

Título da Aula: Estruturas de Repetição: For, While e Do-While

1. Objetivos da Aula

- Entender o conceito de estruturas de repetição.
- Aprender a usar os laços `for`, `while` e `do-while`.
- Identificar cenários práticos para cada tipo de estrutura.
- Aplicar os conhecimentos em exemplos e exercícios.

2. Introdução às Estruturas de Repetição

• O que são?

Estruturas de repetição permitem executar um bloco de código várias vezes, baseando-se em uma condição lógica.

• Por que usar?

- Redução da duplicação de código.
- Facilitação de tarefas repetitivas.
- Melhor organização e legibilidade.

• Tipos de estruturas de repetição em muitas linguagens de programação

- **For**: Quando o número de iterações é conhecido.
- **While**: Quando a repetição depende de uma condição ser verdadeira.
- **Do-While**: Garante ao menos uma execução antes de verificar a condi

3. Estrutura de Repetição: FOR

Quando usar?

Use o `for` quando você souber antecipadamente o número de iterações necessárias ou quando estiver lidando com contadores ou sequências bem definidas.

- Exemplo: Iterar por elementos de um array, executar cálculos sobre uma sequência de números, ou exibir tabelas.

Cenários comuns:

- Criar tabelas de multiplicação.

- Iterar por índices em uma lista ou matriz.
- Executar ações repetidas por um número fixo de vezes.

Sintaxe geral:

```
for (inicialização; condição; incremento/decremento)
{
    // bloco de código
}
```

Funcionamento:

1. Executa a inicialização.
2. Avalia a condição: se for verdadeira, executa o bloco de código.
3. Realiza o incremento/decremento.
4. Repete enquanto a condição for verdadeira.

Exemplo: Imprimir números de 1 a 5.

```
for (let i = 1; i <= 5; i++) {
    console.log(i);
}
```

- **Vantagens:** Estrutura compacta e ideal para laços com contadores.

4. Estrutura de Repetição: WHILE

Quando usar?

Utilize o **while** quando o número de iterações não é conhecido antecipadamente e a repetição depende de uma condição que pode mudar dinamicamente.

- Exemplo: Continuar solicitando entrada do usuário até que ele insira um valor válido, ou processar dados enquanto houver elementos disponíveis.

Cenários comuns:

- Verificar a validade de entradas do usuário.
- Monitorar a chegada de eventos ou mudanças de estado em tempo real.
- Manter o laço até que uma condição externa seja atendida.

- **Sintaxe geral:**

```
while (condição) {  
  
    // bloco de código  
  
}
```

Funcionamento:

1. Avalia a condição antes de executar o bloco.
2. Se verdadeira, executa o bloco de código.
3. Repete enquanto a condição for verdadeira.

Exemplo: Contar até 5.

```
let i = 1;  
  
while (i <= 5) {  
  
    console.log(i);  
  
    i++;  
  
}
```

- **Vantagens:** Adequado para laços cuja quantidade de iterações é desconhecida no início.

5. Estrutura de Repetição: DO-WHILE

Quando usar?

Prefira o `do-while` quando você precisa garantir que o bloco de código será executado pelo menos uma vez, independentemente da condição inicial.

- Exemplo: Apresentar um menu interativo antes de verificar se o usuário deseja continuar ou sair.

Cenários comuns:

- Exibir um menu de opções.
- Executar uma ação inicial antes de validar uma condição.
- Realizar leituras ou ações iniciais obrigatórias.

Sintaxe geral:

```
do {  
  
    // bloco de código  
  
} while (condição);
```

Funcionamento:

1. Executa o bloco de código pelo menos uma vez.
2. Avalia a condição após a execução.
3. Repete enquanto a condição for verdadeira.

Exemplo: Garantir a execução inicial antes de verificar a condição.

```
let i = 1;
```

```
do {  
  
    console.log(i);
```

i++;

} while (i <= 5);

- **Vantagem:** Garante a execução inicial mesmo que a condição seja falsa.

6. Diferenças Entre For, While e Do-While

Aspecto	For While Do-While
Condicional	Avaliada antes do laço Avaliada antes do laço Avaliada após o laço
Execução Mínima	0 vezes 0 vezes 1 vez
Usabilidade	iterações definidas Condições Indefinidas Execução garantida

Resumo: Quando Usar Cada Tipo

Tipo	Quando Usar Exemplo de Cenário
For	Quando o número de iterações é conhecido previamente ou depende de um contador explícito. Percorrer índices de um array ou executar um cálculo fixo.
While	Quando o número de iterações é desconhecido e depende de uma condição lógica avaliada no início. Continuar processando dados até que uma condição seja atendida.

