



Programação

Curso Introductório - Programa de Bolsas

Material de Nivelamento



Te damos boas-vindas a esta etapa do processo seletivo do programa de bolsas!

O processo seletivo do programa de bolsas ainda contará com algumas etapas, mas sua jornada de transformação em um(a) profissional digital já começou!

O que é o Nivelamento?

O Curso Introdutório (também conhecido como "Nivelamento") é uma das principais fases no ingresso de todos os alunos da Digital House e uma das etapas do processo seletivo do programa de bolsas. Através do estudo deste material, você entrará em contato com alguns conceitos e termos básicos de Programação, desenvolvendo a base necessária para responder ao **Desafio Online**, próxima etapa do processo seletivo do programa de bolsas.

Atenção: você irá para a etapa de desafio se atingir 07 pontos ou mais no teste de lógica. Atente-se para o prazo de conclusão do teste, indicado no e-mail que você recebeu.

Estimamos que o estudo do material consumirá de 15 a 25 horas, portanto **programe-se para consumir todo o conteúdo antes de 11/02/2022**, data em que o desafio será enviado ao seu e-mail para você completá-lo.

Quais são as etapas do processo seletivo?

As etapas e seus respectivos prazos são:

1. **Até 09/02** Realização dos testes e estudo para o desafio;
2. **De 11 até 14/02** Conclusão do desafio (para aprovadas na 1ª etapa)
3. **De 17/02 até 04/03** Participar da entrevista (melhores colocadas na 2ª etapa)
4. **Em 08/03:** Boas-vindas às candidatas selecionadas

Se tiver alguma dúvida sobre o processo seletivo, confira as Perguntas Frequentes no site do programa, localizado no rodapé da página.

Se a dúvida persistir, entre em contato com a nossa equipe através do contato-bolsas@digitalhouse.com.

No e-mail, indique seu nome completo, e-mail cadastrado no programa e qual sua dúvida para que possamos ajudá-lo.



Lembre-se: por ser um processo seletivo, os candidatos não têm o auxílio de professores para a seleção ser justa com todos os candidatos. O estudo do material de nivelamento é o material ideal para você se preparar para o desafio :)

Dica: configurando vídeos para o português

A depender do seu curso escolhido, você receberá alguns vídeos para assistir e parte deles podem estar em Espanhol ou Inglês.

Se for o caso deste material e você não tiver domínio do Espanhol ou Inglês, e não souber como colocar legendas em Português, recomendamos que explore este tutorial: <http://bit.ly/2GGjJOz>

Se precisar traduzir uma página do seu navegador, sugerimos seguir os seguintes passos no Google Chrome:

- 1-No computador, abra o Chrome;
- 2-Acesse a página da Web que está em outro idioma;
- 3-Na parte superior, clique em Traduzir;
- 4-Pronto! O Chrome traduzirá a página.



Lista de Materiais

- [Introdução ao VS Code](#)
- [O que é Programação?](#)
- [Possibilidades](#)
- [Algoritmo](#)
- [Linguagens de Programação](#)
- [Introdução a Variáveis](#)
- [Criando Variáveis](#) (PDF)
- [Criando Variáveis](#) (PPT)
- [Tipos de Dados](#) (PDF)
- [Tipos de Dados](#) (PPT)
- [Usando Variáveis](#) (PDF)
- [Usando Variáveis](#) (PPT)
- [Operadores Aritméticos](#) (PDF)
- [Operadores Aritméticos](#) (PPT)
- [Introdução a Condicionais](#)
- [Condicional IF](#) (PDF)
- [Condicional IF](#) (PPT)
- [Operadores Relacionais](#) (PDF)
- [Operadores Relacionais](#) (PPT)
- [Condicional IF ELSE](#) (PDF)
- [Condicional IF ELSE](#) (PPT)
- [Condicional ELSE IF](#) (PDF)
- [Condicional ELSE IF](#) (PPT)
- [Operadores Lógicos](#) (PDF)
- [Operadores Lógicos](#) (PPT)
- [Introdução a Loops](#) (PDF)
- [Loop For](#) (PDF)
- [Loop For](#) (PPT)
- [Loop com Array](#) (PDF)
- [Loop com Array](#) (PPT)
- [Break e Continue](#) (PDF)
- [Break e Continue](#) (PPT)
- [Introdução a Funções](#)
- [Declarando Funções](#) (PDF)
- [Declarando Funções](#) (PPT)
- [Funções Parametrizadas](#) (PDF)
- [Funções Parametrizadas](#) (PPT)
- [Usando Funções](#)
- [Funções em JS](#) (PDF)
- [Funções em JS](#) (PPT)



Lista de Exercícios

O QUE É PROGRAMAÇÃO?

1. O conceito do computador programável surgiu através da máquina analítica de Babbage com qual intuito?
 - ☐ Ajudar a escrever código.
 - ☐ Fazer contas sem qualquer interferência.
 - ☒ Realizar cálculos matemáticos a partir de instruções.
2. A programação afeta a nossa relação com o mundo?
 - ☒ Sim
 - ☐ Não

POSSIBILIDADES

1. A programação nos possibilita criar vários aplicativos e sites. Cite alguma solução de tecnologia que, sem ela, você não conseguiria ficar.

(Exemplos: iFood, Smartphone, Uber, WhatsApp, Sites de Meteorologia)

Resposta pessoal. Spotify.
2. Qual dos termos abaixo é o mais utilizado para descrever objetos cotidianos conectados à internet?
 - ☐ Dispositivos conectivos
 - ☒ Internet das coisas
 - ☐ Servidores

ALGORITMO

1. Qual é a melhor definição de algoritmo?
 - ☐ Executar ações
 - ☒ Conjunto de instruções
 - ☐ Linguagem de programação



2. Algoritmo é uma sequência de instruções com um objetivo claro. Qual das alternativas abaixo exemplifica um algoritmo?
- ☐ Bolo é uma massa de farinha, salgada ou doce, assada em formas ou tabuleiros de diferentes formatos
 - ☐ Faça um bolo
 - ☒ Junte os ingredientes, mexa tudo até formar uma massa homogênea e asse o bolo até dourar

LINGUAGENS DE PROGRAMAÇÃO

1. Qual tipo de linguagem de programação se assemelha mais à linguagem humana?
- ☐ Linguagem de Baixo nível
 - ☒ Linguagem de Alto nível
 - ☐ Código Binário
2. Com base no texto lido anteriormente, escolha a alternativa correta:
- ☒ No navegador temos a visão de usuário e no editor de códigos temos a visão de programador.
 - ☐ No navegador temos a visão de programador e no editor de códigos temos visão de usuário.
 - ☐ Nenhuma das alternativas
3. Qual das alternativas abaixo contém somente linguagens de alto nível?
- ☐ Javascript, Assembly, Cobol
 - ☐ Javascript, PHP, Código de máquina
 - ☒ Javascript, PHP, Java

INTRODUÇÃO À VARIÁVEIS

1. Por que armazenar informações em programação?
- ☐ Porque os seres humanos guardam tudo
 - ☒ Para que possamos reutilizar essas informações, seja para consulta ou para resolver problemas
 - ☐ Em programação, devemos guardar as informações para nunca mais usar
 - ☐ Para resolver os problemas do sistema



2. Onde iremos guardar informações dentro do nosso código?

- ☐ Em uma caixa
- ☐ No sistema
- ☒ Nas variáveis.
- ☐ Na Tupperware.

