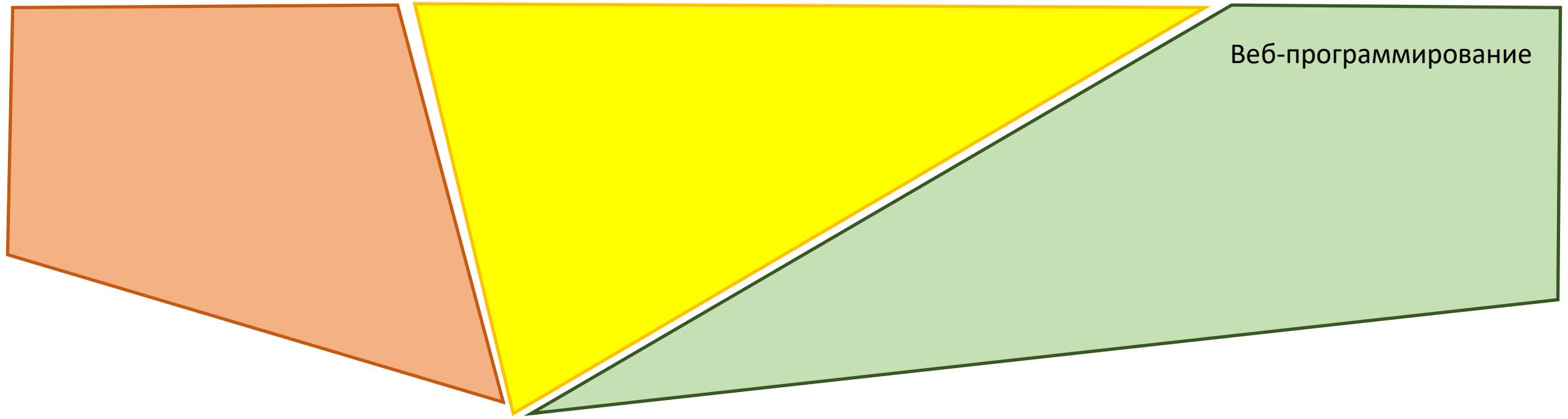


О курсе

- 9 лекций
- 6-7 лабораторных работ
- Итоговый тест
- Зачет

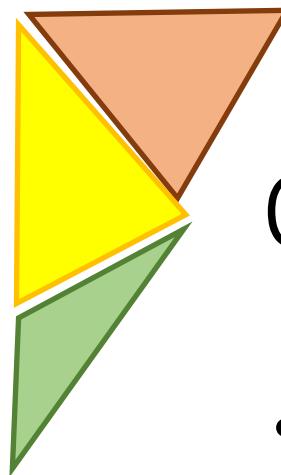
[https://github.com/atatarinova/web course vyatsu](https://github.com/atatarinova/web_course_vyatsu)



Веб-программирование

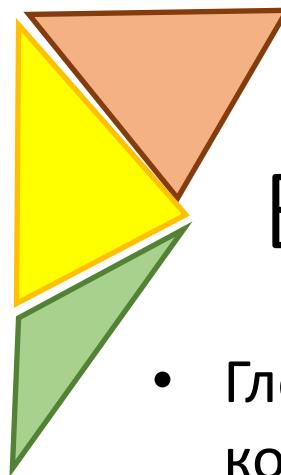
Всемирная паутина HTTP

Татаринова А.Г., каф. ПМИ



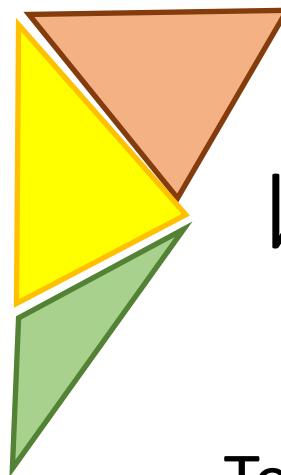
Основные понятия

- Интернет
- Провайдер
- Всемирная паутина (веб)
- Веб-ресурс
- Веб-сервис
- Браузер
- Гиперссылка
- Веб-сервер
- Веб-клиент
- URL
- HTTP
- SSL
- DNS
- DHCP
- SSH



Всемирная паутина

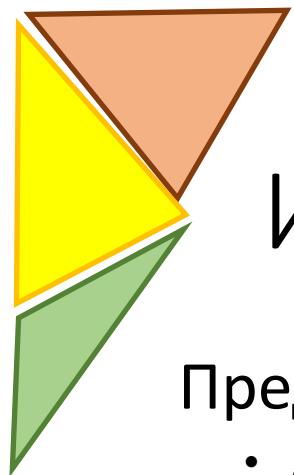
- Глобальная **гипертекстовая** система, построенная на открытых веб-стандартах, которая предоставляет доступ к взаимосвязанным ресурсам по URL через протоколы семейства HTTP(S) и интерпретируется веб-клиентами (браузерами и программными агентами)
- Распределённая **система ресурсов** (документы, данные, приложения), связанных гиперссылками и адресуемых URI, доставляемых по Интернету с использованием HTTP/HTTPS и описываемых веб-стандартами W3C; это надстройка над сетевой инфраструктурой Интернета



История появления Web: идея

Теоретические основы были заложены раньше, например:

- В 1945 г. Вэнивер Буш разработал концепцию "расширения человеческой памяти" Мемех (устройство, в котором человек хранит свои книги, записи и контакты и которое «выдаёт нужную информацию с достаточной скоростью и гибкостью»)
- В 1965 г. термин гипертекста был введён Тедом Нельсоном
 - Система Hanadu (1960 г.)
 - создание единого адресного пространства доступного содержимого с уникальным адресом для каждого байта данных, и виртуальных файлов, которые собирают, обрабатывают и выводят информацию пользователю из единого пространства



История появления Web

Предпосылки:

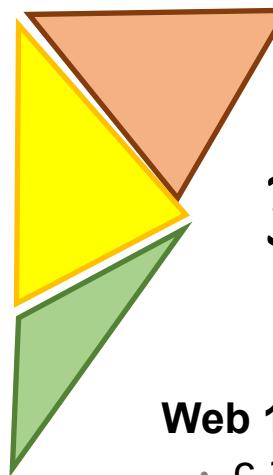
- ARPANET (1969)
 - компьютерная сеть коммутирующая пакеты с промежуточным хранением -> развитие технологий построения глобальных сетей
- 70-е гг.
 - появление БИС,
 - развитие технологий построения локальных сетей
- 1971 г.
 - отправка первого email
- 80-е гг.
 - стек протоколов TCP/IP, стандартизация протоколов
- 1984 г.
 - запуск DNS

1989 г. – появление HTTP, URL, HTML (Тим Бернс-Ли)

1990 г. – появление первого веб-сервера

1991 г. – веб-сервер выложили в Интернет

90-е гг. – появление графических браузеров, а также CSS, JavaScript, php и других технологий



Эволюция Web

Web 1.0 — Read-Only Web

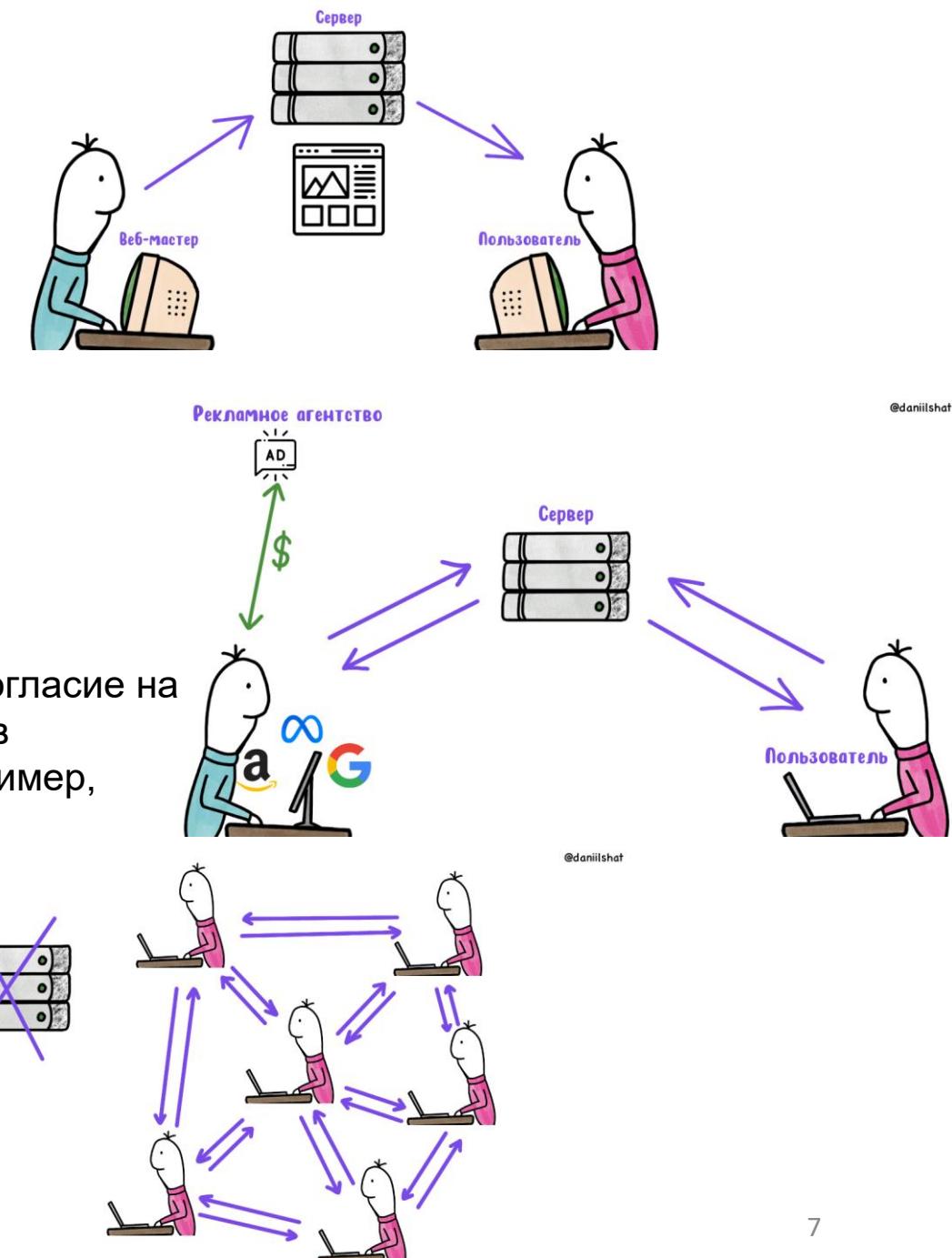
- с 1991 по 2001 (2004) год
- статичное содержание веб-страниц
- аскетичный дизайн
- форумы и гостевые книги

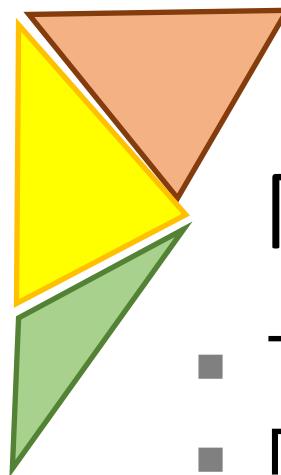
Web 2.0 — Read/Write Web

- с 2001 (2004) до текущего времени
- пользователи добровольно оставляют свои данные и дают согласие на сбор данных взамен на возможность использования ресурсов
- компании имеют возможность зарабатывать на данных, например, продавая их рекламным агентствам
- социальные функции и персонализация
- JSON, XML
- обновление страниц “на лету” (ajax)

Web 3.0 — ???