Do-While	Quando é necessário executar o bloco Apresentar um menu antes de verificar pelo menos uma vez, se o usuário deseja sair. independentemente da condição inicial.
----------	---

7. Aplicações Práticas

1. **For:** Imprimir uma tabela de multiplicação.

```
for (let i = 1; i <= 10; i++) {  
  
    console.log(`5 x ${i} = ${5 * i}`);  
}
```

2. **While:** Verificar senha do usuário até estar correta.1

```
let senha;  
  
while (senha !== "1234") {  
  
    senha = prompt("Digite a senha:");  
}  
  
console.log("Acesso permitido!");
```

3. **Do-While: Menu interativo.**

```
let opcao;  
  
do {
```

```
opcao = prompt("Escolha uma opção:\n1.  
Continuar\n2. Sair");  
  
} while (opcao !== "2");  
  
console.log("Fim do programa.");
```

8. Exercícios

1. Imprimir números de 1 a 10 usando for, while e do-while.
2. Criar um programa que leia números do usuário até que ele digite um número negativo.
3. Escreva um algoritmo que calcule a média dos números digitados pelo usuário, se eles forem pares. Termine a leitura se o usuário digitar zero (0).
4. Escreva um algoritmo que calcule a média aritmética das 3 notas dos alunos de uma classe.
5. O algoritmo deverá ler, além das notas, o código do aluno e deverá ser encerrado quando o código for igual a zero.
6. Construir um algoritmo que calcule a média aritmética de vários valores inteiros positivos, lidos externamente. O final da leitura acontecerá quando for lido um valor negativo.
7. Exibir a soma dos números pares entre 1 e 20 usando um laço for.
8. Implementar um menu interativo com do-while para selecionar opções de um programa.
9. Escreva um programa que imprima os números de 1 a 50.
10. Peça ao usuário um número e exiba a tabuada dele usando **for**.
11. Calcule e exiba a soma de todos os números pares entre 1 e 100.
12. O programa escolhe um número aleatório entre 1 e 10. O usuário deve adivinhar até acertar.
13. Peça ao usuário um número entre 1 e 10. Continue pedindo até que ele forneça um número válido.
14. Escrever um algoritmo que leia um número não determinado de valores e calcule a média aritmética dos valores lidos, a quantidade de valores positivos, a quantidade de valores negativos e o percentual de valores negativos e positivos. Mostre os

resultados.

15. Chico tem 1,50 metro e cresce 2 centímetros por ano, enquanto Zé tem 1,10 metro e cresce 3 centímetros por ano. Construa um algoritmo que calcule e imprima quantos anos serão necessários para que Zé seja maior que Chico.
16. Escrever um algoritmo que gera e escreve os números ímpares entre 100 e 200. 17. Escreva um algoritmo que leia 50 valores e encontre o maior e o menor deles. Mostre o resultado.
18. A prefeitura de uma cidade fez uma pesquisa entre seus habitantes, coletando dados sobre o salário e número de filhos. A prefeitura deseja saber:
- a. média do salário da população;
 - b. média do número de filhos;
 - c. maior salário;
 - d. percentual de pessoas com salário até R\$100,00.

O final da leitura de dados se dará com a entrada de um salário negativo.

19. Em uma eleição presidencial existem quatro candidatos. Os votos são informados através de códigos. Os dados utilizados para a contagem dos votos obedecem à seguinte codificação:

- 1,2,3,4 = voto para os respectivos candidatos;
- 5 = voto nulo;
- 6 = voto em branco;

Elabore um algoritmo que leia o código do candidato em um voto. Calcule e escreva:

- Total de votos para cada candidato;
- Total de votos nulos;
- Total de votos em branco;

Como finalizador do conjunto de votos, tem-se o valor 0.

20. Solicite ao usuário, e armazene-as. Pare de pedir quando ele digitar uma idade

negativa.

21. Considere o código abaixo e complete a tabela com o estado de cada iteração:
let n = 1;

```
while (n < 4) {  
  
    console.log(n * n);  
  
    n++;  
}
```

Preencha a tabela:

iteração	Valor de n antes	saída (n*n)	Valor de n após
1			
2			
3			

22. Analise o seguinte código e preencha a tabela:

```
for (let i = 1; i <= 10; i += 3) {  
  
    console.log(i);  
  
}
```

Preencha

iteração	Valor de i	Saída
----------	------------	-------

1
2
3
4

23. Considere o seguinte código e preencha a tabela de execução:

```
let total = 0;
let contador = 1;
while (contador <= 3) {
    total += contador;
    contador++;
}
console.log(total);
```

Iteração	Valor de contador antes	Valor de total antes	Valor de total após	Valor de contador após
1				
2				
3				

24. Escreva um programa que leia 10 números inteiros e exiba:

- A soma de todos os números.
- A quantidade de números pares.

