



# ETHEREUM 2.0 MASTERY PROGRAM

Instructor: Raja Rizwan Saleem



Module

ONE

# JAVASCRIPT CRASH COURSE

Instructor: Raja Rizwan Saleem



SESSION-3

# Arithmetic Operators

# Arithmetic Operators

- **Mathematical Calculations:** Enable performing basic mathematical operations such as addition, subtraction, multiplication, division, and modulus.
- **Data Manipulation:** Facilitate manipulating numerical data, which is essential for developing dynamic web applications.
- **Algorithm Implementation:** Essential for implementing algorithms that require mathematical operations, such as sorting and searching algorithms.

# Arithmetic Operators

- Financial Calculations: Important for applications involving financial calculations, such as interest rates, pricing, and discounts.
- Gaming Logic: Used in developing games to calculate scores, player positions, and other game mechanics.
- Animation: Enable creating smooth animations by calculating positions, velocities, and other properties over time.
- Form Validation: Useful in validating numerical input in forms, ensuring that data entered by users meets specific criteria

# Arithmetic Operators

- **Statistical Analysis:** Important for performing statistical operations, such as averages, standard deviations, and other statistical measures.
- **Data Analysis and Visualization:** Facilitate data analysis and visualization by performing calculations on datasets and rendering graphs or charts.
- **Conditional Logic:** Often used in conjunction with conditional statements to perform specific actions based on calculated values.
- **Performance Optimization:** Arithmetic operations can be optimized for performance, making them efficient for high-frequency calculations.

# Arithmetic Operators

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

# Addition Operator (+)

## Addition Example:

```
var a = 10;
```

```
var b = 5;
```

```
var sum = a + b;
```

```
console.log(sum); // Output: 15
```

# Subtraction Operator (-)

## Example:

```
var a = 10;
```

```
var b = 5;
```

```
var difference = a - b;
```

```
console.log(difference); // Output: 5
```

# Multiplication Operator (\*)

## Example:

```
var a = 10;
```

```
var b = 5;
```

```
var product = a * b;
```

```
console.log(product); // Output: 50
```

# Division Operator (/)

## Example:

```
var a = 10;
```

```
var b = 5;
```

```
var quotient = a / b;
```

```
console.log(quotient); // Output: 2
```

# Modulus Operator (%)

## Example:

```
var a = 10;
```

```
var b = 3;
```

```
var remainder = a % b;
```

```
console.log(remainder ); // Output: 1
```

# Working with Negative Numbers

## **Example:**

Example:

```
var a = -10;
```

```
var b = 5;
```

```
console.log(a + b); // Output: -5
```

```
console.log(a - b); // Output: -15
```

```
console.log(a * b); // Output: -50
```

```
console.log(a / b); // Output: -2
```

# Operator Precedence

## **Example:**

Example:

```
var a = 10;
```

```
var b = 5;
```

```
var c = 2;
```

```
var result = a + b * c; // Multiplication first, then addition
```

```
console.log(result); // Output: 20
```

# Using Parentheses

## Example:

```
var a = 10;
```

```
var b = 5;
```

```
var c = 2;
```

```
var result = (a + b) * c; // Parentheses change the order
```

```
console.log(result); // Output: 30
```

