TASK1: Explain the role of operators in JavaScript. Why are they essential in programming?

SOLUTION:

Operators in JavaScript are symbols that perform operations on operands. Operands can be variables, values, or expressions. Operators are essential in programming because they allow us to manipulate data and perform calculations

1. perform Arithmetic operation

2.Assign operator

3.Compare Operator

4.Perform logical operations

TASK2: Describe the categorization of operators in JavaScript based on their functionality. Provide examples for each category.

1.Arithmetic operator:

Ex :-

// Addition (+)

const sum = 1 + 2;

// sum is now equal to 3

// Subtraction (-)

const difference = 10 - 5;

// difference is now equal to 5

2.Assignment operator:

Ex:-

// Assignment operator (=)

const myVariable = 10;

// Addition assignment operator (+=)

myVariable += 5;

// myVariable is now equal to 15

3.Comparison operator:

Ex:-

// Equal to (==)

const isEqual = 10 == 10;

// isEqual is now equal to true

// Not equal to (!=)

const isNotEqual = 10 != 10;

// isNotEqual is now equal to false

4.Logical operator:

Ex:-

// AND (&&)

const isAndTrue = true && true;

// isAndTrue is now equal to true const isAndFalse = true && false;

// isAndFalse is now equal to false

// OR (||)

const isOrTrue = true || false;

// isOrTrue is now equal to true const isOrFalse = false || false;

// isOrFalse is now equal to false

TASK:3 Differentiate between unary, and binary operators in JavaScript. Give examples of each.

SOLUTION:

Operators in JavaScript are categorized based on the number of operands they require into the following group

1.Unary operators Unary operators operate on a single operand

Ex:- Increment operator(++),logicalNOT(!)

2.Binary operators Binary operators operate on two operands.

Ex:-Equal (==) Not equal (!=) Less than (<) Greater than (>) Greater than or equal to (>=) Less than or equal to (<=)

TASK:4 Discuss the precedence and associativity of operators in JavaScript. Why is understanding these concepts important?

SOLUTION:

Precedence refers to the order in which operators are evaluated when multiple operators are present in an expression. Operators with higher precedence are executed first. For example, in the expression 3 + 5 \* 2, multiplication (\*) has higher precedence than addition (+), so the multiplication is performed first.

Associativity refers to the order in which operators of the same precedence are evaluated. Some operators associate left to right, meaning they are evaluated from left to right. Others associate right to left, meaning they are evaluated from right to left.

TASK:5 Write a JavaScript program that calculates the simple interest using the formula Simple interest = (principal \* rate \* time) / 100

SOLUTION:

const principal=1000

const rate= 5

const time=2

const result=(principal\*rate\*time)/100

console.log("simple interest=",result)

TASK:6 Write a Javascript program to calculate the Body Mass Index (BMI) using the formula BMI = weight (kg)/ height \* height.

SOLUTION:

const weight=100

const height= 6

const result= weight/(height\*height)

console.log("body mass index=",result)