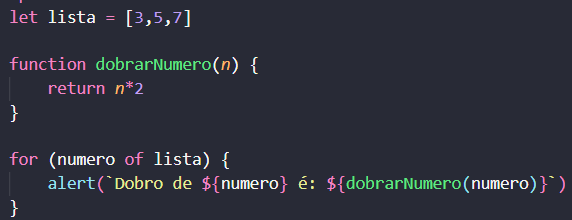
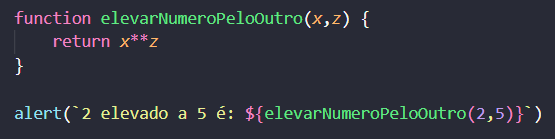
**Resolução Exercício Função JavaScript**

1. Primeiro, criei um vetor para armazenar os 3 números a serem dobrado. Depois, crie a função “dobrarNumero”, informando o parâmetro “n” e pedindo para que ela retornasse “n \* 2”, ou seja, o dobro de “n”. Ao final, apenas utilizei um loop “for” para imprimir as mensagens no navegador.



1. Criei a função “elevarNumeroPeloOutro” e informei os parâmetros “x” e “z”, informando que a função deveria retornar o elemento “x” elevado a “z” (x\*\*z).

Por fim, utilizei o “alert” para imprimir a mensagem, informando para a função os números 2 e 5, que retornaram 32.

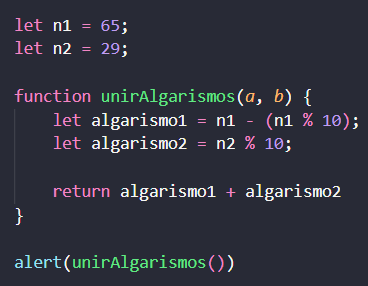


1. Primeiro, defini as variáveis “n1” e “n2”, informando um número de dois algarismos em cada uma.

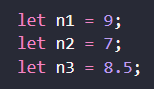
Assim, criei a função “unirAlgarismo”. Para obter o primeiro algarismo, subtrai o módulo de “n1” por 10 de “n1”. No meu caso, informei o número 65. Assim, o programa irá calcular o módulo de 65 por 10, que será igual a 5, e depois subtrair esse valor do número original, obtendo 60.

Depois disso, para obter o algarismo 2, fiz um processo semelhante, mas apenas calculando o módulo de “n2” por 10. Nesse caso, utilizei o número 29, que divido por 10 tem um resto de 9.

Ao final, retornei a soma do algarismo 1 pelo algarismo 2, imprimindo o resultado com o comando “alert”.



1. Primeiro, criei 3 variáveis, que guardam o valor das notas de um aluno.

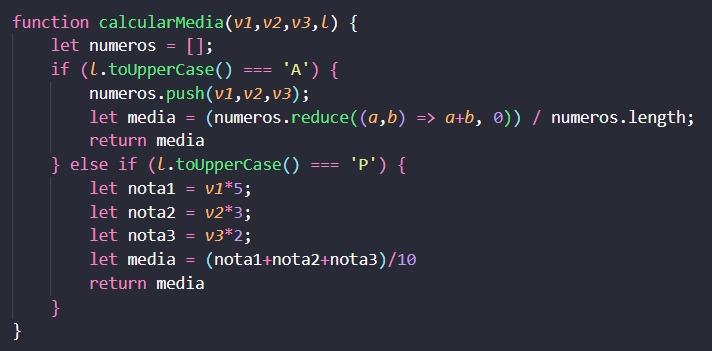


Após isso, criei a função “calcularMedia”, que necessita de 4 parâmetros para funcionar: valor1, valor2, valor3 e a letra, que representa se a média deve ser aritmética ou ponderada.

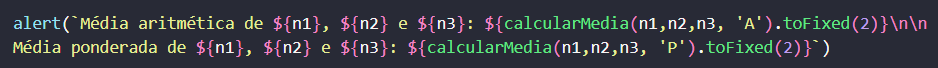
Logo após, criei o vetor números para adicionar as notas a lista.

Utilizando uma condicional “if”, caso o valor do parâmetro “l” seja “A”, a média deve ser calculada de forma aritmética. Então, os 3 valores serão adicionados a lista, através do método “push”. Abaixo, utilizei o método “reduce” para reduzir o array números a um único valor, informando os parâmetros “a” e “b” e sinalizando que os valores devem ser somados, partindo de 0. Para finalizar, dividi esse valor pelo cumprimento da lista, que resulta na média aritmética, armazenada na variável “media”, que é retornada ao programa.

