

```
function dobroNumeros() {
  let numero1 = parseFloat(prompt("Digite aqui o primeiro número:"));
  let numero2 = parseFloat(prompt("Digite aqui o segundo número:"));
  let numero3 = parseFloat(prompt("Digite aqui o terceiro número:"));

  let dobro1 = numero1 * 2;
  let dobro2 = numero2 * 2;
  let dobro3 = numero3 * 2;

  alert(`O dobro de ${numero1} é ${dobro1}, o dobro de ${numero2} é ${dobro2}, e o dobro de ${numero3} é ${dobro3}`);
}

dobroNumeros();
```

1 – Iniciei as atividades com esse código para realizar o dobro de números. Eu poderia utilizar uma estrutura como o for para o prompt, mas decidi realizar de uma maneira menos otimizada mesmo. No código pedi o prompt de 3 números e logo após realizei a multiplicação em dobro dos mesmos, logo exibi o alert.

```
function elevadoNumero() {
  let numero1 = parseFloat(prompt("Digite aqui o primeiro número:"));
  let numero2 = parseFloat(prompt("Digite aqui o segundo número:"));

  let elevado = numero1 ** numero2

  alert(elevado)
}

elevadoNumero();
```

2 – Aqui desenvolvi uma function para elevar um número com base no próximo informado, logo após exibir o resultado

```
function formarNumero(n1, n2) {
  const primeiroDigitoN1 = Math.floor(n1 / 10);
  const segundoDigitoN2 = n2 % 10;

  const n3 = parseInt(`${primeiroDigitoN1}${segundoDigitoN2}`);
  return n3;
}

const n1 = 23;
const n2 = 89;

const n3 = formarNumero(n1, n2);
alert(`O número formado é: ${n3}`);
```

3 – Neste exercício fiz uma função para pegar o primeiro digito do número inicialmente

informado, dividindo o valor por 10 e chamando o `math.floor` para pegar o número inteiro e não quebrado (caso ocorresse uma quebra no momento da divisão). No segundo número fiz pelo módulo de 10 para conseguir obter o último número do elemento (no caso, a casa da unidade no número). Após, juntei os dois números e fiz um `return` para exibição. Ao dei um `alert`.

```
function calcularMedia(nota1, nota2, nota3, tipoMedia) {
    let media;

    if (tipoMedia === 'A') {
        media = (nota1 + nota2 + nota3) / 3;
    } else if (tipoMedia === 'P') {
        media = (nota1 * 5 + nota2 * 3 + nota3 * 2) / (5 + 3 + 2);
    } else {
        return 'Tipo de média inválido.';
    }

    return media.toFixed(2);
}

const nota1 = 8;
const nota2 = 9;
const nota3 = 7;
const tipoMedia = 'P';

const resultado = calcularMedia(nota1, nota2, nota3, tipoMedia);
alert('A média calculada é: ' + resultado);
```

4 – Inicialmente criei uma função para calcular a média das notas, tanto ponderadas quanto aritméticas. Declarei a variável `media` e comecei a codar as condições. Declarei que se caso o tipo e média informada fosse "A" seria aritmética e se fosse "P" seria ponderada. Para a média ponderada fiz o valor vezes o peso de cada, ao fim dividindo pela soma de peso dos valores. Fiz um `else` que iria retornar "tipo de média inválida" caso o que foi inserido não se adequasse a nenhum dos valores anteriores. Ao fim fiz um `return media.toFixed(2)` para caso o valor desse mais de 2 casas. Ao fim fiz um `alert`.

```

var simbolo = prompt("Insira a operação que você deseja. Subtrair, somar, dividir ou multiplicar(+, -, *, /): ")

function operações(num1, num2) {
  if(simbolo = "+") {
    soma = num1 + num2;
    alert(soma)
  } else if (simbolo = "-") {
    subtracao = num1 - num2;
    alert(subtracao)
  } else if (simbolo = "*") {
    multipli = num1 * num2;
    alert(multipli)
  } else if (simbolo = "/") {
    divisao = num1 / num2;
    alert(divisao)
  }
}

operações(4,5);

