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

I'd say learning why it is the way that it is, instead of learning what to do with it one should look at how it performs and try to understand the structure of it. A good example would be learning how to tie a tie over using a pre-tied/clip on tie. You know how to use the tie and even where to put it but you don't know how it was made so when the pre-tied tie becomes undone you're stuck without knowing how to fix it. Same with frameworks where you get functional abstractions and you know how to use them and where to use them but if they stop working you won't know where to look or what to do in order to get it to work again.