**JavaScript Assignment**

* **JavaScript Introduction:**

1. **What is JavaScript? Explain the role of JavaScript in web development.**

**Ans) JavaScript** is a scripting language used to create interactive and dynamic content on websites.

Role in Web Development:  
JavaScript enables user interaction (like clicks, form submissions), updates content without reloading the page, and works with HTML/CSS to build modern, responsive web apps.

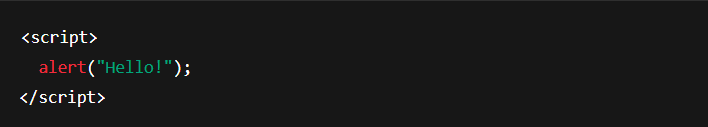
1. **How is JavaScript different from other programming languages like Python or Java?**

**Ans)** JavaScript runs mainly in web browsers for interactive web pages, while Python and Java are general-purpose languages often used for backend development, data science, or mobile apps. JavaScript is event-driven and loosely typed, whereas Python is known for readability and Java for strict object-oriented structure.

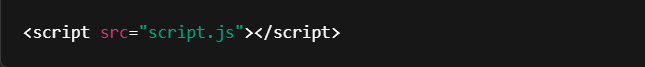
1. **Discuss the use of <script> tag in HTML. How can you link an external JavaScript file to an HTML document?**

**Ans)** The <script> tag in HTML is used to embed or reference JavaScript code in a web page.

To write JavaScript inside HTML:

****

To link an external JavaScript file:

****

Place the <script> tag before the closing </body> tag for better performance.

* **Variables and Data Types:**

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

**Ans)** Variables in JavaScript store data values that can be used and changed during program execution.

Declaration methods:

* var – old way, function-scoped

var name = "John";

* let – modern way, block-scoped

let age = 25;

* const – block-scoped, value can't be reassigned

const PI = 3.14;

Use let and const for cleaner, more predictable code.

1. **Explain the different data types in JavaScript. Provide examples for each.**

**Ans)** JavaScript has 8 main data types, divided into primitive and non-primitive:

🔹 Primitive Data Types:

1. String – Text data

let name = "Alice";

1. Number – Numeric values

let age = 30;

1. Boolean – True or false

let is Online = true;

1. Undefined – Variable declared but not assigned

let x;

1. Null – Intentional absence of any value

let y = null;

1. BigInt – For very large integers

let bigNum = 12345678901234567890n;

1. Symbol – Unique and immutable value

let id = Symbol("id");

🔹 Non-Primitive Data Type:

1. Object – Collection of key-value pairs

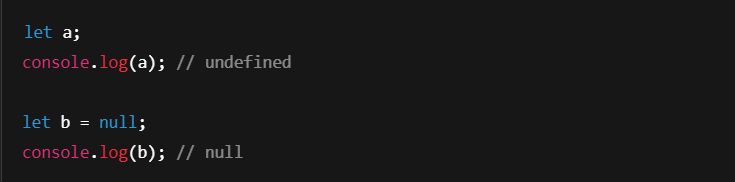
let person = { name: "Alice", age: 30 };

These types help JavaScript handle various kinds of data in programs.

1. **What is the difference between undefined and null in JavaScript?**

**Ans)** undefined means a variable has been declared but hasn't been assigned a value.  
null is an assigned value that represents "no value" or "nothing."

🔸 Example:

****

Key Difference:

* undefined is set by JavaScript by default.
* null is set by the developer intentionally.
* **JavaScript Operators:**
  1. **What are the different types of operators in JavaScript? Explain with examples. Arithmetic operators, Assignment operators, Comparison operators and Logical operators.**

**Ans)** In JavaScript, operators are used to perform operations on variables and values. Here's a breakdown of the main types of operators you asked about, along with examples:

* + 1. **Arithmetic Operators**

These operators perform basic mathematical operations.

| Operator | Description | Example | Result |
| --- | --- | --- | --- |
| + | Addition | 5 + 2 | 7 |
| - | Subtraction | 5 - 2 | 3 |
| \* | Multiplication | 5 \* 2 | 10 |
| / | Division | 10 / 2 | 5 |
| % | Modulus (Remainder) | 10 % 3 | 1 |
| ++ | Increment | let a = 1; a++ | 2 |
| -- | Decrement | let a = 2; a-- | 1 |

* + 1. **Assignment Operators**

These assign values to variables.

| Operator | Description | Example | Equivalent To |
| --- | --- | --- | --- |
| = | Assign | x = 10 | — |
| += | Add and assign | x += 5 | x = x + 5 |
| -= | Subtract and assign | x -= 2 | x = x - 2 |
| \*= | Multiply and assign | x \*= 3 | x = x \* 3 |
| /= | Divide and assign | x /= 2 | x = x / 2 |
| %= | Modulus and assign | x %= 4 | x = x % 4 |

* + 1. **Comparison Operators**

Used to compare two values and return a Boolean (true or false).

