



Universidade Federal do Piauí - UFPI
Campus Senador Helvídio Nunes de Barros - Picos-PI
Sistemas de Informação - SI
Programação Orientada a Objeto

Lista 03 – Introdução a Orientação a Objetos - 23/11/2024

- Resolva todas as questões em Python.
- Objetivo desta lista é revisar conceitos de introdução a programação orientada a objetos e praticar a linguagem Python.
- Data da entrega: 10/12/2024 até as 13:59

- Para cada exercício crie um novo projeto.
1. 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 os dados de uma pessoa. Crie um método para calcular a idade da pessoa.
 2. Crie uma classe Agenda que pode armazenar até 10 pessoas (a classe Pessoa deve conter os atributos: nome e idade) e que seja capaz de realizar as seguintes operações:
 - a. `armazenaPessoa;` // Não permitir armazenar mais de 10 pessoas
 - b. `removePessoa;` // Pelo nome
 - c. `buscaPessoa;` // Busca pelo nome e imprime os dados da pessoa
 - d. `imprimeAgenda();` // imprime os dados de todas as pessoas da agenda
 3. 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:
 - a. 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);
 - b. Entra: para acrescentar uma pessoa no elevador (só deve acrescentar se ainda houver espaço);
 - c. Sai: para remover uma pessoa do elevador (só deve remover se houver alguém dentro dele);
 - d. Sobe: para subir um andar (não deve subir se já estiver no último andar);
 - e. Desce: para descer um andar (não deve descer se já estiver no térreo);
 - f. Encapsular todos os atributos da classe (atributos privados e criar os métodos set e get).
 - g. Criar um menu iterativo para controlar as operações do elevador.
 4. 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. A classe controle remoto deve possuir um menu iterativo para o usuário escolher as opções desejadas.