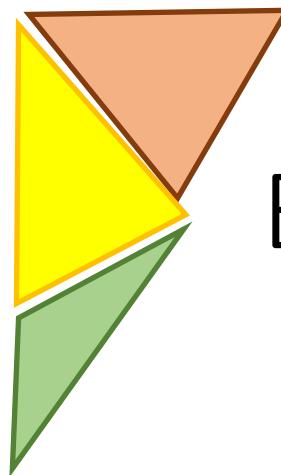
- семантическая паутина
- децентрализация, открытость, ИИ
- уникальные идентификаторы и микроформаты





Примеры веб-приложений

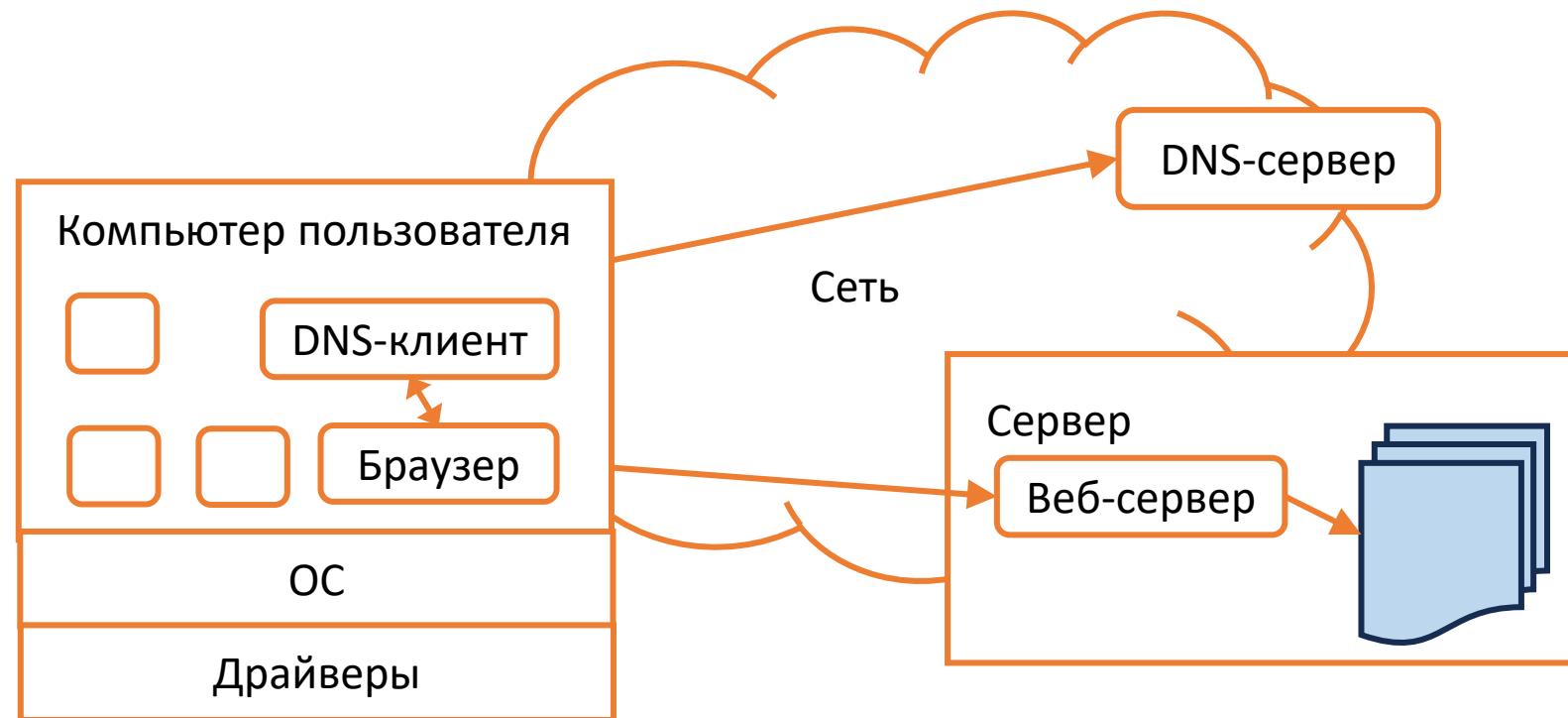
- Традиционные сайты: новости, блоги, wiki
- Потоковые сервисы: Нетфликс, Spotify
- Магазины: Aliexpress, Avito, RZD
- Образование: Степик, Udemy
- Бизнес: Zoom, Teams, Notion, PayPal
- Поиск: Яндекс поиск, Google поиск, DuckDuckGo
- Замена desktop приложений: Google Docs, Trello, Dropbox, 1C online

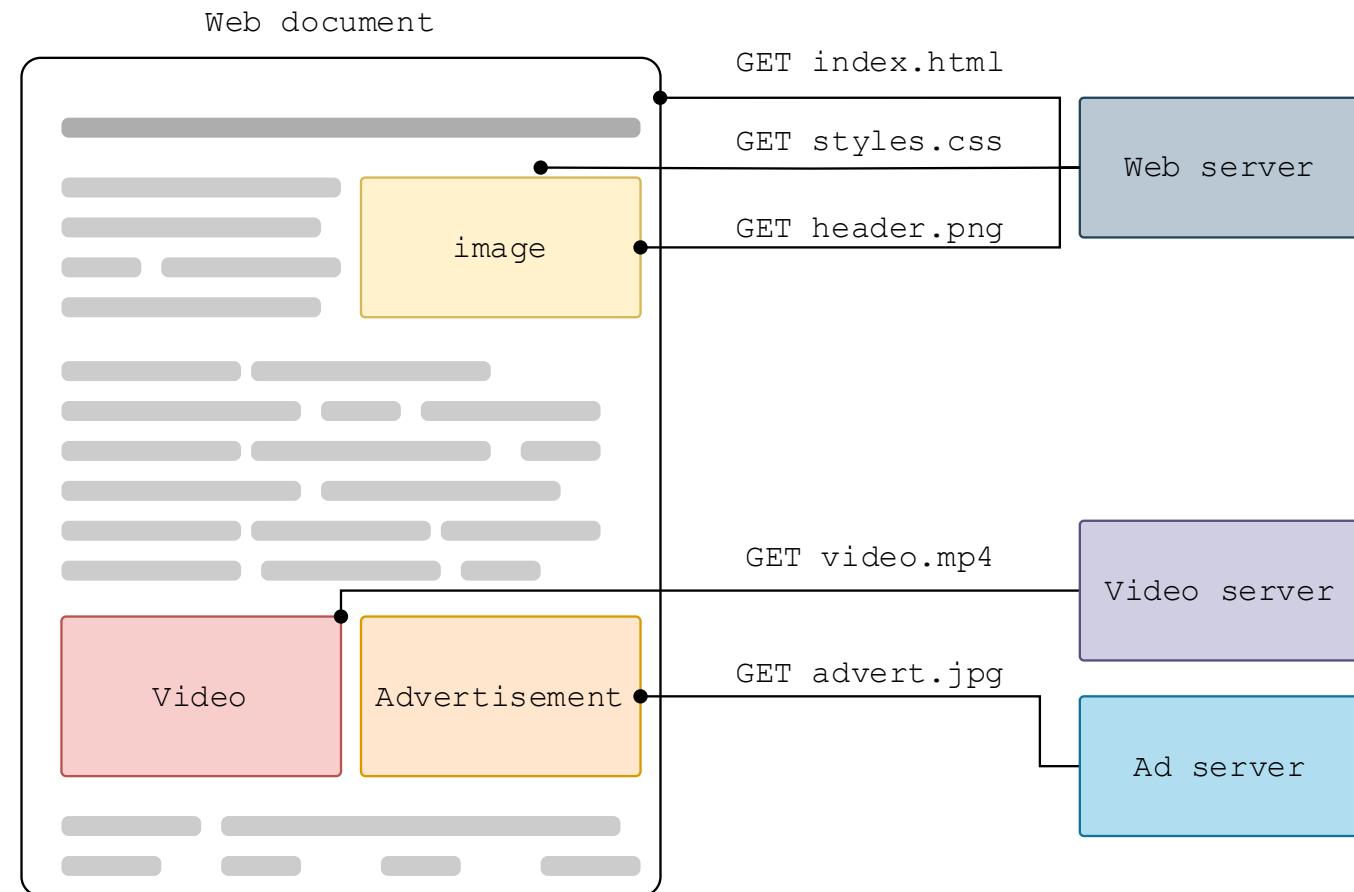


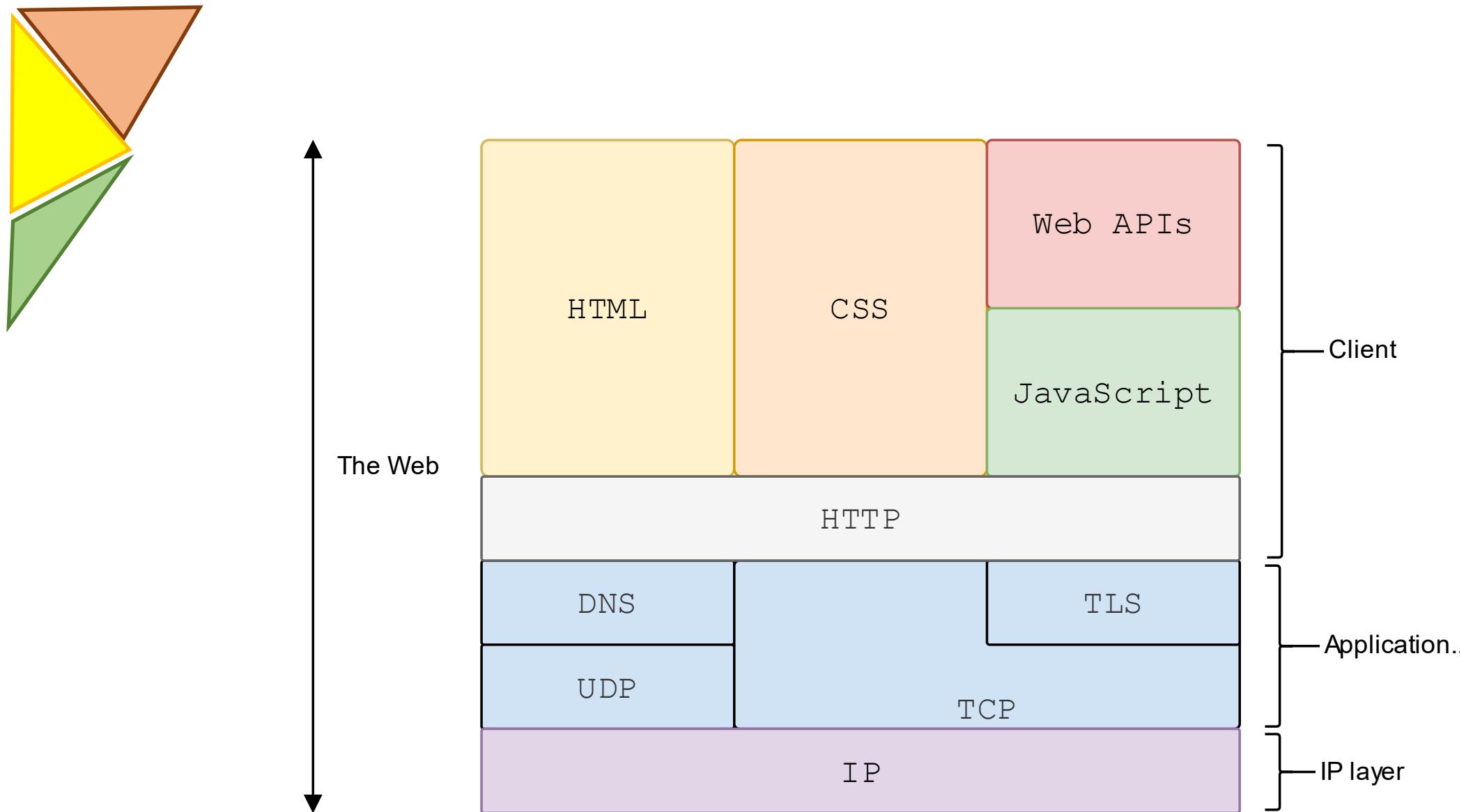
Веб разработка и сопровождение

- Front-end разработка
- Back-end разработка
- Разработка под мобильные платформы
- Инфраструктура
- Архитектура
- Системное программирование
- Прочее (seo, security, management)

