🧩 Task 7: ES6+ Modern JavaScript Features

🎯 Goals:

- Learn modern syntax and patterns used in real-world JavaScript and React code
- Replace older syntax with cleaner, more concise code

🔍 Key Concepts with Descriptions & Examples

🔸 1. let and const

✅ Description:

- let: block-scoped, can be reassigned.
- const: block-scoped, **cannot** be reassigned.

📘 Example:

```
let count = 0;

count = 1; // ✅


const name = "Ali";

// name = "Ahmed"; ❌ Error: Assignment to constant variable
```

✅ **Tip:** Always use const by default, and let only if reassignment is needed.

🔸 2. Arrow Functions

✅ Description:

Shorter syntax for writing functions. They also behave differently with this (important in React).

📘 Example:

```
// Traditional
```

```
function add(a, b) {
 return a + b;
}

// Arrow Function
const add = (a, b) => a + b;
```

🔥 **One-liner syntax:**

If you have just one line that returns a value, omit {} and return.

🔸 **3. Template Literals**

✅ **Description:**

Allow embedded expressions inside strings using backticks (`)

📘 **Example:**

```
const user = "Ali";
const message = `Welcome back, ${user}!`; // ✅
```

🔸 **4. Destructuring**

✅ **Description:**

Extract values from objects or arrays into separate variables.

📘 **Object Destructuring:**

```
const user = { name: "Ali", age: 25 };
const { name, age } = user;
console.log(name); // "Ali"
```

📘 **Array Destructuring:**

```
const services = ["Car Wash", "Plumbing"];
```

```
const [first, second] = services;

console.log(first); // "Car Wash"
```

🔸 5. Spread Operator (...)

✅ **Description:**

Copies values from one array or object into another.

```
const nums = [1, 2, 3];

const moreNums = [...nums, 4, 5]; // [1, 2, 3, 4, 5]


const user = { name: "Ali" };

const updatedUser = { ...user, age: 25 }; // { name: "Ali", age: 25 }
```

🔸 6. Rest Parameters

✅ **Description:**

Allows functions to accept **variable number of arguments**.

```
function sum(...nums) {
  return nums.reduce((total, n) => total + n, 0);
}


sum(1, 2, 3); // 6
```

💉 Practice Exercises

✅ **Exercise 1: Template + Arrow Function**

Write an arrow function greet(name) that returns:

Hello, [name]! Welcome back.

Using template literals.

✅ Exercise 2: Destructuring

Given:

const customer = { name: "Ahmed", service: "Plumbing" };

Use destructuring to print:

"Ahmed booked Plumbing"

✅ Exercise 3: Spread Operator

- Create an array of 3 services
- Add 2 more using spread syntax
- Then add a new key status: "active" to a service object using spread

---

✅ Exercise 4: Rest Parameter

Write a function multiplyAll(...numbers) that returns the product of all numbers passed.

💬 Reflection Questions

1. What's the difference between let and var?
2. When would you choose destructuring?
3. What are the two main use cases of the spread operator?