Assignment theory of JavaScript

1.What is JavaScript?

- JavaScript is a lightweight, cross-platform and interpreted complied programming language which is also known as the scripting language for web pages. Pure JavaScript is also called vanilaJS. It is also known as objective-based scripting language. It is well known for the development of webpages; many non-browser environments also use it.
- JavaScript is a scripting or programming language that allows you to implement complex features on web pages — every time a web page does more than just sit there and display static information for you to look at — displaying timely content updates, interactive maps, animated 2D/3D graphics, scrolling video jukeboxes, etc. — you can bet that JavaScript is probably involved. It is the third layer of the layer cake of standard web technologies, two of which (HTML and CSS) we have covered in much more detail in other parts of the Learning Area.

2.What is the use of is NaN function?

- isNaN() returns true if a number is Not-a-Number. In other words: isNaN() converts the value to a number before testing it.
- In JavaScript, NaN is short for "Not-a-Number". In JavaScript, NaN is a number that is not a legal number. The Number. isNaN() method returns true if the value is NaN , and the type is a Number.

3. What is negative Infinity?

- NEGATIVE_INFINITY is a special numeric value that is returned when an arithmetic operation or mathematical function generates a negative value greater than the largest representable number in JavaScript (i.e., more negative than -Number. MAX_VALUE). JavaScript displays the NEGATIVE_INFINITY value as -Infinity.
- The negative infinity in JavaScript is a constant value that is used to represent a value that is the lowest available. This means that no other number is lesser than this value.
- NEGATIVE_INFINITY returns negative infinity. Number. NEGATIVE_INFINITY is "a number lower than any other number".

4. Which company developed JavaScript?

- JavaScript was created at Netscape Communications by Brendan Eich in 1995. Netscape and Eich designed JavaScript as a scripting language for use with the company's flagship web browser, Netscape Navigator.
- "JavaScript" is a trademark of Oracle Corporation in the United States.

5. What are undeclared and undefined variables?

- Undefined: It occurs when a variable has been declared but has not been assigned any value. Undefined is not a keyword. Undeclared: It occurs when we try to access any variable that is not initialized or declared earlier using the var or const keyword.
- An undefined variable in the source code of a computer program is a variable that is accessed in the code but has not been declared by that code. In some programming languages, an implicit declaration is provided the first time such a variable is encountered at compile time.

6. Write the code for adding new elements dynamically?

- New elements can be dynamically created in JavaScript with the help of createElement() method. The attributes of the created element can be set using the setAttribute() method.

7. What is the difference between ViewState and SessionState?

- Maintained at session level. View state can only be visible from a single page and not multiple pages. Session state value availability is across all pages available in a user session. It will retain values in the event of a post back operation occurring.

8. What is === operator?

- The strict equality (===) operator checks whether its two operands are equal, returning a Boolean result. Unlike the equality operator, the strict equality operator always considers operands of different types to be different.

9. How can the style/class of an element be changed?

- To change the styles of all elements with a specific class:
  - Use the querySelectorAll() method to get a collection of the elements with the specific class.
  - Use the forEach() method to iterate over the collection.
  - On each iteration, use the style object to change the element's styles.

10. How to read and write a file using JavaScript?

- js fs module. The fs. readFile() and rs. writeFile() methods are used to read and write of a file using javascript.
- To read a file, use FileReader , which enables you to read the content of a File object into memory. You can instruct FileReader to read a file as an array buffer, a data URL, or text. // Check if the file is an image.

11. What are all the looping structures in JavaScript?

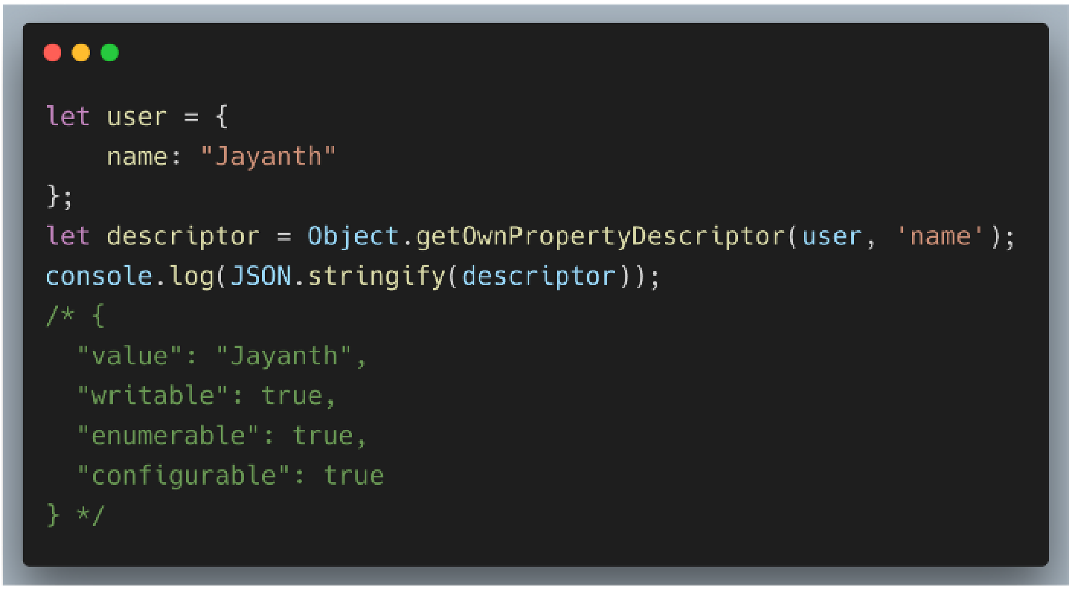
- The statements for loops provided in JavaScript are:
  - for statement.
  - do...while statement.
  - while statement.
  - labeled statement.
  - break statement.
  - continue statement.
  - for...in statement.
  - for...of statement.

