

Programa de Mentoria em Lógica de Programação e Algoritmos

Mentores

Amanda Guerra, Anna Ladewig, Lícia Verêda e Raphael Silva.

AGENDA

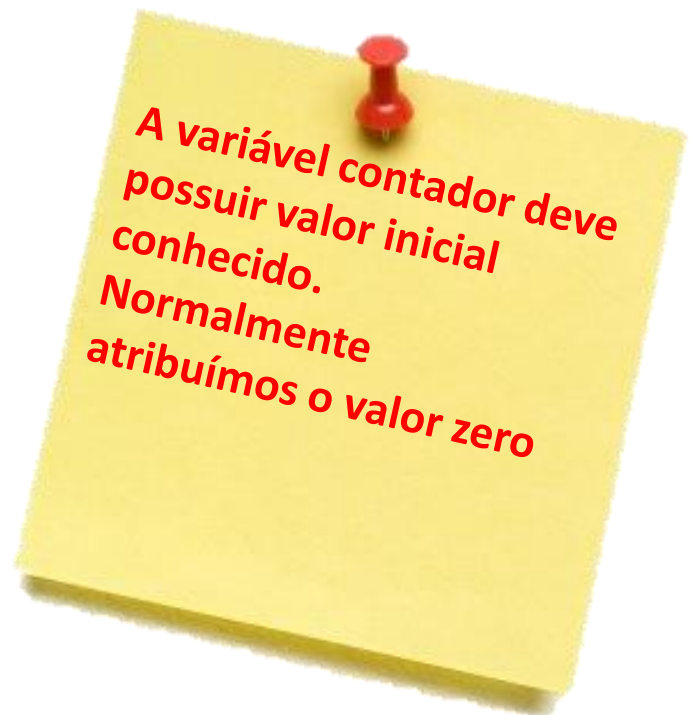
- ✓ Contador;
- ✓ Acumulador;
- ✓ O que é Estrutura de Repetição;
- ✓ Tipos de Estrutura de Repetição;
- ✓ Break;
- ✓ Continue;
- ✓ Exercício.

Contador

- Um contador é uma variável (qualquer) que recebe ela mesma mais um **VALOR CONSTANTE**.

$\text{valor} = \text{valor} + \text{valorConstante}$

```
int contador = 0;
while(contador<=19){
    System.out.print(contador);
    contador = contador + 1;
}
```



Acumulador

- Um acumulador é uma variável que recebe ela mesma mais um **VALOR VARIÁVEL** a cada nova iteração.

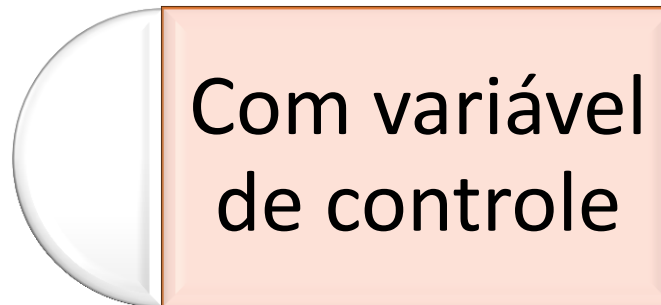
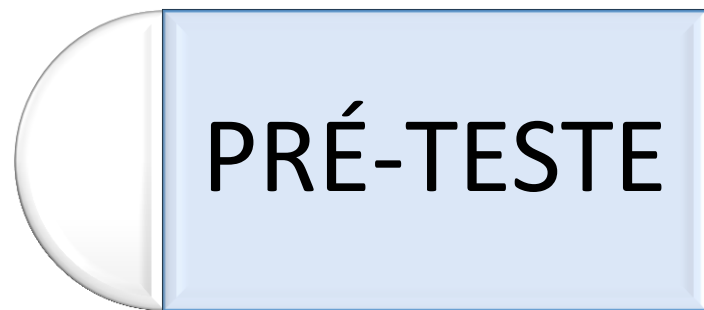
valor = valor + valorVariável

```
5 public class Acumulador {
6
7     public static void main(String[] args) {
8         Scanner scan = new Scanner(System.in);
9         float valorProduto, acumulador = 0;
10
11         for(int i=1; i<=5; i++){
12             System.out.printf("Digite o valor do %dº produto ", i);
13             valorProduto = scan.nextFloat();
14             acumulador = acumulador + valorProduto;
15         }
16         System.out.print("Valor do acumulador é: " + acumulador);
17         scan.close();
18     }
19 }
```

```
Console x Problems Debug Shell
<terminated> Acumulador [Java Application]
Digite o valor do 1º produto 2,30
Digite o valor do 2º produto 4,50
Digite o valor do 3º produto 5,00
Digite o valor do 4º produto 6,20
Digite o valor do 5º produto 7,00
Valor do acumulador é: 25.0
```

Estrutura de Repetição

- É comum chamarmos a estrutura de repetição de loop ou laço.



