



TECNOLOGIA E INOVAÇÃO EM PROL DA INDÚSTRIA





Curso Técnico em Informática



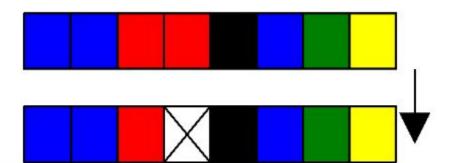
Alocação Dinâmica: Listas

Prof: Diego Corrêa

Vetores n\u00e3o podem ser redimensionados

 Para trabalhar com Vetores, é necessário criar um conjunto de métodos para operá-los

Ex: De contas bancarias

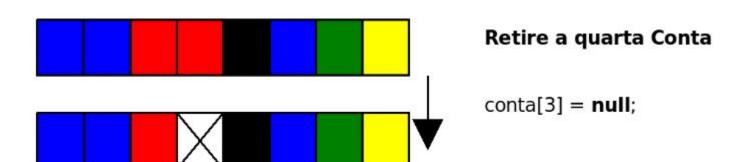


Retire a quarta Conta

conta[3] = null;

 Neste caso, é necessários métodos para analisar se uma posição é nula

 Em outros casos, é necessário métodos para verificar se uma posição já está ocupada, antes de definir objeto



 Devido a isso é necessário uma alocação dinâmica da memória

 Com métodos que tornem possível a operação das Listas



São listas dinâmicas

 Sua grande vantagem é que seu acesso é parecido como os vetores

Isso deixa o acesso mais rápido



Para usá-la, é necessário importar a classe:

- java.util.ArrayList
- java.util.List
- ... existem outros tipos de Lista, como:
 - Set: Elementos não se repetem
 - Map: Mapeamento chave-valor
 - **—** ...



- Instanciando:
 - ArrayList lista = new ArrayList();
- Alguns métodos da ArrayList:
 - lista.add(objeto);
 - lista.add(indice,objeto);
 - lista.size();
 - lista.remove(objeto);
 - lista.remove(indice);
 - lista.forEach((objeto) -> método(objeto))
 - ... diversos outros



<ArrayList> Exercício

- 1. Criar um projeto: Alocacao_Dinamica
- 2. Criar uma lista
- 3. Adicionar 3 Strings: Manoel, Joaquim, Maria
- 4. Imprimir
- 5. Adicionar 2 Strings: Ana, Carlos
- 6. Imprimir



```
package alocacao dinamica;
import java.util.ArrayList;
 * @author lowe
public class Alocacao Dinamica {
     * @param args comandos vindos de fora do programa
    public static void main(String[] args) {
        ArrayList lista = new ArrayList();
        lista.add("Manoel");
        lista.add("Joaquim");
        lista.add("Maria");
        System.out.println(lista);
        lista.add("Ana");
        lista.add("Carlos");
        System.out.println(lista);
```



- É possível limitar o tipo de objeto a ser incluso numa ArrayList
- Casting toda vez que uma Object for usada é um trabalho repetitivo e cansativo
- Para limitar o tipo, devemos fazer a seguinte modificação da declaração

ArrayList<Tipo do Objeto> lista = new ArrayList<Tipo do Objeto>();



- 1. Complementando o exercicio anterior
- 2. Crie um método que receba uma String e imprima na tela
- 3. Na ArrayList lista use o método forEach() para imprimir cada String
- 4. Use o método que diz o tamanho do ArrayList para imprimir quantos elementos tem
- 5. Adícione as Strings: Ana, Carlos
- 6. Remova a String: Manuel
- 7. Faça o 3 e 4 novamente



```
public class Alocacao Dinamica {

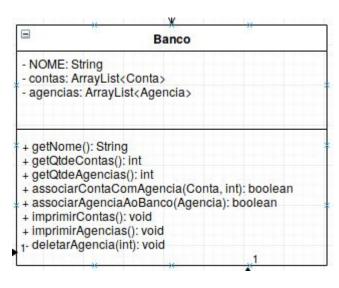
    * @param args comandos vindos de fora do programa.

   public static void main(String[] args) {
        ArrayList<String> lista = new ArrayList();
        lista.add("Manoel");
        lista.add("Joaquim");
        lista.add("Maria");
        lista.forEach((elemento) -> Alocacao Dinamica.imprimirDadosFormatados(elemento));
        System.out.println("O tamanho da list é: " + lista.size());
        lista.add("Ana"):
        lista.add("Carlos");
        lista.remove("Manoel");
        lista.forEach((elemento) -> Alocacao Dinamica.imprimirDadosFormatados(elemento));
        System.out.println("O tamanho da list é: " + lista.size());
    * @param elemento uma String
    * @see
   public static void imprimirDadosFormatados(String elemento) {
        System.out.println("Nome: " + elemento);
```



<ArrayList> UML

No UML a chamada do ArrayList é dado a seguir:





Atividade Final Adaptação do Sistema do Banco



BancoAplicacao + DS SA: Banco + main(): void + preencher_banco(): void + preencher agencias(): void + deletar_agencia(): void Banco NOME: String - contas: ArrayList<Conta> - agencias: ArrayList<Agencia> + getNome(): String + getQtdeContas(): int + getQtdeAgencias(): int + associarContaComAgencia(Conta, int): boolean + associarAgenciaAoBanco(Agencia): boolean + imprimirContas(): void + imprimirAgencias(): void 1- deletarAgencia(int): void Conta - NUMERO: int Agencia - TITULAR: String - NUMERO: int - saldo: double contas: ArrayList<Conta> - limite: double - limiteFixo: double tipoConta: int + getNumero(): int + getTitular(): String + getQtdeContas(): int + atribuirConta(Conta): boolean + getNumero: int + imprimirContas(): void + getSaldo(): double + getLimite(): double + exibeInformacoes(): void + getLimiteFixo(): double + getTipoConta(): int + sacar(double): boolean + depositar(double): void + transferir(Conta, double): boolean + exibeInformacoes(): void + alterarLimite(double): void

SENAI CIMATEC