| **Operator** | **Description** | **Example** | **Result** |
| --- | --- | --- | --- |
| == | Equal to (loose equality) | 5 == '5' | true |
| === | Strict equal (type + value) | 5 === '5' | false |
| != | Not equal | 5 != 3 | true |
| !== | Strict not equal | 5 !== '5' | true |
| > | Greater than | 5 > 3 | true |
| < | Less than | 5 < 10 | true |
| >= | Greater than or equal to | 5 >= 5 | true |
| <= | Less than or equal to | 4 <= 5 | true |

* + 1. **Logical Operators**

Used to combine or invert Boolean values.

| **Operator** | **Description** | **Example** | **Result** |
| --- | --- | --- | --- |
| && | Logical AND | true && false | false |
| ` |  | ` | Logical OR |
| ! | Logical NOT | !true | false |

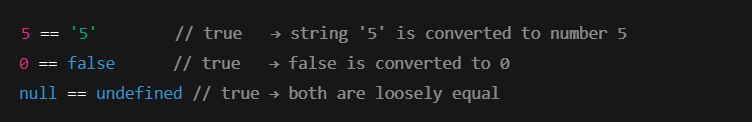
* 1. **What is the difference between == and === in JavaScript?**

**Ans)** Here's the difference between == and === in JavaScript presented in a clear tabular format:

**== (Loose Equality)**

* Compares **values only**.
* **Performs type coercion** (i.e., converts different types to the same type before comparing).

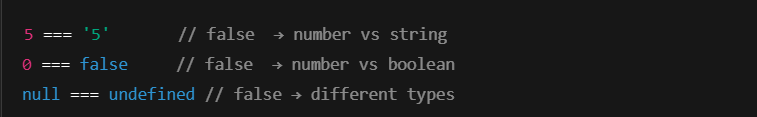
✅ **Examples:**



**=== (Strict Equality)**

* Compares **both value and type**.
* **No type conversion** is done.

❌ **Examples:**



* **Control Flow (If-Else, Switch):**

1. **What is control flow in JavaScript? Explain how if-else statements work with an example.**

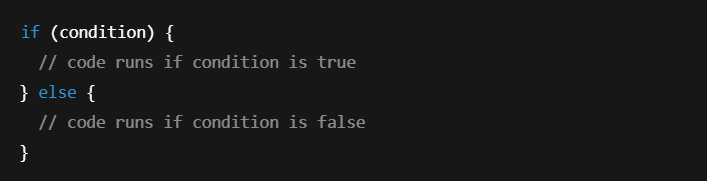
**Ans) What is Control Flow in JavaScript?**

Control flow is the order in which the code is executed in a program.  
By default, JavaScript runs code from top to bottom, line by line — but control flow statements like if-else, switch, for, and while allow you to change this flow, enabling decision-making, loops, and conditional execution.

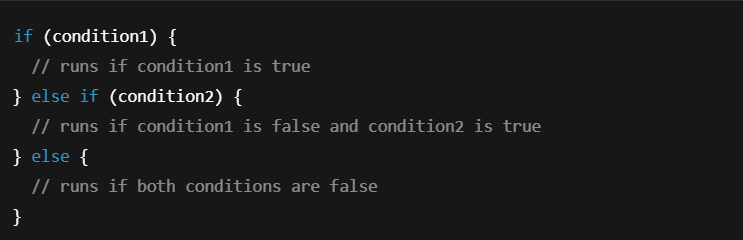
**🔹 if-else Statements in JavaScript**

if-else is used to execute different blocks of code depending on whether a condition is **true** or **false**.

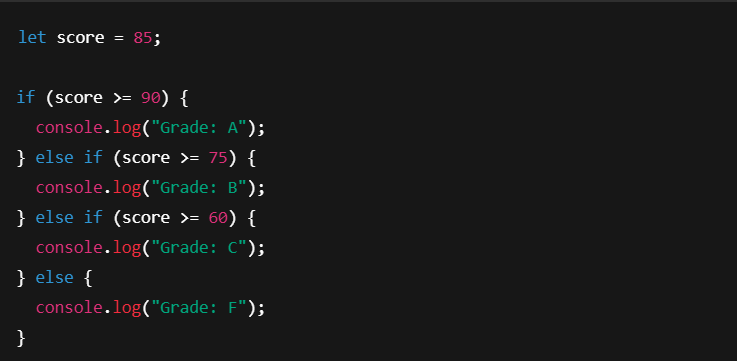
**🧩 Syntax:**

****

**🔄 You can also use else if for multiple conditions:**



**✅ Example:**

****

**💡 Output:**

****

1. **Describe how switch statements work in JavaScript. When should you use a switch statement instead of if-else?**

**Ans) What is a switch Statement in JavaScript?**

A switch statement is a control flow structure used to compare a single value against multiple possible cases. It is an alternative to writing many if-else if statements.

**✅ Syntax:**

****

**🔄 When to Use switch Instead of if-else?**

| Use switch when: | Use if-else when: |
| --- | --- |
| You’re checking one variable against many constant values | You have complex conditions (e.g., ranges or logical expressions) |
| The values are discrete and known | Conditions involve comparisons like <, >, or &&, ` |
| You want cleaner, more readable code for many options | You need to evaluate expressions, not just values |