

## Lista de Exercícios

### ArrayList

- 1) (0,5) Escreva um programa(método main) que recebe diversos números inteiros positivos. Todos os números deverão ser armazenados em um ArrayList tipado para Integer. O programa deverá parar quando o número for negativo. Por fim, o programa deverá imprimir a quantidade de números que foram fornecidos, bem como a soma desses números.  
(Obrigatório o uso de ArrayList)

Entrada	Saída
1 2 33 1 -1	4 soma = 37
1 -1	1 soma = 1

- 2) (0,75) Escreva um programa(método main) que recebe diversos nomes de pessoas. Todos os nomes deverão ser armazenados em um ArrayList tipado para Strings. O programa deverá parar quando o texto digitado for igual a "FIM". O programa deverá mostrar os nomes cadastrados em ordem inversa a qual foram cadastrados, ou seja, do último para o primeiro.

Entrada	Saída
Ana Maria José Pedro Paulo FIM	Paulo Pedro José Maria Ana
Ana Paula FIM	Paula Ana

Lembrete: para comparar dois textos utilize o método equals ou equalsIgnoreCase da classe String.

- 3) (0,75) Crie um programa em Java que utilize a classe ArrayList para manipular uma lista de números inteiros. O programa deve:
- Solicitar ao usuário a digitação de um número inteiro n, que corresponde a quantidade de números a serem inseridos.
  - Solicitar ao usuário que insira os **n números inteiros**, adicionando-os a uma lista (ArrayList).
  - Exibir os números da lista na ordem em que foram inseridos.
  - Calcular e exibir a soma de todos os números da lista.
  - Encontrar e exibir o maior número da lista.
  - Remover todos os números ímpares da lista e exibir a lista final contendo apenas números pares.

(Obrigatório o uso de ArrayList)

Entrada	Saída
4 1 2 3 4	1 2 3 4 10 2 4

- 4) (0,75) Crie um programa em Java que gerencie uma lista de convidados para uma festa. O programa deve:
- Permitir ao usuário adicionar nomes à lista de convidados. A adição deve ser encerrada quando o usuário digitar a palavra **"fim"**.
  - Antes de adicionar um nome, verificar se ele já está na lista. Se já estiver, exibir a mensagem "convidado existe" informando que o convidado já foi adicionado.
  - Após encerrar a adição, exibir todos os nomes da lista.
  - Permitir que o usuário pesquise por um nome na lista de convidados e informe se ele está ou não na lista.

(Obrigatório o uso de ArrayList)

Entrada	Saída
José José Pedro Manoel Pedro fim Ana	convidado existe convidado existe José-Pedro-Manoel Não existe convidado chamado Ana

- 5) (0,75) Escreva um programa que permita cadastrar com 5 nomes em um ArrayList tipado como String. Então o programa deverá imprimir os nomes enumerados com valores entre 1 e 5, exemplo:

Qual dos nomes abaixo você deseja excluir da lista?

- Lucas Alvarenga
- Carla Fragoso

3. João Sttifert
4. Yun Haito
5. Felicia Brum

Por fim, o usuário deverá informar um número entre 1 e 5. Feita a escolha do usuário pelo número correspondente ao nome, o programa deverá excluir o respectivo nome e mostrar na tela os quatro nomes que restaram cadastrados na lista

Entrada do usuário	Saída do sistema
	Digite Cinco números:
Lucas Alvarenga Carla Fragoso João Sttifert Yun Haito Felicia Brum	
	Qual dos nomes abaixo você deseja excluir da lista? 1. Lucas Alvarenga 2. Carla Fragoso 3. João Sttifert 4. Yun Haito 5. Felicia Brum Informe o número correspondente ao nome a ser excluído:
2	
	Resultado: 1. Lucas Alvarenga 2. João Sttifert 3. Yun Haito 4. Felicia Brum

- 6) (2,5) Crie um programa em Java que seja capaz de cadastrar contatos. Para tal, é necessário criar um ArrayList tipado para a classe Contato, que deverá seguir a estrutura abaixo:

A classe Contato conterá os seguintes atributos:

Nome (String)  
Telefone (String)  
Celular (String)  
Email (String)

Os atributos da classe Contato deverão estar encapsulados

A classe Main deverá criar uma lista com base na classe Contato, faça:

```
ArrayList<Contato> listaContatos = new ArrayList<Contato>();
```

Então apresentar o menu abaixo, permitindo ao usuário cadastrar novos contatos e listar os contatos já cadastrados.

Sugestão de Menu:

1. Inserir Contato
2. Listar Contatos
3. Sair

7) (3,0) Crie um programa em Java para gerenciar uma lista de clientes. Cada cliente deve possuir as seguintes informações encapsuladas:

- Nome (String)
- CPF (String)
- Idade (int)

O programa deve:

- a) Permitir que o usuário adicione novos clientes. A entrada de dados deve parar quando o usuário digitar "**fim**" como nome.
- b) Exibir a lista de todos os clientes cadastrados.
- c) Permitir que o usuário pesquise um cliente pelo CPF. Se o cliente for encontrado, exibir todos os seus dados; caso contrário, informar que o cliente não está na lista.
- d) Remover um cliente da lista ao informar seu CPF. Após a remoção, exibir a lista atualizada.
- e) Remover todos os clientes da lista.

Sugestão de Menu:

1. Inserir novo Cliente
2. Listar todos os clientes
3. Buscar Cliente pelo CPF
4. Remover cliente pelo CPF
5. Apagar Lista
6. Sair