# Floating Point Arithmetic

## Example:

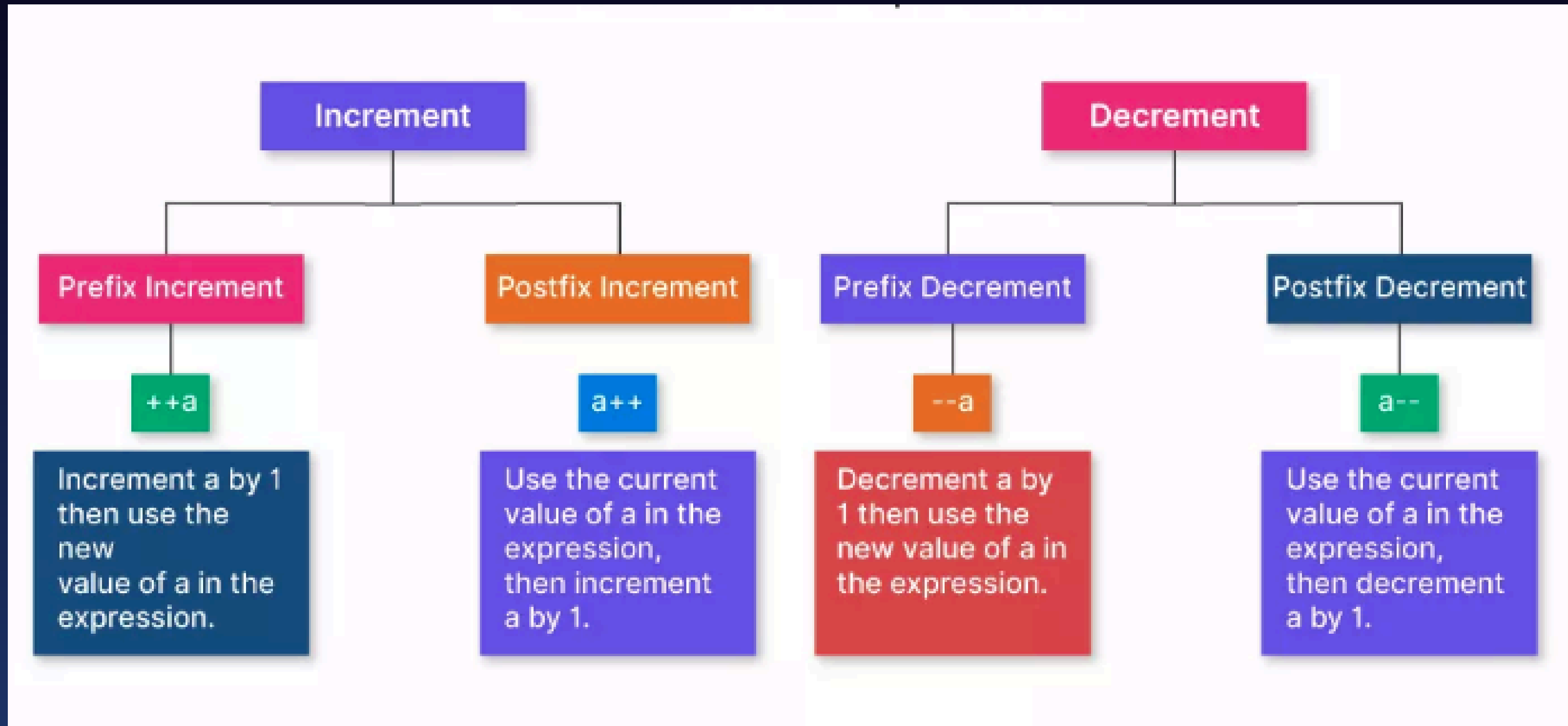
```
var a = 0.1;  
var b = 0.2;  
var sum = a + b;  
console.log(sum); // Output: 0.30000000000000004
```

# Summary of Arithmetic Operators

Operator	Description	Example
<b>** Let X = 10 and Y = 5**</b>		
<b>+</b>	<b>Addition</b>	<b>X+Y = 15</b>
<b>-</b>	<b>Subtraction</b>	<b>X-Y = 5</b>
<b>*</b>	<b>Multiplication</b>	<b>X*Y = 50</b>
<b>/</b>	<b>Division</b>	<b>X/Y = 2</b>
<b>%</b>	<b>Modulus</b>	<b>X%Y = 0</b>
<b>++</b>	<b>Increment</b>	<b>X++ = 11</b>
<b>--</b>	<b>Decrement</b>	<b>X-- = 4</b>

# Increment / Decrement Operators

# Increment and Decrement Operator



# Increment and Decrement Operator

1. While working on application you will frequently required to increase variable by 1 or decrease variable by 1
2. For this situation you can use increment and decrement operator
3. ++ increment operator
4. --decrement operator
5. These operators can be used as prefix and postfix

# Increment and Decrement Operator

1. We can increase or decrease value using existing addition and subtraction operators

```
var a = 12;  
a = a + 1;      // 13  
a = a - 1;      // 12  
var b = 12;  
b += 1;         // 13  
b -= 1;         // 13
```

# Increment and Decrement Operator

Prefix Increment and  
Decrement

```
var age = 12;
```

```
++age;
```

```
alert(age); //Result 13
```

```
--age;
```

```
alert(age); //Result 12
```

# Increment and Decrement Operator

## Postfix Increment and Decrement

```
var age = 12;
```

```
age++;
```

```
alert(age); //Result 13, same as prefix
```

```
age--;
```

```
alert(age); //Result 12, same as postfix
```

# Increment and Decrement Operator

- What is the difference between Prefix and Postfix
- You will NOT find any difference if you will not assign result of prefix and postfix to any other variable
- Prefix operator first increase/decrease the value in variable and then assign result to other variable
- Postfix operator first assign the value in other value then increase/decrease value in actual variable

# Increment and Decrement Operator

## Prefix Increment

```
var age = 12;  
var newAge = ++age;  
  
alert(age); //Result 13  
alert(newAge); //Result 13
```

# Increment and Decrement Operator (Practice)

## Postfix Increment

```
var age = 12;  
var newAge = age++;  
alert(age);           //Result 13  
alert(newAge);       //Result 12 , see the difference
```

