✅ What is **Anonymous Function Expression**?

const sayHi = function() {

console.log("Hi!");

};

This is a **Function Expression** — more specifically, an **Anonymous Function Expression** because the function has **no name**.

**🔍 Parts Explained**

|  |  |
| --- | --- |
| Part | Meaning |
| const sayHi | Declares a constant variable named sayHi |
| = function() { ... } | Assigns an **anonymous function** (no name) to sayHi |
| console.log("Hi!") | The code that runs when the function is called |
| sayHi(); | Calls the function and prints "Hi!" |

**🆚 Compared to Function Declaration**

✅ Function Declaration:

function sayHi() {

console.log("Hi!");

}

sayHi();

**✅ Function Expression (your example):**

const sayHi = function() {

console.log("Hi!");

};

sayHi();

**🔑 Key Differences:**

|  |  |  |
| --- | --- | --- |
| Feature | Function Declaration | Function Expression |
| Hoisting | ✅ Yes (can call before defined) | ❌ No (must define first) |
| Has function name? | ✅ Yes | ❌ Optional (anonymous) |
| Assigned to variable? | ❌ No | ✅ Yes |
| Useful for | Normal functions | Callbacks, inline functions, closures |

**✅ Example Use Case of Function Expression:**

**Used often in things like:**

setTimeout(function()

{

console.log("Waited 1 second");

}, 1000);

Here, the function is **anonymous** and **used only once**.

## ✅ What is setTimeout()?

**setTimeout()** is a **built-in JavaScript function** used to **delay code execution**.

It runs a function **after a specified time**, measured in **milliseconds** (1 second = 1000 ms).

### 🔍 Your Code Explained

**setTimeout(function() {**

**console.log("Waited 1 second");**

**}, 1000);**

|  |  |
| --- | --- |
| Part | Meaning |
| setTimeout(...) | Schedules code to run after some time |
| function() { console.log("Waited 1 second"); } | Anonymous function to run later |
| 1000 | Delay time in milliseconds (1000 ms = 1 second) |

### 🕐 How It Works:

1. JavaScript sees setTimeout(...)
2. Waits **1000 ms (1 second)**
3. Then **runs the function** → **console.log("Waited 1 second");**
4. Output:

Waited 1 second

### ✅ Simple Example with Named Function

function greetLater() {

console.log("Hello after 2 seconds");

}

setTimeout(greetLater, 2000); // Runs after 2 seconds

### 💡 When is setTimeout() Useful?

* Showing a splash screen
* Delaying animations
* Auto-hiding messages
* Simulating network delay

Would you like to see a live example or how to use setInterval() (which repeats)?

## ✅ 1. setTimeout() – Runs Once After Delay

<!DOCTYPE html>

<html>

<head>

<title>setTimeout Demo</title>

</head>

<body>

<h2>Wait for it...</h2>

<p id="demo"></p>

<script>

setTimeout(function () {

document.getElementById("demo").innerText = "⏰ Hello after 3 seconds!";

}, 3000); // 3000 ms = 3 seconds

</script>

</body>

</html>

🔸 What it does:

* Page loads
* After 3 seconds, the text appears in the <p> tag

## 🔁 2. setInterval() – Repeats Code Every X Seconds

<!DOCTYPE html>

<html>

<head>

<title>setInterval Demo</title>

</head>

<body>

<h2>Live Clock (updates every second)</h2>

<p id="clock"></p>

<script>

setInterval(function () {

let now = new Date();

document.getElementById("clock").innerText = now.toLocaleTimeString();

}, 1000); // Updates every 1 second

</script>

</body>

</html>

🔸 What it does:

* Shows a **live clock** by updating the time every 1 second