

Технологии автоматизации сбора данных

Технологии автоматизации сбора данных

Парсеры free edition =)

Чем будем заниматься?

- Что такое парсинг?
- Инструменты для парсинга
- Немного почешем сайты
- Посмотрим на свой цифровой след
- Будем брать паузы на поговорить

P.S: в пайтон версии

Что такое парсинг?

Нууу, это процесс сбора данных
а что ещё говорить-то?

А, ну, ещё программный,
желательно, дааа

The ichor permeates MY FACE MY FACE oh god no NO NOO



*Parsing HTML Using
Regular Expressions*

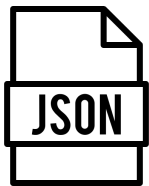
NO stop the an *es are not real ZALGO, HE COMES

O RLY?

DEMON

А что парсим?

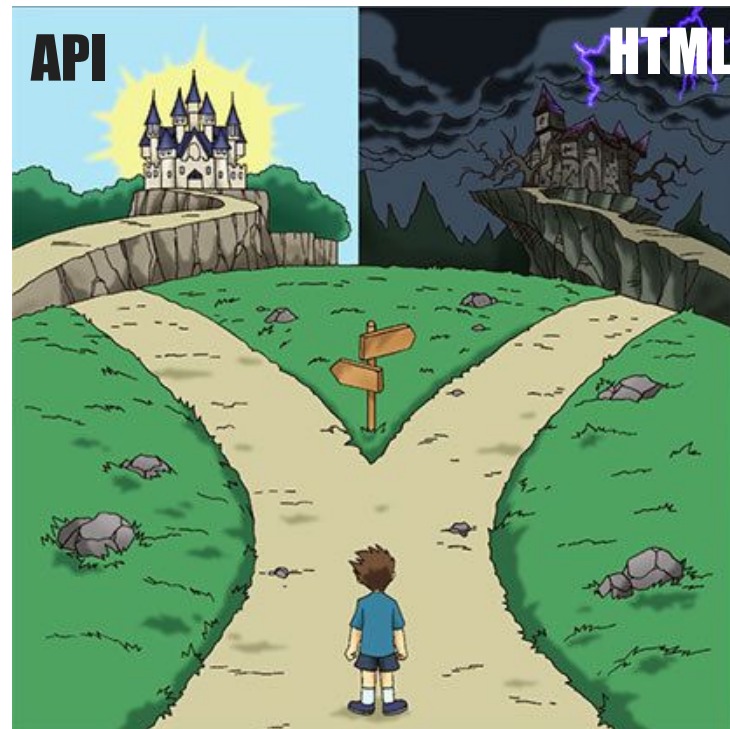
Логично, что данные, но какие?



Веб-краулинг для малышей

Чего хотим?

