

# Exemplos

## Usando coleções de objetos

# Exemplo 1

```
Produto p = new Produto();  
p.setNome("A");
```

```
List<Produto> list = new ArrayList<Produto>();  
list.add(p);  
p.getNome (); // o nome é A  
list.get(0).getNome(); // o nome é A
```

```
// alterando o produto.  
p.setNome ("B");  
p.getNome (); // o nome é B  
list.get(0).getNome() // o nome é B
```

```
// alterando o produto diretamente da lista  
list.get(0).setNome ("C");  
p.getNome (); // o nome é C  
list.get(0).getNome() // o nome é C
```

# Exemplo 2

```
import java.util.ArrayList;
import java.util.List;
Import Pessoa;
public class TesteList {
    public static void main(String[] args) {
        //Declaração da classe genérica List,implementando uma lista de Pessoa
        List<Pessoa> listaPes = new ArrayList<Pessoa>();
        //declaracao e inicializacao dos objetos do tipo Pessoa
        Pessoa p1 = new Pessoa();
        p1.setNome("Joao");
        p1.setAltura(1.7f);
        p1.setIdade(25);
        p1.setSexo('M');
        Pessoa p2 = new Pessoa();
        p2.setNome("Maria");
        p2.setAltura(1.6f);
        p2.setIdade(25);
        p2.setSexo('F');
```

## Exemplo 2(cont.)

//Inserindo os objetos do tipo Pessoa na lista

```
listaPes.add(p1);
```

```
listaPes.add(p2);
```

//acessando e exibindo os valores do ArrayList

//o trecho de código listaPes.get(0) dá acesso ao objeto na posição 0

```
System.out.println(listaPes.get(0).getNome());
```

```
System.out.println(listaPes.get(1).getNome());
```

```
}
```

```
}
```

# Acessando e exibindo os valores do ArrayList

```
//utilizando for e o método size()
    for(int i=0; i<listaPes.size();i++){
        System.out.println(listaPes.get(i).getNome());
    }
```

```
//utilizando for para lista
    for(Pessoa p:listaPes){
        //a cada iteração, o compilador faz o objeto
        //p apontar para o próximo elemento
        System.out.println(p.getNome());
    }
```

# Acessando e exibindo os valores do ArrayList (2)

```
//Declarando um objeto do tipo Iterator para uma lista de Pessoa  
//Inicializando com o metodo do ArrayList que retorna o seu iterator  
Iterator<Pessoa> it = listaPes.iterator();
```

```
//percorrendo o ArrayList  
//it.hasNext() retorna true se a lista ainda nao  
//acabou de ser percorrida  
while(it.hasNext()){
```

```
//it.next() retorna um apontador para o objeto seguinte da lista  
System.out.println(it.next().getNome());
```

# Sobrescrevendo *equals* existente

```
@Override
public boolean equals(Object obj) {
    if(((Pessoa)obj).getNome().equals(this.getNome())){
        return true;
    }else{
        return false;
    }
}
```

# Usando novo *equals*

```
Import Pessoa;
public class TestePessoa {
    public static void main(String[] args) {
        Pessoa p1 = new Pessoa();
        p1.setNome("Pedro");
        Pessoa p2 = new Pessoa();
        p2.setNome("Pedro");
        if(p1.equals(p2)){
            System.out.println("p1 é igual a p2");
        }else{
            System.out.println("p1 é diferente de p2");
        }
    }
}
```