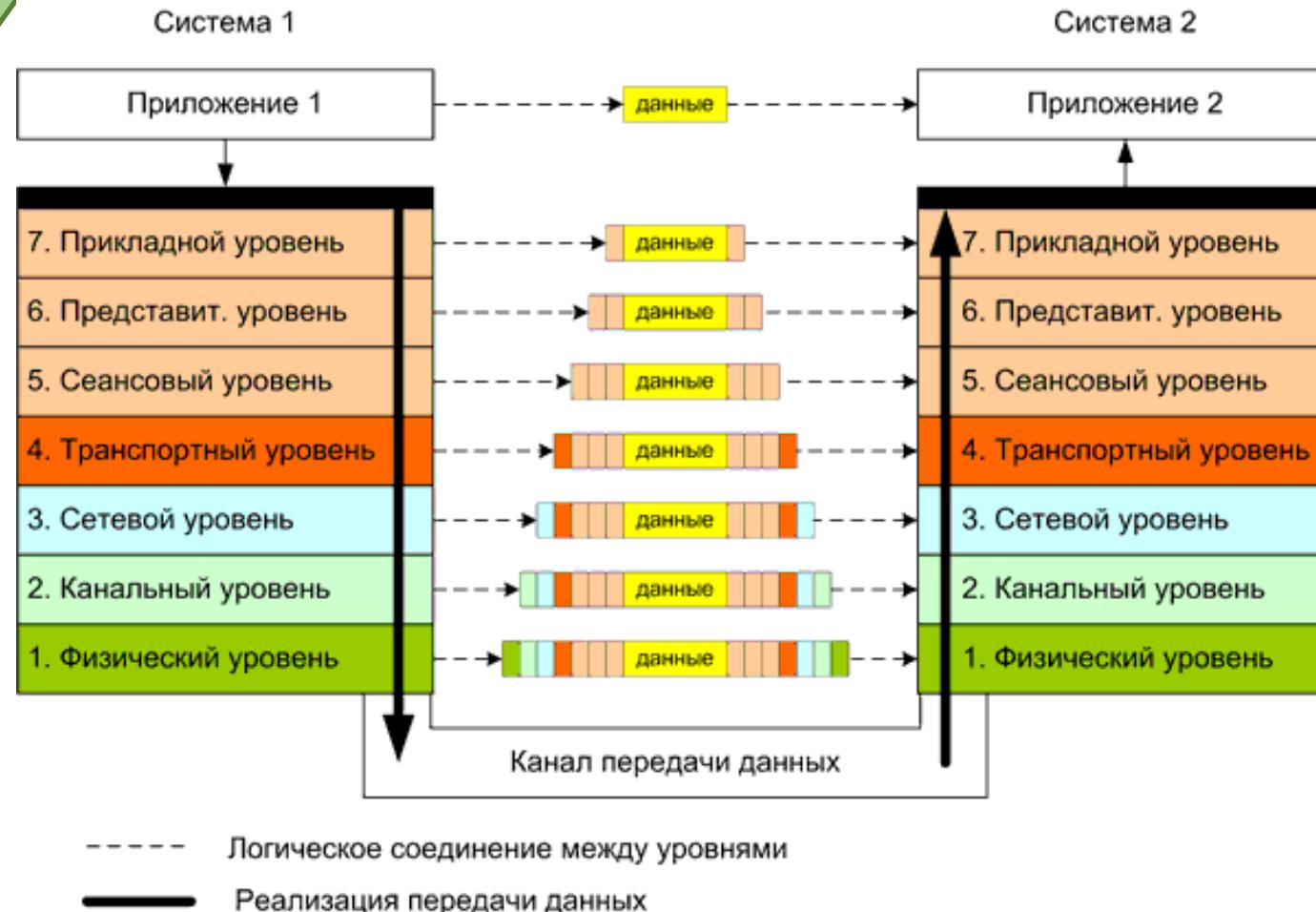
Общая схема доступа к веб-ресурсам



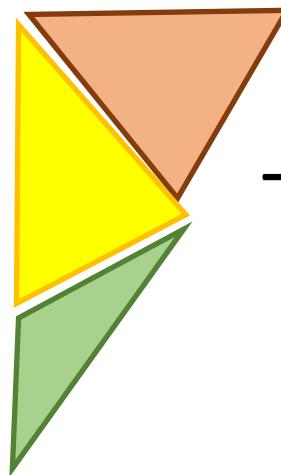




Модель OSI



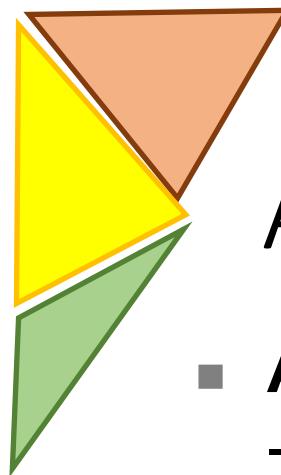
- **Физический:** управление средой передачи данных и сигналом, кодирование потока битов
- **Канальный:** физическая адресация, передача между двумя интерфейсами
- **Сетевой:** логическая адресация, передача данных между двумя хостами, маршрутизация
- **Транспортный:** надежная доставка, передача между двумя приложениями, управление потоком байтов



TCP протокол

- Последовательное двустороннее соединение
- Надежная доставка данных
- Управление потоком данных

- Сокеты (программный интерфейс) – структура данных, которая идентифицирует сетевое соединение
(определяется **сетевым адресом и номером порта**)



Адреса в сети

- **Аппаратные**

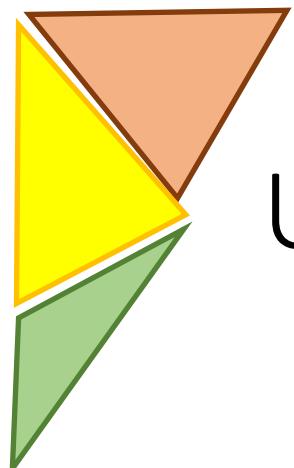
- физический адрес сетевой карты
 - MAC-адрес, пример 00-E1-25-10-1E-D3

- **Сетевые**

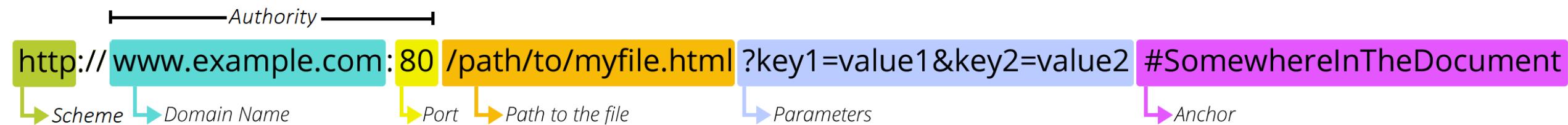
- ip-адреса, пример 192.168.10.153 (четыре восьмивитных числа в десятичном виде)
 - ip-адрес характеризует не отдельный компьютер или маршрутизатор, а одно сетевое соединение

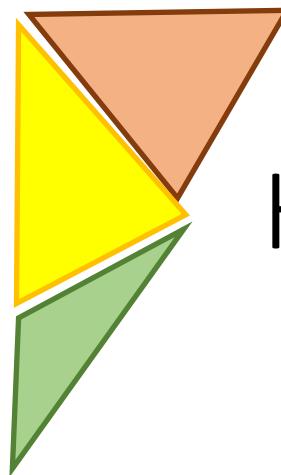
- **Доменные**

- символные имена, служат для удобства представления IP-адресов
 - служба, DNS (Domain Name System), устанавливает соответствие между IP и символными адресами



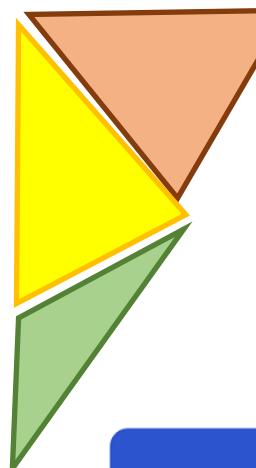
URL (Uniform Resource Locator)



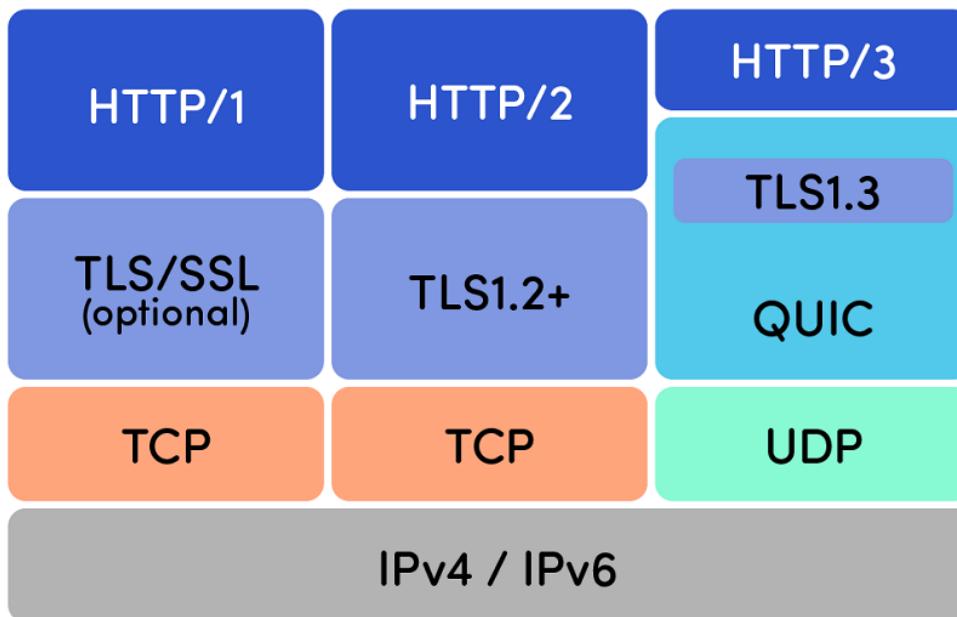


HTTP протокол

- Протокол передачи гипертекста
- Основой является клиент-серверная архитектура
- Текстовый протокол
- HTTP-запрос, HTTP-ответ
- Описывается спецификацией RFC 2616.
- На сегодняшний день наиболее распространенной версией протокола является версия HTTP/2, однако часто можно встретить более раннюю версию HTTP/1.1. Также встречается версия HTTP/3



Версии протокола HTTP



HTTP/1

Веб-браузер делает несколько параллельных запросов для содержания страницы: HTML, изображения, стили, JS

HTTP/2

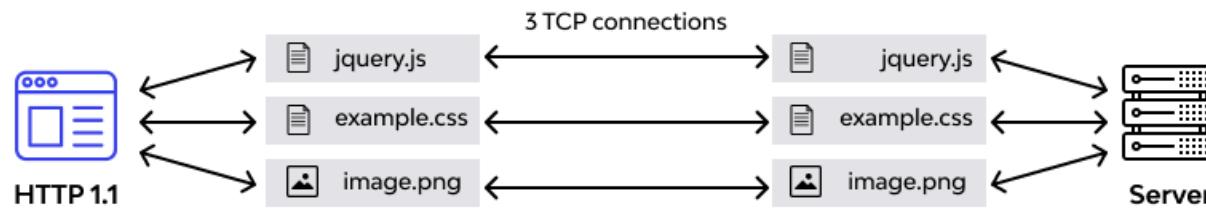
Веб-браузер делает одно TCP соединение с запросом всего содержания страницы в двоичном HTTP/2 потоке

HTTP/3

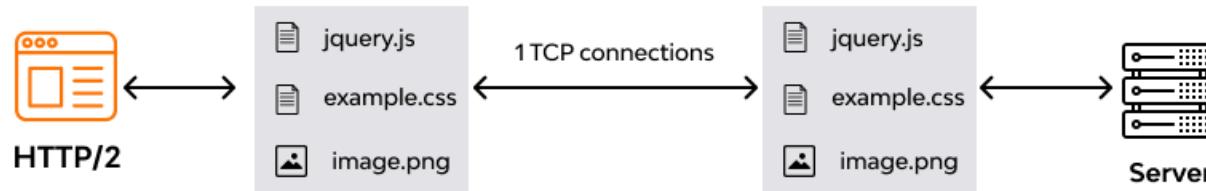
Веб-браузер делает одно QUIC соединение с запросом всего содержания страницы в двоичном QUIC потоке

* - <https://www.spirent.com/blogs/building-next-generation-web-with-http-3>

Multiplexing



HTTP 1.1

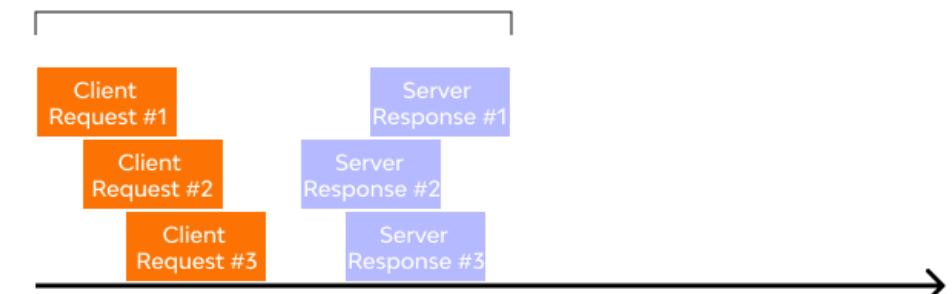


HTTP/2

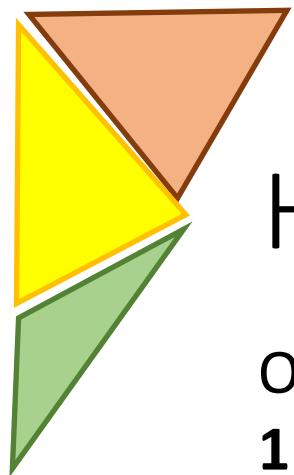
Response time



Response time



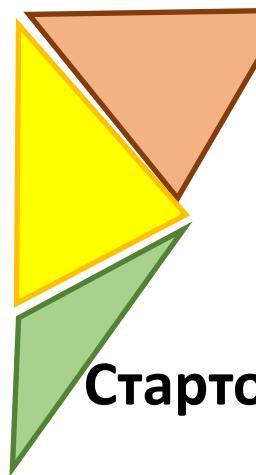
* - <https://www.wallarm.com/what/what-is-http-2-and-how-is-it-different-from-http-1>



HTTP протокол

Общая структура HTTP-сообщения:

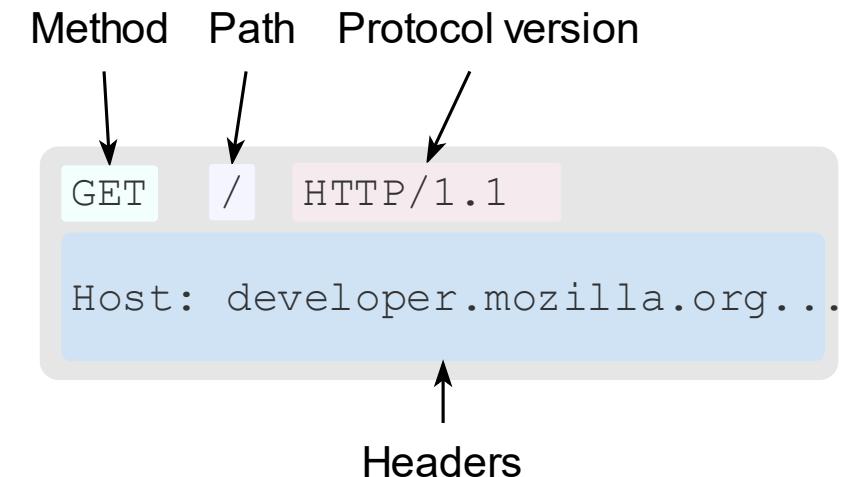
- 1. Стартовая строка (start line)** — используется для описания версии используемого протокола и другой информации (например, запрашиваемый ресурс или код ответа).
- 2. HTTP-заголовки (HTTP Headers)** — несколько строчек текста в определенном формате, которые либо уточняют запрос, либо описывают содержимое тела сообщения.
- 3. Пустая строка**, которая сообщает, что все метаданные для конкретного запроса или ответа были отправлены.
- 4. Опциональное тело сообщения**, которое содержит данные, связанные с запросом, либо документ (например HTML-страницу), передаваемый в ответе.



