Оглавление

[1. О программе 2](#_Toc364290249)

[2. Входные файлы 2](#_Toc364290250)

[2.1 Формат файла для парсинга текстовой информации 2](#_Toc364290251)

[2.2 Формат файла для парсинга изображений 2](#_Toc364290252)

[2.3 Формат файла с настройками 3](#_Toc364290253)

[2.4 Формат файла с прокси-сервером 4](#_Toc364290254)

[2.5 Формат файла с URL для парсинга 4](#_Toc364290255)

[3 Библиотека HtmlAgilityPack 4](#_Toc364290256)

[4 Аргументы командной строки 4](#_Toc364290257)

[3 XPath 5](#_Toc364290258)

# О программе

Программа была написана в Visual Studio 2013 на языке С# 5.0 .NET Framework 4.5 с использованием библиотеки HtmlAgilityPack 1.4.6.

# Входные файлы

# Формат файла для парсинга текстовой информации

Файл для парсинга текстовой информации состоит из:

***имя тега выходного XML-файла:XPath для получения текстовой информации***

Имя тега выходного XML-файла может содержать буквы, цифры, символ ‘\_’.

Формат XPath для получения текстовой информации описан в пункте 4.

Если XPath отсутствует, то будет создан открывающий блочный тег. Для создания закрывающего блочного тега необходимо ещё раз указать имя тега и добавить к нему символ ’.’.

Пример:

|  |  |
| --- | --- |
| Содержимое файла для парсинга текстовой информации | XML файл |
| product:  price://div[@class='price']  namber://\*[@id='item\_new']/h1/span  name://\*[@id='item\_new']/h1/text() | <?xml version="1.0" encoding="utf-8"?>  <catalog id="http://www.citilink.ru">  <product>  <price>7 200</price>  <namber>793123</namber>  <name>Компьютер IRU Corp 319, Intel Celeron G470, DDR3&amp;nbsp;2Гб, 250Гб, Intel HD&amp;nbsp;Graphics, без OS, черный</name>  </product>  </catalog> |

# Формат файла для парсинга изображений

Файл для парсинга текстовой информации состоит из:

***имя получаемого изображения:XPath для получения изображения***

Имя получаемого изображения является не обязательным.

Формат XPath для получения текстовой информации описан в пункте 4.

# Формат файла с настройками

Файл с настройками имеет следующий вид:

***Имя параметра:значение параметра***

Список параметров:

|  |  |
| --- | --- |
| Имя параметра | Описание |
| inputURLsFileName | имя файла с адресами для парсинга |
| outputXMLFileName | имя выходного XML-файла |
| inputContentPropertiesFileName | имя файла для парсинга текстовой информации |
| inputImagesPropertiesFileName | имя файла для парсинга изображений |
| inputProxysFileName | имя файла с прокси-сервером |
| pathToOutputContent | путь, где будет создана папка “images”, в которую будут сохраняться изображения |
| isAutomatically | True – режим автоматического парсинга,  False – режим полуавтоматического парсинга  (только для сайта citilink.ru) |
| isParseContent | True – парсить текстовую информацию,  False – не парсить текстовую информацию |
| isParseImages | True – сохранять изображения,  False – сохранять изображения |
| isProxy | True – использовать прокси,  False – не использовать прокси |
| freezTime | Значение времени задержки между обращениями к серверу (в милисекундах). |

# Формат файла с прокси-сервером

Файл с прокси-сервером имеет следующий вид:

***IP:Port***

# Формат файла с URL для парсинга

Файл со ссылками для парсинга содержит ссылки для парсинга, разделённые переводом строки.

# Библиотека HtmlAgilityPack

Данный парсер написан с использованием библиотеки HtmlAgilityPack. Данная библиотека осуществляет обращение к серверу (загрузку страницы) и парсинг текстовой информации по заданному XPath.

