

# Prova 2

Segunda prova da disciplina Programação 3.

\* Required

1. Email \*

---

Classe Contador: Essa classe representa um contador manual, que pode ser incrementado com cliques ou reiniciado para a contagem inicial de zero elementos.

```
class Contador:
    def __init__(self):
        self._contagem = 0

    def clique(self):
        self._contagem += 1

    def reiniciar(self):
        self._contagem = 0

    def get_contagem(self):
        return self._contagem

# EXEMPLO DE USO
c = Contador()
c.get_contagem() # 0
c.clique()
c.clique()
c.get_contagem() # 2
c.reiniciar()
c.get_contagem() # 0
```

2. Implemente, para a classe Contador, um novo método chamado "desfazer" que permita ao usuário do contador desfazer contagens que foram feitas com os cliques (uma de cada vez). Certifique-se de que não é possível desfazer uma contagem quando o contador já está em zero. A imagem a seguir exemplifica o método "desfazer".

Sua resposta pode ser apenas o método implementado, não precisa copiar o código completo da classe.

```
c = Contador()
c.clique()
c.clique()
c.get_contagem() # 2
c.desfazer()
c.get_contagem() # 1
c.desfazer()
c.get_contagem() # 0
c.desfazer()
c.get_contagem() # 0
```

---

---

---

---

---

3. A imagem a seguir mostra uma modificação feita na classe Contador. Foram adicionados um novo atributo "\_max" e um novo método "set\_max" para atualizar esse mesmo atributo com um valor escolhido pelo usuário. Além disso, o método "clique" teve sua lógica alterada. Analise o código e responda: como o atributo "\_max" altera o funcionamento do contador?

```
class Contador:
    def __init__(self):
        self._contagem = 0
        self._max = 0

    def clique(self):
        if self._max == 0:
            self._contagem += 1
        else:
            if self._contagem < self._max:
                self._contagem += 1
            else:
                print("Limite excedido.")

    def reiniciar(self):
        self._contagem = 0

    def get_contagem(self):
        return self._contagem

    def set_max(self, max):
        self._max = max
```

---

---

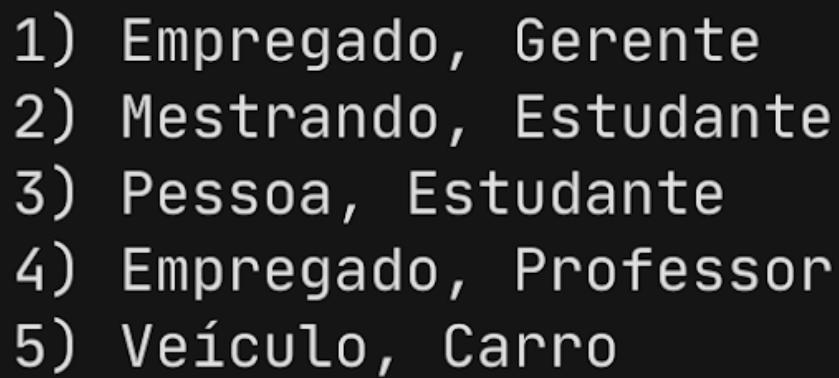
---

---

---

4. Identifique, para cada par de classes com uma relação de herança mostrado na imagem a seguir, o elemento que representa a superclasse.

Sua resposta deve ser no formato "superclasse 1, superclasse 2, superclasse 3, superclasse 4, superclasse 5".  
Lembre-se: a superclasse é sempre a mais genérica!

- 
- 1) Empregado, Gerente
  - 2) Mestrando, Estudante
  - 3) Pessoa, Estudante
  - 4) Empregado, Professor
  - 5) Veículo, Carro

---

Escolha os termos que correspondem às definições dadas a seguir. Cada questão possui apenas uma resposta correta.

5. Método especial usado para criar uma representação de um objeto em formato de string.

*Mark only one oval.*

- ☐ repr
- ☐ tupla
- ☐ super
- ☐ interface pública
- ☐ dicionário
- ☐ busca reversa
- ☐ herança
- ☐ módulo
- ☐ dinâmica

6. Sequência imutável de elementos.

*Mark only one oval.*

- ☐ repr
- ☐ tupla
- ☐ super
- ☐ interface pública
- ☐ dicionário
- ☐ busca reversa
- ☐ herança
- ☐ módulo
- ☐ dinâmica

7. Função usada dentro de uma subclasse para chamar métodos da superclasse.

*Mark only one oval.*

- ☐ repr
- ☐ tupla
- ☐ super
- ☐ interface pública
- ☐ dicionário
- ☐ busca reversa
- ☐ herança
- ☐ módulo
- ☐ dinâmica

8. Coleção de métodos e atributos de livre acesso, que podem ser usados para manipular um determinado objeto.

*Mark only one oval.*

- ☐ repr
- ☐ tupla
- ☐ super
- ☐ interface pública
- ☐ dicionário
- ☐ busca reversa
- ☐ herança
- ☐ módulo
- ☐ dinâmica

9. Tipo que representa um mapeamento de chaves aos seus valores correspondentes.

*Mark only one oval.*

- ☐ repr
- ☐ tupla
- ☐ super
- ☐ interface pública
- ☐ dicionário
- ☐ busca reversa
- ☐ herança
- ☐ módulo
- ☐ dinâmica

10. Operação de dicionário que recebe um valor e encontra uma ou várias chaves que o mapeiem.

*Mark only one oval.*

- ☐ repr
- ☐ tupla
- ☐ super
- ☐ interface pública
- ☐ dicionário
- ☐ busca reversa
- ☐ herança
- ☐ módulo
- ☐ dinâmica



11. Princípio da Programação Orientada a Objetos no qual é definida a criação de uma subclasse que obtém toda a funcionalidade da classe da qual ela deriva, podendo modificá-la e/ou estendê-la.

*Mark only one oval.*

- ☐ repr
- ☐ tupla
- ☐ super
- ☐ interface pública
- ☐ dicionário
- ☐ busca reversa
- ☐ herança
- ☐ módulo
- ☐ dinâmica

12. Coleção de funções e classes intimamente relacionadas que pode ser importada em um programa.

*Mark only one oval.*

- ☐ repr
- ☐ tupla
- ☐ super
- ☐ interface pública
- ☐ dicionário
- ☐ busca reversa
- ☐ herança
- ☐ módulo
- ☐ dinâmica

13. Tipo de busca de métodos utilizada em uma linguagem orientada a objetos, que permite a existência do polimorfismo.

*Mark only one oval.*

- ☐ repr
- ☐ tupla
- ☐ super
- ☐ interface pública
- ☐ dicionário
- ☐ busca reversa
- ☐ herança
- ☐ módulo
- ☐ dinâmica

---

This content is neither created nor endorsed by Google.

Google Forms