 <b>FASAM</b> FACULDADE SUL-AMERICANA	<b>DATA:</b>
<b>Lista POO 3</b> <b>CURSO:</b> Sistemas de Informação <b>DISCIPLINA:</b> Programação Orientada a Objetos <b>PROFESSOR:</b> Luiz Mário Lustosa Pascoal <b>ALUNO:</b>	<b>NOTA:</b>

### HERANÇA E CLASSES ABSTRATAS

1. Crie um programa para gerenciar um restaurante. O programa deve permitir cadastrar as mesas que estão ocupadas por clientes.

Cada mesa deve ter as seguintes informações: numero, nome do cliente, e uma lista dos itens pedidos.

=====

Mesa

=====

- numero : int  
- cliente : String  
- pedido : ArrayList<Pedido>

=====

//métodos

=====

O pedido deve ter o nome do produto e o valor.

=====

Pedido

=====

- nome: String  
- valor: double

=====

//métodos

=====

Armazene as mesas ocupadas em uma lista.

Cada cliente que chega, adicione uma nova mesa na lista. Cada cliente que sai, remova a mesa da lista. O valor da conta da mesa removida deve ser acumulada numa variável "faturamento", para saber quando foi apurado no dia.

=====

Restaurante

=====

- mesas : ArrayList<Mesa>  
- faturamento : double

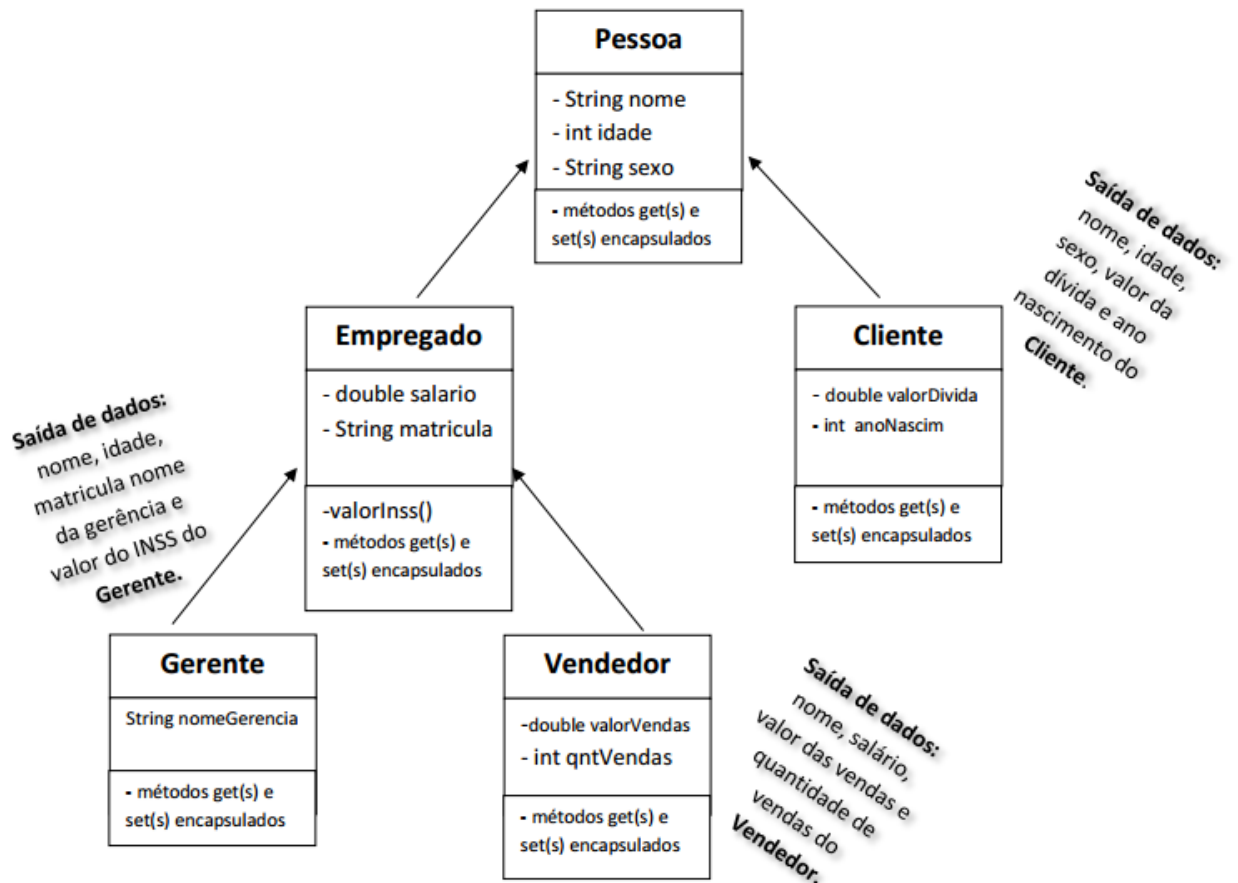
=====

//métodos

=====

Crie os métodos e construtores necessários para o funcionamento do programa. Mantenha o programa funcionando com um menu para adicionar uma nova mesa e sem seguida adicionar um novo pedido a mesa. Ao usuário informar que deseja sair do programa, ele deverá imprimir qual o faturamento do Restaurante aquele dia.

2. Crie as classes solicitadas no diagrama abaixo.
  - a. Faça o relacionamento (herança) entre as classes.
