

Exercícios de revisão sobre POO

1) Crie uma classe para representar uma pessoa, com os atributos privados de nome, idade e altura. Crie os métodos públicos necessários para informar os valores e também um método para imprimir os dados de uma pessoa.

2) Crie um programa em que calcule o valor de juros pago em uma compra parcelada. O usuário deve informar o preço do produto a vista, o valor de cada parcela e a quantidade de parcelas. No final, o programa deve mostrar uma mensagem do tipo (sendo o total parcelado e o valor de juros calculados pelo programa):

Preço a vista: R\$ 100,00

Valor da parcela: R\$ 25,00

Qtd parcelas: 5

Total parcelado:

R\$ 125,00

Valor de Juros: R\$ 25,00

Utilize uma classe chamada `compraparcelsada`, que recebe os valores de preço a vista e parcelas em atributos encapsulados (`private`). Crie os métodos para leitura dos valores e para mostrar o valor do cálculo.

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, excluindo 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);
- `Finaliza`: terminar o programa

Encapsular todos os atributos da classe. Criar um `Loop` com um menu solicitando qual opção acima deve ser executada e, a cada repetição, informar os dados de andar e quantidade de pessoas.

4) Crie uma classe `Agenda` que pode armazenar 10 pessoas (criada no exercício 1) e seja capaz de realizar as seguintes operações através de um `Menu`:

- `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.

5) Crie uma classe `Televisão` 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;
- O controle de canal também permite aumentar e diminuir o número do canal em uma unidade, porém, também possibilita trocar para um canal indicado;
- Também devem existir métodos para consultar o valor do volume de som e o canal selecionado.
- O volume varia entre 0 e 100. Os canais variam entre 1 e 160. Não permita volumes ou canais que não existam.

6) Crie uma classe Calculadora que faça as quatro operações básicas (soma, subtração, multiplicação e divisão). Crie uma classe derivada CalculadoraCientifica que, além das operações básicas, faça extração de raízes e calcule a potência de um número elevado a outro.

7) Responda:

- O que é POO?
- O que é Objeto?
- O que é Classe?
- O que é Atributo?
- O que é Método?
- O que é Herança em POO?
- O que é abstração?
- O que é encapsulamento?
- Quando devo usar uma classe abstrata?
- Existe herança múltipla (de classes) em C#?