

# Atividade JavaScript - Lista 2

Os exercícios a seguir foram designados à nossa turma, com propósito de incentivar o treinamento da linguagem de programação JavaScript, utilizando de seus conceitos básicos para resolver situações problemas, com questões que variavam o nível de dificuldade, mas não requerendo um alto nível de proficiência.

1.

```
<script>
    var lista = [1, 2, 3, 4, 5]
    alert(lista)
</script>
```

O primeiro exercício vinha com a proposta de introduzir os princípios básicos que seriam utilizados em todos os exercícios seguintes. Basicamente, tudo o que fora requisitado era a criação de um vetor, vulgo uma lista, onde 5 números inteiros seriam introduzidos nela, como pode ser observado na primeira linha, e depois mostrá-la, como pode ser visto no 'alert'.

2.

```
<script>
    var lista = [1, 2, 3, 4, 5, 6, 7, 8, 9, 0]
    lista.reverse()
    alert(lista)
</script>
```

O segundo exercício também possuía uma proposta simples e introdutória, onde uma lista contendo 10 números inteiros pré-definidos deveria ser criada, como visto na primeira linha, depois mostrados na ordem inversa, onde a segunda linha tem o papel de inverter a ordem da lista, e a última de mostrar a lista.

3.

```
<script>
var quatro = 4
var um_q_vira_quatro = 1
var lista = []
var sum = 0
while (quatro != 0){
    var nota = parseFloat(prompt('Insira a ' + um_q_vira_quatro + 'ª nota: '))
    lista.push(nota)
    sum += nota
    um_q_vira_quatro += 1
    quatro -= 1
}
alert('As notas são: ' + lista + '\nA média é: ' + sum/um_q_vira_quatro)
</script>
```

O terceiro exercício nos introduz a utilização do recurso 'prompt', que será fundamental em grande maioria dos próximos exercícios. Esse exercício, pelo fato de ainda ser de nível introdutório, possuía um enunciado simples e intuitivo, o que se manteve por enquanto. As primeiras linhas do código criam variáveis, que serão explicadas ao decorrer do código. Ao iniciar, vemos a criação de um loop em while, que, de forma literal, significa 'enquanto a variável nomeada quatro for diferente de zero:', e essa variável possui o valor de 4. Dentro do loop, o prompt é usado para perguntar a x nota, onde x é representado pela variável um\_q\_vira\_quatro, sendo originalmente composta pelo número 1, mas a cada repetição do loop while, 1 é somado a essa variável. Para calcular a média das 4 notas que serão requisitadas, utilizei da variável sum, onde ela armazenará a soma de todas as notas. Quando o prompt for repetido 4 vezes, o alert será executado, informando a média das somas, sendo ela a soma das 4 notas dividido por 4.

4.

```
<script>
var lista = ['b', 'c', 'd', 'f', 'g', 'h', 'j', 'k', 'l', 'm', 'n', 'p', 'q', 'r', 's', 't', 'v', 'w', 'x', 'y', 'z']
var letras = ['e', 'f', 'i', 'd', 'p', 'h', 'q', 'a', 'l', 'c']
var consoante = []
var count = 0
for (char in letras){
  for (cons in lista){
    if (letras[char] == lista[cons]){
      count += 1
      consoante.push(letras[char])
    }
  }
}
alert('Na lista haviam '+ count + ' consoantes, sendo elas: ' + consoante)
</script>
```

O quarto exercício fora o primeiro em que eu utilizei da instrução 'for', na qual eu consigo navegar pelos itens de um vetor, utilizando como base o valor da posição de determinado item dentro da lista, iniciando com 0. As primeiras linhas criam duas listas, uma delas contendo todas as consoantes do alfabeto, e a outra com 10 letras aleatórias. Existe também uma terceira lista, que será utilizada para armazenar somente as consoantes da segunda lista. O count será um contador, para verificar quantas consoantes houverão na lista, inicialmente vazia. Lendo o código, podemos resumir ele em: para as consoantes dentro da lista, iremos compará-las com as letras dentro da lista 'letras', e caso alguma das letras seja uma consoante, ela irá ser adicionada na lista vazia, e o contador aumentará em 1. Isso será feito analisando letra por letra na lista 'letras', e comparando todas elas com a letra 'b', depois com a letra 'c', e assim até acabar a lista 'lista'. Por fim, é mostrado o número de consoantes que estavam na lista 'letras', e quais são elas.

5.

```

<script>
  var tudo = []
  var par = []
  var impar = []
  var count = 20
  while (count != 0){
    var num = parseFloat(prompt('Insira um número (números restantes: '+ count + '): '))
    tudo.push(num)
    if (num % 2 == 0){
      par.push(num)
    } else{
      impar.push(num)
    }
    count -= 1
  }
  alert(tudo + '\n' + par + '\n' + impar)
</script>

```

O quinto exercício faz o uso do prompt para que o usuário insira 20 números, decimais ou inteiros, inserindo-os em uma lista única, e após isso, separando-os em pares e ímpares, sendo uma lista para cada um desses cenários. As 3 listas necessárias foram criadas nas primeiras linhas do script, e a variável count também é definida logo abaixo, sendo que ela vale 20, esse o total de números necessários para encerrar o loop while. Ambos os números são inseridos na lista 'tudo', e depois um if é acionado, onde se o número digitado dividido por 2, tiver um resto igual a zero, o número é adicionado na lista de pares, se não, na lista de ímpares. A variável count é reduzida em 1 a cada final do loop, e assim que ela chega em 0, o loop se encerra. Por fim, as 3 listas são apresentadas por meio do alert, tabuladas por meio de um \n, que força a quebra de linhas.

6.

```

<script>
  var media = []
  var show = []
  var count = 1
  var count2 = 1
  while (true){
    var nota = parseFloat(prompt('Insira a ' + count2 + 'ª nota do ' + count + 'º aluno:'))
    count2 += 1
    var nota2 = parseFloat(prompt('Insira a ' + count2 + 'ª nota do ' + count + 'º aluno:'))
    count2 += 1
    var nota3 = parseFloat(prompt('Insira a ' + count2 + 'ª nota do ' + count + 'º aluno:'))
    count2 += 1
    var nota4 = parseFloat(prompt('Insira a ' + count2 + 'ª nota do ' + count + 'º aluno:'))
    count2 += 1
    var soma = (nota + nota2 + nota3 + nota4)/4
    media.push(soma)
    count2 -= 4
    if (count == 10){
      break
    }
    count += 1
  }
  for (num in media){
    if (media[num] >= 7.0){
      show.push(media[num])
    }
  }
  var show2 = show.length
  alert('O número de alunos com a média maior ou igual à 7.0 é: ' + show2)
</script>

```

O sexto exercício tem como objetivo calcular a média das notas de 10 alunos e contar quantos deles possuem uma média maior ou igual a 7.0. Este exercício introduz o conceito de manipulação de múltiplos dados em um vetor e o uso de condicionais para realizar contagens baseadas em critérios específicos. No início do código, são criadas duas listas: uma chamada *media*, para armazenar as médias das notas dos alunos, e outra chamada *show*, para armazenar apenas as médias que são maiores ou iguais a 7.0. Dois contadores são inicializados: *count* para numerar os alunos e *count2* para rastrear as notas de cada aluno. O código utiliza um loop *while* para coletar as notas de cada aluno. Dentro desse loop, o *prompt* é usado quatro vezes para obter as quatro notas de cada aluno. Essas notas são armazenadas em variáveis individuais (*nota*, *nota2*, *nota3*, *nota4*). Em seguida, calcula-se a média dessas quatro notas e a média é adicionada ao vetor *media*. O contador *count* é incrementado a cada iteração para garantir que o loop continue até que todos os 10 alunos tenham suas médias calculadas. O loop é encerrado quando *count* atinge 10. Após o cálculo das médias, um segundo loop *for* percorre o vetor *media* para verificar quais médias são maiores ou iguais a 7.0. Essas médias são armazenadas no vetor *show*. Finalmente, o comprimento do vetor *show* (que representa o número de alunos com média maior ou igual a 7.0) é calculado e exibido ao usuário por meio de um *alert*.

