

	<p align="center">CAPACITAÇÃO EM TECNOLOGIA, GESTÃO E COMPUTAÇÃO</p> <p align="center">CURSO: Desenvolvimento de Software</p>	
---	---	--

Aluno(a): Ayrton Araújo	Data: 07/05/2025
Professor(a): Jonatas S. Lopes	

Exercício valendo presença (TEMA: Python e Banco de Dados)

1. Em python, o que esse código está fazendo?

```
for key, item in fruits.items():
    print(f"Esta é a {chave} com o valor {valor}")
```

- a) Ele está iterando uma lista de frutas e imprimindo o índice e o nome de cada fruta.
- b) Ele está iterando os itens do dicionário "fruits" e printa cada chave e valor.**
- c) Ele está acessando apenas os valores do dicionário "fruits" e imprimindo-os em ordem alfabética.
- d) Ele está criando um dicionário com as chaves e valores de "fruits" e imprimindo cada item.

2. Qual a palavra reservada em python que cria uma função nomeada?

- a) def**
- b) func
- c) function
- d) lambda

3. Dado o código abaixo, o que está errado?

```
def saudacao():
    print("olá mundo")

saudacao
```

- a) A função print está sendo usada de forma incorreta; deveria ser print{"olá mundo"}.
- b) Falta o uso da palavra return para que a função funcione corretamente.
- c) A função saudacao não está sendo chamada. É necessário usar parênteses: saudacao().**
- d) O nome da função saudacao precisa estar entre aspas para ser chamado corretamente.

4. Quando se utiliza *args e **kwargs em Python?

- a) *args é usado para passar um número variável de argumentos posicionais, e **kwargs para um número variável de argumentos nomeados (por chave e valor) em uma função.**
- b) *args é usado para importar várias bibliotecas ao mesmo tempo, e **kwargs para criar funções anônimas.
- c) *args define argumentos obrigatórios e **kwargs define argumentos opcionais com valor padrão.
- d) *args serve para criar listas e **kwargs serve para criar dicionários dentro da função.

5. Em programação orientada a objetos, qual a diferença entre classe e objeto?

- a) A classe é o código principal do programa e o objeto é o arquivo onde ela está salva.
- b) A classe representa uma função específica, e o objeto é uma variável que armazena essa função.
- c) A classe é criada automaticamente pelo Python quando o programa roda, e o objeto precisa ser declarado manualmente.
- d) A classe é um molde ou modelo que define as características e comportamentos, enquanto o objeto é uma instância concreta dessa classe, com valores próprios.

6. Qual é a palavra reservada em Python para o construtor de uma classe?

- a) __construct__
- b) __start__
- c) constructor
- d) __init__

7. O que é self numa classe em Python?

- a) É uma palavra reservada usada para importar métodos de outras classes.
- b) É uma função interna do Python que cria novos objetos dentro da classe.
- c) É uma variável global que representa o nome da classe no momento da execução.
- d) É uma referência à instância atual da classe e é usada para acessar variáveis que pertencem à classe.

8. Observe o código em Python e responda: Que mecanismo está sendo usado?

```
class SerVivo:
    def respira(self):
        pass

class Humano(SerVivo):
    def __init__(self, nome):
        self.nome = nome

pessoa = Humano('Fulano')
pessoa.respira()
```

- a) Herança
- b) Encapsulamento
- c) Polimorfismo
- d) Instanciação múltipla

9. Preencha as lacunas da frase com as palavras corretas:

Encapsulamento é o conceito de _____ os detalhes internos de um objeto e _____ apenas o que é necessário. Em Python, isso é conseguido através de convenções como o uso de _____ para indicar atributos e métodos _____.

- a) mostrar, ocultar, nomes em caixa alta, públicos
- b) esconder, expor, prefixos de sublinhado, privados
- c) proteger, liberar, asteriscos, internos
- d) separar, agrupar, parênteses, protegidos

10. No conceito de encapsulamento em Python, qual é o nome dado aos métodos utilizados para acessar (ler) atributos privados de uma classe?

