

Trabalho obrigatório para prova A1

Valor: até 1.0 ponto da prova. Trabalho em dupla.

Entrega até às 08:30hs (manhã) e 19:30hs (noite) do dia da prova no laboratório com o projeto compactado com .ZIP (trabalhos em .RAR ou outros compactadores não serão aceitos).

Sejam as classes abaixo:

Classes	Atributos	Métodos
Lancha	. Marca: Texto . Modelo: Texto . NumeroPassageiros: inteiro . Preco: real Tipo : caractere . VelocidadeMaxima: inteiro . TipoCasco: Texto . Identificação: Texto . Banheiro : lógico . Motor1 : Motor	. Setters para todos os atributos . Getters para todos os atributos . Cadastrar (...) - Todos os atributos do objeto . Imprimir () - Exibir todos os atributos do objeto . ValorDesconto () - Retorna o valor da Lancha com um desconto fixo de 10% . EntradaDados() – Efetua a entrada de dados via teclado.
late	. Marca: Texto . Modelo: Texto . NumeroPassageiros: inteiro . NumeroTripulantes : inteiro . Preco: real . VelocidadeMaxima: inteiro . Identificação: Texto . NumeroCabines : inteiro . Piscina : lógico . Motor1 : Motor . Motor2 : Motor	. Setters para todos os atributos . Getters para todos os atributos . Cadastrar (...) - Todos os atributos do objeto . Imprimir () - Exibir todos os atributos do objeto . ValorDesconto () - Retorna o valor da late com um desconto fixo de 8% . EntradaDados() – Efetua a entrada de dados via teclado. . PessoasPorCabine() - Retorna o número de passageiros por cabines do late
JetSki	. Marca: Texto . Modelo: Texto . NumeroPassageiros: inteiro . Preco: real . Tipo : caractere . Reboque: Lógico . VelocidadeMaxima: inteiro . Identificação: Texto . TipoCasco: Texto . Motor1 : Motor	. Setters para todos os atributos . Getters para todos os atributos . Cadastrar (...) - Todos os atributos do objeto . Imprimir () - Exibir todos os atributos do objeto . EntradaDados() – Efetua a entrada de dados via teclado. . ValorDesconto () - Retorna o valor do JetSki com um desconto fixo de 15%
Motor	. Marca: Texto . Modelo: Texto . Potencia: inteiro . TipoCombustivel: Texto . RpmMaxima : inteiro . Rpm : inteiro . NúmeroCilindros : inteiro	. Setters para todos os atributos . Getters para todos os atributos . Cadastrar (...) - Todos os atributos do objeto . Imprimir () - Exibir todos os atributos do objeto . EntradaDados() – Efetua a entrada de dados via teclado. . Acelerar () - Aumenta as rotações (Rpm) em 100 unidades . Desacelerar () - diminui as rotações (Rpm) em 100 unidades

Baseado nessas classes pede-se:

1. Crie uma superclasse que seja capaz de atender às necessidades das subclasses **Lancha**, **JetSki** e **late**, com pelo menos 10 métodos construtores;
2. Criar as subclasses **Lancha**, **late** e **JetSki** utilizando os conceitos de herança a partir da superclasse criada e com o mínimo 10 métodos construtores (cada);
3. Cada subclasse deverá usar no mínimo 5 métodos construtores da superclasse;
4. Não se esqueça de reaproveitar, sempre que possível, métodos da superclasse estendendo suas ações;
5. Crie uma aplicação (em um pacote diferente das demais classes), que crie pelo menos 5 objetos **Lanchas**, 5 objetos **JetSkis** e 5 objetos **lates**, sendo que estes objetos deverão ser criados com o uso dos diferentes construtores disponibilizados por cada objeto. Todos os métodos disponibilizados pelas classes deverão ser utilizados na aplicação pelo menos uma vez.
6. Aplique as regras da boa prática de programação em java.

Obs.: Faça uma parte de cada vez.

Tente dividir o problema em partes menores para que a solução seja mais simples.