Exemplo de entrada:

5, 8, 13, 6, 4, 11, 18, 7, 2, 10

Saída esperada:

Soma: 84

Quantidade de pares: 5

25. Crie um programa que continue lendo números inteiros até que o usuário digite o número zero. Ao final, o programa deve exibir:
- A média dos números positivos.
 - A quantidade de números negativos digitados.

Exemplo de entrada:

4, -3, 6, -1, 0

Saída esperada:

Média dos positivos: 5.00

Quantidade de negativos: 2

26. Escreva um programa que peça a senha ao usuário. O programa deve continuar solicitando a senha até que o usuário insira a senha correta, definida previamente no programa. Após o login bem-sucedido, exiba "Acesso permitido".
27. Implemente um programa que simule um carrinho de compras. O programa deve:
- Mostrar uma lista de produtos com preços.
 - Permitir ao usuário adicionar produtos ao carrinho.
 - Exibir o total da compra e a quantidade de itens ao final.

Requisitos:

- Use um laço do-while para permitir que o usuário continue adicionando itens até decidir parar.

Exemplo de execução:

```
Produtos disponíveis:
1 - Maçã (R$ 2,00)
2 - Banana (R$ 1,50)
3 - Laranja (R$ 3,00)

Digite o número do produto ou 0 para finalizar: 1
Digite o número do produto ou 0 para finalizar: 3
Digite o número do produto ou 0 para finalizar: 0

Total: R$ 5,00
Quantidade de itens: 2
```

28. Peça ao usuário um número inteiro positivo e calcule o seu fatorial usando um laço

for. Exemplo de entrada:

Número: 5

Saída esperada:

5! = 120

29. Peça notas ao usuário até que ele escolha parar. Após isso, exiba a média das notas digitadas.

30. Escreva um programa que gere um número aleatório entre 1 e 100.

- O usuário tem 7 tentativas para adivinhar o número.
- A cada tentativa, o programa deve informar se o número digitado é maior ou menor que o número gerado.
- Se o número for acertado antes de acabar as tentativas, exiba uma mensagem de sucesso.

Requisitos:

- Use `while` para controlar as tentativas.

Exemplo de execução:

```
Tente adivinhar o número (1-100): 50
O número é menor.
Tente novamente: 25
O número é maior.
Tente novamente: 30
Parabéns! Você acertou.
```

31. Escreva um programa que solicite ao usuário dois números inteiros, a e b. O programa deve exibir todos os números múltiplos de 5 entre a e b.

Requisitos:

- a. Use for para iterar pelos números no intervalo.
- b. Use um laço aninhado para verificar se o número é primo.

Exemplo de entrada:

Intervalo: 10 a 20

Saída esperada:

Números múltiplos de 5 entre 10 e 20:

32. Implemente um menu que exiba as opções:

- a. Continuar
- b. Sair

Continue executando até que o usuário escolha "Sair".

33. Faça um programa que verifique e mostre os números entre 1.000 e 2.000 (inclusive) que, quando divididos por 11, produzam um resto igual a 5.

34. Faça um programa que leia cinco grupos de quatro valores (A, B, C, D) e mostre-os na

ordem lida. Em seguida, mostre-os em ordem crescente e decrescente.

35. Uma loja tem 15 clientes cadastrados e deseja enviar uma correspondência a cada um deles anunciando um bônus especial. Faça um programa que leia o nome do cliente e o valor de suas compras no ano passado. Calcule e mostre um bônus de 10% se o valor das compras for menor que R\$1.000,00 e de 15%, caso contrário.
36. Faça um programa que leia cinco pares de valores (a,b), todos inteiros e positivos, um de cada vez. Mostre os números inteiros pares de a até b (inclusive).
37. Faça um programa que leia dez conjuntos de dois valores, o primeiro representando o número do aluno e o segundo representando a sua altura em centímetros. a. Encontre o aluno mais alto e o mais baixo.
- b. Mostre o número do aluno mais alto e o número do mais baixo junto com suas alturas.
38. Foi feita uma estatística em cinco cidades brasileiras para coletar dados sobre acidentes de trânsito. Foram obtidos os seguintes dados:
- a. Código da cidade;
- b. Número de veículos de passeio (em 1999);
- c. Número de acidentes de trânsito com vítimas (em 1999).

