

Exercícios Prático de Fixação - Atividade Avaliativa I

Condições de conclusão

Aberto: quarta-feira, 2 out. 2024, 23:00

Vencimento: terça-feira, 8 out. 2024, 22:00

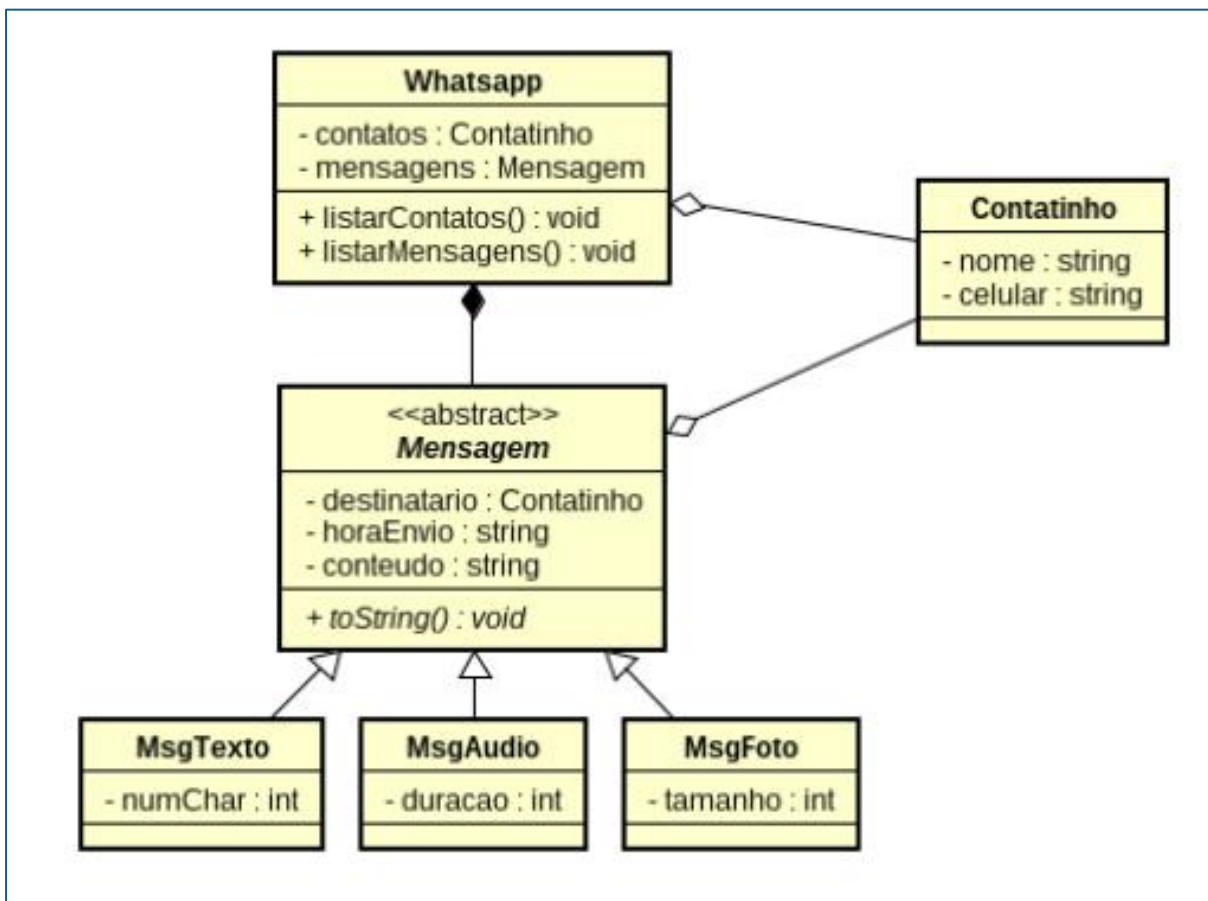
Formato de Entrega: Projeto hospedado em um repositório Git

Instruções:

- Criar um projeto único;
- Separar as implementações em pacotes; e
- Verificar nos diagramas possíveis correções de Design durante o processo de implementação.

Exercício 1 - Com base no diagrama de classes abaixo, implemente um projeto orientado a objetos contendo as abstrações representadas e considerando as seguintes instruções:

- **package:** edu.fatec.lp2.exercicio1 (organizar as classes e outros pacotes dentro deste)
- **contatos** e **mensagens** são Arrays inicializados via construtor;
- Todos os atributos são privados;
- A classe abstrata Mensagem deve possuir um método abstrato **Mensagem sendMessage(String);** e
- **Mensagem sendMessage(String)** deve retornar a sua própria instância.



Exercício 2 - Seu Astolfo recentemente adquiriu uma rede de supermercados e contratou a Você S.A., empresa especializada no desenvolvimento de soluções informatizadas, para implementar uma prova de conceito referente ao seu domínio de compras.

Para isso ele considerou as seguintes regras como sendo suas necessidades:

- **package:** edu.fatec.lp2.exercicio2 (organizar as classes e outros pacotes dentro deste)
- Deve haver uma abstração simplificada que represente seu Supermercado
- Deve haver uma abstração que represente seu produto, conforme diagrama abaixo
- A abstração produto terá seus próprios atributos e um único comportamento para retornar seu nome
- Deve haver uma abstração para representar os itens comprados pelo cliente
- Cada item de compra terá a quantidade adquirida e um desconto que pode variar de 0 ao descontoMaximo
- Deve haver a possibilidade de vincular ao item de compra o produto e o desconto
- Só será possível associar um único produto ao item de compra
- Deve haver uma ação no item de compra que possibilite calcular seu preço independente dos demais
- Deve haver uma abstração que represente a lista de compras e o mesmo deve possuir uma quantidade limite
- Deve haver uma ação na lista de compras que possibilite adicionar itens de compra até seu limite
- Deve haver uma ação na lista de compras que calcule o preço total da lista de compras
- As abstrações de item de compra e lista de compra deverão respeitar uma interface comum

Dadas as necessidades do Seu Astolfo e assumindo-se que todas as classes possuem métodos getter e setter (**usar Lombok ou gerá-las através da IDE**), vamos ajudar o Seu Astolfo a implementar as abstrações representadas no diagrama de classes abaixo e a lógica necessária para atender tais necessidades. Mãos à obra!