Рассмотрим работу библиотеке на примере:

HtmlDocument doc = new HtmlDocument();

List<string> att = new List<string>();

doc.Load("http://mywebsite.com/file.htm");

foreach(HtmlNode link in doc.DocumentElement.SelectNodes("//div[@class=’myclass’"])

{

att.Add(link.InnerText);

}

doc.Save("file.htm");

Данный код на языке C# загружает страницу по адресу “http://mywebsite.com/file.htm” и получает весь внутренний текст всех тегов “div”, у которых атрибут «class» имеет значение «myclass», и сохраняет результат в массив att, а также код самой страницы в файл “file.htm”.XPath

# Аргументы командной строки

Данная программа может быть запущена с аргументами командной строки в обход запуска графического интерфейса. Для этого необходимо указать программе веб-сайт, который будет парситься. Если парсинг сайта уе реализован в коде, то программа автоматически начнёт его парсить. Остальную информацию программа получит из файла “settings.txt”.

Пример:

websiteParser.exe <http://www.citilink.ru>

Список сайтов, парсинг которых реализован:

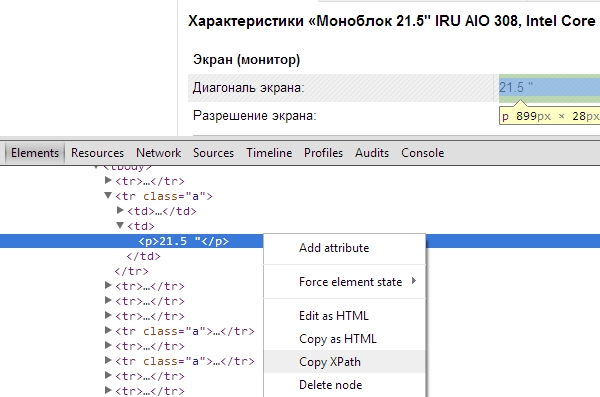
1. <http://www.citilink.ru>
2. <http://www.dns-shop.ru>

# XPath

Для получения информации из DOM-дерева, построенного при помощи библиотеки HtmlAgilityPack используется XPath. Для его получения можно воспользоваться браузером.

Пример 1:

Воспользуемся браузером Google Chrome 28.0.1500.95 m. Для этого выберем нужный узел в разметке и скопируем XPath для нужного узла.



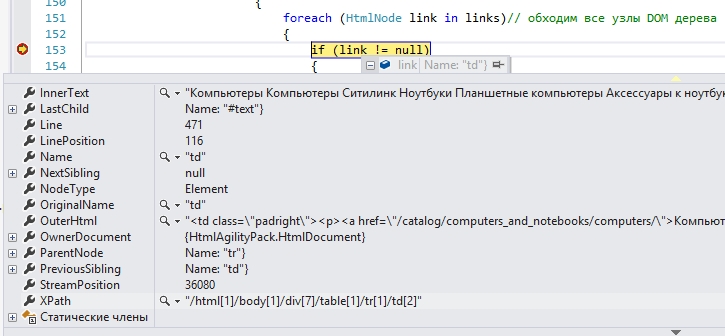
В итоге получим: //\*[@id="t1"]/div[1]/table/tbody/tr[2]/td[2]/p

Аналогично в Mozilla FireFox 22.0 с использованием плагина: id('t1')/x:div[1]/x:table/x:tbody/x:tr[2]/x:td[2]/x:p

Ни один из полученных XPath не является валидным. Необходимо удалить узел tbody и p из полученного XPath (теги, встречающиеся в дереве 1 раз и не имеющие атрибуты не являются узлами дерева). Получим //\*[@id="t1"]/div[1]/table/tr[2]/td[2]. Теперь по данному XPath будет получен нужный узел.

Пример 2:

XPath можно посмотреть в отладчике среды разработки для конкретного узла.



Случай, когда XPath заранее не известен и его придётся получать каким-либо способом (конкатенация строк и т.д.) необходимо реализовывать в коде.

Список источников:

1. <http://htmlagilitypack.codeplex.com/>
2. <http://habrahabr.ru/post/112325/>
3. <http://blog.baskovsky.ru/2012/11/html-agility-pack.html>
4. <http://nesteruk.wordpress.com/tag/html-agility-pack/>
5. <http://kronus.me/2011/04/введение-в-xpath-на-примере-простого-парсе/>
6. [http://ru.wikipedia.org/wiki/XPath](http://ru.wikipedia.org/wiki/XPath#.D0.A4.D1.83.D0.BD.D0.BA.D1.86.D0.B8.D0.B8_.D1.81_.D0.BC.D0.BD.D0.BE.D0.B6.D0.B5.D1.81.D1.82.D0.B2.D0.B0.D0.BC.D0.B8)