HTTP запрос (request)

Стартовая строка HTTP-запроса имеет следующий вид:

[Метод] [URL] [HTTP/версия]

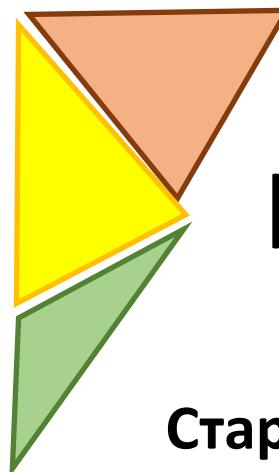


Метод указывает основную операцию над веб-ресурсом, на которую направлен запрос. Метод представляет английское слово, обычно является глаголом. Наиболее часто используемыми методами являются **GET** и **POST**

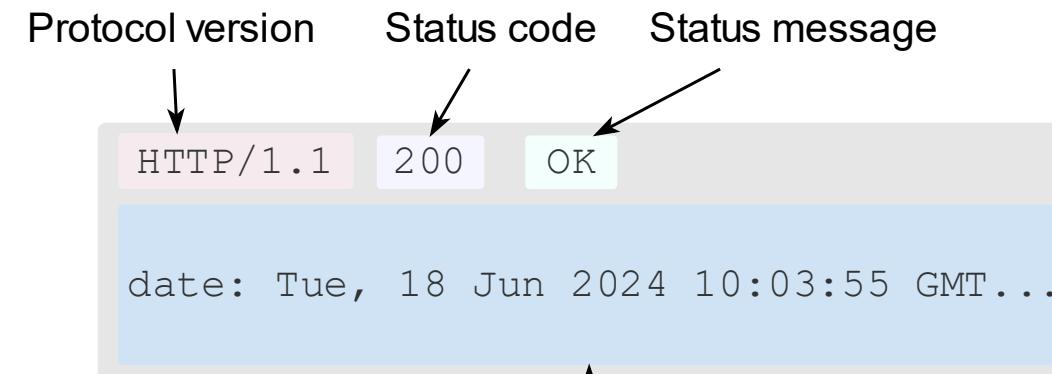
URL указывает путь до конкретного веб-ресурса. Общая структура URL состоит из **схема : иерархическая_часть [? название_параметра=значение_параметра] [фрагмент]**

HTTP-заголовок представляет собой строку формата «Имя:Значение», с двоеточием «:» в качестве разделителя. Название заголовка не учитывает регистр

Метод	Описание
GET	<p>Позволяет запросить конкретный ресурс.</p> <p>Дополнительные данные могут быть переданы через строку запроса (Query String) в составе URL (например ?param=value).</p>
POST	<p>Позволяет отправить данные на сервер.</p> <p>Поддерживает отправку различных типов файлов, среди которых текст, PDF-документы и другие типы данных в двоичном виде.</p> <p>Обычно метод POST используется при отправке информации (например, заполненной формы логина) и загрузке данных на веб-сайт, таких как изображения и документы.</p>
HEAD	<p>Обычно сервер в ответ на запрос возвращает заголовок и тело, в котором содержится запрашиваемый ресурс. Данный метод при использовании его в запросе позволит получить только заголовки, которые сервер бы вернул при получении GET-запроса к этому же ресурсу.</p> <p>Запрос с использованием данного метода обычно производится для того, чтобы узнать размер запрашиваемого ресурса перед его загрузкой.</p>
PUT	<p>Используется для создания (размещения) новых ресурсов на сервере.</p> <p>Если на сервере данный метод разрешен без надлежащего контроля, то это может привести к серьезным проблемам безопасности.</p>
DELETE	<p>Позволяет удалить существующие ресурсы на сервере.</p> <p>Если использование данного метода настроено некорректно, то это может привести к атаке типа «Отказ в обслуживании» (Denial of Service, DoS) из-за удаления критически важных файлов сервера.</p>
OPTIONS	<p>Позволяет запросить информацию о сервере, в том числе информацию о допускаемых к использованию на сервере HTTP-методов.</p>
PATCH	<p>Позволяет внести частичные изменения в указанный ресурс по указанному расположению.</p>



HTTP ответ (response)



Стартовая строка HTTP-ответа является содержит:

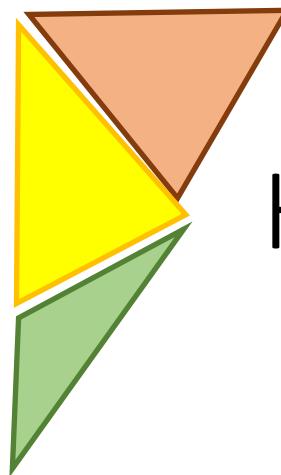
HTTP/версия Код_состояния Пояснения

Headers

Коды состояния HTTP используются для того, чтобы сообщить клиенту статус их запроса.

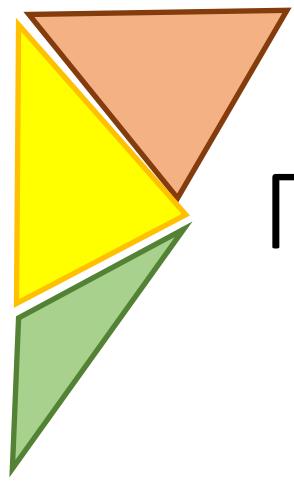
Заголовки ответа имеют аналогичный заголовкам запроса формат: «Имя» : «Значение».

Последней частью ответа является его **ТЕЛО**, которое представляет содержимое запрашиваемого ресурса (html, изображение, текст и т.д.)



Коды ошибок

Категория	Описание
1xx	Коды из данной категории носят исключительно информативный характер и никак не влияют на обработку запроса.
2xx	Коды состояния из этой категории возвращаются в случае успешной обработки клиентского запроса.
3xx	Эта категория содержит коды, которые возвращаются, если серверу нужно перенаправить клиента.
4xx	Коды данной категории означают, что на стороне клиента был отправлен некорректный запрос. Например, клиент в запросе указал не поддерживаемый метод или обратился к ресурсу, к которому у него нет доступа.
5xx	Ответ с кодами из этой категории приходит, если на стороне сервера возникла ошибка.



Пример

Заголовки запроса

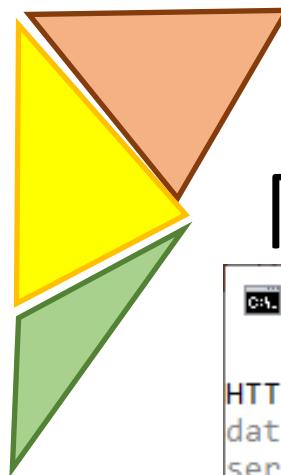
```
GET /content/universitet HTTP/1.1
Host: www.vshu.kirov.ru
User-Agent: Mozilla/5.0 (Windows NT 6.1; rv:16.0) Gecko/20100101 Firefox/16.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ru-RU,ru;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://www.vshu.kirov.ru/content/nauka-0
DNT: 1
Connection: keep-alive
```

Заголовки ответа

```
HTTP/1.1 200 OK
Via: 1.1 SERV
Connection: Keep-Alive
Proxy-Connection: Keep-Alive
Content-Length: 26993
Expires: Sun, 18 Nov 2012 18:30:19 GMT
Date: Sun, 18 Nov 2012 18:30:19 GMT
Age: 0
Content-Type: text/html; charset=utf-8
Etag: "c12ad9a4a27e882b77cdc8cf23280873"
Server: Apache/2.2.17 (Win32) PHP/5.2.17
X-Powered-By: PHP/5.2.17
Last-Modified: Sun, 18 Nov 2012 12:04:29 GMT
Cache-Control: must-revalidate
Keep-Alive: timeout=5, max=100
```

Пример

```
curl -IL https://ru.wikipedia.org
```



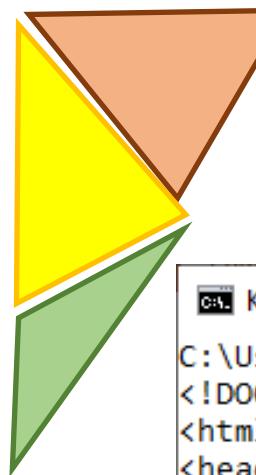
```
Командная строка

HTTP/1.1 200 OK
date: Mon, 03 Feb 2025 13:17:16 GMT
server: mw-web.eqiad.main-85fdcf9895-zdrqd
x-content-type-options: nosniff
content-language: ru
accept-ch:
vary: Accept-Encoding,Cookie,Authorization
last-modified: Mon, 03 Feb 2025 13:16:58 GMT
content-type: text/html; charset=UTF-8
age: 854
x-cache: cp3066 miss, cp3066 hit/1020
x-cache-status: hit-front
server-timing: cache;desc="hit-front", host;desc="cp3066"
strict-transport-security: max-age=106384710; includeSubDomains; preload
report-to: { "group": "wm_nel", "max_age": 604800, "endpoints": [ { "url": "https://intake-logging.wikimedia.org/v1/events?stream=w3c.reportingapi.network_error&schema_uri=/w3c/reportingapi/network_error/1.0.0" } ] }
nel: { "report_to": "wm_nel", "max_age": 604800, "failure_fraction": 0.05, "success_fraction": 0.0}
set-cookie: WMF-Last-Access=03-Feb-2025;Path=/;HttpOnly;secure;Expires=Fri, 07 Mar 2025 12:00:00 GMT
set-cookie: WMF-Last-Access-Global=03-Feb-2025;Path=/;Domain=.wikipedia.org;HttpOnly;secure;Expires=Fri, 07 Mar 2025 12:00:00 GMT
set-cookie: WMF-DP=979;Path=/;HttpOnly;secure;Expires=Tue, 04 Feb 2025 00:00:00 GMT
x-client-ip: 5.167.168.63
cache-control: private, s-maxage=0, max-age=0, must-revalidate, no-transform
set-cookie: GeoIP=RU:KIR:Kirov:58.59:49.65:v4; Path=/; secure; Domain=.wikipedia.org
set-cookie: NetworkProbeLimit=0.001;Path=/;Secure;SameSite=Lax;Max-Age=3600
accept-ranges: bytes
content-length: 165133
```

опция *--location* (или *-L*)
позволяет выполнять
перенаправления

опция *--head* (или *-I*)
позволяет получать
только заголовки

Пример curl -L https://ru.wikipedia.org

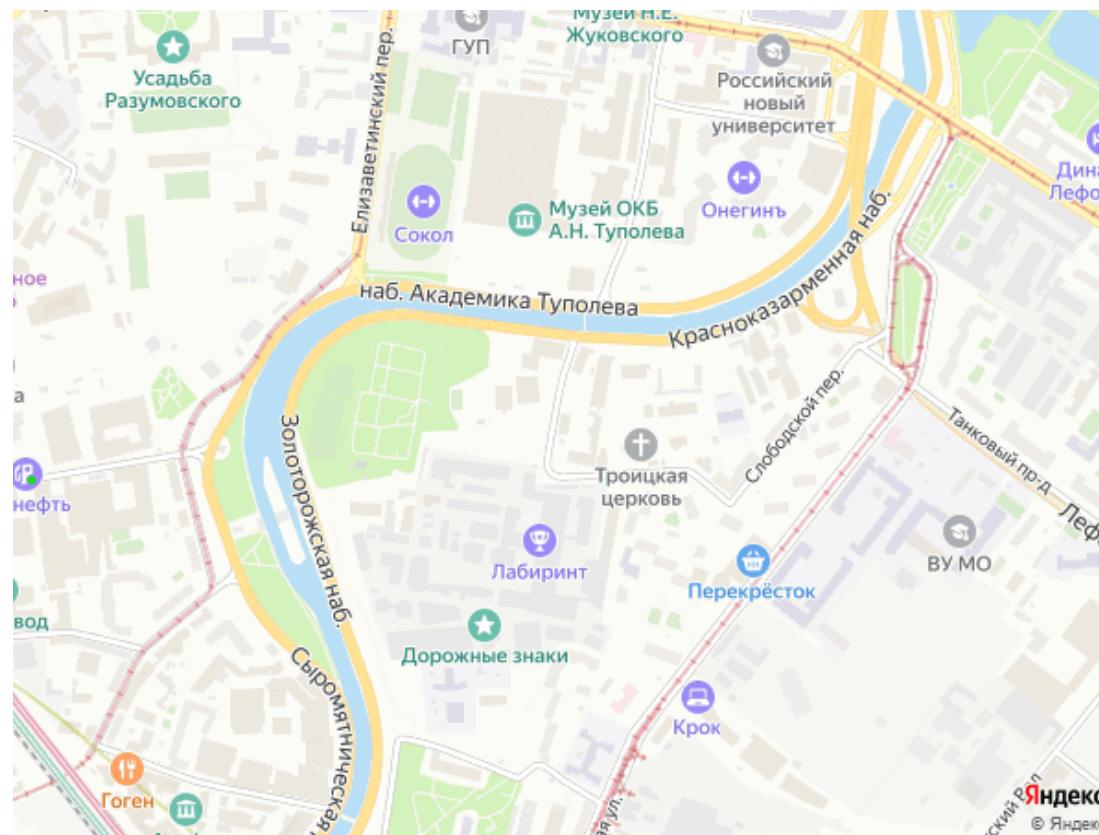


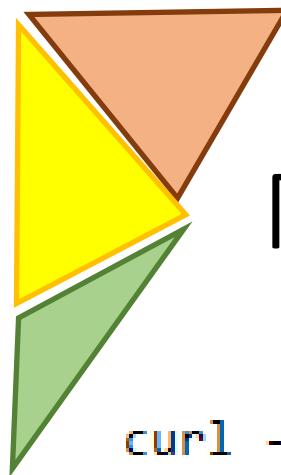
```
Командная строка
C:\Users\Alexandra>curl -L https://ru.wikipedia.org
<!DOCTYPE html>
<html class="client-nojs" lang="ru" dir="ltr">
<head>
<meta charset="UTF-8">
<title>Википедия – свободная энциклопедия</title>
<script>(function(){var className="client-js";var cookie=document.cookie.match(/(?:^| )ruwikimwclient preferences=([^\;]+);/);if(cookie){cookie[1].split('%2C').forEach(function(pref){className.replace(new RegExp('(^| )'+pref.replace(/-/g,'_')+'-clientpref-\w+$|[^\\w-]+/g,''))+'-clientpref-\\\w+(\ |$)'),'$1'+pref+$2');}}document.documentElement.className=className;})();RLCONF={"wgBreakFrames":false,"wgSeparatorTransformTable":[",\t.", "\t,"],"wgDigitTransformTable":[["",""],],"wgDateFormatFormat":"dmy","wgMonthNames":["","","январь","февраль","март","апрель","май","июнь","июль","август","сентябрь","октябрь","ноябрь","декабрь"],"wgRequestId":"2ccfa989-48d4-48d9-b06e-b0daa60a527a","wgCanonicalNamespace":"","wgCanonicalSpecialPageName":false,"wgNamespaceNumber":0,"wgPageName":"Заглавная страница","wgTitle":"Заглавная страница","wgCurRevisionId":142963243,"wgRevisionId":142963243,"wgArticleId":4401,"wgIsArticle":true,"wgIsRedirect":false,"wgAction":"view","wgUserName":null,"wgUserGroups":["*"],"wgCategories":["Википедия:Страницы с ежечасно очищаемым кэшем","Шаблоны:Подстраницы шаблонов","Шаблоны:Работа недели","Википедия:Заглавная страница"],"wgPageViewLanguage":"ru","wgPageContentLanguage":"ru","wgPageContentModel":"wikitext","wgRelevantPageName":"Заглавная страница","wgRelevantArticleId":4401,"wgIsProbablyEditable":false,"wgRelevantPageIsProbablyEditable":false,"wgRestrictionEdit":["sysop"],"wgRestrictionMove":["sysop"],"wgIs MainPage":true,"wgNoticeProject":"wikipedia","wgCiteReferencePreviewsActive":false,"wgFlaggedRevsParams":{"tags":{"accuracy":{"levels":1}}},"wgStableRevisionId":142963243,"wgMediaViewerOnClick":true,"wgMediaViewerEnabledByDefault":true,"wgPopupsFlags":0,"wgVisualEditor":{"pageLanguageCode":"ru","pageLanguageDir":"ltr","pageVariantFallbacks":"ru"},"wgMFDisplayWikibaseDescriptions":{"search":true,"watchlist":true,"tagline":false,"nearby":true}, "wgWMESchemaEditAttemptStepOversample":false,"wgWMEPageLength":8000,"wgEditSubmitButtonLabelPublish":true,"wgULSPosition":"interlanguage","wgULSIsCompactLinksEnabled":true,"wgVector2022LanguageInHeader":false,"wgULSIsLanguageSelectorEmpty":false,"wgWikibaseItemId":"Q5296","wgCheckUserClientHintsHeadersJsApi":["brands","architecture","bitness","fullVersionList","mobile","model","pla
```

Пример (тестировать API сервиса)

```
curl -X GET "https://static-maps.yandex.ru/1.x/?ll=37.677751,55.757718&spn=0.016457,0.00619&l=map" --output 1.png
```

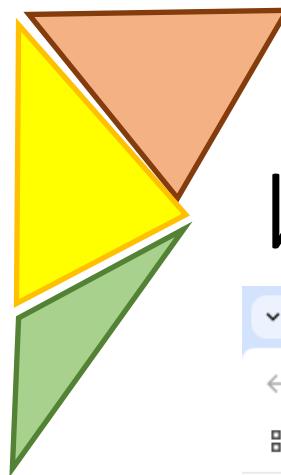
% Total	% Received	% Xferd	Average Speed	Time	Time	Time	Current
Dload	Upload	Total	Spent	Left	Speed		
100	68786	100	68786	0	0	120k	0
				--:--:--	--:--:--	--:--:--	121k



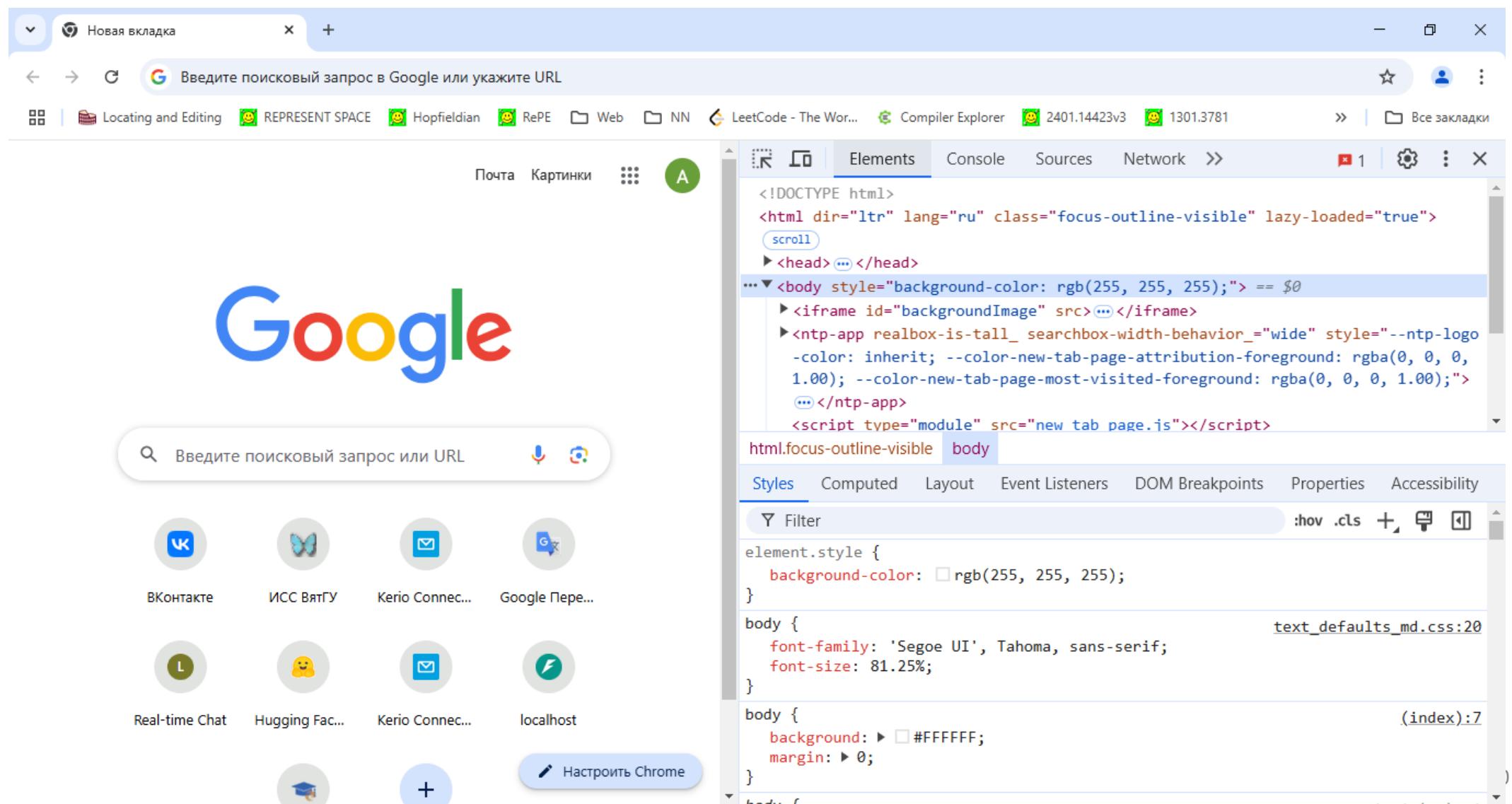


Пример (скачать файл)

```
curl -O https://www.vyatsu.ru/reports/schedule/Group/24053_2_03022025_16022025.pdf
% Total    % Received % Xferd  Average Speed   Time     Time     Time  Current
                                         Dload  Upload   Total   Spent    Left  Speed
100  402k  100  402k    0      0  223k      0  0:00:01  0:00:01  --:--:--  224k
```



Инструмент разработчика в браузере



Новая вкладка

Введите поисковый запрос в Google или укажите URL

Locating and Editing REPRESENT SPACE Hopfieldian RePE Web NN LeetCode - The Wor... Compiler Explorer 2401.14423v3 1301.3781 Все закладки

Почта Картинки A

Google

Введите поисковый запрос или URL

ВКонтакте ИСС ВятГУ Kerio Connec... Google Пере...

Real-time Chat Hugging Fac... Kerio Connec... localhost

Настройте Chrome

Elements Console Sources Network

```
<!DOCTYPE html>
<html dir="ltr" lang="ru" class="focus-outline-visible lazy-loaded="true">
  scroll
  > <head> ... </head>
... <body style="background-color: rgb(255, 255, 255);"> == $0
  > <iframe id="backgroundImage" src>...</iframe>
  > <ntp-app realbox-is-tall_ searchbox-width-behavior_=wide" style="--ntp-logo-color: inherit; --color-new-tab-page-attribution-foreground: rgba(0, 0, 0, 1.00); --color-new-tab-page-most-visited-foreground: rgba(0, 0, 0, 1.00);">
    ...
  </ntp-app>
  <script type="module" src="new tab page.js"></script>

```

html.focus-outline-visible body

Styles Computed Layout Event Listeners DOM Breakpoints Properties Accessibility

Filter :hover .cls +, □, □

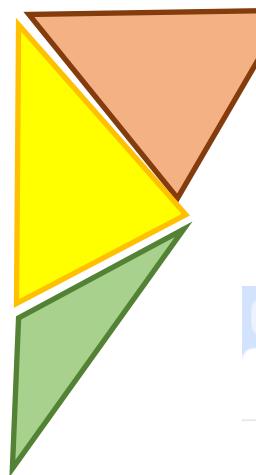
```
element.style {
  background-color: □rgb(255, 255, 255);
}

body {
  font-family: 'Segoe UI', Tahoma, sans-serif;
  font-size: 81.25%;
}

body {
  background: □ #FFFFFF;
  margin: □ 0;
}
```

text_defaults_md.css:20 (index):7

Инструмент разработчика в браузере



A screenshot of a web browser window showing the search results for "kirov" on Google. The browser's developer tools are open, specifically the Network tab, which displays a timeline and a list of network requests.

The Network tab shows the following details:

- Timeline:** A horizontal bar at the top indicates request times from 1,000 ms to 7,000 ms.
- Filter:** Options include All, Fetch/XHR, Doc, CSS, JS, Font, Img, Media, Manifest, WS, Wasm, and Other.
- Table:** A list of network requests with columns for Name, Status, Protocol, Type, Initiator, Size, and Time.

Name	Status	Protocol	Type	Initiator	Size	Time
search?q=kirov&oq=ki... (pending)	200	h3	document	Other	306 kB	559 ms
4UaRrENHsxJIGDuGo1OIJfC... (disk cache)	200	h3	font	/search?q=kirov&	(disk cache)	25 ms
tia.png	200	h3	png	/search?q=kirov&	(disk cache)	38 ms
AF2bZyizNMSuKjmieTAD4Krl... (disk cache)	302	h3	text/html	/search?q=kirov&	(disk cache)	136 ms
desktop_searchbox_sprites3... (disk cache)	200	h3	webp	/search?q=kirov&	(disk cache)	37 ms
4UaRrENHsxJIGDuGo1OIJfC... (disk cache)	200	h3	font	/search?q=kirov&	(disk cache)	3 ms
ALs6j_FeWY-gNhk4cV0B5C1... (disk cache)	200	h3	png	AF2bZyizNMSuKj...	454 B	103 ms
16521530460/?gad_source=... (pending)	(pending)			search?q=kirov&	0 B	Pending
16521530460/?gad_source=... (memoized)	204	h3	text/css	search?q=kirov&	24 B	70 ms
data:image/gif;base64... (memoized)	200	data	gif	/search?q=kirov&	(memoized)	0 ms
data:image/jpeg;bas... (memoized)	200	data	jpeg	search?q=kirov&	(memoized)	2 ms
data:image/png;base64... (memoized)	200	data	png	search?q=kirov&	(memoized)	0 ms

At the bottom of the Network tab, summary statistics are displayed: 140 requests, 510 kB transferred, 12.0 MB resources, Finish: 6.82 s, and DOMContentLoaded: 1.13 s.

Пример с http2

The screenshot shows a web browser window displaying the Wikipedia page for 'Киров'. The browser's address bar shows the URL ru.wikipedia.org/wiki/Киров. The developer tools are open, specifically the Network tab, which lists network requests. One request in the table is highlighted with an orange border around the 'Protocol' column, which contains the value 'h2'. This indicates that the page was loaded using the HTTP/2 protocol.

Name	Status	Protocol	Type	Initiator	Size	Time
%D0%9A%D0%B8%D1%80...	200	h2	docum...	Other	128 kB	357 ms

Пример с http2

The screenshot shows a web browser window displaying the Wikipedia page for Kirov. The browser's address bar shows the URL ru.wikipedia.org/wiki/Kirov. The developer tools are open, specifically the Network tab, which provides a detailed timeline of the HTTP2 requests made during the page load.

Page Content Summary:

- Page Title:** Киров
- Coordinates:** 58°36' с. ш. 49°39' в. д.
- Text Summary:** Материал из Википедии — свободной энциклопедии
- Image:** A large image of a colorful church building in Kirov.
- Table of Contents:** Includes sections for Киров (до 5 декабря 1934 года), Вятка, Хлынов, and Kirov city.
- Side Panel:** Includes links like Участие, Сообщить об ошибке, Как править статьи, Сообщество, Форум, Свежие правки, Новые страницы, Справка, Инструменты, Ссылки сюда, Связанные правки, Служебные страницы, Постоянная ссылка, Сведения о странице, and Имитировать страницу.

Network Tab Timeline:

- Started at 1.55 ms:** Resource Scheduling
- 1.55 ms:** Queueing
- 116.04 ms:** Connection Start (Stalled)
- 0.29 ms:** Request sent
- 109.57 ms:** Waiting for server response
- 130.63 ms:** Content Download

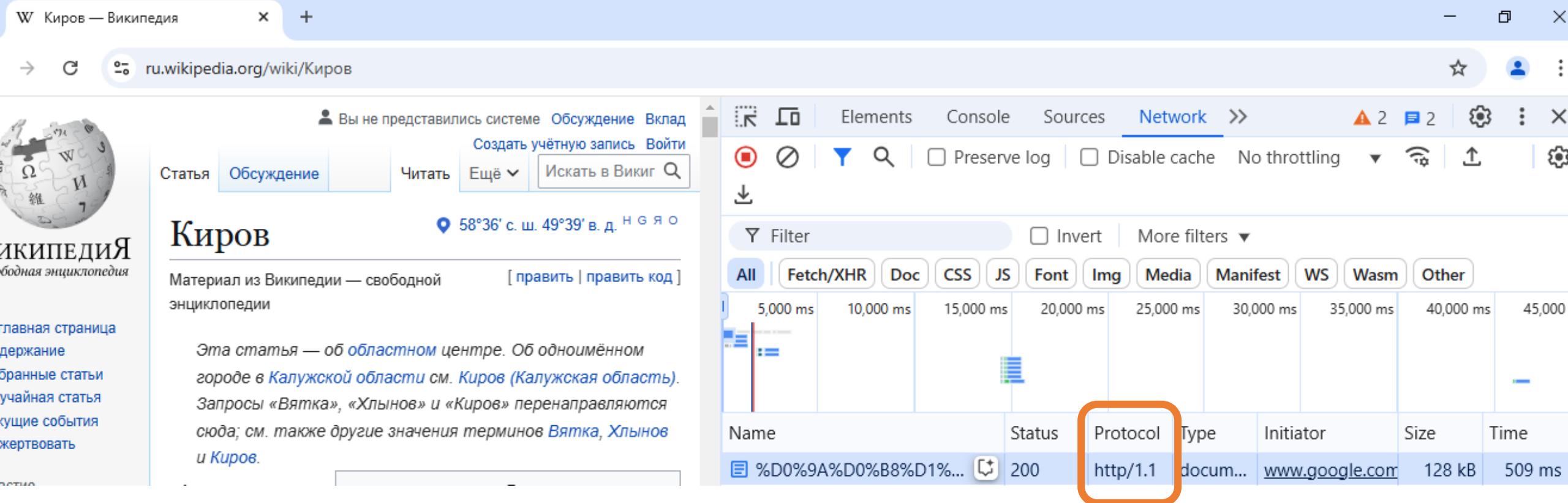
Request Headers:

- %D0%9A%D0%B...
- load.php?lang=ru...
- load.php?lang=ru...
- load.php?lang=ru...
- load.php?lang=ru...
- 305px-%D0%92%...
- 151px-Da_svet_p...
- 151px-%D0%A1%...
- 151px-%D0%A2%...
- 151px-%D0%90%...
- 305px-%D0%97%...
- 160px-Flag_of_Kir...

Timing Metrics:

- DURATION: 1.55 ms
- DURATION: 116.04 ms
- DURATION: 0.29 ms
- DURATION: 109.57 ms
- DURATION: 130.63 ms

Пример с http 1.1



The screenshot shows a web browser window displaying the Wikipedia page for 'Киров'. The browser's address bar shows the URL ru.wikipedia.org/wiki/Киров. The developer tools Network tab is open, showing a list of network requests. One specific request to www.google.com is highlighted with an orange border around the 'Protocol' column, which displays 'http/1.1'. The Network tab also includes various filters and performance metrics like time and size.

Name	Status	Protocol	Type	Initiator	Size	Time
%D0%9A%D0%B8%D1%	200	http/1.1	document	www.google.com	128 kB	509 ms

Пример с http 1.1

The screenshot shows a web browser window displaying the Russian Wikipedia page for the city of Kirov. The page title is "Киров" (Kirov). The network tab of the developer tools is active, showing the timeline of requests and their durations.

Network Tab Headers:

- Preserve log
- Disable cache
- No throttling
- Filter: All, Fetch/XHR, Doc, CSS, JS, Font, Img, Media, Manifest, WS, Wasm, Other
- Timing: DURATION, Queueing, Connection Start, Request/Response, Content Download

Resource Scheduling:

- Queueing: 2.26 ms
- Connection Start: 21.24 ms
- Request/Response:
 - Request sent: 0.41 ms
 - Waiting for server response: 73.48 ms
 - Content Download: 413.64 ms

Resource List:

Name	DURATION
%D0%9A%D0%B...	
load.php?lang=ru...	2.26 ms
load.php?lang=ru...	
load.php?lang=ru...	
load.php?lang=ru...	
305px-%D0%92%...	
151px-Da_svet_p...	
151px-%D0%A1%...	
151px-%D0%A2%...	
151px-%D0%90%...	
305px-%D0%97%...	
160px-Flag_of_Kir...	

Page Content Summary:

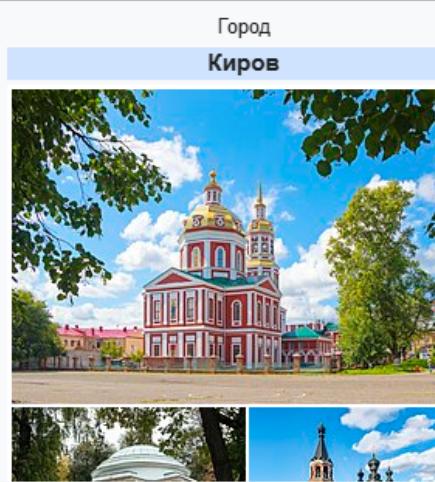
- Статья: Киров
- Обсуждение: Обсуждение
- Читать: Читать
- Ещё: Ещё
- Искать в Викиг: Искать в Викиг

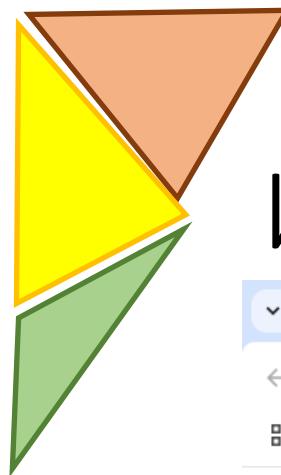
Page Content:

Материал из Википедии — свободной энциклопедии [править | править код]

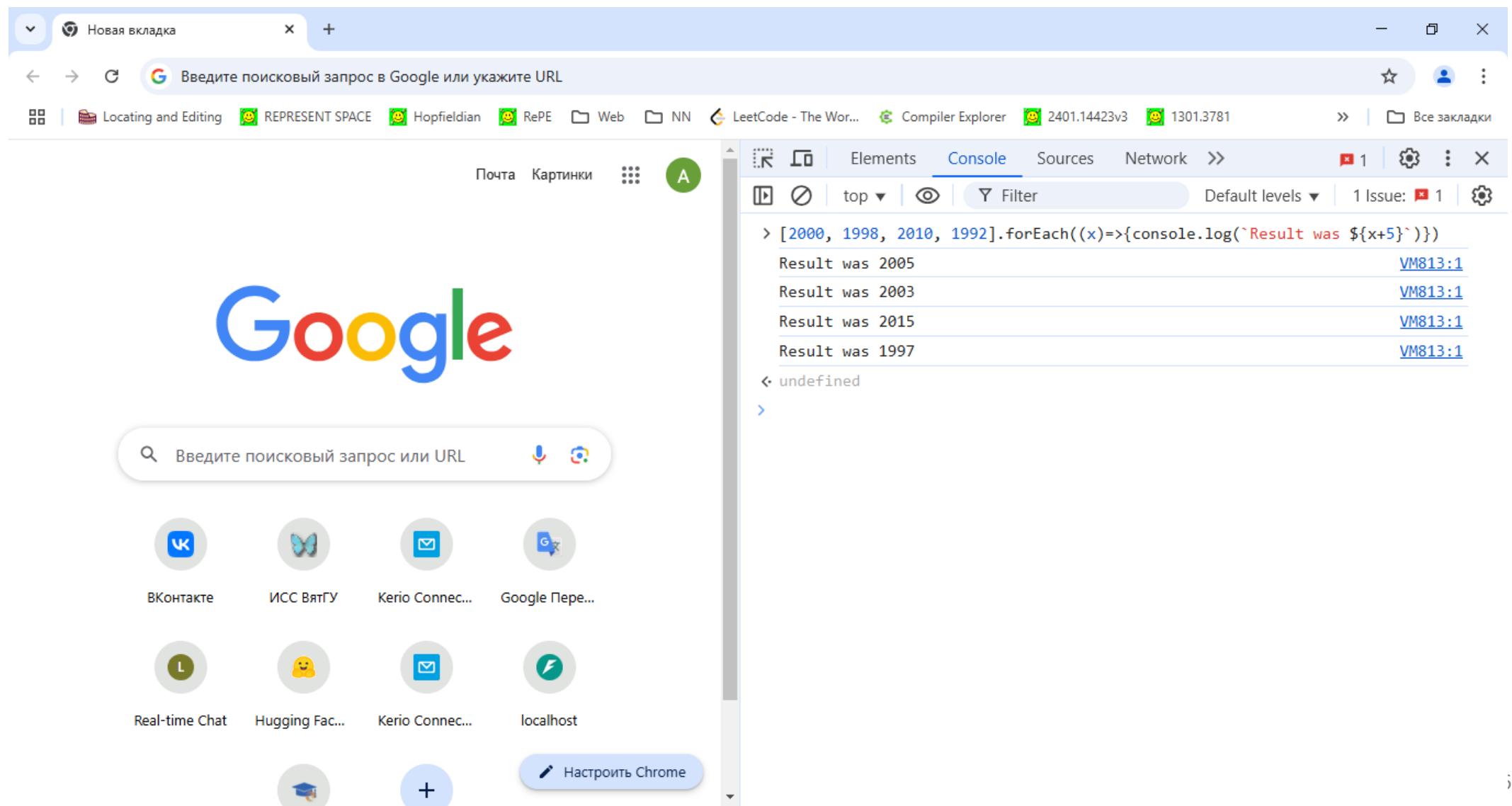
Эта статья — об **областном** центре. Об одноимённом городе в **Калужской области** см. **Киров (Калужская область)**. Запросы «**Вятка**», «**Хлынов**» и «**Киров**» перенаправляются сюда; см. также другие значения терминов **Вятка**, **Хлынов** и **Киров**.

