



Exercício: Verificação de Número Par

Instruções:

1. **Defina um número inteiro fixo:** O programa deve receber um número inteiro fixo (por exemplo, 10). Se desejar, você pode alterar esse valor para qualquer outro número inteiro.
2. **Verificação de paridade:** O operador `%` (resto da divisão) é usado para verificar se o número é divisível por 2. Se o resto da divisão for igual a 0, o número é par; caso contrário, é ímpar.
3. **Exibição da mensagem:** Dependendo do resultado da verificação, o programa exibirá a mensagem apropriada: "O número é par" ou "O número é ímpar".

Código do Exercício:

```
public class VerificacaoParImpar {  
    public static void main(String[] args) {  
        int numero = 10; // Número fixo (pode ser alterado)  
  
        if (numero % 2 == 0) {  
            System.out.println("O número é par.");  
        } else {  
            System.out.println("O número é ímpar.");  
        }  
    }  
}
```

Explicação do Código:

1. O programa define um número inteiro fixo (neste caso, 10).
2. O operador `%` é usado para calcular o resto da divisão desse número por 2.
3. Se o resto for igual a 0, o número é par; caso contrário, é ímpar.
4. O programa exibe a mensagem apropriada com base no resultado da verificação.



Exercício: Determinando o Nível de Satisfação

O objetivo deste exercício é criar um programa em Java que, com base em um número fornecido pelo usuário (representando o nível de satisfação), determine e exiba o feedback correspondente.

Instruções:

1. Verificação do Número:

- Declare uma variável inteira chamada `nivel` para armazenar o valor fornecido pelo usuário.
- Utilize a estrutura de controle `switch` para avaliar o valor de `nivel` e atribuir o feedback apropriado com base nos seguintes critérios:
 - Se `nivel` for 1, o feedback é "Muito insatisfeito".
 - Se `nivel` for 2, o feedback é "Insatisfeito".
 - Se `nivel` for 3, o feedback é "Neutro".
 - Se `nivel` for 4, o feedback é "Satisfeito".
 - Se `nivel` for 5, o feedback é "Muito satisfeito".
 - Caso contrário, o feedback é "Opção inválida (deve estar entre 1 e 5)".
- Exiba o feedback resultante na saída padrão usando `System.out.println`.

2. Tipo de switch a ser usado:

- Use o switch expressao, aquele com a setinha `->`.

Código do Exercício:

```
public class NivelSatisfacao {  
    public static void main(String[] args) {  
        // Suponha que o usuário forneceu o número 3 para representar  
        "Neutro"  
        int nivel = 3;  
  
        String feedback;  
  
        switch (nivel) {  
            case 1 -> feedback = "Muito insatisfeito";  
            case 2 -> feedback = "Insatisfeito";  
            case 3 -> feedback = "Neutro";  
            case 4 -> feedback = "Satisfeito";  
            case 5 -> feedback = "Muito satisfeito";  
            default -> feedback = "Opção inválida (deve estar entre 1  
e 5)";  
        }  
  
        System.out.println("Nível de satisfação: " + feedback);  
    }  
}
```

Explicação do Código:

- O programa recebe um número inteiro (de 1 a 5) que representa o nível de satisfação do usuário.
- O `switch` verifica o valor da variável `nivel` e atribui o feedback correspondente à variável `feedback`.
- Se o número não estiver no intervalo de 1 a 5, a mensagem "Opção inválida" será exibida.



Exercício Complementar: Verificador de Mês

Criar um programa em Java que verifica o valor de uma variável representando um mês do ano e exibe o nome desse mês.

Instruções

1. Crie uma variável chamada `mes` para representar um mês do ano (1 para janeiro, 2 para fevereiro, etc.).
2. Use a estrutura `switch` para verificar o valor de `mes`.
3. Com base no valor de `mes`, exiba uma mensagem correspondente ao mês.

Tipo de switch a ser usado

- Use o switch legacy (legado), o switch antigo (que precisa colocar break).

Código do Exercício

```
public class MesDoAnoExercicio {
    public static void main(String[] args) {
        int mes = 5; // Exemplo: 5 representa maio

        switch (mes) {
            case 1:
                System.out.println("Janeiro");
                break;
            case 2:
                System.out.println("Fevereiro");
                break;
            case 3:
                System.out.println("Março");
                break;
            case 4:
                System.out.println("Abril");
                break;
            case 5:
                System.out.println("Maio");
                break;
            case 6:
                System.out.println("Junho");
                break;
            case 7:
                System.out.println("Julho");
                break;
            case 8:
                System.out.println("Agosto");
                break;
            case 9:
                System.out.println("Setembro");
                break;
            case 10:
                System.out.println("Outubro");
                break;
            case 11:
                System.out.println("Novembro");
                break;
            case 12:
                System.out.println("Dezembro");
                break;
            default:
                System.out.println("Valor inválido para mês.");
        }
    }
}
```

Explicação do Código

- Criamos uma variável `mes` e atribuímos um valor (por exemplo, 5 para maio).
- O bloco `switch` verifica o valor de `mes`.
- Cada `case` representa um mês e exibe a mensagem correspondente.
- O `default` é acionado se o valor não corresponder a nenhum dos casos definidos.



