

As linguagens de programação são tipicamente classificadas de acordo com suas características peculiares. Nesse sentido, considere as seguintes características de paradigmas de programação:

I - Seus comandos são basicamente funções.

II - Tem por base, o reuso, através da implementação de herança e polimorfismo.

III - Focado em eficiência.

IV - LISP é uma linguagem desse paradigma.

Assinale a opção que apresenta, em ordem, a correta classificação em paradigmas. Pode-se repetir paradigmas nas respostas.

(Ref.: 202309808644)

- ☐ funcional, orientação a objetos, estruturado, lógico.
- ☐ funcional, orientação a objetos, orientado a objetos, lógico.
- ☐ estruturado, orientação a objetos, concorrente, lógico.
- ☐ funcional, orientação a objetos, estruturado, funcional.
- ☐ estruturado, orientação a objetos, estruturado, lógico.

1 ponto

2. No contexto de programação para Web, temos 2 diferentes ambientes de desenvolvimento: a camada de apresentação e a camada de lógica do negócio. Assinale a opção que apresenta apenas linguagens voltadas para o desenvolvimento da camada de lógica do negócio nos servidores WEB.

(Ref.: 202309808833)

- ☐ C# e Ruby.
- ☐ .NET, Java e CSS.
- ☐ PHP, ASP e Javascript.
- ☐ HTML, CSS e Javascript.
- ☐ Python e HTML.

1 ponto

3. Três tipos de dados fundamentais em Python são as listas ("lists"), sequências ou 'tuplas' ("tuples") e dicionários ("dictionaries"). A respeito dessas estruturas, é correto afirmar:

(Ref.: 202307311068)

- ☐ Listas podem ser modificadas, mas seu tamanho não pode ser modificado após a criação, ao passo que sequências e dicionários não têm essa limitação.
- ☐ Listas e sequências são indexadas por inteiros, ao passo que dicionários podem ser indexados por "strings".
- ☐ Listas não podem ser modificadas depois de criadas, ao passo que sequências e dicionários podem.
- ☐ Dicionários não podem ser modificados depois de criados, ao passo que listas e sequências podem.
- ☐ Listas e dicionários são indexados por inteiros, ao passo que sequências podem ser indexadas por "strings".

1 ponto

4. (Quadrix/COREN-RS/2018 - adaptada) Python é uma linguagem extremamente versátil e de fácil assimilação. Sua sintaxe permite a manipulação de estruturas complexas com um grau de dificuldade inferior as linguagens concorrentes. No que se refere à linguagem de programação Python, assinale a alternativa correta.

(Ref.: 202309643977)

- ☐ Python é uma linguagem de alto nível e robusta. Possui seu próprio framework e é incompatível com frameworks de terceiros.
- ☐ Python permite que os conteúdos das variáveis sejam sempre alterados, não existindo, dessa forma, tipos imutáveis.
- ☐ O caractere "/" marca o início de comentário. Qualquer texto depois do "/" será ignorado até o fim da linha.
- ☐ Pode ser utilizada como linguagem principal no desenvolvimento de sistemas e também pode ser utilizada como linguagem script em vários softwares.

- ☐ Python utiliza a duck typing (tipagem dinâmica), que nada mais é do que definir um tipo para a variável, com as operações que podem ser aplicadas, antes mesmo de ela ter sido criada, com base em conhecimento prévio do programa. Essa tarefa é executada pelo interpretador.

1 ponto

5. Analise o trecho de código, em Python 3, a seguir.

```
def fc(x,y):  
    s=0  
    a = x.lower()  
    for i in a:  
        if (i==y):  
            s = s + 1  
  
    return s  
  
a = 'Aracajú/Sergipe'  
x = fc(a,'a')*100  
y = fc(a,'e')*10  
z = fc(a,'i')  
print(x+y+z)
```

O valor impresso pela função **print**, na última linha do trecho de código, é: (dica: a função **lower** retorna todas as letras de uma string em minúsculo)

(Ref.: 202307314121)

- ☐ 132
☐ 321
☐ 111
☐ 123
☐ 333

1 ponto

6. (FGV/2018) Considere a seguinte definição da função f, declarada na sintaxe Python.

```
def f(n):  
    if n < 3:  
        return n-1  
    else:  
        return f(n-2) + f(n-1)  
print f(10)
```

Assinale o valor produzido pela execução do código acima.

(Ref.: 202309730369)

- ☐ 17
☐ 55
☐ 34
☐ 13
☐ 21

1 ponto

7. (IFB/2017 - Adaptada) Com relação a Algoritmos e Estrutura de Dados, dadas as afirmações abaixo, assinale a alternativa CORRETA:

(Ref.: 202308130386)

- ☐ Em comandos condicionais ou de repetição, a separação de blocos de código em Python é feita utilizando a indentação.
☐ Python é uma linguagem procedural a qual não contempla orientação a objetos.