Estrutura de Repetição

- Tipos:



WHILE

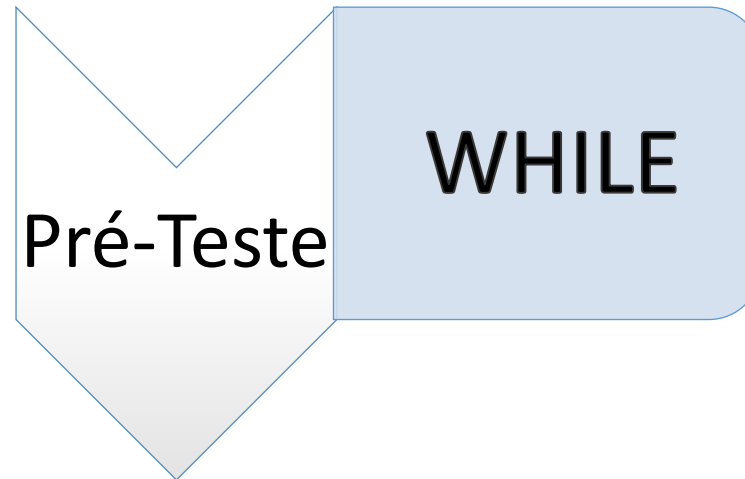


DO WHILE



FOR

Estrutura de Repetição



Estrutura de Repetição - WHILE

- Sintaxe em Java

```
while(condição){  
    /*Instruções a executar enquanto condição  
    for VERDADEIRA*/  
}
```

Enquanto a condição for verdadeira, os comandos dentro do bloco de código serão executados

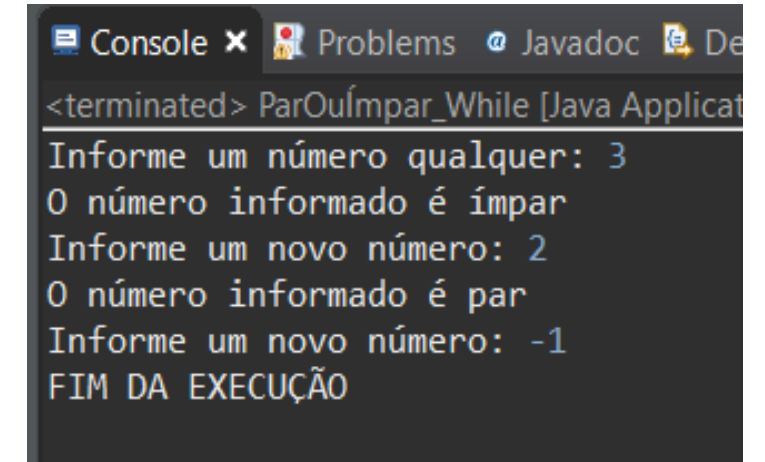
Exemplo 1:

```
3 public class Repetição_While {  
4  
5     public static void main(String[] args) {  
6         int contador = 1;  
7  
8         while (contador<=10) {  
9             System.out.println(contador);  
10            contador++;  
11        }  
12    }  
13 }  
14  
15 }  
16
```

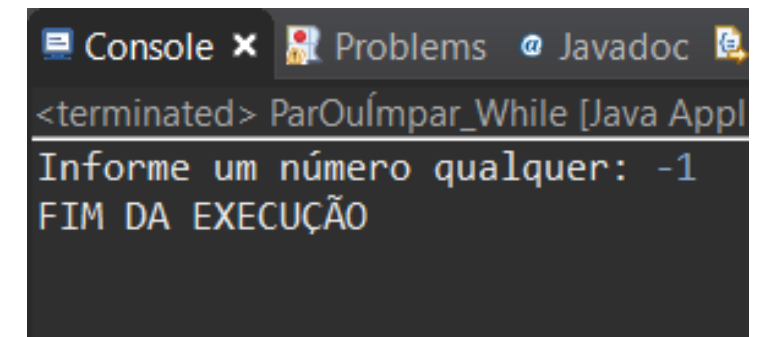

Estrutura de Repetição - WHILE

- Exemplo 2: Condição de parada dada pelo usuário.

```
5 public class ParOuÍmpar_While {
6
7     public static void main(String[] args) {
8         Scanner scan = new Scanner (System.in);
9
10        int numero;
11
12        System.out.print("Informe um número qualquer: ");
13        numero = scan.nextInt();
14
15        while(numero!=-1) {
16            if(numero % 2 ==0) {
17                System.out.println("O número informado é par");
18            }
19            else {
20                System.out.println("O número informado é ímpar");
21            }
22            System.out.print("Informe um novo número: ");
23            numero = scan.nextInt();
24
25        }
26        System.out.println("FIM DA EXECUÇÃO");
27        scan.close();
28    }
29 }
30
```



```
<terminated> ParOuÍmpar_While [Java Applicat
Informe um número qualquer: 3
O número informado é ímpar
Informe um novo número: 2
O número informado é par
Informe um novo número: -1
FIM DA EXECUÇÃO
```



```
<terminated> ParOuÍmpar_While [Java Appl
Informe um número qualquer: -1
FIM DA EXECUÇÃO
```

