**Group assignment:**

**Members**

**1.Sivalarunan**

**2. Juganya K (UKI STU 0871)**

1. Variable is a named storage for data, which gold values and can be manipulated by the program.
2. In JS you can use these ‘var’,’ let’ and ‘const’ keywords to declare variables
3. var: Function-scoped, hoisted, can be re-declared and updated.

let: Block-scoped, not hoisted for initialization, cannot be re-declared within the same scope, but can be updated.

const: Block-scoped, not hoisted for initialization, cannot be re-declared or updated.

1. Variable hoisting in JavaScript is a behaviour in which variable and function declarations are moved to the top of their containing scope (global or function) during the compile phase before the code execution. This means that you can use a variable or a function before it is declared in the code.
2. Function Scope: Variables declared with var are scoped to the function in which they are declared. If they are declared outside of any function, they are globally scoped.

Block Scope: Variables declared with let are block-scoped, meaning they are only accessible within the block (delimited by {}) in which they are defined.

Like let, variables declared with const are block scoped.

Global Scope: var is globally scoped if declared outside a function.

1. Template literals are enclosed by backticks (`) to create strings that are easier to read, write, and maintain compared to traditional string concatenation methods
2. Number: Integers and floating-point numbers.

String: Textual data.

Boolean: true or false.

Undefined: Uninitialized variables.

Null: Intentional absence of any value.

Symbol: Unique, immutable identifiers.

BigInt: Large integers with arbitrary precision.

1. Both represents the absence of actual value but undefined means the variable was declared but not initialized and Null is assigned as value that explicitly indicates that a variable has no value
2. Using the ‘typeof’ operator, it will return a string of variables type.
3. Primitive Types: Immutable, stored directly in memory, accessed by value.

Reference Types: Mutable, stored as references to memory locations, accessed by reference.

1. Type coercion in JavaScript refers to the automatic or implicit conversion of values from one data type to another during the evaluation of expressions. This process can happen implicitly by JavaScript or explicitly by the programmer
2. The typeof operator is used to determine the type of a primitive value or an object. It returns a string indicating the type of the unevaluated operand.

The instanceof operator is used to check if an object is an instance of a specific constructor or class. It checks the prototype chain to see if an object is derived from a constructor's prototype.

1. Use Number() when you need a straightforward conversion and are confident the string is a valid number.

Use parseInt() when you need to convert a string to an integer and optionally specify a radix.

Use parseFloat() when you need to convert a string to a floating-point number.

Use the unary plus (+) operator for a concise and quick conversion.

1. **String()**: Use this for straightforward conversion when you don’t need any specific formatting.

**toString()**: Use this when you want to convert a number to a string, especially if need to represent the number in a different base.

1. Implicit type conversion, also known as type coercion, is a feature in JavaScript where the language automatically converts values from one data type to another during the evaluation of expressions.
2. Number()

let str1 = "123"; let num1 = Number(str1); console.log(num1);

parseInt()

let str1 = "123"; let num1 = parseInt(str1); console.log(num1);

plus()

let str1 = "123"; let num1 = +str1; console.log(num1);

1. Convert the string to a number using the Number() function before adding

Use parseInt() for integer conversion or parseFloat() for floating-point number conversion

The unary plus operator (+) is a concise way to convert a string to a number.

1. The parseInt() function parses a string argument and returns an integer

The parseFloat() function parses a string argument and returns a floating-point number

1. Using Array Literals

let fruits = ["Apple", "Banana", "Cherry"]; console.log(fruits);

Using the Array Constructor

let fruits = new Array("Apple", "Banana", "Cherry"); console.log(fruits);

1. An object is a standalone entity, with properties and type. It's a collection of key-value pairs where the keys are strings or symbols, and the values can be any data type