

## Exercício - Classes

O notebook Jupyter deste exercício pode ser acessado [clikando aqui](#).

1) Vamos criar aos pouquinhos uma classe Evento que registra eventos de um calendário. Pra começar, nessa questão você deve:

- Criar uma classe chamada Evento
- Adicionar um atributo de classe `total_eventos` que será usado para contar o número total de eventos (instâncias de classe) criadas.
- Implemente o método construtor que deve receber os parâmetros `titulo` (string), `data_hora` (datetime), `descricao` (string). Crie e inicialize um atributo de instância para cada um dos parâmetros citados.
- O construtor também deve inicializar um atributo de instância `is_concluido = False`. Esse atributo será utilizado mais à frente.
- No construtor você também deve incrementar o atributo de classe `total_eventos` em 1 a cada nova instância criada.

Teste o seu código criando duas diferentes instâncias de evento (preencha os atributos como quiser) e imprimindo todos os seus atributos (de classe e de instância).

Consulte [este tutorial biblioteca datetime](#) caso não esteja familiarizado.

2) Agora vamos adicionar métodos à nossa classe. Lembre dos diferentes decoradores que aprendemos. Você deve redefinir a classe com os seguintes métodos:

- Um método `isConcluido()` que avalia se a `data_hora` do evento é menor que `datetime.now()` (a data e hora atual). Em caso positivo, atualiza o atributo de instância `is_concluido` para o valor `True`.
- Um método de classe `num_eventos()` que retorna o valor do atributo de classe `total_eventos`.
- Um método estático `valida_evento(nome, data_hora, descricao)` que recebe os atributos de um evento e testa os tipos de cada variável, retornando `True` caso todos estejam corretos e `False` caso contrário. Para o teste, use a função nativa do Python [isinstance](#).

Para testar sua classe atualizada:

- Crie uma instância de `Evento` com valor passado e chame o método `isConcluido()` para a instância criada. Em seguida imprima o atributo `is_concluido`.
- Invoque o método de classe `num_eventos()` a partir da classe `Evento` (ou seja, sem criar nenhuma instância).

- Chame o método estático `valida_evento()` a partir da classe `Evento`. Experimente passar valores corretos e incorretos.

### 3) Vamos incluir métodos mágicos! ✨🔮🔮

Redefina a classe incluindo:

- Método `__str__` que imprime os atributos do evento na forma "Evento: titulo, Data: data\_hora, Descrição: descricao, Concluido: is\_concluido".
- Implemente os métodos de comparação `__eq__`, `__ne__`, `__lt__`, `__le__`, `__gt__` e `__ge__` para comparar eventos baseados no atributo `data_hora`. Esses métodos devem comparar duas instâncias de `Evento` e retornar os resultados apropriados (`True` ou `False`).

Para testar, crie duas instâncias de `Evento` com datas diferentes. Imprima as instâncias com a função `print()` e apresente o resultado das comparações entre eventos (`==`, `!=`, `<`, `<=`, `>`, `>=`).

**Instruções para submissão:** Baixe o arquivo 2.04-Classes.ipynb (link a seguir), abra para edição no Jupyter Notebook, e preencha as respostas das atividades.

<https://github.com/camilalaranjeira/python-intermediario-exercicios/blob/main/modulo2/2.04-Classes.ipynb>

**Instruções para submissão:** envie o arquivo acima preenchido com as suas respostas para a pasta modulo2 do seu repositório.