- ☐ Python é uma linguagem compilável, a qual, após esse processo, gera um arquivo executável, do tipo binário .py.
- ☐ A declaração de atributos é algo obrigatório em Python.
- ☐ Listas, em Python, são conjuntos de valores, os quais só podem ser do mesmo tipo, acessados por um índice numérico que inicia em 1 (um).

1 ponto

8. (TRE-CE/2012 - Adaptada) Considere:

```
1 class A:
2     a = 1

3 class B(A):
4     _c = 3

5     def __init__(self):
6         print self.a
7         print self._c

8 a = A()
9 print isinstance(a, B)

10 b = B()
11 print isinstance(b, B)

12 b = A()
13 print isinstance(b, A)
```

Em relação à sintaxe em Python, é correto afirmar que na:

- I. Linha 3 a classe B está herdando A.
- II. Linha 4 o atributo é considerado privado.
- III. Linha 5 exibe um método construtor.
- IV. Linha 9 o objeto a é uma instância da classe B.
- V. Linha 13 o objeto b é uma instância da classe A.

Marque a alternativa correta.

(Ref.: 202309808626)

- ☐ II, III e IV, apenas.
- ☐ I, II e IV, apenas.
- ☐ I, II, III e V.
- ☐ I e II, apenas.
- ☐ I e IV, apenas.

1 ponto

9. (CESGRANRIO/2019 - Adaptada) Um cientista de dados utilizou a linguagem Python para identificar outliers em um conjunto numérico. Os dados foram agrupados em uma lista da seguinte forma:

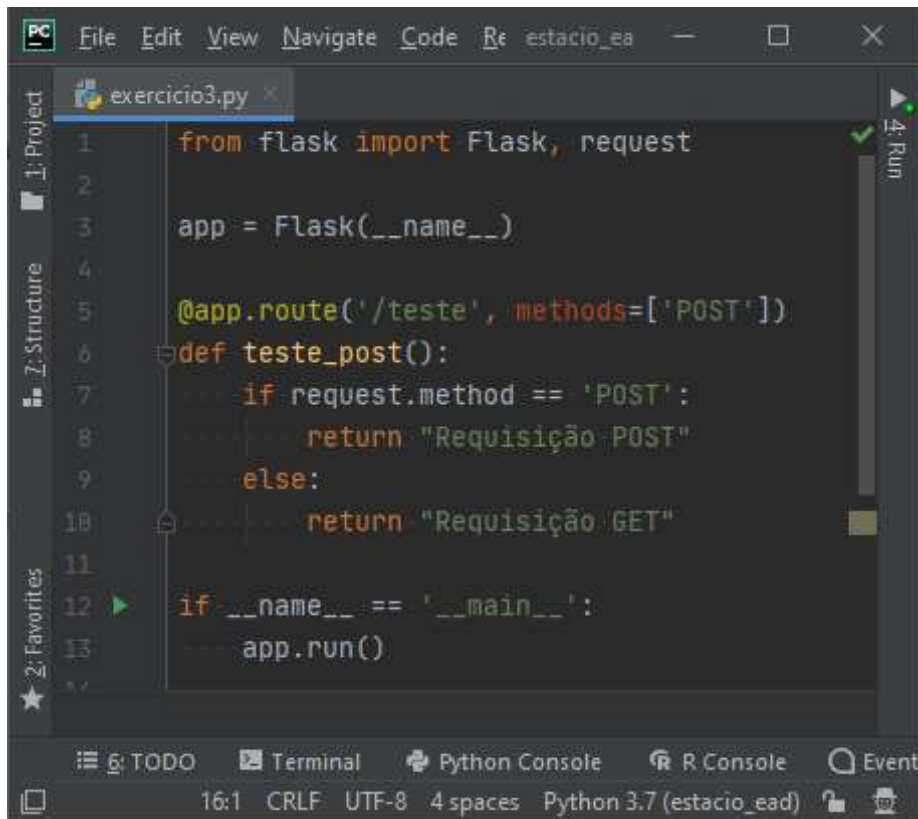
L = [-20,-10,0,10,20,30]

O comando `print(L[-1::-2])` apresentará o resultado

(Ref.: 202309809504)

- ☐ [-20, 10]
- ☐ [30]
- ☐ [30, 10, -10]
- ☐ [20]
- ☐ [30, 20]

10. Considere o código a seguir, onde temos um servidor *Flask* escutando na porta 5000, e responda:



```
1 from flask import Flask, request
2
3 app = Flask(__name__)
4
5 @app.route('/teste', methods=['POST'])
6 def teste_post():
7     if request.method == 'POST':
8         return "Requisição POST"
9     else:
10        return "Requisição GET"
11
12 if __name__ == '__main__':
13     app.run()
```

O que será apresentado no navegador se acessarmos a URL <http://127.0.0.1:5000/teste>

(Ref.: 202307314102)

- ☐ Requisição GET
- ☐ Erro 405 - Método não permitido
- ☐ Requisição POST
- ☐ Erro 404 - Página não encontrada
- ☐ Uma página em branco