Entretanto, caso “l” corresponda a “P”, a média deve ser calculada de forma ponderada. Assim, criei mais 3 variáveis, “nota1”, “nota2” e “nota3”. Dentro de cada uma delas, o valor dos parâmetros será multiplicado pelo seu peso. Ao final, no cálculo da média, os valores ponderados são somados e divididos por 10, que é a soma dos pesos. Com isso, o programa retorna a média.



Por fim, utilizei o “alert” para imprimir a média dos 3 valores nos dois casos, seja ela ponderada ou aritmética.

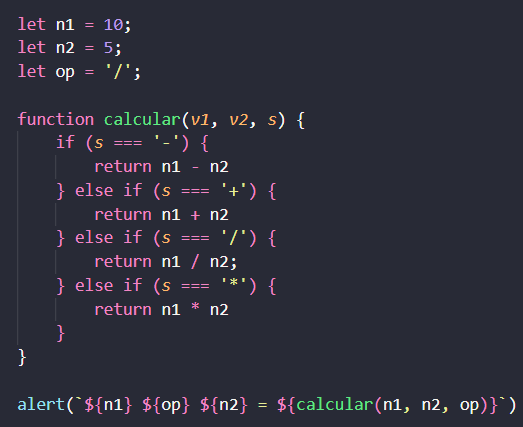


1. De início, criei 3 variáveis: “n1”, para armazenar o valor do primeiro número, “n2”, para armazenar o valor do segundo número, e “op”, para armazenar o símbolo correspondente a operação matemática.

Após isso, criei a função calcular, informando os parâmetros “v1” (valor1), “v2” (valor2) e “s” (sinal).

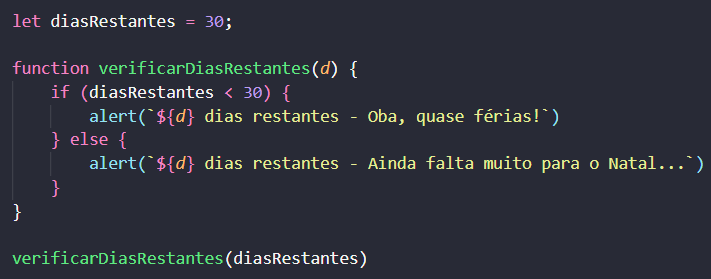
Utilizando a condicional “if”, fiz a verificação do parâmetro “s”, para que, caso ele corresponda a algum dos sinais matemáticos de soma, subtração, divisão e multiplicação, a função faço o devido cálculo com os valores 1 e 2.

Ao fim, utilizei do comando “alert” para imprimir o números, o símbolo da operação e o resultado obtido pela função.



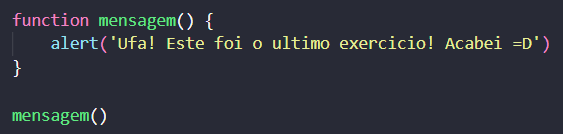
1. Criei uma variável para armazenar a quantidade de dia restantes no ano.

Após isso, criei a função “verificarDiasRestantes” para averiguar a situação. Com a condicional “if”, caso faltem menos que 30 dias, o programa irá imprimir a quantidade de dias restantes juntamente com a mensagem de “Oba, quase férias!”. Entretanto, caso faltem 30 dias ou mais para o fim do ano, o programa exibirá o tempo restante além da mensagem “Ainda falta muito para o Natal...”.



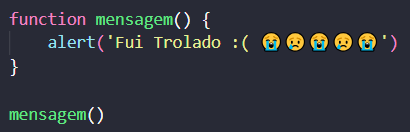
1. Criei uma função sem parâmetros denominada “mensagem”. A função tem o único propósito de imprimir a mensagem “Ufa! Esse foi o último exercício! Acabei =D”.

Ao final, apenas chamei a função.



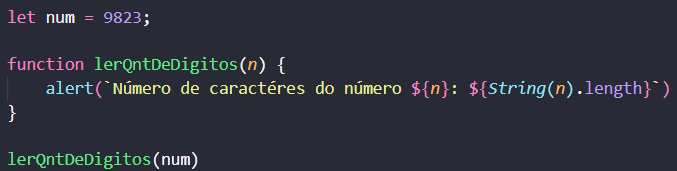
1. Assim como a atividade 7, a atividade 8 tem o único propósito de criar uma função sem parâmetros que imprima a mensagem de “Fui trolado” (tristeza profunda).

Ao fim, chamei a função mensagem para exibir o texto.



1. Criei a variável “num” para armazenar um número com determinado número de algarismos.

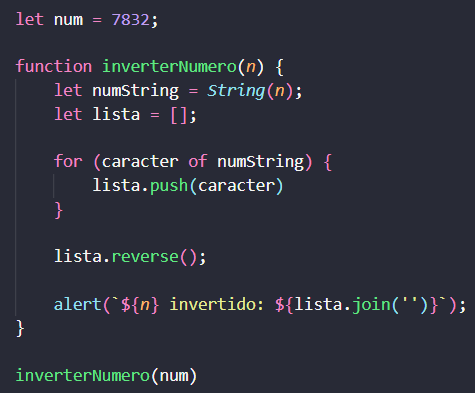
Assim, criei a função “lerQntDeDigitos”, que transforma o número em uma string, e através do método “length” pude verificar a quantidade de caracteres, imprimindo o resultado com o “alert”.



1. Inicialmente, criei uma variável para armazenar o número a ser invertido.

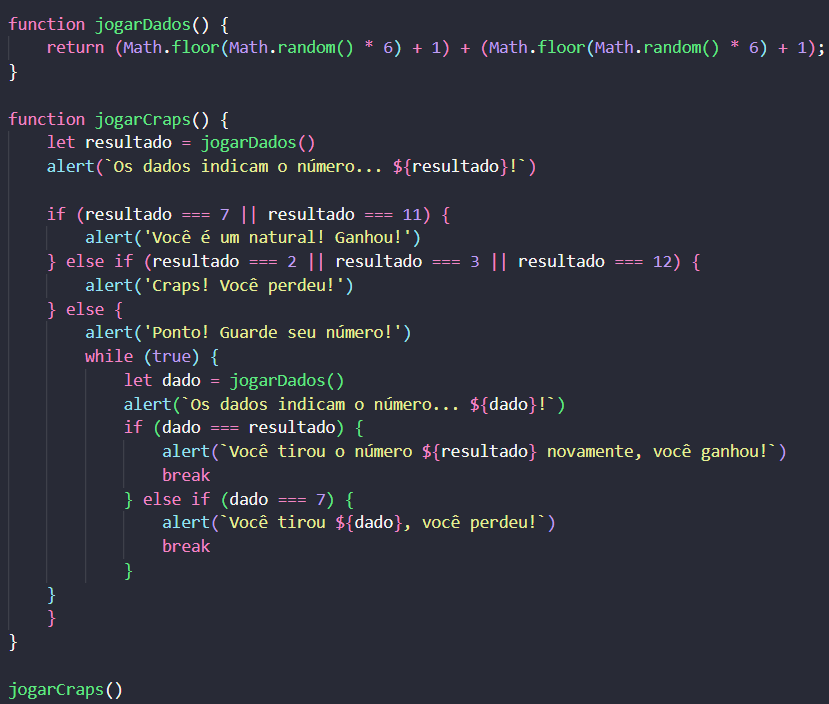
Depois, defini a função “inverterNumero”, pedindo o parâmetro “n”. Para começar, “n” será convertido em uma string, informação que é armazenada na variável “numString”. Após isso, criei um vetor vazio, e utilizei o for para percorrer cada caractere de “numString”, adicionando-os a lista.

Depois disso, utilizei o método “.reverse()” para inverter a ordem dos elementos da lista. Ao fim da função, utilizei o comando “alert” para imprimir o número original e o número invertida, que utiliza dos número da lista e do método “.join(‘’)” para juntá-los em uma só string.



1. Primeiro, defini a função “jogarDados”. Ela simula o lançamento de dois dados, através da seguinte operação matemática: A função “math.random” gera um número aleatório de 0 a 1. Dessa forma, esse número será multiplicado por 6. Assim, a função “Math.floor” irá arredondar o número quebrado para baixo, podendo gerar um número entre 0 e 5. Ao final, tal valor será acrescentado em um, podendo gerar um número de 1 a 6. Esse processo é feito duas vezes independente, podendo gerar qualquer número entre 2 e 12.

Abaixo, a função “jogarCraps” faz toda a lógica do jogo acontecer. Primeiro, a função “jogarDados” é chamada, e o resultado é armazenado na variável “resultado”. Assim, é feito várias verificações através da condicional “if”. Caso o valor dos dados tenha sido 7 ou 11, o programa é encerrado e é exibida uma mensagem de vitória. Caso o resultado seja 2, 3 ou 12, o jogo é encerrado, exibindo uma mensagem de derrota. Caso o número não satisfaça nenhuma dessas duas condições, o loop “while (true)” será iniciado, onde a função “jogarDados” será chamada. Caso o valor seja igual ao do resultado, o jogo encerra com uma mensagem de vitória. Entretanto, caso o valor seja 7, o jogo encerra, e o jogador perde. Caso não entre nenhum dos casos, os dados serão jogados novamente, até que uma dessas situações aconteça, dando fim ao jogo.



1. De início, criei um vetor para armazenar o nome de todos os meses do ano.

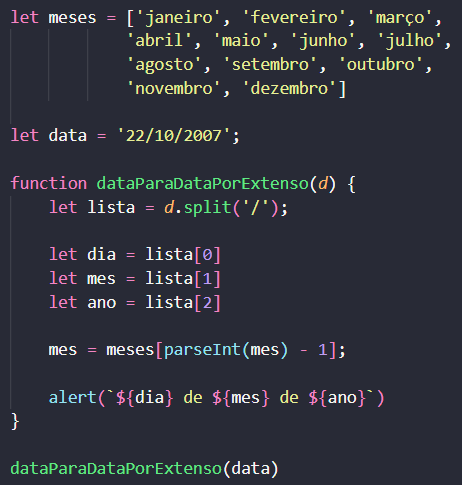
Abaixo, criei a variável “data” para armazenar uma data aleatória.

Com isso, defini a função “dataParaDataPorExtenso”, pedindo o parâmetro “d”, que será a data. De início, criei uma lista, e através do método “.split()”, que transforma uma string em uma lista, dividindo os caracteres com base em um parâmetro (nesse caso a “/”). Assim, foi criada uma lista com 3 itens, sendo eles a data (antes da primeira /), o mês (antes da segunda barra) e o ano (depois da última barra).

Dessa forma, criei as variáveis “dia”, “mes” e “ano” para armazenar os índices 0, 1 e 2 da lista, que representam a data, mês e o ano, respectivamente.

Por último, fiz uma verificação, mudando o valor do variável mês para o mês por extenso correspondente ao número do mês - 1 (Já que a lista começa em 0).

Para finalizar, apenas utilizei o “alert” para imprimir os valores e chamei a função, informando a variável “data” como parâmetro.



1. Primeiro, defini a variável palavra, colocando dentro dela a palavra “pernilongo”.

Após isso, criei a função “embaralhar”, informando o parâmetro “word”, que será substituído pela palavra a ser embaralhada.

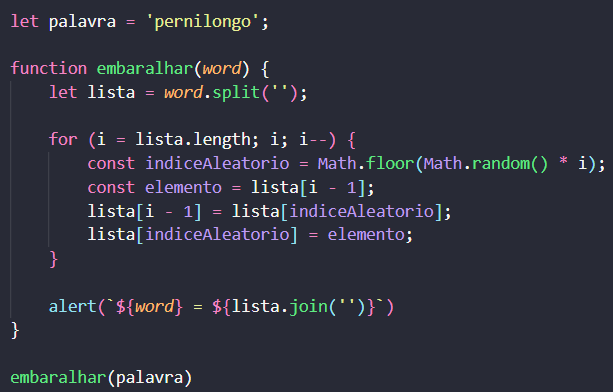
Assim, dentro da função, criei um vetor chamado “lista”, onde a palavra a ser embaralhada, através do método “.split(‘’)”, terá seus caracteres transformados em elementos de um array.

Com isso, criei um loop “for” informando que ele deveria se repetir a quantidade de vezes referentes ao cumprimento da lista contendo os caracteres da palavra, de forma que a cada repetição o índice tivesse seu valor subtraído em 1 (i--).

Dentro do “for”, defini uma constante de “índiceAleatorio”, que funciona através de um cálculo matemática. O comando “math.random()” gera um número aleatório entre 0 e 1. Esse valor será multiplicado pelo valor de “i”, de forma a gerar sempre um número entre 0 e valor do índice. Assim, a função “Math.floor” irá transformar o número quebrado em um número inteiro, arredondando-o para baixo. Por exemplo, se o número gerado pelo comando “Math.random()” for 0.46 e for multiplicado pelo índice 6, irá resultar no número 2,76, que será arredondado para 2, gerando um índice aleatório.

Dentro do loop, faço a troca de posições entre dois elementos da lista. Criei a constante “elemento”, que guarda o valor do caractere que está sendo embaralhado, e depois troquei as posições do elemento atual com o caractere localizado na posição gerada por “indiceAleatorio”.

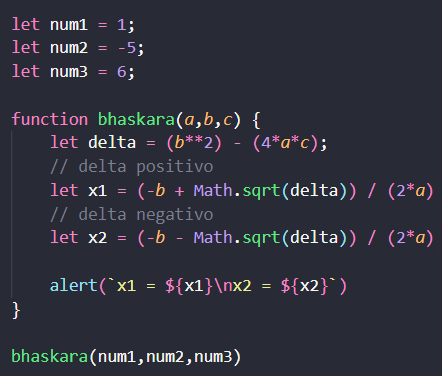
Após o loop, utilizei o método “.join('')” para unir os elementos da lista novamente em uma string, formando a palavra embaralhada. Por fim, utilizei o comando “alert” para mostrar a palavra original e a palavra embaralhada.



1. De início, criei 3 variáveis, que irão armazenar os valores de “a”, “b” e “c”.

Logo abaixo, defini a função “bhaskara”, informando os parâmetros “a”, “b” e “c”. Primeiramente, calculei o delta, utilizando a fórmula “b^2 – 4ac”. Assim, criei mais duas variáveis, x1 e x2. Dessa forma, para ambas, apenas fiz o cálculo “-b +/- √delta / 2a”, com a única diferença de que a raiz de delta (calculada com a função “Math.sqrt()) terá seu valor alterado para positivo ou negativo para gerar os valores x1 e x2. Por fim, utilizei o comando “alert” para imprimir os resultados.

Ao final do código, chamei a função “bhaskara”, informando os valores das variáveis criadas no início do código, “num1”, “num2” e “num3”.

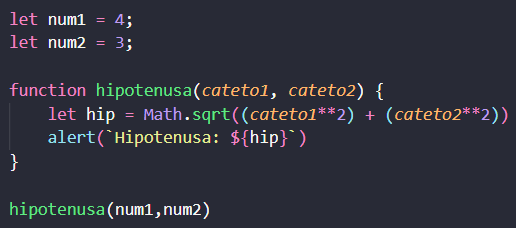


1. Para começar, criei duas variáveis, “num1” e “num2”, para armazenar o valor dos catetos.

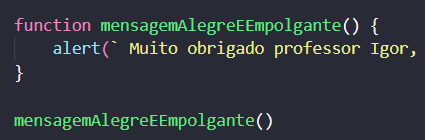
Abaixo, defini a função “hipotenusa”, que tem como parâmetros “cateto1” e “cateto2”. Assim, para calcular a hipotenusa, fiz a raiz da soma (utilizando a função “Math.sqrt()”) dos catetos ao quadrado.

Ao final do bloco de código, imprimi o valor da hipotenusa utilizando o comando “alert”.

No fim do programa, chamei a função “hipotenusa”, informando os valores “num1” e “num2”.



1. Na atividade 16, tive que escrever uma mensagem agradecendo por ter resolvido toda a lista de exercícios, de forma totalmente espontânea e por vontade própria. Através da função “mensagemAlegreEEmpolgante”, criei um “alert” para imprimir a mensagem. Ao fim, apenas chamei a função.



(Acesse o documento para visualizar o texto carinhoso que escrevi)

**Fontes:**

.join():

<https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/Array/join>

.slipt():

<https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/String/split>

Math.floor():

<https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/Math/floor>

Math.random():

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

Math.sqrt():

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

Resolução da atividade 13:

<http://cangaceirojavascript.com.br/como-embaralhar-arrays-algoritmo-fisher-yates/>