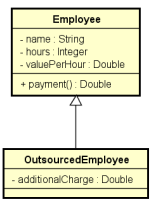
Atividade de POO - Data para entrega : 10/11/2023

Email : [rafael.fbatista@aluno.uepa.br](mailto:rafael.fbatista@aluno.uepa.br)

1 ) **Funcionários -** Uma empresa possui funcionários próprios e terceirizados. Para cada funcionário, deseja-se registrar nome, horas trabalhadas e valor por hora. Funcionários terceirizados possuem ainda uma despesa adicional. O pagamento dos funcionários corresponde ao valor da hora multiplicado pelas horas trabalhadas, sendo que os funcionários terceirizados ainda recebem um bônus correspondente a 110% de sua despesa adicional.

Fazer um programa para ler os dados de N funcionários (N fornecido pelo usuário) e armazená-los em um array. Depois de ler todos os dados, mostrar nome e pagamento de cada funcionário na mesma ordem em que foram digitados.

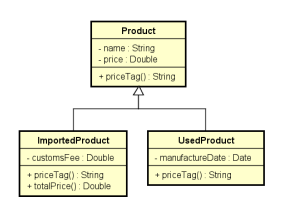
**Obs : Uso do super e Polimorfismo (Override)**



| **Exemplo de entrada** | **Exemplo de Saída** |
| --- | --- |
| Enter the number of employees: 3  Employee #1 data:  Outsourced (y/n)? n  Name: Alex  Hours: 50  Value per hour: 20.00  Employee #2 data:  Outsourced (y/n)? y  Name: Bob  Hours: 100  Value per hour: 15.00  Additional charge: 200.00  Employee #3 data:  Outsourced (y/n)? n  Name: Maria  Hours: 60  Value per hour: 20.00 | PAYMENTS:  Alex - $ 1000.00  Bob - $ 1720.00  Maria - $ 1200.00 |

2 ) **Produtos** - Fazer um programa para ler os dados de N produtos (N fornecido pelo usuário). Ao final, mostrar a etiqueta de preço de cada produto na mesma ordem em que foram digitados. Todo produto possui nome e preço. Produtos importados possuem uma taxa de alfândega, e produtos usados possuem data de fabricação. Estes dados específicos devem ser acrescentados na etiqueta de preço conforme exemplo (próxima página). Para produtos importados, a taxa de alfândega deve ser acrescentada ao preço final do produto.

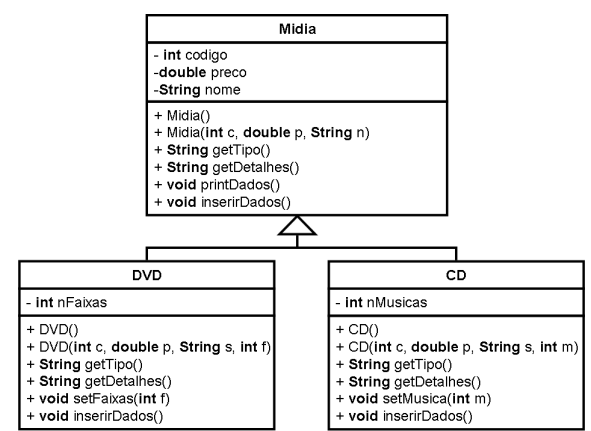
**Obs : Uso do Override e Classe SimpleDateFormat**

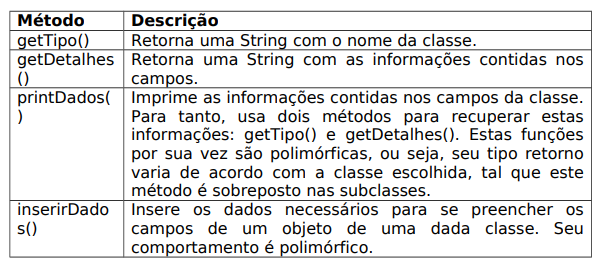


| **Exemplo de Entrada** | **Exemplo de Saída** |
| --- | --- |
| Enter the number of products: 3  Product #1 data:  Common, used or imported (c/u/i)? i  Name: Tablet  Price: 260.00  Customs fee: 20.00  Product #2 data:  Common, used or imported (c/u/i)? c  Name: Notebook  Price: 1100.00  Product #3 data:  Common, used or imported (c/u/i)? u  Name: Iphone  Price: 400.00  Manufacture date (DD/MM/YYYY): 15/03/2017 | PRICE TAGS:  Tablet $ 280.00 (Customs fee: $ 20.00)  Notebook $ 1100.00  Iphone (used) $ 400.00 (Manufacture date: 15/03/2017) |

3 ) **Loja de CD e DVD** - Uma loja que vende CD e DVDS deseja construir um cadastro com dos seus produtos.

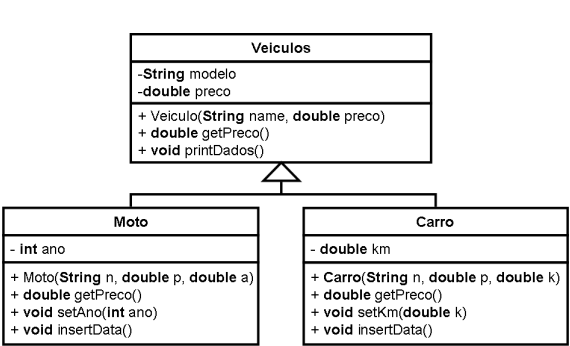
**Obs : Super e Uso de polimorfismo**





Além dos métodos descritos na Tabela, deverão ser criados os métodos get e set correspondentes para retornar e modificar o conteúdo dos campos, respectivamente, bem como os construtores com e sem parâmetros de cada classe. Criar um programa que simule o uso de um cadastro de CD e DVDs

4 ) **Controle de Veículos** - Simular a operação de um cadastro de veículos de uma revenda. Existem dois tipos de veículos: Carro, e Moto.

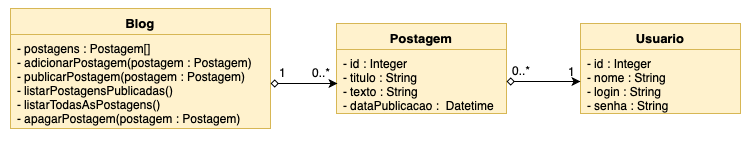


Obs 1.: O usuário decide se deseja inserir uma moto ou um carro. Depois insere as informações necessárias e finalmente imprime o relatório com as características de cada veículo contido no cadastro.

Obs 2.: Após o usuário inserir as informações deverá ser realizado um levantamento acerca das motos e carros. Primeiro será obtido o total dos preços dos veículos contidos no cadastro. Depois, é verificado e uma moto tiver no campo ano um valor maior ou igual que 2008 seu valor será reajustado em 10%. Se um carro tiver um valor no campo km um valor maior que 100000, então, seu valor será reduzido em 8%. Por fim, um novo cálculo acerca do total de preços é realizado e impresso.

5 ) **Blog -** Crie um programa para representar um blog. O blog contém várias postagens que são cadastradas por usuários autenticados. Siga o diagrama de classes abaixo para a implementação.

Obs : Menu de Opções e ArrayList (List)



Obs 1.: O blog contém uma lista de postagens

Obs 2.: O método “listarPostagensPublicadas()” deverá listar apenas as postagens cuja data de publicação já foi atingida. Do contrário, as postagens serão visíveis apenas pelo método “listarTodasAsPostagens()”

Obs 3.: Para criar uma postagem, o usuário deverá estar autenticado no sistema

Obs 4.: A Postagem deverá identificar o usuário que fez a sua criação.