7.

```

<script>
  var num = [1, 2, 3, 4, 5]
  var soma = 0
  var mult = 1
  for (i in num){
    soma += num[i]
    mult *= num[i]
  }
  alert('Números: ' + num + '\nSoma: ' + soma + '\nMultiplicação: ' + mult)
</script>

```

O sétimo exercício tem a proposta de trabalhar com operações básicas em vetores e exibir os resultados. Neste exercício, o objetivo é ler um vetor contendo 5 números inteiros, calcular a soma e a multiplicação desses números e, em seguida, mostrar os resultados. Primeiramente, o vetor num é criado contendo 5 números inteiros predefinidos: [1, 2, 3, 4, 5]. A seguir, duas variáveis são inicializadas: soma, que será usada para armazenar a soma dos números, e mult, que armazenará o produto dos números. A variável soma é iniciada com o valor 0 para que possa acumular a soma dos números. A variável mult é iniciada com 1 porque o produto de uma série de números deve ser inicializado com 1 para não afetar o resultado final da multiplicação. Em seguida, um loop for é utilizado para iterar sobre cada elemento do vetor num. Durante cada iteração do loop, o valor atual do número é adicionado à variável soma, acumulando a soma total dos números. Simultaneamente, o valor do número é multiplicado à variável mult, acumulando o produto total dos números. Depois que o loop termina, um alert é usado para exibir os resultados. O alert mostra o vetor num, a soma dos números e o resultado da multiplicação dos números. A mensagem exibida pelo alert é formatada para apresentar claramente todos esses valores.

8.

```

<script>
  var idade = []
  var altura = []
  var count = 1
  while (true){
    var idade_ = parseInt(prompt('Insira a idade, pessoa ' + count + ': '))
    idade.push(idade_)
    var altura_ = parseFloat(prompt('Insira a altura, pessoa ' + count + ': '))
    altura.push(altura_)
    if (count == 5){
      break
    }
    count += 1
  }
  altura.reverse()
  idade.reverse()
  alert(altura + '\n'+idade)
</script>

```

O oitavo exercício tem como objetivo coletar dados de 5 pessoas, especificamente suas idades e alturas, armazenar essas informações em vetores distintos e, posteriormente, exibir esses dados na ordem inversa da ordem em que foram inseridos. Primeiramente, dois vetores são criados: idade para armazenar as idades das pessoas e altura para armazenar suas

alturas. Ambos os vetores são inicialmente vazios. Em seguida, um contador count é inicializado com o valor 1 para rastrear o número da pessoa cuja informação está sendo inserida. Um loop while é utilizado para coletar os dados de cada pessoa. Dentro do loop, o código solicita ao usuário a idade e a altura da pessoa atual usando prompt. A idade é convertida para um número inteiro com parseInt e a altura é convertida para um número de ponto flutuante com parseFloat. Esses valores são então adicionados aos vetores idade e altura, respectivamente. O loop continua a coletar informações até que o contador count atinja o valor 5, indicando que as informações de 5 pessoas foram inseridas. Após a coleta dos dados, o método reverse é aplicado aos vetores altura e idade para inverter a ordem dos elementos, de modo que os dados sejam exibidos na ordem inversa da ordem em que foram inseridos. Finalmente, um alert é utilizado para mostrar os vetores altura e idade com seus valores na ordem inversa. A mensagem exibida pelo alert mostra primeiro as alturas e, em seguida, as idades.

