



Collections - exercícios

Exercício 1:

1. Criar um dicionário que representa os primeiros 5 números da "Loteria dos Sonhos", junto com seu significado. Para isso, o dicionário deverá ter como chave Integer e como valor uma String.

Números da Loteria dos Sonhos:

0 → Ovos
1 → Água
2 → Escopeta
3 → Cavalo
4 → Dentista
5 → Fogo

2. Criar um dicionário que represente os apelidos que uso para chamar os meus amigos. Para isso, o dicionário deverá ter como chave String e como valor uma lista de Strings.

João → Juan, Fissura, Maromba
Miguel → Night Watch, Bruce Wayne, Tampinha
Maria → Wonder Woman, Mary, Marilene
Lucas → Lukinha, Jorge, George

Em seguida crie uma classe Exercício 1 com seu método main, imprimir no Main todas as chaves junto com seus valores associados. Para isso, utilizar o foreach com as chaves.





Collections - exercícios

Exercício 2:

3. Definir e inicializar uma nova lista (list) vazia. Adicionar os seguintes elementos à lista vazia: 1, 5, 5, 6, 7, 8, 8, 8.

Imprimir o resultado na tela.

4. Definir e inicializar um novo conjunto (set) vazio. Adicionar os seguintes elementos ao conjunto vazio: 1, 5, 5, 6, 7, 8, 8, 8.

Imprimir o resultado na tela. Qual é a diferença entre este exercício e o anterior?

Em seguida crie uma classe Exercício 2 com seu método main, imprimir no Main todos seus valores. Para isso, utilizar o foreach percorrendo os valores

Exercício 3:

Criar uma nova classe chamada Prova. Definir e implementar o seguinte método:

somaTotal(conjuntoDeInteiros: MutableSet<Int>)

O método deve percorrer o conjunto de inteiros, ir acumulando o total da soma dos valores e, no final, imprimir na tela o valor da soma total.

Em seguida crie uma classe Exercício 3 com seu método main, imprimir no Main todos valores associados. Para isso, utilizar o foreach.



Collections - exercícios

Exercício 4:

SaveTheRoupa S.A.

A SaveTheRoupa S.A. é uma empresa que deseja implementar um sistema informatizado de guarda-volumes de nível mundial. O sistema permite que uma pessoa guarde seus pertences no guarda-volumes e depois os retire de maneira simples, bastando apresentar o número de identificação recebido.

A classe **Pertence** é representados no sistema por algo abstrato chamado **peça**, que tem marca e modelo. Assim, se a pessoa perder o número, também poderia retirá-los com essas informações. No entanto, o método de retirada será projetado em outro momento.

1. Criar a classe abstrata **Peça**, que contenha as variáveis marca e modelo, ambas de tipo String e um método abstrato chamado retirada.
2. Criar a classe **GuardaVolumes**, que contenha como variável um dicionário e um contador que será utilizado como identificador. As chaves do dicionário serão Integer e, como valor, haverá uma lista de peças.
3. Criar o método **guardarPecas(MutableList<Peca> listaDePeca): Int** na Classe GuardaVolumes, que recebe uma lista de peças adiciona essa lista de peças em nosso dicionário e retorna o número de identificação, ou seja, a chave do dicionário onde guardamos as peças, que neste caso é o nosso contador.
4. Criar o método **mostrarPecas()** na Classe GuardaVolumes, que imprime na tela todas as peças que estão no guarda-volumes, junto com o número correspondente.



Collections - exercícios

5. Criar o método **mostrarPecas(numero: Int)** na Classe GuardaVolumes, que imprima as peças que estão associadas ao número recebido
6. Criar o método **devolverPecas(numero: Int)** na Classe GuardaVolumes, que remova a lista de peças que está associada ao número recebido.
7. Na classe Main, criar um cenário em que alguém guarda duas peças, recebe o código e depois retira suas peças.