

ITD523

Introduction to JavaScript fundamentals

Prepared by:

Assoc. Prof. Dr. Islam A.T.F. Taj-Eddin

Learning Objectives

- Understand different types of JavaScript operators
- Apply arithmetic, logical, and comparison operators
- Use operator precedence correctly
- Interact with users using dialog boxes
- Build simple interactive programs

What are Operators?

- Symbols used to perform operations on values (operands)
- Operands can be:
 - Variables
 - Values

```
let x = 5;  
let y = 2;  
console.log(x + y);
```

Types of Operators

- Assignment
- Arithmetic
- Logical
- Comparison
- String
- Conditional (Ternary)

Unary → one operand (typeof x)

Binary → two operands (x + y)

Ternary → three operands (condition ? a : b)

Assignment Operators

- Assign values to variables

```
let name = "Alice";
```

Right-to-left evaluation:

```
let year;
```

```
let newYear = year = 2051;
```

Arithmetic Operators

Operator	Meaning
+	Addition
-	Subtraction
*	Multiplication
/	Division
%	Remainder
**	Power

Operator Precedence

- Use parentheses to control order

- let x = 5, y = 2;

- console.log(x + y); // 7

- console.log(x ** y); // 25

- console.log(2 + 2 * 2); // 6

- console.log((2 + 2) * 2); // 8

Unary Operators

- +, - convert to number
 - let n = +"123"; // 123
 - let n = -"123"; // -123

Increment/Decrement

- ++ and -- (prefix & postfix)
- Prefix → returns new value
++x
- Postfix → returns old value
x++

Floating Point Issue

- $0.2 + 0.1 \neq 0.3$ exactly
 - `Console.log(0.2 + 0.1); // 0.30000000000000004`
 - JavaScript uses floating-point numbers

Compound Assignment

- $+=$, $-=$, $*=$, $/=$, $**=$

– $x += 5$; // $x = x + 5$

– $x *= 2$;

– $x **= 3$;

Logical Operators

- &&, ||, !

Operator	Meaning	
&&	AND	<code>console.log(true && false); // false</code>
	OR	<code>console.log(true false); // true</code>
!	NOT	<code>console.log(!true); // false</code>

Short-Circuit

- Stops evaluation early

```
let x = 0;
```

```
console.log(x && 5); // 0
```

Logical with Non-Boolean

- `console.log(0 || 5); // 5`
- `console.log("A" && "B"); // "B"`

Logical Assignment

`a &&= false;`

`b ||= true;`

String Operator

- Concatenation using +

```
let msg = "Hello " + "World";
```

Comparison Operators

- ==, ===, !=, !==, >, <

Operator	Meaning	
==	Equal	
===	Strict Equal	
!=	Not Equal	
!==	Strict Not Equal	<code>console.log(10 > 5); // true</code>
> < >= <=	Comparis on	<code>console.log("b" > "a"); // true</code>

Equality vs Strict

- `===` preferred over `==`

```
10 == "10" // true
```

```
10 === "10" // false
```

Other Operators

- `typeof`, `instanceof`, `delete`

`typeof` → returns type

`instanceof` → checks object type

`delete` → removes object property

Ternary Operator

- `condition ? a : b`

```
let result = condition ? "Yes" : "No";
```

```
let name = 1 > 2 ? "Alice" : "Bob";
```

User Interaction

- Programs interact with users

User inputs data

Program reacts accordingly

Examples:

Calculator

Forms

Client-side (Browser) → interactive

Server-side (Node.js) → minimal interaction

Dialog Boxes

- alert,
- confirm,
- Prompt

Input is always string
Convert if needed
Number(age)

Alert

- Displays message
- Stops execution

```
alert("Hello World");
```

Confirm

- OK → true
- Cancel → false

```
let result = confirm("Continue?");
```

```
let remove = confirm("Delete?");
```

```
let msg = remove ? "Deleted" : "Cancelled";
```

Prompt

- Returns string
- Cancel returns null

```
let name = prompt("Enter your name:");
```

```
let age = prompt("Enter age:");
```

```
alert("Age is " + age);
```

Summary

- Learned different operators
- Understood precedence
- Used dialog boxes for interaction