Variables and Data Types

Question 1: What are variables in JavaScript? How do you declare a variable using var, let, and const?

Answer: In JavaScript, variables are used to store data values, which can be accessed or modified later.

- var: Declares a variable with function scope (older method, can be re-declared and updated).
- var x = 10;
- let: Declares a variable with block scope (preferred for modern code, can be updated but not re-declared in the same block).
- let y = 20;
- const: Declares a constant variable with block scope (cannot be updated or re-declared).
- const z = 30;

Question 2: Explain the different data types in JavaScript. Provide examples for each.

Answer: JavaScript has several data types, which can be categorized into primitive and non-primitive types.

1. String: Represents text.

let name = "John"; // Example

2. Number: Represents both integers and floating-point numbers.

let age = 25;   // Integer

let price = 19.99; // Float

3. Boolean: Represents true or false.

let isActive = true; // Example

4. Undefined: A variable that is declared but not assigned a value.

let x;

console.log(x); // Output: undefined

5. Null: Represents a deliberate absence of any value.

let person = null; // Example

6. Object: A collection of key-value pairs.

let user = { name: "Alice", age: 30 }; // Example

7. Symbol: A unique and immutable value (used for unique identifiers).

let sym = Symbol('id'); // Example

8. BigInt: Represents integers larger than Number.MAX_SAFE_INTEGER.

let bigNumber = 123456789012345678901234567890n; // Example

These are the main data types used in JavaScript.

Question 3: What is the difference between undefined and null in JavaScript?

Answer: In JavaScript, both undefined and null represent the absence of a value, but they are used in different contexts:

- **undefined: Means a variable is declared but hasn't been assigned a value yet.**
- **let x;**
- **console.log(x); // Output: undefined**
- **null: Represents an intentional absence of any object or value. It is explicitly assigned.**
- **let y = null;**
- **console.log(y); // Output: null**

So, undefined is a default value, while null is explicitly set to indicate no value.