12. How can you convert the string of any base to an integer in JavaScript?

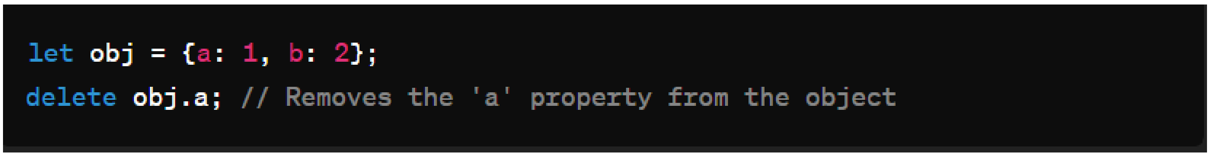
- There's a function called parseInt() in JavaScript, this is used for parsing a string as an argument and it returns an integer of the specified radix (basically the base of the numerical system) as output.

13. What is the function of the delete operator?

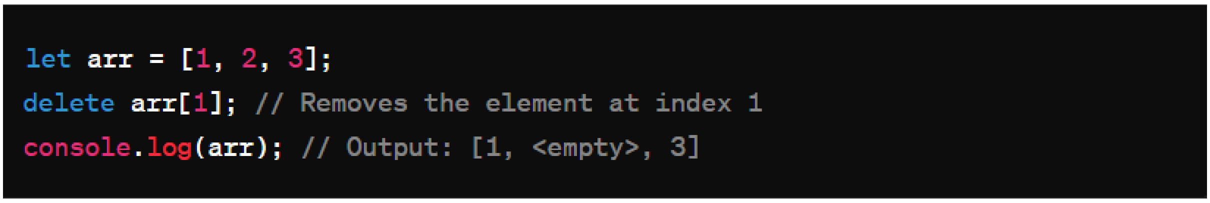
- The delete operator removes a property from an object. It can't delete a variable. Any property declared with var can't be deleted from the global scope or from a function's scope.



```
let user = {
    name: "Jayanth"
};
let descriptor = Object.getOwnPropertyDescriptor(user, 'name');
console.log(JSON.stringify(descriptor));
/* {
  "value": "Jayanth",
  "writable": true,
  "enumerable": true,
  "configurable": true
} */
```



```
let obj = {a: 1, b: 2};
delete obj.a; // Removes the 'a' property from the object
```



```
let arr = [1, 2, 3];
delete arr[1]; // Removes the element at index 1
console.log(arr); // Output: [1, <empty>, 3]
```

14. What are all the types of Pop-up boxes available in JavaScript?

- JavaScript has three kind of popup boxes: Alert box, Confirm box, and Prompt box.

15. What is the use of Void (0)?

- JavaScript void 0 means returning undefined (void) as a primitive value. You might come across the term “JavaScript:void(0)” while going through HTML documents. It is used to prevent any side effects caused while inserting an expression in a web page.

16. How can a page be forced to load another page in JavaScript?

- In JavaScript, we can use window. location object to force a page to load another page. We can use the location object to set the URL of a new page.

17. What are the disadvantages of using innerHTML in JavaScript?

- It is very slow because as inner HTML already parses the content even, we have to parse the content again so that's why it takes time. When we have used the event handlers then the event handlers are not automatically attached to the new elements created by innerHTML.

- While the innerHTML property in JavaScript is powerful and convenient for manipulating HTML content within elements, it also has some disadvantages:

  - Security Risks:

Using innerHTML to dynamically insert content from untrusted sources can expose your application to cross-site scripting (XSS) attacks. If the inserted content contains executable scripts, they will be executed in the context of your webpage, potentially compromising user security.Careful sanitization and validation of input data are necessary to mitigate this risk.

- Performance Overhead:

Manipulating innerHTML involves re-parsing and re-rendering the entire content of the element every time it's modified. This can lead to performance overhead, especially when dealing with large or complex HTML structures.In contrast, DOM manipulation methods like appendChild or createElement may offer better performance for certain tasks, as they only modify specific parts of the DOM tree.

- Event Handlers:

If elements with event handlers are replaced using innerHTML, the event handlers associated with those elements will be lost. This is because innerHTML recreates the entire content of the element, removing any previously attached event listeners.You would need to reattach event listeners after modifying the innerHTML, which can be cumbersome and error-prone.

- Limited Functionality:

innerHTML is primarily used for replacing or appending HTML content within an element. It doesn't provide fine-grained control over individual elements or their attributes.For more complex manipulations, such as modifying specific attributes or traversing the DOM tree, using DOM manipulation methods directly may be more appropriate.

- Compatibility Issues:

While widely supported in modern browsers, there may be subtle differences in how innerHTML is implemented across various browser versions.Additionally, older browsers or non-browser environments (like server-side JavaScript) may have limited or inconsistent support for innerHTML.

- ➢ In summary, while innerHTML is a convenient tool for dynamic HTML manipulation, it should be used judiciously, considering its security implications, performance considerations, and limitations compared to other DOM manipulation techniques.