

taller1_jupyter

May 3, 2025

Taller 1:

Estudiante: Damián Rosero Vinueza

Descargamos las librerías iniciales para crear los datos con Faker, Pandas para trabajar con los datos y numpy para las operaciones matemáticas

```
[1]: !pip install Faker pandas numpy
```

Collecting Faker

Using cached faker-37.1.0-py3-none-any.whl.metadata (15 kB)

Collecting pandas

Using cached pandas-2.2.3-cp312-cp312-win_amd64.whl.metadata (19 kB)

Collecting numpy

Using cached numpy-2.2.5-cp312-cp312-win_amd64.whl.metadata (60 kB)

Collecting tzdata (from Faker)

Using cached tzdata-2025.2-py2.py3-none-any.whl.metadata (1.4 kB)

Requirement already satisfied: python-dateutil>=2.8.2 in

c:\utpl\hia\taller1\lib\site-packages (from pandas) (2.9.0.post0)

Collecting pytz>=2020.1 (from pandas)

Using cached pytz-2025.2-py2.py3-none-any.whl.metadata (22 kB)

Requirement already satisfied: six>=1.5 in c:\utpl\hia\taller1\lib\site-packages (from python-dateutil>=2.8.2->pandas) (1.17.0)

Using cached faker-37.1.0-py3-none-any.whl (1.9 MB)

Using cached pandas-2.2.3-cp312-cp312-win_amd64.whl (11.5 MB)

Using cached numpy-2.2.5-cp312-cp312-win_amd64.whl (12.6 MB)

Using cached pytz-2025.2-py2.py3-none-any.whl (509 kB)

Using cached tzdata-2025.2-py2.py3-none-any.whl (347 kB)

Installing collected packages: pytz, tzdata, numpy, pandas, Faker

```
----- 0/5 [pytz]
----- 0/5 [pytz]
----- 1/5 [tzdata]
----- 1/5 [tzdata]
----- 2/5 [numpy]
----- 2/5 [numpy]
----- 2/5 [numpy]
----- 2/5 [numpy]
----- 2/5 [numpy]
----- 2/5 [numpy]
```

[illegible]

[illegible]

Successfully installed Faker-37.1.0 numpy-2.2.5 pandas-2.2.3 pytz-2025.2
tzdata-2025.2

Importamos las librerías

```
[1]: import pandas as pd
import numpy as np
from faker import Faker
import random
```

Creamos 1000 registros para el dataset

```
[2]: num_registros = 1000
```

Generamos una lista con los tipos de productos que va a tener el dataset

```
[3]: productos_lista = ['Laptop', 'Teclado', 'Mouse', 'Monitor', 'Webcam',  
    ↪ 'Microfono', 'Impresora', 'Router', 'Disco Duro', 'USB', 'Auriculares',  
    ↪ 'Altavoces']
```

Generamos un diccionario con los ´precios de los productos

```
[5]: precios_lista = {
    'Laptop': 1200, 'Teclado': 75, 'Mouse': 25, 'Monitor': 300, 'Webcam': 50,
    'Microfono': 80, 'Impresora': 250, 'Router': 100, 'Disco Duro': 150,
    'USB': 15, 'Auriculares': 60, 'Altavoces': 120
}
```

```
}
```

Generamos una lista de regiones en donde se va a realizar las ventas

```
[6]: regiones_lista = ['Norte', 'Sur', 'Este', 'Oeste', 'Centro', 'Noroeste', '↵  
    ↪ Sureste']
```

Generamos la estructura del Dataframe

```
[8]: data = {  
    'ID_Venta': [],  
    'Producto': [],  
    'Cantidad': [],  
    'Precio': [],  
    'Fecha_Venta': [],  
    'Region': []  
}
```

Creamos los datos par el dataset

```
[9]: fake = Faker()
```

```
[10]: for i in range(num_registros):  
    producto = random.choice(productos_lista)  
    precio = precios_lista[producto]  
    # Asegurarse de que el precio no sea negativo  
    precio = max(1.0, precio)  
  
    data['ID_Venta'].append(i + 1)  
    data['Producto'].append(producto)  
    data['Cantidad'].append(random.randint(1, 10)) # Cantidad entre 1 y 10  
    data['Precio'].append(precio)  
    data['Fecha_Venta'].append(fake.date_between(start_date='-365d', ↵  
    ↪ end_date='today').strftime('%Y-%m-%d'))  
    data['Region'].append(random.choice(regiones_lista))  
  
df = pd.DataFrame(data)  
df.to_csv('datos_ventas.csv', index=False)
```

Leemos el dataset creado del archivo .csv

```
[11]: #import pandas as pd  
df_ventas = pd.read_csv('datos_ventas.csv', parse_dates=['Fecha_Venta'])
```

Lectura de las cabeceras del dataframe

```
[12]: print(df_ventas.head())
```

| | ID_Venta | Producto | Cantidad | Precio | Fecha_Venta | Region |
|---|----------|----------|----------|--------|-------------|---------|
| 0 | 1 | Teclado | 8 | 75 | 2024-07-22 | Sureste |

| | | | | | | |
|---|---|-------------|---|-----|------------|---------|
| 1 | 2 | Router | 4 | 100 | 2024-07-10 | Norte |
| 2 | 3 | Microfono | 6 | 80 | 2025-01-17 | Sureste |
| 3 | 4 | Impresora | 4 | 250 | 2025-01-17 | Este |
| 4 | 5 | Auriculares | 9 | 60 | 2024-09-15 | Sureste |

Verificación de los tipos de datos

```
[13]: print(df_ventas.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 6 columns):
#   Column          Non-Null Count  Dtype
---  -
0   ID_Venta        1000 non-null   int64
1   Producto        1000 non-null   object
2   Cantidad        1000 non-null   int64
3   Precio         1000 non-null   int64
4   Fecha_Venta     1000 non-null   datetime64[ns]
5   Region         1000 non-null   object
dtypes: datetime64[ns](1), int64(3), object(2)
memory usage: 47.0+ KB
None
```

```
[ ]: Agregamos cuatro columnas al dataframe
```

```
[14]: # Columna con el valor total de la venta realizada
df_ventas['Total_Venta'] = df_ventas['Cantidad'] * df_ventas['Precio']
# Columna adicional solo con el año
df_ventas['Año_Venta'] = df_ventas['Fecha_Venta'].dt.year
# Columna adicional solo con el mes
df_ventas['Mes_Venta'] = df_ventas['Fecha_Venta'].dt.month
# Día de la semana de venta
df_ventas['Dia_Semana_Venta'] = df_ventas['Fecha_Venta'].dt.day_name()
#Verificamos si la venta se realizó el fin de semana
df_ventas['Es_Fin_Semana'] = df_ventas['Fecha_Venta'].dt.day_name().
    .isin(['Saturday', 'Sunday'])

print("\nDataFrame con nuevas columnas:")
print(df_ventas.head())
```

DataFrame con nuevas columnas:

| | ID_Venta | Producto | Cantidad | Precio | Fecha_Venta | Region | Total_Venta | \ |
|---|----------|-------------|----------|--------|-------------|---------|-------------|---|
| 0 | 1 | Teclado | 8 | 75 | 2024-07-22 | Sureste | 600 | |
| 1 | 2 | Router | 4 | 100 | 2024-07-10 | Norte | 400 | |
| 2 | 3 | Microfono | 6 | 80 | 2025-01-17 | Sureste | 480 | |
| 3 | 4 | Impresora | 4 | 250 | 2025-01-17 | Este | 1000 | |
| 4 | 5 | Auriculares | 9 | 60 | 2024-09-15 | Sureste | 540 | |

| | Año_Venta | Mes_Venta | Dia_Semana_Venta | Es_Fin_Semana |
|---|-----------|-----------|------------------|---------------|
| 0 | 2024 | 7 | Monday | False |
| 1 | 2024 | 7 | Wednesday | False |
| 2 | 2025 | 1 | Friday | False |
| 3 | 2025 | 1 | Friday | False |
| 4 | 2024 | 9 | Sunday | True |

[]:

[15]: `df_ventas.to_csv('datos_ventas4.csv', index=False)`

[]: VISUALIZACIÓN

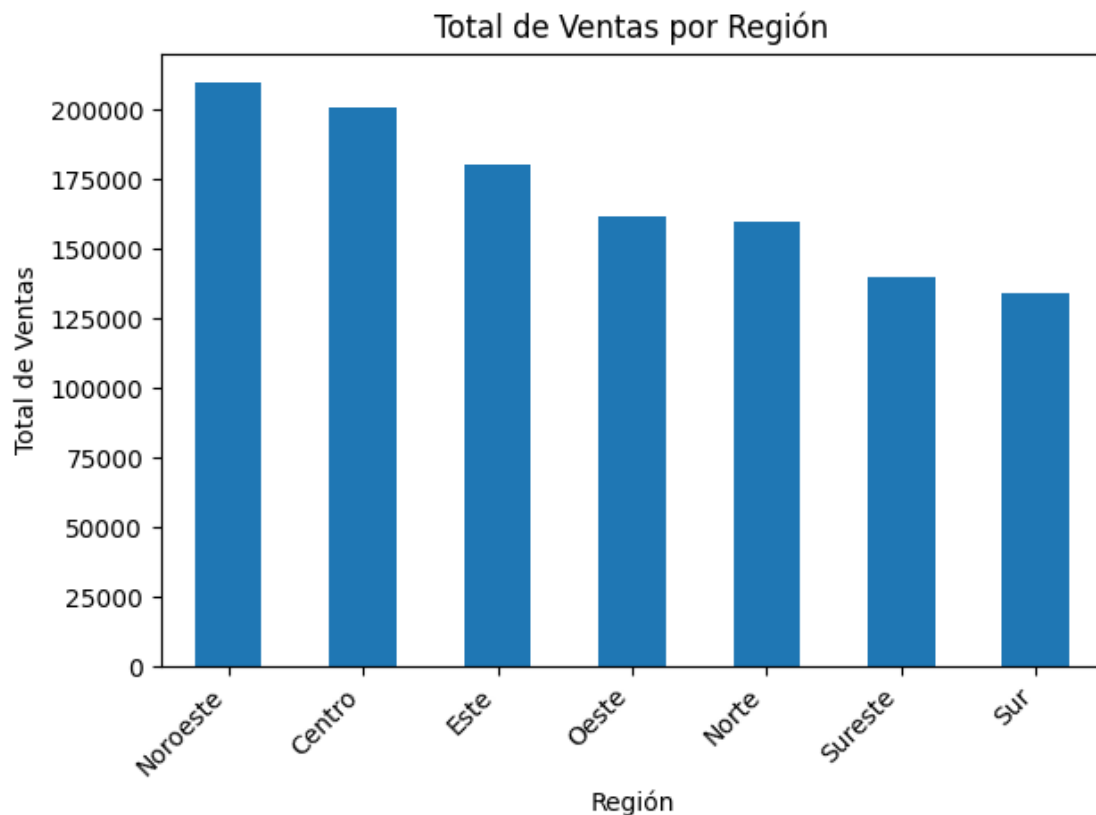
[16]: `!pip install matplotlib seaborn`

Requirement already satisfied: matplotlib in c:\utpl\hia\taller1\lib\site-packages (3.10.1)
Requirement already satisfied: seaborn in c:\utpl\hia\taller1\lib\site-packages (0.13.2)
Requirement already satisfied: contourpy>=1.0.1 in c:\utpl\hia\taller1\lib\site-packages (from matplotlib) (1.3.2)
Requirement already satisfied: cycler>=0.10 in c:\utpl\hia\taller1\lib\site-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in c:\utpl\hia\taller1\lib\site-packages (from matplotlib) (4.57.0)
Requirement already satisfied: kiwisolver>=1.3.1 in c:\utpl\hia\taller1\lib\site-packages (from matplotlib) (1.4.8)
Requirement already satisfied: numpy>=1.23 in c:\utpl\hia\taller1\lib\site-packages (from matplotlib) (2.2.5)
Requirement already satisfied: packaging>=20.0 in c:\utpl\hia\taller1\lib\site-packages (from matplotlib) (25.0)
Requirement already satisfied: pillow>=8 in c:\utpl\hia\taller1\lib\site-packages (from matplotlib) (11.2.1)
Requirement already satisfied: pyparsing>=2.3.1 in c:\utpl\hia\taller1\lib\site-packages (from matplotlib) (3.2.3)
Requirement already satisfied: python-dateutil>=2.7 in c:\utpl\hia\taller1\lib\site-packages (from matplotlib) (2.9.0.post0)
Requirement already satisfied: pandas>=1.2 in c:\utpl\hia\taller1\lib\site-packages (from seaborn) (2.2.3)
Requirement already satisfied: pytz>=2020.1 in c:\utpl\hia\taller1\lib\site-packages (from pandas>=1.2->seaborn) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in c:\utpl\hia\taller1\lib\site-packages (from pandas>=1.2->seaborn) (2025.2)
Requirement already satisfied: six>=1.5 in c:\utpl\hia\taller1\lib\site-packages (from python-dateutil>=2.7->matplotlib) (1.17.0)

```
[17]: import matplotlib.pyplot as plt
import seaborn as sns
```

