

# JavaScript **Function()** In Detail

Swipe >



Mallikarjun | Web Developer

[f](#) [t](#) [i](#) [y](#) @CodeBustler



# JS Functions

A JavaScript function is a block of code designed to perform a **particular task**.

A JavaScript function is executed when "something" invokes it (*calls it*).

Syntax

```
function name(parameter1, parameter2, parameter3) {  
  // code to be executed  
}
```

A JS function is defined with the **function keyword**, followed by a name, followed by parentheses ().

Function **names can** contain letters, digits, underscores, and dollar signs (same rules as variables)

The parentheses may include **parameter names** separated by commas: (parameter1, parameter2, ...)





# Function Invocation

The code inside the function will execute when "something" invokes (calls) the function:

- When an event occurs (when a user clicks a button)
- When it is invoked (called) from JavaScript code
- Automatically (self invoked)

# Function Return

When JavaScript reaches a return statement, the function will stop executing.

Functions often compute a return value. The return value is **"returned" back to the "caller"**





# Why Functions?

To define the **code once**, and **use it many times**.

You can use the same code many times with different arguments, to produce different results.

```
function myFunction() {  
  console.log("Hello, This message is from function");  
  console.log("Welcome to codebustler");  
  console.log("");  
}
```

```
myFunction();  
myFunction();  
myFunction();
```

Example :

> Console × +

Hello, This message is from function  
Welcome to codebustler

Hello, This message is from function  
Welcome to codebustler

Hello, This message is from function  
Welcome to codebustler



# Function with parameters

Function parameters are listed inside the parentheses () in the function definition.

Function arguments are the values received by the function when it is invoked.

Inside the function, the arguments (the parameters) behave as local variables.

```
function add(x, y) {  
  return x + y;  
}  
  
console.log(add(10, 20));
```

Example :

> Console × +

30

Hint: hit control+c anytime to enter REPL.

✚ □





# Local Variables

Variables declared within a JavaScript function, become **LOCAL to the function**.

Local variables can only be accessed from within the function.

Example :

```
// code here can NOT use carName

function myFunction() {
  let carName = "Volvo";
  // code here CAN use carName
}

// code here can NOT use carName
```





@CodeBustler

>> Swipe



Also Checkout  
This Awesome  
Posts



Tap Here



@CodeBustler







@CodeBustler



**@CodeBustler**

Mallikarjun Mavnoor | Web Developer

*FOLLOW FOR MORE !*  
*FOLLOW FOR MORE !*  
*FOLLOW FOR MORE !*  
*FOLLOW FOR MORE !*  
*FOLLOW FOR MORE !*  
*FOLLOW FOR MORE !*



Please Like This Post

Save It For Later

