

Simulado da Prova

ana.amaral@aluno.faculdadeimpacta.com.br [Alternar conta](#)



O nome, a foto e o e-mail associados à sua Conta do Google serão registrados quando você fizer upload de arquivos e enviar este formulário.

***Obrigatório**

Questões Objetivas

Esse conceito refere-se à separação dos aspectos externos de um objeto, que são acessíveis a outros objetos, dos detalhes internos da implementação, que estão escondidos de outros objetos. Isso evita que partes de um programa se tornem tão independentes que uma pequena mudança tenha grandes efeitos em cascata. O conceito citado no texto é conhecido como: *

6 pontos

- ☒ generalização
- ☐ coleção
- ☐ polimorfismo
- ☐ encapsulamento
- ☐ compartilhamento

Qual a forma correta de definir um dicionário? *

6 pontos

- ☐ `my_dict = ["key1": "Value1", "key2": "Value2", "key3": "Value3"]`
- ☒ `my_dict = {"key1": "Value1", "key2": "Value2", "key3": "Value3"}`
- ☐ `my_dict = dict("key1": "Value1", "key2": "Value2", "key3": "Value3")`
- ☐ `my_dict = ("key1": "Value1", "key2": "Value2", "key3": "Value3")`



Não é um dos princípios de orientação a objetos: *

6 pontos

- ☒ Qualquer coisa do mundo pode ser um objeto
- ☐ Classes são organizadas de forma hierárquica
- ☐ Cada objeto é originado a partir de uma classe
- ☐ Tarefas são realizadas por meio de atributos

São, dentre outros, recursos essenciais em uma aplicação orientada a objetos para se obter polimorfismo: *

6 pontos

- ☐ Herança e sobrescrita de métodos
- ☒ Herança e classes abstratas
- ☐ Classes abstratas, sem subclasses
- ☐ Classes abstratas, com métodos privados

Podem ser atributos de uma classe Professor: *

6 pontos

- ☐ Nome, falar, cor, telefone
- ☒ Nome, andar, correr, matricula
- ☐ Nome, endereço, telefone
- ☐ Nome, lecionar, CPF, RG



De acordo com o código abaixo, o que será exibido no terminal ao executar o programa? *

6 pontos

```
1  class A:
2      ...def __init__(self):
3          ...self.i = None
4
5  class B(A):
6      ...def __init__(self):
7          ...self.j = None
8
9      ...def display(self):
10         ...print(self.i)
11
12
13  obj = B()
14  obj.j = 2
15  obj.display()
16
```

- ☐ AttributeError
- ☒ 0
- ☐ Todas as alternativas estão incorretas
- ☐ 2
- ☐ 1



O dono de uma fábrica de brinquedos solicitou que seus engenheiros criassem um mesmo controle remoto para os brinquedos avião, carro e barco. A única restrição era que cada brinquedo atendesse aos comandos específicos definidos pelo controle. O controle remoto teria vários botões, sendo que todos eles seriam úteis para todos os brinquedos. Por exemplo, quando o usuário apertasse o botão mover, o controle enviaria o sinal MOVER para todos os brinquedos que estivessem em um raio de 2 metros. Desta forma, quando o brinquedo recebesse o sinal MOVER, ele se moveria de acordo com a sua função. Para o avião, mover significa VOAR, para o barco significa NAVEGAR e, para o automóvel, significa CORRER. Observe que os brinquedos respondem ao mesmo sinal de formas diferentes. Na programação orientada a objetos, este exemplo ilustra um caso de: *

6 pontos

- ☐ Herança
- ☐ Classe abstrata
- ☒ Polimorfismo
- ☐ Sobrecarga de métodos
- ☐ Hierarquia de herança

A respeito de métodos privados e de métodos públicos, assinale a opção CORRETA. *

6 pontos

- ☒ Todas as alternativas estão incorretas
- ☐ Um método privado nunca pode ser chamado dentro da classe onde se encontra.
- ☐ O método privado definido numa determinada classe é acessível a partir de qualquer outra classe com exceção do programa principal.
- ☐ Um método privado nunca pode ser chamado de uma classe externa à classe onde está definido.
- ☐ O método público definido numa determinada classe não é acessível a partir do programa principal.



De acordo com o código abaixo, o que será exibido no terminal ao executar o programa? *

6 pontos

```
1  number = 5.0
2  try:
3      r = 10/number
4      print(r)
5  except:
6      print("Oops! Error occurred.")
7
```

- ☒ None object
- ☐ Oops! Error occurred.
- ☐ 2.0
- ☐ 2.0 Oops! Error occurred.



De acordo com o código abaixo, o que será exibido no terminal ao executar o programa? *

6 pontos

```
1  class XPTO:
2      def __init__(self, ctx):
3          self.__msg = "Echo"
4          if (ctx):
5              self.__var1 = 3
6              self.__var2= 0.0
7          else:
8              self.__var1 = 3
9              self.__var2 = 2.0
10
11     def get_var1(self):
12         return self.__var1
13
14     def get_var2(self):
15         return self.__var2
16
17     def susp(self, par):
18         val = 1
19         if (par % 3 != 0):
20             val += par % 2
21         else:
22             val += par * 2
23         return val
24
25 my_obj = XPTO(True)
26 print(my_obj.susp(4))
27
```

- ☐ 3
- ☒ 1
- ☐ 7
- ☐ 6
- ☐ 2



Google Formulários

