

6

OPERATORS

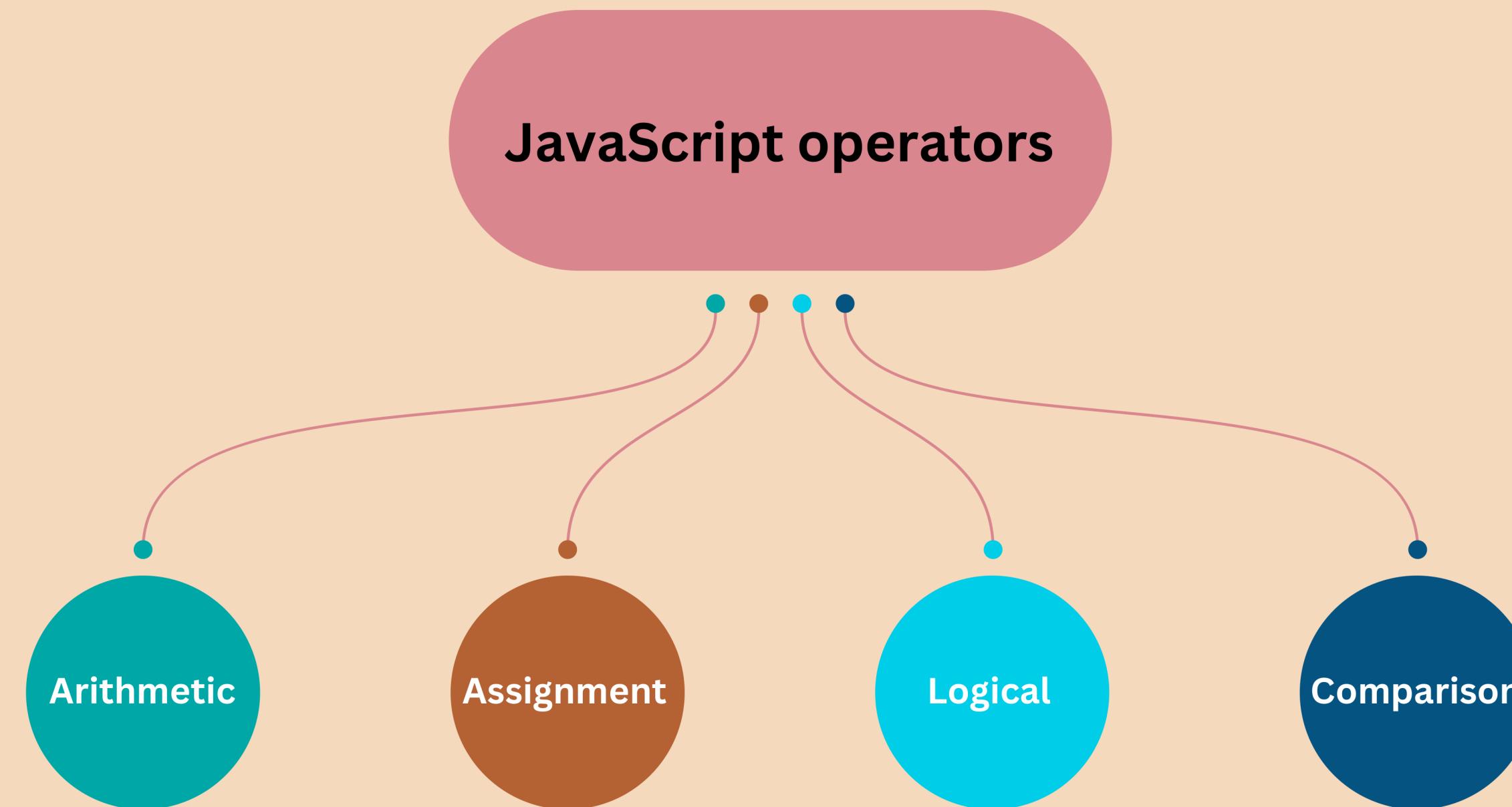
JS

WHAT TO EXPECT

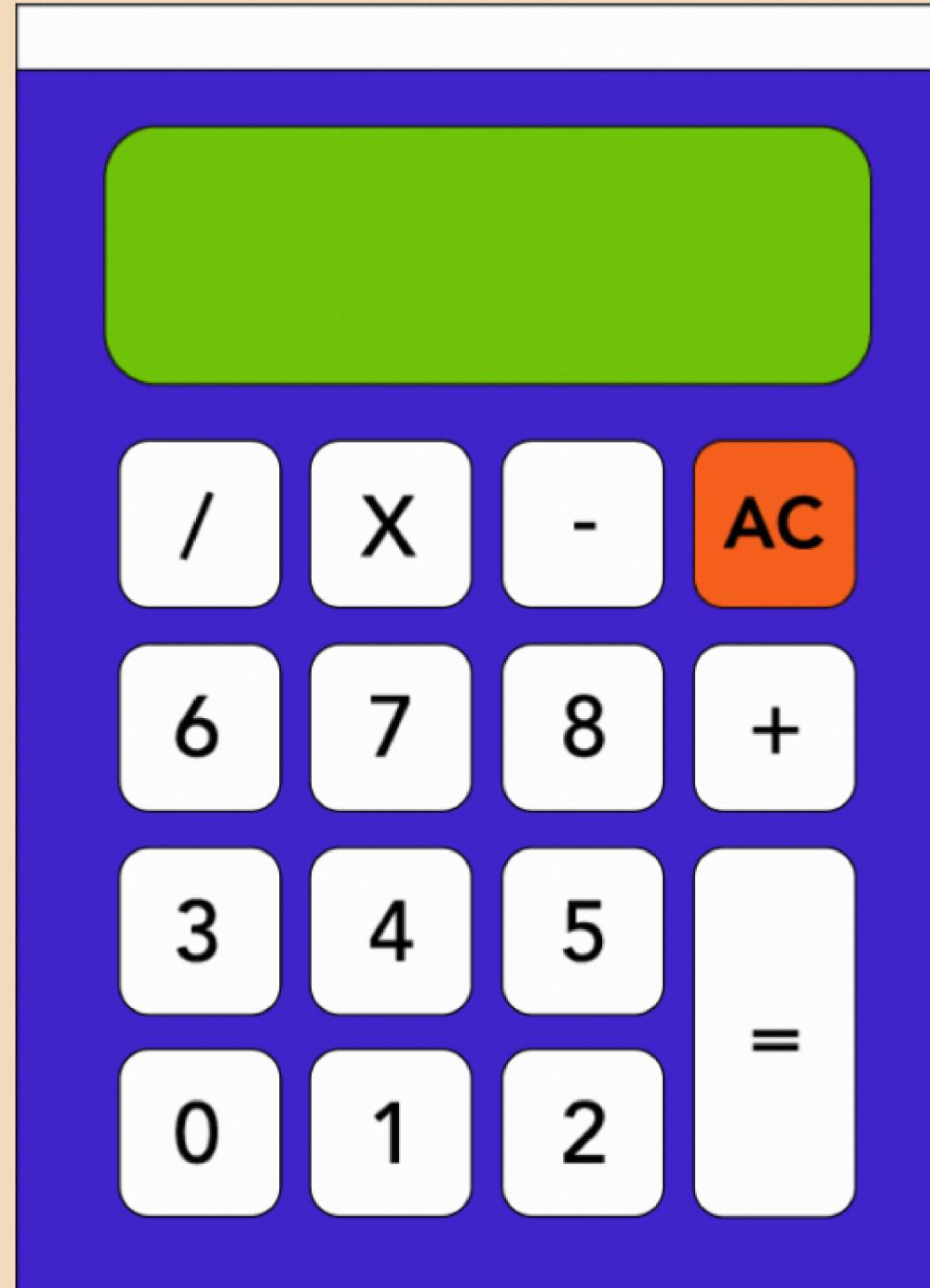


- 1 Introduction
- 2 Arithmetic Operators
- 3 Assignment Operators
- 4 Logical Operators
- 5 Comparison Operators

