Run Scrapy from a script

실행

```
import scrapy
from scrapy.crawler import CrawlerProcess

class MySpider(scrapy.Spider):
    # Your spider definition
    ...

process = CrawlerProcess(settings={
    "FEEDS": {
        "items.json": {"format": "json"},
    },
})

process.crawl(MySpider)
process.start() # the script will block here until the crawling is finished
```

scrapy.cfg 사용

scrapy.cfg

```
[settings]
default: lab.settings
```

settings.py

```
FEEDS = {
    "items.json": {
        "format": "json",
        "encoding": "utf8",
        "store_empty": False,
        "fields": None,
        "indent": 4,
        "item_export_kwargs": {
            "export_empty_fields": True,
        },
    }
}


CONCURRENT_REQUESTS = 30
CONCURRENT_REQUESTS_PER_DOMAIN = 30
```

```
AUTOTHROTTLE_ENABLED = False
# RANDOMIZE_DOWNLOAD_DELAY": False,
# "REACTOR_THREADPOOL_MAXSIZE": 100,
RETRY_TIMES = 10
DOWNLOAD_TIMEOUT = 15
# TWISTED_REACTOR":
"twisted.internet.asyncioreactor.AsyncioSelectorReactor",
ITEM_PIPELINES = {"lab.pipelines.JsonWriterPipeline": 500}
DOWNLOADER_MIDDLEWARES = {
    "scrapy.downloadermiddlewares.retry.RetryMiddleware": None,
"lab.middlewares.custom_downloader_middleware.CustomDownloaderMiddleware":
543,
    "scrapy.downloadermiddlewares.httpproxy.HttpProxyMiddleware": 610,
}
```

middleware

```
import logging
from random import choice

from lab.overseas_fashion.fendi_spider import log_wrap
from util.requests_util import RequestsUtil

logger = logging.getLogger(__name__)


class CustomDownloaderMiddleware:

    @classmethod
    def from_crawler(cls, crawler):
        cls.proxy_list = RequestsUtil.proxy_crawl()
        # s = cls(crawler.settings)
        s = cls()
        crawler.signals.connect(s.spider_error, signal=s.spider_error)
        return s

    def spider_error(self, failure, response, spider):
        print(
            "Error on {0}, traceback: {1}".format(response.url,
failure.getTraceback())
        )

    @log_wrap
    def process_request(self, request, spider):
        proxy = choice(self.proxy_list)
        logger.info(proxy)
        request.meta["proxy"] = f"http://{proxy}"

    def change_proxy(self, request):
```

```
        proxy = choice(self.proxy_list)
        logger.info(proxy)
        request.meta["proxy"] = f"http://{proxy}"
        return request

    @log_wrap
    def process_exception(self, request, exception, spider):
        print(exception)
        return self.change_proxy(request)

    @log_wrap
    def process_response(self, request, response, spider):
        print(response)
        return response
```

출처

- https://docs.scrapy.org/en/latest/topics/practices.html

관련 문서

Plugin Backlinks: 아무 것도 없습니다.