JavaScript

1. Why is it important to write clean code?

**Implementing clean code principles is a foundational skill that pays off especially well when it's time to refactor code or bring code under test, it's  lead to source code that's highly modular and thus easier to read and test .**

2. What is the difference between good comments and bad comments?

**Comments are used to describe how and why the code works.**

**Explanatory comments are usually bad, but Comments that explain the solution are very important. They help to continue development the right way.**

3. What is an array?

**An array is an object that can store multiple values at once** .

4. What are arrays useful for?

**An array can hold many values under a single name, and you can access the values by referring to an index number.**

5. How do you access an array element?

**You can access elements of an array using indices (0, 1, 2 …). For example: (myArray[0])**

6. How do you change an array element?

**By writing the name of the array then the index value that you want to change it , then write the new value.**

7. What are some useful array properties?

**property is an association between a name (or key) and a value, such as : length,sort ,** **constructor, etc.**

8. What are some useful array methods?

**forEach() method, map() method, push() method, pop() method , etc.**

9. What are loops useful for?

**Loops are handy, if you want to run the same code over and over again, each time with a different value.**

10.What is the break statement?

**"jumps out" of a loop.**

11. What is the continue statement?

**"jumps over" one iteration in the loop.**

12.What is the DOM?

**Document Object Model, is constructed as a tree of Objects. It is a standard for how to get, change, add, or delete HTML elements.**  
13.How do you target the nodes you want to work with?

**By methods : getElementById or ByTagName or ByClassName.**

14.How do you create an element in the DOM?

**Document.createElement(elmnt)**

15.How do you add an element to the DOM?

**Document.appendChild(elmnt)**

16.How do you remove an element from the DOM?

**Document.removeChild(elmnt)**

17.How can you alter an element in the DOM?

**By innerHTML property**

18.When adding text to a DOM element, should you use textContent or innerHTML?

**textContent**

19.Where should you include your JavaScript tag in your HTML file when working with DOM nodes?

**In the <head> , or sometimes in the end of <body> , it depends.**

20.How do “events” and “listeners” work?

* **"Events" are any and all the activities performed on the web page,** **‘listener’ is the handler function that responds to the occurring event.**

21.What are three ways to use events in your code?

**HTML event handler attribute, element’s event handler property, and addEventListener()…………………………………..**

22.Why are event listeners the preferred way to handle events?

**It doesn’t require the use of a third-party library, it’s the most performance-friendly solution to add interactive functionality to HTML elements.**

23.What are the benefits of using named functions in your listeners?

**……………………………………..**

24.How do you attach listeners to groups of nodes?

**use the [querySelectorAll()](https://developer.mozilla.org/en-US/docs/Web/API/Document/querySelectorAll" \t "_blank) method, loop through the elements, and add an event listener to each one.**

25.What is the difference between the return values of querySelector and querySelectorAll?

**querySelectorAll() returns a NodeList instead of a single element as in querySelector().**

26.What does a “nodelist” contain?

**A NodeList object is a list (collection) of nodes extracted from a document.**

27.Explain the difference between “capture” and “bubbling”.

**With bubbling, the event is first captured and handled by the innermost element and then propagated to outer elements.**

**With capturing, the event is first captured by the outermost element and propagated to the inner elements.**

**Capturing is also called "trickling", which helps remember the propagation order.**

28.What is the difference between objects and arrays?

**javaScript Objects are mutable data structure in javascript which is used to represent a ‘Thing’. they are just collection of properties mapped to values.**

**Arrays store the data in an ordered collection in which the data can be accessed using a numerical index. They are also mutable and data can be modified at any index.**

**Every JavaScript array is an object but every object is not an array.**

29.How do you access object properties

**Dot property accessor: object.property**

**Square brackets property access: object['property']**

**Object destructuring: const { property } = object**