**Desafío 3:** Data Engineer Jr.

**Nos interesa que puedas hacer lo siguiente:**

* Ingreses al link de la carpeta: [inputEjemplos](https://drive.google.com/drive/folders/1RtpTNQUAoQlJNO-Fr4vtyNuxL_lpDKhq)
* Deserializá el alguno de los archivos. Nuestro preferido es: [MPE1004\_0000000000.json.gz](https://drive.google.com/file/d/1QDPpo_JA9GXWysLNnut5GVTwvQnc36GB/view?usp=sharing)
* Lee la información en un DataFrame.
* Guardá la información deserializada en un archivo CSV con el siguiente patrón:

Sitio/Mes/Año/dia/(archivo).csv. Por Ejemplo MPE/2020/08/28/[NoNosImporta].csv

* **Bonus:** Usar Spark Scala + IntelliJ - También puede ser Pyspark + Pycharm

# Solución Propuesta al desafío

La solución se compone de un script en lenguaje python, utilicé Spyder como IDE y las librerías: Pandas, Json, Date y Path.

1. El script comienza importando librerías y asignando valor a los parametros de input file y fecha proceso.
2. Abro el json y lo leo en una variable llamada data.
3. Utilizo la funcion json\_normalize para aplanar el json. Link a la documentacion: <https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.json_normalize.html>
4. Una vez generado el dataframe normalizado, realizo una pequeña validación de cantidad de registros y mostrando un mensaje de datos ok o datos faltantes.
5. Genero dinamicamente la ruta donde se guarda el archivo usando las variables fecha\_proceso y site\_id.
6. Escribo el dataframe en un csv.