

Para algunos de los ejercicios, se uso Google Collaboratory como plataforma.

Punto #1: Carga de Información:

Para empezar, importe la librería de pandas, para así manejar los datos de una mejor manera:

```
[151] import pandas as pd
      import requests
      from io import StringIO
```

Luego, desde una carpeta alojada en un repositorio público de Github, accedo el archivo para su manipulación:

```
[139] # URL del archivo en el repositorio de GitHub
      url = 'https://raw.githubusercontent.com/Merxxotas/Prueba-DataKnow/main/Prueba_Tecnica/Datos3/OFEI1204.txt'

      # Descargar el archivo
      response = requests.get(url)
      response.raise_for_status()

      # Guardar el contenido del archivo en una variable
      file_content = response.text
```

Visualizo los datos como están inicialmente:

```
[141] data_frame = pd.read_table(StringIO(file_content), encoding='utf-8', header=None, skip_blank_lines=True)
      old_data_frame = data_frame

[142] # Visualización de los datos
      print(data_frame)

0      0      Ofertas Iniciales para: 2017-12-04
1      1      AGENTE: AES CHIVOR
2      2      CHIVOR1 , D, 125, 125, 125, 125, 125, 12...
3      3      CHIVOR2 , D, 125, 125, 125, 125, 125, 12...
4      4      CHIVOR3 , D, 125, 125, 125, 125, 125, 12...
..      ...
513     FLORES1 , CONF, 1
514     FLORES1 , C, GAS
515 FLORES4B , D, 450, 450, 450, 450, 450, 4...
516 FLORES4B , CONF, 1
517 FLORES4B , C, GAS

[518 rows x 1 columns]

[143] # Detalle de los datos
      print(data_frame.describe(include='all'))

count      518
unique      518
top      Ofertas Iniciales para: 2017-12-04
freq      1
```

empiezo a realizar el manejo de los datos, separando las líneas relevantes con las columnas que contengan la palabra “AGENTE” y sean columnas tipo D, además de crear un DataFrame con las filas filtradas.

```
✓ [144] # Separar los datos de las líneas relevantes
0s
    filtered_rows = []
    current_agent = None

    for _, row in data_frame.iterrows():
        line = row[0].strip() # Trabajar con la columna única cargada como texto
        if line.startswith("AGENTE: "):
            current_agent = line.replace("AGENTE: ", "").strip()
        elif ', D, ' in line:
            parts = line.split(', ')
            planta = parts[0].strip()
            tipo = parts[1].strip()
            horas = [float(h.strip()) for h in parts[2:]]
            filtered_rows.append([current_agent, planta] + horas)

✓ [145] # Crear un DataFrame con las filas filtradas
0s
    columns = ['Agente', 'Planta'] + [f'Hora_{i}' for i in range(1, 25)]
    df = pd.DataFrame(filtered_rows, columns=columns)
```

Visualizo los nuevos resultados:

```
[ ] # Visualizar resultados
print(df.head())
```

	Agente	Planta	Hora_1	Hora_2	Hora_3	Hora_4	Hora_5	Hora_6	\
0	AES CHIVOR	CHIVOR1	125.0	125.0	125.0	125.0	125.0	125.0	
1	AES CHIVOR	CHIVOR2	125.0	125.0	125.0	125.0	125.0	125.0	
2	AES CHIVOR	CHIVOR3	125.0	125.0	125.0	125.0	125.0	125.0	
3	AES CHIVOR	CHIVOR4	125.0	125.0	125.0	125.0	125.0	125.0	
4	AES CHIVOR	CHIVOR5	125.0	125.0	125.0	125.0	125.0	125.0	

	Hora_7	Hora_8	...	Hora_15	Hora_16	Hora_17	Hora_18	Hora_19	Hora_20	\
0	125.0	125.0	...	125.0	125.0	125.0	125.0	125.0	125.0	
1	125.0	125.0	...	125.0	125.0	125.0	125.0	125.0	125.0	
2	125.0	125.0	...	125.0	125.0	125.0	125.0	125.0	125.0	
3	125.0	125.0	...	125.0	125.0	125.0	125.0	125.0	125.0	
4	125.0	125.0	...	125.0	125.0	125.0	125.0	125.0	125.0	

	Hora_21	Hora_22	Hora_23	Hora_24
0	125.0	125.0	125.0	125.0
1	125.0	125.0	125.0	125.0
2	125.0	125.0	125.0	125.0
3	125.0	125.0	125.0	125.0
4	125.0	125.0	125.0	125.0

[5 rows x 26 columns]

A los nuevos resultados, los guardo en una nueva tabla llamada: “Tabla resultante” en formato .CSV, y lo visualizo de dos maneras para su comprobación (el archivo .CSV se encuentra de igual manera en el repositorio, junto con el código utilizado)

