

# Atividade de Laboratório

## Programação Orientada à Objeto

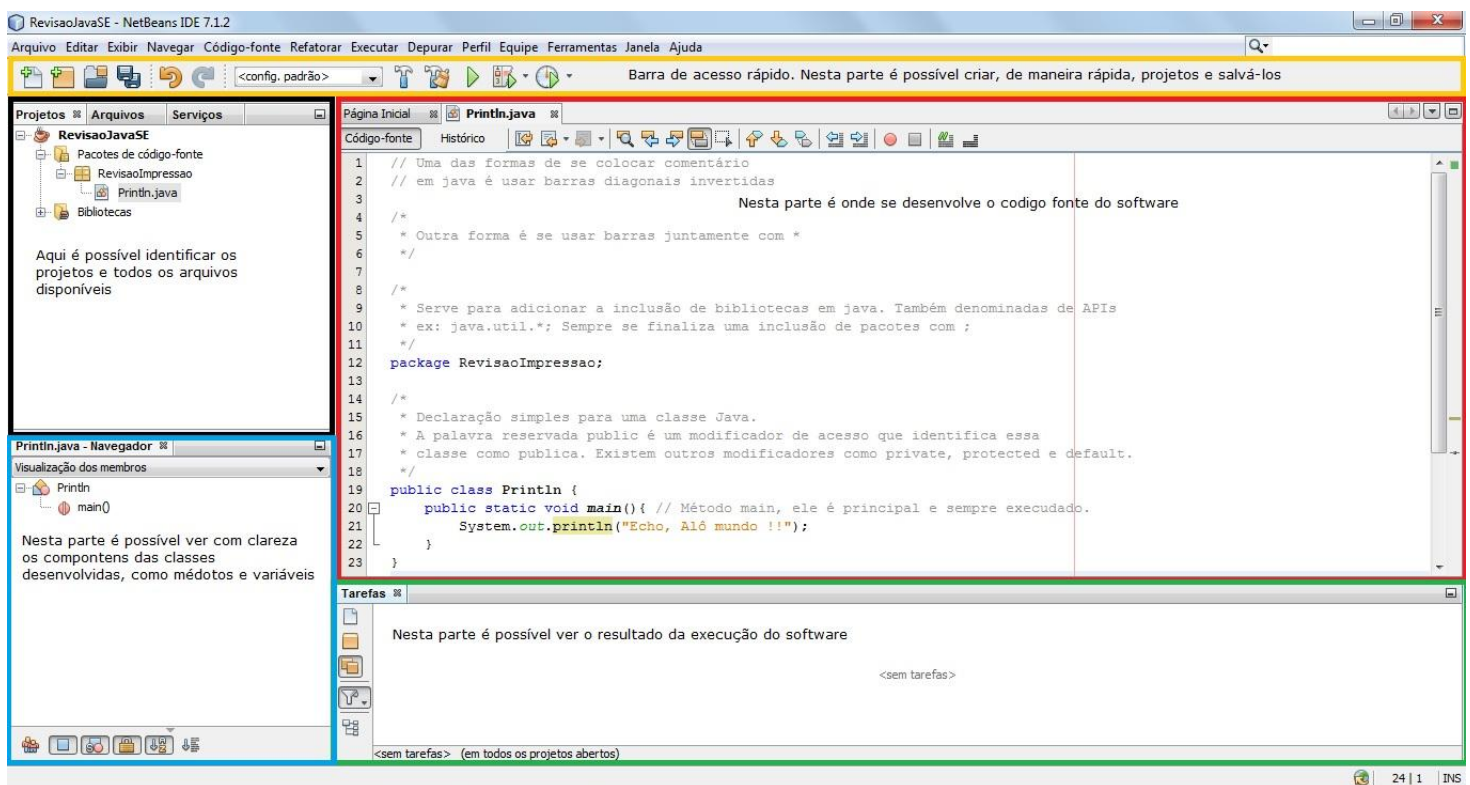
Professor Me. Eng. Gerson Neto.

### Revisão Java SE.

Programas utilizados:

- NetBeans.
- Plataforma Java SE.

Identificação rápida do NetBeans



Atividades a serem realizadas:

- Revisão dos métodos de impressão print, println e printf.
- Revisão das estruturas de seleção if e else.
- Revisão das estruturas de repetição for e while.
- Revisão da criação de classes e métodos.
- Revisão dos operadores aritméticos.
- Revisão dos operadores lógicos.
- Revisão de tópicos em Orientação a Objetos como Classes, Herança e Encapsulamento.

