

setTimeout() FUNCTION

The `setTimeout()` function is a built-in function in JavaScript that allows you to schedule the execution of a function after a specified delay in milliseconds. It is commonly used to create a delay before running a particular piece of code or a function.

Here is the basic syntax of `setTimeout()`:

Example:

```
setTimeout(function, delay, param1, param2, ...);
```

function: The function to be executed after the delay.

delay: The time to wait before executing the function, in milliseconds.

param1, param2, ...: Optional parameters to pass to the function when it is executed.

Here's a simple example:

javascript

Copy code

```
function sayHello() {  
    console.log("Hello, world!");  
}
```

```
// Call the sayHello function after a delay of 2000 milliseconds (2 seconds)
setTimeout(sayHello, 2000);
```

In the above example, the sayHello function will be executed after a delay of 2000 milliseconds.

You can also use an anonymous function directly within setTimeout:

Example:

```
// Using an anonymous function
setTimeout(function() {
    console.log("Delayed message");
}, 3000);
```

If you need to pass parameters to the function being executed, you can include them after the delay:

javascript

Copy code

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

```
// Call the greet function with parameters after a delay of 1000
milliseconds (1 second)
```

```
setTimeout(greet, 1000, "John", "Hi");
```

In this example, the greet function is called with the parameters "John" and "Hi" after a delay of 1000 milliseconds.

Keep in mind that `setTimeout()` is a one-time timer. If you need a function to be repeatedly executed at intervals, you might want to use `setInterval()` instead.