Description of the scraper:
Web Scraper scrapes data concerning films that are going to have their premiers in a span of the current year using the **IMDB.** It also includes the first recommended movie (from section *more like this*) for each film from *coming soon* list.

How it works:
Scraper in Scrapy consists of 3 files (path to files: `imdb/imdb/spiders/`) The first one (`imdb_1`) extracts links to all months from *coming soon* page. All links are saved in `.csv` format. The second one (`imdb_2`) opens list of links saved in the first `.csv` file. From each page it gets links to movies, creates a list of them and saves it as another `.csv`. The last one is the most complicated. The third file (`imdb_3`) opens links from previous `.csv` file (links for all *coming soon* movies) and gets all necessary data (title, directors', writers' and actors' names, genres, country of origin, language spoken, release date and duration). Then it gets into the first movie from *more like this* and scraps the same information. After that it is forbidden to go any further, so Scrapy goes back to the next movie from *upcoming* list and repeats the process until it reaches 100 films. However, there may be more than 100 films in the result file (`movies.csv`). It is because Scrapy noticed it reached the limit while still having some pending requests that were being processed (explanation can be found on stackoverflow). In order to avoid duplicates, in `pipelines.py` two separate classes were created – `DuplicatesPipelineLinks` and `DuplicatesPipelineItems`. I refer to them through `custom_settings` variable in `imbd_2` and `imdb_3` files.

Scrapy files are supposed to be open in particular order: `imdb_1`, `imdb_2` and `imdb_3` using commends:

1. `scrapy crawl link_list -o link_list.csv`
2. `scrapy crawl links -o links.csv`
3. `scrapy crawl movies -o movies.csv`

analisys.py file: visualisation of the data contained in the file `movies.csv` using matplotlib. It saves created charts.