

Fundamentos de Java

Exercícios Propostos

Coleções de Dados



1 Exercício

Crie uma aplicação que simula jogadas feitas em um tabuleiro de jogo da velha. O primeiro passo é construir o tabuleiro, que deve ser definido como um *array* de duas dimensões (matriz), onde cada elemento é do tipo char. Depois o tabuleiro deve ser limpo. Para isto, atribua espaços em branco às posições.

Com o tabuleiro criado, é necessário definir as jogadas. As jogadas devem ser armazenadas numa coleção, de forma que a ordem de inserção das mesmas na coleção é a ordem que será utilizada para executá-las na sequência. Portanto escolha uma coleção que garanta esta regra.

Cada item da coleção de jogadas é um *array* de três posições. A primeira indica em qual linha do tabuleiro deve ser efetuada a jogada, enquanto a segunda indica a coluna. Como o tabuleiro do jogo da velha possui tamanho 3x3, estes valores podem ir de 0 a 2. A terceira posição do *array* de jogadas é o elemento que deve ser inserido no tabuleiro, que pode ser 'X' ou '0'.

Depois de montar a lista de jogadas, escreva um *loop* que itera sobre as jogadas e as realiza no tabuleiro, marcando cada elemento na posição especificada pela jogada. Por fim, escreva outro *loop* que imprime o tabuleiro com as jogadas realizadas.

Como sugestão, tente chegar ao resultado mostrado abaixo. A matriz da esquerda mostra o tabuleiro resultante, enquanto a da direita mostra a ordem das jogadas realizadas:

		0
0	X	X
X		X

		4
6	1	5
3		2

2 Exercício

Crie uma classe Produto com dois atributos: nome (String) e valor (double). Implemente a interface Comparable de forma que os produtos possam ser ordenados em ordem crescente de valor quando adicionados a um Set. Sobrescreva também os métodos equals() e hashCode(), considerando que produtos iguais são produtos que possuem o mesmo nome. E por último sobrescreva também o método toString(), para mostrar uma representação amigável do produto quando ele for impresso no console.

Na sequência crie uma classe Produtos, responsável por armazenar os produtos criados. Esta classe tem um atributo produtos, do tipo Collection<Produto>, e os métodos adicionar(), que adiciona um produto à coleção, e imprimir(), que imprime todos os produtos.



Crie uma aplicação que cria os seguintes produtos:

Nome	Valor
Laranja	2,50
Laranja	2,70
Maçã	1,45
Mamão	4,95
Limão	2,30

Experimente adicionar os produtos acima a coleções de diversos tipos, como ArrayList, HashSet e TreeSet, e imprima os resultados. Lembre-se que todos estes tipos podem ser atribuídos ao atributo produtos, pois todos são do tipo Collection. Procure perceber o que acontece com os elementos quando você muda o tipo de coleção na qual o produto está inserido, com relação à duplicidade de elementos e ordenação.

3 Exercício

Implemente a classe Colecao e duas subclasses: Pilha e Fila. Uma coleção tem um *array* de dados que fazem parte da coleção.

Tanto a pilha como a fila são coleções. A diferença entre elas está na disciplina de acesso. Na pilha, o último elemento inserido é o primeiro a ser removido (como numa pilha de pratos). Na fila, o primeiro elemento inserido é o primeiro a ser removido (como numa fila de banco).

Os métodos da classe Colecao responsáveis por estas operações são:

```
void inserirItem(Object item)
Object removerItem()
```

Crie um método que recebe uma coleção, adiciona alguns elementos e remove estes mesmos elementos. Imprima os elementos removidos e veja a diferença no resultado.

4 Exercício

Crie a classe Figura que representa figuras geométricas, representadas pelas classes Quadrado e Retangulo. Uma figura pode ter sua área calculada a partir do método calcularArea(), que retorna a área calculada da figura em forma de um double.



Crie também a classe FiguraComplexa. Uma figura complexa é também uma figura, mas a diferença é que ela é composta por várias figuras (quadrados, retângulos ou até outras figuras complexas). Para calcular a área de uma figura complexa, basta somar a área de todas as figuras que a compõem.

Para executar a aplicação, crie a classe Calculador, que é responsável por criar uma figura complexa e calcular a sua área. Esta figura deve ser composta por:

- 1 quadrado com 3 de lado
- 1 quadrado com 10 de lado
- 1 retângulo com lados 2 e 7
- 1 retângulo com lados 5 e 3

Dica: Perceba a diferença entre uma classe ser uma figura e ter uma ou mais figuras. A primeira relação é de herança, enquanto a segunda implica em uma composição.