JavaScript

A Brief **Introduction** From **Scratch**

What is JavaScript!

JavaScript is a **lightweight** programming language that web developers commonly use to create more **dynamic interactions** when developing **web pages**, **applications**, **servers**, and or even **games**.

01
JavaScript Basics

Learn the basic fundamentals

Chapter 01: JS Basics

01 console.log()

👉 You can write it to write anything to your **browser's console**. Literally the easiest way to get started.

02 Variables

👉 Variables are **placeholders** to store information and data in **your app**. if you have a **variable** like this: **name = 'Ismail'**
console.log(name) is the same as **console.log('Ismail')**

03 Comments

👉 Your browser will **skip those lines** and **won't run them**. Use it to **make notes** within your code for yourself or other developers.

Chapter 01: JS Basics

04 Operators

👉 Operators are used to **assign values**, **compare values**, perform **arithmetic operations**.

Basically, stuff you see in math class... (**+ , - , = , * , /** etc.)

06 Prompt()

👉 The **prompt()** is a **built-in JavaScript functionality** that helps you get inputs from a user through the browser.

06 alert()

The alert() is like prompt(), but it is only for output.

06 Number('10')

Number('10') converts the string 100 to number 100.
Number('10') 👉 10

Chapter 01: JS Basics

05 Your First JS App

👉 Create a **mini app** 📱 where you take the **price for the food** 🍔, and the **tip percentage** 💁‍♂️ the customer wants to give.

👉 Then calculate the **tipAmount** and print that in. 💡 **food * tipPercentage**

👉 Also calculate the **totalAmount**. 💡 **total = food + tipAmount**

02

Baby Weather ☁️ App

Some More Basics.....

Chapter 02:
Baby Weather App

01 Data Types

a) Numbers

Any kind of **Number**, including **decimals**.

```
1  // Numbers
2  console.log(5)
```

b) String

Zero or **more characters** written inside **quotes**.

```
1  // String
2  console.log('hello')
```

01 Data Types

a) Array 🤔 Will talk about it later

```
1 // Array
2 myArray = [
3     'hi how is it going?'
4 ]
```

b) Objects 🤔 Will talk about it later

```
1 // Array
2 myObject = {
3     greeting: 'hi how is it going?'
4 }
```

b) boolean 👉 true / false

```
1 // Boolean
2 myBoolean = true
```

Chapter 02: Baby Weather App

02 Math Operations

a) Multiply 👉 *

```
1 console.log(2 * 4)
```

b) Divide 👉 /

```
1 console.log(4 / 4)
```

c) Exponent 👉 **

```
1 console.log(2 ** 3)  // 2^3 = 2*2*2 = 8
```

d) Módulo/ Reminder 👉 %

```
1 console.log(5 % 3)
```

e) Add 👉 +

```
1 console.log(2 + 4)
```

f) Subtract 👉 -

```
1 console.log(10 - 4)
```

Chapter 02: Baby Weather App

03 Math Methods

a) Floor

👉 The **Math.floor()** method rounds a number **down** to the nearest integer, and returns the result.

b) Ceil

👉 The **Math.ceil()** method rounds a number **up** to the nearest integer, and returns the result.

Chapter 02: Baby Weather App

04 Variables 😊 (Types)

a) const

👉 The value cannot be changed.

😕 If you still changing the value. it will throw error.

Uncaught TypeError: Assignment to constant variable.

b) let

👉 The value can be changed.

c) var

👉 The value can be changed.

⚠️ **Not recommended**

Chapter 02: Baby Weather App

05 Conditional Operators 👉 (return a boolean)

👉 Conditionals are used in statements to **compare** variables' values and/or **data types**.

👉 They always return a **boolean** - ***true or false***

Chapter 02: Baby Weather App

```
var x = 5

// == (double equal) - checks for equality
x == 8              // false
x == 5              // true
x == '5'            // true

// === (triple equal sign) - checks for equality and data type
x === 5             // true
x === '5'           // false

// >, < - greater than, less than
x > 8               // false
x < 8               // true

// >=, <= - greater than or equal to, less than or equal to
5 >= x              // true
8 <= x              // false

// != - not equal to
x != 8              // true

// !== - not equal value or type
x !== '5'           // true - same value, different type
x !== 5             // false - same value, same type
```