- Просить у сайта страницы
- Просить у сайта API
- Знать, как это делать



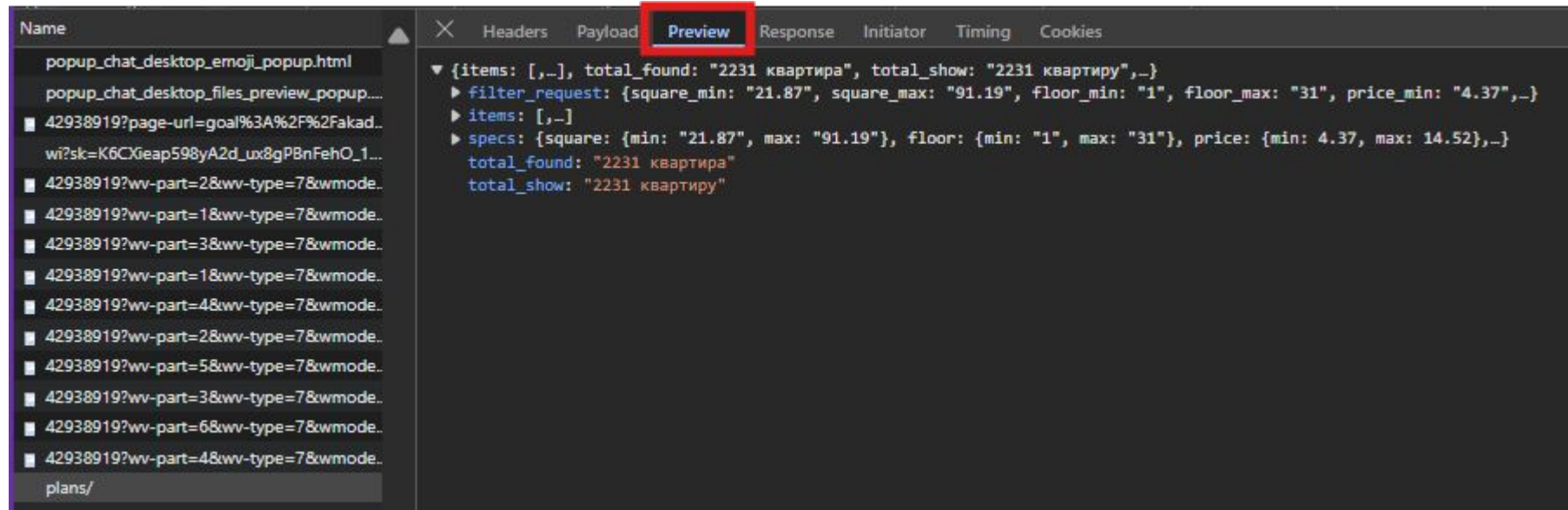
Ищем API: devtools -> network

The screenshot shows the Chrome DevTools Network tab. The top bar has the 'Network' tab selected. The filter 'Fetch/XHR' is applied. The list of requests on the left includes several 'popup_chat_desktop' requests and a series of '42938919?wv-part=...' requests. The last request, 'plans/', is highlighted. The right pane shows the details for this request:

General	
Request URL	https://akademicheskij.org/plans/
Request Method	POST
Status Code	200 OK
Remote Address	217.198.81.229:443
Referrer Policy	strict-origin-when-cross-origin

Response Headers	
Cache-Control	no-store, no-cache, must-revalidate
Content-Encoding	gzip
Content-Length	6657
Content-Type	text/html; charset=utf-8
Date	Sun, 06 Jul 2025 16:00:36 GMT
Expires	Thu, 19 Nov 1981 08:52:00 GMT

Ищем API: анализируем ответ



The screenshot shows a web browser's developer tools with the 'Network' tab selected. A list of network requests is on the left, and the 'Preview' tab of a selected request is on the right. The 'Preview' tab shows a JSON response with the following structure:

```
{items: [...], total_found: "2231 квартира", total_show: "2231 квартиру",...}
  filter_request: {square_min: "21.87", square_max: "91.19", floor_min: "1", floor_max: "31", price_min: "4.37",...}
  items: [...],
  specs: {square: {min: "21.87", max: "91.19"}, floor: {min: "1", max: "31"}, price: {min: 4.37, max: 14.52},...}
  total_found: "2231 квартира"
  total_show: "2231 квартиру"
```

The 'Preview' tab is highlighted with a red box. The 'filter_request' object contains the following data:

Property	Value
square_min	"21.87"
square_max	"91.19"
floor_min	"1"
floor_max	"31"
price_min	"4.37"

The 'specs' object contains the following data:

Property	Value
square	{min: "21.87", max: "91.19"}
floor	{min: "1", max: "31"}
price	{min: 4.37, max: 14.52}

Тестируем API



insomnia



POSTMAN

CURL



Позиционирует себя, как платформа для desing и work с API.

