JavaScript Crash Course

CPU AGENCY | July 2025

**Prepared by:** Ali Sayegh

# Table of Contents

[Table of Contents](#_bookmark0) [3](#_bookmark0)

1. [Introduction](#_bookmark1) [3](#_bookmark1)
2. [Setting Up Your Environment](#_bookmark2) [4](#_bookmark2)
3. [Basic Syntax & console.log](#_bookmark3) [4](#_bookmark3)
4. [Variables: var, let, const](#_bookmark4) [4](#_bookmark4)
5. [Data Types](#_bookmark5) [5](#_bookmark5)
6. [Operators](#_bookmark6) [5](#_bookmark6)
7. [Control Flow](#_bookmark7) [6](#_bookmark7)
8. [Loops](#_bookmark8) [6](#_bookmark8)
9. [Functions](#_bookmark9) [6](#_bookmark9)
10. [Objects](#_bookmark10) [7](#_bookmark10)
11. [Arrays & Methods](#_bookmark11) [7](#_bookmark11)
12. [ES6+ Features](#_bookmark12) [7](#_bookmark12)
13. [DOM Manipulation](#_bookmark13) [7](#_bookmark13)
14. [Events](#_bookmark14) [8](#_bookmark14)
15. [JSON](#_bookmark15) [8](#_bookmark15)
16. [Async JavaScript](#_bookmark16) [8](#_bookmark16)
17. [Mini Project: To-Do List](#_bookmark17) [8](#_bookmark17)
18. [Next Steps & Resources](#_bookmark18) [9](#_bookmark18)



**Table of Contents**

1. Introduction
2. Setting Up Your Environment
3. Basic Syntax & console.log
4. **Variables:** var, let, const
5. Data Types
6. Operators
7. Control Flow
8. Loops
9. Functions
10. Objects
11. Arrays & Methods
12. ES6+ Features
13. DOM Manipulation
14. Events
15. JSON
16. Async JavaScript
17. **Mini Project:** To-Do List
18. Next Steps & Resources



# Introduction

JavaScript is the programming language of the web. It makes websites interactive by:

* + - Responding to user actions (clicks, form submissions).
    - Dynamically updating content without page reloads.
    - Fetching data from servers (e.g., loading tweets).

## Key Concepts Covered:

* + - Syntax fundamentals.
    - Working with data (variables, objects, arrays).
    - Controlling program flow (loops, conditionals).
    - DOM manipulation (changing webpage content).
    - Async operations (fetching data).



# Setting Up Your Environment

## Tools You Need:

1. **Browser:** Chrome/Firefox (for running JS).
2. **Code Editor:** VS Code (free, with extensions like Live Server).
3. **Optional:** Node.js (to run JS outside the browser).

## Steps:

1. **Install VS Code:** [https://code.visualstudio.com](https://code.visualstudio.com/).
2. Create an index.html file and link a script.js file:



# Basic Syntax & console.log

## First Program:

console.log("Hello, CPU AGENCY!"); // Prints to browser console

* + - **console.log():** Used for debugging/output.
    - **Semicolons (;):** Optional but recommended.



# Variables: var, let, const

**VARIABLE TYPES**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Keyword** | **Scope** | **Reassignable** | **Hoisting** | **Example** |
| var | Function | Yes | Yes | var x = 10; |
| let | Block | Yes | No | let name = "Ali"; |
| const | Block | No | No | const PI = 3.14; |

**Best Practice:** Use const by default, let if reassigning, avoid var.



# Data Types

JavaScript has dynamic types (no need to declare types).

|  |  |  |
| --- | --- | --- |
| **Type** | **Example** | **Description** |
| String | "Hello" | Text (use '' or ""). |
| Number | 42, 3.14 | Integers/decimals. |
| Boolean | true, false | Logical values. |
| Null | null | Intentional empty value. |
| Undefined | undefined | Variable not assigned. |

## Check Type:

console.log(typeof "Hello"); // "string"



# Operators

## Arithmetic:

let sum = 5 + 3; // 8

## Comparison:

5 == "5" // true (loose equality)

5 === "5" // false (strict equality)

## Logical:

if (age > 18 && hasLicense) { ... }



# Control Flow

## if/else:

if (score >= 90) { grade = "A";

} else if (score >= 80) { grade = "B";

} else { grade = "C";

}



# Loops

## for Loop:

for (let i = 0; i < 5; i++) { console.log(i); // 0, 1, 2, 3, 4

}

## while Loop:

let i = 0; while (i < 5) {

console.log(i); i++;

}



# Functions

## Declaration:

function greet(name) { return `Hello, ${name}`;

}

## Arrow Function (ES6+):

const greet = name => `Hello, ${name}`;



# Objects

const person = { name: "Ali", age: 25, greet() {

console.log(`Hi, I'm ${this.name}`);

}

};

person.greet(); // "Hi, I'm Ali"



# Arrays & Methods

const fruits = ["apple", "banana"]; fruits.push("cherry"); // Adds to end

fruits.forEach(fruit => console.log(fruit)); // Logs each item



# ES6+ Features

## Destructuring:

const { name, age } = person; // Extract properties

## Spread Operator:

const newArr = [...fruits, "orange"];



# DOM Manipulation

document.getElementById("title").textContent = "New Heading";



# Events

button.addEventListener("click", () => { alert("Button clicked!");

});



# JSON

const json = JSON.stringify(person); // Object → String const obj = JSON.parse(json); // String → Object



# ASync Javascript

## Promises:

fetch("url")

.then(response => response.json())

.then(data => console.log(data));

## async/await:

async function fetchData() {

const response = await fetch("url"); const data = await response.json();

}



# Mini Project: To-Do List

## HTML:

<input type="text" id="task">

<button id="add">Add Task</button>

<ul id="list"></ul>

## JavaScript:

const input = document.getElementById("task"); const btn = document.getElementById("add"); const list = document.getElementById("list");

btn.addEventListener("click", () => { const task = input.value.trim();

if (!task) return; // Ignore empty input

const li = document.createElement("li"); li.textContent = task;

li.addEventListener("click", () => li.remove()); // Click to delete

list.appendChild(li); input.value = ""; // Clear input

});



# Next Steps & Resources

* + - **MDN Web Docs:** [https://developer.mozilla.org](https://developer.mozilla.org/)
    - **JavaScript.info:** [https://javascript.info](https://javascript.info/)
    - **Practice:** Build small projects (calculator, weather app).



CPU AGENCY | Ali Sayegh | JavaScript Crash Course | July 2025