ЛАБОРАТОРНАЯ РАБОТА №11

Веб-скрапинг с помощью пакета rvest

Цель работы: научиться извлекать данные с сайтов с помощью пакета rvest.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Общие сведения

Веб-скрапинг – процесс извлечения неструктурированных данных (как правило просто помеченных тегами HTML) из веб-страниц для дальнейшего приведения их к структурируемому виду.

Существует четыре принципиальных подхода извлечения данных с веб-ресурсов:

Метод Copy-Paste. Метод основан на ручном копировании необходимой информации с веб-ресурсов и помещение их в структурированный вид (база данных, файлы форматов Excel, CSV и т.д.). Метод позволяет получить достаточно точные результаты, при этом метод слишком долгий для больших объемов данных. Применяется для извлечения отдельных значений (не большого количества значений), когда писать средства автоматизации займет дольше времени, чем извлечь все данные вручную.

Поиск по регулярным выражениям. Метод основан на сопоставлении данных некоторому шаблону и извлечению их с дальнейшим переводом к нужному формату. Особенность метода заключается в необходимости разработать паттерн позволяющий извлечь все (максимум) нужные данные.

Интерфейс API. Метод предполагает, что сайт предоставляет необходимый функционал для извлечения данных. Самый удобный из всех методов, однако возможность реализации этого метода зависит от разработчика веб-сервиса.

Парсинг DOM. Модель DOM представляет иерархическую структуру (Рисунок 11.1). Извлечение данных в этом случае осуществляется с использованием соответствующих тегов, которыми помечены данные.

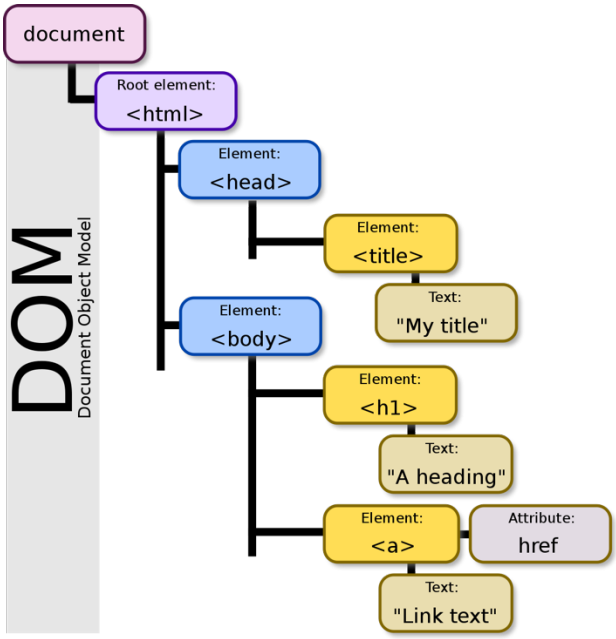

Рисунок 11.1 – Объектная модель документа на примере HTML-документа

Основы работы с пакетом rvest

Пакет rvest предполагает универсальное средство для веб-скрапинга, реализованного по модели парсинга DOM. Основные функции пакета, используемые для веб-скрапинга:

`read_html()` – функция принимающая в качестве аргумента адрес веб-страницы и выводящая лист класса "xml_document" "xml_node"

`html_nodes()` – функция позволяющая извлечь нужные части документа, принимает в качестве первого аргумента веб-страницу, полученную с помощью функции read_html(), а в качестве второго – название селектора CSS.

`html_text()` – функция переводящая в текст результаты, полученные с помощью функции html_nodes(), на вход функции передается вывод функции html_nodes().

Сведения о структуре сайта IMDB.com

Для извлечения данных будут использоваться сведения с сайта Internet Movie Database. Общий вариант ссылки будет иметь следующий вид:
https://www.imdb.com/search/title/?count=1000&genres=action&sort=user_rating,desc&title_type=feature

Синтаксис по типу: ?переменная=значение можно регулировать под необходимые цели. Описание некоторые переменных: count – количество в выдаче, genre – жанр, sort – сортировка по значению, title_type – тип (например, художественные), release_date – дата выхода (принимает два значения через запятую, например 2016,2017 будет означать что представлены фильмы с 1 января 2016 по 31 декабря 2017).

Селекторы сайта imdb.com:

.text-primary – номер
.lister-item-header a – заголовок
.ratings-bar+ .text-muted – описание фильма
.text-muted .runtime – продолжительность фильма
.genre – жанр
.ratings-imdb-rating strong – рейтинг

ПРАКТИЧЕСКАЯ ЧАСТЬ

Задание 1 – Используя данные варианты постройте датафрейм, содержащий следующие сведения о художественных фильмах: название, жанры (все жанры к которым относится фильм), продолжительность, рейтинг IMDB.

Вариант 1 – Комедии 2008 года.
Вариант 2 – Триллеры 1996 года.
Вариант 3 – Драмы 2012 года.
Вариант 4 – Научная фантастика 2019 года.
Вариант 5 – Приключения 2014 года.
Вариант 6 – Комедии 2013 года.
Вариант 7 – Триллеры 1992 года.
Вариант 8 – Драмы 2001 года.
Вариант 9 – Научная фантастика 2011 года.
Вариант 10 – Приключения 1993 года.
Вариант 11 – Комедии 1998 года.
Вариант 12 – Триллеры 2017 года.
Вариант 13 – Драмы 1985 года.
Вариант 14 – Научная фантастика 1990 года.
Вариант 15 – Приключения 2003 года.
Вариант 16 – Комедии 1995 года.
Вариант 17 – Триллеры 1997 года.
Вариант 18 – Драмы 2014 года.
Вариант 19 – Научная фантастика 2000 года.
Вариант 20 – Приключения 2019 года.

Контрольные вопросы:

1. Опишите принцип построения исходный ссылки веб-скрапинга сайта.
2. Какие типы данных оптимально использовать для хранения собранных данных?