Используем для анализа запросов:

- Смотрим внутренности запроса
- Дедовский метод проверки защиты сайта
- Экономим своё время

Postman



Генерим сниппет в
реальном времени и
ничего не делаем
ручками =)



Code snippet



Python - Requests



```
1 import requests
2
3 url = "https://basket-18.wbbasket.ru/vol3032/part303240/303240677/info/ru/card.
    json"
4
5 payload = {}
6 headers = {
7     'accept': '*/*',
8     'accept-language': 'ru,en;q=0.9,en-GB;q=0.8,en-US;q=0.7',
9     'origin': 'https://www.wildberries.ru',
10    'priority': 'u=1, i',
11    'referer': 'https://www.wildberries.ru/catalog/303240677/detail.aspx',
12    'sec-ch-ua': '"Not)A;Brand";v="8", "Chromium";v="138", "Microsoft Edge";v="138"',
13    'sec-ch-ua-mobile': '?0',
14    'sec-ch-ua-platform': '"Windows"',
15    'sec-fetch-dest': 'empty',
16    'sec-fetch-mode': 'cors',
17    'sec-fetch-site': 'cross-site',
18    'user-agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
    (KHTML, like Gecko) Chrome/138.0.0.0 Safari/537.36 Edg/138.0.0.0'
19 }
20
21 response = requests.request("GET", url, headers=headers, data=payload)
22
23 print(response.text)
24
```

GET

https://www.ozon.ru/highlight/globalpromo/

Send

Params

Authorization

Headers (21)

Body

Pre-request Script

Tests

Settings

Cookies

Query Params

	Key	Value	Bulk Edit
	Key	Value	

Body

Cookies (2)

Headers (8)

Test Results

Status: 403 Forbidden

Time: 118 ms

Size: 7.04 KB

Save Response

Pretty

Raw

Preview

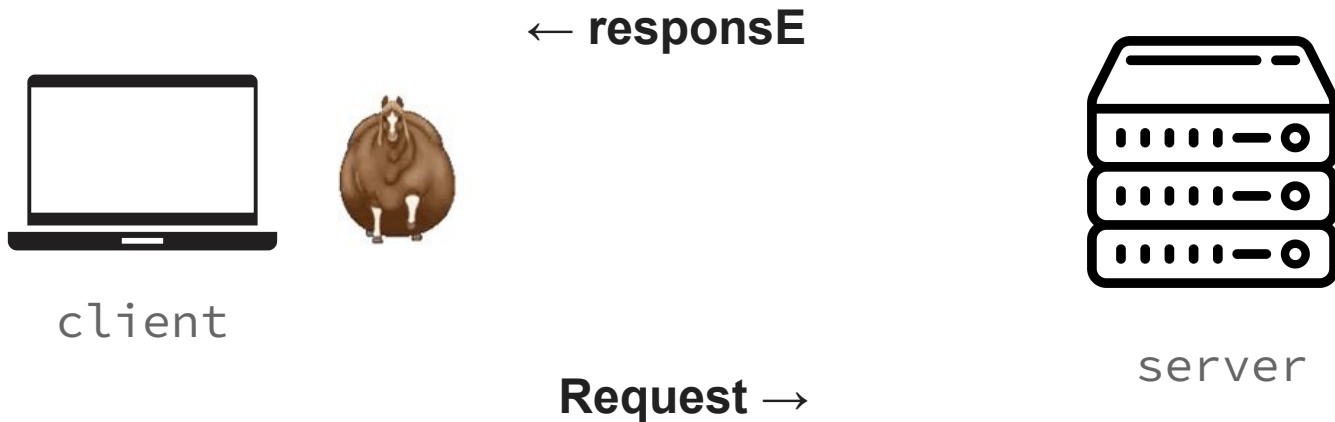
Visualize

HTML

```
15 <noscript>
16   <div class="container">
17     <div class="message">
18       <div class="variant">
19         <h2 class="title">Пожалуйста, включите JavaScript для продолжения</h2>
20         <span class="subtitle">Нам нужно убедиться, что вы не робот.</span>
21       </div>
22       <div class="variant" lang="en">
23         <h2 class="title">Please, enable JavaScript to continue</h2>
24         <span class="subtitle">We need to make sure that you are not a robot.</span>
25       </div>
26     </div>
27     <div class="details">
28       <span class="details-text"><b>ID:</b> fab_chlg_20250706145529_01JZG3RAWNEP9KG9K6G7D7BWM9</span>,
```

Это конечно всё круто, но что из себя представляет http-запрос?

