Lista de Exercícios

Lista I

Programação Orientada a Objetos

Data de entrega: 29 - 07 - 2015

1) Defina programação orientada a objetos, diferenciando da programação estruturada.

É um paradigma de programação onde os elementos da vida real ou não são representados como classes e objetos, onde esses objetos tem vários atributos de tipos mais simples e métodos que representam as ações que o elemento modelado pode realizar.

Esse método de modelagem possibilita entre outras coisas o encapsulamento do código e herança entre classes. Diferentemente da programação estruturada onde o código é mais fragmentado. As variáveis e métodos muitas vezes são projetados para aquela aplicação.

2) Defina: classe, objeto, método e atributo.

Classe: É um tipo pré-definido ou criado pelo usuário que serve de molde para objetos.

Objeto: É uma instancia de uma classe.

Método:É uma ação que um objeto pode realizar.

Atributo:Uma característica que o objeto tem.

3) Considerando os tipos de dados nativos em Java, escolha o tipo de dado ou classe mais

adequada para representar:

* O número de municípios de um estado do Brasil.int
* O nome de um estado do Brasil.String
* A população de um estado do Brasil. int
* A área do Brasil em quilômetros quadrados. float
* A população total do mundo.longint
* O CEP de um endereço no Brasil. int
* O nome de uma rua em um endereço no Brasil. String
* A altura de uma pessoa em metros. float
* O peso de uma pessoa em quilos. float
* A temperatura corporal de uma pessoa. float
* O sexo de uma pessoa. boolean
* A altura de uma pessoa em milímetros.float

4) Crie uma classe DVD, que tenha como características: marca, modelo, peso, se é slim ou

não, e o preço.

5) Quais os conceitos de orientação a objetos você consegue identificar no código abaixo

e onde:

**publicclass**Q1{ //Declaraçãodeumaclasse.

**privateint**x,y; //Atributosdaclasse.

**public** Q1(**int**a, **int**b){ //Construtorcomargumentos.

**if**(a<b){

x = a;

y = b;

}**else** {

x = b;

y = a;

}

}

**public** Q1(){ //Construtorsemargumentos(polimorfismo).

x = y = 0;

}

**publicvoid**setX(**int**a){ //Metodosparasetarosatributosprivados.

x = a;

}

**publicvoid** setY(**int**a){

y = a;

}

// ...

}

6) Descreva em no máximo 200 palavras o que é um automóvel e o que ele faz. Liste os

substantivos e verbos separadamente. Cada substantivo corresponde a um objeto que

precisará ser construído para implementar um sistema, nesse caso, um carro. Selecione

5 dos objetos que você listou e, para cada um, liste vários atributos e comportamentos.

Descreva brevemente como esses objetos interagem entre si e com outros objetos na

sua descrição. Estes passos que você seguiu são típicos do projeto orientado a objetos.

È um meio de transporte terrestre, dotado de um motor alimentado a gasolina, diesel , etc e energia elétrica que sustenta vários sistemas como sinalização, limpador de parabrisa, buzina etc.

Possue quatro rodas, freios, direção, transmição , capacidade de pessoas dependendo do porte do automóvel. Placa, peso, cor, marca.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Objeto | Motor | Transmissão | Freios | Rodas | SistemaEletrico |
| Atributos | Combustível  RPM  Temperatura | Marcha  Tipo(manual/automatica) | Tipo | Tamanho do aro  Marca do pnel  Pressão do pnel | Tensão  Corrente |
| Comportamento | Acelerar | Engatar  Liberar | Frenar  Liberar | --- | Ligar  Desligar |

O primeiro sistema a ser ligado é o eletrico, responsavel por praticamente toda a sinalização do automóvel e pela ignição do motor.

O motor converte a energia da queima do combustível em movimento circular que é transmitido através do sistema de transmissão paras a s rodas.

Os freios reduzem a velocidade do veiculo impondo resistência ao movimento das rodas.

7) A fim de representar empregados em uma firma, crie uma classe chamada Empregado

que inclui as três informações a seguir como atributos:

* um primeiro nome,
* um sobrenome, e
* um salário mensal.

Sua classe deve ter um construtor que inicializa os três atributos. Forneça um método

set e get para cada atributo.

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**publicclass** Empregado {

Stringprimeiro\_nome;

String sobrenome;

**float**salário\_mensal;

**public**StringgetPrimeiro\_nome() {

**return**primeiro\_nome;

}

**publicvoid**setPrimeiro\_nome(Stringprimeiro\_nome) {

**this**.primeiro\_nome = primeiro\_nome;

}

**public** String getSobrenome() {

**return**sobrenome;

}

**publicvoid**setSobrenome(Stringsobrenome) {

**this**.sobrenome = sobrenome;

}

**publicfloat**getSalário\_mensal() {

**return**salário\_mensal;

}

**publicvoid**setSalário\_mensal(**float**salário\_mensal) {

**this**.salário\_mensal = salário\_mensal;

}

**public** Empregado(Stringprimeiro\_nome, Stringsobrenome,

**float**salário\_mensal) {

**super**();

**this**.primeiro\_nome = primeiro\_nome;

**this**.sobrenome = sobrenome;

**this**.salário\_mensal = salário\_mensal;

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

8) Represente, em Java, a seguinte afirmação: "A classe B é composta pelos objetos o1 e

o2 que pertencem, respectivamente, às classes A e C".

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**publicclass**A{

**int**atributo1;

**char**atributo2;

**public** A(**int**atributo1,**char**atributo2){

**super**();

**this**.atributo1 = atributo1;

**this**.atributo2 = atributo2;

}

}

**publicclass**C{

String atributo1;

**char**atributo2;

**public** C(String atributo1,**char**atributo2){

**super**();

**this**.atributo1 = atributo1;

**this**.atributo2 = atributo2;

}

}

**publicclass**B{

A atributo1;

C atributo2;

**public**C(A atributo1,C atributo2){

**super**();

**this**.atributo1 = **new**A(atributo1);

**this**.atributo2 = **new**C(atributo2);

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

9) Considere a classe A abaixo. Implemente o método "m" conforme as seguintes regras:

o método retorna true se o valor proposto no parâmetro de m for maior que zero. O

método retorna false se o valor proposto no parâmetro de m for menor ou igual à zero.

Se o método retornar true então ele deve, também, alterar o valor do atributo a para 10 vezes o valor proposto no parâmetro do método m.

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**import**java.util.\*;

**publicclass** A {

**staticint***a*;

**public** A() {

*a*=0;

}

**publicstaticvoid** main(String[] args) {

**int**aux;

Scanner sc = **new**Scanner(System.***in***);

System.***out***.println("Entre com um inteiro");//Saidadedados.

aux = sc.nextInt();

**if**(*m*(aux)){

*a* = aux\*10;

System.***out***.println("a = "+*a*);

}

**else**{

System.***out***.println("Valor invalido!");

}

}

**publicstaticboolean** m(**int**valor){

**if**(valor>0) **returntrue**;

**elsereturnfalse**;

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

10) Qual é o valor da variável x após a invocação do método "m" dentro do método "t"?

10