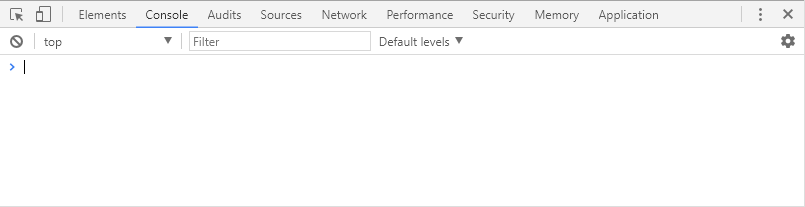
DevTools В БРАУЗЕРЕ CHROME

В каждом браузере есть свой **инструмент разработчика** – это то, что многие называют просто «консолью», рассмотрим Консоль в браузере Chrome. Выглядит вот так:



Консоль в браузере Chrome

На самом деле Консоль в браузере Chrome — это инструмент, с помощью которого можно не только посмотреть наполнение страницы, выводимой браузером, а также существующие ошибки (что чаще всего и делают тестировщики), но и исправить эти ошибки намного проще и быстрее (что чаще всего делают девелоперы), замерять различные показатели и манипулировать страницей.

**Как открыть консоль в браузере Chrome:**

— клавиша F12;

— нажав одновременно клавиши Ctrl + Shift + I;

— ПКМ по элементу страницы –> Просмотреть код;

— меню браузера –> Дополнительные Инструменты –> Инструменты Разработчика.

Располагаться она может внизу страницы или сбоку, можно открепить в отдельное окно. Итак, рассмотрим по отдельности каждую вкладку открытой консоли.

Итого в ней есть 8 вкладок, каждая из которых отображает определенные данные:

1 – *Elements* (содержит весь html/css код страницы и позволяет выбрать элементы для исследования, а также редактировать их)

2 – *Console* (отображает наличие/отсутствие ошибок/предупреждений в коде)

3 – *Sources*(позволяет выполнять операции с кодом страницы)

4 – *Network* (отслеживает время исполнения определенных запросов и сами запросы)

5 – *Timeline* (измеряет время загрузки страницы)

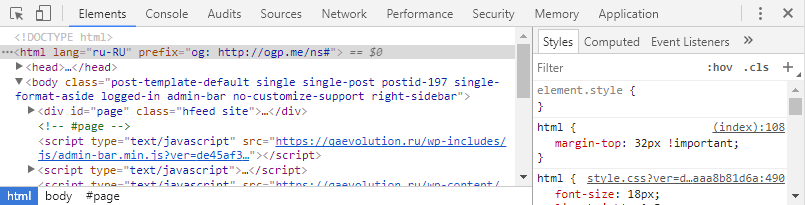
6 – *Profiles*(позволяет создавать JavaScript, профили CPU)

7 *– Resources (*позволяет просмотреть определенные сохраненные данные)

8 –*Lighthouse*(проводит проверки определенных параметров)

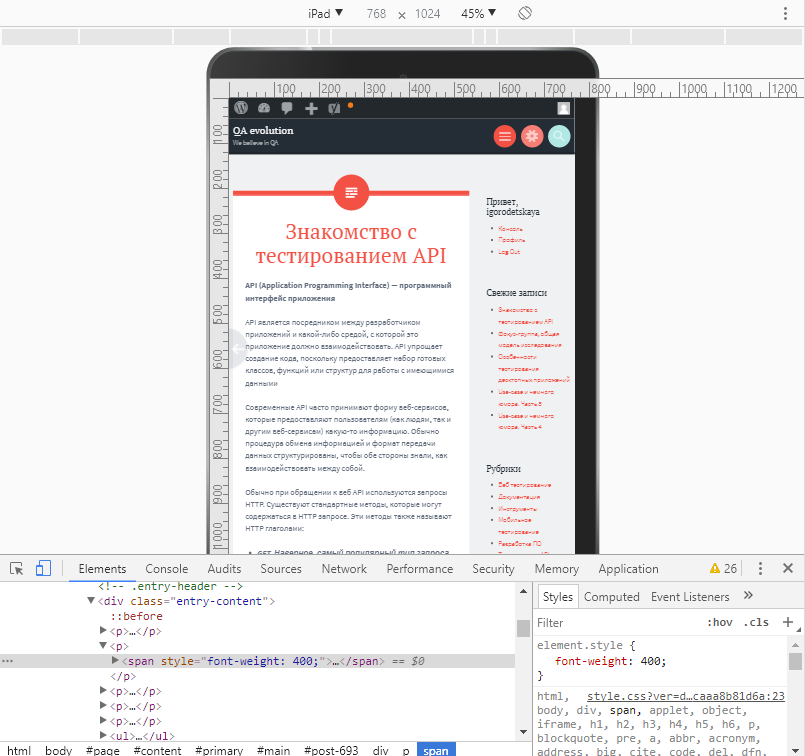
Теперь пройдемся по каждой из них в отдельности и поподробнее:

