**Aula XIII - Desenvolvimento Web III - Trabalho Avaliativo - 4,0 pontos N1**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Parte Teórica: 2, 0 pontos. Ao finalizar, enviar as questões e códigos para o email: aulas.qiti@gmail. No assunto colocar Trab.02 + nome completo**

**Responda as questões abaixo:**

1. Para cada valor apresentado abaixo, identifique o tipo de variável indicado entre String, int, double, char ou inválido:

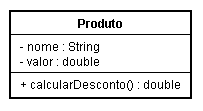
|  |  |
| --- | --- |
| **Variável (atributo)** | **Tipo de dado** |
| **1** | **inteiro** |
| 2.5 | double |
| nome | string |
| 3,14159F | Char |
| 2verificar | invalido |
| i | String |
| -2 | Int |

1. Qual a importância do encapsulamento na programação Orientada a Objeto?

É importante para que os detalhes internos do funcionamento dos métodos de uma classe permaneçam ocultos para os objetos.

1. Quanto a visibilidade das informações de uma classe no Java, defina esta visibilidade e a relação entre a execução de um projeto:

Quando o projeto é separado em classes sua segurança fica mais rígida, para que possíveis erros limitem o comando ao acesso dos dados e os altere.

1. Um importante recurso utilizado na orientação a objetos quando se trata da proteção do estado interno de um objeto é chamado de encapsulamento. Sobre este recurso, é correto afirmar que:
2. ( ) Encapsulamento define as características de um objeto de uma classe.
3. ( ) É caracterizado por tornar público todos os atributos das classes.
4. ( ) Encapsulamento é utilizado para se definir o valor inicial dos objetos no momento em que a instância é criada.
5. (X) Caracterizado por se privar os atributos de uma classe, dando acesso ao mesmo através de métodos acessores públicos.
6. ( ) Permite que apenas a classe principal possa acessar diretamente o estado interno dos objetos.
7. Analisando o seguinte diagrama de classes, qual o código fonte abaixo o representa da linguagem Java sabendo que o cálculo de desconto do produto é de 2,5% sobre o valor do mesmo.
   1. ( ) public class Produto{

private nome: String

private valor: real

public double calcularDesconto(){

return valor \* 0.025

}

* 1. ( ) public class Produto{

public double calcularDesconto(){

return valor \* 0.025

}

}

* 1. (X) public class Produto{

private String nome;

private double valor;

public double calcularDesconto(){

return valor \* 0.025

}

}

* 1. ( ) public class Produto{

public nome: double

public valor: double

public double calcularDesconto(){

return valor \* 0.025

}

}

* 1. ( ) Algoritmo "Produto"

funcao calcularDesconto(valor: real):real

inicio

return valor \* 0.025

fimfuncao

1. Analise o seguinte código desenvolvido na linguagem Java:

|  |  |
| --- | --- |
|  | **Agora, analise as seguintes afirmações:**  I. O método criado na linha 4 é considerado o método construtor da classe.  II. O método criado na linha 17 permite a leitura do atributo valor da classe Produto  III. Caso o produto possua um valor superior a 1000, será apresentado o preço de venda com 8% de desconto.  IV. Os métodos criados nas linhas 8, 11, 14 e 17 encapsulam os atributos da classe Produto.  São corretas as afirmativas?   1. (X) I e II 2. ( ) I, III e IV 3. ( ) II e III 4. ( ) I, II e III 5. ( ) I, II, III e IV |

1. O que são métodos assessores? Qual sua função?

Métodos acessores são utilizados para outras classes obterem valores de atributos privados

* Descreva as características do **método get()**:

Método get e usado para solicitar um valor já determinado;

* Descreva as características do **método set()**:

Método set e para configurar e gravar um valor de um atributo;

8. Analise as imagens abaixo e descreva o significado de suas sintaxes e a função destes métodos:

|  |  |
| --- | --- |
|  | Método assessor para solicita valor privado do int valor |
|  | Método para atribuir um valor ao atributo valor |
|  | Alterar o valor do atributo número, para receber o valor do atributo “valor” |
|  | Método assessor para solicita valor privado da String nome |

9. Observe o código abaixo e responda as questões:

|  |  |
| --- | --- |
|  | 1 - Qual o nome do construtor Scanner? New Scanner (System.in)  2 - Quais os tipos de valor que o Scanner está lendo? Double  3 - Quais os tipos primitivos de variáveis que o programa está lendo? Double e IF ELSE  4 - Qual o nome da classe que o main está executando? Qual o seu construtor? nome e T1  New Triangulo ()  5 - Qual o nome da instância da classe vinculada? Nome Triangulo |

10. A respeito dos laços de repetição utilizados na linguagem Java, assinale a alternativa correta.

1. ( ) O comando associado ao laço do-while é executado no mínimo uma vez.
2. ( ) O comando associado ao laço while é executado no mínimo uma vez.
3. (X) O comando associado ao laço do-while é executado no mínimo uma vez desde que a condição do laço seja verdadeira.
4. ( ) O comando associado ao laço do-while é executado no mínimo uma vez desde que a condição do laço seja falsa.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Parte Prática - Valor 2,0 pontos**

1. Crie um programa em Java que use os seguintes recursos (Não pode ser copiado dos que já foram feitos)
   1. Classe Main.
   2. If e else.
   3. Um tipo de laço de repetição (livre).
2. Crie um programa em Java que execute uma calculadora com as 4 operações e cálculo de porcentagem.
3. Imprima a soma de 1 até 1000.
4. Imprima todos os números de 150 a 300.
5. Crie um programa que leia "n" valores reais. Quando o usuário informar um valor negativo o programa deve parar de solicitar valores e exibir a média dos valores informados (Soma de todos os valores dividida pelo total de valores).
6. Crie um programa que receba vários números, calcule e mostre:
   1. •A soma dos números digitados
   2. •A quantidade de números digitados
   3. •A média dos números digitados
   4. •O maior número digitado
   5. •O menor número digitado