Киров (до 5 декабря 1934 года — **Вятка**, с 1457 по 1780 год также ^[8] **Хлынов**) — город в **России**, административный центр **Кировской области**. Вместе с подчинёнными населёнными пунктами образует **одноимённый**





Инструмент разработчика в браузере



Новая вкладка

Ведите поисковый запрос в Google или укажите URL

Locating and Editing REPRENT SPACE Hopfieldian RePE Web NN LeetCode - The Wor... Compiler Explorer 2401.14423v3 1301.3781 Все закладки

Почта Картинки A

Console

Elements Sources Network

top Filter Default levels 1 Issue: 1

```
> [2000, 1998, 2010, 1992].forEach((x)=>{console.log(`Result was ${x+5}`)})  
Result was 2005 VM813:1  
Result was 2003 VM813:1  
Result was 2015 VM813:1  
Result was 1997 VM813:1  
< undefined  
>
```

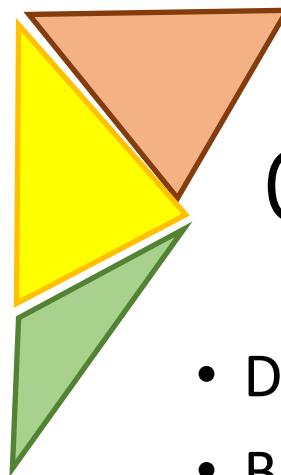
Ведите поисковый запрос или URL

VK ИСС ВятГУ Kerio Connec... Google Пере...

ВКонтакте ИСС ВятГУ Kerio Connec... Google Пере...

Real-time Chat Hugging Fac... Kerio Connec... localhost

Настройте Chrome



Объектная модель документа (DOM)

- DOM была придумана создателями браузеров
- В браузере есть движок рендеринга (отображения веб-страниц)
Например, WebKit (Safari), Blink (Chrome), Gecko (Firefox), и Trident/EdgeHTML (IE/Edge)
- Когда пользователь запрашивает определенную веб-страницу, то механизм рендеринга начинает получать HTML, CSS и JS-файлы запрошенной веб-страницы через сетевой уровень
- После получения содержимого страницы, HTML и CSS-файлы анализируются по частям и узлы DOM создаются для формирования **дерева DOM (document object model)** и **дерева CSSOM (CSS object model)**
- Используя DOM и CSSOM строится **дерево рендеринга**, на основе которого затем формируется макет страницы, производится послойная отрисовка элементов и сведение слоёв. В результате получается веб-страница, которую пользователь видит на экране

Пример DOM

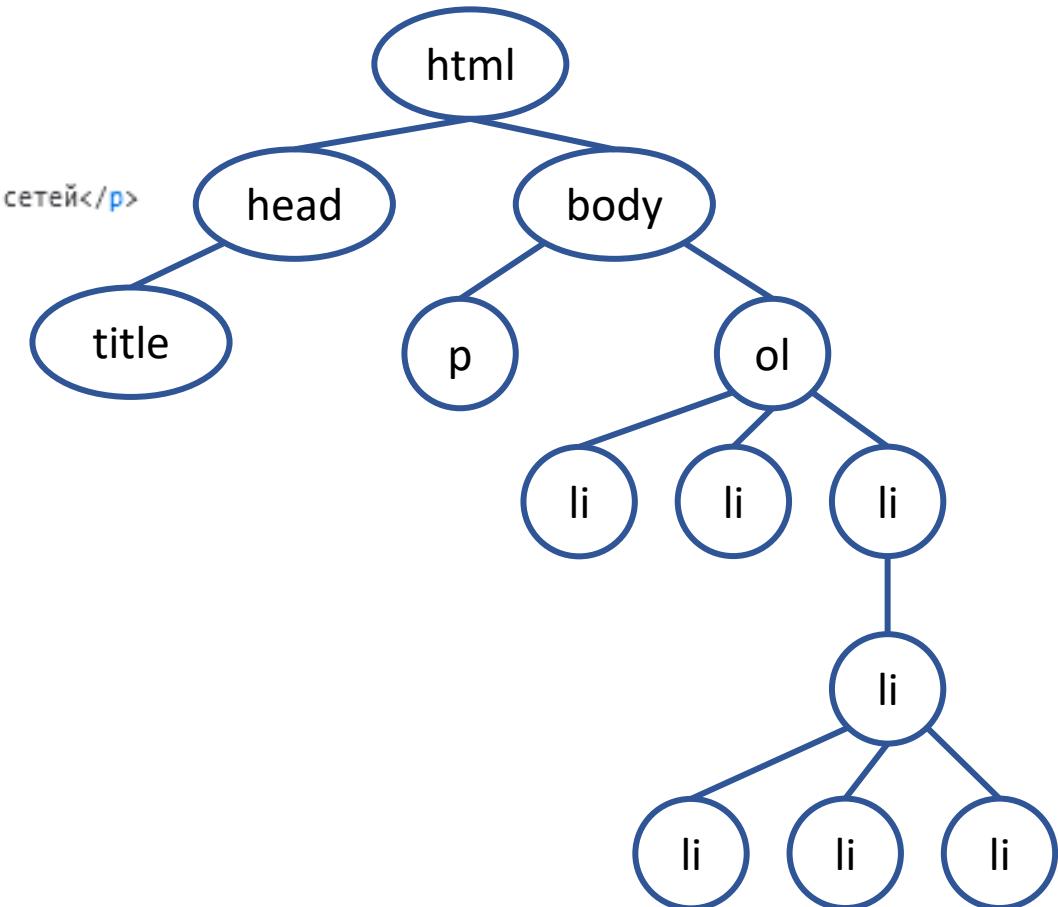
Слои нейронной сети и примеры нейронных сетей

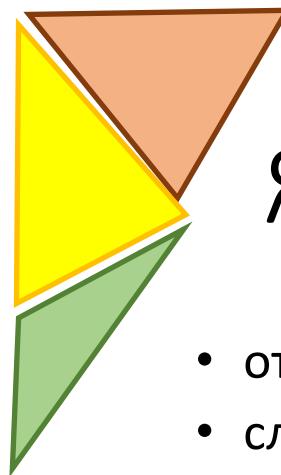
1. Свёрточные
2. Рекуррентные
3. Трансформерные
 - a. BERT
 - b. GPT
 - c. T5

Инспектор Консоль Отладчик Сеть Стили >

Поиск в HTML

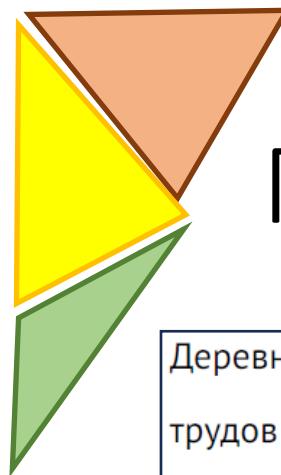
```
<!DOCTYPE html>
<html lang="ru">
  <head>
    <title>Test page</title>
  </head>
  <body>
    <p>Слои нейронной сети и примеры нейронных сетей</p>
    <ol>
      <li>...</li>
      <li>...</li>
      <li>
        ::marker
        Трансформерные
        <ol type="a">
          <li>...</li>
          <li>...</li>
          <li>
            ::marker
            T5
          </li>
        </ol>
      </li>
    </ol>
  </body>
</html>
```





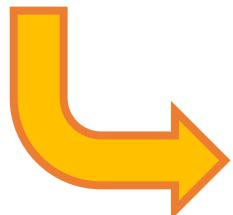
Язык HTML

- от англ. *HyperText Markup Language* – «язык разметки гипертекста»
- служит для создания веб-страниц
- HTML-документы представляют собой файлы с текстом и дополнительными инструкциями (**тегами**)
- Официальной спецификации HTML 1.0 не существует. До 1995 года существовало множество неофициальных спецификаций HTML, появившихся в ходе браузерных войн
- RFC 1866 — HTML 2.0, одобренный как официальный стандарт 22 сентября 1995 года
- HTML 3 (март 1996) - не нашла поддержки у разработчиков
- HTML 3.2 — 14 января 1997 года
- HTML 4.0 — 18 декабря 1997 года (многие унаследованные элементы были отмечены как устаревшие и нерекомендованные к использованию)
- HTML 4.01 — 24 декабря 1999 года (версия включала малозаметные, но существенные изменения по сравнению с предыдущей)
- ISO/IEC 15445:2000 (ISO HTML, основан на HTML 4.01 Strict) — 15 мая 2000 года
- HTML 5 — разработан и принят W3C в 2014



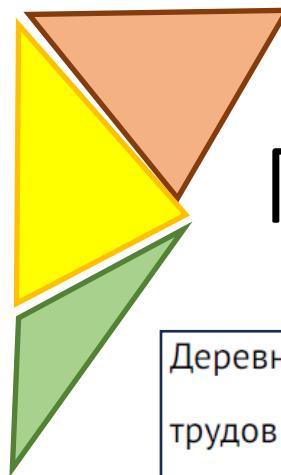
Пример

Деревня Приветствуя тебя, пустынnyй уголок, приют спокойствия, трудов и вдохновенья, где льется дней моих невидимый поток на лоне счастья и забвенья. Я твой – я променял порочный двор Цирцей, роскошные пиры, забавы, заблужденья на мирный шум дубров, на тишину полей, на праздность вольную, подругу размышленья.



Деревня
Приветствуя тебя, пустынnyй уголок,
Приют спокойствия, трудов и вдохновенья,
Где льется дней моих невидимый поток
На лоне счастья и забвенья.

Я твой – я променял порочный двор Цирцей,
Носкошные пиры, забавы, заблужденья
На мирный шум дубров, на тишину полей,
На праздность вольную, подругу размышленья.



Пример

Деревня Приветствуя тебя, пустынnyй уголок, приют спокойствия, трудов и вдохновенья, где льется дней моих невидимый поток на лоне счастья и забвенья. Я твой – я променял порочный двор Цирцей, роскошные пиры, забавы, заблужденья на мирный шум дубров, на тишину полей, на праздность вольную, подругу размышленья.

Деревня

Приветствуя тебя, пустынnyй уголок,
Приют спокойствия, трудов и вдохновенья,
Где льется дней моих невидимый поток
На лоне счастья и забвенья.

Я твой – я променял порочный двор Цирцей,
Носкошные пиры, забавы, заблужденья
На мирный шум дубров, на тишину полей,
На праздность вольную, подругу размышленья.

