Data Migration - Round 2

Team TheThree - Task 2

a) HTML to RST parser

- We implemented a class `MyHTMLParser` by inheriting from the `html.parser.HTMLParser` and overridden its handler methods to carry out the desired behavior when start tags, end tags, and data are encountered.

- List of tags that our parser can handle:

    - <body></body>
    - <div></div>
    -
    - <p></p>
    - , , <code></code>
    -
, <a>
    - , <ol>,

- Example:

```
text = '<ol><li>Coffee<ul><li>Arabica</li><li>Robusta</li></ul></li>
<li>Tea</li><li>Milk</li></ol>'
parser = MyHTMLParser()
parser.feed(text)
parser.get_rst()

### Output:
#
#
# #. Coffee
#
#      * Arabica
#      * Robusta
#
# #. Tea
# #. Milk
#
#
```

- Reference: python-html2rest

b) JSON to RST parser

- We implemented a class `RstBuilder` which have methods to write components to file

- `RstBuilder` have some method

- heading(str): to add heading
- newline(): to add new line
- subheading(str): to add subheading
- directive(name, attributes): to add directive with name and its attributes
- directives(directives): to add directives to rst

c) Usage

The program have 2 arguments. You can check the documentation by using the command

```
python main.py -h
```

```
usage: main.py [-h] [-i INPUT_FILE] [-o OUTPUT_FILE]

options:
  -h, --help            show this help message and exit
  -i INPUT_FILE, --input_file INPUT_FILE
                        Directory to input file. Accepts file *.json only
  -o OUTPUT_FILE, --output_file OUTPUT_FILE
                        Directory to output *.rst file.
  -s SETTINGS, --settings SETTINGS
                        Directory to configure settings *.yml file
```

Example command:

```
python main.py -i Requirements.json -o Requirements.rst -s config.yml
```