3. As variáveis precisam de um nome?

- ☒ Sim
- ☐ Não

CRIANDO VARIÁVEIS

Exercício

Vamos escrever nossas primeiras variáveis juntos. Você irá usar tudo que aprendeu anteriormente!

Declare duas variáveis chamadas `idade` e `peso`, e atribua um valor numérico a ambas.

Dica: Criação de variáveis requer o uso de uma palavra reservada

Atenção! Resolva este exercício no VSCODE.

Exercício

Agora para deixar todo esse conhecimento fixo na mente, precisamos praticar um pouco mais!

Declare 4 variáveis: `nome`, `sobrenome`, `numeroDaSorte` e `idade`. Atribua um valor para cada uma delas, conforme o tipo.

Dica: Verifique se os nomes das variáveis estão escritos igual ao enunciado

Atenção! Resolva este exercício no VSCODE.



TIPOS DE DADOS

Exercício

Escreva no arquivo script as variáveis expostas abaixo substituindo os **tipos de dados** citados por **valores**, conforme vimos no material teórico:

```
var idade = Number  
var sobrenome = String  
var salarioMinimo = Float
```

Dica: Atente-se para os tipos de dados esperados por cada variável.

Atenção! Resolva este exercício no VSCODE.

Quiz

1. Com base no que estudamos sobre os tipos de variáveis, e na prática anterior, selecione a alternativa que CONTÉM ERRO na escrita do código:
 - ☒ var nome = Hendy Almeida
 - ☐ var Nome = "Hendy Almeida"
 - ☐ var nome = "HeNdY AlMeida"
 - ☐ var variavel = "texto qualquer"
2. Qual tipo de dados utilizamos para guardar uma lista?
 - ☐ Variáveis
 - ☒ Array
 - ☐ String
 - ☐ Boolean



Exercício de Fixação

Vamos continuar praticando a declaração de variáveis e atribuição de valores:

Declare cinco variáveis com a nomenclatura a seguir e atribua a tais, um valor que atenda o tipo de dado indicado:

- nome (string)
- idade (number)
- altura (float)
- possuiPet (boolean)
- filmesFavoritos (array)

Dica: Leia o enunciado com atenção. O tipo de dado solicitado por cada variável está indicado nele.

Atenção! Resolva este exercício no VSCODE.

USANDO VARIÁVEIS

1. Temos uma variável chamada apresenta o código correto para atribuir um novo valor a variável? da frutaDaEstacao que já foi declarada e armazena o nome de uma fruta. Qual a alternativa?
 - ☐ var frutaDaEstacao = "Maçã"
 - ☐ var = "Maçã"
 - ☐ frutaDaEstacao "Maçã"
 - ☒ frutaDaEstacao = "Maçã"
2. Qual comando correto para concatenar as variáveis nome e sobrenome e passar esse resultado para a variável nomeCompleto?
 - ☒ var nomeCompleto = nome + sobrenome
 - ☐ var nomeCompleto = "nome + sobrenome"
 - ☐ var nome = nomeCompleto
 - ☐ var nomeCompleto = "Nome Completo"



Exercício de Fixação

Temos um desafio para você!

Precisamos criar um código que gere cartão de visita com nome, sobrenome e profissão dos clientes, para isso:

- Crie uma variável para guardar cada informação (cada uma deve receber um dos 3 nomes acima),
- Depois faça a concatenação das informações em uma nova variável chamada `cartaoDeVisita`.

Exemplo:

```
nome: Taylor  
sobrenome: Alvarenga  
profissão: Programadora
```

Cartão de visita esperado: "**Taylor Alvarenga - Programadora**"

OBS: no desenvolvimento da variável `cartaoDeVisita` deixar um espaço entre caractere e caractere de concatenação.

Dica: Atenção aos detalhes, como hífen e espaços.

Atenção! Resolva este exercício no VSCODE.

OPERADORES ARITMÉTICAS

1. Qual das operações abaixo exemplifica o uso do operador módulo?

- ☐ `a / b`
- ☐ `a * b`
- ☒ `a % b`
- ☐ `a - b`

2. Qual das alternativas abaixo exemplifica os operadores incremento e o decremento?

- ☐ `++ | *`
- ☒ `++ | --`
- ☐ `-- | +`
- ☐ `-- | /`



Exercício: Cálculos

Declare 6 variáveis de memória com os seguintes nomes: numeroA, numeroB, soma, subtracao, multiplicacao e divisao (observem que nomes de variáveis não tem acento).

Dica: Use as variáveis para fazer os cálculos

Atenção! Resolva este exercício no VSCODE.

Exercício: Cálculo de Lucros

Uma empresa separou os seus ganhos dos meses nas seguintes variáveis:

```
let janeiro = 100
let fevereiro = 300
let março = 90
```

Ela precisa da sua ajuda para saber o total do lucro adquirido. Sua missão é usar os operadores com as variáveis para realizar esse cálculo e use o console.log para imprimir o resultado!

Para lhe auxiliarmos, já deixamos o código inicial:

```
let janeiro = 100
let fevereiro = 300
let marco = 90
let valorTotal = ?
```

Dica: Lembre-se de usar as variáveis na soma do valor total

Atenção! Resolva este exercício no VSCODE.



Exercício: Cálculo de Média

Um sistema de gestão escolar quer calcular a média das notas de um aluno.

Ele guarda em variáveis as notas de cada um dos trabalhos realizados:

```
let trabalhoDeHistoria = 8.0
let trabalhoDeMatematica = 7.0
let trabalhoDeCiencia = 10
let trabalhoDeGeografia = 9.5
```

Sua missão é completar o código para fazer com que o sistema calcule a média das notas, somando-as e dividindo o total pela quantidade de trabalhos (variável quantidadeDeTrabalhos).

Atribua o resultado dessa conta a uma variável chamada **media**

Dica: Você se lembra como fazer o cálculo de média? Basta somar os valores e dividir pela quantidade de notas!

Atenção! Resolva este exercício no VSCODE.

INTRODUÇÃO À CONDICIONAIS

1. Qual termo usamos em programação para representar verificações?

- ☐ Perguntas e respostas
- ☐ Variáveis
- ☒ Condicional
- ☐ Códigos

2. Qual dos exemplos abaixo podem representar uma condicional?

- ☐ Se eu quero eu simplesmente compro!
- ☒ Se tiver bolo, eu vou comer bolo com café, se não, irei comer bolacha com café!
- ☐ Você sabia que o céu é azul?
- ☐ Tomara que o caminho seja curto!