Procedimentos para revisão dos métodos de impressão:

- Crie um projeto, com nome de sua escolha.
- Crie uma classe para cada tipo de método, ex: Println.java.
- Em cada classe crie o método main.
- No escopo do método main, execute os métodos de impressão, veja figuras abaixo.

```
// Uma das formas de se colocar comentário
// em java é usar barras diagonais invertidas

/*
 * Outra forma é se usar barras juntamente com *
 */

/*
 * Serve para adicionar a inclusão de bibliotecas em java. Também denominadas de APIs
 * ex: java.util.*; Sempre se finaliza uma inclusão de pacotes com ;
 */

package RevisaoImpressao;

/*
 * Declaração simples para uma classe Java.
 * A palavra reservada public é um modificador de acesso que identifica essa
 * classe como publica. Existem outros modificadores como private, protected e default.
 */

public class Print {
    public static void main(String args[]){ // Método main, ele é principal e sempre executado.
        System.out.print("Print, Alô mundo !!");
    }
}
```

```
package RevisaoImpressao;

public class Println {
    public static void main(String args[]){
        System.out.println("Println, Alô mundo !!");
    }
}
```

```
package RevisaoImpressao;

public class Printf {
    public static void main(String args[]){
        System.out.printf("%s %s", "Printf", "Alô mundo!");
    }
}
```

Seguindo o exemplo de código abaixo, imprima um quadrado na tela.

```
package RevisaoImpressao;

public class ImpressaoQuadradoOvalSetaLosango {
    public static void main(String args[]){
        /*
         * Impressão Quadrado
         */
        System.out.println("*****");
        System.out.println("**      *");
        System.out.println("**      *");
        System.out.println("**      *");
        System.out.println("**      *");
        System.out.println("**      *");
        System.out.println("**      *");
        System.out.println("**      *");
        System.out.println("**      *");
        System.out.println("**      *");
        System.out.println("*****");
    }
}
```

Procedimentos para revisão das estruturas de seleção:

- Crie um projeto com um nome de sua escolha.
- Crie uma classe com um nome de sua escolha.
- Crie o método main na classe.
- Crie uma instância da classe scanner.
- Utilize a instância da classe scanner para ler dois números inteiros.
- Compare se os números são iguais, qual número é maior ou se são diferentes. Siga o exemplo abaixo.

```
package RevisaoEstruturaSelecao;

public class RevisaoIfElse {
    public static void main(String args[]){
        java.util.Scanner leitor = new java.util.Scanner(System.in);
        int numero1 = 0;
        int numero2 = 0;
        System.out.println("Insira um valor inteiro para o número 1: ");
        numero1 = leitor.nextInt();
        System.out.println("Insira um valor inteiro para o número 2: ");
        numero2 = leitor.nextInt();

        if(numero1 > numero2)
            System.out.println("Numero 1 maior que o numero 2");
        if(numero1 < numero2)
            System.out.println("Numero 2 maior que o numero 1");
        if(numero1 == numero2)
            System.out.println("Numero 1 igual ao numero 2");
        if(numero1 != numero2)
            System.out.println("Numero 1 diferente do numero 2");
    }
}
```

Repita o procedimento como na figura abaixo:

```
package RevisaoEstruturaSelecao;

public class RevisaoIfElse {
    public static void main(String args[]){
        java.util.Scanner leitor = new java.util.Scanner(System.in);
        int numero1 = 0;
        int numero2 = 0;
        System.out.println("Insira um valor inteiro para o número 1: ");
        numero1 = leitor.nextInt();
        System.out.println("Insira um valor inteiro para o número 2: ");
        numero2 = leitor.nextInt();

        if(numero1 > numero2)
            System.out.println("Numero 1 maior que o numero 2");
        else if(numero1 < numero2)
            System.out.println("Numero 2 maior que o numero 1");
        if(numero1 == numero2)
            System.out.println("Numero 1 igual ao numero 2");
        else if(numero1 != numero2)
            System.out.println("Numero 1 diferente do numero 2");
    }
}
```

Procedimento para revisão das estruturas de repetição e operadores aritméticos:

- Crie um projeto, com o nome de sua escolha.
- Crie uma classe para cada estrutura de repetição, ex: rwhile.java.
- Crie o método main para cada classe.
- Utilize a estrutura de repetição para o mesmo exemplo de impressão do quadrado de asteriscos feitos anteriormente, como na figura abaixo.

```
package RevisaoEstruturasRepeticao;

public class rwhile {
    public static void main(String args[]){
        int quantidadeAsteristico = 8;
        System.out.println("*****");
        while( quantidadeAsteristico > 0){
            System.out.println("**          **");
            quantidadeAsteristico--;
        }
        System.out.println("*****");
    }
}
```



Repita o procedimento agora usando a estrutura for, como na figura abaixo:

```
package RevisaoEstruturasRepeticao;

public class rfor {
    public static void main(String args[]){
        System.out.println("*****");
        for(int quantidadeAsteristico = 8; quantidadeAsteristico > 0; quantidadeAsteristico-- )
            System.out.println("**      *");
        System.out.println("*****");
    }
}
```

Utilizando a estrutura de repetição while e a estrutura de seleção if, faça uma calculadora básica, como na figura abaixo. Complete as opções que faltam.

```
package RevisaoEstruturasRepeticao;
import java.util.Scanner;

public class cfor {
    public static void main(String args[]){
        Scanner leitor = new java.util.Scanner(System.in);
        boolean executar = true;
        while(executar){
            System.out.println("Calculadora basica. Para fazer cálculos com apenas dois numeros");
            System.out.println("Digite a opção que você deseja: ");
            System.out.println(" 1-SOMA");
            System.out.println(" 2-SUBTRACAO");
            System.out.println(" 3-MULTIPLICACAO");
            System.out.println(" 4-DIVISAO");
            System.out.println(" 5-SAIR");
            int opcao = leitor.nextInt();
            if(opcao == 1){
                System.out.println("Digite o primeiro numero");
                int numero1 = leitor.nextInt();
                System.out.println("Digite o segundo numero");
                int numero2 = leitor.nextInt();
                System.out.printf("Resultado: %d\n", numero1+numero2);
            }
            /*
             * Complete as opções que faltam
             */
            if(opcao == 5){
                executar = false;
            }
        }
    }
}
```

Procedimento para revisão dos operadores lógicos:

- Crie um projeto com o nome de sua escolha.
- Crie uma classe com o nome de sua escolha.
- Crie o método main na classe.
- Crie um método para ler dois vetores booleanos e montar a tabela verdade do operador AND. Como na figura abaixo.

```
package RevisaoOperadoresLogicos;

public class OperadoresLogicos {
    public void and(boolean[] coluna1, boolean[] coluna2, boolean[] resultado){
        for(int n = 0; n < 4; n++){
            resultado[n] = coluna1[n] && coluna2[n];
            System.out.printf("%s and %s = %s \n",coluna1[n],coluna2[n],resultado[n]);
        }
    }
    public static void main(String args[]){
        boolean[] coluna1 = {true,true,false,false};
        boolean[] coluna2 = {true,false,true,false};
        boolean[] resultado = new boolean[4];
        OperadoresLogicos operadores = new OperadoresLogicos();

        operadores.and(coluna1,coluna2,resultado);
    }
}
```

Crie novos métodos para as outras tabelas verdade, OR e NOT.

Procedimento para revisão de classes, herança e encapsulamento:

- Crie um projeto com nome de sua escolha.
- Crie uma classe abstrata Pessoa.
- Crie uma classe Aluno que herda da classe Pessoa.
- Crie uma classe Professor que herda da classe Pessoa.
- Utilize como base o código das figuras abaixo.

```
package RevisaoClasse;

public abstract class Pessoa {
    private String nome;

    public Pessoa(String nome){
        this.nome = nome;
    }

    public String getNome(){
        return this.nome;
    }
}
```

```
package RevisaoClasse;

public class Aluno extends Pessoa {
    public int numeroMatricula;
    public String turma;
    public Aluno(String nome, int numeroMatricula, String turma){
        super(nome);
        this.numeroMatricula = numeroMatricula;
        this.turma = turma;
    }
}
```

```
package RevisaoClasse;

public class Professor extends Pessoa {
    public int numeroMatricula;
    public String[] turmas;

    public Professor(String nome, int numeroMatricula){
        super(nome);
        this.numeroMatricula = numeroMatricula;
    }

    public void setDisciplinas(String[] turmas){
        this.turmas = turmas;
    }
}
```