

INFORMAÇÕES SOBRE A AVALIAÇÃO

AV2 AVALIAÇÃO - 100 PONTOS - Prática

INFORMAÇÕES DOCENTE

CURSO:	DISCIPLINA:	TURNO	MANHÃ	TARDE	NOITE	PERÍODO/SALA:
ADS	POO				X	

PROFESSOR (A): Michelle Hanne Soares de Andrade

INFORMAÇÕES DISCENTE

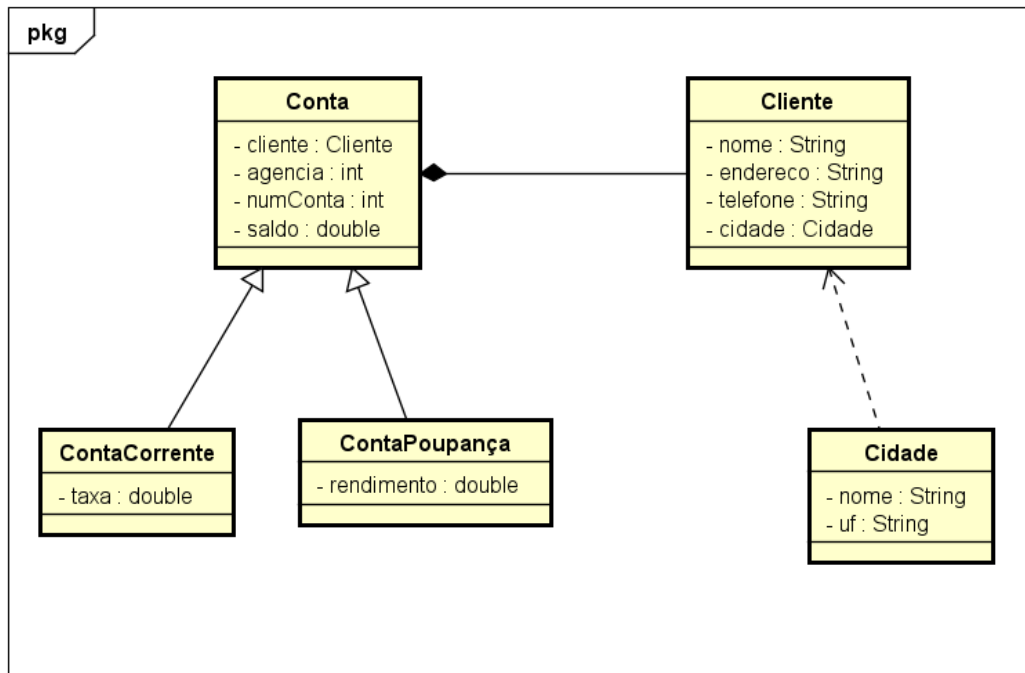
ALUNO(A):	RA:
DATA:	NOTA:

INSTRUÇÕES:

- 1) Respostas deverão ser NECESSARIAMENTE registradas à caneta azul ou preta, de corpo transparente. Respostas à lápis e com caneta "escreve e apaga" não serão corrigidas.
- 2) Caso seja identificada rasura ou duplo preenchimento da questão objetiva, a questão será anulada.
- 3) Nas questões discursivas, quando aplicável/necessário, deve ser apresentada a memória dos cálculos (ou o raciocínio desenvolvido).
- 4) Desenvolva a prova discursiva com letra legível. Caso a caligrafia impossibilite o entendimento do conteúdo, o mesmo não será corrigido.
- 5) Não será permitido o uso de celulares, relógios digitais ou de qualquer outro aparelho eletrônico durante a realização da prova, exceto o uso de calculadora portátil no caso de disciplinas que a autorizam.
- 6) Será eliminado do processo avaliativo, com atribuição de nota 0 (zero), o aluno que, durante a prova, utilizar meios ilícitos, como por exemplo: comunicação com outros alunos, consulta a qualquer tipo de material extraprova e uso de aparelhos eletrônicos não autorizados. Também não poderá haver empréstimo de qualquer material.

Boa Prova!

Questão 1) Analise o contexto abaixo e crie a estrutura de classes: (25 pontos)



Um banco solicita um sistema construído com recursos de orientação à objetos em Java.

O sistema deve permitir o cadastro de:

Cidade: Criar o construtor e os métodos setters e getters.

Cliente: Criar o construtor e os métodos setters e getters. Usar associação do objeto Cidade.

Conta: Criar o construtor e os métodos setters e getters. Usar composição.

Criar os métodos: public void sacar() e public void depositar()

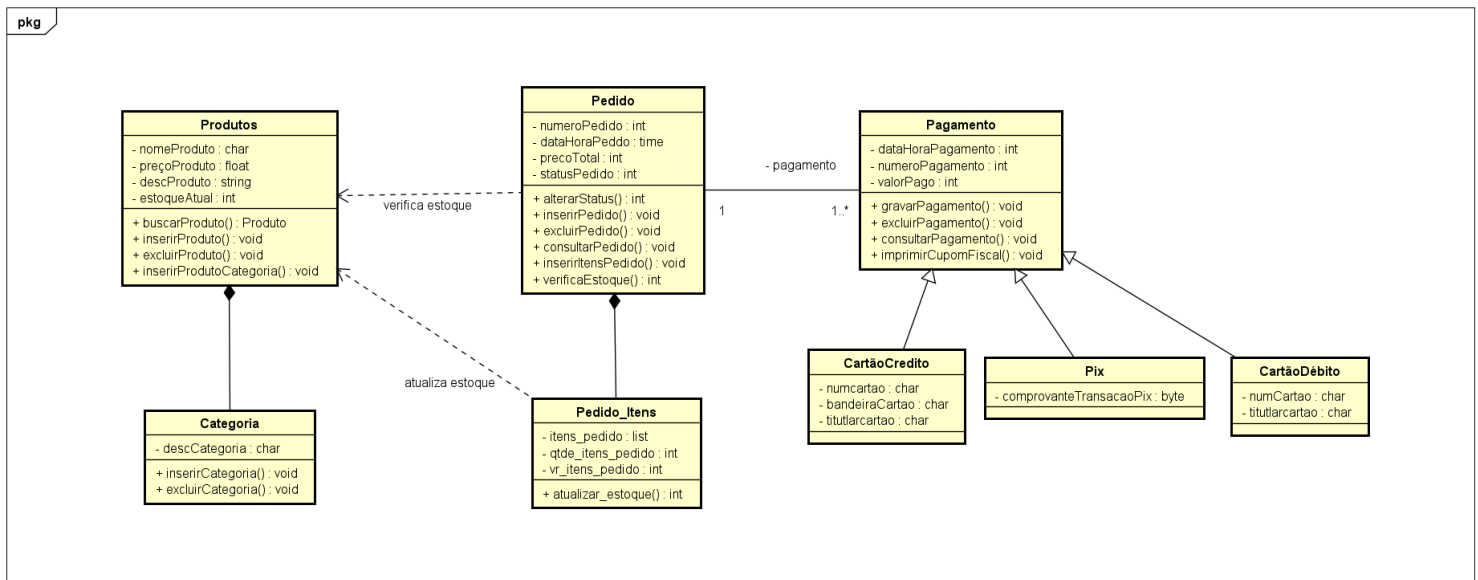
Conta Cliente: Criar o construtor e os métodos setters e getters, usar herança.

Conta Poupança: Criar o construtor e os métodos setters e getters, usar herança.

Crie o **Main()** em outra classe que deverá solicitar os dados usando **System.out.println** ou **JOptionPane**, criar os objetos do tipo **contacorrente** e **contapoupanca** e usar os métodos **depósito** e **saque**. Ao final imprimir o saldo do cliente.

Questão 2) Analise o contexto abaixo e crie a estrutura de classes: (25 pontos)

Implementar as classes **Pedido** e **Pedido_Itens**. Criar os métodos conforme descritos no diagrama, **sem conteúdo**. Criar o **main()** que irá instanciar um **Pedido** e um **ArrayList** de **Pedido_Itens**.



Questão 3) Sobre os conceitos de Programação Orientada à Objeto (POO) observe as afirmações abaixo: **(5 pontos)**

I) Herança permite o reaproveitamento de código; Polimorfismo permite a economia de código; Encapsulamento permite o controle de acesso a propriedades e métodos;

II) Classes representam somente abstrações do mundo real e não representam objetos de software;

III) Uma das vantagens de POO é o registro das mudanças que forem feitas nos arquivos monitorados;

Marque a opção que corresponde a(s) asserção (ões) correta(s):

- a) I e II.
- b) I e III
- c) II e III.
- d) I, II e III.
- e) Nenhuma está correta.

Questão 4) O que será exibido pelo programa abaixo? **(5 pontos)**

```
public class A {
    public int i;
    void display() {
        System.out.println(i);
    }
}

public class B extends A {
    public int j;
    void display() {
        System.out.println(j);
    }
}

public class questaoP001 {

    public static void main(String[] args) {
        B obj = new B();
        obj.i=1;
        obj.j=2;
        A obj1 = new A();
        Obj1.i=3;
        Obj1.display();
    }
}
```

- a) 1.
- b) 2.
- c) 3.
- d) Nada, pois há um erro de execução.
- e) Nada, pois há um erro de compilação.

Questão 5) Marque a alternativa INCORRETA em relação à definição de polimorfismo: **(5 pontos)**

- a) Polimorfismo e encapsulamento são duas características que não coexistem.
- b) Polimorfismo permite que um objeto implemente uma mesma operação de várias formas distintas.
- c) Há dois tipos de polimorfismo: sobrecarga e sobrescrita.
- d) Polimorfismo de sobrecarga consiste em reescrever um método de uma classe mudando seus argumentos de entrada e/ou de retorno preservando seu nome.
- e) Polimorfismo de sobrescrita consiste em reescrever um método de uma classe base numa classe mãe mudando sua implementação.

Questão 6) Se uma Classe A herda os métodos abstratos de uma Classe B, mas concretiza todos os seus métodos abstratos. Então a Classe A poderá ser classificada como: **(5 pontos)**

- a) Classe abstrata.
- b) Classe simples.
- c) Classe estática.
- d) Classe final.
- e) Classe concreta.

Questão 7) Assinale a assertiva INCORRETA: (5 pontos)

- a) membros privados de uma classe podem ser herdados por uma subclasse, tornando-se membros protegidos na subclasse.
- b) membros privados de uma classe somente podem ser acessados por outros membros dessa mesma classe.
- c) membros protegidos de uma classe podem ser herdados por uma subclasse.
- d) membros públicos de uma classe podem ser acessados por qualquer parte do código do programa.
- e) não é obrigatório que membros de uma classe sejam definidos como públicos, privados ou protegidos.

Questão 8) A respeito da Programação Orientada a Objetos, analise as afirmativas abaixo e assinale a alternativa correta. (5 pontos)

- I. Na definição de uma classe é possível não especificar atributos.
 - II. Durante a execução de um programa, o número de objetos instanciados deve ser igual ao de classes definidas.
 - III. Toda classe deve, obrigatoriamente, possuir pelo menos um método.
-
- a) Somente a afirmativa I está correta.
 - b) Somente as afirmativas I e III estão corretas.
 - c) Somente as afirmativas II e III estão corretas.
 - d) Somente a afirmativa II está correta.
 - e) Todas as afirmativas estão corretas.

Questão 9) Qual o recurso de Programação Orientada à Objetos mais adequado a ser utilizado na situação: criar as classes Cliente, Conta e Agência de um sistema bancário, onde a classe Conta tem um cliente e tem uma agência vinculados. (5 pontos)

- a) Herança (estender uma classe)
- b) Composição (ter como atributo de uma classe um objeto de outra)
- c) Sobrecarga
- d) Sobrescrita
- e) Instanciação

Questão 10) Qual o recurso de Programação Orientada à Objetos mais adequado a ser utilizado na seguinte situação: **(5 pontos)**

Criar as classes CD, Livro e Revistas para o sistema de uma biblioteca.

Todas as classes deverão ter os atributos titulo, autor, anoDeLancamento e localNaBiblioteca.

- a) Herança (estender uma classe)
- b) Composição (ter como atributo de uma classe um objeto de outra)
- c) Sobrecarga
- d) Sobrescrita
- e) Instanciação

Questão 11) Considere o seguinte trecho de código: **(5 pontos)**

```
public class MyClass {  
  
    public MyClass() {  
        /*código*/  
    }  
  
    // mais código...  
}
```

Qual a assertiva VÁLIDA para instanciar MyClass?

- a) `MyClass mc = new MyClass();`
- b) `MyClass mc = MyClass();`
- c) `MyClass mc = MyClass;`
- d) `MyClass mc = new MyClass;`
- e) O construtor de MyClass deve ser definido como `public void MyClass(){/*código*/}`

Questão 12) Para controlar quais estruturas terão acesso às propriedades e métodos de uma classe a linguagem Java conta com o recurso: **(5 pontos)**

- a) Composição
- b) Herança
- c) Polimorfismo
- d) Encapsulamento
- e) Instanciação