Introduction to JavaScript Operators



Arithmetic Operators



Arithmetic Operators

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

```
// Simple calculations
let result1 = 2 + 7; // result1 becomes 9
let difference = 15 - 4; // difference becomes 11

// Combined calculations
// area becomes 50 after
// assigning length = 5 and width = 10
let area = length * width;

// Using increment/decrement
let counter = 0;
counter++; // counter becomes 1
counter--; // counter becomes 0

// Modulus application
let remainder = 13 % 4; // remainder becomes 1
```

Arithmetic Operators

- Arithmetic operators work with numerical values.
- Division by zero results in an error.
- Increment/decrement operators are convenient for iterative updates.

Assignment Operators



Assignment Operators

Simple assignment (=)

Addition assignment (+=)

Subtraction assignment (-=)

Multiplication assignment (*=)

Division assignment (/=)

Modulus assignment (%=)

```
let score = 75;
```

```
score += 10; // score becomes 85
```

```
score -= 5; // score becomes 80
```

```
score *= 2; // score becomes 160
```

```
score /= 4; // score becomes 40
```

```
score %= 7; // score becomes 6
```

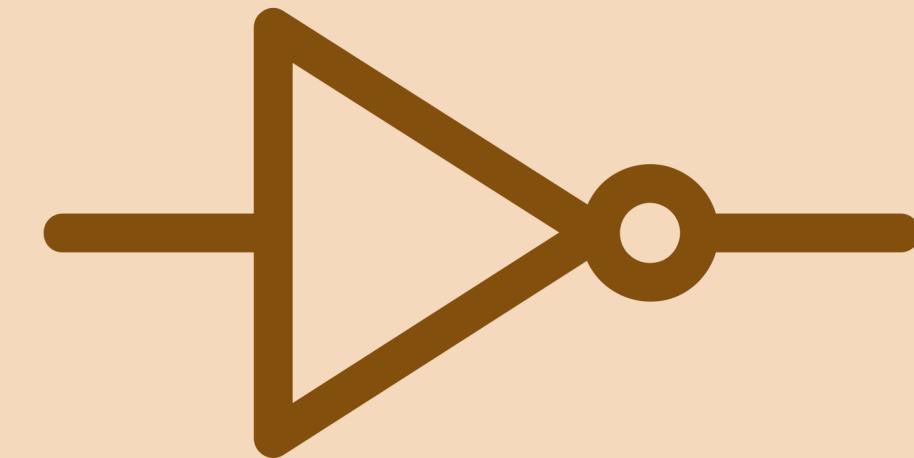
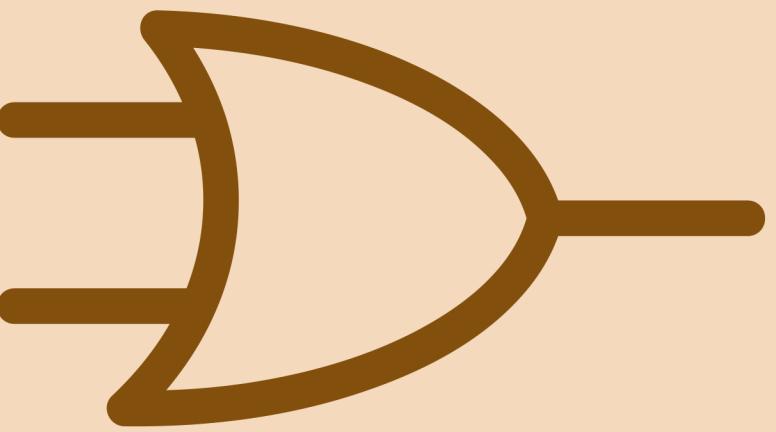
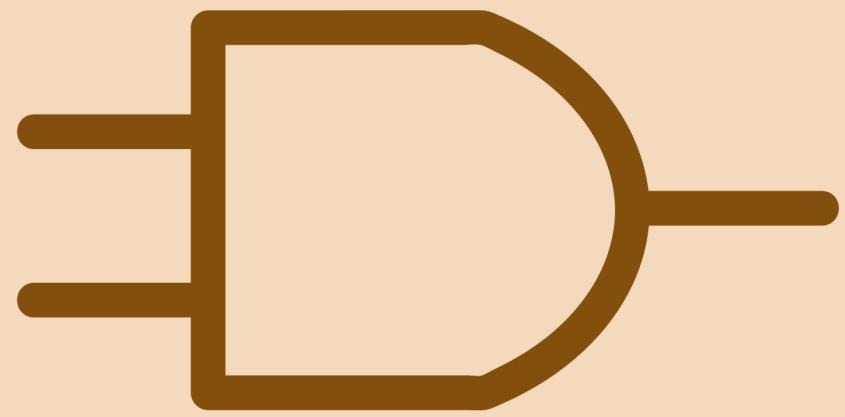
```
// Output: Final score: 6
```

```
console.log("Final score:", score);
```