Visualización 1: Total de Ventas por Región

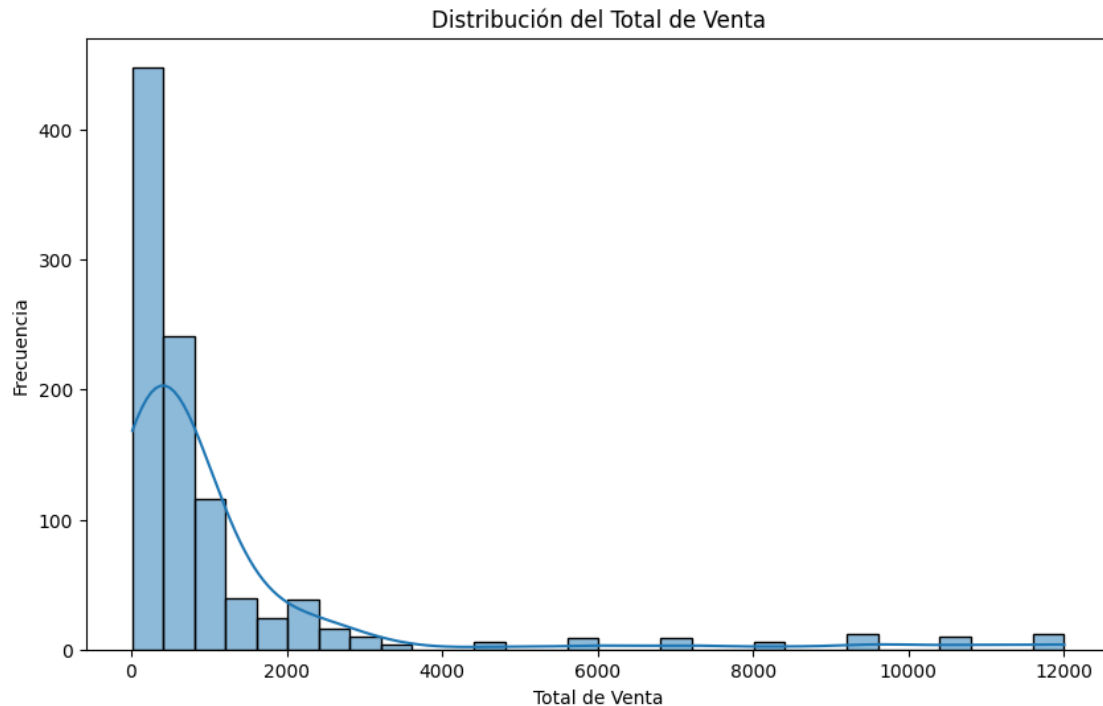
```
[18]: ventas_por_region = df_ventas.groupby('Region')['Total_Venta'].sum().
      ↪sort_values(ascending=False)
ventas_por_region.plot(kind='bar')
plt.title('Total de Ventas por Región')
plt.xlabel('Región')
plt.ylabel('Total de Ventas')
plt.xticks(rotation=45, ha='right')
plt.tight_layout() # Ajustar diseño para que las etiquetas no se corten
plt.show()
```



Visualización 2: Distribución del Total de Venta (Histograma)