CONDICIONAL IF

1. Qual das opções abaixo apresenta a sintaxe correta do condicional if?

- ☐ if a == b { console.log('A condição é verdadeira') }
- ☒ if (a == b) { console.log('A condição é verdadeira') }
- ☐ if (a == b console.log('A condição é verdadeira'))
- ☐ if (console.log('A condição é verdadeira')) { a == b }

2. Em javascript, existe alguma forma além do if que podemos utilizar para criar uma condição?

- ☒ Sim
- ☐ Não

Exercício

Precisamos de um código que verifique a idade do usuário para ver se já tem idade mínima para adquirir carta de habilitação. Para isso, teremos uma variável nome que guarda um valor do tipo string, e uma variável idade que guarda um valor do tipo numérico.

O sistema deve cumprimentar o usuário independente da idade, o resultado seria assim:

Caso o usuário tenha idade mínima para dirigir imprima:

"Olá, NOME_DO_USUARIO_AQUI"

"Você passou no nosso teste e já pode dirigir!".

Caso o usuário não tenha a idade mínima para dirigir imprima somente o cumprimento:

"Olá, NOME_DO_USUARIO_AQUI"

Dica: Para verificar valores maiores ou igual a 18, podemos usar >=, exemplo: (idade >=18)

Atenção! Resolva este exercício no VSCODE.



Exercício: Conta Bancária

Queremos um código que oriente o usuário de acordo com o saldo da conta bancária. Para isso precisamos de uma variável saldo que guarda um número decimal(float), e imprime uma mensagem de acordo com a situação financeira.

Se o saldo for **maior que o (zero)** imprima:

"Seu saldo está positivo! Gostaria de fazer um investimento?"

Se o saldo for **menor que o (zero)** imprima:

"Seu saldo está negativo! Gostaria de fazer um empréstimo?"

Dica: Utilize a condicional if para verificar o saldo

Atenção! Resolva este exercício no VSCODE.

OPERADORES RELACIONAIS

1. Qual das alternativas abaixo melhor exemplifica os operadores relacionais de igualdade em relação à comparação por valor e tipo de dado?

- ☐ =
- ☐ ==
- ☒ ===

2. Qual das alternativas verifica se a variável a tem um valor a partir de 100?

- ☐ a == 100
- ☐ a => 100
- ☒ a >= 100
- ☐ a > 100



3. Considerando o código abaixo, qual condicional retorna o valor false?

```
var a = 10  
var b = 10
```

- ☐ a == b
- ☐ a === b
- ☒ a < b
- ☐ a <= b

Exercício: Votação

No sistema político atual, temos algumas regras para participar das eleições. A principal delas, é que o voto é obrigatório a partir dos 18 anos, e opcional quando idade for igual a 16!

Diante desse cenário, escreva um código que de acordo com a variável idade informe ao usuário usando `console.log()`, "Você é obrigado a votar" caso ele cumpra o requisito ou "Seu voto é opcional".

Dica: Utilize `if` para criar as condicionais

Atenção! Resolva este exercício no VSCODE.

CONDICIONAL IF ELSE

1. É obrigatório a utilização do `else` em cada `if`?
 - ☐ Sim
 - ☒ Não

2. Qual alternativa apresenta a sintaxe correta?
 - ☒ `if(condição) { //faz alguma coisa } else { // faz outra coisa }`
 - ☐ `if { //faz alguma coisa } else (condição) { // faz outra coisa }`
 - ☐ `if (condição) else { // faz alguma coisa }`
 - ☐ `else (condição) { // faz outra coisa }`



3. De acordo com o código abaixo, qual mensagem vamos imprimir?

```
var grauCelsius = 18;  
if (grauCelsius < 18) {  
    console.log("Está ficando frio. Não esqueça de levar um  
    agasalho!")  
} else {  
    console.log("O clima está agradável! Não esqueça o óculos de  
    sol :)")  
}
```

- ☐ "Está ficando frio. Não esqueça de levar um agasalho!"
☒ "O clima está agradável! Não esqueça o óculos de sol :)"
☐ Nenhuma das alternativas anteriores

Exercício: Autoescola

Vamos otimizar nosso sistema de autoescola!

Caso o usuário tenha 18 anos ou mais, ele pode dirigir, se a afirmação for negativa precisamos imprimir a mensagem "Ops, você ainda não tem a idade mínima para dirigir!".

Parte do código já está pronto, agora é sua vez de utilizar o **else** para dar um feedback ao usuário!

Dica: O else vem logo depois do fechamento de chaves do if!

Atenção! Resolva este exercício no VSCODE.

Exercício: Estrangeiro

Utilize a condicional **if/else** para verificar a variável **estrangeiro** que guarda um valor booleano. Caso o valor seja verdadeiro solicite o Registro Nacional de Estrangeiros(RNE). Se o valor for falso solicite o Cadastro de Pessoa Física (CPF).

Mensagem para estrangeiro: "Você poderia apresentar seu RNE, por favor?"

Mensagem para brasileiro: "Você poderia apresentar seu CPF, por favor?"

Dica: Para verificar se uma variável booleana é verdadeira, basta chama-la entre os parenteses do if, não é necessário utilizar "=== true"

Atenção! Resolva este exercício no VSCODE.



Exercício: Aposentado

Precisamos verificar se uma pessoa pode se aposentar com base na idade. Utilize a condicional `if else` para verificar se a variável `idade` é maior que 65 e imprima uma mensagem para cada caso:

Condição verdadeira : "Você já pode se aposentar"

Condição falsa: "Você ainda não pode se aposentar"

Dica: Utilize operadores relacionais para verificar a idade mínima.

Atenção! Resolva este exercício no VSCODE.

Exercício: Par ou Ímpar?

Dentro do código estará criada uma variável **numeroDaSorte** contendo um número. Sua missão é verificar se o valor da variável é par ou ímpar utilizando o operador relacional módulo (%). Imprima "Par" ou "Ímpar" de acordo com o resultado.

Dica: Para verificar se um número é par, basta fazer a expressão do número dividido por 2 e verificar se o resto é igual à 0 (zero). Exemplo: `(42 % 2 == 0)`

Atenção! Resolva este exercício no VSCODE.



CONDICIONAL ELSE IF

Exercício: Petshop

Precisamos criar um código que ajude um petshop a dizer se o pet está com o peso ideal. E para isso eles deixaram para você as informações que eles usam para fazer essa classificação:

- Abaixo de 4kg = "Abaixo do Peso"
- Maior que 10kg = "Acima do Peso"
- Se tiver entre esses dois valores = "Peso normal"

Utilizando o **ELSE IF** e crie o código para suprir essa necessidade e imprimindo as mensagens de acordo com a lista acima!

Dica: Não esqueça de utilizar ELSE IF

Atenção! Resolva este exercício no VSCODE.

Exercício: Lado Iguais

Precisamos de ajuda para escrever um código que de acordo com a quantidade de lados iguais de um triângulo, exiba com console.log se ele é Equilátero, Isósceles ou Escaleno, sem a necessidade de repetir no código a palavra "triângulo".