Assignment Operators

- Assignment operators create or modify variable values.
- They offer succinct ways to combine assignments and calculations.
- Choose the appropriate operator based on the desired operation.

Logical Operators



Logical Operators

- AND (`&&`): Both conditions must be true for the overall expression to be true.
- OR (`||`): At least one condition must be true for the overall expression to be true.
- NOT (`!`): Reverses the truth value of a condition.

Truth Table for AND

Input A	Input B	Output
True	True	True
True	False	False
False	True	False
False	False	False

Truth Table for OR

Input A	Input B	Output
True	True	True
True	False	True
False	True	True
False	False	False

Logical Operators

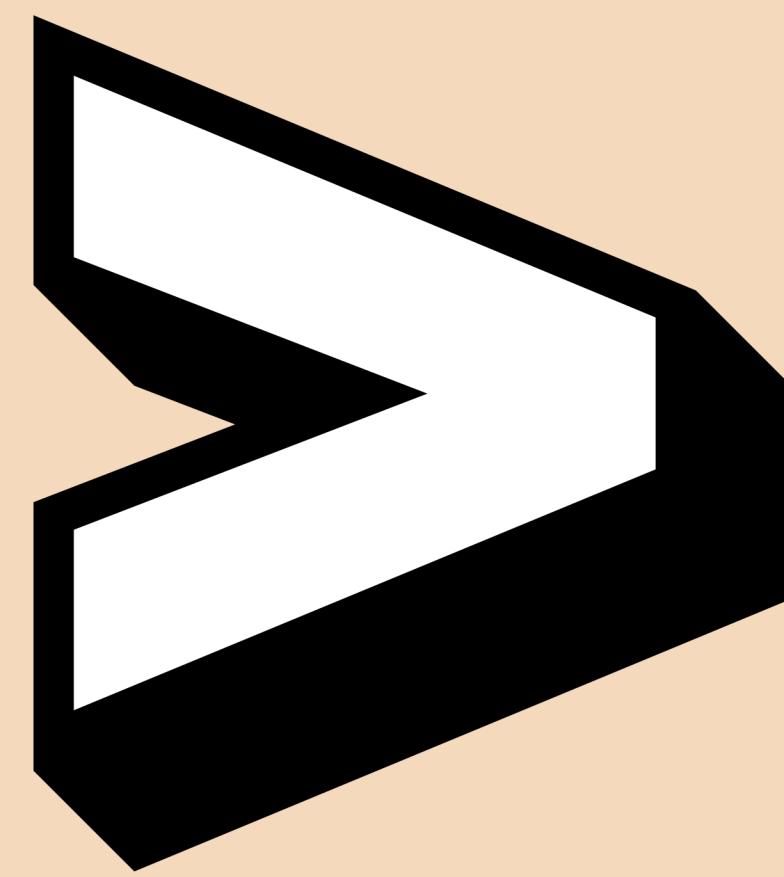
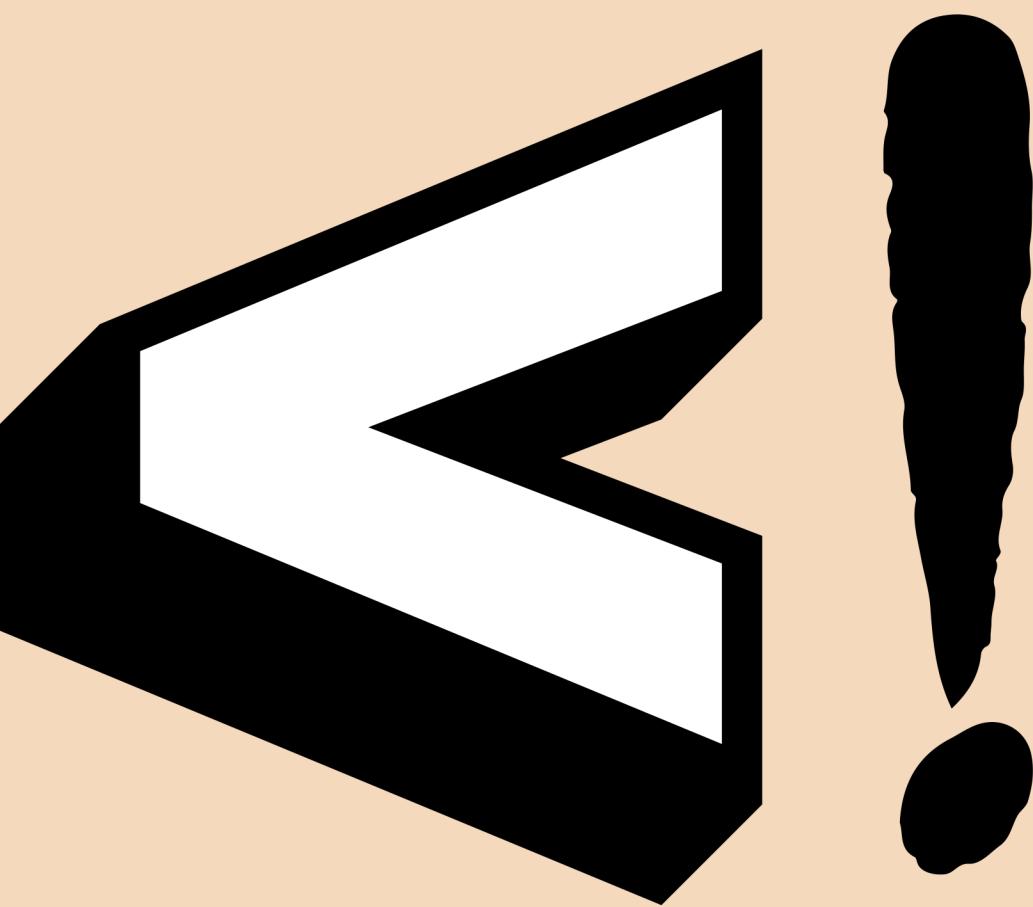
```
let isSunshine = true;
let isWeekend = false;

//false
console.log(isSunshine && isWeekend);

//true
console.log(isSunshine || isWeekend);

//false
console.log(!isSunshine || !isWeekend);
```

Comparison Operators



Comparison Operators

Equality Operators:

Equal to (==): Checks for value equality e.g., `5 == "5"` is true.

Strict equal to (===): Checks for both value and type equality (e.g., `5 === "5"` is false).

Comparison Operators

Inequality Operators:

Not equal to (!=): Checks for value inequality (e.g., `10 != 20` is true).

Strict not equal to (!==): Checks for both value and type inequality (e.g., `10 !== "10"` is true).

Comparison Operators

Relational Operators:

Greater than (>): Checks if the left value is greater than the right value (e.g., $7 > 3$ is true).

Less than (<): Checks if the left value is less than the right value (e.g., $2 < 5$ is true).

Greater than or equal to (\geq): Checks if the left value is greater than or equal to the right value (e.g., $8 \geq 8$ is true).

Less than or equal to (\leq): Checks if the left value is less than or equal to the right value (e.g., $4 \leq 6$ is true).



“

See you in Next Video

