Готовимся в проектной работе

Что из себя представляет проектная работа?

Что по срокам?

Где будем работать?

Что является конечным результатом работы?

Структура проектной работы

class Crawler

class Parser

class (название на выбор), в котором будут обрабатываться ваши данные

class DataBaseHelper

Пойдем по порядку

Установка Anaconda и Jupyter Notebook

Пойдем по порядку

Установка Anaconda и Jupyter Notebook

Установка webdriver

Пойдем по порядку

Установка Anaconda и Jupyter Notebook

Установка webdriver

Выбор сайта, который будет парситься, к следующему занятию

Пойдем по порядку

Установка Anaconda и Jupyter Notebook

Установка webdriver

Выбор сайта, который будет парситься, к следующему занятию

Выбор библиотек для работы с парсером

1-ый вариант requests и bs4

```
from bs4 import BeautifulSoup
import requests

url = 'https://lenta.ru/parts/news/'

page = requests.get(url)
soup = BeautifulSoup(page.text, "html.parser")
all_news = []
filteredNews = []
all_news = soup.findAll('li', class_='parts-page__item’)

for data in all_news:
    filteredNews.append(data.text)

for data in filteredNews:
    print(data)
```

Оценка 1-го варианта

Достоинства:

- Скорость
- Простота использования
- Нет огромного нагромождения символов
- Дальнейшая работа с текстом выйдет проще

Недостатки:

- Не позволит построить большие массивы данных
- Не дает доступ к сайту через капчу
- Если нужно проваливаться на сайт через пагинацию – не работает

2-ый вариант bs4 и Selenium

Из файлика parser

Оценка 2-го варианта

Достоинства:

- Позволяет настроить перемещение по сайту
- Позволяет быстро достать то, что вам нужно
- Симулирует работу пользователя в браузере
- Позволяет обработать большие массивы данных
- Пагинация и капча – не проблема

Недостатки:

- Сложное использование, придется повозиться в документации
- Довольно объемный по коду
- Сложен в установке webdriver’a
- Нужны чуть бОльшие знания HTML