```

5 – Coloquei um prompt para o usuário inserir qual seria a operação realizada. A seguir fiz um function para calcular qualquer possibilidade dentre as operações, e fiz respectivamente cada operação entre o "num1" e "num2" que se resultariam em alguma operação que foi escolhida dos símbolos anteriormente inseridos. Ao fim, eu chamo a variável.

```

function verificarDiasRestantes(diasRestantes) {
  if (diasRestantes > 30) {
    alert("Ainda falta muito para o natal.");
  } else {
    alert("Oba, quase ferias!");
  }
}

verificarDiasRestantes(100);

```

6 – Fiz esta função que identifica se os números restantes são superiores ou inferiores a 30, se forem superiores = falta muito para o natal. Se forem inferiores = Oba, quase férias!

```

function imprime_ultima_frase() {
  alert("Ufa! Este foi o ultimo exercicio! Acabei =D")
}

imprime_ultima_frase();

```

7 – Neste exercício realizei uma brincadeirinha que o Igor propôs e digitei que este seria o último exercício. 🍷 🍷

```
function fui_trolado() {  
  alert("Fui Trolado :(")  
}  
  
fui_trolado()
```

8 – Este exercício segue a mesma linha da brincadeira, colocando que as atividades não acabaram e eu apenas fui trolado. 🤖

```
function contarDigitos(numero) {  
  const numStr = numero.toString();  
  return numStr.length;  
}  
  
alert(contarDigitos(64645654456))
```

9 – Criei esta function para contar os dígitos de um número. Fiz uma constante para transformar os números coletados em string, sendo assim capaz de contar cada valor como um caractere, logo após fiz um retorno dos valores e dei um .length para contar quantos caracteres tinham no total. Ao fim dei um alert para exibir um número que eu coloquei aleatoriamente.

```
function reversoNumero(numero) {  
  const numeroInvertido = numero.toString().split('').reverse().join('');  
  return parseInt(numeroInvertido);  
}  
  
const numeroOriginal = 127;  
const numeroReverso = reversoNumero(numeroOriginal);  
alert(numeroReverso);
```

10 – Fiz uma function para reverter o número. Colocar ele invertido (ao inverso). Para isso fiz um .toString para transformar o número em string, após fiz um split para separar os caracteres em um array, fiz um .reverse para reverter os valores, e um join para fazer os dados se tornarem uma string novamente. Inseri os valores e ao fim dei um alert.

```

function lancarDados() {
    return Math.floor(Math.random() * 6) + 1;
}

function jogarCraps() {
    const dado1 = lancarDados();
    const dado2 = lancarDados();
    const soma = dado1 + dado2;
    alert(`Você tirou: ${dado1} + ${dado2} = ${soma}`);

    if (soma === 7 || soma === 11) {
        alert("Parabéns! Você tirou um Natural e ganhou!");
    } else if (soma === 2 || soma === 3 || soma === 12) {
        alert("Craps! Você perdeu.");
    } else {
        const ponto = soma;
        alert(`Seu ponto é ${ponto}. Continue jogando...`);

        while (true) {
            const dado1 = lancarDados();
            const dado2 = lancarDados();
            const soma = dado1 + dado2;
            alert(`Você tirou: ${dado1} + ${dado2} = ${soma}`);

            if (soma === ponto) {
                alert("Parabéns! Você ganhou tirando seu ponto novamente!");
                break;
            } else if (soma === 7) {
                alert("Você tirou um 7 antes de tirar seu ponto. Você perdeu.");
                break;
            }
        }
    }
}

jogarCraps();

