ADO -1 Projeto Integrador

1. Qual a diferença entre classe e objetos?

R: A classe pode ser classificada como um modelo/planejamento, já o objeto é basicamente a materialização desta classe com seus atributos.

1. Qual é a finalidade do método construtor?

R: Definir as ações que devem ser executadas quando instanciado o objeto.

1. Identifique na instrução abaixo: a classe, o objeto, o construtor e a operação de instanciação.

Computador computador = new Computador();

R: Computador -> classe

computador -> objeto

New -> construtor

Computador() -> Operação de instanciação

1. O que diferencia um construtor de um método qualquer?

R: O construtor tem o mesmo nome da classe; inicia com a diretiva de encapsulamento public, não possui um tipo de retorno e não pode ser precedido por nenhum tipo de qualificador.

1. Observe que a classe abaixo não possui um construtor, porém para que seja criado um objeto sempre é necessário utilizar o operador new seguido do nome do construtor. Desta forma, não será possível criar um objeto desta classe? Explique.

Public class Garrafa{

private String tipo;

public void setTipo(String tipo){

this.tipo = tipo;

}

public String getTipo(){

return tipo;

}

}

R: Quando não é declarado explicitamente um construtor em uma classe, o compilador adiciona um construtor padrão (default) durante o processo de compilação. Logo será possível criar um objeto desta classe porque após a compilação a mesa irá possuir um construtor padrão.

1. A classe abaixo é parecida com a classe do exercício anterior, porém agora ela possui um construtor alternativo. O que irá acontecer se a outra classe que você desejar criar um objeto dela com a seguinte instrução:

Garrafa gar = new Garrafa();

Public class Garrafa{

private String tipo;

public Garrafa(String tipo){

this.tipo = tipo;

}

public void setTipo(String tipo){

this.tipo = tipo;

}

public String getTipo(){

return tipo;

}

}

R: Irá ocorrer erro de compilação, pois ao definir

1. As duas classes abaixo não estão no mesmo pacote. Faça uma análise das mesmas e corrija possíveis erros de compilação.

Public class Carro{

Protected int litrosNoTanque;

Protected boolean carroLigado;

Private void encherTanque(int litros){

litrosNoTanque = litros;

}

}

Public class TesteCarro{

Public static void main (String args[]){

Carro carro = new Carro();

carro.encherTanque(10);

carro.carroLigado = true;

}

}

R: Deverá importar a classe Carro em outra

1. Observe as classes Circulo e Teste Circulo e TesteCirculo a seguir e realize as seguintes tarefas:

- Declare o atributo raio da classe Circulo como privado;

- Crie os métodos necessários na classe Circulo de modo que seja possível obter ou alterar o valor do atributo raio Pela classe TesteCirculo;

- Caso o objetvo seja alterar o valor do atributo raio, faça a seguinte verificação: Se o valor do novo raio for positivo, faça a atribuição, caso contrário não faça;

- Siga as orientações presentes na classe TesteCurculo e execute cada tarefa;

public class TesteCirculo {

public static void main(String[] args) {

Circulo circulo = new Circulo();

circulo.raio = 10.0;

System.out.println("Raio do Circulo: "+circulo.raio);

}

}

public class Circulo {

double raio;

}

1. Observe as classes AcessaBanco e TesteBanco abaixo e realize as seguintes tarefas:

- Declare os atributos login e conectado da classe AcessaBanco como privado;

- Crie os métodos necessários na classe AcessaBanco de modo login e conectado pela classe TesteBanco;

- Siga as orientações presentes na classe TesteBanco e execute cada tarefa;

R: