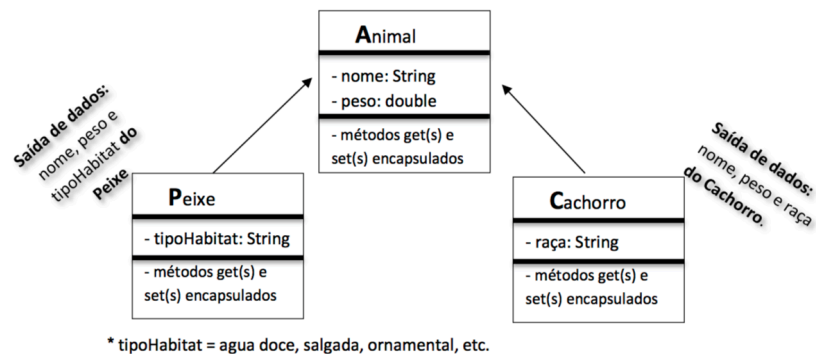




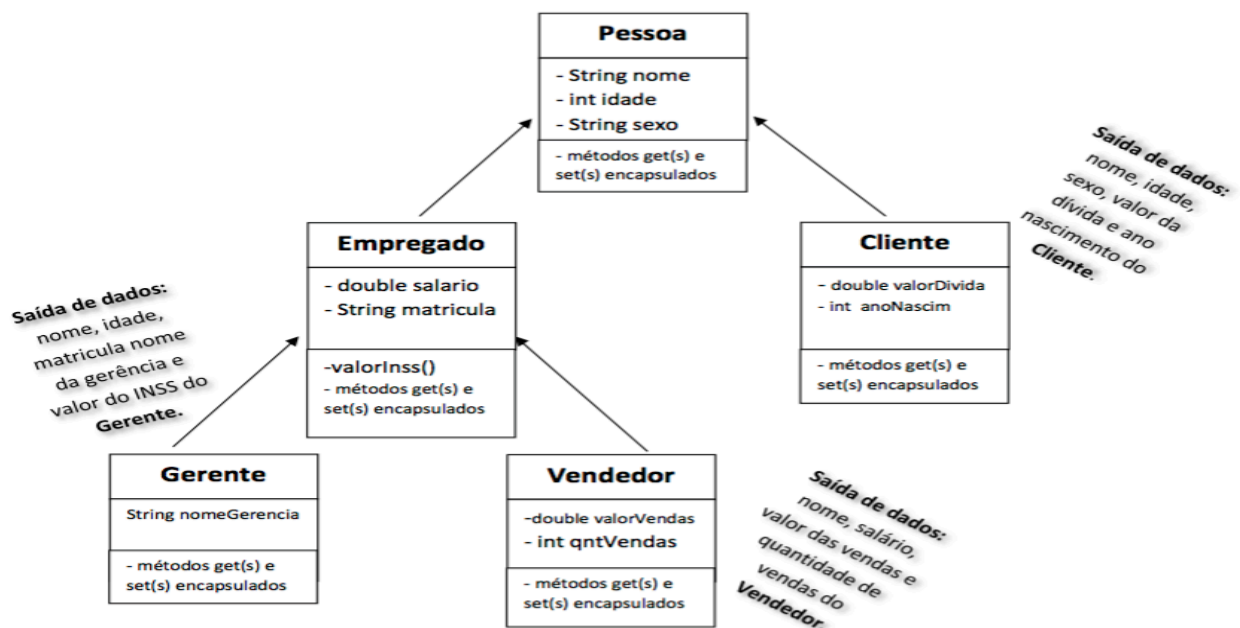
LISTA DE EXERCÍCIOS - 4

1) Implemente o seguinte classes seguindo as diretrizes dadas abaixo:



- Crie as classes solicitadas.
- Faça o relacionamento (herança) entre as classes.
- Defina a saída dos dados (sobrescrevendo o método `toString`) nas classes indicadas. A classe onde tem a indicação é onde estará a saída.
- Faça a classe Main com um método main e crie instâncias das classes exercitando o uso dos métodos existentes.

2) Implemente o seguinte classes seguindo as diretrizes dadas abaixo:



- a) Crie as classes solicitadas.
- b) Faça o relacionamento (herança) entre as classes.
- c) Defina a saída dos dados (toString()) nas classes indicadas. A classe onde tem a indicação é onde estará a saída – toString().
- d) Faça a classe Main com um método main e crie instâncias das classes exercitando o uso dos métodos existentes.
- e) O método valorInss() utiliza a fórmula (salário * 11%) para compor o resultado.

3) Estenda as classes do exercícios anterior de acordo com os seguintes diretivas:

- a) Considere, como outra 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).
- b) Implemente a classe Empregado com métodos seletores e modificadores e um método calcularSalario.
 - i) * Escreva um programa de teste adequado para a classe Empregado.
- c) 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).
 - i) *Escreva um programa de teste adequado para esta classe.
- d) 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).
 - i) *Escreva um programa de teste adequado para esta classe.
- e) 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).
 - i) *Escreva um programa de teste adequado para esta classe.

f)

- 4) Tente criar você mesmo um conjunto de classes que tenham o relacionamento de herança que não tenha como base nenhum dos nossos exercícios (desta lista ou outros anteriores), e implemente – o objetivo desse exercício é você mesmo criar outro programa e implementar sem copiar dos nossos exemplos e exercícios.**

Bom treino!