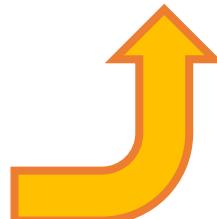
[начало заголовка] Деревня [конец заголовка]

[начало абзаца] Приветствуя тебя, пустынnyй уголок, [перенос]

Приют спокойствия, трудов и вдохновенья, [перенос]

Где льется дней моих невидимый поток [перенос]

На лоне счастья и забвенья. [конец абзаца]

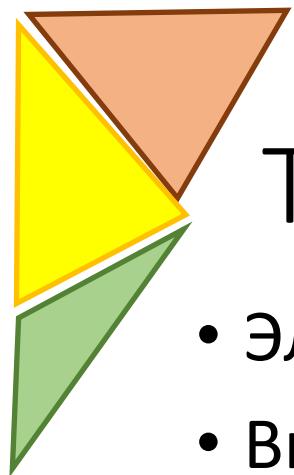


[начало абзаца] Я твой – я променял порочный двор Цирцей, [перенос]

Носкошные пиры, забавы, заблужденья [перенос]

На мирный шум дубров, на тишину полей, [перенос]

На праздность вольную, подругу размышленья. [конец абзаца]

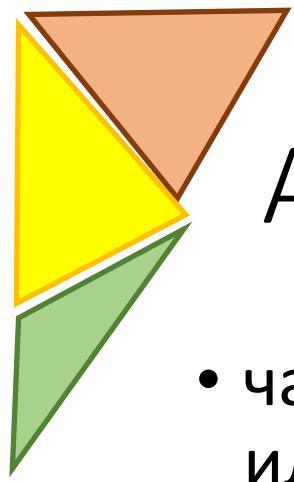


Тег

- Элемент языка разметки, заключённый между символами < и >
- Виды:
 - Двойные. Например: <html></html>
 - Одиночные. Например:
- Регистронезависимы
, , , - один и тот же тег
- Необходимо соблюдать правильную вложенность тегов

01. <i>очень длинный текст очень </i> – Правильно

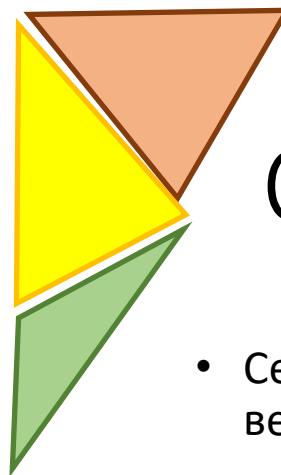
02. <i>очень длинный текст </i> очень – Неправильно



Атрибуты тегов

- часть языка разметки, используемая для настройки поведения или отображения элемента HTML

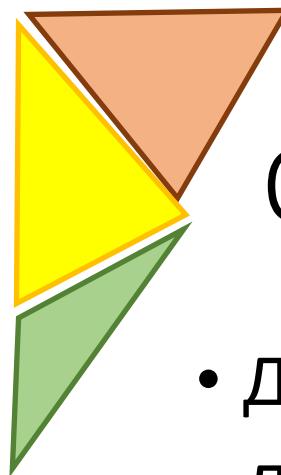
01. `<div class="he'a'der">` – в двойных кавычках
02. `<input checked='checked'>` – в апострофах
03. `<div class=footers>` – без кавычек
04. `<input checked>` – одиночные атрибуты



Структура HTML документа

- Секция DOCTYPE указывает браузеру тип документа и версию использованного языка разметки
- Мета-тег HTML — это элемент разметки html, описывающий свойства документа как такового (метаданные). Назначение мета-тега определяется набором его атрибутов, которые задаются в теге <meta>. Мета-тег размещают в блоке <head>...</head> веб-страницы. Они не являются обязательными элементами, но могут быть весьма полезны
- В блоке <head>...</head> также происходит подключение скриптов, указание заголовка страницы
- Блок <body>...</body> представляет содержимое веб-страницы

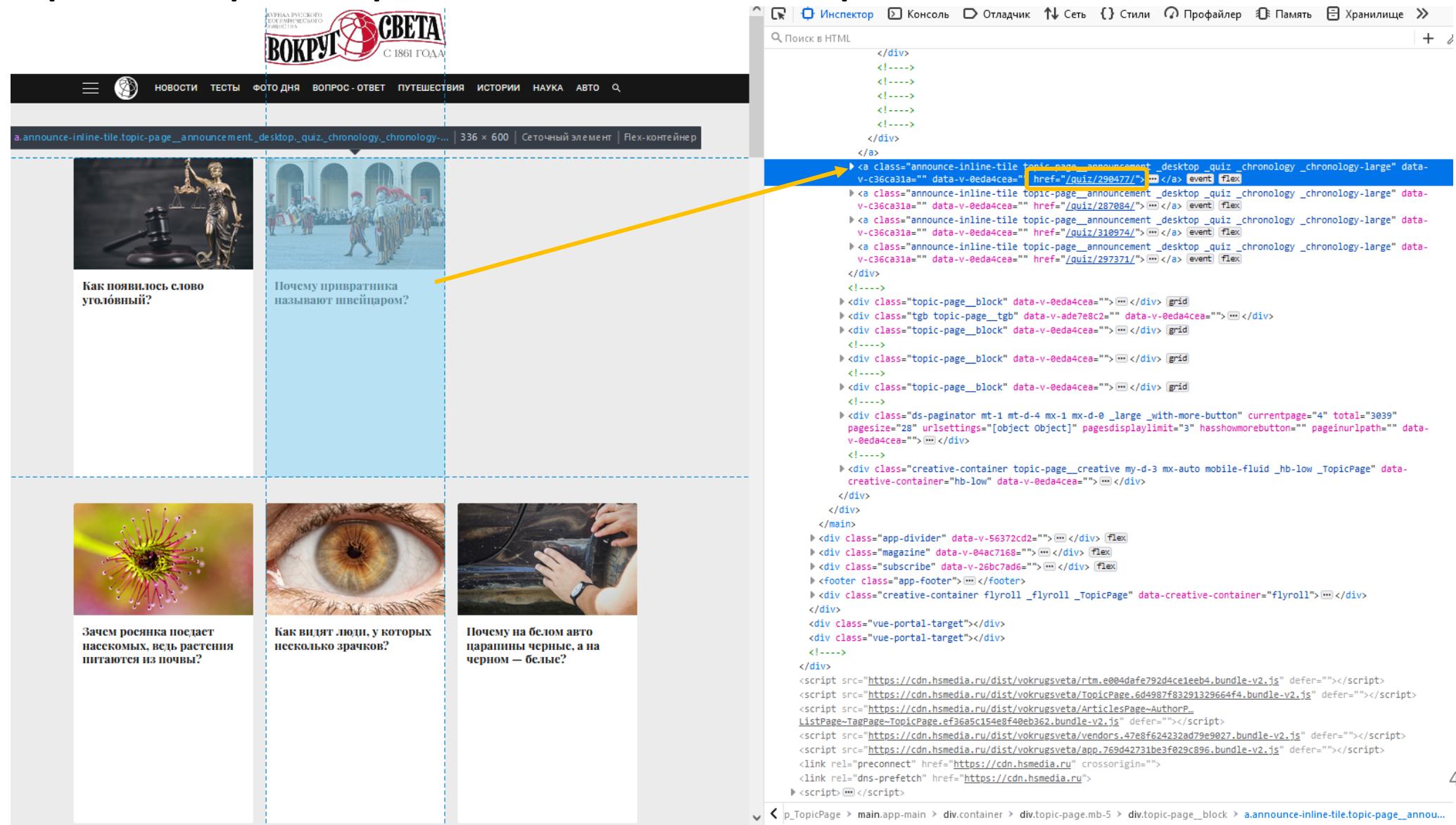
```
<!DOCTYPE html>
<html lang="ru">
  <head>
    <!-- Служебная информация -->
  </head>
  <body>
    <!-- Содержание -->
  </body>
</html>
```



Основные виды тегов

- Для работы с текстом
- Для работы с изображениями
- Для работы со списками
- Для работы со ссылками
- Для работы с таблицами

Пример парсинга веб-страниц



The screenshot shows a web browser displaying the Vokrug Sveta website. The page features a header with the logo 'ВОКРУГ СВЕТА' and the text 'С 1861 ГОДА'. Below the header is a navigation bar with links: НОВОСТИ, ТЕСТЫ, ФОТО ДНЯ, ВОПРОС - ОТВЕТ, ПУТЕШЕСТВИЯ, ИСТОРИИ, НАУКА, АВТО, and a search icon.

The main content area displays a grid of six articles:

- Top Left:** An image of scales of justice and a gavel, with the question "Как появилось слово утоливший?"
- Top Middle:** An image of a guard in historical attire, with the question "Почему привратника называют пивейцаром?"
- Bottom Left:** An image of a carnivorous plant, with the question "Зачем росинка поедает насекомых, ведь растения питаются из почвы?"
- Bottom Middle:** An image of a human eye, with the question "Как видят люди, у которых несколько зрачков?"
- Bottom Right:** An image of a person's hand touching a car tire, with the question "Почему на белом авто царапины черные, а на черном — белые?"

A yellow arrow points from the question in the bottom-right article to the corresponding line of code in the developer tools console, which highlights the href attribute of the anchor tag.

```
<div>
<!-->
<!-->
<!-->
<!-->
<!-->
<!-->
</div>
<a class="announce-inline-tile topic-page__announcement_desktop_quiz_chronology_chronology-large" data-v-c36ca31a="" data-v-0eda4cea="" href="/quiz/290477/">............

46


```

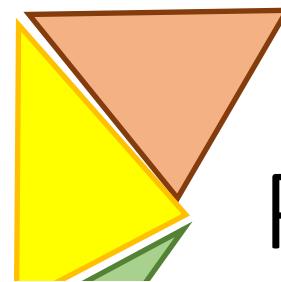
Пример парсинга веб-страниц

The screenshot shows a news article from the website 'Вокруг Света' (Vokrug Sveta). The article title is 'Зачем росянка поедает насекомых, ведь растения питаются из почвы?' (Why does a sundew eat insects, since plants grow from the soil?). Below the title is a large image of a sundew plant with its characteristic sticky traps. The article content discusses the plant's diet and its unique ability to grow in water-saturated peat bogs due to a lack of mineral nutrients in the water. At the bottom of the page is a feedback section with a button labeled 'ЗАДАТЬ СВОЙ ВОПРОС' (Ask your question).

The screenshot shows the browser developer tools' 'Инспектор' (Inspector) tab with the 'Поиск в HTML' (Search in HTML) feature active. A search term has been entered into the search bar. The results are displayed as a tree structure of HTML elements. Several specific text blocks from the article are highlighted with blue boxes and connected by yellow arrows to their corresponding occurrences in the HTML code. These highlighted sections include:

- 'Не хватает обычного питания' (Not enough ordinary nutrition)
- 'Роснянка растет на верховых болотах. Влага попадает туда в основном благодаря дождям и содержит мало минеральных веществ. Из грунта питание к растению не поступает, поскольку вода в таких болотах почти не перемешивается.' (The sundew grows on upland bogs. Water gets there mainly through rain and contains few mineral substances. It does not receive nutrition from the soil because water in such bogs hardly mixes.)
- 'Между тем для синтеза белков необходим азот, а для нуклеиновых кислот – также и фосфор. Еще ряд элементов, особенно калий, нужен для нормальной жизнедеятельности клеток. Все эти вещества росянка получает из насекомых.' (Meanwhile, for protein synthesis, nitrogen is required, and for nucleic acids, also phosphorus. A series of other elements, especially potassium, are needed for normal cellular activity. All these substances the sundew gets from insects.)

```
1 quiz_href_list = list()
2 for i in tqdm(range(1, 109)):
3     URL_TEMPLATE = f"https://www.vokrugsveta.ru/quiz/page-{i}/"
4     r = requests.get(URL_TEMPLATE)
5     if r.status_code != 200:
6         print(f"{URL_TEMPLATE} - {r.status_code}")
7
8     soup = bs(r.text, "html.parser")
9     quiz_page_href_list =[x["href"] for x in soup.find_all('a', class_="announce-inline-tile
topic-page_announcement_desktop_quiz_chronology_chronology-large")]
10    if quiz_page_href_list:
11        quiz_href_list.extend(quiz_page_href_list)
12    else:
13        print(f"{URL_TEMPLATE} - empty href list")
14
15 qa_knw_list = list()
16 i = 0
17 for x in tqdm(quiz_href_list):
18     i = i + 1
19     URL = ROOT_URL+x
20     r = requests.get(URL, timeout=10)
21     if r.status_code != 200:
22         print(f"Error access for {URL}")
23         continue
24     html_text = r.text
25     y = get_data_common(html_text)
26     if y:
27         y["source"] = URL
28         qa_knw_list.append(y)
29     else:
30         print(f"Processing error of {URL}")
31     if i % 100 == 0:
32         save_data(qa_knw_list)
33 save_data(qa_knw_list)
```



Результат

```
[{"question": "В какую сторону крутится пропеллер у самолета?",\n"answer": "Бывает по-разному",\n"knowledge": "На одномоторных самолетах винт, как правило, вращается по часовой стрелке, если смотреть спереди. Вероятно, эта традиция сохранилась с тех времен, когда двигатель запускали, раскручивая винт вручную: технику-правше удобнее браться за лопасть справа и тянуть ее вниз. На небольших двухмоторных самолетах обычно сохраняется это направление. Если бы винты вращались в противоположные стороны, их крутящие моменты взаимно компенсировали бы, что упрощало бы пилотирование. Однако на практике важнее, чтобы детали двух двигателей были взаимозаменяемы. И все же на тяжелых четырехмоторных транспортных самолетах (Airbus A400M, Hercules C-130) винты крутятся в противоположных направлениях. А на самых мощных турбовинтовых машинах (Ан-22, Ту-95) на каждый двигатель ставят сдвоенные винты, которые вращаются навстречу друг другу.",\n"source": "https://www.vokrugsveta.ru/quiz/278413/\\"},\n...\n{\n"question": "Почему батарейки в фонарике разряжаются сами собой?",\n"answer": "Из-за утечек",\n"knowledge": "От контактов батарейки к выключателю фонарика идут проводники. Даже если цепь разомкнута, расстояние между контактами выключателя составляет около миллиметра. Через такой разрыв может протекать слабый ток, который разряжает батарейки. Чтобы этого избежать, надо изолировать батарейку от одного из контактов парой слоев полизтилена.",\n"source": "https://www.vokrugsveta.ru/quiz/243740/\\"\n]
```