



@CODE.CLASH

# JavaScript setTimeout()



NEXT →

## setTimeout()

The `setTimeout()` method **executes** a **block of code after** the specified **time**.

- The method **executes** the code **only once**.
- Commonly used syntax :

```
setTimeout(function, milliseconds);
```

Its parameters are:

- **function** – a function containing a block of code
- **milliseconds** – the time after which the function is executed

The **setTimeout()** method returns an **intervalID**, which is a positive integer.

## Ex 1: Display a Text Once After 3 Second

```
// program to display a text using setTimeout method
function greet() {
    console.log('Hello world');
}

setTimeout(greet, 3000);
console.log('This message is shown first');

/* Output
This message is shown first
Hello world
*/
```

In the above program, the `setTimeout()` method calls the `greet()` function after 3000 milliseconds.



The `setTimeout()` method returns the **interval id**.

For example,

```
function greet() {  
  console.log('Hello world');  
}  
  
let intervalId = setTimeout(greet, 3000);  
console.log('Id: ' + intervalId);  
  
/* Output  
   Id: 3  
   Hello world  
  */
```

## clearTimeout()

If you **want to stop** this function call, you can use the **clearTimeout()** method.

```
let count = 0;

// function creation
function increaseCount() {
  // increasing the count by 1
  count += 1;
  console.log(count);
}

let id = setTimeout(increaseCount, 3000);

// clearTimeout
clearTimeout(id);
console.log('setTimeout is stopped.');

// setTimeout is stopped.
```

You can also **pass additional** arguments to the **setTimeout()** method.

```
setTimeout(function, milliseconds, parameter1, ....parameterN);
```

When you **pass** additional parameters, these parameters will be passed to the **specified function**.

```
// program to display a name
function greet(name, lastName) {
  console.log('Hello' + ' ' + name + ' ' + lastName);
}

// passing argument to setTimeout
setTimeout(greet, 1000, 'Imtiyaz', 'Nandasaniya');

// Imtiyaz Nandasaniya
```