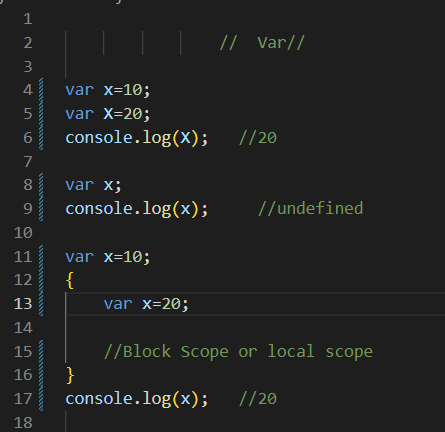
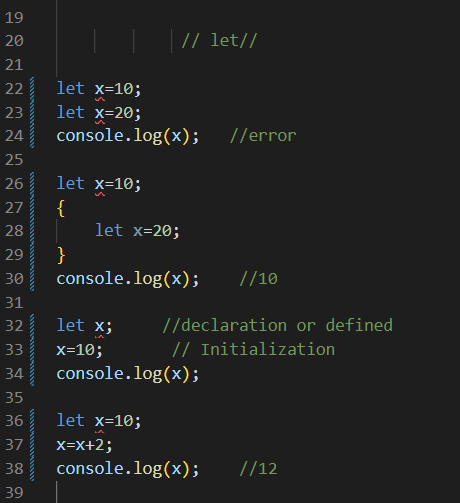
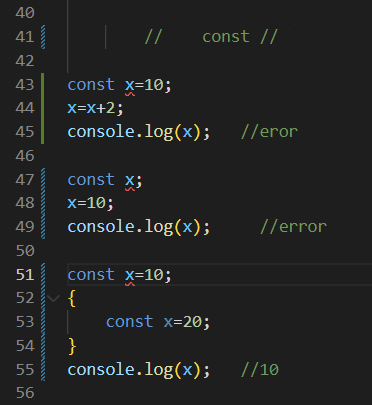
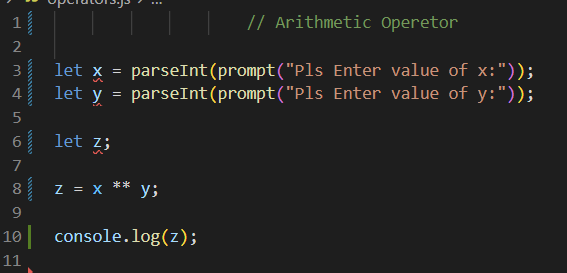
1. Write a JS program that demonstrate all examples of var, let and const variable declaration.