  - b. Defina a saída dos dados (toString()) nas classes indicadas. A classe onde tem a indicação é onde estará a saída – toString().
  - c. Faça a classe de teste e execute.
  - d. O método valorInss() tem a fórmula (salário \* 11%).



3. Implemente uma **classe abstrata** de nome **Forma** que tem dois métodos:
  - float calcularArea();
  - float calcularPerimetro();

Crie, como subclasse de Forma, uma classe de nome **Retângulo** cujas instâncias são caracterizadas pelos atributos lado e altura ambos do tipo float. Implemente na classe Rectangulo os métodos herdados de Forma e outros que ache necessários.

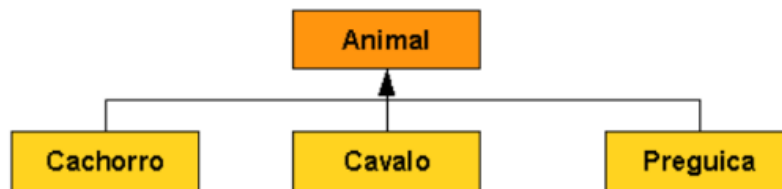
Crie, como subclasse de Forma, uma classe de nome **Circulo** cujas instâncias são caracterizadas pelo atributo raio do tipo float. Programe na classe Circulo os métodos herdados de Forma e outros que ache necessários. Nota: poderá aceder ao valor de Pi fazendo Math.Pi.

Crie, como subclasse de Retângulo, uma classe de nome **Quadrado** cujas instâncias são caracterizadas por terem os atributos lado e altura com o mesmo valor.

Elabore um programa de teste onde é declarado um ArrayList do tipo <Forma>. Nesse arraylist devem ser guardadas instâncias de Retângulo, Circulo e Quadrado, pelo menos 2 de cada, seguindo uma ordem aleatória. Depois implemente um ciclo que percorra o arraylist imprimindo os resultados, em relação a cada um dos objetos guardados, dos métodos calcularArea e calcularPerimetro. Ou seja, imprima a área e o perímetro de todos os objetos armazenados no ArrayList.

