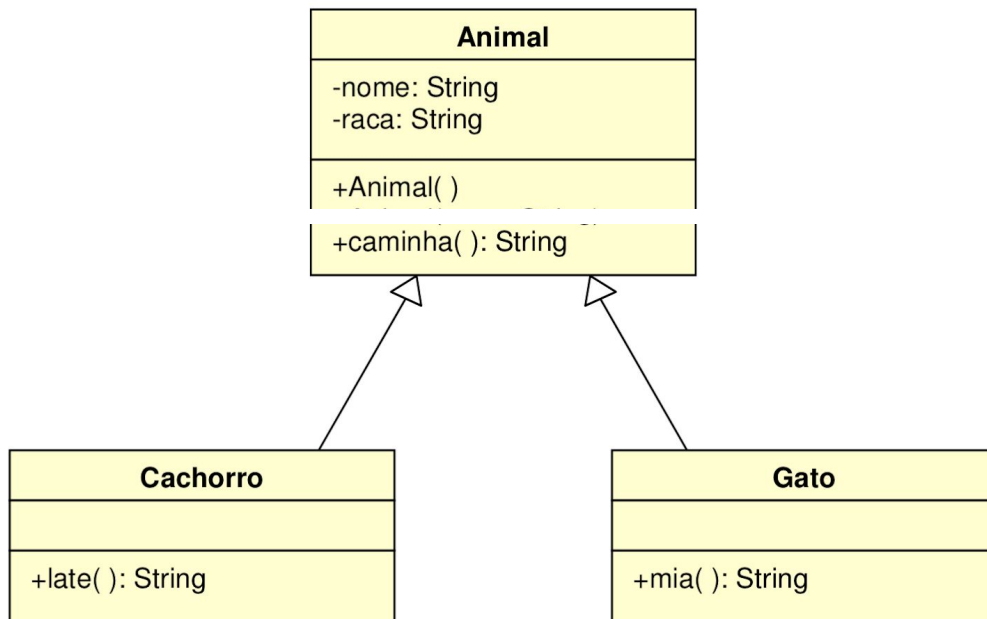


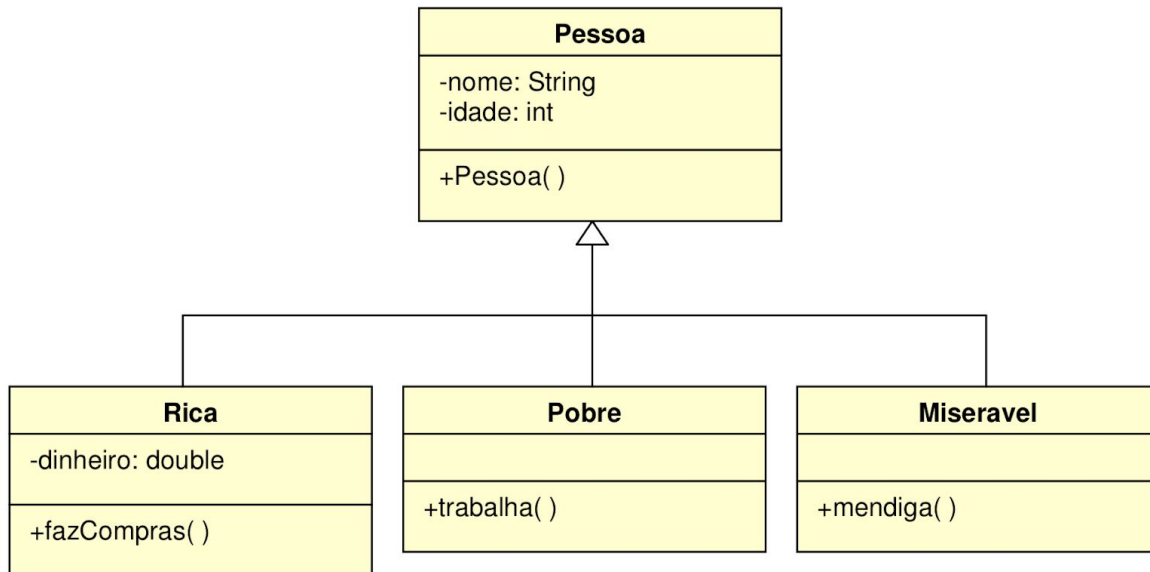
Lista de Exercícios

1. Analise o panorama a seguir, desenhe o diagrama de classes e a implemente. No presente projeto, identificou-se a classe `Funcionario` e a classe `Gerente`. `Funcionario` tem nome, id e salário como características. Além disso, `Gerente` tem nome, id, salário, gratificação e agência que administra como características.

- crie as classes acima e implemente seus construtores, get, sets e str.
- implemente um método chamado `exibeResponsabilidades()` que retorna uma lista de responsabilidades em cada uma das classes.
- crie a classe `Assistente`, que também é um funcionário, e possui uma vigência do contrato (período de contrato funcionário, dado que `Assistente` é uma modalidade temporária) como característica extra. Sobrescreva o método `exibeResponsabilidades()`.
- sabendo que os `Assistentes Técnicos` possuem um bônus salarial e que os `Assistentes Administrativos` possuem um turno (dia ou noite) e um adicional noturno, crie as classes `Tecnico` e `Administrativo`.
- Implemente o arquivo teste e teste as estruturas (Por exemplo, crie um assistente administrativo e um técnico. Imprima o período do contrato e o nome de cada um deles).

2. Planeje, implemente e teste o projeto a seguir de acordo com os diagramas de classe abaixo:





- crie um animal do tipo cachorro e faça-o latir. Crie um gato e faça-o miar. Faça os dois animais caminharem.
- teste (como quiser) as classes Rica, Pobre e Miseravel.

3. Crie uma classe chamada Ingresso que possui um valor em reais e um método imprimeValor().

- crie uma classe VIP, que herda Ingresso e possui um valor adicional. Crie um método que retorne o valor do ingresso VIP (com o adicional incluído).
- crie uma classe Normal, que herda Ingresso e possui um método que imprime: "Ingresso Normal".
- crie uma classe CamaroteInferior (que possui a localização do ingresso e métodos para acessar e imprimir esta localização) e uma classe CamaroteSuperior, que é mais cara (possui valor adicional). Esta última possui um método para retornar o valor do ingresso. Ambas as classes herdam a classe VIP.
- Implemente o arquivo teste.
- crie um ingresso. Peça para o usuário digitar 1 para normal e 2 para VIP. Conforme a escolha do usuário, diga se o ingresso é do tipo normal ou VIP. Se for VIP, peça para ele digitar 1 para camarote superior e 2 para camarote inferior. Conforme a escolha do usuário, diga se que o VIP é camarote superior ou inferior. Imprima o valor do ingresso.

4. Crie a classe Imovel, que possui um endereço e um preço.

- crie uma classe Novo, que herda Imovel e possui um adicional no preço. Crie métodos de acesso e impressão deste valor adicional.
- crie uma classe Velho, que herda Imovel e possui um desconto no preço. Crie métodos de acesso e impressão para este desconto.
- crie um imóvel. Peça para o usuário digitar 1 para novo e 2 para velho. Conforme a definição do usuário, imprima o valor final do imóvel.