

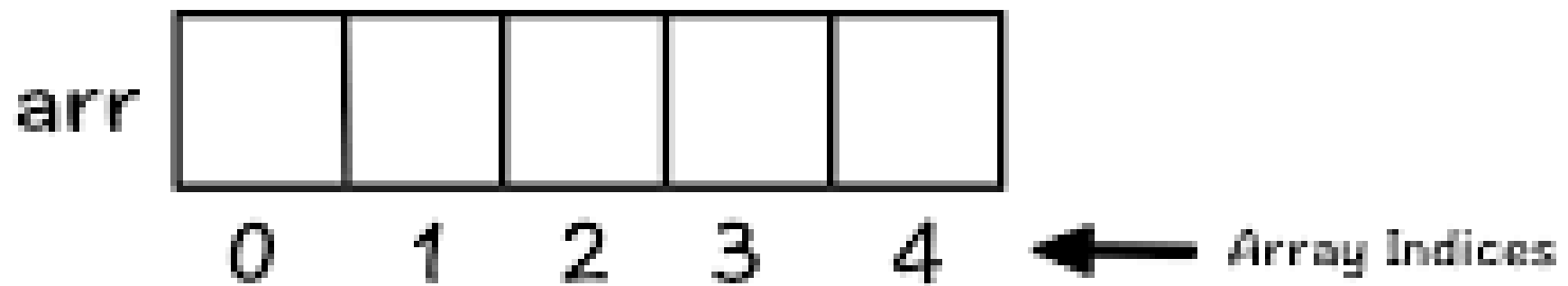


# LINGUAGEM DE PROGRAMAÇÃO I

MSc. Fernanda Dias

CENTRO UNIVERSITÁRIO UNESP

# Array



```
import java.util.Scanner;
```

```
public class Main {  
    public static void main(String[] args) {
```

```
        Scanner scanner = new Scanner(System.in);
```

```
        Pessoa p = new Pessoa();
```

```
        CadastroPessoas cp = new CadastroPessoas();
```

```
        System.out.println("Digite o nome da pessoa a ser cadastrada:");
```

```
        p.setNome(scanner.next());
```

```
        System.out.println("Digite a idade da pessoa a ser cadastrada:");
```

```
        p.setIdade(scanner.nextInt());
```

```
        cp.cadastrarPessoa(p);
```

```
        cp.listar();
```

```
    }
```

```
}
```

```
public class Pessoa {  
    private String nome;  
    private int idade;  
  
    public String getNome() {  
        return nome;  
    }  
  
    public void setNome(String nome) {  
        this.nome = nome;  
    }  
  
    //get e set para idade  
}
```

```
import java.util.ArrayList;
```

```
public class CadastroPessoas {  
    private ArrayList<Pessoa> pessoas = new ArrayList<Pessoa>();  
  
    public void cadastrarPessoa(Pessoa p) {  
        pessoas.add(p);  
        System.out.println("Pessoa cadastrada com sucesso!");  
    }  
  
    public void listar(){  
        for (Pessoa p : pessoas) {  
            System.out.println("Nome:" + p.getNome() + "Idade:" + p.getIdade());  
        }  
    }  
}
```

# HERANÇA?

```
public class PessoaFisica extends Pessoa {  
    @Override  
    public String toString() {  
        return "Nome: " + getNome() + ", Idade: " + getIdade() + ", Tipo: Pessoa Física";  
    }  
}
```

```
public class PessoaJuridica extends Pessoa {  
    private String cnpj;
```

```
//getse sets
```

```
    @Override  
    public String toString() {  
        return "Nome: " + getNome() + ", Idade: " + getIdade() + ", Tipo: Pessoa Jurídica: " + cnpj;  
    }  
}
```

//na classe cadastro de pessoas, adicionar o toString

@Override

```
public String toString() {  
    return "Nome: " + nome + ", Idade: " + idade;  
}
```

//na classe cadastro de pessoas, trocar o void listar

```
public void listar() {  
    for (Pessoa p : pessoas) {  
        System.out.println(p);  
    }  
}
```



```
PessoaFisica pessoaFisica = new PessoaFisica();
```

```
System.out.println("Digite o nome da pessoa física:");
```

```
pessoaFisica.setNome(scanner.next());
```

```
System.out.println("Digite a idade da pessoa física:");
```

```
pessoaFisica.setIdade(scanner.nextInt());
```

```
cp.cadastrarPessoa(pessoaFisica);
```

```
cp.listar();
```

# Polimorfismo?

```
public abstract class Pessoa {
    private String nome;
    private int idade;

    public String getNome() {
        return nome;
    }

    public int getIdade() {
        return idade;
    }

    public void setNome(String nome) {
        this.nome = nome;
    }

    public void setIdade(int idade) {
        this.idade = idade;
    }

    @Override
    public String toString() {
        return "Nome: " + nome + ", Idade: " + idade;
    }
}
```

```
import java.util.ArrayList;

public class CadastroPessoas {
    private ArrayList<Pessoa> pessoas = new ArrayList<>();

    public void cadastrarPessoa(Pessoa pessoa) {
        pessoas.add(pessoa);
        System.out.println("Pessoa cadastrada com sucesso!");
    }

    public void listar() {
        for (Pessoa p : pessoas) {
            System.out.println(p);
        }
    }
}
```

```
public class PessoaFisica extends Pessoa {  
    @Override  
    public String toString() {  
        return super.toString() + ", Tipo: Pessoa Física";  
    }  
}
```

```
public class PessoaJuridica extends Pessoa {  
    private String cnpj;  
  
    public String getCnpj() {  
        return cnpj;  
    }  
  
    public void setCnpj(String cnpj) {  
        this.cnpj = cnpj;  
    }  
  
    @Override  
    public String toString() {  
        return super.toString() + ", Tipo: Pessoa Jurídica" + cnpj;  
    }  
}
```

```
import java.util.Scanner;
public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        CadastroPessoas cp = new CadastroPessoas();

        while (true) {
            System.out.println("Escolha uma opção:");
            System.out.println("1. Cadastrar Pessoa Física");
            System.out.println("2. Cadastrar Pessoa Jurídica");
            System.out.println("3. Listar Pessoas");

            int opcao = scanner.nextInt();

            switch (opcao) {
                case 1:
                    cadastrarPessoa(cp, new PessoaFisica(), scanner);
                    break;
                case 2:
                    cadastrarPessoa(cp, new PessoaJuridica(), scanner);
                    break;
                case 3:
                    cp.listar();
                    break;
            }
        }
    }
}
```

```
default:
    System.out.println("Opção inválida");
}
}
}
```

```
private static void cadastrarPessoa(CadastroPessoas cp, Pessoa pessoa, Scanner scanner) {
```

```
    System.out.println("Digite o nome da pessoa:");
```

```
    pessoa.setNome(scanner.nextLine());
```

```
    System.out.println("Digite a idade da pessoa:");
```

```
    pessoa.setIdade(scanner.nextInt());
```

```
    scanner.nextLine();
```

```
    if (pessoa instanceof PessoaJuridica) {
```

```
        System.out.println("Digite o CNPJ:");
```

```
        ((PessoaJuridica) pessoa).setCnpj(scanner.next());
```

```
    }
```

```
    cp.cadastrarPessoa(pessoa);
```

```
}
```

```
}
```

instanceof



**OBSERVAÇÃO!**  
instanceof

# Lembram dos instrumentos?

```
public class Main {  
    public static void main(String[] args) {  
  
        Instrumento i = new Violao();  
        i.afinar();  
        i.tocar();  
  
        if (i instanceof Cordas) {  
            ((Cordas) i).trocar_cordas();  
            // Agora acessa o método da interface 😊  
        }  
    }  
}
```





**DÚVIDAS?**  
**Vamos falar de Projeto!**