# Increment and Decrement Operator

## Prefix Decrement

```
var age = 12;
```

```
var newAge = --age;
```

```
alert(age); //Result 11
```

```
alert(newAge); //Result 11
```

# Increment and Decrement Operator

## Postfix Decrement

```
var age = 12;  
var newAge = age--;  
alert(age);           //Result 11  
alert(newAge);       //Result 12 , see the difference
```

# Increment and Decrement Operator

## Example

```
var a = 4;
```

```
var b = 2;
```

```
var c = a++ - b; // first value of a placed here
```

which is 4 then increase 1 in a so will become 5

```
alert(a); // 5
```

```
alert(b); // 2
```

```
alert(c); // 2
```

# Increment and Decrement Operator

## Example 2

```
var a = 4;
```

```
var b = 2;
```

```
var c = ++a + b; // first value of a increased so  
will become 5 then value of a will be placed here, 5
```

```
alert(a); // 5
```

```
alert(b); // 2
```

```
alert(c); // 7
```

# Increment and Decrement Operator

## Example 3

```
var a = 4;  
var b = 3;  
var c = a++ + --b - --a;  
alert(a);           // 4  
alert(b);           // 2  
alert(c);           // 2
```

# Increment and Decrement Operator

## Example 4

```
var a = 4;  
var b = 3;  
var c = ++a + b++ - a + --b;  
alert(a);           // 5  
alert(b);           // 3  
alert(c);           // 6
```

# String Concatenation

## String Concatenate

"Hello" + "World" = "HelloWorld"

String 1    String 2    Result

# Concatenating Text strings

Strings cannot be divided, multiplied, or subtracted, but the + operator can be used on them. It does not add, but it concatenates—it glues two strings together. The following line will produce the string

```
"concatenate": "con" + "cat" + "e" + "nate"
```

# Concatenating Text strings

1. The + operator can also be used for concatenating strings
2. E.g:

```
var firstName = "Shahbaz";  
var lastName = "Hussain";  
  
//concatenate firstName, space character and lastName  
var fullName = firstName + " " + lastName;  
  
alert(fullName);
```

# Concatenating strings and numbers

1. Adding two numbers, will return the sum, but adding a number and a string will return a string:

```
var a = "6" + 2;           // "62"  
  
var b = 3 + "6";          // "36"  
  
var c = "Hello " + 2;     // "Hello 2"  
  
var d = 2 + "Hello ";    // "2 Hello"
```

# Concatenating strings with variables

```
let text1 = "sea";  
let text2 = "food";  
let result =  
    text1.concat(text2);  
console.log(result);
```

# Concatenating strings with variables

```
let text1 = "Hello";  
let text2 = "world!";  
let result = text1.concat(" ", text2);  
console.log(result);
```

# Prompt and Parsing String

# prompt or window.prompt()

1. The `prompt()/window.prompt()` method displays a dialog box that prompts the visitor for input.
2. In a prompt, you can specify a second string. This is the default response that appears in the field when the prompt displays.
3. The `prompt()` method returns the input value if the user clicks "OK". If the user clicks "cancel" the method returns null.

# prompt or window.prompt()

```
var question = "What is your name?";  
var defaultAnswer = "Rizwan";  
var name = prompt(question, defaultAnswer);  
console.log("Name = " + name);
```

# prompt or window.prompt()

1. If you ask for number in prompt and try apply addition operator on it then it will concatenate the value because prompt returns string

```
var question = "What is your age?";  
var age = prompt(question); // Assuming input 12  
var newAge = age + 5; // It will concatenate  
console.log("New Age = "+ newAge); //result 125
```

# prompt or window.prompt()

```
let age = prompt("What is your age?");  
console.log(age);
```

# The confirm() Method

The `confirm()` method displays a dialog box with a message, an OK button, and a Cancel button. The `confirm()` method returns `true` if the user clicked "OK", otherwise `false`. A confirm box is often used if you want the user to verify or accept something. A confirm box takes the focus away from the current window, and forces the user to read the message. Do not overuse this method. It prevents the user from accessing other parts of the page until the box is closed.

```
let deletePost = confirm("Do you really want to  
delete this post?");  
console.log(deletePost);
```

# Convert string to integer

1. Addition operator in string concatenate values, even if value in string is number

```
var value1 = "3";  
var value2 = "5";  
var value3 = value1+value2;  
console.log(value3); // result, 35
```

# Convert string to integer

1. prompt function also returns string even if you provide number in input box

```
//Assuming we will provide value 5 in input
```

```
var age = prompt("What is your age");
```

```
var num = 4;
```

```
var sum = age + num;
```

```
console.log(sum); // result, 54
```

# THANK-YOU

