



TECNOLOGIA E INOVAÇÃO EM PROL DA INDÚSTRIA





Curso Técnico em Informática



Desenvolvimento de Sistemas I – 100h

Profa: Francisleide Almeida



Arrays no JAVA

- O array no JAVA será um objeto;
- Fazem parte do pacote java.util
- São objetos de recipientes que contém um número fixo de valores de um único tipo.
- Cada item em um array é chamado de elemento, e cada elemento é acessado pelo número, o índice.



Exemplo

```
int[] idades;
idades = new int[10];
```

Observe a que é instanciado um objeto idades;



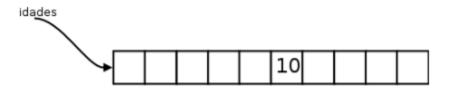


 E para atribuir valores para alguma destas posições do vetor?



 E para atribuir valores para alguma destas posições do vetor?

idades[5] = 10



Os índices do array vão de 0 a n-1, onde n é o tamanho dado no momento em que você criou o array.



• E se quisermos criar um array de objetos?

Conta[] minhasContas; minhasContas = new Conta[10];





Conta[] minhasContas; minhasContas = new Conta[10];

E Quantas contas foram criadas aqui?



Conta[] minhasContas; minhasContas = new Conta[10];

- E Quantas contas foram criadas aqui?
- Na verdade nenhuma!
- Foram criados 10 espaços que você pode utilizar para guardar uma referência a uma Conta... Tente imprimir o valor de saldo da posição 0 do array...



Populando array

```
Conta contaNova = new Conta();
contaNova.saldo = 1000.0;
minhasContas[0] = contaNova;
```

• Ou:

```
minhasContas[1] = newConta();
minhasContas[1].saldo = 3000.0;
```



Populando array

```
public static void main(String args[]) {
int[] idades = new int[10];
for (int i = 0; i < 10; i++) {
idades[i] = i * 10;
for (int i = 0; i < 10; i++) {
  System.out.println(idades[i]); }
```



 Mas se você não souber o tamanho exato do seu array?

```
void imprimeArray(int[] array) {
for (int i = 0; i < array.length; i++) {
   System.out.println(array[i]);
  }
}</pre>
```



ArrayList Java

- Importar o ArrayList do java.util
- Novas propriedades;
- Dinâmico;



Exemplo

```
☐ import java.util.ArrayList;
  public class Casa {
      private String cor;
      ArrayList<Porta> portas;
      public ArrayList<Porta> getPortas() {
          return portas;
      public void setPortas(ArrayList<Porta> portas) {
          this.portas = portas;
      public String getCor() {
          return this.cor;
public void setCor(String novaCor) {
          this.cor = novaCor;
```



Métodos set do Array

- Temos duas opções para fazer o set do seu ArrayList, seguindo o exemplo anterior, temos:
 - setPorta(Porta porta);
 - setPortas(ArrayList<Porta> portas);



Exemplo

```
import java.util.ArrayList;
  public class Casa {
      private String cor;
      ArrayList<Porta> portas;
      public Casa() {
          this.portas = new ArrayList<Porta>();
      public void setPorta(Porta porta) {
          this.portas.add(porta);
```



Atividade 1

- Crie uma classe com o nome Aluno, que vão possuir os seguintes atributos:
 - Nome;
 - Nota;
- Depois crie alguns objetos, armazenandoos em um vetor de 30 posições (Não precisa criar 30 objetos);
- Crie uma estrutura de repetição para imprimir o nome e as notas de todos alunos armazenados neste array.



Atividade 2

- 1 De forma incremental, traduza o seguinte conjunto de classes em um programa Java. Importante: Não são permitidas chamadas a System.in, System.out ou similares de dentro das classes criadas (6,0).
- Classe: Porta Atributos: aberta, cor, dimensaoX, dimensaoY, dimensaoZ Métodos: void abre(), void fecha(), void pinta(String s), boolean estaAberta() Para testar, crie uma porta, abra e feche a mesma, pinte-a de diversas cores, altere suas dimensões e use o método estaAberta para verificar se ela está aberta.
- Classe: Casa Atributos: cor, e quantas portas ele queira inserir na casa Método: void pinta(String s), int quantasPortasEstaoAbertas(), int totalDePortas(), void cadastraPortas(Porta porta).
- Para testar, crie uma casa, pinte-a e cadastre as portas(sempre pergunte se ele deseja cadastrar mais portas através do JOptionPane, recebendo sim ou não). As portas cadastradas na ordem par, feche-as e as de ordem ímpar abra-as. Utilize o método quantasPortasEstaoAbertas para imprimir o número de portas abertas.
- Classe: Edificio Atributos: cor, nome de cada andar quantos o usuário queira, e quantas portas ele queira inserir no edifício. Métodos: void pinta(String s), int quantasPortasEstaoAbertas(), void cadastraPortas (Porta p), int totalDePortas(), void adicionarAndar(), int totalDeAndares() Para testar, crie um edifício, pinte-o. Crie seis portas e coloque-as no edifício através do método adicionaPorta, abra e feche-as como desejar. Utilize o método quantasPortasEstaoAbertas para imprimir o número de portas abertas e o método totalDePortas para imprimir o total de portas em seu edifício. Cria alguns andares utilizado o método adicionarAndar e retorne o número total de andares utilizando o método totalDeAndares.
- O que você faria para otimizar o código acima? Crie um novo projeto com a nova proposta, copiando os códigos em comum e justificando sua resposta no verso desta prova.