**Java Script**

**Functions**:

**Function** 🡺 block of code to perform a specific task

**Syntax**:

Graphical user interface, application

Description automatically generated

* **function** 🡪 keyword
* **printName** 🡪 function name
* **()** 🡪 parentheses
  + **()** 🡪 no parameters
  + (**param1, param2,…)** 🡪 include parameters

parameters: a way to pass information to the function

Ex 🡪 for function takes parameter **age**

Graphical user interface, text

Description automatically generated

**How to execute the function:**

* To execute the function 🡪 we should call or invoke the function
* calling a function without parameter, using function name and parentheses ()

**Ex:**

Graphical user interface, text

Description automatically generated with medium confidence

* calling a function with one parameter

**Ex**

**Logo

Description automatically generated**

25 🡪 is the function argument

So, the function will receive the value 25 when it is invoked

**Full Ex for function without parameters**

A screenshot of a computer

Description automatically generated with medium confidence

**Full Ex for a function takes one parameter**

A screenshot of a computer

Description automatically generated with medium confidence

Note: what happened if we don’t send a value to a function takes param , Ex 🡪age🡪 undefined

Graphical user interface, text

Description automatically generated

**Examples**:

A screenshot of a computer

Description automatically generated with medium confidence

🡪we send age but we don’t send shouldPrintAge 🡪 so when we print shouldPrintAge 🡪 undefined

Text

Description automatically generated

If we check that the function **received** the value of **shouldPrintAge** parameter or not

If received 🡪 true

If not 🡪 false

Text

Description automatically generated

Here we send true which is Boolean type so when using if condition to check the value exists or not

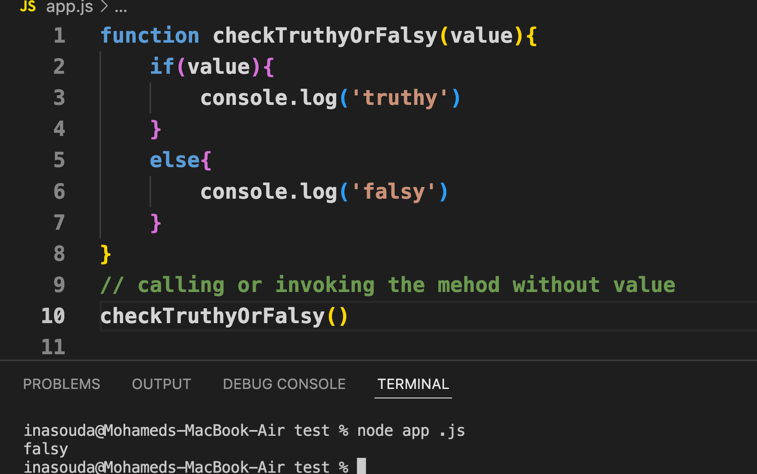
**Note**: if we send any **value type**, if condition will be executed as well, as it the function received value of the parameter.

**Ex:**

Text

Description automatically generated

**Ex:** checkTruthyOrFalsy

 Text

Description automatically generated

Text

Description automatically generated

**Note**:

* + if the argument type Number
    - if the argument (value) is **0**  **🡪** the result is **false**
    - otherwise 🡪 the result is **true,** For example, (1,-1,2,-2,etc
  + if the value **“0”** 🡪 the result is **true** 🡪 value type is string

**Text

Description automatically generated**

**Built-in function:**

**Ex:** Math object is a built-in object has methods we can use.

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Math>

Ex: Math.random()

Math.trunc(number)

Text

Description automatically generated **Text

Description automatically generated**

**Graphical user interface, text, application, chat or text message

Description automatically generated**

**Returning values:**

* To return a value from a function 🡪 using keyword **return** and the value or the parameter name

Ex: return true. Ex: return fName Ex: return age

* **receive a value**
  + invoking the function
  + defining a variable to receive what the function returned

Ex: let fName = getName ()

Or we can print it directly by invoking the method and the returning value will be printed

Ex: console.log (getName ())

**Ex:**

Text

Description automatically generated

**- descCheck** function takes 2 parameters **(valueA, valueB),** and return true or false (Boolean type) according to the if condition.

- To call or invoke the function 🡪 use function name and pass the values through () 🡪 **descCheck (245, 768)**

- return value will end up in result variable

**let result = descCheck (245, 768)**

**Ex:**

Text

Description automatically generated

* sort function takes 3 parameters valueA, valueB, condition

#note: here condition value is the function name 🡪 descCheck

* condition value could be anything else according to the function name
* as condition param = descCheck🡪 so, condition (valueA, valueB) is descCheck (valueA, valueB)
* **if (condition (valueA, valueB))** will be **if (descCheck (valueA, valueB))**

Ex: to get type of return value

Text

Description automatically generated Text

Description automatically generated

* if we need to assign a **default** value to parameter valueA=12
* Ex: function descCheck (**valueA=12**, valueB){}

Notes:

* Function signature:name +param list
* Each function must have it ’s own signature
* Overloading -> same function name but different in param list (type, quantity)

**Arrow function:**

Syntax:

//traditional way

function getData (param1, param2) {}

// arrow function

const getData = (param1, param2) => {}

**Ex:**

Graphical user interface, text, application

Description automatically generated

**Ex: runAndPrint function**

Text

Description automatically generated

takes 3 params

* The first param 🡪method 🡪 takes value which is a **function**
* When we call the function runAndPrint🡪 passing the **first value** to the function runAndPrint 🡪 (a, b) => {

return (a\*b)

}

* The 2nd and 3rd params taking values 10, 20
* So, when we invoke method (params1, params2)🡪

We invoke to the function which passed to method param

* method (params1, params2) 🡪is

(a,b) => {

return a\*b

}

* console.log () which takes 🡪 method(param1, param2)
* param1, param2 🡪 10,20 🡪 (a, b) {

return a\*b 🡪 10\*20

}