|  |  |  |  |
| --- | --- | --- | --- |
|  | (X) Prova  ( ) Trabalho  ( ) Prova Modular  ( ) Prática de Laboratório  ( ) Prova Substitutiva | | **Nota:** |
| Disciplina: Programação de Scripts (Com Python) | | Turma: TSI | |
| Professor: Franciela Nissola | | Data: 01/07/2020 | |
| Aluno (a): Cleyton A. L. e Júlia B. C | | | |

**INSTRUÇÕES – LEIA COM ATENÇÃO OS CRITÉRIOS PARA ENTREGA**

* Este instrumento de avaliação contém cinco (5) questões;
* Cada questão vale 2 pontos totalizando 10,00;
* As questões devem ser realizadas de forma individual pelo aluno;
* A resolução de cada questão deve ser realizada utilizando a linguagem Python;
* A resolução de cada questão deve ser entregue via arquivo do tipo “Python File”;
* Crie um diretório src e adicione todas as classe (e classes de teste) dentro do src;
* Cada questão deverá ser nomeada utilizando o padrão abaixo:

Questao01.py, TestaQuestao01.py ... Questao05.py, TestaQuestao05.py

* Os 5 arquivos do tipo “Python File” deverão ser zipados para ser enviados;
* A entrega das questões deve ser via e-mail: [franciela.nissola@unisociesc.com.br](mailto:franciela.nissola@unisociesc.com.br)
* Confirme via sala de aula (Zoom) o recebimento do arquivo;
* A entrega e confirmação de entrega devem ser realizadas até 22hrs (03/07/2020).

**QUESTÃO 01:**

Escreva em Python uma classe chamada Triangulo.

Atributos da classe: LadoA, LadoB, LadoC.

Utilizar encapsulamento de atributos da classe.

Crie os métodos get/set fazendo uso do padrão Decorador.

Escreva um construtor com os 4 parâmetros, inicializando atributos da classe

Métodos das classes: calcularPerímetro(ladoA, ladoB, ladoC), getMaiorLado()

Crie um programa de teste para essa classe.

O programa de teste deve pedir (no console) ao usuário todas as medidas do triangulo.

Posteriormente deve calcular o perímetro fazendo uso do calcularPerímetro.

Por fim, também deve retornar o maior lado fazendo uso do getMaiorLado.

**QUESTÃO 02:**

Escreva em Python uma classe chamada Funcionario.

Atributos da classe: Nome, Salario.

Utilizar encapsulamento de atributos da classe.

Crie os métodos get/set fazendo uso do padrão Decorador.

Escreva um construtor com os 2 parâmetros, inicializando atributos da classe.

Método da classe: aumentarSalario(percentualDeAumento).

Este método deve aumentar o salário do funcionário, considerando percentual.

Por fim, crie um programa de teste para essa classe.

O programa de teste deve imprimir nome do funcionário e novo salário, fazendo uso do método. (Por exemplo, se o salário inicial era de 5.000 e teve aumento de 8%, vai imprimir 5.400).

**QUESTÃO 03:**

Escreva em Python uma classe chamada Livro.

Atributos da classe: nome, qtdPaginas, autor e preço.

Utilizar encapsulamento de atributos da classe.

Crie os métodos get/set fazendo uso do padrão Decorador.

Escreva um construtor com os 4 parâmetros, inicializando atributos da classe.

Métodos das classes: retornaPreco(), mudaPreco(), etiqueta().

O método retornaPreco () deve retornar o preço do livro.

O método mudaPreco () deve atualizar o preço do livro, setar novo valor.

O método etiqueta() deve imprimir todos os dados dos atributos em tela.

Crie um programa de teste para essa classe.

Este programa de teste deve chamar os 3 métodos na ordem citada acima.

**QUESTÃO 04:**

Escreva em Python uma classe chamada Aluno.

Atributos da classe: nome, curso, tempoSemDormir (em horas).

Utilizar encapsulamento de atributos da classe.

Crie os métodos get/set fazendo uso do padrão Decorador.

Escreva um construtor com os 3 parâmetros, inicializando atributos da classe.

Métodos das classes: estudar (que recebe como parâmetro a quantidade de horas de estudo e acrescenta esse valor no atributo tempoSemDormir), dormir (que recebe como parâmetro a quantidade de horas de sono e reduz esse valor no atributo tempoSemDormir)

Crie um programa de teste para essa classe.

Faça uso dos métodos estudar e dormir diversas vezes.

Este programa deve ao final imprimir quanto tempo o aluno está sem dormir.

**QUESTÃO 05:**

Crie duas (2) classes definidas por você.

Utilize o conceito de herança entre essas duas (2) classes definidas por você.

Cada uma das duas (2) classes deve conter no mínimo três atributos.

Cada uma das duas (2) classes deve conter um construtor que inicializa os atributos por parâmetro;

Ainda, no construtor da classe filha deve ser possível passar os atributos de inicialização (construtor) para a classe mãe, ou seja, posso inicializar os atributos da classe mãe, instanciando apenas a classe filha.

Os atributos das classes devem fazer uso encapsulamento.

Os atributos das classes devem todos ter métodos get/set associados usando o padrão decorator.

Nessas duas (2) classes que tem relação de herança, você deverá fazer uso de um método com polimorfismo, ou seja, crie um método na classe mãe e crie um método na classe filha, ambos métodos devem ter a mesma assinatura (nome e nro de paramentos).

Crie um programa de teste para essas classes.

Instancie ambas as classes.

Faça uso de ambos os métodos na classe de teste.