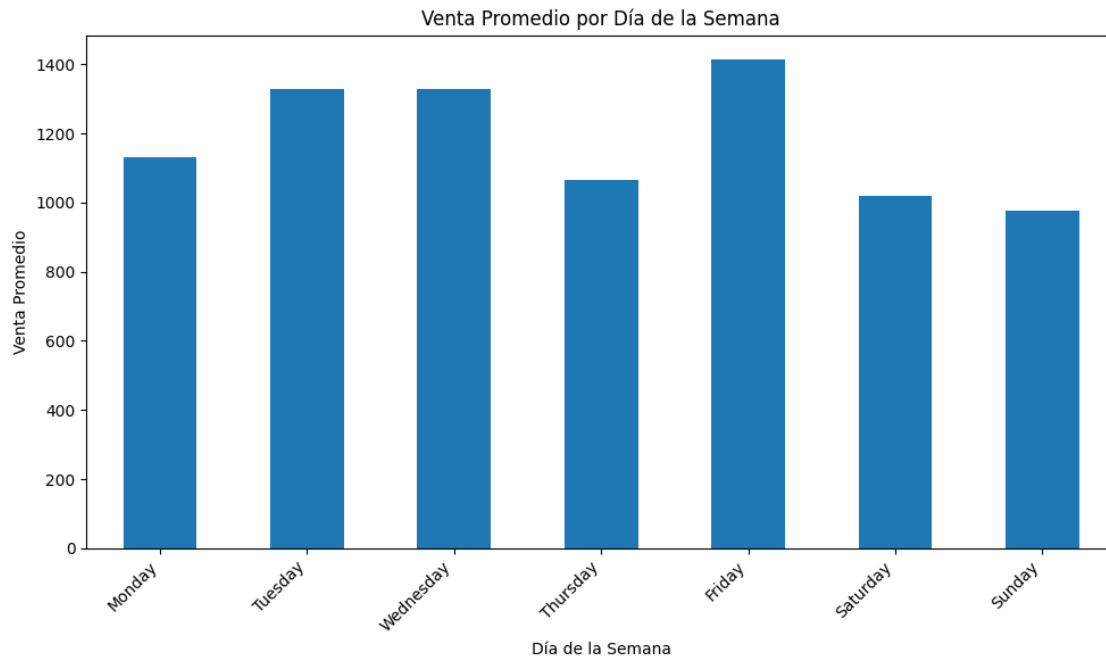
```
[19]: plt.figure(figsize=(10, 6))
sns.histplot(df_ventas['Total_Venta'], kde=True, bins=30) # kde=True añade una
      ↪curva de densidad
plt.title('Distribución del Total de Venta')
```

```
plt.xlabel('Total de Venta')
plt.ylabel('Frecuencia')
plt.show()
```



Visualización 3 Ventas promedio por Día de la Semana

```
[22]: plt.figure(figsize=(10, 6))
ventas_por_dia_semana = df_ventas.groupby('Dia_Semana_Venta')['Total_Venta'].
    .mean().reindex([
        'Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday',
        'Sunday']) # Ordenar días
ventas_por_dia_semana.plot(kind='bar')
plt.title('Venta Promedio por Día de la Semana')
plt.xlabel('Día de la Semana')
plt.ylabel('Venta Promedio')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```