- Triângulo Equilátero: possui os 3 lados iguais.
- Triângulo Isósceles: possui 2 lados iguais.
- Triângulo Escaleno: possui 3 lados diferentes

Nesse exercício a variável **ladosIguais** já estará definida.

Dica: Utilize console.log para imprimir.

Atenção! Resolva este exercício no VSCODE.



OPERADORES LÓGICOS

1. Podemos fazer **apenas 3** condições usando o operador AND (&&).

- ☐ Verdadeiro
- ☒ Falso

2. Sobre o operador **OR** (||), qual dos códigos abaixo está **correto**?

- ☐ dia == 'quarta' || 'sábado'
- ☐ dia == 'quarta' && dia == 'sábado'
- ☒ dia == 'quarta' || dia == 'sábado'
- ☐ 'quarta' || 'sábado'

3. Um desenvolvedor precisa de ajuda para escolher o operador correto na sua condição. A ideia dele é exibir uma mensagem "Esse produto não está ativo" quando a variável **produtoAtivo** estiver com o valor **falso**.

```
var produtoAtivo = false
if() {
  console.log('Esse produto não está ativo')
}
```

Qual condição se encaixaria neste código:

- ☐ produtoAtivo != false
- ☐ produtoAtivo || false
- ☐ produtoAtivo && false
- ☒ !produtoAtivo



Exercício: Operadores Lógicos

Queremos criar um código que ajude o usuário a saber se ele pode comprar um produto ou não. Para acontecer a venda, a quantidade do produto no estoque tem que ser maior que zero e o produto está ativo.

Para escrever este código, teremos duas variáveis já definidas: **produtoQtd**, **produtoAtivo**.

Faça uma condição que supra a necessidade acima, se o usuário puder comprar o produto exiba a seguinte mensagem "Você pode finalizar essa compra". Caso contrário exiba "Produto não está disponível para compra".

Dica: Utilize somente 1 if para resolver esse exercício.

Atenção! Resolva este exercício no VSCODE.

Exercício: Pode subir?

Um parque de diversão te contratou para criar um código para ajudar os usuários a saber se eles podem ir em uns dos brinquedos. As regras são:

- Ser maior que 1,50m ou ter 21 anos ou mais.

Diante disso, crie condições que supra essa necessidade, exiba a seguinte mensagem caso o usuário possa usar o brinquedo: "Você pode subir". Caso ele não possa: "Desculpe, você não atende os requisitos para usar o brinquedo".

Vale dizer que para esse exercício você terá duas variáveis: **usuarioAltura** e **usuarioldade**

Dica: Utilize somente 1 if para resolver esse exercício.

Atenção! Resolva este exercício no VSCODE.



Exercício: Será que posso ir ao banco?

Precisamos criar um código que ajude os usuários a saber se podem ir ao banco. Sabemos que o banco está aberto em todos os os dias da semana, exceto em finais de semana. Caso o usuário possa ir ao banco você deve exibir com **console.log** a seguinte mensagem: "Você pode ir ao banco", caso contrário, "O banco está fechado, tente outro dia".

Para esse exercício leve em consideração as seguintes informações:

- Dias da semana pro sistema: segunda, terça, quarta, quinta, sexta, sabado e domingo.
- Você terá a seguinte variável no código: **diaSemana**
- Para esse exercício você deve usar na condição do if o operador **&&** para juntar as condições lógicas junto ao operador diferente de(!=).

Dica: Utilize somente um if para resolver esse exercício

Atenção! Resolva este exercício no VSCODE.

INTRODUÇÃO A LOOPS

1. Selecione TODOS os termos utilizados para definir um trecho de código que foi programado para se repetir.
 - ☒ Loop
 - ☐ Condicional
 - ☒ Laço de repetição
 - ☒ Ciclos
 - ☒ Loop For
2. Quais são as informações que precisamos definir para o loop **for**?
 - ☒ início, condição de parada e incremento
 - ☐ início e incremento
 - ☐ condição de parada e fim



Exercício: Tabuada

Precisamos de um código que calcule a tabuada de multiplicação do 7 e imprima a expressão seguido do resultado.

Exemplo:

$$7 \times 1 = 7$$

$$7 \times 2 = 14$$

$$7 \times 3 = 21$$

$$7 \times 4 = 28$$

$$7 \times 5 = 35$$

$$7 \times 6 = 42$$

$$7 \times 7 = 49$$

$$7 \times 8 = 56$$

$$7 \times 9 = 63$$

$$7 \times 10 = 70$$

Lembre-se de fazer a multiplicação do **7 x 1** até **7 x 10**!

Ah, uma última dica: Você deve usar a variável `i` que é nosso contador para concatenar na hora de montar a mensagem "7 x 1"!

Afinal os valores depois do " x " (1,2,3,4,...) são os valores que mudam de acordo com cada interação do loop.

Dica: Temos código que deve se repetir? Que tal usar o loop `for`?

Atenção! Resolva este exercício no VSCODE.



Exercício: Bom dia!

Vamos criar um robô que manda "Bom dia, grupo!" 7 vezes. Utilize o laço de repetição for para fazer com mais praticidade!

Dica: Fique atento a sintaxe do for, na condição de parada podemos usar o operador relacional que mais se adequa a situação, o importante é fazer o laço repetir 7 vezes.

Atenção! Resolva este exercício no VSCODE.

Exercício: Loop com Array

Uma agência de carros quer exibir os itens do seu catálogo para os clientes. Eles exportaram os nomes dos carros no formato de array e guardamos essas informações na variável listaDeCarros que já vai estar disponível no código.

Crie um loop que atenda o problema acima utilizando uma variável chamada i como contador. Use o console.log para exibir o nome dos carro de acordo com o exemplo abaixo:

"Nome do Carro: NOME_DO_CARRO"

Dica: Você se lembra como pegar um índice do array? Basta trazer nome do array seguido de colchetes com numero da posição.

Atenção! Resolva este exercício no VSCODE.

Exercício: Lucros

Uma empresa separou em uma lista, os valores dos lucros mensais. Com isso você terá no código um array com o seguinte nome: listaDeLucro contendo em cada posição o valor de recebido de cada mês!

```
var listaDeLucro = [100, 30, 300, -10, 600, 10]
```

Seu trabalho será criar um loop que calcule o valor total baseado nessa lista, e coloque o valor em uma variável já existente no código chamada: lucroTotal

Dica: Utilize .length para trazer o tamanho do array.

Atenção! Resolva este exercício no VSCODE.



Exercício: Saldo Negativo

Uma empresa mandou uma lista contendo os números mensais de tudo o que ela faturou, e nosso trabalho é ajudá-los a criar um relatório que exiba em quantos meses eles tiveram o saldo **negativo**.

```
var listaDeGanhos = [10, 30, -10, -5, -1, 40]
```

Com base no array acima, que está disponível no código, faça um loop que verifique quantos meses tiveram valores negativos e armazene a contagem uma variável chamada **totalNegativos** que também está disponível no código.

Dica: Utilize `.length` para trazer o tamanho do array.

Atenção! Resolva este exercício no VSCODE.

