

Acadêmicos: João Vítor Oliveira da Inhaia e Raquel Marafon

Disciplina: Programação Web I

Professor: Rafael Vivian

Data: 21/08/2025

## Manipulação de Datas e Horas em Python utilizando o módulo datetime

### 1. Importando o módulo

```
from datetime import datetime, timedelta  
import locale
```

O `datetime` é um módulo da biblioteca padrão do Python (não precisa instalar nada).

Dentro dele existem várias classes para lidar com datas e horários.==

Aqui usamos:

- **datetime** → representa datas e horas completas (ano, mês, dia, hora, minuto, segundo).
- **timedelta** → representa intervalos de tempo (diferença em dias, horas, segundos etc.).
- **locale** → permite exibir meses/dias por extenso em português.

---

### 2. Criar uma data/hora atual

```
agora = datetime.now()  
print("Data e hora atuais:", agora.strftime("%d/%m/%Y %H:%M:%S"), "\n")
```

- `datetime.now()` pega a data e hora exata do sistema.
  - Exemplo de saída: 21/08/2025 08:15:32.
-

### 3. Criar uma data específica

```
data_especifica = datetime(2025, 12, 25, 20, 30, 0)
print("Data específica (Natal):", data_especifica.strftime("%d/%m/%Y %H:%M:%S"))
print("Data específica (por extenso):", data_especifica.strftime("%d de %B de %Y, %H:%M:%S"), "\n")
```

- Aqui passamos: ano, mês, dia, hora, minuto, segundo.  
Exemplo: 2025-12-25 20:30:00 (Natal às 20h30).
  - O uso de `strftime("%d de %B de %Y")` exibe a data **por extenso**:  
25 de dezembro de 2025, 20:30:00.
- 

### 4. Formatar datas para string (exibição)

```
data_formatada = agora.strftime("%d/%m/%Y %H:%M:%S")
print("Data atual formatada:", data_formatada, "\n")
```

- `strftime` (string format time) transforma uma data em texto formatado.
- Exemplos de códigos:
  - `%d` → dia
  - `%m` → mês
  - `%Y` → ano
  - `%H:%M:%S` → hora:minuto:segundo

Exemplo de saída: 21/08/2025 08:15:32.

---

### 5. Converter string em data

```
data_str = "21/08/2025 14:30:00"
data_convertida = datetime.strptime(data_str, "%d/%m/%Y %H:%M:%S")
print("String original:", data_str)
print("Convertida (formato numérico):", data_convertida.strftime("%d/%m/%Y %H:%M:%S"))
```

```
print("Convertida (por extenso):", data_convertida.strftime("%d de %B de %Y, %H:%M:%S"), "\n")
```

- `strptime` (string parse time) faz o contrário do `strftime`: transforma texto em data.
  - Exemplo: `"21/08/2025 14:30:00" → datetime(2025, 8, 21, 14, 30, 0)`.
- 

## 6. Calcular diferença entre duas datas

```
diferenca = data_especifica - agora
print("Diferença até o Natal:", diferenca)          # mostra dias + horas + minutos
print("Dias até o Natal:", diferenca.days, "\n")
```

- Subtraindo datas, obtemos um objeto `timedelta`.
  - Ele mostra quantos dias, horas, minutos, segundos faltam.
  - `.days` retorna apenas os dias inteiros.
  -
- 

## 7. Somar ou subtrair tempo (`timedelta`)

```
amanha = agora + timedelta(days=1)
ontem = agora - timedelta(days=1)
print("Amanhã será:", amanhã.strftime("%d/%m/%Y %H:%M:%S"))
print("Ontem foi:", ontem.strftime("%d/%m/%Y %H:%M:%S"), "\n")
```

- `timedelta(days=1)` representa **1 dia**.
- Também podemos usar `hours`, `minutes`, `seconds`.

Isso serve para calcular prazos, agendamentos, etc.

```
mais_tempo = agora + timedelta(days=7, hours=3, minutes=15)
```

```
print("Daqui a 7 dias, 3h e 15min será:", mais_tempo.strftime("%d/%m/%Y %H:%M:%S"))
print("Daqui a 7 dias, 3h e 15min (por extenso):", mais_tempo.strftime("%d de %B de %Y, %H:%M:%S"))
```

Aqui combinamos **dias + horas + minutos** em um mesmo cálculo.

Exemplo: 28/08/2025 11:45:32 → 28 de agosto de 2025, 11:45:32.

## CÓDIGO

```
# =====  
# Manipulação de Datas e Horas em Python  
# Usando o módulo datetime  
# =====  
  
from datetime import datetime, timedelta  
import locale  
  
# Configurar locale para português (meses/dias por extenso)  
# OBS: se não funcionar no Windows, use  
"Portuguese_Brazil.1252"  
locale.setlocale(locale.LC_TIME, "pt_BR.utf8")  
  
# 1. PEGAR DATA E HORA ATUAIS  
agora = datetime.now()  
print("Data e hora atuais:", agora.strftime("%d/%m/%Y  
%H:%M:%S"), "\n")  
  
# 2. CRIAR UMA DATA ESPECÍFICA  
data_especifica = datetime(2025, 12, 25, 20, 30, 0)  
print("Data específica (Natal):",  
data_especifica.strftime("%d/%m/%Y %H:%M:%S"))  
print("Data específica (por extenso):",  
data_especifica.strftime("%d de %B de %Y, %H:%M:%S"), "\n")  
  
# 3. FORMATAR DATA PARA STRING  
data_formatada = agora.strftime("%d/%m/%Y %H:%M:%S")  
print("Data atual formatada:", data_formatada, "\n")  
  
# 4. CONVERTER STRING PARA DATETIME  
data_str = "21/08/2025 14:30:00"  
data_convertida = datetime.strptime(data_str, "%d/%m/%Y  
%H:%M:%S")  
print("String original:", data_str)  
print("Convertida (formato numérico):",  
data_convertida.strftime("%d/%m/%Y %H:%M:%S"))  
print("Convertida (por extenso):",  
data_convertida.strftime("%d de %B de %Y, %H:%M:%S"), "\n")  
  
# 5. CALCULAR DIFERENÇA ENTRE DUAS DATAS  
diferenca = data_especifica - agora
```

```
print("Diferença até o Natal:", diferenca)    # mostra dias +
horas + minutos
print("Dias até o Natal:", diferenca.days, "\n")
```

#### **# 6. SOMAR E SUBTRAIR TEMPO COM timedelta**

```
amanha = agora + timedelta(days=1)
ontem = agora - timedelta(days=1)
print("Amanhã será:", amanhã.strftime("%d/%m/%Y %H:%M:%S"))
print("Ontem foi:", ontem.strftime("%d/%m/%Y %H:%M:%S"), "\n")
```

#### **# 7. EXEMPLO MAIS COMPLETO DE timedelta**

```
mais_tempo = agora + timedelta(days=7, hours=3, minutes=15)
print("Daqui a 7 dias, 3h e 15min será:",
mais_tempo.strftime("%d/%m/%Y %H:%M:%S"))
print("Daqui a 7 dias, 3h e 15min (por extenso):",
mais_tempo.strftime("%d de %B de %Y, %H:%M:%S"))
```