***Панель Elements***

**Панель элементов**

Панель Elements показывает разметку страницы точно так же, как она рендерится в браузере. Можно визуально менять наполнение сайта посредством изменения html/css кода в панели элементов. В левом окошке отображается html–документ, в правом – css. Проводя нехитрые манипуляции с данными можно изменить наполнение и дизайн открытой страницы. Например, можно поменять текст в окне, если редактировать его в теле html, а также изменить шрифт страницы поменяв его значение в поле css. Но это не сохранит введенных данных, а поможет просто визуально оценить примененные изменения. Для этого нужно райткликнуть его и выбрать команду «Посмотреть код».

В панели Elements есть одна функция. Можно посмотреть, как бы выглядела открытая страница на каком-нибудь девайсе с другим расширением экрана. Кликом по иконке телефона слева от вкладки Elements вызывается окно, в котором можно менять размер предполагаемого экрана, таким образом эмулируя тот или иной девайс и контролировать отображение страницы на нем. Выглядит это так:

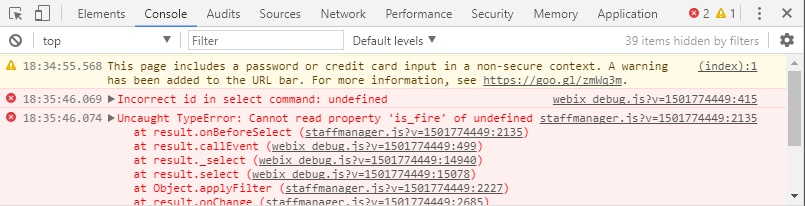
**Эмулятор девайсов**

При клике на кнопку Select Model выпадет дропдаун с огромнейшим выбором девайсов. Выбираете что вам нужно. Страница отображена так, как если бы это был девайс. Таким образом панель Elements можно использовать не только для просмотра или редактирования страницы, но и для эмуляции устройств отображения.

**Ключевые возможности**:

* Просмотр и редактирование в лайв режиме любого элемента DOM.
* Просмотр и изменение html /CSS правил применяемых к любому выбранному элементу в панели Styles.
* Просмотр всего списка событий и свойств для элемента на соответствующих вкладках.

***Панель Console***

**Панель console**

Самая популярная вкладка тестировщиков, поскольку именно здесь мы видим найденные при выполнении скрипта ошибки в коде. Также данная панель отображает различного рода предупреждения и рекомендации (как на картинке выше). Все выводимые во вкладке сообщения можно фильтровать. В ошибке также отображается ее расположение и кликнув по нему вы переместитесь во вкладку Sources, где ошибка будет выведена в общей конструкции страницы.

Очистить поле вкладки Console можно кликнув иконку перечеркнутого круга. Фильтровать сообщения в консоли можно по типу — ошибки, предупреждения, инфо, стандартный вывод, сообщения отладчика, исправленные — выбрав одну из доступных опций консоли.

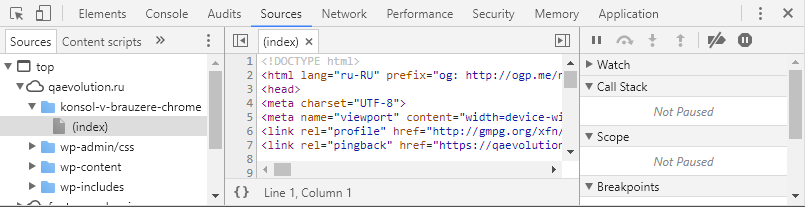
**Ключевые возможности**:

* Использование консоли как отдельной панели или как окна рядом с любой другой панелью.
* Возможность группировать большое колличество сообщения или выведите их на отдельных строках.
* Очистка всех логов или сохранения их между перезагрузкой страниц, сохранение логов в отдельный файл.
* Фильтрация по типу сообщения или по регулярному выражению.
* Логирование XHR запросов.

***Панель Sources***

Как правило, в данной вкладке проводится отладка кода программистами. Она имеет 3 окна (слева направо):

1. Зона исходных файлов. В ней находятся все подключённые к странице файлы, включая JS/CSS.
2. Зона текста. В ней находится текст файлов.
3. Зона информации и контроля.

**Панель Sources**

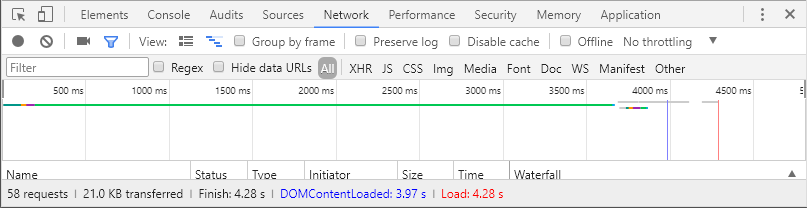
В зоне исходных файлов выбирается необходимый элемент, в зоне текст выполняется непосредственно его отладка, а в зоне информации и контроля можно запускать/останавливать отлаженный код.

Мы можем посмотреть их содержимое, отредактировать код, скопировать его или сохранить измененный файл, как новый файл. Данную вкладку можно использовать и как полноценный редактор кода подключаясь к локальным файлам через [Workspaces](https://developers.google.com/web/tools/setup/setup-workflow).

**Ключевые возможности**:

* Отладка Вашего кода с помощью брейкпоинтов.
* Можно запускать/останавливать отлаженный код.
* Запуск сниппетов с любой страницы.

***Панель Network***

**Панель Network**

Основная функция данной вкладки – запись сетевого журнала. Она дает представление о запрашиваемых и загружаемых ресурсах в режиме реального времени. Можно выявить, загрузка и обработка каких именно ресурсов занимает большее количество времени, чтобы впоследствии знать где и в чем именно можно оптимизировать страницу.

Ее удобно использовать для оптимизация загрузки страниц и мониторинг запросов.

На панели отображается таблица всех запросов к данным и файлам, над ней располагаются кнопки для фильтрации нужных Вам запросов, очистки таблицы или включения/отключения записи запросов, кнопки управления отображением таблицы. Также есть дополнительные переключатели: Preserve log — не очищать таблицу при перезагрузке страницы, Disable cache — отключить кэш браузера (будет работать только при открытом Dev Tools), Offline — эмулирует отсутствие интернета, также соседний переключатель позволяющий эмулировать скорость скачивания/загрузки данных и ping для различных типов сетей.

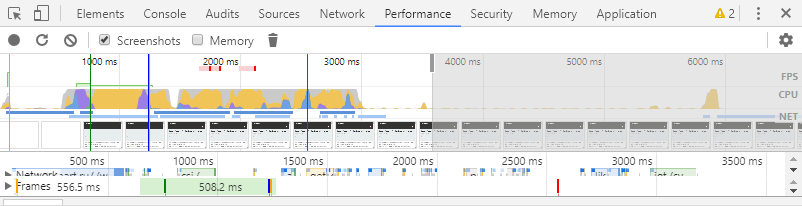
Под таблицей указано количество всех запросов, общее количество загруженных данных, общее время загрузки всех данных, время загрузки и построения DOM дерева и время загрузки всех ресурсов влияющих на отображение этой страницы.

Также стоит отметить что именно в этой вкладке в режиме Large request rows можно просмотреть запросы, которые отправляются на сервер, а также ответы, которые приходят с него, их содержание и характеристики.

**Ключевые возможности**:

* Возможность отключить кэширование или установление ограничения пропускной способности.
* Получение подробной таблицы с информацией о каждом запросе.
* Фильтрация и поиск по всему списку запросов.

***Панель Performance***

**Performance панель**

Данная вкладка используется при необходимости полного обзора затраченного времени. На что оно было потрачено, как много его потребовалось на тот или иной процесс.  Здесь отображается абсолютно вся активность, включая загрузку ресурсов и выполнение Javascript.

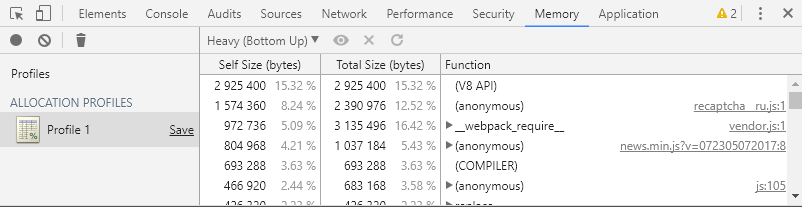
Панель отображает таймлайн использования сети, выполнения JavaScript кода и загрузки памяти. После первоначального построения графиков таймлайн, будут доступны подробные данные о выполнение кода и всем жизненном цикле страницы. Будет возможно ознакомится с временем исполнения отдельных частей кода, появится возможность выбрать отдельный промежуток на временной шкале и ознакомится с тем какие процессы происходили в этот момент.