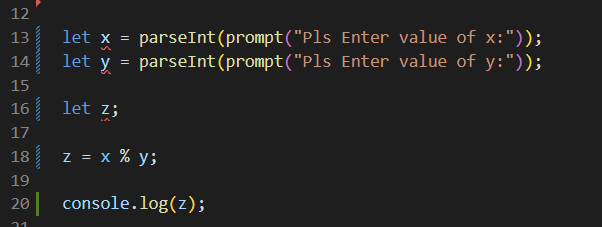


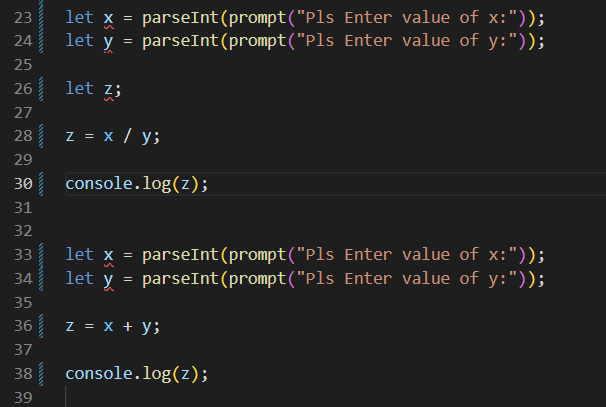


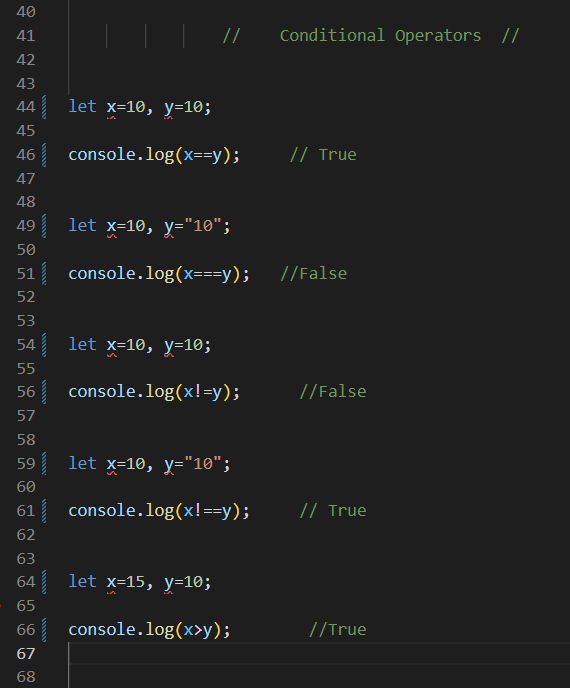


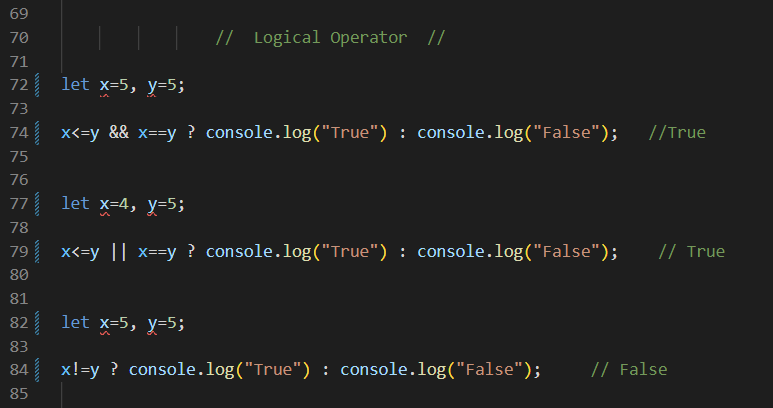
1. Write a JS program that demonstrate all examples of different operators.

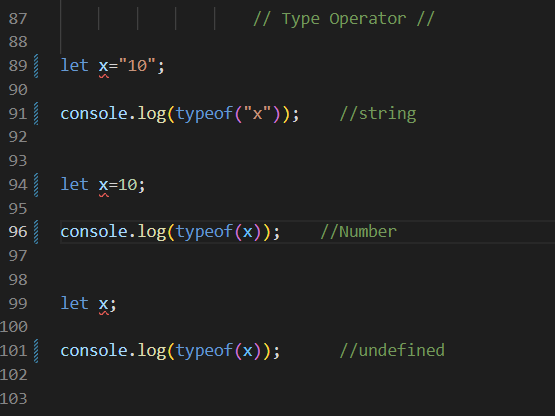












* **Interview Questions:**

1. What is JavaScript?

Ans:

JavaScript is a programming language that is commonly used for creating interactive and dynamic websites. It is a high-level language, which means that it is designed to be easy for humans to read and write. JavaScript is also a client-side language, which means that it runs on the user's computer, rather than on a server.

JavaScript is often used in combination with HTML and CSS, the other two languages that make up the core of web development. With JavaScript, developers can add functionality to web pages, such as interactive forms, animations, and user interface components. It can also be used to create complex web applications, such as online games, social networks, and productivity tools.

JavaScript is supported by all major web browsers, including Chrome, Firefox, Safari, and Internet Explorer, making it a popular choice for web developers. Additionally, JavaScript has a large and active community of developers who contribute to libraries and frameworks that make it easier to build complex applications.

1. What is advantages and disadvantages of JavaScript?

Ans :

|  |  |
| --- | --- |
| **Advantages of JavaScript:** | **Disadvantages of JavaScript:** |
| Interactivity: JavaScript makes websites more interactive by allowing developers to create dynamic user interfaces, add animations and effects, and respond to user input without having to reload the entire page. | Security concerns: JavaScript code can be vulnerable to attacks such as cross-site scripting (XSS) and cross-site request forgery (CSRF), which can compromise the security of a website or application. |
| Versatility: JavaScript can be used both on the client-side (in the user's browser) and on the server-side (using Node.js), which makes it a versatile language for building a wide range of applications. | Browser compatibility issues: JavaScript may behave differently across different web browsers, which can make it difficult to create consistent user experiences. |
| Large community: JavaScript has a large and active community of developers who contribute to open source libraries and frameworks, making it easier for developers to build complex applications.  Easy to learn: JavaScript is relatively easy to learn compared to other programming languages, which makes it accessible to beginners who are just starting to learn how to code. | Performance limitations: JavaScript performance can be slower compared to compiled languages like Java and C++, especially when dealing with large amounts of data or complex algorithms.  Lack of type checking: JavaScript is a dynamically-typed language, which means that it does not perform type checking at compile time. This can lead to errors and bugs that are difficult to catch during development. |
| Fast development: JavaScript allows for fast development and prototyping, as it can be added to existing HTML and CSS files without requiring a separate compilation step. | Accessibility concerns: JavaScript can be a barrier to accessibility for users who rely on screen readers or other assistive technologies, as it can interfere with the normal behaviour of these tools. |

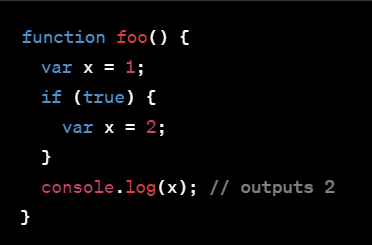
3. What is the purpose of the let keyword?

Ans:

The let keyword is used to declare block-scoped variables in JavaScript. Prior to the introduction of let in ECMAScript 6 (ES6), the only way to declare variables in JavaScript was to use the var keyword, which declared variables at the function scope or global scope.

The let keyword allows developers to declare variables that are only accessible within a specific block of code, such as inside a loop or a conditional statement. This can help to prevent bugs and improve code clarity, as it makes it clear which variables are only intended to be used within a specific scope.

**For example, consider the following code:**

****

4.Give the difference between var, let and const.

Ans:

|  |  |  |
| --- | --- | --- |
| ‘var’ | ‘let’ | ‘const’ |
| ‘var’ is function scoped, meaning that the variable declared with var is only accessible inside the function in which it is declared, or globally if it is declared outside of any function. var can be redeclared and reassigned within its scope, and it is also hoisted to the top of its scope. | ‘let’ and ‘const’ are block scoped, meaning that they are only accessible within the block in which they are declared, including for-loops, if-statements, and functions. let can be reassigned but cannot be redeclared within its scope. const, on the other hand, cannot be reassigned or redeclared once it has been assigned a value. | ‘const’ and ‘let’ are not hoisted to the top of their scope. Therefore, if you try to access them before they are declared, you will get a Reference Error. |
| Use ‘var’ when you need to declare a variable that is accessible within the function or globally. | Use ‘let’ when you need to declare a variable that has a limited scope and can be reassigned. | Use const when you need to declare a variable that has a limited scope and should not be reassigned. |

1. What is the difference between =, == and === operator?

Ans**:**

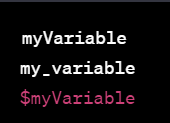
|  |  |  |
| --- | --- | --- |
| **‘=’** | **‘==’** | **‘===’** |
| = operator: assigns a value to a variable. For example, x = 5 assigns the value 5 to the variable x. | == operator: compares two values for equality, but performs type coercion if the values have different types. For example, 1 == "1" is true because the string "1" is converted to the number 1 before the comparison. | === operator: compares two values for equality, but does not perform type coercion. It checks both the value and the type of the operands. For example, 1 === "1" is false because the two operands have different types. |
| Use = operator to assign a value to a variable. | Use == operator to compare two values for equality, with type coercion. | Use === operator to compare two values for equality, without type coercion. |

1. What is identifier? Give the rules to declare identifier.

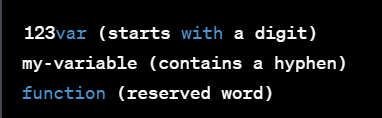
Ans:

* An identifier is a name given to a variable, function, or any other user-defined item. It is used to refer to that item in the code. Identifiers must follow certain rules for their declaration:
* An identifier can only contain letters (both uppercase and lowercase), digits, and underscores (\_).
* An identifier must begin with a letter, underscore, or dollar sign ($). It cannot begin with a digit.
* Identifiers are case sensitive, meaning that ‘myVariable’ and ‘myvariable’ are two different identifiers.
* Identifiers cannot be reserved words, such as ‘if’, ‘while’, ‘for’, ‘function’, and so on.
* Identifiers should be meaningful and descriptive of the item they refer to. For example, ‘firstName’ is a more descriptive identifier than ‘fn’.

**Examples of valid identifiers:**

****

**Examples of invalid identifiers:**

****

1. List features of JavaScript.

Ans**:**

JavaScript is a powerful and versatile programming language that is widely used for web development. Here are some of its features:

* Dynamic and loosely typed: JavaScript is a dynamic language, meaning that variables can change their type during runtime. It is also loosely typed, allowing variables to be assigned values of different types.
* Client-side scripting: JavaScript runs on the client side of web applications, allowing it to interact with the user and modify the page dynamically.
* Event-driven: JavaScript is event-driven, meaning that it can respond to events such as clicks, input, and page loads.
* Functional programming: JavaScript supports functional programming concepts such as higher-order functions and closures.
* Object-oriented programming: JavaScript is also object-oriented, allowing for the creation of objects with properties and methods.
* Easy to learn: JavaScript has a relatively simple syntax and is easy to learn, making it a popular choice for beginners.
* Cross-platform: JavaScript can run on multiple platforms and devices, including desktops, mobile devices, and servers.
* Large ecosystem: JavaScript has a vast ecosystem of libraries, frameworks, and tools that make it easy to develop complex applications.
* Asynchronous programming: JavaScript supports asynchronous programming through callbacks, promises, and async/await, allowing for non-blocking I/O operations and better performance.
* To summarize, JavaScript is a versatile language that combines both functional and object-oriented programming paradigms and supports asynchronous programming, making it an ideal choice for building dynamic and responsive web applications.