**// Crie uma função que receba dois argumentos e retorne a soma dos mesmos.**

*function soma (x,y) {*

*... return x + y;*

*... }*

**// Declare uma variável que receba a invocação da função criada acima, passando dois números quaisquer por argumento, e somando `5` ao resultado retornado da função.**

*> var somaUM = soma (5,5) + 5*

**// Qual o valor atualizado dessa variável?**

*15*

**// Declare uma nova variável, sem valor.**

*var nada*

**/\***

**Crie uma função que adicione um valor à variável criada acima, e retorne a string:**

**O valor da variável agora é VALOR.**

**Onde VALOR é o novo valor da variável.**

**\*/**

*> function math () {*

*... var nada = 2;*

*... return `O valor da variável agora é ` + nada; ----- Ao usar o + com uma string, o restante, ou seja, o valor da variável, será transformado também em String.*

*... }*

*> math ()*

*'O valor da variável agora é 2'*

*OU*

*function math () {*

*... var nada = 10;*

*... return `O valor da variável agora é ${nada}`;*

*... }*

*DA FORMA ABAIXO, O VALOR FOI ATRIBUIDO AO ARGUMENTO PRIMEIRO, NÃO À VARIÁVEL DIRETAMENTE. MAS TAMBÉM É VALIDO PELO EXERCICIO FEITO:*

*var nada;*

*undefinied*

*var nada = function math2 (x) {*

*... return `O valor da variável agora é ${x}`;*

*... }*

// **Invoque a função criada acima.**

*nada (2)*

*'O valor da variável agora é 2'*

**// Qual o retorno da função? (Use comentários de bloco).**

*/\* O retorno da função é o seu argumento (x), ou seja, o valor que lhe for adicionado. Neste caso, 'O valor da variável agora é 2'.\*/*

**/\***

**Crie uma função com as seguintes características:**

**1. A função deve receber 3 argumentos;**

**2. Se qualquer um dos três argumentos não estiverem preenchidos, a função deve retornar a string:**

**Preencha todos os valores corretamente!**

**3. O retorno da função deve ser a multiplicação dos 3 argumentos, somando `2` ao resultado da multiplicação.**

**\*/**

*function zion (a,b,c) {*

*... if (a === undefined || b === undefined || c === undefined) {*

*... return `Preencha todos os valores corretamente!`;*

*... }*

*... else {*

*... return a \* b \* c + 2;*

*... }*

*... }*

*OU*

*function math (a,b,c) {*

*... if (!a || !b || !c) {*

*... return “Preencha todos os valores correctamente!”;*

*... }*

*... return a \* b \* c + 2;*

*...};*

*math (1,1,1)*

*3*

*math (1,1)*

*‘Preencha todos os valores correctamente!’*

**// Invoque a função criada acima, passando só dois números como argumento.**

*math (2,2)*

*'Preencha todos os valores correctamente!'*

**// Qual o resultado da invocação acima? (Use comentários para mostrar o valor retornado).**

/\* O resultado é a string: *'Preencha todos os valores correctamente!'* \*/

**// Agora invoque novamente a função criada acima, mas passando todos os três argumentos necessários.**

*math (2,2,2)*

*10*

**// Qual o resultado da invocação acima? (Use comentários para mostrar o valor retornado).**

*/\* O resultado da função é 10, que é o resultado de 2 \* 2 \* 2 + 2 \*/*

**/\***

**Crie uma função com as seguintes características:**

**1. A função deve receber 3 argumentos.**

**2. Se somente um argumento for passado, retorne o valor do argumento.**

**3. Se dois argumentos forem passados, retorne a soma dos dois argumentos.**

**4. Se todos os argumentos forem passados, retorne a soma do primeiro com o segundo, e o resultado, dividido pelo terceiro.**

**5. Se nenhum argumento for passado, retorne o valor booleano `false`.**

**6. E ainda, se nenhuma das condições acima forem atendidas, retorne `null`.**

**\*/**

*function master (a,b,c) {*

*... if (a != undefined && b === undefined && c === undefined) {*

*... return a;*

*... } else if (a != undefined && b != undefined && c === undefined) {*

*... return a + b;*

*... } else if (a != undefined && b != undefined && c != undefined) {*

*... return (a + b) / c;*

*... } else if (a === undefined && b === undefined && c === undefined) {*

*... return `false`;*

*... } return `null`;*

*... }*

**// Invoque a função acima utilizando todas as possibilidades (com nenhum argumento, com um, com dois e com três.) Coloque um comentário de linha ao lado da função com o resultado de cada invocação.**

*/\* master ()*

*'false' \*/*

*/\* master (2)*

*2 \*/*

*/\* master (5,5)*

*10 \*/*

*/\* master (2,2,10)*

*0.4 \*/*