

- Diferença entre Python time, datetime e pendulum
  - As bibliotecas time e datetime são módulos nativos do Python que permitem trabalhar com unidades de tempo, como segundos, minutos, horas, dias, meses e anos.
  - A biblioteca Pendulum
  - Comparação entre as bibliotecas Python Time, Datetime e Pendulum com base nos critérios:
- Exemplos
  - Importando libs
  - Obter a data e hora atual no fuso horário local
    - Usando time
    - Usando datetime
    - Usando pendulum
  - Obter a diferença entre duas datas
    - Usando time
    - Usando datetime
    - Usando pendulum
  - Adicionar ou subtrair uma unidade de tempo a uma data ou hora
    - Usando time
    - Usando datetime
    - Usando pendulum
  - Fontes

## Diferença entre Python **time**, **datetime** e **pendulum**

---

**As bibliotecas **time** e **datetime** são módulos nativos do Python que permitem trabalhar com unidades de tempo, como segundos, minutos, horas, dias, meses e anos.**

---

A principal diferença entre elas é que:

- O módulo `time` é focado em trabalhar com timestamps Unix, que são números de ponto flutuante que representam o tempo em segundos desde a época Unix (00:00:00 UTC, 1 de janeiro de 1970).
- O módulo `datetime` pode suportar muitas das mesmas operações, mas fornece um conjunto de tipos mais orientados a objetos, como `date`, `time`, `datetime` e `timedelta`, e também tem algum suporte limitado para fusos horários.

. A biblioteca `pendulum` é uma biblioteca externa que visa fornecer uma alternativa melhor e mais fácil de usar do que as bibliotecas nativas.

## A biblioteca Pendulum

---

Ela é compatível com a API do `datetime`, mas tem algumas vantagens, como:

- Suporte completo para fusos horários e horário de verão, usando a [base de dados IANA](#).
- Criação e análise simplificadas de objetos de data e hora, usando strings naturais e formatos flexíveis.
- Manipulação conveniente de objetos de data e hora, usando [métodos encadeados](#) e [aritmética intuitiva](#).
- Formatação e humanização personalizáveis de objetos de data e hora, usando [localização](#) e [gramática](#).

## Comparação entre as bibliotecas Python Time, Datetime e Pendulum com base nos critérios:

---

### 1. Adaptabilidade:

- **Time:** A biblioteca Time oferece mecanismos para obter os valores do tempo no Python. No entanto, ela não é tão adaptável quando se trata de manipulações de data e hora mais complexas.
- **Datetime:** O módulo Datetime é um dos módulos integrados mais importantes em Python. É muito flexível e poderoso, pois fornece muitas soluções prontas para uso para problemas reais de programação.

- **Pendulum:** O Pendulum é uma excelente substituição para o módulo datetime do Python. Ele resolve todos os problemas do módulo datetime integrado e é muito adaptável, especialmente quando se lida com fusos horários.

## 2. Recursividade:

- Todas as três bibliotecas podem ser usadas de maneira recursiva, dependendo das necessidades do código.

## 3. Velocidade de processamento:

- A velocidade de processamento pode variar dependendo da complexidade das operações que realizadas. No entanto, todas as três bibliotecas são geralmente eficientes em termos de velocidade.

## 4. Usabilidade:

- **Time:** A biblioteca Time é bastante simples de usar para obter valores de tempo.
- **Datetime:** O módulo Datetime é muito intuitivo e fácil de usar para a maioria das operações de data e hora.
- **Pendulum:** O Pendulum fornece uma API mais limpa e fácil de usar. Ele simplifica o problema de manipulações de datas complexas envolvendo fusos horários.

[Fonte das informações](#)

# Exemplos

---

- Códigos que mostram como usar as três bibliotecas para realizar algumas tarefas comuns:

## Importando libs

```
import time
import datetime
import pendulum
```

# Obter a data e hora atual no fuso horário local

---

## Usando time

```
t = time.time()
local_time = time.localtime(t)
print(time.strftime('%Y-%m-%d %H:%M %Z', local_time))
# 2023-10-09 14:28 BRT
```

## Usando datetime

```
now = datetime.datetime.now()
print(now.strftime('%Y-%m-%d %H:%M %Z'))
# 2023-10-09 14:28
```

## Usando pendulum

```
now = pendulum.now()
print(now.format('YYYY-MM-DD HH:mm Z'))
# 2023-10-09 14:28 -03:00
```

# Obter a diferença entre duas datas

---

## Usando time

```
t1 = time.mktime((2023, 1, 1, 0, 0, 0, 0, 0, 0))
t2 = time.mktime((2023, 12, 31, 23, 59, 59, 0, 0, 0))
diff = t2 - t1
print(diff)
# 31535999.0 segundos
```

# Usando datetime

```
d1 = datetime.date(2023, 1, 1)
d2 = datetime.date(2023, 12, 31)
diff = d2 - d1
print(diff)
# 364 dias
```

# Usando pendulum

```
d1 = pendulum.date(2023, 1, 1)
d2 = pendulum.date(2023, 12, 31)
diff = d2 - d1
print(diff)
# 364 dias
print(diff.in_words())
# 11 meses e 30 dias
```

# Adicionar ou subtrair uma unidade de tempo a uma data ou hora

---

# Usando time

```
t = time.mktime((2023, 10, 9, 14, 28, 45, 0, 0, 0))
t_plus_1h = t + 3600 # Adicionar uma hora em segundos
t_minus_1d = t - 86400 # Subtrair um dia em segundos
print(time.strftime('%Y-%m-%d %H:%M', time.localtime(t_plus_1h)))
# 2023-10-09 15:28
print(time.strftime('%Y-%m-%d %H:%M', time.localtime(t_minus_1d)))
# 2023-10-08 14:28
```

# Usando datetime

```
dt = datetime.datetime(2023, 10, 9, 14, 28, 45)
dt_plus_1h = dt + datetime.timedelta(hours=1) # Adicionar uma hora usando timedelta
```

```
dt_minus_1d = dt - datetime.timedelta(days=1) # Subtrair um dia usando timedelta
print(dt_plus_1h.strftime('%Y-%m-%d %H:%M'))
# 2023-10-09 15:28
print(dt_minus_1d.strftime('%Y-%m-%d %H:%M'))
# 2023-10-08 14:28
```

## Usando pendulum

```
dt = pendulum.datetime(2023, 10, 9, 14, 28, 45)
dt_plus_1h = dt.add(hours=1) # Adicionar uma hora usando método add
dt_minus_1d = dt.subtract(days=1) # Subtrair um dia usando método subtract
print(dt_plus_1h.format('YYYY-MM-DD HH:mm'))
# 2023-10-09 15:28
print(dt_minus_1d.format('YYYY-MM-DD HH:mm'))
# 2023-10-08 14:28
```

## Fontes

---

- [Lidando com datas no Python: as bibliotecas time e datetime.](#)
- [Pendulum: provavelmente a melhor biblioteca de DateTime em Python.](#)
- [datetime — Tipos básicos de data e hora — documentação Python 3.12.0.](#)
- [Python - Módulo Pendulum – Acervo Lima.](#)