A classe datetime é um tipo de dado que representa uma data e hora específicas em Python. Ela permite trabalhar e manipular dados de ano, mês, dia, hora, minuto, segundo e microssegundo, além de dias da semana, como segunda, terça, quarta, etc. Essa classe é definida no módulo datetime e é fundamental na manipulação de datas e horários em Python. Com ela, é possível realizar operações de cálculo de diferenças entre datas, formatação de datas e horas em diferentes formatos, além de ser muito útil em análises de dados que envolvem séries temporais.

Com a biblioteca Pandas, é possível realizar diversas operações com datas e horários, como a conversão de dados de string para datetime, a filtragem de dados baseada em intervalos de tempo específicos, a agregação de dados por hora, dia, mês ou ano, entre outras funcionalidades. Desse modo, com essa biblioteca, é possível realizar diversas operações com datas de forma simples e eficiente dentro de um conjunto de dados.

## **Biblioteca datetime**

Conseguimos trabalhar diretamente com o datetime através da biblioteca datetime, uma biblioteca padrão do Python que fornece classes para trabalhar com datas e horas. Com essa biblioteca, é possível criar objetos de data e hora, realizar cálculos de tempo, formatar datas e horas em diferentes formatos e muito mais.

Dentro da biblioteca datetime, existe a classe datetime, que representa uma data e hora específicas. Veja um exemplo:

import datetime

# criando um objeto datetime com a data e hora atual

agora = datetime.datetime.now()

print("Data e hora atual:", agora)

Neste exemplo, o método now() da classe datetime é usado para criar um objeto que representa a data e a hora atual. O objeto resultante é armazenado na variável agora e, em seguida, é impresso na tela usando a função print().

Outra classe importante na biblioteca datetime é a classe date, que representa apenas uma data. Veja um exemplo:

import datetime

# criando um objeto date com a data de hoje

hoje = datetime.date.today()

print("Data de hoje:", hoje)

Neste exemplo, o método today() da classe date é usado para criar um objeto que representa a data de hoje. O objeto resultante é armazenado na variável hoje e, em seguida, é impresso na tela usando a função print().

A biblioteca datetime também permite realizar operações matemáticas com datas e horas. Veja um exemplo de como calcular a diferença entre duas datas:

import datetime

# criando dois objetos date com datas diferentes

data\_1 = datetime.date(2022, 1, 1)

data\_2 = datetime.date(2023, 1, 1)

# calculando a diferença entre as duas datas

diferenca = data\_2 - data\_1

print("Diferença entre as duas datas:", diferenca)

Neste exemplo, dois objetos date são criados com datas diferentes. Em seguida, o operador de subtração é usado para calcular a diferença entre as duas datas. O resultado é armazenado na variável diferenca e, em seguida, é impresso na tela usando a função print().

Para se aprofundar nos usos do tipo datetime consulte o artigo [Python datetime: Como faço para definir data e hora em Python?](https://www.alura.com.br/artigos/lidando-com-datas-e-horarios-no-python) e também a documentação da [biblioteca datetime](https://docs.python.org/3/library/datetime.html).