

NOTA: Esta ficha está dividida em duas partes (Parte I e Parte II), pretende-se que a primeira parte seja seguida com os slides da aula teórico-prática. Na segunda parte pretende-se que o aluno consiga realizar os exercícios pondo em prática a matéria abordada nos slides e praticada na Parte I.

Parte I

Exercício 1

Testar a classe `TwoTypePair` nos *slides 26 e 27* da **Aula 01** e observar o resultado.

Exercício 2

Testar a classe `Pair` no *slide 32* da **Aula 01**.

Exercício 3

Alterar a classe `UnorderedPair` no *slide 37* da **Aula 01** de forma a que seja possível efetuar duas operações:

- Obter o valor do primeiro e segundo elemento (2 métodos).
- Verificar se os dois elementos introduzidos são iguais.

Exercício 4

Teste o código apresentado no *slide 40* da **Aula 01** de forma a obter a mensagem de erro na compilação. Consegue perceber o porquê do erro?

Exercício 5

Apresente uma solução para o problema apresentado no *slide 44* da **Aula 01**.

Parte II

Exercício 1

Preencha os espaços em branco:

- A declaração `Store<T>` é ().
- `Store` é uma (), e `T` é o ().
- `Store<String>` é ().
- O uso de um tipo parametrizado é conhecido como ().

Exercício 2

Considere os seguintes fragmentos de código Java (as primeiras três linhas são iguais em todos; apenas a última linha apresenta diferenças). Para cada um dos fragmentos, o código compila corretamente? Caso compile, é executado sem erros, ou existe alguma exceção?

```
Point[] a = new Point[10];
```

```
Object[] b;  
b = a;  
b[0] = new Point(10,20);  
  
Point[] a = new Point[10];  
Object[] b;  
b = a;  
b[0] = "Magical Mystery Tour";  
  
Point[] a = new Point[10];  
Object[] b;  
b = a;  
a[0] = "Magical Mystery Tour";
```

Exercício 3

O que acontece se escrevermos código análogo ao da 2ª questão, mas que faça uso de um `ArrayList`? Por exemplo:

```
ArrayList<Point> a = new ArrayList<Point>();  
ArrayList<Object> b;  
b = a;  
b.add(new Point(10,20));
```

Exercício 4

Desenvolver uma aplicação que ordene uma lista de strings pré-definidas baseadas no tamanho da string. Usar o método `Collections.sort`. Atenção, deverá fazer uso de *Generics*.

Exercício 5

Escrever uma classe que haja como uma livraria para os seguintes tipos de média: livro, vídeo e CD de música. Atenção, deverá fazer uso de *Generics*. Adicione APIs adicionais para armazenar e obter média.