```
[23]: !pip install nbconvert
```

```
Requirement already satisfied: nbconvert in c:\utpl\hia\taller1\lib\site-
packages (7.16.6)
Requirement already satisfied: beautifulsoup4 in c:\utpl\hia\taller1\lib\site-
packages (from nbconvert) (4.13.4)
Requirement already satisfied: bleach!=5.0.0 in c:\utpl\hia\taller1\lib\site-
packages (from bleach[css]!=5.0.0->nbconvert) (6.2.0)
Requirement already satisfied: defusedxml in c:\utpl\hia\taller1\lib\site-
packages (from nbconvert) (0.7.1)
Requirement already satisfied: Jinja2>=3.0 in c:\utpl\hia\taller1\lib\site-
packages (from nbconvert) (3.1.6)
Requirement already satisfied: jupyter-core>=4.7 in
c:\utpl\hia\taller1\lib\site-packages (from nbconvert) (5.7.2)
Requirement already satisfied: jupyterlab-pygments in
c:\utpl\hia\taller1\lib\site-packages (from nbconvert) (0.3.0)
Requirement already satisfied: MarkupSafe>=2.0 in c:\utpl\hia\taller1\lib\site-
packages (from nbconvert) (3.0.2)
Requirement already satisfied: mistune<4,>=2.0.3 in
c:\utpl\hia\taller1\lib\site-packages (from nbconvert) (3.1.3)
Requirement already satisfied: nbclient>=0.5.0 in c:\utpl\hia\taller1\lib\site-
packages (from nbconvert) (0.10.2)
Requirement already satisfied: nbformat>=5.7 in c:\utpl\hia\taller1\lib\site-
packages (from nbconvert) (5.10.4)
Requirement already satisfied: packaging in c:\utpl\hia\taller1\lib\site-
packages (from nbconvert) (25.0)
```

Requirement already satisfied: pandocfilters>=1.4.1 in c:\utpl\hia\taller1\lib\site-packages (from nbconvert) (1.5.1)

Requirement already satisfied: pygments>=2.4.1 in c:\utpl\hia\taller1\lib\site-packages (from nbconvert) (2.19.1)

Requirement already satisfied: traitlets>=5.1 in c:\utpl\hia\taller1\lib\site-packages (from nbconvert) (5.14.3)

Requirement already satisfied: webencodings in c:\utpl\hia\taller1\lib\site-packages (from bleach!=5.0.0->bleach[css]!=5.0.0->nbconvert) (0.5.1)

Requirement already satisfied: tinycss2<1.5,>=1.1.0 in c:\utpl\hia\taller1\lib\site-packages (from bleach[css]!=5.0.0->nbconvert) (1.4.0)

Requirement already satisfied: platformdirs>=2.5 in c:\utpl\hia\taller1\lib\site-packages (from jupyter-core>=4.7->nbconvert) (4.3.7)

Requirement already satisfied: pywin32>=300 in c:\utpl\hia\taller1\lib\site-packages (from jupyter-core>=4.7->nbconvert) (310)

Requirement already satisfied: jupyter-client>=6.1.12 in c:\utpl\hia\taller1\lib\site-packages (from nbclient>=0.5.0->nbconvert) (8.6.3)

Requirement already satisfied: python-dateutil>=2.8.2 in c:\utpl\hia\taller1\lib\site-packages (from jupyter-client>=6.1.12->nbclient>=0.5.0->nbconvert) (2.9.0.post0)

Requirement already satisfied: pyzmq>=23.0 in c:\utpl\hia\taller1\lib\site-packages (from jupyter-client>=6.1.12->nbclient>=0.5.0->nbconvert) (26.4.0)

Requirement already satisfied: tornado>=6.2 in c:\utpl\hia\taller1\lib\site-packages (from jupyter-client>=6.1.12->nbclient>=0.5.0->nbconvert) (6.4.2)

Requirement already satisfied: fastjsonschema>=2.15 in c:\utpl\hia\taller1\lib\site-packages (from nbformat>=5.7->nbconvert) (2.21.1)

Requirement already satisfied: jsonschema>=2.6 in c:\utpl\hia\taller1\lib\site-packages (from nbformat>=5.7->nbconvert) (4.23.0)

Requirement already satisfied: attrs>=22.2.0 in c:\utpl\hia\taller1\lib\site-packages (from jsonschema>=2.6->nbformat>=5.7->nbconvert) (25.3.0)

Requirement already satisfied: jsonschema-specifications>=2023.03.6 in c:\utpl\hia\taller1\lib\site-packages (from jsonschema>=2.6->nbformat>=5.7->nbconvert) (2025.4.1)

Requirement already satisfied: referencing>=0.28.4 in c:\utpl\hia\taller1\lib\site-packages (from jsonschema>=2.6->nbformat>=5.7->nbconvert) (0.36.2)

Requirement already satisfied: rpds-py>=0.7.1 in c:\utpl\hia\taller1\lib\site-packages (from jsonschema>=2.6->nbformat>=5.7->nbconvert) (0.24.0)

Requirement already satisfied: six>=1.5 in c:\utpl\hia\taller1\lib\site-packages (from python-dateutil>=2.8.2->jupyter-client>=6.1.12->nbclient>=0.5.0->nbconvert) (1.17.0)

Requirement already satisfied: typing-extensions>=4.4.0 in c:\utpl\hia\taller1\lib\site-packages (from referencing>=0.28.4->jsonschema>=2.6->nbformat>=5.7->nbconvert) (4.13.2)

Requirement already satisfied: soupsieve>1.2 in c:\utpl\hia\taller1\lib\site-packages (from beautifulsoup4->nbconvert) (2.7)

[]: