P.12 - Borme + PDFtotext

El objetivo de la práctica es ver cómo extraer y estructurar un texto plano extraído de un PDF.

Instalación

Para realizar la práctica necesitaremos instalar la librería `pdftotext` que vamos a usar para extraer texto del PDF.

```
pip3 install pdftotext
```

Estructura el proyecto

- Copia el archivo de spider que hemos visto en la práctica anterior y pégalo en una nueva carpeta para esta práctica.

- Crea el código necesario para estructurar el proyecto con las estructuras de carpetas que vimos en clase. Algo así. (Puede ser que venv lo tengas en una carpeta superior, la de la asignatura, no pasa nada si no aparece.)

  ```
  P12_Borme
  |__ borme
  | |__ spider.py
  | |__ crawler.py
  ```

```
| |__ __main__.py
|__ data
| |__ inputs
| |__ outputs
| |__ logs
|__ venv
|__ requirements.txt
|__ README.md
```

- Adapta todo para que al invocar el programa entre por __main__, como vimos en clase, y pueda ejecutar:
  - La fase spider recibiendo un día en formato YYYYMMDD.
  - Si hiciste los extras de la práctica anterior la fase spider recibiendo la ruta de un archivo .txt con fechas (estará en data/inputs).
  - La fase crawler (la vamos a hacer en esta práctica) recibiendo la ruta de uno de los archivos PDF que descargamos en la práctica anterior.
  - La fase crawler recibiendo la ruta a la carpeta donde están todos los archivos para que los lea uno a uno secuencialmente.

- Si no has montada nada de esto nunca no intentes hacer estas cosas de primeras, ve poco a poco. Primero vincula el spider y cuando funcione bien te pones con el crawler.

- Puedes usar la librería `argparse` que está en el core de Python, échale un ojo.

  Se utiliza más o menos así.

```
import argparse

if __name__ == '__main__':
  '''
  '''
  argument_parser = argparse.ArgumentParser(
      description='Download and parse data from Borme.'
  )
  argument_parser.add_argument(
```

```
        '-p',
        '--phase',
        required=True,
        help='Execution phase. Can be spider or crawler'
    )
    argument_parser.add_argument(
        '-d',
        '--day',
        help='Target date. Format YYYYMMDD.'
    )
    argument_parser.add_argument(
        '-fp',
        '--filepath',
        help='Target file. Must be a PDF.'
    )
    argument_parser.add_argument(
        '-dp',
        '--directorypath',
        help='Target directory. Must be a directory full of PDFs.'
    )

    # Parse execution arguments.
    args = argument_parser.parse_args()

...
```

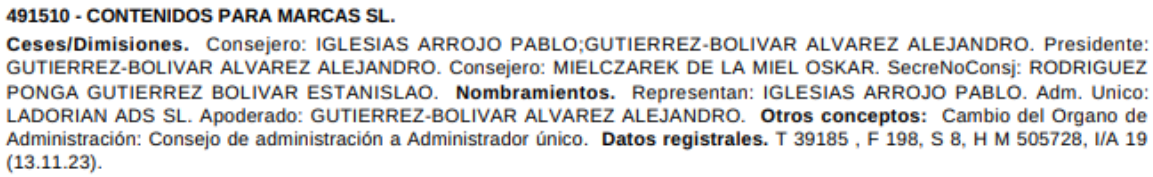
Estructurar texto plano

- Ahora que tenemos la estructura vamos a montar la fase crawler. Desarrolla el código para abrir uno de los archivos PDF que descargaste en la práctica anterior. Si no la acabaste lo puedes descargar a mano en la siguiente url para seguir la clase.

> BOE.es - Sumario del día 27/11/2023
>
> Sumario del día 27/11/2023
>
> https://www.boe.es/borme/dias/2023/11/27/index.php

- Primero estudia el documento. Nota que cada párrafo del PDF habla de una empresa en concreto y dentro del párrafo aparecen diferentes actos legales. Al acabar el párrafo aparecen siempre los ‘Datos registrales’.

  Por ejemplo CONTENIDOS PARA MARCAS SL. ha hecho tres actos legales a la vez: ‘Ceses/Dimisiones’, ‘Nombramientos’ y ’Otros conceptos’.



491510 - CONTENIDOS PARA MARCAS SL.

Ceses/Dimisiones. Consejero: IGLESIAS ARROJO PABLO;GUTIERREZ-BOLIVAR ALVAREZ ALEJANDRO. Presidente: GUTIERREZ-BOLIVAR ALVAREZ ALEJANDRO. Consejero: MIELCZAREK DE LA MIEL OSKAR. SecreNoConsj: RODRIGUEZ PONGA GUTIERREZ BOLIVAR ESTANISLAO. **Nombramientos.** Representan: IGLESIAS ARROJO PABLO. Adm. Unico: LADORIAN ADS SL. Apoderado: GUTIERREZ-BOLIVAR ALVAREZ ALEJANDRO. **Otros conceptos:** Cambio del Organo de Administración: Consejo de administración a Administrador único. **Datos registrales.** T 39185 , F 198, S 8, H M 505728, I/A 19 (13.11.23).

- Lee el PDF línea a línea usando `pdftotext` como hemos visto en la teoría.

- Estructura el contenido separando cada párrafo. Ten cuidado con la información en la cabecera del PDF y los textos en los márgenes.

> 💡 **IGUAL TE ES ÚTIL...**
> Para extraer el contenido igual te es útil utilizar la funcionalidad de Python word.isdigit() que devolverá True si la string "word" es un número y False si no lo es.
> Por ejemplo: "498053".isdigit() >> True, "7up".isdigit() >> False.

- Una vez tengamos la información bien separada en párrafos nos vamos a centrar en las empresas que se han constituido y han cerrado en el día que estamos estudiando. Los actos legales de ‘Constitución’ y ‘Extinción’. Saca y estructura toda la información que puedas rascar.

- Mete todos los diccionarios en una lista de Python y guárdalo en un JSON o un JSONL.

Tiene que quedar algo así:

```
[
  {
    'Nombre': 'AMARILLA BUSINESS SL',
    'Id': '492102',
    'Acto legal': 'Constitución',
    'Comienzo de operaciones': '13.11.23',
    'Objeto social': 'Intermediarios del comercio de productos diversos',
    'Domicilio': 'C/ MAGDALENA DIEZ 3 - PUERTA A, PLANTA 1 (MADRID)',
    'Capital': '3.000,00'
   },{
     'Nombre': 'ILUMINAMAS SL',
     'Id': '491525',
     'Acto legal': 'Extinción',
     'Disolución': 'Voluntaria'
    }
    ...
```

- Una vez veas que te funciona con un PDF concreto desarrolla el código necesario para que lea de una carpeta y vaya abriendo los PDFs secuencialmente.