4. Faça o que se pede:

- a) Crie uma Classe Pessoa, contendo os atributos encapsulados, com seus respectivos seletores (getters) e modificadores (setters), e ainda o construtor padrão e pelo menos mais duas opções de construtores conforme sua percepção. Atributos: String nome; String endereço; String telefone;
  - b) Considere como subclasse da classe Pessoa (desenvolvida no exercício anterior) a classe Fornecedor. Considere que cada instância da classe Fornecedor tem, para além dos atributos que caracterizam a classe Pessoa, os atributos valorCredito (correspondente ao crédito máximo atribuído ao fornecedor) e valorDivida (montante da dívida para com o fornecedor). Implemente na classe Fornecedor, para além dos usuais métodos seletores e modificadores, um método obterSaldo() que devolve a diferença entre os valores dos atributos valorCredito e valorDivida. Depois de implementada a classe Fornecedor, crie uma classe de teste adequado que lhe permita verificar o funcionamento dos métodos implementados na classe Fornecedor e os herdados da classe Pessoa.
  - c) Considere, como subclasse da classe Pessoa, a classe Empregado. Considere que cada instância da classe Empregado tem, para além dos atributos que caracterizam a classe Pessoa, os atributos codigoSetor (inteiro), salarioBase (vencimento base) e imposto (porcentagem retida dos impostos). Implemente a classe Empregado com métodos seletores e modificadores e um método calcularSalario. Altere o programa de teste adequado para a classe Empregado.
  - d) Implemente a classe Administrador como subclasse da classe Empregado. Um determinado administrador tem como atributos, para além dos atributos da classe Pessoa e da classe Empregado, o atributo ajudaDeCusto (ajudas referentes a viagens, estadias, ...). Note que deverá redefinir na classe Administrador o método herdado calcularSalario (o salário de um administrador é equivalente ao salário de um empregado usual acrescido das ajuda de custo). Altere novamente a classe de teste adequado para esta classe.
  - e) Implemente a classe Operario como subclasse da classe Empregado. Um determinado operário tem como atributos, para além dos atributos da classe Pessoa e da classe Empregado, o atributo valorProducao (que corresponde ao valor monetário dos artigos efetivamente produzidos pelo operário) e comissao (que corresponde à porcentagem do valorProducao que será adicionado ao vencimento base do operário). Note que deverá redefinir nesta subclasse o método herdado calcularSalario (o salário de um operário é equivalente ao salário de um empregado usual acrescido da referida comissão). Escreva um programa de teste adequado para esta classe.
  - f) Implemente a classe Vendedor como subclasse da classe Empregado. Um determinado vendedor tem como atributos, para além dos atributos da classe Pessoa e da classe Empregado, o atributo valorVendas (correspondente ao valor monetário dos artigos vendidos) e o atributo comissao (porcentagem do valorVendas que será adicionado ao vencimento base do Vendedor). Note que deverá redefinir nesta subclasse o método herdado calcularSalario (o salário de um vendedor é equivalente ao salário de um empregado usual acrescido da referida comissão). Escreva um programa de teste adequado para esta classe.
5. Crie uma hierarquia de classes conforme abaixo com os seguintes atributos e comportamentos (observe a tabela), utilize os seus conhecimentos e distribua as características de forma que tudo o que for comum a todos os animais fique na classe Animal:



Cachorro	Cavalo	Preguica
Possui Nome	Possui Nome	Possui Nome
Possui Idade	Possui Idade	Possui Idade
Deve emitir som	Deve emitir som	Deve emitir som
Deve correr	Deve correr	Deve subir em árvores

- Implemente um programa que crie os 3 tipos de animais definidos no exercício anterior e invoque o método que emite o som de cada um de forma polimórfica, isto é, independente do tipo de animal.
- Implemente uma classe Veterinario que contenha um método examinar() cujo parâmetro de entrada é um Animal, quando o animal for examinado ele deve emitir um som, passe os 3 animais com parâmetro.
- Crie uma classe Zoologico, com 10 jaulas (utilize um array) coloque em cada jaula um animal diferente, percorra cada jaula e emita o som e, se o tipo de animal possuir o comportamento, faça-o correr.