9.

```
<script>
  var A = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
  var B = []
  var show = 0
  for (i in A){
    var sum = A[i]*A[i]
    B.push(sum)
  }
  for (x in B){
    show += B[x]
  }
  alert(show)
</script>
```

O nono exercício tem como objetivo calcular a soma dos quadrados dos elementos de um vetor contendo 10 números inteiros e mostrar esse resultado. Inicialmente, é criado um vetor chamado A contendo 10 números inteiros predefinidos: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]. Em seguida, dois outros vetores são definidos: B, que será utilizado para armazenar os quadrados dos elementos de A, e show, que é uma variável usada para acumular a soma total dos quadrados. Um loop for é utilizado para percorrer cada elemento do vetor A. Dentro do loop, o quadrado de cada elemento é calculado multiplicando o valor do elemento por ele mesmo ( $A[i] * A[i]$ ). Esse valor é então adicionado ao vetor B. Após calcular e armazenar todos os quadrados no vetor B, um segundo loop for é utilizado para somar todos os elementos de B. A soma total é acumulada na variável show. Por fim, um alert é utilizado para mostrar o resultado da soma dos quadrados dos elementos do vetor A.

10.

```

<script>
    var lista1 = [1, 3, 5, 7, 9, 11, 13, 15, 17, 19]
    var lista2 = [2, 4, 6, 8, 10, 12, 14, 16, 18, 20]
    var lista3 = []
    for (num in lista1){
        lista3.push(lista1[num])
        lista3.push(lista2[num])
    }
    alert(lista3)
</script>

```

O décimo exercício tem como objetivo criar um novo vetor que combina dois vetores existentes de 10 elementos cada, intercalando seus valores. Primeiramente, dois vetores são definidos: lista1 e lista2. O vetor lista1 contém os números ímpares de 1 a 19, e o vetor lista2 contém os números pares de 2 a 20. Ambos os vetores têm 10 elementos. Em seguida, um terceiro vetor chamado lista3 é criado para armazenar os elementos intercalados de lista1 e lista2. Esse vetor começa vazio e será preenchido com os valores dos dois vetores originais. Um loop for é utilizado para percorrer os índices dos vetores lista1 e lista2. Em cada iteração do loop, o elemento atual de lista1 é adicionado ao vetor lista3, seguido pelo elemento correspondente de lista2. Esse processo garante que os elementos de lista1 e lista2 sejam inseridos alternadamente em lista3. Depois que o loop termina, o vetor lista3 contém todos os 20 elementos, intercalando os valores de lista1 e lista2 conforme o solicitado. Por fim, um alert é usado para exibir o vetor lista3, mostrando os elementos intercalados.

11.

```

<script>
    var lista1 = [1, 3, 5, 7, 9, 11, 13, 15, 17, 19]
    var lista2 = [2, 4, 6, 8, 10, 12, 14, 16, 18, 20]
    var lista3 = [10, 20, 30, 40, 50, 60, 70, 80, 90, 10]
    var lista4 = []
    for (num in lista1){
        lista4.push(lista1[num])
        lista4.push(lista2[num])
        lista4.push(lista3[num])
    }
    alert(lista4)
</script>

```

O décimo primeiro exercício amplia o conceito do exercício anterior, intercalando três vetores em vez de dois. O objetivo é criar um novo vetor que combine elementos de três vetores existentes de 10 elementos cada, intercalando seus valores. Primeiramente, três vetores são definidos: lista1, lista2 e lista3. O vetor lista1 contém números ímpares de 1 a 19, o vetor lista2 contém números pares de 2 a 20 e o vetor lista3 contém uma sequência de múltiplos de 10. Todos os vetores possuem 10 elementos. Em seguida, um quarto vetor

chamado lista4 é criado para armazenar os elementos intercalados dos três vetores. Este vetor começa vazio e será preenchido com os valores de lista1, lista2 e lista3. Um loop for é utilizado para percorrer os índices dos vetores lista1, lista2 e lista3. Durante cada iteração do loop, o elemento correspondente de lista1 é adicionado ao vetor lista4, seguido pelo elemento correspondente de lista2, e depois pelo elemento correspondente de lista3. Esse processo garante que os elementos dos três vetores sejam inseridos alternadamente em lista4. Após a conclusão do loop, o vetor lista4 contém todos os 30 elementos, intercalando os valores dos três vetores originais. Por fim, um alert é utilizado para exibir o vetor lista4, mostrando os elementos intercalados de lista1, lista2 e lista3.

12.

```
<script>
var altura = []
var altura13 = []
var count = 1
var soma = 0
var count2 = 0
while (true){
    var idade = parseInt(prompt('Insira a idade, aluno ' + count + ': '))
    if (idade > 13){
        var altura_ = parseFloat(prompt('Insira a altura, aluno ' + count + ': '))
        altura13.push(altura_)
        altura.push(altura_)
    } else {
        var altura__ = parseFloat(prompt('Insira a altura, aluno ' + count + ': '))
        altura.push(altura__)
    }
    if (count == 30){
        break
    }
    count += 1
}
for (i in altura){
    soma += altura[i]
}
soma /= count
for (alt in altura13){
    if (altura13[alt] < soma){
        count2 += 1
    }
}
alert('O número de alunos com mais de 13 anos que possuem altura inferior à média de altura é: ' + count2)
</script>
```

O décimo segundo exercício tem como objetivo determinar quantos alunos com mais de 13 anos têm uma altura inferior à média de altura de todos os 30 alunos. O código inicia com a criação de dois vetores: altura para armazenar a altura de todos os alunos e altura13 para armazenar apenas a altura dos alunos com mais de 13 anos. Além disso, são inicializadas as variáveis count e count2. A variável count serve como contador para os alunos e count2 para contar o número de alunos com mais de 13 anos cuja altura é inferior à média. Um loop while é utilizado para coletar a idade e a altura de cada um dos 30 alunos. Dentro do loop, se a idade do aluno for maior que 13, sua altura é armazenada tanto no vetor altura13 quanto no vetor altura. Caso contrário, a altura é apenas armazenada no vetor altura. A cada iteração, o contador count é incrementado e o loop termina quando count atinge 30. Após coletar todos os dados, um segundo loop for é utilizado para calcular a soma de todas as alturas armazenadas no vetor altura. A média de altura é então calculada dividindo a soma total pelo número de alunos (count). Em seguida, outro loop for é usado para verificar quantos alunos com mais de 13 anos (ou seja, cujas alturas estão no vetor altura13) têm uma altura inferior à média calculada. A cada verificação, se a altura do aluno é menor que a média, o contador



count2 é incrementado. Finalmente, um alert é utilizado para exibir o número de alunos com mais de 13 anos cuja altura é inferior à média de altura de todos os alunos.

13.

```
<script>
var meses = ['janeiro', 'fevereiro', 'março', 'abril', 'maio', 'junho', 'julho', 'agosto', 'setembro', 'outubro', 'novembro', 'dezembro']
var meses_de_novo = ['janeiro', 'fevereiro', 'março', 'abril', 'maio', 'junho', 'julho', 'agosto', 'setembro', 'outubro', 'novembro', 'dezembro']
var temp = []
var meses_mais_uma_vez = []
var passaram = []
var lista = []
var count = 12
var soma = 0
while (count != 0){
    var num = parseFloat(prompt('Insira aqui a média de temperatura do mês de ' + meses.shift() + ':'))
    temp.push(num)
    count -= 1
}
for (temperaturas in temp){
    soma += temp[temperaturas]
}
soma /= 12
for (temperaturas in temp){
    if (soma < temp[temperaturas]){
        passaram.push(temp[temperaturas])
        var mes_passou = meses_de_novo.shift()
        meses_mais_uma_vez.push(mes_passou)
    } else {
        meses_de_novo.shift()
    }
}
for (show in passaram){
    lista.push(passaram[show])
    lista.push(' - ')
    lista.push(meses_mais_uma_vez[show])
    lista.push('\n')
}
alert(lista)
</script>
```

