1. **Сторонние библиотеки и модули**

Для работы программы необходимы дополнительные модули Python:

* «requests» - библиотека для работы с HTTP запросами в Python.
* «bs4 (BeautifulSoup4)» - библиотека предоставляющая множество способов навигации по HTML документу с возможностью поиска по тегам, классам, id.
* «html5lib» - библиотека на чистом Python для анализа HTML. Разработана в соответствии со спецификацией HTML WHATWG, реализованной во всех основных веб-браузерах.
* «pandas» - библиотека для работы с различными форматами данных в частности Excel.
* «openpyxl» - библиотека Python для чтения/записи файлов Excel 2010 xlsx/xlsm
* «json» - стандартная библиотека для работы с данными в формате JSON.

1. **Описание функционала программы**

Программа позволяет найти все карточки товаров на заданном сайте, скопировать данные о каждом товаре и сохранить их в заданном формате, описанном в техническом задании.

Структуру программы можно условно представить в виде трех основных разделов, два из которых универсальные:

* Модуль main (класс main) – основной алгоритм программы, привязанный к конкретному сайту, указанному в техническом задании. Эта часть программы индивидуальна. Ее дальнейшее применение в отношении других интернет-ресурсов невозможно без существенных изменений. В данном классе заложены механизмы поиска страниц для парсинга (карточек товаров) по определенному тегу «product-info». Данный тег позволяет отфильтровать страницы карточек товара, т.к. присутствует только на этих страницах. В пакете «pages» заложены все теги для извлечения данных из карточки товара:
  1. Наименование товара сохраняется в поле «name» из тега «h1»
  2. Артикул товара сохраняется в поле «artikul» из тега «div» с классом «description». При отсутствии артикула у товара в поле артикул сохраняются данные поля «model»
  3. Модель товара сохраняется в поле «model» из тега «div» с классом «description».
  4. Производитель товара сохраняется в поле «manufacturer» из тега «div» с классом «description».
  5. Описание товара сохраняется в поле «description» из тега «div» с классом «tab-content».
  6. Изображение товара сохраняется в виде файла в папку «/image» и в виде ссылки в поле «image»
  7. Стоимость товара сохраняется в поле «price» из тега «div» с классом «price».
  8. Ссылку на изображение товара на сайте получаем из тега «a» с классом «cloud-zoom» и сохраняем в поле «link»
* Пакет «pages» - универсальный пакет, разработанный для парсинга страниц любого типа на основе сторонних модулей. Этот пакет может быть использован для работы с любыми сайтами, кроме описанного в техническом задании. В пакет входит два модуля «data» и «image» с соответствующими классами.

Класс «data» отвечает за загрузку страницы из сети по ссылке, поиск дочерних ссылок внутри страницы, поиск тегов и извлечение информации из них, а также проверку страницы на вхождение определенного тега для фильтрации.

Класс «image» отвечает за скачивание и сохранение изображений из сети по ссылке, а также за сохранение результатов работы программы в виже файлов.

1. **Описание алгоритмов и структур данных, используемых при создании программного продукта.**

Структуру программы можно представить в виде UML-модели, представленной на рисунке 1.

* Основной модуль «main» класс «main» содержит следующие константы и методы:

1. Константа «URL» - ссылка на гавную страницу сайта «https://novotechnic.ru» из технического задания.
2. Константа «DOMEN» - содержит строку с доменом сайта «novotechnic.ru» и предназначена для фильтрации внутренних страниц сайта.