Estrutura de Repetição



Estrutura de Repetição – DO WHILE

- Sintaxe em Java

```
do{  
    //Instruções a executar  
}  
  
while(condição)
```

```
5 public class Banco_DoWhile {  
6  
7     public static void main(String[] args) {  
8         Scanner scan = new Scanner(System.in);  
9         int operacao;  
10        boolean erro;  
11  
12        do {  
13            System.out.println("Digite 1 - VER SALDO / 2- SACAR : ");  
14            operacao = scan.nextInt();  
15  
16            if(operacao==1 || operacao == 2) {  
17                System.out.print("OPERAÇÃO VÁLIDA\n");  
18                erro = false;  
19  
20            }  
21            else {  
22                System.out.print("OPERAÇÃO INVÁLIDA\n");  
23                erro = true;  
24            }  
25        }  
26        while(erro);  
27  
28        System.out.print("FIM DA EXECUÇÃO");  
29        scan.close();  
30    }  
31  
32 }
```

Estrutura de Repetição – DO WHILE

- Exemplo 2:

```
5 public class ParOuÍmpar_DoWhile {
6
7     public static void main(String[] args) {
8         Scanner scan = new Scanner (System.in);
9
10        int numero;
11
12        System.out.print("Informe um número qualquer: ");
13        numero = scan.nextInt();
14
15        do {
16            if(numero % 2 ==0) {
17                System.out.println("O número informado é par");
18            }
19            else {
20                System.out.println("O número informado é ímpar");
21            }
22            System.out.print("Informe um novo número: ");
23            numero = scan.nextInt();
24        }
25        while(numero!=-1);
26
27        System.out.println("FIM DA EXECUÇÃO");
28        scan.close();
29    }
30
31 }
32
```

```
Console x Problems @ Javadoc Declarat
<terminated> ParOuÍmpar_DoWhile [Java Applic
Informe um número qualquer: 3
O número informado é ímpar
Informe um novo número: 2
O número informado é par
Informe um novo número: -1
FIM DA EXECUÇÃO
```

```
Console x Problems @ Javadoc Declarat
<terminated> ParOuÍmpar_DoWhile [Java Application
Informe um número qualquer: -1
O número informado é ímpar
Informe um novo número: -1
FIM DA EXECUÇÃO
```

Estrutura de Repetição - Pré-Teste x Pós-Teste

```
5 public class ParOuImpar_While {
6
7     public static void main(String[] args) {
8         Scanner scan = new Scanner (System.in);
9
10        int numero;
11
12        System.out.print("Informe um número qualquer: ");
13        numero = scan.nextInt();
14
15        while(numero!=-1) {
16            if(numero % 2 ==0) {
17                System.out.println("O número informado é par");
18            }
19            else {
20                System.out.println("O número informado é ímpar");
21            }
22            System.out.print("Informe um novo número: ");
23            numero = scan.nextInt();
24        }
25        System.out.println("FIM DA EXECUÇÃO");
26        scan.close();
27    }
28 }
29
30
```

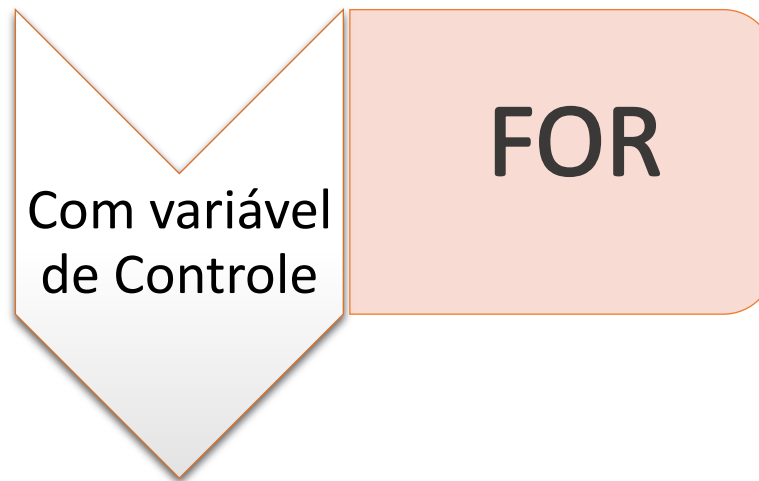


```
5 public class ParOuImpar_DoWhile {
6
7     public static void main(String[] args) {
8         Scanner scan = new Scanner (System.in);
9
10        int numero;
11
12        System.out.print("Informe um número qualquer: ");
13        numero = scan.nextInt();
14
15        do {
16            if(numero % 2 ==0) {
17                System.out.println("O número informado é par");
18            }
19            else {
20                System.out.println("O número informado é ímpar");
21            }
22            System.out.print("Informe um novo número: ");
23            numero = scan.nextInt();
24        }
25        while(numero!=-1);
26
27        System.out.println("FIM DA EXECUÇÃO");
28        scan.close();
29    }
30 }
31
32
```

```
Console x Problems @ Javadoc
<terminated> ParOuImpar_While [Java Appl
Informe um número qualquer: -1
FIM DA EXECUÇÃO
```

```
Console x Problems @ Javadoc Dec
<terminated> ParOuImpar_DoWhile [Java Applic
Informe um número qualquer: -1
O número informado é ímpar
Informe um novo número: -1
FIM DA EXECUÇÃO
```

