Report:

JavaScript Operators and Conditional Statements

# 1. JavaScript Operators:

Operators in JavaScript are used to perform operations on variables and values.

## a) Arithmetic Operators:

+ (Addition), - (Subtraction), \* (Multiplication), / (Division), % (Modulus), ++ (Increment), -- (Decrement)

## b) Assignment Operators:

= (Assign), +=, -=, \*=, /=, %= (Compound assignments)

## c) Comparison Operators:

== (Equal), === (Strict Equal), != (Not Equal), !== (Strict Not Equal), >, <, >=, <=

## d) Logical Operators:

&& (Logical AND), || (Logical OR), ! (Logical NOT)

## e) Bitwise Operators:

&, |, ^, ~, <<, >>, >>> (Operate on binary representations)

## f) String Operators:

+ (Concatenation), += (Concatenate and assign)

## g) Type Operators:

typeof (Returns the type of a variable), instanceof (Checks object type)

# 2. JavaScript Conditional Statements:

Conditional statements control the flow of execution based on different conditions.

## a) if Statement:

Executes a block of code if the condition is true.

Example:

let age = 18;  
if (age >= 18) {  
 console.log("You are an adult.");  
}

## b) if...else Statement:

Executes one block of code if the condition is true, and another if it is false.

Example:

let score = 50;  
if (score >= 60) {  
 console.log("Passed");  
} else {  
 console.log("Failed");  
}

## c) else if Statement:

Allows multiple conditions to be checked in sequence.

Example:

let marks = 85;  
if (marks >= 90) {  
 console.log("A Grade");  
} else if (marks >= 75) {  
 console.log("B Grade");  
} else {  
 console.log("C Grade");  
}

## d) switch Statement:

Evaluates an expression and matches it against multiple case values.

Example:

let day = 2;  
switch(day) {  
 case 1:  
 console.log("Monday");  
 break;  
 case 2:  
 console.log("Tuesday");  
 break;  
 default:  
 console.log("Another day");  
}

# Conclusion:

Operators help in performing actions on data, while conditional statements help in decision-making in programs. Both are essential for creating dynamic and functional JavaScript applications.