O décimo terceiro exercício tem como objetivo calcular a média anual das temperaturas mensais, identificar quais temperaturas estão acima dessa média e mostrar em que meses essas temperaturas ocorreram. O código inicia com a definição de dois vetores: meses, que contém os nomes dos meses do ano em português, e meses\_de\_novo, uma cópia desse vetor que será usada posteriormente. Além disso, são criados os vetores temp para armazenar as temperaturas e meses\_mais\_uma\_vez para armazenar os meses correspondentes às temperaturas que estão acima da média. Também são definidas as variáveis passaram para armazenar as temperaturas acima da média e lista para formatar a saída final. Um loop while é utilizado para coletar a temperatura média de cada mês. A cada iteração, a temperatura média do mês é solicitada ao usuário via prompt, e a temperatura é armazenada no vetor temp. O mês é removido do vetor meses para garantir que a coleta de dados corresponda aos meses corretos. O loop continua até que todas as 12 temperaturas sejam coletadas. Após coletar todas as temperaturas, um loop for calcula a soma de todas as temperaturas armazenadas no vetor temp, e a média anual é obtida dividindo a soma total por 12. Outro loop for verifica quais temperaturas estão acima da média anual. Se uma temperatura estiver acima da média, ela é adicionada ao vetor passaram, e o nome do mês correspondente é removido do vetor meses\_de\_novo e adicionado ao vetor meses\_mais\_uma\_vez. Finalmente, um loop for é usado para formatar a saída, onde as temperaturas acima da média e seus respectivos meses são concatenados e armazenados no vetor lista. A lista formatada é então exibida através de um alert. O resultado final mostra todas as temperaturas que estão acima da média anual e os meses correspondentes onde essas temperaturas ocorreram.

14.

```

<script>
var perguntas = ["Telefonou para a vítima?", "Esteve no local do crime?", "Mora perto da vítima?", "Devia para a vítima?", "Já trabalhou com a vítima?"]
count_respostas = 0
for (i = 0; i < perguntas.length; i++) {
  var pergunta = prompt(perguntas[i]).toUpperCase();
  if (pergunta === 'SIM' || pergunta === 'S') {
    count_respostas++;
  }
}
if (count_respostas == 2){
  alert('Suspeita')
} else if (count_respostas == 3 || count_respostas == 4){
  alert('Cúmplice')
} else if (count_respostas == 5){
  alert('Assassino')
} else{
  alert('Inocente')
}
</script>

```

O décimo quarto exercício consiste em um programa que faz uma série de 5 perguntas a uma pessoa sobre sua possível participação em um crime e, com base nas respostas, classifica o grau de envolvimento da pessoa. A primeira etapa do código envolve a criação de um vetor chamado perguntas, que contém as seguintes questões: "Telefonou para a vítima?", "Esteve no local do crime?", "Mora perto da vítima?", "Devia para a vítima?" e "Já trabalhou com a vítima?". Esse vetor é utilizado para armazenar as perguntas que serão feitas ao usuário. Em seguida, o código inicializa uma variável count\_respostas com o valor 0. Esta variável será usada para contar o número de respostas afirmativas fornecidas pela pessoa. Um loop for é utilizado para iterar sobre o vetor perguntas. A cada iteração, o programa exibe uma pergunta usando o prompt e captura a resposta do usuário, convertendo-a para maiúsculas com o método toUpperCase() para garantir consistência na comparação. Se a resposta for "SIM" ou "S", a variável count\_respostas é incrementada. Após a coleta das respostas, o programa utiliza uma série de instruções if-else para determinar a classificação da pessoa com base no número total de respostas afirmativas. Se a variável count\_respostas for igual a 2, a pessoa é classificada como "Suspeita". Se for 3 ou 4, a pessoa é classificada como "Cúmplice". Se for igual a 5, a pessoa é classificada como "Assassino". Para qualquer outro número de respostas afirmativas, a pessoa é classificada como "Inocente". Finalmente, o programa utiliza alert para mostrar a classificação da pessoa ao usuário, fornecendo uma avaliação clara do envolvimento com base nas respostas fornecidas.

