

Exercícios de Programação Orientada a Objetos

Classes e Objetos – Conceitos básicos

Atividades da disciplina teórica:

1 – Desenvolva um conjunto de classes para armazenar informações sobre veículos. Devem existir duas classes, sendo elas: Carro e Moto. Os atributos da classe Carro são:

- Nome, marca, motor, cor e quantidade de portas

Os atributos da classe Moto são:

- Nome, marca, cilindrada, cor e tipo (esportiva, trilha ou custom)

Crie um mecanismo para que um usuário possa preencher os valores de um carro e de uma moto, o programa deve criar um objeto da classe carro e um objeto da classe moto com os dados fornecidos pelo usuário.

O programa deve mostrar as informações da moto e do carro por meio de um método desenvolvido nessas classes.

2 – Ainda trabalhando no código da questão 1, adicione um parâmetro preço nas duas classes. Desenvolva um método em cada uma das classes para simular o valor total a ser pago mediante um financiamento. Supondo que uma loja permita as seguintes opções de financiamento com o devido aumento:

Carro:

- 24X → 30% a mais do valor total.
- 36X → 45% a mais do valor total.
- 48X → 60% a mais do valor total.

Moto:

- 24X → 35% a mais do valor total.
- 36X → 50% a mais do valor total.
- 48X → 70% a mais do valor total.

O método deve ser capaz de calcular o valor total e o valor de cada parcela de acordo com as opções escolhidas.

Crie um mecanismo para que um usuário possa escolher se quer financiar um carro ou uma moto, em seguida, o usuário deve informar os dados do veículo escolhido e a quantidade de parcelas. O programa deve mostrar as informações do veículo, bem como o valor total após o financiamento e o valor de cada parcela.

Atividades de laboratório:

3 – Crie duas classes chamadas: Triangulo e TrianguloInvertido. Essas duas classes possuem um parâmetro inteiro chamado qtdLinhas. Crie um mecanismo para que um usuário possa escolher se quer imprimir um triângulo ou um triângulo invertido, em seguida, ele deve informar a quantidade de linhas que serão utilizadas para formar esse triângulo. A classe deve ter um método para imprimir o triângulo com *. Por exemplo, supondo que um usuário escolheu um triângulo com 6 linhas, o programa deve mostrar o seguinte:

```
*
**
***
****
*****
*****
```

Como outro exemplo, se o usuário escolher um triângulo invertido com 4 linhas, o programa deve mostrar o seguinte:

```
****
***
**
*
```

4 – Crie um programa para cadastrar e mostrar informações de um aluno. A classe do aluno deve possuir os seguintes atributos:

- Nome, notaFinal e faltas.

Crie um método para instanciar todos os atributos baseados nos parâmetros do método.
Crie um método para gerar um relatório do aluno, mostrando suas informações, a situação (aprovado ou reprovado) e a justificativa em caso de reprovação.

Para ser aprovado, o aluno precisa ter, pelo menos, 60 pontos de nota final e não pode ter mais do que 25 faltas.

Por exemplo, se João tem 60 pontos e 13 faltas, o relatório deve ser o seguinte:

João está aprovado.

Se José possui 59 pontos e 10 faltas, o relatório deve ser o seguinte:

José está reprovado, pois não atingiu os 60 pontos.

Se Pedro possui 80 pontos e 26 faltas, o relatório deve ser o seguinte:

Pedro está reprovado, pois possui mais de 25 faltas.

Se Carlos possui 30 pontos e 30 faltas, o relatório deve ser o seguinte:

Carlos está reprovado, pois não atingiu os 60 pontos e possui mais de 25 faltas.