Инструмент применяется для улучшение производительности работы Вашей страницы в целом.

**Ключевые возможности:**

* Возможность сделать запись чтобы проанализировать каждое событие, которое произошло после загрузки страницы или взаимодействия с пользователем.
* Щелкните по событию в диаграмме, чтобы посмотреть детали об этом.
* Возможность изменить масштаб таймлайн, чтобы сделать анализ проще.

***Панель Memory***

**Панель Memory**

Данная панель дает возможность профилировать время исполнения и использование памяти веб приложением или страницей, таким образом помогая понять где именно тратится много ресурсов и как можно оптимизировать код.

Содержит несколько различных профайлеров для отслеживания нагрузки которую оказывает выполнение кода на систему:

JavaScript CPU Profiler (был вынесен в отдельную панель JavaScript Profiler ) — позволяет узнать сколько процессорного времени занимает выполнение различных частей вашего JS кода.

Take Heap Snapshot — показывает распределение памяти среди JS объектов и связанные с ним элементы DOM.

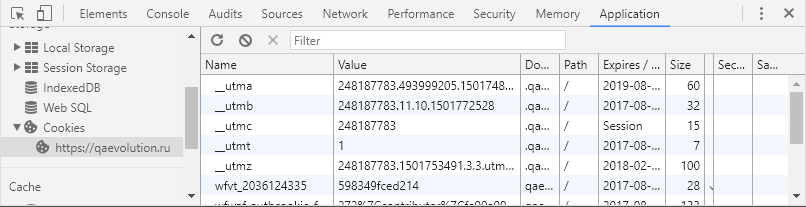
Record Allocation Timeline — записывает и отображает распределение памяти между переменными в коде. Эффективен для устранения утечек памяти.

Record Allocation Profile — записывает и отображает распределение памяти на выполнение отдельных JS функций.

**Ключевые возможности**:

* [Исправление проблем с памятью](https://developers.google.com/web/tools/chrome-devtools/memory-problems/).
* [Профилирование CPU при работе с JavaScript](https://developers.google.com/web/tools/chrome-devtools/rendering-tools/js-execution).

