AI4Bharat Data Engineer Internship Task

Problem Statement:

There are two problems associated with the task, one related to extracting content from wikipedia pages and the other refers to extracting text from pdf files. You are expected to complete both the tasks within the stipulated time limit. More details below:

1. Write a script (named *“wiki_extractor.py”*) that performs a Wikipedia search using the provided keyword, and returns urls of “n” **related** Wikipedia pages. Additionally, the script should also extract one paragraph from each such page and return the result as a json file (Details given below).

Input arguments: --keyword (string argument to define the query string)
--num_urls (integer argument for number of wikipedia pages to extract from)
--output (output json-file name)

Response: <output> (json file containing response as a list of dictionaries, where each dictionary has two keys; “url”, “paragraph” as shown below!)

```
[
   {
       “url”: <url link to the page1>,
       “paragraph”: <text content of one paragraph>
   },
   {
       “url”: <url link to the page2>,
       “paragraph”: <text content of one paragraph>
   },
   {
       “url”: <url link to the page3>,
       “paragraph”: <text content of one paragraph>
   },
         ……………………….
         ……………………….
         ……………………….
         ……………………….
]
```

Example:

```
python wiki_extractor.py --keyword=”Indian Historical Events” --num_urls=100
--output=”out.json”
```

2. Extract pdf content from all the url’s present in the following google spreadsheet:

Data Engineer Task

There are two kind of url’s in the following sheet:

A. Urls which are ending with “.pdf” which directly trigger downloading of the pdf file.

B. Urls with the link to the page where “pdf” files are located. In order to extract from files like these you would need to scrape the html content and look for pdf links in it!

Response: pdf_extract.json (json file containing response as a list of dictionaries, where each dictionary has three keys; “page-url”, “pdf-url”, “pdf-content” as shown below! Note: page-url and pdf-url will be the same in case A, but different in case B.)

```
[
   {
       “page-url”: <url link to the page1>,
       “pdf-url”: <url link to the pdf1>,
       “paragraph”: <text content of one paragraph>
   },
   {
       “url”: <url link to the page2>,
       “paragraph”: <text content of one paragraph>
   },
   {
       “url”: <url link to the page3>,
       “paragraph”: <text content of one paragraph>
   },
         ……………………….
         ……………………….
         ……………………….
         ……………………….
]
```

FAQ:

1. **How do I submit my work?**
   Push all your code to Github with instructions on how to use it in the README.md file, and send us the repo link. Make the README as detailed as possible - using it as a way to document your design process.

2. **What is the deadline for the task?**
   You will be given a maximum of 48 hours duration to complete the task. If you have any challenges with finding time or want to extend the deadline, please let us know.

3. **Terms and Conditions?**
   - You agree to not share this confidential document with anyone.
   - You agree to open-source your code under any license you prefer.
   - Do not mention our company's name anywhere in the code or repo.
   - We will never use your source code under any circumstances for any commercial purposes; this is just a basic assessment task.

All the best!