Default parameters

Let's dive into **Default Parameters** in JavaScript — a handy feature introduced in **ES6 (ES2015)** that makes function definitions cleaner and more flexible.

🔷 What Are Default Parameters?

Default parameters allow you to assign a **default value** to a function parameter **if no argument is provided** (or if it's explicitly `undefined`).

✅ Basic Syntax

```
function functionName(param = defaultValue) {
  // function body
}
```

🧪 Example: Without Default Parameters

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

greet(); // Hello, undefined
```

Here, `name` is `undefined` because no value was passed.

✅ Example: With Default Parameter

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

```
greet();        // Hello, Guest
greet("Abhi");  // Hello, Abhi
```

🔄 Multiple Default Parameters

```
function createUser(name = "Anonymous", age = 18) {
  console.log(`Name: ${name}, Age: ${age}`);
}

createUser();             // Name: Anonymous, Age: 18
createUser("Abhi");       // Name: Abhi, Age: 18
createUser("Abhi", 25);   // Name: Abhi, Age: 25
```

🧠 How JavaScript Handles Default Values

Default values are only used when:

- No argument is passed
- The passed argument is `undefined`

❗ But NOT if the argument is `null`:

```
function sayHello(name = "Guest") {
  console.log(`Hello, ${name}`);
}

sayHello(undefined);  // Hello, Guest ✅
sayHello(null);       // Hello, null ❌
```

🧪 Default Parameter as Expression

You can even set default values using **expressions**:

```
function calculatePrice(price = 100, tax = price * 0.1) {
  return price + tax;
}

console.log(calculatePrice());      // 110
console.log(calculatePrice(200));   // 220
```

> Default values can depend on earlier parameters!

⚠️ Order Matters

If a parameter with a default value is followed by one **without a default**, and you skip the first argument, you must explicitly pass `undefined`:

```
function showInfo(name = "Guest", age) {
  console.log(name, age);
}

showInfo();            // Guest undefined
showInfo(undefined, 25) // Guest 25
```

✅ Benefits of Using Default Parameters

- Makes your functions **more robust**
- Avoids manual checks like:

  ```
  name = name || "Guest"; // older JS trick
  ```

- Cleaner and easier to read

🧾 Summary

Feature	Description

Syntax	`function greet(name = "Guest")`
Triggers if argument is	`undefined`
Not triggered if argument is	`null`
Can use expressions	Yes ✅
Order matters	Yes, defaulted parameters should come last

Would you like to go over **Rest Parameters (`...args`)** next?