

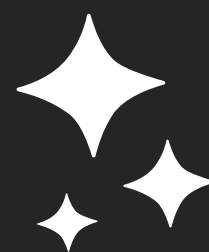


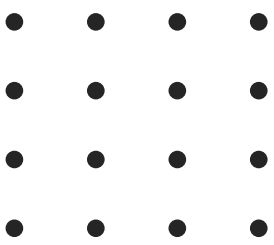
PROGRAMA MONITORIA



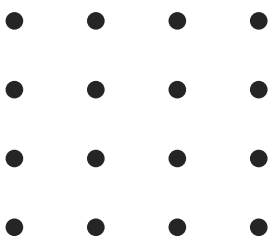
LISTAS DE OBJETOS

PROGRAMAÇÃO ORIENTADA A OBJETOS





```
List<Livro> compras = new ArrayList();  
  
compras.add(new Livro("Java", 25.50));  
compras.add(new Livro("HTML e CSS", 21.30));  
compras.add(new Livro("Android", 22.54));
```



O QUE É LISTA DE OBJETOS?

01

Lista de objetos é uma estrutura de dados que permite armazenar e manipular um **conjunto de objetos** de um mesmo tipo em Java POO.

02

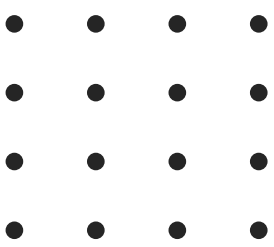
Para criar uma lista de objetos em Java, é necessário criar uma **classe** que represente os objetos que serão armazenados na lista

03

Em seguida, é necessário especificar o tipo de objeto que será armazenado na lista usando a sintaxe **List<TipoDoObjeto>**.

04

Por fim, especificar a classe que implementa a interface List, nesse caso o **Arraylist**.



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

public class ExemploListaDeObjetos {
    public static void main(String[] args) {
        // Cria uma nova lista de objetos Pessoa
        List<Pessoa> pessoas = new ArrayList();

        // Adiciona objetos Pessoa à lista
        pessoas.add(new Pessoa("João")); //primeiro objeto
        pessoas.add(new Pessoa("Maria")); //segundo objeto

        // Acessa o objeto Pessoa na posição 1 (segundo objeto)
        Pessoa segundaPessoa = pessoas.get(1);
        System.out.println("A segunda pessoa é: " + segundaPessoa.getNome());

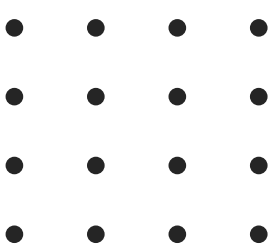
        // Remove o objeto Pessoa na posição 0 (primeiro objeto)
        pessoas.remove(0);
    }
}
```

01

Para **adicionar** um objeto à lista, utiliza-se o método **add()** da lista.

02

Para **acessar** um objeto na lista, utiliza-se o método **get()** da lista, informando o **índice** do objeto que deseja acessar.



03

Para **remover** um objeto da lista, utiliza-se o método **remove()** da lista, informando o **índice** do objeto que se deseja remover.

04

Para **acessar** os atributos de um objeto da lista, utiliza-se os métodos **getters** da classe do objeto.

O QUE É ARRAYLIST?

01

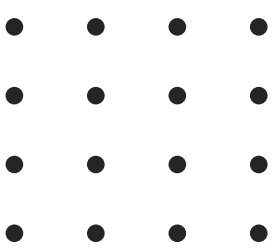
ArrayList é uma classe em Java que implementa a interface List e permite criar listas de objetos em Java POO.

02

Uma classe do pacote java.util e pode ser importada no programa usando a sintaxe **import java.util.ArrayList;**

03

Ela oferece diversas operações para manipulação da lista, como **ordenação**, **busca**, **adição**, **remoção** e **cópia**.



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

public class ExemploListaDeObjetos {
    public static void main(String[] args) {
        // Cria uma nova lista de objetos Pessoa
        List<Pessoa> pessoas = new ArrayList();

        // Adiciona objetos Pessoa à lista
        pessoas.add(new Pessoa("João")); //primeiro objeto
        pessoas.add(new Pessoa("Maria")); //segundo objeto

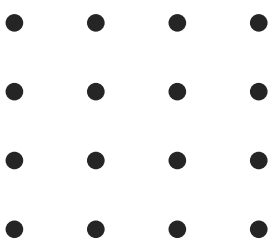
        // Acessa o objeto Pessoa na posição 1 (segundo objeto)
        Pessoa segundaPessoa = pessoas.get(1);
        System.out.println("A segunda pessoa é: " + segundaPessoa.getNome());

        // Remove o objeto Pessoa na posição 0 (primeiro objeto)
        pessoas.remove(0);
    }
```

```
// Percorrendo a lista com um loop foreach
```

```
/*for(<Tipo> <identificação> : <nomeDaLista>) {
    <comando>
}*/
```

```
for (Pessoa pessoa : pessoas) {
    System.out.println(pessoa.getNome());
}
```



CONCEITO: FOR EACH

01


O **loop forEach** é um método da classe ArrayList em Java que permite **percorrer** todos os elementos de uma lista ArrayList e realizar uma operação em cada um deles.

02

Ele é uma alternativa **mais simples** e concisa ao loop for tradicional para percorrer uma lista e realizar uma operação em cada elemento.



ACESSE OS MATERIAIS NO GITHUB :)



Exemplos de atividades/códigos com tudo que envolve
o novo Assunto (Lista de Objetos)

https://github.com/IsabelaSilvaz/Monitoria_Java_POO