Deseja-se saber:

I) Qual o maior e menor índice de acidentes de trânsito e a que cidade

pertence; II) Qual a média de veículos nas cinco cidades juntas;

III) Qual a média de acidentes de trânsito nas cidades com menos de 2.000 veículos de passeio.

39. Uma empresa contratou 15 funcionários temporários. De acordo com o valor das vendas mensais, os funcionários adquirem pontos que determinarão seus salários ao final de cada mês. Sabe-se que esses funcionários trabalharão nos meses de novembro a janeiro do ano subsequente.

Faça um programa que:

a) Leia as pontuações nos três meses de cada funcionário;

- b) Calcule e mostre a pontuação geral de cada funcionário nos três meses;
- c) Calcule e mostre a média das pontuações de cada funcionário nos três meses;
- d) Determine e mostre a maior pontuação atingida entre os todos os funcionários nos três meses.

40. Faça um programa que receba duas notas de seis alunos, calcule e

mostre:

- A média aritmética das duas notas de cada aluno;

- A mensagem que está na tabela a seguir:

Média aritmética	Mensagem
Até 3,0	Reprovado
Entre 3,0 e 7,0	Exame
Acima de 7,0	Aprovado

- O total de alunos aprovados;
- O total de alunos de exame;
- Total de alunos reprovados;
- A média da classe.

41. Desenvolver um algoritmo que efetue a soma de todos os números ímpares que são múltiplos de três e que se encontram no conjunto dos números de 1 até 500.

42. Desenvolver um algoritmo que leia a altura de 15 pessoas. Este programa deverá calcular e mostrar:

- a. A menor altura do grupo;
- b. A maior altura do grupo;

43. Desenvolver um algoritmo que leia um número não determinado de valores e calcule e escreva a média aritmética dos valores lidos, a quantidade de valores positivos, a

quantidade de valores negativos e o percentual de valores negativos e positivos.

44. Escrever um algoritmo que leia uma quantidade desconhecida de números e conte quantos deles estão nos seguintes intervalos: [0-25], [26-50], [51-75] e [76-100]. A

entrada de dados deve terminar quando for lido um número negativo. 45. Questão:

Sistema de Resfriamento de Leite

Você foi incumbido de desenvolver um programa em JavaScript para um sistema de resfriamento de leite em uma fazenda. O sistema registra diariamente a quantidade de leite entregue pelos produtores. O tanque de resfriamento tem uma capacidade máxima de mil litros.

O programa deve permitir que os produtores informem a quantidade de leite entregue diariamente em seus baldes. Cada vez que um produtor registra sua entrega, o programa deve calcular e exibir a quantidade total de leite no tanque de resfriamento. O sistema deve impedir que a quantidade total exceda a capacidade máxima do tanque (mil litros).

46. Após cada registro, o programa deve perguntar ao usuário se deseja registrar outra entrega de leite. O loop de registro deve continuar enquanto o usuário escolher continuar (digitando 'S' ou 's') e enquanto houver espaço no tanque.

Ao finalizar os registros, o programa deve exibir algumas estatísticas, incluindo a quantidade total de leite armazenada no tanque.

Requisitos:

- Utilize um loop `do-while` para permitir o registro de múltiplas entregas. -

Impedir que a quantidade total de leite exceda a capacidade máxima do tanque. -

Exiba mensagens apropriadas se o tanque estiver cheio.

- Exemplo de Execução:

Informe a quantidade de leite entregue hoje (em litros):

300 Quantidade total de leite no tanque: 300 litros

Deseja registrar outra entrega de leite? (S/N): S

Informe a quantidade de leite entregue hoje (em litros):

400 Quantidade total de leite no tanque: 700 litros

Deseja registrar outra entrega de leite? (S/N): S

Informe a quantidade de leite entregue hoje (em litros):

350 Quantidade total de leite no tanque: 1050 litros

Tanque de resfriamento cheio! Não é possível registrar mais leite.

Observações:

- Certifique-se de lidar corretamente com as entradas do usuário. - Utilize

variáveis apropriadas para armazenar totais e valores individuais. -

Organize o código de maneira clara e compreensível.

- Considere validar a entrada do usuário para garantir que a quantidade de leite seja positiva e que o tanque não exceda sua capacidade.