JavaScript Operators

Question 1: What are the different types of operators in JavaScript? Explain with examples. • Arithmetic operators • Assignment operators • Comparison operators • Logical operators

Ans.

JavaScript employs various operators to perform operations on data. Here are explanations and examples of the specified types:

1. Arithmetic Operators:
   Arithmetic operators perform mathematical calculations on numerical values.
   let a = 10;
    let b = 5; let sum = a + b; // Addition: 15
   let difference = a - b; // Subtraction: 5
   let product = a * b; // Multiplication: 50
   let quotient = a / b; // Division: 2
   let remainder = a % 3; // Modulus (remainder of division): 1
   let exponent = b ** 2; // Exponentiation (5 to the power of 2): 25

2. Assignment Operators :
   Assignment operators are used to assign values to variables. They can also combine an arithmetic operation with an assignment.

   let x = 10; // Simple assignment
   x += 5; // Equivalent to: x = x + 5; (x is now 15)
   x -= 3; // Equivalent to: x = x - 3; (x is now 12)
    x *= 2; // Equivalent to: x = x * 2; (x is now 24)
   x /= 4; // Equivalent to: x = x / 4; (x is now 6)
   x %= 5; // Equivalent to: x = x % 5; (x is now 1)

3. <u>Comparison Operators:</u>

   Comparison operators compare two values and return a Boolean (true or false) result.

   ```
   let p = 7;
   let q = 10;
   let r = "7";
   console.log(p == r); // Loose equality: true (compares value only)
   console.log(p === r); // Strict equality: false (compares value and type)
   console.log(p != q); // Not equal: true
   console.log(p !== r); // Strict not equal: true
   console.log(p < q); // Less than: true
   console.log(q > p); // Greater than: true
   console.log(p <= 7); // Less than or equal to: true
   console.log(q >= 10); // Greater than or equal to: true
   ```

4. <u>Logical Operators:</u>

   Logical operators combine or modify Boolean expressions.

   ```
   let isAdult = true;
   let hasLicense = false;

   // Logical AND (&&): Returns true if both operands are true
   console.log(isAdult && hasLicense); // false

   // Logical OR (||): Returns true if at least one operand is true
   console.log(isAdult || hasLicense); // true

   // Logical NOT (!): Inverts the Boolean value of the operand
   console.log(!isAdult); // false
   ```

Question 2: What is the difference between == and === in JavaScript?

Ans.

== (Loose Equality) :

• Compares values only, not data types.

• If the types are different, JavaScript automatically converts (coerces) one or both values to a common type before comparing.

Example:

5 == '5' // true → because '5' (string) is converted to 5 (number)

0 == false // true → false is converted to 0

null == undefined // true → they are loosely equal

=== (Strict Equality)

• Compares both value and data type.

• No type conversion is done — both must be identical in type and value to return true.

Example:

5 === '5' // false → number is not the same type as string

0 === false // false → number is not the same type as Boolean

 null === undefined // false → different types

5 === 5 // true → same value and same type