

Programação Orientada a Objetos

Prof. Paulo Henrique Pisani

Prof. Saul de Castro Leite

<http://professor.ufabc.edu.br/~paulo.pisani/>

30 min

Exercício 1 - Pilha - clone()

- Implemente o método **clone()** na classe **PilhaE** (cópia profunda);
- Utilize o projeto **projPilhaClone**, disponível no site da disciplina;
- A classe **Eletronico** é abstrata e possui o método “**public String getSerial()**” (observe que este método **NÃO** é abstrato;

Não altere o arquivo Principal.java

Exercício 1 - Pilha - clone()

- A saída do programa deverá ser:

```
Item P1: Notebook-0000-25  
Item P1: Notebook-1234567890-2  
Item P1: Tablet-1234  
  
Item P2: NotebookGrande-ASD-25  
Item P2: NotebookGrande-QWE-25  
Item P2: Tablet-ABCD  
Item P2: Notebook-0000-25  
Item P2: Notebook-5678-2  
Item P2: Tablet-1234
```

20 min

Exercício 2 - Pilha genérica

- Utilize o projeto **projPilhaGenerica**, disponível no site da disciplina;
- A versão atual da classe **PilhaGenerica** (neste projeto) armazena elementos da classe **Object**;
- Altere a **PilhaGenerica** de forma que seja possível especificar o tipo de objeto que é armazenado (use **Generics**);
- **Faça as alterações necessárias no método main** (arquivo **PrincipalGenerics.java**) para especificar o tipo de dados armazenado na **PilhaGenerica**.

Exercício 2 - Pilha genérica

- A saída do programa deverá ser:

```
Item P1: Notebook-0000-25  
Item P1: Notebook-5678-2  
Item P1: Tablet-1234
```

```
Item P2: GHI  
Item P2: DEF  
Item P2: ABC  
Pilha esta vazia!
```

- Não deve aparecer a seguinte mensagem na compilação:

```
Note: estrutura\PilhaGenerica.java uses unchecked or unsafe operations.  
Note: Recompile with -Xlint:unchecked for details.
```

Exercício 3 - Pilha de eletrônicos

- Implemente a classe `PilhaGenericaEletronicos` que estende `PilhaGenerica` (que fizemos no exercício 2). Essa nova classe permite que apenas eletrônicos (e suas subclasses) sejam passados no parâmetro de tipo;
- Use o mesmo projeto do exercício anterior e faça as adaptações necessárias (por exemplo, instanciar `PilhaGenericaEletronicos` ao invés de `PilhaGenerica` no `main`);
- Observe que a segunda pilha (que armazena Strings) precisará ser removida, pois String não é subclasse de `Eletronico`.

10 min

Exercício 4 - Pilha recurso (extra)

- O diretor de uma empresa percebeu que seus funcionários sempre aplicavam a **PilhaE** e, após o seu uso, esqueciam essas pilhas com elementos empilhados;
- Cansado disso, ele pediu para um aluno da disciplina de POO criar uma classe **PilhaERecurso**, que estende PilhaE e pode ser usada com o *try with resources*;
- Utilize o projeto **projPilhaRecurso**, disponível no site da disciplina.

Exercício 4 - Pilha recurso

- A saída do programa deverá ser:

```
Item P1: NotebookGrande-1111-25  
Item P1: NotebookGrande-0000-25  
Item P1: Tablet-ABCD  
Pilha foi esvaziada! Tinha 3 itens.
```


Exercício 5

- Repetir os exercícios 2, 3 e 4 para a Fila.