

Obs.: Em todos os programas, não se esqueça de exibir mensagens para criar uma interface amigável para o usuário.

1) Escreva em Java a classe Contador, que encapsule um valor usado para contagem de eventos (incrementador). A classe deve oferecer métodos que devem:

- a) Zerar;
- b) Incrementar;
- c) Retornar o valor do contador.

Implemente o programa principal que irá instanciar um objeto da classe Contador. O programa deverá solicitar ao usuário qual é a operação que será realizada: 1) Incrementar contador; 2) Zerar Contador; 3) Mostrar o valor do contador.

2) Construa a classe Livros em Java, que obedeça à descrição abaixo:

Livro
- titulo: String
- qtdPaginas: Integer
- paginasLidas: Integer
+ Livro()
+ Livro(nome: String, qtdPaginas: Integer)
+ verificarProgresso(): void

- Crie os métodos get e set para cada um dos atributos.
- Crie ainda o método *verificarProgresso* que deverá calcular a porcentagem de leitura do livro até o momento. Para isso, deverá usar os valores dos atributos *qtdPaginas* e *paginasLidas*, através da formula: **porcentagem = paginasLidas * 100 / qtdPaginas**. O valor da porcentagem deverá ser mostrado na tela conforme a mensagem “Você já leu X por cento do livro”, onde o valor de X é o valor calculado pela fórmula apresentada anteriormente.

Crie um programa principal que instancie um objeto da Classe Livro. Em seguida, mostre os dados do livro cadastrado e o progresso de leitura. Altere o título e a quantidade de páginas lidas. Apresente novamente os dados do livro e o progresso de leitura.

3) Utilizando o paradigma Orientado a Objetos na linguagem Java, implemente a classe Universidade que tem as seguintes características:

- **Atributo:** - nome da universidade
- **Métodos:** - Retornar o nome da Universidade;
- Construtor;

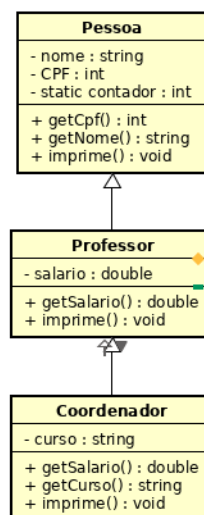
Em seguida, implemente a classe Pessoa. Esta classe deverá contemplar:

- **Atributos:** dia, mês e ano de nascimento, idade, nome da pessoa e a universidade em que ela trabalha (objeto da Classe Universidade).
- **Métodos:**
 - Construtor;
 - Calcular a idade da pessoa (*obs.: considere apenas o ano de nascimento para o cálculo da idade!*)
 - Retornar o valor da idade;
 - Retornar o nome da pessoa;
 - Retornar o nome da Universidade em que a pessoa trabalha.

Por fim, o programa principal deverá:

- Armazenar em um ArrayList as informações de diversas pessoas.
- Apresentar um relatório completo das pessoas cadastradas, constando todos os dados, inclusive idade e o nome da Universidade em que cada um trabalhou.

4) Utilizando a linguagem Java, implemente um programa que irá cadastrar um ArrayList de professor e outro ArrayList de coordenador, conforme a estrutura descrita no Diagrama de Classes abaixo.



O programa principal deverá ter um menu para: **solicitar que o usuário digite as informações | exibir um relatório com todos os dados cadastrados de todos os Professores e Coordenadores | Sair**

5) Construa a classe Contato em Java, que obedeça à descrição abaixo:

- A classe deve possuir dois atributos privados: nome(do tipo String) e telefone (do tipo String)
- Crie um construtor para a classe que recebe 2 parâmetros, cada um representando um dos atributos da classe.
- Crie os métodos de acesso (get e set)para os atributos da classe

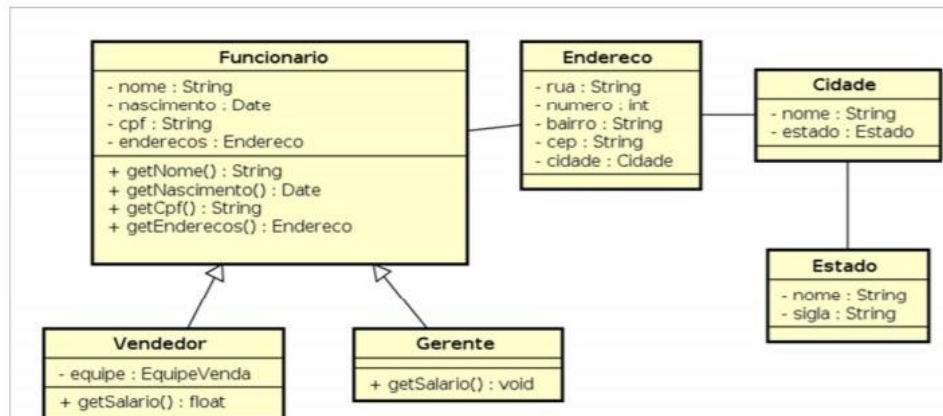
Contato
- nome: String - telefone: String
+ getNome(): String + setNome(String): void + getTelefone(): String + setTelefone(String): void +Contato(String, String)

Em seguida, construa a classe Agenda, a qual deve obedecer a descrição abaixo:

-Crie um método main na classe
-Crie um vetor de objetos Contato do tipo ArrayList (ex: ArrayList<Contato>agendaDeContatos)
-Exiba um menu na tela que ofereça as opções abaixo. O seu programa deve ficar executando enquanto o usuário não digitar a opção 4(sair). Solicite que o usuário informe a opção desejada:

- Na opção **Cadastrar Contato** (opção 1), solicite que o usuário digite o nome e o telefone do contato que ele deseja inserir. Crie um objeto do tipo Contato com as informações digitadas pelo usuário e insira o objeto no vetor. .
- Na opção **Buscar Contato**(opção 2), solicite que o usuário informe o nome do contato que ele deseja buscar. Percorra o vetor de objetos de forma a verificar se o contato que o usuário deseja buscar existe na coleção. Caso exista, escreva na tela o número de telefone do contato. Se não existir, exiba a mensagem “contato inexistente”.
- Na opção **Imprimir Agenda**(opção 3), escreva na tela as informações(nome e telefone)de todos os contatos que existem no vetor.
- Encerre a execução somente quando a opção **Sair**(opção 4)for digitada.

6) Implemente um programa para fazer um cadastro completo dos vendedores e do gerente de um Empresa de Equipamentos Tecnológicos - **utilizar ARRAYLIST**. Ao final do cadastro, o programa deverá apresentar um relatório completo de todas as informações dos funcionários.



7) Implemente um programa para cadastrar as pessoas (usuário ou pessoa jurídica) que trabalham na Faculdade de Computação **utilizar ARRAYLIST**. Ao final do cadastro, o programa deverá apresentar um relatório completo de todas as informações dos funcionários.