- a) Getters
- b) Private methods
- c) Setters
- d) Public methods

11. No conceito de encapsulamento em Python, qual é o nome dado aos métodos utilizados para modificar/alterar atributos privados de uma classe?

- a) Getters
- b) Private methods
- c) Setters
- d) Public methods

12. No contexto de Ciência de Dados, qual é a sequência correta?

- a) Dado -> Informação -> Conhecimento
- b) Conhecimento -> Dado -> Informação
- c) Informação -> Dado -> Conhecimento
- d) Dado -> Conhecimento -> Informação

13. Qual dos exemplos abaixo é caracterizado por um banco de dados computacional não gerenciado por SGBD?

- a) Ata física
- b) Planilha excel
- c) Login da trivago
- d) Catálogo de vídeos do youtube

14. Qual dos exemplos abaixo é caracterizado por um banco de dados não-computacional?

- a) Ata física
- b) Planilha excel
- c) Login da trivago
- d) Catálogo de vídeos do youtube

15. “É a representação genérica de um componente do mundo real, sobre o qual desejamos armazenar informações”. No contexto de modelagem de dados, essa é a definição de _____ .

- a) Atributo
- b) Tabela
- c) Relacionamento
- d) Entidade

16. No contexto de modelagem de dados, o que é um atributo multivalorado?

- a) É um atributo que armazena apenas valores numéricos múltiplos, como inteiros ou decimais.
- b) É um atributo que contém valores duplicados em diferentes instâncias da entidade.
- c) É um atributo de uma entidade em um banco de dados que pode conter múltiplos valores para uma única instância dessa entidade.
- d) É um atributo calculado com base em outros atributos de uma entidade.

17. Em modelagem de dados, qual a diferença entre atributos multivalorados e atributos compostos?

- a) Ambos armazenam vários valores, mas os atributos compostos são usados apenas para números e os multivalorados apenas para textos.
- b) Atributos multivalorados podem guardar vários valores diferentes pra uma mesma entidade (tipo: telefone — a pessoa pode ter mais de um número). Já os atributos compostos são aqueles que podem ser quebrados em partes menores (tipo: endereço — que vira rua, número, bairro etc.).
- c) Atributos compostos precisam ser normalizados, enquanto atributos multivalorados não são usados em bancos de dados relacionais.
- d) Atributos multivalorados formam relacionamentos entre tabelas, enquanto atributos compostos são sempre chaves primárias.

18. Qual é a diferença entre chave primária natural e chave primária artificial em bancos de dados?

- a) A chave primária natural é gerada automaticamente pelo sistema, enquanto a artificial é definida manualmente pelo usuário.
- b) A chave primária artificial precisa ser composta por mais de um atributo, e a natural é sempre um único campo.
- c) A chave primária natural nunca pode ser usada como identificador único em tabelas relacionais.
- d) A chave primária natural é formada por atributos já existentes na entidade e com significado real, e a artificial é criado automaticamente e exclusivamente para ser chave.

19. Seguindo o processo de modelagem de dados, qual é a ordem correta das etapas, partindo da análise das regras de negócio até a implementação no banco de dados?

- a) Modelagem Física → Modelagem Conceitual → Modelagem Lógica
- b) Modelagem Lógica → Modelagem Conceitual → Modelagem Física
- c) Modelagem Física → Modelagem Lógica → Modelagem Conceitual
- d) Modelagem Conceitual → Modelagem Lógica → Modelagem Física

20. Em modelagem de dados, qual das cardinalidades abaixo geralmente requer a criação de uma tabela associativa para representar a relação entre duas entidades?

- a) 1:1 (um-para-um)
- b) n:n (muitos-para-muitos)
- c) 1:n (um-para-muitos)
- d) n:1 (muitos-para-um)

Bom exercício!!