**Цель**: Получение практических навыков автоматизированного сбора большого массива информации.

**Задание**: Реализовать программу для сбора и хранения объявлений о продаже квартир на сайте farpost.

**Описание**: На сайте farpost существует раздел, посвященный объявлениям о продаже квартир. Этот раздел состоит из 2 частей: таблицы со ссылками на объявления по каждой квартире; отдельные страницы для каждого объявления с подробной информацией по ним. Вам необходимо разработать программу, предназначенную для сбора и хранения информации об этих объявлениях. Программа осуществляет запросы к списку объявлений, получает адреса объявлений и последовательно парсит данные со страниц объявлений, после чего записывает их в базу.

**Требования**:

* Программа должна собирать все доступные объявления
* Необходимо собирать (при наличии):
  + url объявление
  + Адрес объявления
  + Стоимость квартиры
  + Район
  + Вид квартиры
  + Площадь
* Все собранные данные необходимо заносить в базу данных
* Нельзя дублировать объявления, сохраненные в базе.
* Программа должна постоянно выводить пользователю информацию о своем выполнении в процессе работы.
* Парсинг не должен блокироваться сайтом.

**Уточнения:**

* Программа осуществляет навигацию по сайту с помощью selenium.
* Необходимо внимательно относиться к сбору данных, т.к. при слишком частых запросах сайт блокирует доступ. Начните с 15 секундной задержки между запросами.
* В качестве субд используйте mongodb.
* Если в объявлении отсутствуют данные полей - вы оставляете эти поля пустыми в бд.
* Для анализа страниц удобно использовать режим разработчика в браузерах. Для google chrome данный режим включается по нажатию кнопки F12. В этом режиме вы можете анализировать код страницы и просматривать запросы, совершенные вашим браузером к сайту.

**Хинты**:

**Поиск объявлений**:

1. Списки объявлений хранятся в тегах <table>. Удобно сначала найти необходимую таблицу, а затем уже искать элементы объявлений в этой таблице.
2. Хотя на сайте данные загружаются при прокручивании страницы вниз, вы можете получать наборы с данными с помощью запросов типа: <https://www.farpost.ru/vladivostok/realty/sell_flats/?page=1>, где page=1 – номер заращиваемого набора объявлений. Теперь на каждой странице будет только одна таблица с данными, хранящая запрашиваемый набор.
3. Количество наборов данных можно получить из тега <span> с классом class="pageCount". Данный элемент является уникальным для всей страницы.
4. Таблицы с данными имеют классы class="viewdirBulletinTable pageableContent dummy-done", класс viewdirBulletinTable является уникальным для таблиц с данными.
5. Внутри таблиц из пункта 4 содержатся объявления, расположенные в тегах <td> с классами class="bull-item\_\_content js-brief\_content". По одному объявлению на элемент с тегом <td>, множество элементов в таблице.
6. Сслылку на объявление можно получить из атрибута href тега <a> с классами class="bulletinLink bull-item\_\_self-link auto-shy". Данный тег следует искать внутри тегов из пункта 5.
7. Так же в блоках из пункта 5 можно найти стоимость квартир из текста элемента по тегу <span> с классом class="price-block\_\_price".

**Данные в объявлении**:

1. Название объявления можно найти в тексте тега <header> с классом class="viewbull-header"
2. Вся информация по объявлению находится по тегу <div> с id id="fieldsetView".
3. Цену квартиры можно найти в тексте элемента по тегу <span> с классом class="viewbull-summary-price\_\_value". Данный тег следует искать внутри элемента из пункта 2.
4. В элементах под тегом <div> с классом class=”field” находится большая часть полезной информации. Внутри таких элементов находятся два элемента <div> - один с классом label, второй с классом value. Текст первого хранит в себе имя параметра, текст второго – его значение. Первый элемент может отсутствовать. Имеет смысл найти все элементы с таким тегом, а затем просматривать названия параметров и выбирать нужные. Данный тег следует искать внутри элемента из пункта 2. Проверьте имена параметров, отображаемые на странице.
5. Информация по параметру «Район» ищется по пункту 4.
6. Информация по параметру «Адрес» ищется по пункту 4.
7. Информация по параметру «Вид квартиры» ищется по пункту 4.
8. Информация по параметру «Площадь» ищется по пункту 4.

**Источники**:

1. Selenium: https://www.selenium.dev/

**Источники**:

1. <https://www.selenium.dev/>
2. https://www.selenium.dev/documentation/en/getting\_started/
3. Ссылки на веб-драйвер: <https://www.selenium.dev/documentation/en/webdriver/driver_requirements>
4. https://docs.python.org/3/library/re.html
5. <https://tproger.ru/translations/regular-expression-python/>
6. https://habr.com/ru/post/280238/