Lista para testar diferentes linguagens de programação em ORIENTAÇÃO A OBJETOS:

CLASSES:

1 – Elabore um programa contendo a seguinte classe:
*CaixaDeNumero
Considere o atributo:
*numeroSecreto
Considere as seguintes operações(Métodos):
*ehPar
*dizerTabuada
*dizerQualNumero

Na função principal deve ter a funcionalidade “ADVINHE O NÚMERO” no qual deve gerar um valor randômico de 1 a 10, dizer se o numero é par, mostrar a tabuada desse número e exibir esse número secreto no final.

---

2 - Efetuar o mesmo programa do exercício 1 porém acrescentar mais dois métodos:
* estaEntreDoisNumeros
* palpiteRealizado
No qual o usuário digitar um intervalo e verificar se o numero secreto está neste intervalo, e o outro método para que o usuário dê um palpite sobre qual é o número e informar se o palpite está correto.
Além disso o programa deve apresentar um menu de opções para escolha do usuário e só deve encerrar quando o usuário escolher a opção sair; segue abaixo como deve ficar o menu:
escreva("1 - VERIFICA SE É PAR OU ÍMPAR");
escreva("2 - VERIFICA SE ESTÁ ENTRE UM INTERVALO FORNECIDO PELO USUÁRIO");
escreva("3 - ARRISCAR DIZER QUAL NÚMERO");
escreva("4 - EXIBI A TABUADA DO NÚMERO SECRETO");
escreva ("5 - MOSTRA O NÚMERO SECRETO");
escreva ("6 - SAIR");

CLASSES COM MAIS DE UM CONSTRUTOR:

3 - Elabore um programa Com 2 classes:

Pessoa -> ATRIBUTOS: nome, idade, peso
CONSTRUTOR: (String nome, Int idade, Float peso)
CONSTURTOR: (String nome)

Carro-> ATRIBUTOS: Pessoa piloto
CONSTRUTOR: ()
CONSTRUTOR: (Pessoa pes)

FUNÇÃO PRINCIPAL: Criar 4 objetos:
pessoa1 => exibir nome, idade, peso de uma pessoa

pessoa2 => exibir nome de uma pessoa
carro1 => exibir nome do piloto do carro1
carro2 => exibir nome, idade, peso do piloto do carro2

HERANÇA:

4 - Elabore um programa contendo 2 classes:
*Pessoa
*Aluno
Considere os seguintes atributos:
*nome
*idade
*matricula
Considere as seguites operações:
*dizerNome
*dizerMatricula
*dizerNomeMatricula

=> O programa deve exibir todos os métodos criados dizerNome, dizerMatricula, dizerNomeMatricula.
NOTA: Atributos e operações comuns devem ficar na classe de mais alto nível na hierarquia.

---

5 - Elabore um programa contendo 2 classes:
* Veículo
* Carro
* Motocicleta
Considere os seguintes atributos
* marca
* cor
* motorLigado (boolean)
* estilo: trail, naked, custom
* portaMalasCheio (boolean)
Considere as seguintes operações
* Liga/desliga motor
* enche/esvazia porta malas
* mostraAtributos

Deve instanciar um carro e uma moto. Deve-se ligar a moto e mostrar seus atributos. Em seguida, deve-se encher o porta malas do carro, ligá-lo e mostrar seus atributos Nota: Atributos e operações comuns devem ficar na classe de mais alto nível na hierarquia.

---

6 - Elabore um programa contendo as classes:

●Pessoa

●Professor

●Aluno

Considere os seguintes atributos

●Nome

●CPF

●Salario

●Nota

Considere as seguintes operações

●AtribuiNota

●ReajustaSalario

●MostraAtributos

Na função principal deve instanciar um professor e dois alunos.

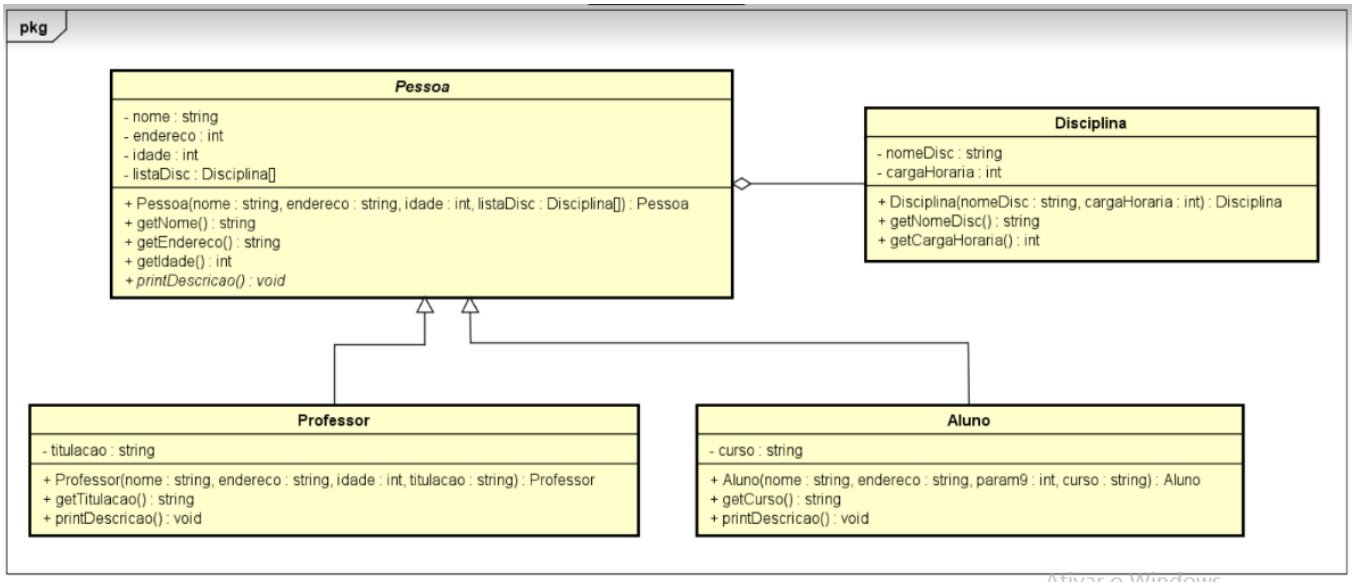
Nota: Atributos em comum devem ficar na classe mais alta

CLASSE ABSTRATA, HERANÇA E RELACIONAMENTOS, GETTERS:

7 – Implementar a seguinte modelagem, os que estão em itálico são classes e métodos abstratos. Tatos alunos quanto professores possuem relacionamento com disciplina. Notes através dos métodos getters que os atributos são privados.

*Alunos Cursam Disciplina

*Professores Ministram Disciplina



8 – A partir do exercício 6 implementar o método insereDisc, que permite inserir disciplinas para professores e alunos

---

POLIMORFISMO:

9 – Suponha que se tenha um conjunto de documentos de diferentes formatos (doc, pdf, etc). Se eu tiver que implementar operações específicas para formatos específicos irá dar muito trabalho. Uma maneira de realizar esse procedimento é utilizando polimorfismo. Faça um programa demostrando o polimorfismo para os documentos PDF e WORD, criado uma superclasse abstrata documento. Na classe filha word e pdf deve conter apenas o método abstrato visualizar dando a informação se é PDF ou WORD.
Na função Principal gere uma lista de documentos e imprima-os.