**Ways to implement JavaScript**

There are three primary ways to implement JavaScript in a web page:

**1. External JavaScript**

* This is the most common and recommended method for larger projects and is considered a best practice.
* You write your JavaScript code in a separate file with a .js extension, and then you link to that file from your HTML document using the <script> tag with the src attribute.

**Why use it?**

* **Organization:** It keeps your JavaScript, HTML, and CSS files separate, making your code cleaner and easier to manage.
* **Reusability:** You can use the same .js file across multiple HTML pages without having to copy and paste the code.
* **Performance:** The browser can cache the external file, which means it doesn't have to download it again on subsequent page visits, leading to faster load times.

**2. Internal JavaScript**

* With this method, you write your JavaScript code directly inside the <script> tags within the HTML file.
* This is useful for small, simple scripts that are only needed on a single page.
* **Where to place it?**
  + You can place the <script> tag in the <head> or the <body>.
  + The best practice is to place it just before the closing </body> tag.
  + This ensures the HTML content is fully loaded and rendered before the JavaScript runs, preventing potential errors if the script tries to manipulate an element that doesn't exist yet.

**3. Inline JavaScript**

* This method involves embedding JavaScript code directly into an HTML element's attribute.
* It's often used with event handlers like onclick or onmouseover.

**Variables in JavaScript**

* Variables are named containers that store data values in memory.
* They act as references to manipulate information throughout your code.
* JavaScript provides three ways to declare variables: var, let, and const.

**Variable Declaration Rule -**

* Variable name start with alphabet,alphanum.
* Never use reserve keyword for variable declaration

Ex. this, for, if, else, return etc.

* Variable name is a combination of letters, numbers or special character (only two character allowed $ or \_)
* Variable doesn’t start with numbers.

**Key Concept -**

**Declaration**

Declare means creating a variable’s name in your code without assigning a value

Ex.

let age;

let fullname;

**Assignment**

Assign means providing a value to a previously declared variable.

Ex.

age = 30

fullname = “Virat Kohli”

**Initialize**

Initialize means declaring a variable and assigning a value at same time

Ex.

const email = [virat@gmail.com](mailto:virat@gmail.com)

**var Keyword –**

1. **Variable declaration start with var**

var age = 25,   var name = “virat”

1. **we can redeclare and reassign the variable using var keyword in any scope**

var age = 25,       var age = 30     - - - - no error

**let keyword –**

1. **Variable declaration start with let**

let age = 25,   let name = “virat”

1. **we can not redeclared the variable with same name using let keyword in same scope (SyntaxError)**

let age = 25           let age = 30  - - - error (SyntaxError)

1. **we can reassign the variable.**

let age = 25           age = 30       - - - reassign (no error)

1. **let keyword support block scope  {  }**

**const keyword –**

1. **Variable declaration start with const**

const age = 25,   const name = “virat”

1. **we can not redeclared and reassign the variable with same name using const keyword  in same scope**

let age = 25           let age = 30  - - - error (SyntaxError)

let name = “virat”     name = “ms” - - - error (TypeError)

1. **const keyword support block scope  {   }**