Estrutura de Repetição



Sabemos de antemão quantas repetições são necessárias!!

Estrutura de Repetição - FOR

- Sintaxe em Java

```
for (Inicialização; Expressão de Teste; Variável de Controle){  
    //Instruções a executar  
}  
for (int Variável = vlr_inicial; Expressão de Teste ; incremento após iteração){  
}
```

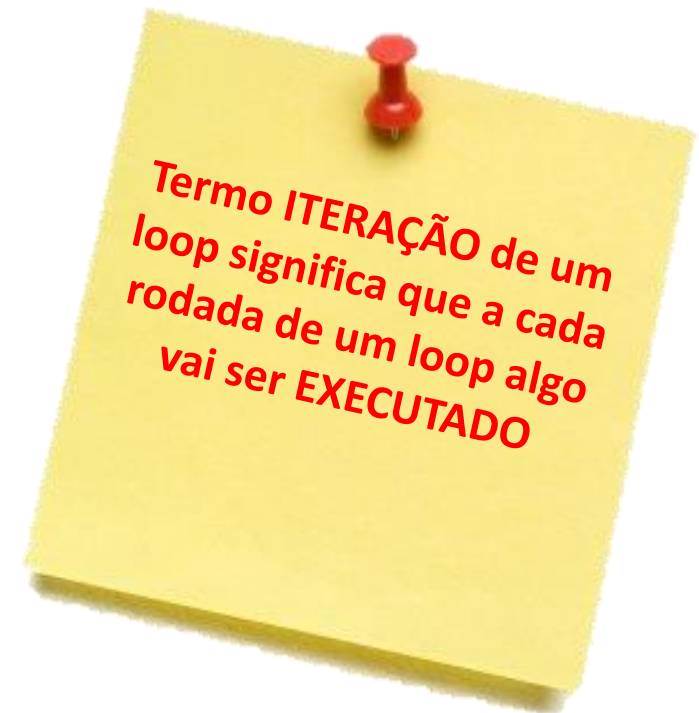
```
3 public class Repetição_FOR {  
4  
5     public static void main(String[] args) {  
6  
7         for (int i = 0; i < 10; i++) {  
8             System.out.println("Número é: " +i);  
9  
10        }  
11    }  
12 }
```

Estrutura de Repetição - FOR

- Sintaxe em Java

```
for (Inicialização; Expressão de Teste; Variável de Controle){  
    //Instruções a executar  
}  
for (int Variável = vlr_inicial; Expressão de Teste ; incremento após iteração){  
}
```

```
3 public class Repetição_FOR {  
4  
5     public static void main(String[] args) {  
6  
7         for (int i = 0; i < 10; i++) {  
8             System.out.println("Número é: " +i);  
9  
10        }  
11    }  
12 }
```

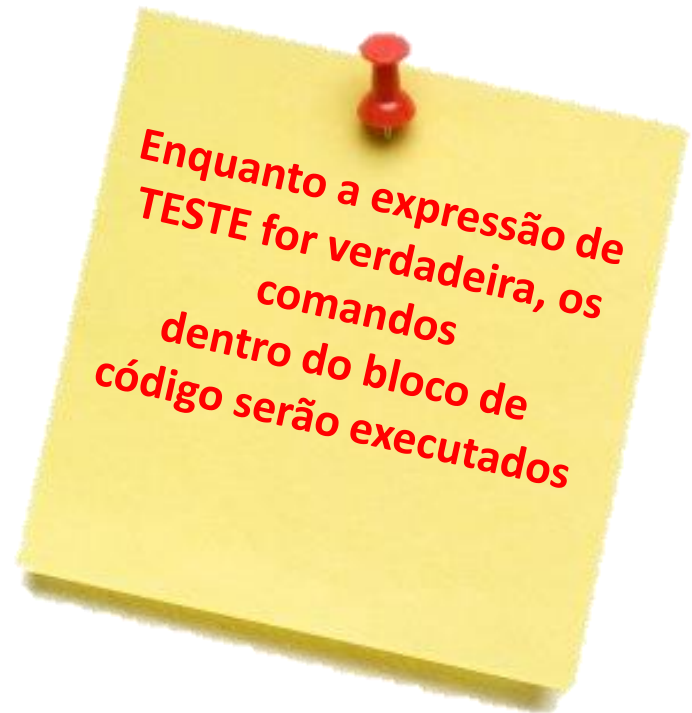


Estrutura de Repetição - FOR

- Sintaxe em Java

```
for (Inicialização; Expressão de Teste; Variável de Controle){  
    //Instruções a executar  
}  
for (int Variável = vlr_inicial; Expressão de Teste ; incremento após iteração){  
}
```

```
3 public class Repetição_FOR {  
4  
5     public static void main(String[] args) {  
6  
7         for (int i = 0; i < 10; i++) {  
8             System.out.println("Número é: " +i);  
9  
10        }  
11    }  
12 }
```



Estrutura de Repetição - FOR

- FOR Decrescente

```
for (Inicialização; Expressão de Teste; Variável de Controle){  
    //Instruções a executar  
}  
for (int Variável = vlr_inicial; Expressão de Teste ; decremento após iteração){  
}
```

```
3 public class Repetição_FOR_Decrescente {  
4  
5     public static void main(String[] args) {  
6  
7         for(int numero = 10 ; numero >= 1; numero--){  
8             System.out.println(" Número é: " + numero);  
9         }  
10    }  
11 }
```

Console x Problema

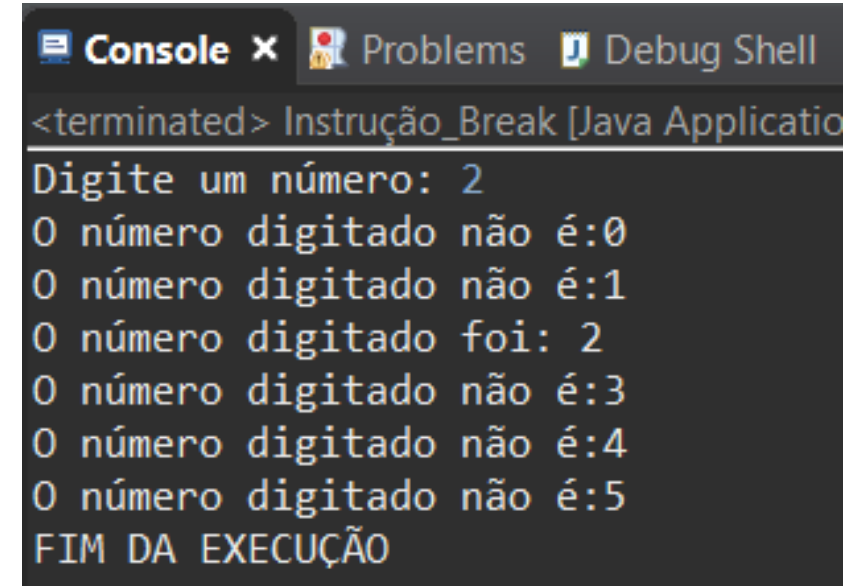
<terminated> Repetição

Número é: 10
Número é: 9
Número é: 8
Número é: 7
Número é: 6
Número é: 5
Número é: 4
Número é: 3
Número é: 2
Número é: 1

Break

- Break quebra a execução do ciclo, sai do loop.

```
5 public class Instrução_Sem_Break {
6
7     public static void main(String[] args) {
8         Scanner scan = new Scanner(System.in);
9
10        int numeroProcurado;
11
12        System.out.print("Digite um número: ");
13        numeroProcurado = scan.nextInt();
14
15        for (int i = 0; i <= 5; i++) {
16
17            if(numeroProcurado == i) {
18                System.out.println("O número digitado foi: " +i);
19            } else {
20                System.out.println("O número digitado não é:" +i);
21            }
22        }
23        System.out.print("FIM DA EXECUÇÃO");
24        scan.close();
25    }
26 }
```



Console x Problems Debug Shell

<terminated> Instrução_Break [Java Applicatio

Digite um número: 2

O número digitado não é:0

O número digitado não é:1

O número digitado foi: 2

O número digitado não é:3

O número digitado não é:4

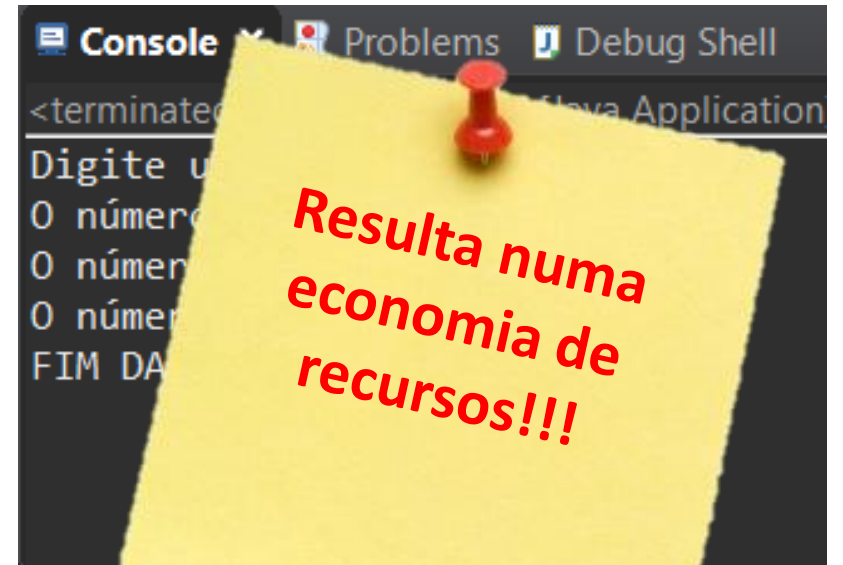
O número digitado não é:5

FIM DA EXECUÇÃO

Break

- Break quebra a execução do ciclo, sai do loop.

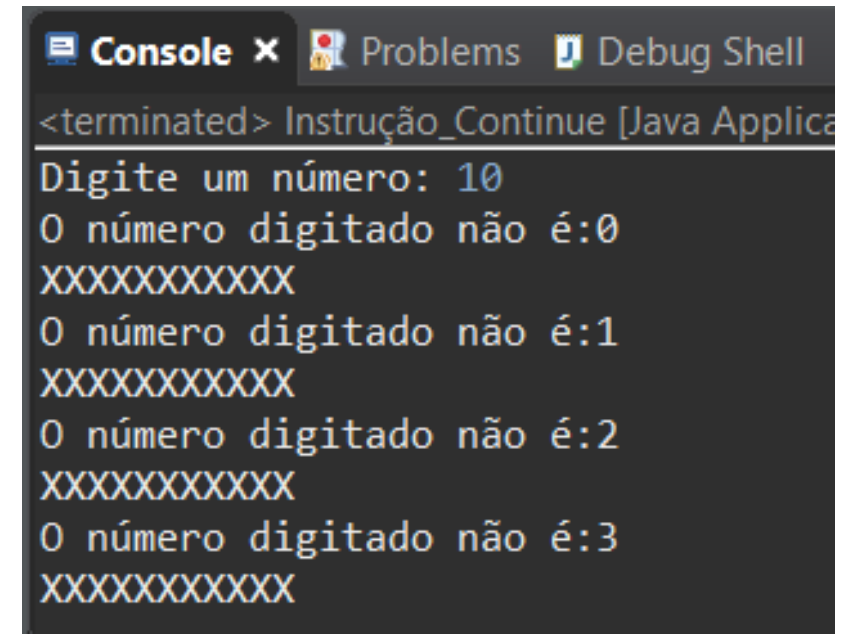
```
5 public class Instrução_Break {  
6  
7     public static void main(String[] args) {  
8         Scanner scan = new Scanner(System.in);  
9  
10        int numeroProcurado;  
11  
12        System.out.print("Digite um número: ");  
13        numeroProcurado = scan.nextInt();  
14  
15        for (int i = 0; i <= 5; i++) {  
16  
17            if(numeroProcurado == i) {  
18                System.out.println("O número digitado foi: " +i);  
19                break;  
20            } else {  
21                System.out.println("O número digitado não é:" +i);  
22            }  
23        }  
24        System.out.print("FIM DA EXECUÇÃO");  
25        scan.close();  
26    }  
27 }  
28
```



Continue

- Continue interrompe o laço.

```
5 public class Instrução_Sem_Continue {  
6  
7     public static void main(String[] args) {  
8         Scanner scan = new Scanner(System.in);  
9  
10        int numeroProcurado;  
11  
12  
13        System.out.print("Digite um número: ");  
14        numeroProcurado = scan.nextInt();  
15  
16        for (int i = 0; i <= 3; i++) {  
17  
18            if(numeroProcurado == i) {  
19                System.out.println("O número digitado foi: " +i);  
20            } else {  
21                System.out.println("O número digitado não é:" +i);  
22            }  
23            System.out.println("XXXXXXXXXXXX");  
24        }  
25        scan.close();  
26  
27    }  
28 }
```

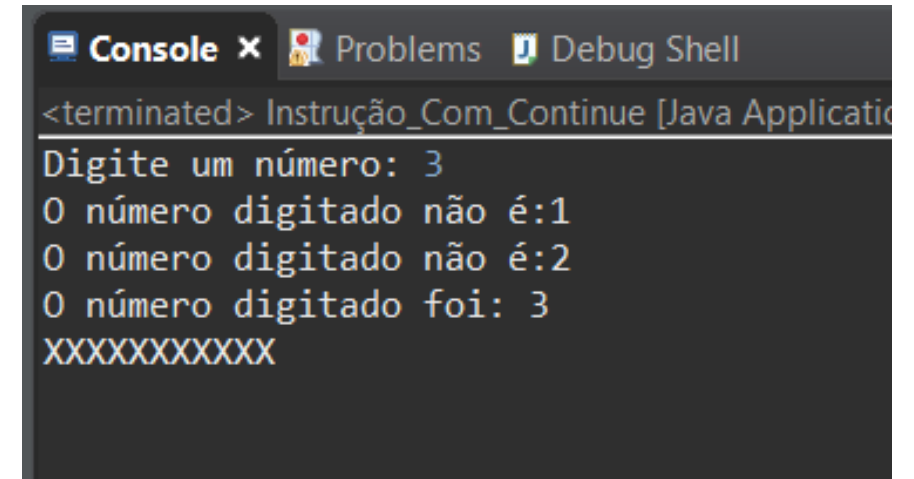


```
Console x Problems Debug Shell  
<terminated> Instrução_Continue [Java Applica  
Digite um número: 10  
O número digitado não é:0  
XXXXXXXXXXXX  
O número digitado não é:1  
XXXXXXXXXXXX  
O número digitado não é:2  
XXXXXXXXXXXX  
O número digitado não é:3  
XXXXXXXXXXXX
```

Continue

- Continue interrompe o laço.

```
5 public class Instrução_Com_Continue {
6
7     public static void main(String[] args) {
8         Scanner scan = new Scanner(System.in);
9
10        int numeroProcurado;
11
12
13        System.out.print("Digite um número: ");
14        numeroProcurado = scan.nextInt();
15
16        for (int i = 1; i <= 3; i++) {
17
18            if(numeroProcurado == i) {
19                System.out.println("O número digitado foi: " +i);
20            } else {
21                System.out.println("O número digitado não é:" +i);
22                continue;
23            }
24            System.out.println("XXXXXXXXXXXX");
25        }
26        scan.close();
27    }
28 }
29 }
```



```
Console x Problems Debug Shell
<terminated> Instrução_Com_Continue [Java Applicati
Digite um número: 3
O número digitado não é:1
O número digitado não é:2
O número digitado foi: 3
XXXXXXXXXXXX
```

Exercício 1

Escreva um programa para receber o nome e idade de vários alunos, até que seja digitado XX.

- Se a idade do aluno for menor que 15 anos, informar que o aluno NÃO PODE SE MATRICULAR;
- Se a idade do aluno for maior ou igual a 15 anos e menor que 21 anos, informar que o aluno SÓ PODERÁ SE MATRICULAR COM O RESPONSÁVEL e receber o nome do responsável;
- Se a idade for igual ou maior que 21 anos, informar PODE SE MATRICULAR.

>>No final mostrar o percentual de matrículas rejeitadas.<<

Exercício 1

```
5 public class Exercício1 {
6
7     public static void main(String[] args) {
8         Scanner scan = new Scanner(System.in);
9
10        String nomeAluno, nomeResponsavel;
11        int contadorMatriculaRejeitada = 0, qtdeAluno = 0, idadeAluno;
12        float percentualMatriculaRejeitada;
13
14        System.out.print("Digite o nome do aluno:");
15        nomeAluno = scan.next();
16
17        while(!nomeAluno.equalsIgnoreCase("XX")) {
18
19            System.out.print("Digite a idade do aluno:");
20            idadeAluno = scan.nextInt();
21
22            if(idadeAluno >= 15 && idadeAluno < 21) {
23                System.out.println("O aluno: SÓ PODERÁ SE MATRICULAR COM O RESPONSÁVEL.\n");
24                System.out.print("Digite o nome do responsável:");
25                nomeResponsavel = scan.next();
26            }
27            else if(idadeAluno >= 21) {
28                System.out.println("O aluno: PODERÁ SE MATRICULAR.\n");
29            }
30            else {
31                System.out.println("O aluno: NÃO PODE SE MATRICULAR.\n");
32                contadorMatriculaRejeitada++;
33            }
34
35            qtdeAluno++;
36            System.out.print("Digite o nome de um novo aluno:");
37            nomeAluno = scan.next();
38        }
39        percentualMatriculaRejeitada = (float) contadorMatriculaRejeitada / qtdeAluno;
40        System.out.printf("O percentual de matrículas rejeitadas foi de: %.2f", percentualMatriculaRejeitada);
41        scan.close();
42    }
43 }
```

Console x Problems Javadoc Declaration

<terminated> Exercício1 [Java Application] C:\Program Files\Java\jre1.8

Digite o nome do aluno: Maria
Digite a idade do aluno: 22
O aluno: PODERÁ SE MATRICULAR.

Digite o nome de um novo aluno: Carla
Digite a idade do aluno: 10
O aluno: NÃO PODE SE MATRICULAR.

Digite o nome de um novo aluno: José
Digite a idade do aluno: 16
O aluno: SÓ PODERÁ SE MATRICULAR COM O RESPONSÁVEL.

Digite o nome do responsável: Roberta
Digite o nome de um novo aluno: XX
O percentual de matrículas rejeitadas foi de: 0,33

Exercício 2

- Escreva um programa para receber o final da placa e o valor de 4 veículos.

