Pdf Scraping

Video Link : https://www.youtube.com/watch?v=3Xw9YGh00aM
Notebook:
https://colab.research.google.com/drive/1mNhUTij7LdsjxgcfOKgfsmbFOI526y2t

---

Libraries used:

- requests: Scraping with Python Requests will allow us to **send HTTP/1.1 requests using Python**. With it, we can add content like headers, form data, multipart files, and parameters via simple Python libraries. https://docs.python-requests.org/en/latest/
- urllib.request: module defines functions and classes which help in opening URLs (mostly HTTP) in a complex world — basic and digest authentication, redirections, cookies and more.
- urllib.parse: This module defines a standard interface to break Uniform Resource Locator (URL) strings up in components (addressing scheme, network location, path etc.), to combine the components back into a URL string, and to convert a “relative URL” to an absolute URL given a “base URL.”
- bs4: Scraping with Python Beautiful Soup is a Python library that is used for **web scraping purposes to pull the data out of HTML and XML files**. It creates a parse tree from page source code that can be used to extract data in a hierarchical and more readable manner. https://beautiful-soup-4.readthedocs.io/en/latest/#
- Tabula: Tabula allows you to extract that data into a CSV or Microsoft Excel spreadsheet using a simple, easy-to-use interface. Tabula works on Mac, Windows and Linux. Tabula can read pdf files like pandas reads csv files.

> To read pdf file using Tabula:
>
> ```
> tabula.read_pdf(pdf_file_name,  pages='page_number')
> ```

> To convert pdf into csv file:

> `from tabula import convert_into`
>
> `tabula.io.convert_into(_input_path_, _output_path_, _output_format='csv'_, _java_options=None_, _**kwargs_)`
>
> Output file will be saved into output_path