

Higher Order Function is a function that takes another function as an argument or they return another function.

We will see an initial example of something called set interval. Set interval takes a function and it will call that function at a specific interval that we provide. It is a way to have something repeat every second or at any other given interval time. Right now, with a loop we can repeat things but that happens immediately, with set interval things repeat at an interval of a specific time.

We will look at how set interval works, but before we dive into that we are going to build the function that we are going to pass in set interval.

We are going to set the function that sings Twinkle Twinkle and we are going to schedule it to run at set interval of every second. First, we need to write the Twinkle Twinkle function code.

```
function sing() {  
    console.log("twinkle twinkle....");  
    console.log("how I wonder....");  
}
```

Now after we press *Enter*, the function will get registered in JavaScript. Now if we call the sing function, it will produce the following output.

```
sing()
```

Output:

```
twinkle twinkle....
```

```
how I wonder....
```

```
sing()
```

Output:

```
twinkle twinkle....
```

how I wonder....

sing()

Output:

twinkle twinkle....

how I wonder....

Next, we want to write set interval to call the *sing* function every 1000ms or every second. The syntax of the set interval goes as below

setInterval(AnotherFunction, Interval);

Here the name of set interval function is named as *setInterval* and it takes two arguments. The first one is a function named as *AnotherFunction* and the second argument is *Interval* which takes the number of milliseconds after which each interval will occur and the function, that is passed as the first argument, will be called.

Now, lets build our actual set interval function.

setInterval(sing, 1000);

Output:

number

twinkle twinkle....

how I wonder....

After 1000ms,

twinkle twinkle....

how I wonder....

After 1000ms,

twinkle twinkle....

how I wonder....

After 1000ms,

twinkle twinkle....

how I wonder....

Another thing that we need to notice is that when we called the *sing* function, we did not use parentheses after the function name, that is because we are not calling the *sing* function, the *setInterval* function is calling the *sing* function. So, we do not want to execute it, we want set interval to execute it. We are just passing the name *sing*, and the code inside of it but we are not running it, we are leaving that to set interval.

Another thing is that when we pressed *Enter* and before you got the output, we received a *number*. This number can be any number given to us by the set interval function and we need to use it to stop the execution of the set interval function. The function we use to stop the execution of set interval is called *clearInterval()* and we pass that number as an argument to the *clearInterval* function.

clearInterval(number);

➔ Function execution stops