TABELA DE PAGAMENTO	
Final da Placa	Mês de Pagto
1 a 3	Janeiro
4 a 6	Fevereiro
7 a 9	Março
0	Abril

- ✓ Informar o mês de pagamento do IPVA;
- ✓ Mostrar o total arrecadado com o IPVA.

>>O valor do IPVA é 10% do valor do veículo<<

Exercício 2

```
70 public static void main(String[] args) {
71     Scanner scan = new Scanner (System.in);
72
73     int placaVeiculo;
74     float valorCarro, totalIPVA = 0;
75     final float PERCENTUAL_IPVA = 0.1f;
76
77     for(int i = 1; i <= 4; i++) {
78         System.out.print("Digite o número final da placa: ");
79         placaVeiculo = scan.nextInt();
80
81         System.out.print("Digite o valor do carro: ");
82         valorCarro = scan.nextFloat();
83
84         if(placaVeiculo >= 1 && placaVeiculo <= 3) {
85             System.out.println("MÊS PAGAMENTO DO IPVA: JANEIRO\n");
86         }
87         else if( placaVeiculo >= 4 && placaVeiculo <=6) {
88             System.out.println("MÊS PAGAMENTO DO IPVA: FEVEREIRO\n");
89         }
90         else if (placaVeiculo >= 7 && placaVeiculo<=9) {
91             System.out.println("MÊS PAGAMENTO DO IPVA: MARÇO\n");
92         }
93         else{
94             System.out.println("MÊS PAGAMENTO DO IPVA: ABRIL\n");
95         }
96         totalIPVA += valorCarro * PERCENTUAL_IPVA;
97     }
98
99     System.out.printf("O valor total arrecadado é de:R$ %.2f.", totalIPVA);
100    scan.close();
101 }
102 }
```

Console x Problems Javadoc Declaration

<terminated> Exercício3_FOR [Java Application] C:\Program

Digite o número final da placa: 0
Digite o valor do carro: 1000
MÊS PAGAMENTO DO IPVA: ABRIL

Digite o número final da placa: 2
Digite o valor do carro: 1000
MÊS PAGAMENTO DO IPVA: JANEIRO

Digite o número final da placa: 5
Digite o valor do carro: 1000
MÊS PAGAMENTO DO IPVA: FEVEREIRO

Digite o número final da placa: 8
Digite o valor do carro: 1000
MÊS PAGAMENTO DO IPVA: MARÇO

O valor total arrecadado é de:R\$ 400,00.

Exercício 3

- **Problema:** Controlar o fundo de caixa* de um estabelecimento.
- **Requisitos:** Como gerente do estabelecimento eu gostaria que o sistema registrasse os valores das despesas de um dia que foram debitadas do fundo de caixa e no final mostrar o saldo do fundo de caixa.
- **Regras do negócio:**
- O valor diário mantido para o fundo de caixa é de R\$ 500,00.
- No dia ocorre pelo menos uma despesa debitada no fundo do caixa
- Nenhuma despesa poderá ser lançada se o valor da despesa for maior que o saldo do fundo de caixa.

>> Reserva em dinheiro mantida no estabelecimento para o pagamento de pequenas despesas do dia a dia<<

Exercício 3

```
5 public class Exercício_DoWhile {
6
7     public static void main(String[] args) {
8
9         Scanner scan = new Scanner(System.in);
10         final float VALOR_DIARIO = 500f;
11         float debito, sobra;
12         String continuar = "S";
13
14         sobra = VALOR_DIARIO;
15
16         do {
17             System.out.print("Digite o valor do débito: ");
18             debito = scan.nextFloat();
19
20             sobra = sobra - debito;
21
22             if(sobra < 0) {
23                 System.out.println("Não pode realizar o débito. Fundo de caixa insuficiente.");
24                 sobra = sobra + debito;
25                 continue;
26             }
27             System.out.printf("O saldo para o fundo de caixa é de %.2f \n", sobra);
28             System.out.print("Deseja continuar (S/N)? ");
29             continuar = scan.next();
30
31             if(continuar.equalsIgnoreCase("N")) {
32                 System.out.println("Sessão encerrada.");
33                 break;
34             }
35
36         } while(sobra >= 0);
37         scan.close();
38     }
```

```
Console x Problems Javadoc Declaration
<terminated> Exercício_DoWhile [Java Application] C:\Program Files\Java\jre
Digite o valor do débito: 100
O saldo para o fundo de caixa é de 400,00
Deseja continuar (S/N)? S
Digite o valor do débito: 200
O saldo para o fundo de caixa é de 200,00
Deseja continuar (S/N)? S
Digite o valor do débito: 300
Não pode realizar o débito. Fundo de caixa insuficiente.
Digite o valor do débito: 200
O saldo para o fundo de caixa é de 0,00
Deseja continuar (S/N)? N
Sessão encerrada.
```

OBRIGADA!



Lícia Verêda



/lícia-verêda



/LiciaVereda