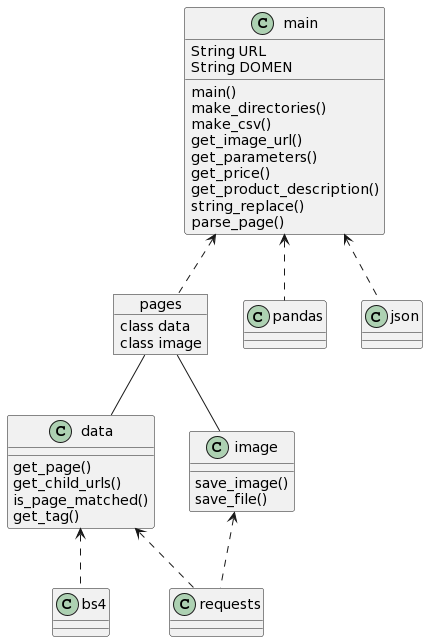


Рисунок 1

1. Метод «main()» содержит основной код программы – основной алгоритм. Запуск программы начинается с вызова этого метода. В начале происходит создание директорий для сохранения файлов. Затем загружается главная страница сайта, сохраненная в «URL». На главной странице происходит поиск всех ссылок, отфильтровываются и исключаются все внешние ссылки по имени домена в константе «DOMEN», ссылки, содержащие GET-параметры и ссылка на главную. В полученном массиве производится поиск страниц с категориями товаров по тегу «div» с классом «category-info» и сохраняется в отдельный список. Все прочие страницы, кроме карточек товаров попадают в список «down» для предотвращения их повторного парсинга в следующих итерациях. Из страниц категорий собираются все дочерние ссылки, кроме внесенных в список «down». Ссылки проверяются на наличие тега «div» с классом «product-info» и сохраняются в отдельный список для карточек товаров. Затем для каждой ссылки из списка карточек вызывается функция «parse\_page», создается массив данных о товарах и в итоге сохраняется в требуемых форматах, указанных в техническом задании.
2. Метод «make\_directories()» проверяет наличие директорий для сохранения результатов работы программы и создает их при необходимости.
3. Метод «parse\_page()» получает на вход html страницу карточки товара и возвращает массив данных с результатом парсинга в формате JSON. Также в данном методе происходит вызов метода сохранения изображения.
4. Метод «make\_csv()» получает на вход массив данных о товарах в формате JSON и возвращает строку в формате текстового файла CSV с разделителем «;» в соответствии с техническим заданием.
5. Метод «get\_parameters()» получает на вход знечение тега «div» с калссом «description», выделяет данные о производителе, артикуле и моделе товара и возвращает в виде списка.
6. Метод «get\_image\_url» получает на вход html страницу и возвращает ссылку на изображение товара.
7. Метод «get\_price» получает на вход html страницу и возвращает значение стоимости товара.
8. Метод «get\_product\_description» получает на вход html страницу и возвращает подробное описание товара в виде «очищенной» строки.
9. Метод «string\_replace» получает на вход строку, «очищает» ее от определенных символов и возвращает «очищенную строку». Из строки удаляются символы табуляции, переноса строки, двойные кавычки (разделитель в JSON) и символ «;» (разделитель CSV).

* Пакет «pages» содержит два модуля с соответствующими классами «data» и «imsge».
* Модуль «data» содержит универсальные методы для парсинга html документа

1. Метод «get\_page()» получает на вход ссылку на страницу в сети в виде строки и возвращает html страницу. Метод использует стороннюю библиотеку «requests» для работы с запросами.
2. Метод «get\_child\_urls()» получает на вход html страницу и возвращает список всех найденных ссылок на этой странице.
3. Метод «is\_page\_matched()» получает на вход html страницу, искомый тег и его класс. Если тег с соответствующим классом найден возвращается «True» иначе «False».
4. Метод «get\_tag» получает на вход html страницу, искомый тег и его класс и возвращает найденный тег или «None», если тег не найден.

* Модуль «image» содержит универсальные методы для скачивания и сохранения изображений, а также для сохранения результатов работы программы на диск (метод для сохранения файлов).

1. Метод «save\_image()» получает на вход ссылку на изображение и имя файла для сохранения скаченного изображения на диск.
2. Метод «save\_file()» получает на вход данные для сохранения, имя файла и формат данных (бинарный или текстовый).

**Приложение №1**

**Техническое задание на разработку парсера сайта novotechnic.ru.**

Необходимо перенести товары из одного интернет-магазина в другой. Перенос контента с одного сайта на другой необходим в связи со сменой системы управления магазином (CMS). У сайтов разные CMS системы, разные структуры баз данных. При этом страницы карточек товаров исходного сайта однотипные.

1. **Сайт-источник парсинга.**

Главная страница: https://novotechnic.ru

1. **Пример страниц с данными.**

Все товары размещены в категориях. Все категории доступны в главном меню сайта, в т.ч. на главной странице.

Пример страницы категории с товарами: <https://novotechnic.ru/watering/watering_complects/>

Пример страницы карточки товара: <https://novotechnic.ru/watering/kit_t36-0.html>

1. **Алгоритм прохода сайта парсером.**

Необходимо найти все карточки товаров на сайте и скопировать данные. Для поиска всех карточек можно использовать все страницы категорий товаров. Товары в категориях могут дублироваться. Дубликаты в результатах парсинга необходимо удалить.

Данные для парсинга в карточке товара:

* Наименование товара (name)
* Артикул товара (artikul)
* Модель товара (model)
* Производитель товара (manufacturer)
* Описание товара (description)
* Изображение товара – необходимо сохранить картинку и ссылку на сохраненный файл (image)
* Стоимость товара (price)
* Ссылку на изображение товара на сайте (link)

1. **Формат сохранения собранных данных**
   1. Файлы с изображениями необходимо сохранить в отдельную папку image/ с сохранением исходных названий файлов и их расширений, например, name.jpg.
   2. Собранные данные необходимо сохранить в трех форматах:

* JSON

Файл в формате JSON

*['[{"name":"name1", artikul":"art1", "model":"model1", "manufacturer":"brend1", "description":"descr1", "image":"images/image1.jpg", "price":"2000.0", "link":"http://novotechnic.ru/image/cache/data/catalog/image1.jpg"}]', [{"name":"name2", artikul":"art2", "model":"model2", "manufacturer":"brend2", "description":"descr2", "image":"images/image2.jpg", "price":"3000.0", "link":"http://novotechnic.ru/image/cache/data/catalog/image2.jpg"}]']*

* Текстовый файл с разделителем «;» CSV

*name;artikul;model;manufacturer;description;image;price;link*

*name1;art1;model1;brend1;descr1;images/image1.jpg;2000.0;http://novotechnic.ru/image/cache/data/catalog/image1.jpg*

*name2;art2;model2;brend2;descr2;images/image2.jpg;3000.0;http://novotechnic.ru/image/cache/data/catalog/image2.jpg*

* Excel

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **name** | **artikul** | **model** | **manufacturer** | **description** | **image** | **price** | **link** |
| name1 | art1 | model1 | brend1 | descr1 | images/image1.jpg | 2000.0 | http://novotechnic.ru/image/cache/data/catalog/image1.jpg |
| name2 | art2 | model2 | brend2 | descr2 | images/image2.jpg | 3000.0 | http://novotechnic.ru/image/cache/data/catalog/image2.jpg |