15.

```

<script>
    var lista = []
    var maiores = []
    var soma = 0
    var media = 0
    var qtd_maiores = 0
    var qtd = 0
    var reverso = 0
    var qtd_sete = 0
    var count = 0
    var mensagem = ''
    while (true){
        var num = parseFloat(prompt('Insira a nota (-1 para encerrar): '))
        if (num == -1){
            break
        }else{
            lista.push(num)
        }
    }
    qtd = lista.length

    for (i in lista){
        soma += lista[i]
    }
    reverso = lista.reverse()
    for (i = reverso.length - 1; i >= 0; i--) {
        mensagem += reverso[i] + '\n'
    }
    media = soma/qtd
    for (i in lista){
        if (lista[i] > media){
            maiores.push(lista[i])
        }
    }
    qtd_maiores = maiores.length
    for (i in lista){
        if (lista[i] < 7){
            qtd_sete += 1
        }
    }

    alert('Quantidade de termos digitados: ' + qtd)
    alert('Lista completa ' + lista)
    alert('Números digitados, um embaixo do outro:\n'+mensagem)
    alert('Soma dos números digitados: ' + soma)
    alert('Média: ' + media)
    alert('Quantidade de números maiores que a média: ' + qtd_maiores)
    alert('Quantidade de números menores que 7: ' + qtd_sete)
    alert('Acabou.')
</script>

```

O décimo quinto exercício tem como objetivo criar um programa que leia um número indeterminado de notas inseridas pelo usuário, interrompendo a coleta quando um valor igual a -1 for informado, valor que não deve ser armazenado. O código começa com a inicialização de um vetor chamado lista, que será utilizado para armazenar as notas inseridas. Além disso,

são inicializadas várias variáveis: maiores para armazenar as notas que são superiores à média calculada, soma para acumular a soma total das notas, media para calcular a média das notas, qtd\_maiores para contar o número de notas acima da média, qtd para registrar a quantidade total de notas, reverso para armazenar a lista em ordem inversa e qtd\_sete para contar as notas abaixo de 7. A variável mensagem será utilizada para formatar a exibição dos valores na ordem inversa. Um loop while é empregado para coletar as notas do usuário. Dentro do loop, o programa solicita ao usuário que insira uma nota usando o prompt. Se a nota inserida for -1, o loop é encerrado; caso contrário, a nota é adicionada ao vetor lista. Após a coleta dos dados, o número total de notas é calculado e armazenado em qtd. O próximo passo do código é calcular a soma de todas as notas. Um loop for é utilizado para iterar sobre cada elemento do vetor lista, acumulando a soma das notas na variável soma. Em seguida, a ordem do vetor lista é invertida usando o método reverse, e as notas são formatadas em uma string chamada mensagem, com cada nota exibida em uma nova linha. A média das notas é então calculada dividindo a soma total das notas pelo número total de notas. Para determinar quantas notas são superiores à média, um loop for é utilizado para verificar cada nota, adicionando as notas que são maiores que a média ao vetor maiores. O número de notas acima da média é armazenado em qtd\_maiores. Outro loop for é utilizado para contar quantas notas são menores que 7, atualizando a variável qtd\_sete com esta quantidade. Por fim, o programa utiliza uma série de alerts para exibir os resultados. O primeiro alert mostra a quantidade total de notas inseridas, o segundo exibe a lista completa de notas na ordem em que foram inseridas, e o terceiro apresenta as notas na ordem inversa, uma abaixo da outra. Em seguida, o programa exibe a soma das notas, a média calculada, a quantidade de notas acima da média e a quantidade de notas menores que 7. O programa termina com uma mensagem final que indica o encerramento do processo.

```

<script>
var lista = []
var lista2 = []
var meta = 200
var meta_max = 299

while (true) {
  var num = parseFloat(prompt('Insira a venda bruta do vendedor (-1 para encerrar): '))
  if (num === -1) {
    break
  }
  lista.push(num)
}

for (var i = 0; i < lista.length; i++) {
  var comissao = lista[i] * 0.09 + 200
  var categoria = 2

  while (comissao > meta_max) {
    meta += 100
    meta_max += 100
    categoria++
  }

  lista2.push(categoria)
}

var categorias = {} //lembrar que isso é um dicionário
for (var i = 2; i <= 9; i++) {
  categorias[i] = lista2.filter(x => x === i).length
}
categorias[10] = lista2.filter(x => x >= 10).length

alert('$200 - $299 = ' + categorias[2] +
      '\n$300 - $399 = ' + categorias[3] +
      '\n$400 - $499 = ' + categorias[4] +
      '\n$500 - $599 = ' + categorias[5] +
      '\n$600 - $699 = ' + categorias[6] +
      '\n$700 - $799 = ' + categorias[7] +
      '\n$800 - $899 = ' + categorias[8] +
      '\n$900 - $999 = ' + categorias[9] +
      '\n$1000 em diante = ' + categorias[10])
</script>

