



INSTRUÇÃO PRÁTICA		PO-04
MÓDULO	PO3 - PRÁTICA ORIENTADA	
OBJETIVO DA ATIVIDADE		TEMPO
PRATICAR CONCEITOS DE CLASSES, OBJETOS, SOBRECARGA, HERANÇA E POLIMORFISMO EM JAVA		8h
DESCRIÇÃO		
<p>1.</p> <p>Crie uma classe chamada Calculadora que contenha métodos para realizar operações matemáticas básicas, como soma, subtração, multiplicação e divisão.</p> <p>Sobrecarregue os métodos para aceitar diferentes tipos de dados, como inteiros e float.</p> <p>Crie uma versão para receber listas de números (ArrayLists) e fazer operações em sequência.</p> <p>Certifique-se de fornecer exemplos de uso no método main.</p> <p>2.</p> <p>Considere um sistema de gerenciamento de pedidos em uma loja online.</p> <p>Crie uma classe chamada Pedido que represente um pedido de compra.</p> <p>A classe deve ter atributos como número do pedido, cpf do cliente e uma lista de itens do pedido.</p> <p>Implemente métodos para adicionar itens ao pedido, calcular o total do pedido e exibir as informações do pedido.</p> <p>Sobrecarregue o método de adicionar itens para ele receber uma lista de itens ao invés de um único.</p> <p>Sobrecarregue o método calcular o total do pedido para que ele informe o valor à vista (um percentual de desconto é passado como parâmetro) ou o valor total a prazo (o número de prestações e o juro é passado como parâmetro)</p> <p>Além disso, forneça exemplos no método main para demonstrar a</p>		

INSTRUÇÃO PRÁTICA	PO-04
<p>sobrecarga de métodos em diferentes situações, como um pedido com desconto e outro sem desconto.</p> <p>3.</p> <p>Crie uma hierarquia de classes para representar diferentes tipos de veículos, como Veiculo, Carro, e Moto. A classe base Veiculo deve conter atributos comuns a todos os veículos, como modelo, cor e ano.</p> <p>Crie métodos genéricos acelerar() que escreve a mensagem "Veículo acelerando" e ligar() que escreve "Veículo ligado" e parar() ("Veículo parado").</p> <p>Adicione métodos nas classes derivadas (Carro e Moto) que aproveitem o polimorfismo para realizar ações específicas, como ligar ("moto/carro ligado(a)" acelerar, e parar. Certifique-se de sobrescrever os métodos na classe Veiculo quando necessário.</p> <p>No método main, crie instâncias das classes derivadas e chame os métodos para demonstrar o polimorfismo em ação.</p> <p>4.</p> <p>Amplie o exemplo anterior introduzindo uma classe adicional chamada Garagem. A classe Garagem deve ter uma lista de veículos e métodos para adicionar (estacionar) e remover veículos. A garagem também deve ter um atributo indicando se possui ou não tomada de carga elétrica para veículos.</p> <p>Modifique sua classe Veículo para ele ter um flag indicando se é um veículo elétrico ou não.</p> <p>Crie o método estacionar() na classe veículo. Este método recebe uma garagem como parâmetro. O veículo deve ser incluído na lista de veículos daquela garagem.</p> <p>"Veículo estacionado!" ou "Carro/Moto estacionado(a)"</p> <p>Todavia, se a garagem possuir tomada de carga elétrica, e o veículo for do tipo elétrico, deve-se acrescentar a mensagem "Veículo carregando" ou "Carro/Moto carregando".</p> <p>No método main, crie instâncias das classes derivadas e chame os métodos para demonstrar o polimorfismo em ação.</p>	



INSTRUÇÃO PRÁTICA	PO-04
REFERÊNCIAS	
<p data-bbox="240 360 1347 434">Apostila do curso ALURA: https://www.caelum.com.br/apostila/apostila-java-web.pdf</p> <p data-bbox="240 524 1347 636">Java Básico e Orientado a Objeto: https://canal.cecierj.edu.br/012016/d7d8367338445d5a49b4d5a49f6ad2b9.pdf</p> <p data-bbox="240 725 1347 837">Java para Desenvolvimento WEB: https://docente.ifsc.edu.br/mello/livros/java/caelum-java-web-fj21-banco-de-dados-jdbc.pdf</p>	