
1. Criando e Manipulando um ArrayList

Objetivo: Ensinar a criação e manipulação básica de um ArrayList.

Exercício:

- Crie um ArrayList de Strings chamado nomes.
- Adicione cinco nomes à lista.
- Exiba todos os nomes no console.
- Remova o terceiro nome da lista.
- Substitua o último nome por “João”.
- Exiba a lista atualizada.

```
import java.util.ArrayList;

public class ExercicioArrayList {
    public static void main(String[] args) {
        ArrayList<String> nomes = new ArrayList<>();

        // Adicionando elementos
        nomes.add("Ana");
        nomes.add("Carlos");
        nomes.add("Maria");
        nomes.add("Pedro");
        nomes.add("Lucas");

        System.out.println("Lista inicial: " + nomes);

        // Removendo o terceiro nome
        nomes.remove(2);

        // Substituindo o último nome
        nomes.set(nomes.size() - 1, "João");

        System.out.println("Lista final: " + nomes);
    }
}
```

2. Criando e Manipulando um LinkedList

Objetivo: Demonstrar o uso de um LinkedList e suas vantagens na inserção e remoção de elementos.

Exercício:

- Crie um LinkedList de números inteiros.
- Adicione os números 10, 20, 30, 40, 50.
- Adicione o número 5 no início e 60 no final da lista.
- Remova o primeiro e o último elemento.
- Exiba a lista final.

```
import java.util.LinkedList;

public class ExercicioLinkedList {
    public static void main(String[] args) {
        LinkedList<Integer> numeros = new LinkedList<>();

        // Adicionando elementos
        numeros.add(10);
        numeros.add(20);
        numeros.add(30);
        numeros.add(40);
        numeros.add(50);

        System.out.println("Lista inicial: " + numeros);

        // Adicionando no início e no fim
        numeros.addFirst(5);
        numeros.addLast(60);

        // Removendo o primeiro e o último elemento
        numeros.removeFirst();
        numeros.removeLast();

        System.out.println("Lista final: " + numeros);
    }
}
```

3. Comparação de Desempenho

Objetivo: Mostrar na prática a diferença de desempenho entre ArrayList e LinkedList ao adicionar elementos no início da lista.

Exercício:

- Crie um ArrayList<Integer> e um LinkedList<Integer>.
- Adicione 100.000 elementos no **início** da lista em cada uma delas.
- Meça e compare o tempo de execução.

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

public class ComparacaoListas {
    public static void main(String[] args) {
        int quantidade = 100000;
```

```

// Teste com ArrayList
List<Integer> arrayList = new ArrayList<>();
long inicioArrayList = System.nanoTime();
for (int i = 0; i < quantidade; i++) {
    arrayList.add(0, i);
}
long fimArrayList = System.nanoTime();

// Teste com LinkedList
List<Integer> linkedList = new LinkedList<>();
long inicioLinkedList = System.nanoTime();
for (int i = 0; i < quantidade; i++) {
    linkedList.add(0, i);
}
long fimLinkedList = System.nanoTime();

// Exibindo os tempos
System.out.println("Tempo ArrayList: " + (fimArrayList - inicioArrayList) / 1e6 + " ms");
System.out.println("Tempo LinkedList: " + (fimLinkedList - inicioLinkedList) / 1e6 + "
ms");
}
}

```

Explicação: Esse exercício permite que os alunos observem que a inserção no início de um ArrayList é muito mais lenta que em um LinkedList, pois o ArrayList precisa deslocar todos os elementos.

4. Exercício de Busca em Lista

Objetivo: Ensinar os alunos a buscar elementos em listas.

Exercício:

- Crie um ArrayList com os seguintes nomes: ["Ana", "Bruno", "Carlos", "Daniel", "Eduardo"].
- Solicite ao usuário um nome para pesquisar.
- Informe se o nome existe na lista e em qual posição ele está.

```

import java.util.ArrayList;
import java.util.Scanner;

public class BuscaArrayList {
    public static void main(String[] args) {
        ArrayList<String> nomes = new ArrayList<>();
        nomes.add("Ana");
        nomes.add("Bruno");
        nomes.add("Carlos");
        nomes.add("Daniel");
        nomes.add("Eduardo");

        Scanner scanner = new Scanner(System.in);
        System.out.print("Digite um nome para buscar: ");
        String nomeBusca = scanner.nextLine();

        int indice = nomes.indexOf(nomeBusca);

        if (indice != -1) {
            System.out.println("O nome " + nomeBusca + " foi encontrado na posição " + indice);
        } else {
            System.out.println("O nome " + nomeBusca + " não está na lista.");
        }

        scanner.close();
    }
}

```