**🧠 JavaScript Fundamentals**

**Variables**

\* Declared with:

\* `let` — block-scoped, can be reassigned.

\* `const` — block-scoped, cannot be reassigned (constant).

\* `var` — function-scoped (older, avoid in modern code).

Example:

```js

let username = "Alice";

const pi = 3.14159;

```

**Data Types**

\* Common primitive types: `string`, `number`, `boolean`, `null`, `undefined`

\* Complex types: `object`, `array`, `function`

**Functions**

Functions are reusable blocks of code performing tasks or returning values.

Example:

```js

function greet(name) {

return `Hello, ${name}!`;

}

**🔷 1. Variables in JavaScript**

JavaScript has 3 ways to declare variables:

var name = "John"; // old (function-scoped)

let age = 25; // block-scoped (recommended)

const pi = 3.14; // block-scoped and constant

| **Keyword** | **Scope** | **Reassignable** | **Redeclarable** |
| --- | --- | --- | --- |
| var | Function | ✅ Yes | ✅ Yes |
| let | Block | ✅ Yes | ❌ No |
| const | Block | ❌ No | ❌ No |

**🔶 2. Data Types in JavaScript**

🔸 A. Primitive Types

let name = "Alice"; // String

let age = 21; // Number

let isStudent = true; // Boolean

let x = null; // Null (intentional empty)

let y = undefined; // Undefined (no value assigned)

let id = Symbol("id"); // Unique symbol

let bigInt = 12345678901234567890n; // BigInt

🔸 B. **Non-Primitive Types**

let person = { name: "Bob", age: 22 }; // Object

let fruits = ["apple", "banana"]; // Array (object)

let greet = function() { console.log("Hi"); }; // Function

**🔷 3. Functions in JavaScript**

**🔸 A. Function Declaration**

function greet(name) {

return "Hello, " + name;

}

console.log(greet("Janvi")); // Hello, Janvi

🔸 B. **Function Expression**

const add = function(a, b) {

return a + b;

};

console.log(add(2, 3)); // 5

🔸 C. **Arrow Function (ES6+)**

const multiply = (a, b) => a \* b;

console.log(multiply(3, 4)); // 12

🔸 D. **Default Parameters**

function greet(name = "Guest") {

return `Hello, ${name}`;

}

console.log(greet()); // Hello, Guest