

JavaScript Basics: Comprehensive Guide



JavaScript Basics: Comprehensive Guide	1
What is JavaScript?	2
1. Variables and Data Types	2
Variables	2
Data Types	3
2. Operators	3

Learn more HTML, CSS, JavaScript Web Development at <https://basescripts.com/> Laurence Svekis

Examples:	3
3. Functions	3
Syntax:	3
4. Control Flow	3
Conditional Statements:	3
Loops:	4
5. Arrays and Objects	4
Arrays:	4
Objects:	4
Examples	4
Example 1: Calculate the Sum of Two Numbers	4
Example 2: Check Even or Odd	4
Exercises	5
Exercise 1: Greet a User	5
Solution:	5
Exercise 2: Find the Largest Number	5
Solution:	5
Multiple-Choice Questions	5
Question 1:	5
Question 2:	5
Question 3:	6
Advanced Example: Create a To-Do List	6

JavaScript is a versatile programming language essential for creating dynamic and interactive web applications. This guide introduces JavaScript fundamentals, including variables, data types, operators, functions, and control flow, with examples, exercises, and quiz questions.

What is JavaScript?

JavaScript is a programming language that runs in the browser and allows developers to add interactivity, handle events, and manipulate web page content.

1. Variables and Data Types

Variables

Variables store data that can be used and modified in your program.

```
let name = "John"; // String
const age = 25;    // Number
```

Learn more HTML, CSS, JavaScript Web Development at <https://basescripts.com/> Laurence Svekis

```
var isStudent = true; // Boolean
```

Data Types

1. **Primitive Types:** String, Number, Boolean, Null, Undefined, Symbol.
2. **Complex Types:** Objects, Arrays, Functions.

2. Operators

Operators are used to perform operations on variables and values.

Examples:

- **Arithmetic Operators:** +, -, *, /, %
- **Comparison Operators:** ==, ===, !=, <, >, <=, >=
- **Logical Operators:** &&, ||, !

```
let a = 10;
let b = 5;
console.log(a + b); // 15
console.log(a > b && b > 0); // true
```

3. Functions

Functions are reusable blocks of code.

Syntax:

```
function greet(name) {
  return `Hello, ${name}!`;
}
console.log(greet("Alice")); // Hello, Alice!
```

4. Control Flow

Control flow structures manage the execution of code blocks.

Conditional Statements:

```
let age = 18;
if (age >= 18) {
  console.log("You are an adult.");
} else {
```

Learn more HTML, CSS, JavaScript Web Development at <https://basescripts.com/> Laurence Svekis

```
    console.log("You are a minor.");
}
```

Loops:

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

5. Arrays and Objects

Arrays:

Arrays store lists of values.

```
let fruits = ["Apple", "Banana", "Cherry"];
console.log(fruits[1]); // Banana
```

Objects:

Objects store key-value pairs.

```
let person = {
    name: "John",
    age: 30
};
console.log(person.name); // John
```

Examples

Example 1: Calculate the Sum of Two Numbers

```
function calculateSum(a, b) {
    return a + b;
}
console.log(calculateSum(5, 7)); // 12
```

Example 2: Check Even or Odd

```
function isEven(num) {
    if (num % 2 === 0) {
        return "Even";
    } else {
        return "Odd";
    }
}
```

Learn more HTML, CSS, JavaScript Web Development at <https://basescripts.com/> Laurence Svekis

```
    }  
  }  
  console.log(isEven(10)); // Even  
  console.log(isEven(7)); // Odd
```

Exercises

Exercise 1: Greet a User

Write a function that accepts a name as input and prints a greeting message.

Solution:

```
function greet(name) {  
  console.log(`Hello, ${name}!`);  
}  
greet("Alice");
```

Exercise 2: Find the Largest Number

Write a function that takes three numbers as arguments and returns the largest.

Solution:

```
function findLargest(a, b, c) {  
  return Math.max(a, b, c);  
}  
console.log(findLargest(5, 10, 8)); // 10
```

Multiple-Choice Questions

Question 1:

What keyword is used to declare a variable that can be reassigned?

1. var
2. let
3. const
4. Both 1 and 2

Answer: 4. Both 1 and 2

Explanation: Variables declared with var or let can be reassigned, but const cannot.

Question 2:

Learn more HTML, CSS, JavaScript Web Development at <https://basescripts.com/> Laurence Svekis

Which operator checks both value and type?

1. ==
2. !=
3. ===
4. <=

Answer: 3. ===

Explanation: The === operator checks for strict equality, considering both value and type.

Question 3:

What is the correct syntax for a function declaration?

1. let myFunc = function {}
2. function myFunc() {}
3. func myFunc() {}
4. define myFunc() {}

Answer: 2. function myFunc() {}

Explanation: This is the correct syntax for declaring a function.

Advanced Example: Create a To-Do List

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>To-Do List</title>
</head>
<body>
  <h1>To-Do List</h1>
  <input type="text" id="taskInput" placeholder="Add a task" />
  <button onclick="addTask()">Add Task</button>
  <ul id="taskList"></ul>
  <script>
    function addTask() {
      const taskInput = document.getElementById("taskInput");
      const taskList = document.getElementById("taskList");
      const task = taskInput.value;
      if (task) {
```

Learn more HTML, CSS, JavaScript Web Development at <https://basescripts.com/> Laurence Svekis

```
    const listItem = document.createElement("li");
    listItem.textContent = task;
    taskList.appendChild(listItem);
    taskInput.value = "";
  }
}
</script>
</body>
</html>
```