

Revisão de POO em Java

Para cada exercício crie um novo projeto ou separe as classes por pacotes (packages). Crie testes automatizados para todas as classes e seus métodos. Tentar utilizar tratamento de exceções onde e se fizer sentido.

1. Criar a classe Pessoa com as seguintes características:

- atributos: idade e dia, mês e ano de nascimento, nome da pessoa
- métodos:
 - o calculaIdade(), que recebe a data atual em dias, mês e anos e calcula e armazena no atributo idade a idade atual da pessoa
 - o informaIdade(), que retorna o valor da idade
 - o informaNome(), que retorna o nome da pessoa
 - o ajustaDataDeNascimento(), que recebe dia, mês e ano de nascimento como parâmetros e preenche nos atributos correspondentes do objeto.
- Criar dois objetos da classe Pessoa, um representando Albert Einstein (nascido em 14/3/1879) e o outro representando Isaac Newton (nascido em 4/1/1643)
- Fazer uma classe principal que instancie os objetos, inicialize e mostre quais seriam as idades de Einstein e Newton caso estivessem vivos.

2. Implementar a classe PolReg, que define um polígono regular

- **Atributos:** número de lados, tamanho do lado
- **Métodos:** cálculo do perímetro, cálculo do ângulo interno e cálculo de área. Este último deve retornar o valor zero, dado que não é possível calcular a área de um polígono regular genérico
- Construtor que inicializa os valores dos atributos

Fazer a classe TrianguloEq derivada de PolReg, que:

- redefine/sobreescreve o método de cálculo de área para calcular e retornar a área do triângulo equilátero

Fazer a classe Quadrado, também derivada de PolReg, que:

- redefine/sobreescreve o método de cálculo de área para calcular e retornar a área do quadrado

Fazer um programa que:

- pergunte ao usuário que tipo de objeto ele quer criar (polígono regular, triângulo equilátero ou quadrado)
- pergunte número de lados (se for polígono regular) e tamanho dos lados
- instancie o objeto e mostre o valor do perímetro, ângulo interno e área calculados

3. Escrever a classe Pessoa com as seguintes características:

- atributos: nome e idade
- métodos: construtora para inicializar os parâmetros e mostraDados() que exibe os dados da pessoa no console na forma:

Nome da pessoa: xxx

Idade da pessoa: yyy

Escrever a classe Aluno, derivada de Pessoa, com as seguintes características:

- atributos: nome do curso que está cursando
- métodos: construtora para inicializar os atributos e redefinição do método mostraDados() para exibir as seguintes mensagens:

Nome do aluno: xxx

Idade do aluno: yyy

Curso do aluno: zzz

Elaborar um programa em Java que:

- declare uma referência para objeto da classe Pessoa
- pergunte ao usuário, via console, se ele deseja instanciar um aluno ou uma pessoa
- crie o objeto correspondente, referencie com a referência já criada e chame o método mostraDados() para exibir os dados da pessoa ou do aluno

4. Crie uma classe para representar uma pessoa, com os atributos privados de nome, data de nascimento e altura. Crie os métodos públicos necessários para sets e gets e também um método para imprimir todos dados de uma pessoa. Crie um método para calcular a idade da pessoa.

5. Crie uma classe Agenda que pode armazenar 10 pessoas e que seja capaz de realizar as seguintes operações:

```
void armazenaPessoa(String nome, int idade, float altura);
```

```
void removePessoa(String nome);
```

```
int buscaPessoa(String nome); // informa em que posição da agenda está a pessoa  
void imprimeAgenda(); // imprime os dados de todas as pessoas da agenda
```

```
void imprimePessoa(int index); // imprime os dados da pessoa que está na posição "i"  
da agenda.
```

6. Crie uma classe denominada Elevador para armazenar as informações de um elevador dentro de um prédio. A classe deve armazenar o andar atual (térreo = 0), total de andares no prédio (desconsiderando o térreo), capacidade do elevador e quantas pessoas estão presentes nele. A classe deve também disponibilizar os seguintes métodos:

Inicializa : que deve receber como parâmetros a capacidade do elevador e o total de andares no prédio (os elevadores sempre começam no térreo e vazio);

Entra : para acrescentar uma pessoa no elevador (só deve acrescentar se ainda houver espaço);

Sai : para remover uma pessoa do elevador (só deve remover se houver alguém dentro dele);

Sobe : para subir um andar (não deve subir se já estiver no último andar);

Desce : para descer um andar (não deve descer se já estiver no térreo); Encapsular todos os atributos da classe (criar os métodos set e get).

7. Crie uma classe Televisao e uma classe ControleRemoto que pode controlar o volume e trocar os canais da televisão.

O controle de volume permite:

- aumentar ou diminuir a potência do volume de som em uma unidade de cada vez;
- aumentar e diminuir o número do canal em uma unidade
- trocar para um canal indicado;
- consultar o valor do volume de som e o canal selecionado.