Это конечно всё круто, но что из себя представляет http-запрос?



Http-request

Сообщение, которое клиент отправляет серверу, чтобы получить ресурс

Состоит из:

- строки запроса
- заголовков
- тела (если запрос на изменение данных)

Http-request

Сообщение, которое клиент отправляет серверу, чтобы получить ресурс

Состоит из:

- строки запроса
- заголовков
- тела

<https://www.ozon.ru/api/entrypoint-api.bx/?ozon=fu>

User-Agent, Accept, Referer, etc.

{"asyncData":"blablabla"}

GET



https://www.ozon.ru/

Send



Params Authorization **Headers (21)** Body Pre-request Script Tests Settings

[Cookies](#)

<input checked="" type="checkbox"/>	accept	text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,im...
<input checked="" type="checkbox"/>	accept-language	ru,en;q=0.9,en-GB;q=0.8,en-US;q=0.7
<input checked="" type="checkbox"/>	cache-control	no-cache
<input checked="" type="checkbox"/>	priority	u=0, i
<input checked="" type="checkbox"/>	referrer	https://ntp.msn.com/
<input checked="" type="checkbox"/>	sec-ch-ua	"Not)A;Brand";v="8", "Chromium";v="138", "Microsoft Edge";v="138"
<input checked="" type="checkbox"/>	sec-ch-ua-mobile	?0
<input checked="" type="checkbox"/>	sec-ch-ua-platform	"Windows"
<input checked="" type="checkbox"/>	sec-fetch-dest	document
<input checked="" type="checkbox"/>	sec-fetch-mode	navigate
<input checked="" type="checkbox"/>	sec-fetch-site	same-origin
<input checked="" type="checkbox"/>	sec-fetch-user	?1
<input checked="" type="checkbox"/>	upgrade-insecure-requests	1
<input checked="" type="checkbox"/>	user-agent	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Ge...

User-Agent

Ваш паспорт в мире веб-краулинга

Mozilla/5.0 (Windows NT 10.0; Win64; x64)

AppleWebKit/537.36 (KHTML, like Gecko)

Chrome/138.0.0.0 Safari/537.36 Edg/138.0.0.0

P.S: демонстрирует ваш браузер, его версию, ОС и тип устройства

Плохой агент

- Mozilla/5.0 ((Windows NT 10.0; Win64; x64))
- python-requests/Mozilla/5.0 (Windows NT 10.0; Win64; x64)
- пустой агент
- (Windows NT 10.0; Win64; x64), а в заголовке sec-ch-ua-platform стоит “Android”

Accept и Content-Type



Cookie

Идентифицируют вашу сессию. Это маленькие фрагменты данных, которые запоминают вашу сессию: время, проведенное на странице, ваш регион, предпочтения, пройденные капчи и т.д. с целью сообщить об этом серверу посредством запросов.

Сервер в свою очередь тоже что-то сообщает, но не вам, а вашему браузеру. Этаким взаимолайк с целью побольше за вами последить.

Куки хранятся файлами на вашем ПК :)

Как это проверяется?

Обработка заголовков происходит в нескольких уровнях приложения:

- Балансировщик нагрузки или фаервол (user-agent, referer, origin)
- Сервер (user-agent, connection, encoding)
- Приложение/бекенд-код (auth, accept, cookie, content-type)

Не нашли API: дёргаем страницы

Увы и ах, но не всегда в жизни идёт всё, как нам хочется

Некоторые сайты не предоставляют открытый API и нам приходится расчехлять грустный, но 100%-ный вариант - html-простыню сайта.