```

O décimo sexto exercício tem como objetivo determinar a distribuição de salários dos vendedores em diferentes intervalos usando um array de contadores. O programa começa criando duas listas: uma para armazenar as vendas brutas dos vendedores e outra para categorizar os salários resultantes. O salário de um vendedor é calculado com base em sua venda bruta, com uma fórmula que adiciona uma comissão de 9% sobre a venda bruta a um pagamento fixo de \$200. O usuário insere as vendas brutas até que o valor -1 seja informado para encerrar a entrada de dados. Para cada venda bruta inserida, o salário correspondente é calculado e categorizado em uma faixa específica. A categorização é realizada através de uma série de condições, ajustando um intervalo de faixa salarial que é incrementado a cada iteração para encontrar a faixa correta. A lista de categorias é então analisada para contar quantos salários caem em cada intervalo específico. Os resultados são armazenados em um dicionário, e o número de vendedores em cada faixa salarial é exibido por meio de um alert,

que detalha quantos vendedores receberam salários em cada uma das faixas de \$200 a \$299, \$300 a \$399, \$400 a \$499, e assim por diante, até \$1000 e além.

17.

```
<script>
    var count = 1
    var nomes = []
    var saltos = []
    var medias = []
    var historico = []
    var media = 0
    var maior = 0
    var id = 0
    while (true){
        var nome = prompt('Insira o nome do atleta (deixe em branco para encerrar): ')
        if (nome === ''){
            break
        }
        nomes.push(nome)
        while (count != 6){
            var salto = parseFloat(prompt('Insira o valor do '+count+'º salto: '))
            saltos.push(salto)
            count += 1
        }
        count = 1
        for (salto in saltos){
            media += saltos[salto]
        }
        media == media/5
        medias.push(media)
        historico.push(saltos)
        saltos = []
    }
    for (i in medias){
        if (medias[i]>maior){
            maior = medias[i]
            id = i
        }
    }

    alert('GANHADOR: ' + nomes[id] + '\nSaltos: '+historico[id].join(' - ')+'\nMédia: '+maior)
</script>
```

O décimo sétimo exercício tem como objetivo calcular a média dos saltos de distância de um atleta em uma competição e determinar o vencedor com base na maior média de saltos. O programa começa solicitando o nome do atleta e as distâncias dos cinco saltos realizados por ele. Esse processo se repete até que o nome do atleta não seja informado, indicando o término da entrada de dados. Para cada atleta, o nome e os cinco saltos são armazenados em listas separadas. Em seguida, o programa calcula a média dos cinco saltos, que é armazenada em uma lista de médias. Após a coleta de todos os dados, o programa determina qual atleta possui a maior média de saltos, comparando todas as médias armazenadas. O resultado final é exibido em um alert, mostrando o nome do atleta com a maior média, os valores de todos os saltos realizados por esse atleta e a média dos saltos. A saída é formatada de acordo com o exemplo fornecido, incluindo os saltos em ordem e a média calculada.

18.