```

11 - Fiz uma function para jogar dados no Craps. Inicialmente, a função lancarDados gera um número aleatório entre 1 e 6 usando Math.random, multiplicado por 6 e arredondado para o valor inteiro inferior com Math.floor. Depois, na função jogarCraps, lancei dois dados e somei os valores. Mostrei o resultado com um alert. Caso a soma seja 7 ou 11, mostro uma mensagem de vitória ('Natural'). Se for 2, 3 ou 12, a função exibe uma mensagem de derrota ('Craps'). Caso contrário, defino o 'ponto' e continuo jogando até o jogador tirar o valor do ponto novamente ou perder ao tirar um 7, sempre exibindo o valor dos dados com alert.

```
function converterData(data) {
  const mesesPorExtenso = [
    "janeiro", "fevereiro", "março", "abril", "maio", "junho",
    "julho", "agosto", "setembro", "outubro", "novembro", "dezembro"
  ];

  const [dia, mes, ano] = data.split("/");

  const mesPorExtenso = mesesPorExtenso[parseInt(mes) - 1];

  return `${parseInt(dia)} de ${mesPorExtenso} de ${ano}`;
}

const dataFormatada = converterData("27/09/2024");
alert(dataFormatada);
```

12 – Iniciei a function, logo após criei uma constante para exibir os meses em extenso. Criei outra constante para organizar de maneira separada o dia, mes e ano, com o split, deixando-os separados na string. Converto o número do mês com parseInt e obtenho o nome correspondente do array. Finalmente, monto a string no formato 'dia de mês de ano' e retornei o valor. Ao fim dei o alert

```
let frase = 'orangotango careca doido';

function misturar(texto) {
  let caracteres = texto.split('');

  for (let i = caracteres.length; i; i--) {
    const indiceAleatorio = Math.floor(Math.random() * i);
    const elemento = caracteres[i - 1];
    caracteres[i - 1] = caracteres[indiceAleatorio];
    caracteres[indiceAleatorio] = elemento;
  }

  alert(`${texto} = ${caracteres.join('')}`)
}

misturar(frase);
```

13 - Nesta atividade, criei uma função chamada misturar que embaralha os caracteres de

uma string. Primeiro, usei o split para transformar a string em um array de caracteres. Depois, fiz um loop para embaralhar esses caracteres, escolhendo índices aleatórios e trocando os elementos de lugar. Ao final, a função retorna a string original e a string embaralhada. Ao fim, dei alert para exibir o resultado.

```
function bhaskara(a, b, c) {  
  if (a === 0) {  
    return "O coeficiente 'a' não pode ser 0";  
  }  
  
  const delta = (b * b) - (4 * a * c);  
  
  if (delta < 0) {  
    return "Não há raízes reais";  
  }  
  
  const raiz1 = (-b + Math.sqrt(delta)) / (2 * a);  
  const raiz2 = (-b - Math.sqrt(delta)) / (2 * a);  
  
  return [raiz1, raiz2];  
}  
  
const a = 1, b = -3, c = 2;  
alert(bhaskara(a, b, c));
```

14 - Aqui, implementei a fórmula de Bhaskara para calcular as raízes de uma equação quadrática. A função recebe três coeficientes: a, b e c. Primeiro, verifico se o coeficiente 'a' é diferente de zero, já que uma equação quadrática exige isso. Em seguida, calculei o delta e determinei as raízes reais usando a fórmula padrão. Se o delta for negativo, a função retorna que não existem raízes reais. Ao fim, dei alert para mostrar o resultado das raízes calculadas.

```
function pitagoras(c1,c2) {
  let hipotenusa = Math.sqrt(c1*c1 + c2*c2)
  alert(`O resultado da hipotenusa é ${hipotenusa}`)
}

pitagoras(4,5);
```

15 - Nesta questão, fiz uma função para calcular a hipotenusa de um triângulo retângulo com base nos comprimentos dos dois catetos. A função recebe como parâmetros c1 e c2, que representam os comprimentos dos catetos. Utilizei a fórmula do Teorema de Pitágoras ($a^2 + b^2 = c^2$) para calcular a hipotenusa. Ao fim, dei alert para exibir o resultado da hipotenusa calculada.

```
function aprendi() {
  alert("Nesta atividade aprendi bastante sobre a linguagem js(javascript). Aprendi conceitos como Math.sqrt")
}

aprendi()
```

16 – Dei um alert para representar que aprendi bastante na atividade, representei citando o conceito do math.sqrt, que eu não conhecia.