Лабораторная 1 - парсер

[github classroom task](https://classroom.github.com/a/qDJFQXSE)

Порядок сдачи и защиты лабы описан [тут](https://docs.google.com/document/d/1pj0LIO8kuOzqxgxW-dfZkQFhwZ9dkQ1hIxg5Va39x7U/edit?usp=drive_link)

Дедлайн указан в таблице с баллами

# Задание

Выберите сайт и напишите программу для краулинга и парсинга его данных. Результатом выполнения программы должен быть датасет с данными с сайта.

Краулинг - получение и скачивание списка страниц или элементов, содержащих искомые данные.

Парсинг - извлечение данных из неструктурированного текста. В данном случае из html-кода страниц или элементов страницы.

Датасет должен:

* содержать как минимум 500 строк
* содержать как минимум 6 переменных из которых:
  + как минимум 2 категориальные
  + как минимум 2 целочисленные
  + как минимум 2 вещественные

Датасет может:

* содержать переменные других типов данных: картинок, аудио, видео, рядов, json’ов

# Рекомендации

* размер датасета от 3000 до 1’000’000 строк
* данные других типов хранятся в отдельной папке, в датасете хранятся пути к файлам  
  Например если у каждой записи датасета есть картинка, то хранить в таблице стоит путь к ней. Саму картинку хранить файлом в папке /pics/
* количество переменных от 10 до 30
* хранить датасет удобно в формате csv
* выбирайте сайт с умом. На этих данных вы будете обучать алгоритмы, которые реализуете в следующих лабах
* **не стесняйтесь задавать вопросы в чат курса**
* в некоторых сайтах на карточке объекта написано мало данных. Но много данных на странице, куда ссылается карточка. Скраулите эти страницы, распарсите их. ПолУчите намного больше данных

Для реализации используйте python библиотеки

* requests для краулинга
* lxml для парсинга
* multiprocessing для параллелизации скачивания, если скачивание занимает больше 15 минут

Учтите, что у многих сайтов есть rate limit запросов, часто это 2 запроса в секунду. Для ускорения можете использовать [набор прокси](https://stackoverflow.com/questions/36446657/requests-multiple-proxies-python)

# Популярные замечания

* используйте конфиг файл для констант
* используйте класс для выражения сущности и методы класса для извлечения ее параметров. Станет удобнее, красивее и понятнее
* не коммитить .idea
* функции длиннее 20 строк - плохо. Их можно декомпозировать
* датафрейм удобнее создавать с помощью DataFrame.from\_records()

# FAQ

1. Если нужно проматывать сайт для дозагрузки объектов, то это можно делать программно. Откройте консоль браузера через Inspect code, во вкладке Консоль напишите js код для промотки. Найти такой код в интернете легко

2. Какие ошибки отлавливать? Те, с которыми падает ваше приложение, сильно упарываться не нужно. Ловить базовый Exception - плохой подход, стрельнет в ногу когда-нибудь

3. Скачал url страниц, а некоторые уже удалили. Скачивайте страницы, когда получаете url. Сможете распарсить сохраненную копию

4. У объекта много картинок/аудио/etc. В датасете указываете адрес папки с картинками