Exercício: Frutas

Um sacolão montou uma lista com as frutas que eles vendem, e de acordo com a fruta que o usuário busca eles querem informar se existe a fruta na lista ou não!

```
var listaDeFrutas = [ "Uva", "Banana", "Manga", "Cajá",  
"Pinha"]
```

Você deverá criar um loop que verifique se a fruta contida na variável **busca** existe na lista de frutas do sacolão. Se existe basta exibir uma mensagem, "Sim, temos a fruta banana disponível". Use a variável `busca` para exibir o nome da fruta nessa mensagem de forma dinâmica.

Dica: Utilize `.length` para trazer o tamanho do array.

Atenção! Resolva este exercício no VSCODE.



BREAK E CONTINUE

Exercício: Bingo

Vamos criar um bingo, onde os números da **cartela** serão representados por um array chamado **cartela**, e o número sorteado deve ficar em uma variável chamada **numeroSorteado**. Seu trabalho será verificar se existe o número sorteado na cartela, quando encontrar deve imprimir "Encontrei o número!" e parar o loop!

Dica: Utilize **for**, **if**, e **break** para solucionar esse desafio.

Atenção! Resolva este exercício no VSCODE.

Exercício: Números Pares

Precisamos imprimir somente os números pares de 0 a 20. Mas temos alguns requisitos, você precisa utilizar, o loop **for**, o comando **continue** e o número deve ser o contador do loop! Vamos lá?

Dica: no seu **for**, utilize como contador uma variável chamada **i**.

Atenção! Resolva este exercício no VSCODE.

Exercício: sobrenome da família

Temos uma lista com nomes de todos os integrantes da família mas esqueceram de colocar o sobrenome!

Seu desafio será **imprimir** nome junto com o sobrenome "Macedo" **para cada integrante** da família. Mas temos uma **exceção**, temos um integrante com sobrenome diferente, se tiver algum "Pedro", coloque o sobrenome "Sousa".

Dica: Utilize **for** e **continue** para criar a solução. Acrescente um espaço antes do sobrenome, por exemplo: " Sousa".

Atenção! Resolva este exercício no VSCODE.



Exercício: Baralho

Temos um **array** em uma variável chamada **baralho** e precisamos de um script que procure a carta "Rei" entre as cartas do baralho. Assim que encontrar o "Rei" pare o loop utilizando **break** e exiba um `console.log` a seguinte frase: Encontrei o Rei!

Dica: Utilize `.length` para trazer o tamanho do array.

Atenção! Resolva este exercício no VSCODE.

INTRODUÇÃO A FUNÇÕES

1. Qual das seguintes afirmações podem ser usadas para descrever as **funções** em programação?
 - ☐ É o nome de um tipo de variável
 - ☐ É os recursos que tem dentro do controle da televisão
 - ☒ É forma de agrupar instruções ao computador para utilizarmos em diversos momentos
 - ☐ É a forma que falamos para definir os `if` e `else`
2. Em uma função só podemos utilizar `console.log()`!
 - ☐ Verdadeiro
 - ☒ Falso

DECLARANDO FUNÇÕES

1. Qual palavra reservada usamos para criar funções em Javascript?
 - ☐ `var`
 - ☐ `funções`
 - ☐ `criar`
 - ☒ `function`



2. O que está errado na criação da função abaixo?

```
function calcular{  
  console.log(A + B)  
}
```

- ☐ Não podemos usar console.log em funções
- ☒ Toda função deve ter os parênteses, que pode receber informações para a função
- ☐ Não podemos colocar nome em português
- ☐ A função não está errada!

3. Podemos usar **caracteres especiais** em **nome de funções**?

- ☐ Sim
- ☒ Não

Exercício

Um mercado tem uma lista de produtos, que eles querem exibir para todo novo cliente que chegar! Como o processo é um pouco trabalhoso e repetitivo eles precisam do seu conhecimento em funções para ajudá-los.

Eles já tem um código, porém toda vez eles precisam reescrever o código para exibir ao cliente:

```
var lista = [ 'Leite', 'tomate', 'Biscoito', 'Tapioca']  
for(var i = 0; i < lista.length; i++){  
  console.log(lista[i])  
}
```

Seu trabalho é melhorar esse código, para ficar mais simples utilizá-lo. Precisamos que você transforme o código acima em uma **função** chamada **listarProdutos**.

Dica: Para declarar uma função utilize o comando function.

Atenção! Resolva este exercício no VSCODE.



Exercício: Cartão de Visita

Crie uma função chamada **cartaoDeVisitas**, ela deverá imprimir o seu nome em conjunto com sobrenome!

Para isso lembre da estrutura de que uma função deve ter. E também como podemos exibir informações no console do playground. Se precisar, basta volta no conteúdo de revisão desse módulo.

Dica: Para declarar a função utilize o comando `function`.

Atenção! Resolva este exercício no VSCODE.

Exercício: Tabuada do Sete

Uma professora quer ajudar os alunos a decorarem a tabuada do 7 exibindo o resultado para eles! Crie uma função chamada **tabuadaDoSete** que **imprima** a tabuada do sete utilizando um loop. O resultado da sua função deve ser:

$7 \times 1 = 7$

$7 \times 2 = 14$

$7 \times 3 = 21$

$7 \times 4 = 28$

$7 \times 5 = 35$

$7 \times 6 = 42$

$7 \times 7 = 49$

$7 \times 8 = 56$

$7 \times 9 = 63$

$7 \times 10 = 70$

Nesse exercício, não é necessário executar sua função na solução.

Dica: Lembre-se de fazer a multiplicação do 7×1 até 7×10 ! Última dica, você pode usar a variável `i` que é nosso contador para concatenar na hora de montar a mensagem " 7×1 ". Afinal os valores depois do X (1,2,3,4...) são os valores que mudam de acordo com cada loop.

Atenção! Resolva este exercício no VSCODE.



FUNÇÕES PARAMETRIZADAS

1. Sobre a criação de uma função é **correto** afirmar que:

- ☐ Toda função obrigatoriamente deve retornar algum valor.
- ☐ Toda função deve ser definida para receber pelo menos um parâmetro.
- ☐ Sempre que a linha de um return for executada a função será finalizada.
- ☐ O uso de parâmetro é obrigatório na função

2. O que há de errado na criação da função abaixo:

```
function maiorNumero() {  
    if(a > b) {  
        return a;  
    } else {  
        return b;  
    }  
}
```

- ☐ A função deveria ser definida para receber a e b como parâmetros, no entanto, não está definida para não receber nada.
- ☐ A função não está retornando o maior número, como o nome dela sugere.
- ☒ A função não chega até o final por que executa um return antes.

Exercício: Menor Número

Escreva uma função chamada **menorNumero**. Ela deve receber dois números como **parâmetro** e **retornar** o menor entre eles. Caso os números sejam iguais, basta que se retorne qualquer um deles.

Dica: A sua função deve ser definida (preparada) para receber dois parâmetros. Você pode chamar esses parâmetros de a e b, por exemplo.

Atenção! Resolva este exercício no VSCODE.