Exercício Complementar: Número de Dias no Mês

Instruções

1. Crie uma classe chamada `DiasNoMes`.
2. Dentro da classe, declare uma variável chamada `mes` e inicialize-a com um valor (por exemplo, "Abril").
3. Use um bloco `switch` para determinar o número de dias no mês com base no valor de `mes`.
4. Considere os seguintes cenários:
 - Se `mes` for "Janeiro", "Março", "Maio", "Julho", "Agosto", "Outubro" ou "Dezembro", o número de dias é 31.
 - Se `mes` for "Abril", "Junho", "Setembro" ou "Novembro", o número de dias é 30.
 - Se `mes` for "Fevereiro", o número de dias é 28 (considerando um ano não bissexto).
 - Se `mes` não corresponder a nenhum dos casos anteriores, exiba "Mês desconhecido".
5. Imprima o resultado na saída.

Tipo de switch a ser usado

- Use o **switch inicialização**.

Código do Exercício

```
public class DiasNoMes {
    public static void main(String[] args) {
        String mes = "Abril"; // Altere o valor para testar
        diferentes meses

        int numeroDeDias = switch (mes) {
            case "Janeiro", "Março", "Maio", "Julho", "Agosto",
"Outubro", "Dezembro" -> 31;
            case "Abril", "Junho", "Setembro", "Novembro" -> 30;
            case "Fevereiro" -> 28; // Considerando um ano não
bissexto
            default -> -1; // Valor inválido para mês desconhecido
        };

        if (numeroDeDias != -1) {
            System.out.println("Número de dias em " + mes + ": " +
numeroDeDias);
        } else {
            System.out.println("Mês desconhecido.");
        }
    }
}
```

Explicação do Código

- A variável `mes` é inicializada com um valor (neste caso, "Abril").
- O bloco `switch` avalia o valor de `mes` e retorna o número de dias correspondente.
- Os casos representam diferentes meses e seus respectivos números de dias.
- O bloco `default` é acionado se o valor de `mes` não corresponder a nenhum dos casos anteriores.
- O programa exibe o número de dias na saída.



Exercício Complementar: Verificação de Aprovação Simplificada

Você está desenvolvendo um sistema de notas para uma escola. O sistema deve verificar se um aluno foi aprovado em uma disciplina com base em suas notas. Implemente um programa que utilize operadores lógicos para determinar se um aluno foi aprovado ou não.

Orientações:

1. Declare duas variáveis inteiras: `nota1` e `nota2`.
2. Atribua valores às variáveis `nota1` (primeira nota) e `nota2` (segunda nota).
3. Considere que a aprovação ocorre se a média das notas ($(\text{nota1} + \text{nota2}) / 2$) for maior ou igual a 6.
4. Utilize operadores lógicos para verificar se o aluno foi aprovado.
5. Exiba uma mensagem indicando se o aluno foi aprovado ou não.

Código do Exercício:

```
public class VerificacaoAprovacaoSimplificada {  
    public static void main(String[] args) {  
        int nota1 = 7;  
        int nota2 = 5;  
  
        double media = (nota1 + nota2) / 2.0;  
  
        if (media >= 6) {  
            System.out.println("Aluno aprovado!");  
        } else {  
            System.out.println("Aluno não aprovado.");  
        }  
    }  
}
```

Explicação do Código:

1. As variáveis `nota1` e `nota2` são declaradas e inicializadas com valores fictícios.
2. Calculamos a média das notas usando a fórmula `(nota1 + nota2) / 2.0`.
3. A expressão `(media >= 6 ? "aprovado!" : "não aprovado.")` verifica se a média é maior ou igual a 6 e exibe a mensagem correspondente.



Exercício: Verificação de Aprovação com Base em Notas e Presença

Você está desenvolvendo um sistema de notas para uma escola. O sistema deve verificar se um aluno foi aprovado em uma disciplina com base em suas notas e presença. Implemente um programa que utilize operadores lógicos para determinar se um aluno foi aprovado ou não.

Orientações:

1. Declare três variáveis inteiras: `nota1`, `nota2` e `frequencia`.
2. Atribua valores às variáveis `nota1` (primeira nota), `nota2` (segunda nota) e `frequencia` (percentual de frequência do aluno).
3. Considere que a aprovação ocorre se:
 - A média das notas ($(\text{nota1} + \text{nota2}) / 2$) for maior ou igual a 7.
 - A frequência for igual ou superior a 75%.
4. Utilize operadores lógicos para verificar se o aluno foi aprovado.
5. Exiba uma mensagem indicando se o aluno foi aprovado ou não.

Código do Exercício:

```
public class VerificacaoAprovacaoNotasPresenca {
    public static void main(String[] args) {
        int nota1 = 8;
        int nota2 = 6;
        int frequencia = 80;

        double media = (nota1 + nota2) / 2.0;

        boolean aprovado = media >= 7 && frequencia >= 75;

        if (aprovado) {
            System.out.println("Aluno aprovado!");
        } else {
            System.out.println("Aluno não aprovado.");
        }
    }
}
```

Explicação do Código:

1. As variáveis `nota1`, `nota2` e `frequencia` são declaradas e inicializadas com valores fictícios.
2. Calculamos a média das notas usando a fórmula `(nota1 + nota2) / 2.0`.
3. Utilizamos o operador `&&` para combinar as condições de aprovação (média e frequência).
4. Se o aluno atender a ambas as condições, a variável `aprovado` será verdadeira, caso contrário, será falsa.
5. A mensagem de aprovação ou reprovação é exibida com base no valor da variável `aprovado`.