



Lista Avaliativa 1

Instruções:

Resolva as questões abaixo e mande sua resolução no teams anexada na tarefa no formato de sua preferência. Pode ser em pdf, txt, zip, como um notebook jupyter, ou qualquer outra coisa que você preferir.

Entregas até dia 03 de outubro às 23:59. Não serão aceitas entregas após o prazo, fiquem atentos.

Entregas com códigos iguais a de outros alunos **não** serão aceitas. A lista é **individual**.

Podem usar as linguagens de programação que preferirem nas últimas duas questões, desde que se encaixem no paradigma.

Questão 1

```

1  package main
2
3  import "fmt"
4
5  func main() {
6      |
7      var x int
8      var y int
9      fmt.Print("Entre com um número: ")
10     fmt.Scan(&x)
11     fmt.Print("Entre com outro número:")
12     fmt.Scan(&y)
13     fmt.Println(add(x, y))
14     fmt.Println(sub(x, y))
15 }
16
17 func add(x int, y int) int {
18     | return x + y
19 }
20
21 func sub(a int, b ) int {
22     | return a-b
23 }
24
25

```

A respeito do seguinte código na linguagem GO, marque verdadeiro ou falso para as seguintes afirmações:

- () O código retornaria um erro na linha 10, pois não se pode usar '&' junto com o nome da variável.
- () Caso você entrasse com os valores '5' e '12', o valor impresso na linha 13 usando a função *add()* seria igual a 17. (Assumindo que o código rode sem erros.)
- () É permitido que declaremos uma variável sem ter que definirmos seu valor inicialmente.
- () A função *sub* está correto, pois não é sempre necessário declarar o tipo da variável.

Questão 2

```
1  class Animal:
2      def __init__(self, nome):
3          self.nome= nome
4      def p_nome(self):
5          print('Nome : ' + self.nome)
6
7
8  class Dog:
9      def __init__(self, nome, idade):
10         self.nome = nome
11         self.idade = idade
12     def print_nome(self):
13         print(self.nome + " é o nome do cachorro")
14
15
16 class BigDog(Dog):
17     def __init__(self, nome, idade, tamanho):
18         super().__init__(nome, idade)
19         self.tamanho = tamanho
20
21 my_animal = Animal('bob')
22 my_dog = Dog('willie', 6)
23 my_big_dog = BigDog('rover', 3, 'grande')
24
25 my_dog.print_nome()
26 my_big_dog.print_nome()
27 print(my_big_dog.tamanho)
28 my_animal.p_nome()
29
```

A respeito do seguinte código em Python, podemos afirmar que:

- a) A classe **Dog** herda da classe **Animal**, e é possível chamar todos os métodos da classe **Animal**.
- b) A linha 26 resultaria em um erro, pois não é possível chamar o método `print_nome` da classe **BigDog** sem declararmos esse método dentro da classe **BigDog** antes.
- c) A classe **BigDog** herda da classe **Dog**, e é possível chamar todos os métodos da classe **Dog**.

d) O método `p_nome` na linha 28 resultaria em um erro, pois a classe **Animal** não herda da classe **Dog**.

Questão 3

Faça um função que leia o raio de um círculo, e que então imprima sua área e seu diâmetro.

Depois, leia um valor do usuário e entre esse valor como parâmetro da função.

Questão 4

- Crie uma classe televisão que receba como parâmetro seu modelo.
- Dentro dessa classe, crie atributos de volume e canal. (O volume começará em 0)
- Crie um método que aumenta o volume e outro que diminua ele. Ambos receberam como parâmetro a quantidade que deverá ser aumentada ou diminuída da quantidade total.
- Crie um método para trocar de canal. Esse método receberá como parâmetro um novo canal.
- Crie um método que imprima o modelo, junto com o volume atual e o canal atual.
- Por fim, chame os métodos de aumentar e diminuir com os valores 70 e 27, respectivamente, e troque o canal para 'Canal #1', e então chame o método para imprimir tudo. Não se esqueça de criar um objeto que contenha o modelo!