Exercício: Autonomia

Os engenheiros de uma montadora estão projetando o computador de bordo de um carro. Eles precisam de uma função que possa calcular a autonomia atual do automóvel, em outras palavras, quantos quilômetros ele consegue andar com a quantidade de combustível atual. A autonomia pode ser obtida multiplicando a quantidade de combustível pelo rendimento. Será que você consegue ajudá-los?

Escreva uma função chamada **autonomia**:

```
function autonomia(quantidadeDeCombustivel, rendimento) {  
    ...  
}
```

Essa função deve receber dois parâmetros:

- O primeiro, que represente a quantidade de combustível que está no tanque
- O segundo, que represente o rendimento do automóvel

A função deve retornar a autonomia do automóvel.

Lembre-se: a autonomia do automóvel pode ser obtido multiplicando o rendimento pela quantidade de combustível presente no tanque.

Dica: Lembre de retornar `quantidadeDeCombustivel * rendimento`.

Atenção! Resolva este exercício no VSCODE.



Exercício: CalculaValorDevido

A lavanderia **DigitalLaundry** lava roupa por quilo. Ela cobra dos seus clientes R\$ 5,00 por cada quilo de roupa suja. Atualmente, eles usam um caderninho e uma calculadora para descobrir o valor que cada cliente tem a pagar. Precisamos automatizar essa empresa!

Escreva uma função **calculaValorDevido**

```
function calculaValorDevido(pesoDaRoupaSuja) {  
    ...  
}
```

A função recebe como parâmetro o peso de roupa suja (em quilos) e deve retornar o valor a ser cobrado do cliente.

Dica: Lembre de retornar $\text{pesoDaRoupaSuja} * 5$, sendo que 5 é o valor que o cliente deve pagar para cada quilo de roupa suja a ser lavada.

Atenção! Resolva este exercício no VSCODE.

Exercício: CalculaValorDevido2

Depois de nossa consultoria, a lavanderia **DigitalLaundry** percebeu que poderia deixar a sua cobrança mais sofisticada e justa. Ela decidiu cobrar **R\$10,00 fixo**, a título de taxa de serviço (independente da quantidade de roupa), mais **R\$ 3,00 por quilo** de roupa suja. Reescreva a função **calculaValorDevido**

```
function calculaValorDevido(pesoDeRoupaSuja) {  
    ...  
}
```

Essa função recebe como único parâmetro a quantidade de roupa suja. Ela deve retornar o valor a ser cobrado do cliente usando a **nova política de preços**.

Dica: Perceba que o valor a ser retornado é o valor fixo (10) mais o resultado da multiplicação do peso da roupa suja por 3.

Atenção! Resolva este exercício no VSCODE.



USANDO FUNÇÕES

Exercício

Para este exercício considere uma função que já foi definida, ela se chama **validaCPF**.

Esta função recebe um parâmetro que representa um possível CPF e tudo que ela faz resume-se em um único objetivo: retornar true caso o parâmetro seja um CPF válido ou false caso contrário.

Eis aqui algo legal sobre funções: tudo que você precisa para utilizá-las é saber **o que ela faz**. Você não precisa ver **como** ela faz.

Utilize a função validaCPF para imprimir "CPF válido" caso o cpf "576.524.020-85" seja válido. Caso contrário, imprima "CPF inválido"

Dica: Lembre de passar o "576.524.020-85" entre aspas

Atenção! Resolva este exercício no VSCODE.

Exercício

Uma rede de cinemas quer implementar um sistema para controlar a entrada nas suas salas. Os clientes, antes de entrar na sala, devem apresentar o RG em uma câmera. A câmera seria capaz de ler a data de nascimento do cliente e, caso ele não tenha idade suficiente para assistir a sessão, o acesso dele não seria autorizado.

Um dos programadores já fez uma função que calcula a idade com base na data de nascimento. Ela recebe como parâmetro uma data de nascimento no formato dd/mm/aaaa (dia com dois dígitos, mês com dois dígitos e ano com quatro dígitos) e retorna idade da pessoa hoje. Precisamos agora que você escreva a função deixaEntrar(dataDeNascimento, censura). A função deve receber dois parâmetros:

- O primeiro deve representar a data de nascimento do cliente no formato dd/mm/aaaa
- O segundo deve representar a censura da sessão, ou seja, a idade mínima do cliente para que ele possa acessar a sala.

A função deve retornar **true** caso o cliente tenha idade maior ou igual a censura e **false** caso contrário.

Dica: Você deve utilizar a função calcularIdade dentro da função deixaEntrar

Atenção! Resolva este exercício no VSCODE.



Exercício

Uma loja virtual permite a seus visitantes filtrar produtos pelo preço. Existe um array com os preços dos produtos. Um programador já criou uma função `maisBaratosQue(valor, precos)` que retorna um array com os preços dos produtos mais baratos que o valor passado como parâmetro. Outro programador já criou uma função `maisCarosQue(valor, precos)` que retorna um array com os preços mais caros que o valor passado como parâmetro. Chegou sua vez!

Crie uma função `precosEntre(valorMenor, valorMaior, precos)` que deve utilizar as funções `maisBaratosQue` e `maisCarosQue` para retornar os preços que estão entre o `valorMenor` e o `valorMaior`. Sua função deve receber então dois parâmetros:

- `valorMenor` para representar o valor mínimo dos preços a serem listados
- `valorMaior` para representar o valor máximo dos preços a serem listados
- `precos` para representar um array com os preços dos produtos

Ela deve retornar um array com todos os preços entre `valorMenor` e `valorMaior`

Dica: Para utilizar uma função não é necessário o comando `function`, somente o nome da função seguido de parênteses e parâmetros se necessário.

Atenção! Resolva este exercício no VSCODE.

FUNÇÕES EM JS

1. Qual das alternativas apresenta somente funções nativas que podemos usar com **arrays**?
 - ☐ `pop()`, `Math.min()`, `toUpperCase()`
 - ☐ `push()`, `pop()`, `indexOf()`
 - ☐ `round()`, `min()`, `max()`
 - ☐ `array()`, `addArray()`, `removeArray()`
2. Sabemos que a função `Math.min()` recebe parâmetros, compara e retorna o menor valor entre eles. Quantos parâmetros podemos passar para que ocorra a comparação?
 - ☐ Somente 1 parâmetro por vez
 - ☐ Somente 2 parâmetros por vez
 - ☐ Podemos passar 2 ou mais parâmetros por vez
 - ☐ Não precisamos passar parâmetros



3. Para identificar o índice de um valor no array, utilizamos a função `indexOf()`. Qual será o retorno do código abaixo?

```
var linguagensDeProgramacao = ["Javascript", "PHP", "Node.js", "Cobol", "C#",  
"Python"]  
  
linguagensDeProgramacao.indexOf("Node.js")
```

- ☐ Retorna "Node.js"
- ☐ Retorna 2
- ☐ Retorna 3
- ☐ Retorna linguagensDeProgramacao

Exercício: estava presente na aula?

Nas escolas do futuro não haverá necessidade de chamadas. Ao entrar em sala, uma câmera fará o reconhecimento facial do aluno e registrará sua presença!

Imagine que, a cada aula, um array guardasse o nome de todos os alunos que estiveram presentes na aula.

Escreva uma função `estavaPresenteNaAula(nomeDoAluno, nomesDosPresentes)`

Essa função deve receber dois parâmetros:

- O primeiro deve representar o nome de um aluno
- O segundo deve representar um array com o os nomes dos alunos presentes.

A sua função deve retornar **true** caso o nome do aluno passado como parâmetro estiver entre os presentes. **false** caso contrário.

Dica: Não é necessário percorrer o array inteiro com um `for`. Lembre da `indexOf`

Atenção! Resolva este exercício no VSCODE.



Exercício: transformaParaMaiusculo

Algumas vezes precisamos padronizar as informações antes de guardá-las no nosso sistema.

Escreva uma função **transformaParaMaiusculo(palavras)**. Essa função recebe como parâmetro um array de palavras. Ela deve retornar um array com a versão MAIÚSCULA de todas as palavras contidas no array passado como parâmetro.

Dica: lembre da função `.toUpperCase()`

Atenção! Resolva este exercício no VSCODE.

Exercício: Gerar Dezenas

Uma fábrica de biscoitos da sorte precisa de um sistema. O sistema deve gerar seis dezenas aleatórias para serem impressas. Sabe como é: Palpites para seus clientes jogarem na mega sena! O seu trabalho é escrever uma função **gerarDezenas()**. Essa função não recebe nenhum parâmetro. Tudo que ela deve fazer é gerar e retornar um array contendo seis números aleatórios entre 1 e 60.

Dica: Lembre-se da função `Math.random()` e da `Math.round()`

Atenção! Resolva este exercício no VSCODE.

Exercício: Máximo e Mínimo

Escreva uma função **maxmin** (a, b, c, d, e) que receba cinco números como parâmetros.

Sua função deve retornar um array onde o primeiro elemento é o menor dos parâmetros da função **maxmin** e o segundo elemento é o maior dos parâmetros da função **maxmin**.

Dica: Lembre-se da função `Math.max(...)` e da `Math.min(...)`

Atenção! Resolva este exercício no VSCODE.



Gabarito dos Quizzes

O QUE É PROGRAMAÇÃO?

1. O conceito do computador programável surgiu através da máquina analítica de Babbage com qual intuito?

- ☐ Ajudar a escrever código.
- ☐ Fazer contas sem qualquer interferência.
- ☒ Realizar cálculos matemáticos a partir de instruções.

2. A programação afeta a nossa relação com o mundo?

- ☒ Sim
- ☐ Não

POSSIBILIDADES

1. A programação nos possibilita criar vários aplicativos e sites. Cite alguma solução de tecnologia que, sem ela, você não conseguiria ficar.

(Exemplos: iFood, Smartphone, Uber, WhatsApp, Sites de Meteorologia)

Resposta pessoal.

2. Qual dos termos abaixo é o mais utilizado para descrever objetos cotidianos conectados à internet?

- ☐ Dispositivos conectivos
- ☒ Internet das coisas
- ☐ Servidores

ALGORITMO

1. Qual é a melhor definição de algoritmo?

- ☐ Executar ações
- ☒ Conjunto de instruções
- ☐ Linguagem de programação



2. Algoritmo é uma sequência de instruções com um objetivo claro. Qual das alternativas abaixo exemplifica um algoritmo?
- ☐ Bolo é uma massa de farinha, salgada ou doce, assada em formas ou tabuleiros de diferentes formatos
 - ☐ Faça um bolo
 - ☒ Junte os ingredientes, mexa tudo até formar uma massa homogênea e asse o bolo até dourar

LINGUAGENS DE PROGRAMAÇÃO

1. Qual tipo de linguagem de programação se assemelha mais à linguagem humana?
- ☐ Linguagem de Baixo nível
 - ☒ Linguagem de Alto nível
 - ☐ Código Binário
2. Com base no texto lido anteriormente, escolha a alternativa correta:
- ☒ No navegador temos a visão de usuário e no editor de códigos temos a visão de programador.
 - ☐ No navegador temos a visão de programador e no editor de códigos temos visão de usuário.
 - ☐ Nenhuma das alternativas
3. Qual das alternativas abaixo contém somente linguagens de alto nível?
- ☐ Javascript, Assembly, Cobol
 - ☐ Javascript, PHP, Código de máquina
 - ☒ Javascript, PHP, Java

INTRODUÇÃO À VARIÁVEIS

1. Por que armazenar informações em programação?
- ☐ Porque os seres humanos guardam tudo
 - ☒ Para que possamos reutilizar essas informações, seja para consulta ou para resolver problemas
 - ☐ Em programação, devemos guardar as informações para nunca mais usar
 - ☐ Para resolver os problemas do sistema



2. Onde iremos guardar informações dentro do nosso código?

- ☐ Em uma caixa
- ☐ No sistema
- ☒ Nas variáveis.
- ☐ Na Tupperware.

3. As variáveis precisam de um nome?

- ☒ Sim
- ☐ Não

TIPOS DE DADOS

1. Com base no que estudamos sobre os tipos de variáveis, e na prática anterior, selecione a alternativa que CONTÉM ERRO na escrita do código:

- ☒ var nome = Hendy Almeida
- ☐ var Nome = "Hendy Almeida"
- ☐ var nome = "HeNdY AlMeida"
- ☐ var variavel = "texto qualquer"

2. Qual tipo de dados utilizamos para guardar uma lista?

- ☐ Variáveis
- ☒ Array
- ☐ String
- ☐ Boolean

USANDO VARIÁVEIS

1. Temos uma variável chamada apresenta o código correto para atribuir um novo valor a variável? da frutaDaEstacao que já foi declarada e armazena o nome de uma fruta. Qual a alternati

- ☐ var frutaDaEstacao = "Maçã"
- ☐ var = "Maçã"
- ☐ frutaDaEstacao "Maçã"
- ☒ frutaDaEstacao = "Maçã"



2. Qual comando correto para concatenar as variáveis nome e sobrenome e passar esse resultado para a variável nomeCompleto?

- ☒ `var nomeCompleto = nome + sobrenome`
- ☐ `var nomeCompleto = "nome + sobrenome"`
- ☐ `var nome = nomeCompleto`
- ☐ `var nomeCompleto = "Nome Completo"`

OPERADORES ARITMÉTICAS

1. Qual das operações abaixo exemplifica o uso do operador módulo?

- ☐ `a / b`
- ☐ `a * b`
- ☒ `a % b`
- ☐ `a - b`

2. Qual das alternativas abaixo exemplifica os operadores incremento e o decremento?

- ☐ `++ | *`
- ☒ `++ | --`
- ☐ `-- | +`
- ☐ `-- | /`

INTRODUÇÃO À CONDICIONAIS

1. Qual termo usamos em programação para representar verificações?

- ☐ Perguntas e respostas
- ☐ Variáveis
- ☒ Condicional
- ☐ Códigos

2. Qual dos exemplos abaixo podem representar uma condicional?

- ☐ Se eu quero eu simplesmente compro!
- ☒ Se tiver bolo, eu vou comer bolo com café, se não, irei comer bolacha com café!
- ☐ Você sabia que o céu é azul?
- ☐ Tomara que o caminho seja curto!



CONDICIONAL IF

1. Qual das opções abaixo apresenta a sintaxe correta do condicional if?

- ☐ if a == b { console.log('A condição é verdadeira') }
- ☒ if (a == b) { console.log('A condição é verdadeira') }
- ☐ if (a == b console.log('A condição é verdadeira'))
- ☐ if (console.log('A condição é verdadeira')) { a == b }

2. Em javascript, existe alguma forma além do if que podemos utilizar para criar uma condição?

- ☒ Sim
- ☐ Não

OPERADORES RELACIONAIS

1. Qual das alternativas abaixo melhor exemplifica os operadores relacionais de igualdade em relação à comparação por valor e tipo de dado?

- ☐ =
- ☐ ==
- ☒ ===

2. Qual das alternativas verifica se a variável a tem um valor a partir de 100?

- ☐ a == 100
- ☐ a => 100
- ☒ a >= 100
- ☐ a > 100



3. Considerando o código abaixo, qual condicional retorna o valor false?

```
var a = 10  
var b = 10
```

- ☐ a == b
- ☐ a === b
- ☒ a < b
- ☐ a <= b

CONDICIONAL IF ELSE

1. É obrigatório a utilização do else em cada if?

- ☐ Sim
- ☒ Não

2. Qual alternativa apresenta a sintaxe correta?

- ☒ if(condição) { //faz alguma coisa } else { // faz outra coisa }
- ☐ if { //faz alguma coisa } else (condição) { // faz outra coisa }
- ☐ if (condição) else { // faz alguma coisa }
- ☐ else (condição) { // faz outra coisa }

3. De acordo com o código abaixo, qual mensagem vamos imprimir?

```
var grauCelsius = 18;  
if (grauCelsius < 18) {  
    console.log("Está ficando frio. Não esqueça de levar um  
    agasalho!")  
} else {  
    console.log("O clima está agradável! Não esqueça o óculos de  
    sol :)")  
}
```

- ☐ "Está ficando frio. Não esqueça de levar um agasalho!"
- ☒ "O clima está agradável! Não esqueça o óculos de sol :)"
- ☐ Nenhuma das alternativas anteriores



OPERADORES LÓGICOS

1. Podemos fazer **apenas 3** condições usando o operador AND (&&).

☐ Verdadeiro

☒ Falso

2. Sobre o operador **OR** (||), qual dos códigos abaixo está **correto**?

☐ dia == 'quarta' || 'sábado'

☐ dia == 'quarta' && dia == 'sábado'

☒ dia == 'quarta' || dia == 'sábado'

☐ 'quarta' || 'sábado'

3. Um desenvolvedor precisa de ajuda para escolher o operador correto na sua condição. A ideia dele é exibir uma mensagem "Esse produto não está ativo" quando a variável **produtoAtivo** estiver com o valor **falso**.

```
var produtoAtivo = false
if() {
  console.log('Esse produto não está ativo')
}
```

Qual condição se encaixaria neste código:

☐ produtoAtivo != false

☐ produtoAtivo || false

☐ produtoAtivo && false

☒ !produtoAtivo

INTRODUÇÃO A LOOPS

1. Selecione **TODOS** os termos utilizados para definir um trecho de código que foi programado para se repetir.

☒ Loop

☐ Condicional

☒ Laço de repetição

☒ Ciclos



2. Qual a importância de ter uma condição de parada para o loop?

- ☐ Para criar variáveis
- ☒ Para não criar um laço de repetição infinito
- ☐ Para não ter retrabalho
- ☐ Não é importante/obrigatório

LOOP FOR

1. Quais são as informações que precisamos definir para o loop **for**?

- ☒ início, condição de parada e incremento
- ☐ início e incremento
- ☐ condição de parada e fim

INTRODUÇÃO A FUNÇÕES

1. Qual das seguintes afirmações podem ser usadas para descrever as **funções** em programação?

- ☐ É o nome de um tipo de variável
- ☐ É os recursos que tem dentro do controle da televisão
- ☒ É forma de agrupar instruções ao computador para utilizarmos em diversos momentos
- ☐ É a forma que falamos para definir os if e else

2. Em uma função só podemos utilizar console.log()!

- ☐ Verdadeiro
- ☒ Falso

DECLARANDO FUNÇÕES

1. Qual palavra reservada usamos para criar funções em Javascript?

- ☐ var
- ☐ funções
- ☐ criar
- ☒ function



2. O que está errado na criação da função abaixo?

```
function calcular{  
  console.log(A + B)  
}
```

- ☐ Não podemos usar console.log em funções
- ☒ Toda função deve ter os parênteses, que pode receber informações para a função
- ☐ Não podemos colocar nome em português
- ☐ A função não está errada!

3. Podemos usar **caracteres especiais** em **nome de funções**?

- ☐ Sim
- ☒ Não

FUNÇÕES PARAMETRIZADAS

1. Sobre a criação de uma função é **correto** afirmar que:

- ☐ Toda função obrigatoriamente deve a retornar algum valor.
- ☐ Toda função deve ser definida para receber pelo menos um parâmetro.
- ☒ Sempre que a linha de um return for executada a função será finalizada.
- ☐ O uso de parâmetro é obrigatório na função

2. O que há de errado na criação da função abaixo:

```
function maiorNumero() {  
  if(a > b) {  
    return a;  
  } else {  
    return b;  
  }  
}
```

- ☒ A função deveria ser definida para receber a e b como parâmetros, no entanto, não está definida para não receber nada.
- ☐ A função não está retornando o maior número, como o nome dela sugere.
- ☐ A função não chega até o final por que executa um return antes.



FUNÇÕES EM JS

1. Qual das alternativas apresenta somente funções nativas que podemos usar com **arrays**?

- ☐ pop(), Math.min(), toUppercase()
- ☒ push(), pop(), indexOf()
- ☐ round(), min(), max()
- ☐ array(), addArray(), removeArray()

2. Sabemos que a função Math.min() recebe parâmetros, compara e retorna o menor valor entre eles. Quantos parâmetros podemos passar para que ocorra a comparação?

- ☐ Somente 1 parâmetro por vez
- ☐ Somente 2 parâmetros por vez
- ☒ Podemos passar 2 ou mais parâmetros por vez
- ☐ Não precisamos passar parâmetros

3. Para identificar o índice de um valor no array, utilizamos a função indexOf(). Qual será o retorno do código abaixo?

```
var linguagensDeProgramacao = ["Javascript", "PHP", "Node.js", "Cobol", "C#",  
"Python"]
```

```
linguagensDeProgramacao.indexOf("Node.js")
```

- ☐ Retorna "Node.js"
- ☒ Retorna 2
- ☐ Retorna 3
- ☐ Retorna linguagensDeProgramacao