

Exercícios - Listas

1. Fazer um programa para ler um número inteiro N e depois os dados (id, nome e salario) de N funcionários. Não deve haver repetição de id. Em seguida, efetuar o aumento de X por cento no salário de um determinado funcionário. Para isso, o programa deve ler um id e o valor X. Se o id informado não existir, mostrar uma mensagem e abortar a operação. Ao final, mostrar a listagem atualizada dos funcionários, conforme exemplos. Um salário só pode ser aumentado com base em uma operação de aumento por porcentagem dada.

```
Quantos funcionários serão cadastrados? 3

Funcionário #1:
Id: 333
Nome: Maria Brown
Salário: 4000.00
Funcionário #2:
Id: 536
Nome: Alex Grey
Salário: 3000.00
Funcionário #3:
Id: 772
Nome: Bob Green
Salário: 5000.00
Informe o Id do funcionário que terá aumento salarial : 536
Digite a porcentagem: 10.0
Lista de funcionários:
333, Maria Brown, 4000.00
536, Alex Grey, 3300.00
772, Bob Green, 5000.00
```

2. **Lista de Compras:** Escreva um programa que ajude alguém a fazer uma lista de compras. O programa deve permitir adicionar itens à lista, remover itens e exibir a lista completa de compras.
3. **Contagem de Ocorrências:** Escreva um programa que receba uma lista de strings e conte quantas vezes cada string aparece na lista. Em seguida, exiba cada string junto com sua contagem de ocorrências.

Desafios

1. Fazer um programa que simula uma lista de tarefas. O programa deve permitir adicionar, remover e exibir tarefas, bem como marcar tarefas como concluídas. As tarefas concluídas devem ser removidas da lista e inseridas em outra (lista de tarefas concluídas)
2. Criar um programa que simula um sistema simples de gerenciamento de livros em uma biblioteca. O programa permitirá adicionar livros, buscar livros pelo título e exibir a lista de todos os livros na biblioteca. Veja o método **equalsIgnoreCase** para Strings.
3. Crie um programa que simula um jogo de adivinhação de números. O programa gerará um número aleatório (1 a 200) que o jogador deve adivinhar. O jogador terá várias tentativas para acertar o número, e o programa deve fornecer dicas para indicar se o palpite do jogador é muito alto ou muito baixo. Dica: **Veja o método random.nextInt**