Дёргаем страницы

The screenshot shows the Chrome DevTools Network tab. The top timeline shows a request at approximately 100,000 ms. The left sidebar lists the loaded resources, with `akademicheskij.org` highlighted. The right pane shows the details of the selected request.

Request Details:

- Request URL:** `https://akademicheskij.org/`
- Request Method:** `GET`
- Status Code:** `200 OK`
- Remote Address:** `217.198.81.229:443`
- Referrer Policy:** `no-referrer-when-downgrade`

Response Headers:

- Cache-Control:** `no-store, no-cache, must-revalidate`
- Content-Encoding:** `gzip`
- Content-Length:** `20279`
- Content-Type:** `text/html; charset=utf-8`

Технологии автоматизации сбора данных

Парсеры not so free edition =)

Веб-краулинг для уже смешариков

Чего хотим?

- Игрушки помощнее
- Спарсить Озон (без бана)
- Подвести итоги



Предпосылки быть смешариком

Не каждый сайт страдает благотворительностью. С каждым днём всё больше сайтов от малых до великих хотят защититься от злых ботов, парсеров и бедных студентов хитрыми и не очень способами.

У кого-то получается, а у кто-то не вывозит нагрузку
cloudflare =)

А мы посмотрим, кто на что горазд

Как защищаются сайты?

Способов защит бесконечное множество в своих изящных вариациях, но есть типовые:

- Анализ заголовков запроса (отдельно проезжаются по кукам)
- Rate limiting
- JS Challenge
- CLOUDFLARE (со всеми вытекающими капчами)
- Captcha разных видов и сортов
- Поведенческий анализ
- Fingerprinting

Rate limiting

При слишком частом обращении к сайту (а парсер можно разогнать очень сильно) нам может выстрелить в ногу 429 (Too Many Requests).

Более того, мы собственноручно можем отправить сайт отдыхать в таверну без байбека и не получить так нужные нам данные :)

Rate limiting: контра

Добросовестный краулер обязан уважать то место, где он ест. Поэтому максимум, что я могу вам посоветовать, так это рассчитывать свои и не только силы и не перестараться.

Тем не менее, есть ряд кодов ответа, которые мы можем подвергнуть попытке Retry:

Когда мы можем:

429 (Too Many Requests)
500 (Internal server error)
502 (Bad Gateway)
503 (Service Unavailable)
504 (Gateway Timeout)

Когда это бесполезно:

(Bad Request) 400
(Unauthorized) 401
(Forbidden) 403
(Not Found) 404
(Method Not Allowed) 405

Новые игрушки: эмуляторы

В момент встречи с JS Challenge или просто сайтом с динамической подгрузкой данных мы понимаем: простыми библиотеками тут не обойтись.

На помощь приходят три богатыря





Проверенный временем и появился раньше двух своих собратьев. Позиционируется, как инструмент для программного управления браузером, и, как следствие, автоматизации различных коммерческих процессов.

Например, тестирование и, конечно же, наш любимый парсинг :)

к сожалению, обычными версиями наших трёх богатырей едва ли можно справиться с фингерпринтом, даже playwright'у не под силу (хотя во всех источниках пишут обратное)

Fingerprint

Цифровой отпечаток вашего браузера – это куча характеристик одного, собранная в один хэш.

По последнему сайт мониторит ваши посещения и наличие автоматизации в сессии

Фингерпринт собирает в себе многое:

- Характеристики браузера
- Характеристики ОС
- Размер экрана
- Плагины
- Языки и локали
- Графические характеристики (информация о гри, отрисовка пикселей)

Fingerprint: наш ответ

Чисто теоретически с обычным селеном его можно подделать вручную, но у вас не хватит нервов, сил и времени, чтобы перечесать каждый параметр, который вас спалит (

Поэтому идейные люди дали одели наших богатырей в мехи, заставили сесть в робота, и получилось вот это:

UNDETECTED/ABLE/STEALTH



Спасибо за внимание

Надеюсь, вы слушали