***Панель Application***

**Панель Application**

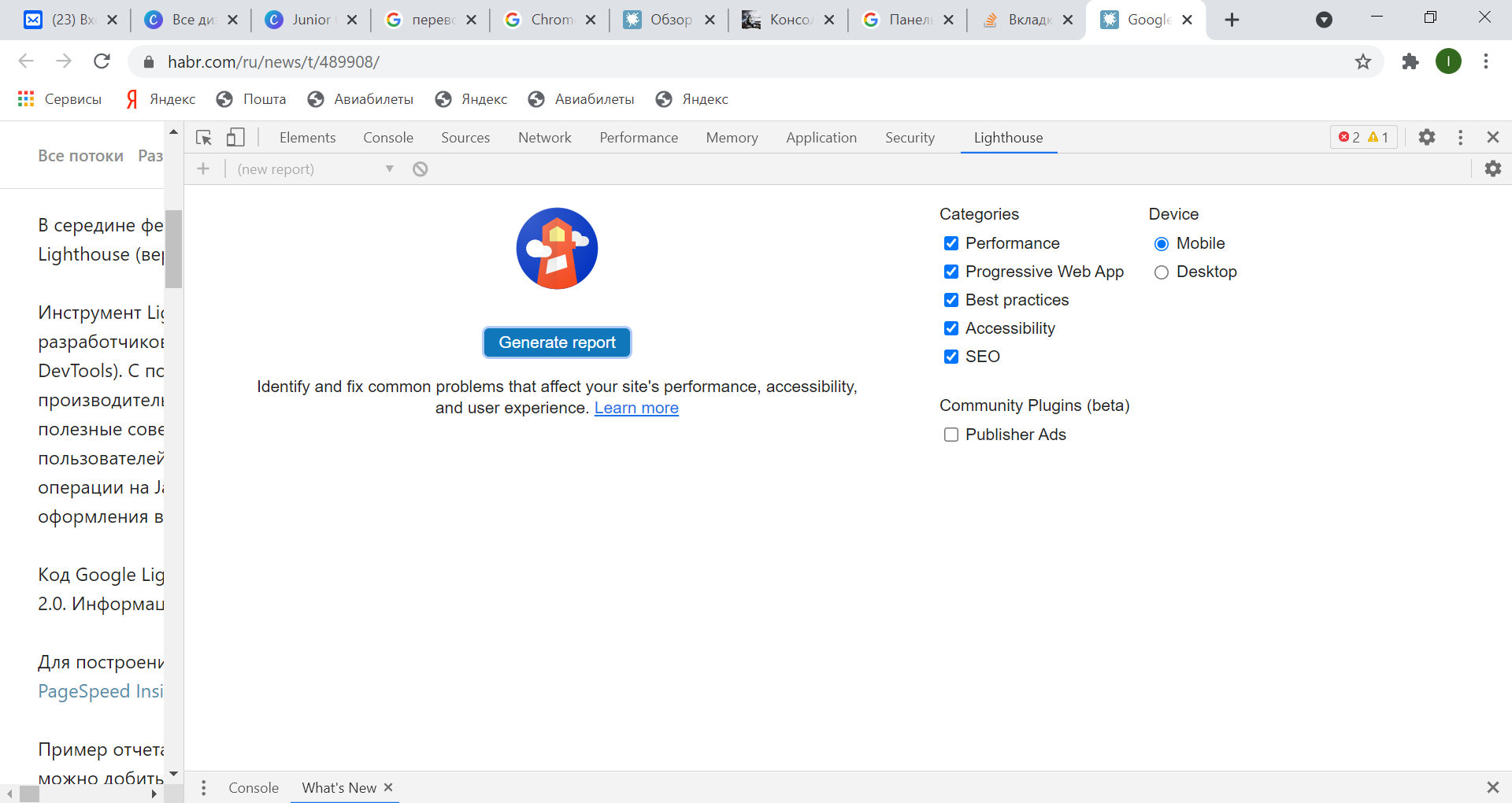
Предназначена для исследования загруженных элементов. Позволяет взаимодействовать с HTML5 Database, Local Storage, Cookies, AppCache и т.д. В данной вкладке можно почистить куки, открыв данную вкладку и кликнув иконку перечеркнутого круга

**Ключевые возможности**:

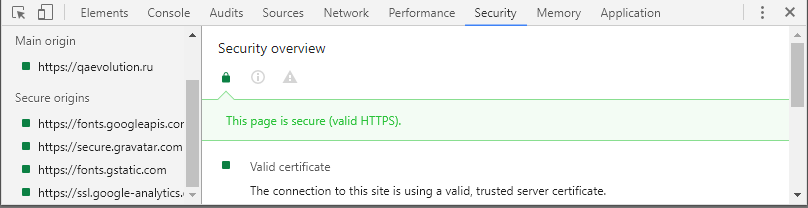
* Быстрая очистка хранилищ и кэша.
* Инспектирование и управление хранилищами, базами данных и кэшем.
* Инспектирование и удаление файлов cookie.

***Панель Lighthouse***

Инструмент Lighthouse является одним из штатных инструментов для веб-разработчиков, который входит в состав браузера Chrome (встроен в Chrome DevTools). С помощью Lighthouse можно, например, проанализировать производительность веб-приложений или скорость загрузки веб-страниц, получить полезные советы от Google о том, как можно оптимизировать веб-сайт для пользователей и повысить его эффективность, выявив излишние или ресурсоемкие операции на JavaScript. Вдобавок Lighthouse помогает оценить оптимальность оформления веб-сайта для индексации поисковыми системами (SEO).



***Панель Secuirity***

**Security панель**

Security Panel показывает информации о каждом запросе и подсвечивает те, из-за которых сайт не получает заветной зелёной иконки в статусе.

Кроме того можно получить следующую информацию:

* о проверке сертификата, подтвердил ли сайт свою подлинность TLS-сертификатом;
* TLS-соединении, обозначая использует ли ваш сайт современные безопасные протоколы;
* безопасность подгружаемых второстепенных источников.

**Ключевые возможности**:

* Окно Security Overview быстро подскажет безопасна ли текущая страница или нет.
* Возможность просмотреть отдельные источники, чтобы просмотреть соединение и детали сертификата (для безопасных источников) или узнать, какие запросы не защищены (для небезопасных источников).

***Панель****[Audits](https://developer.chrome.com/devtools" \l "audits)*

После выбора нужных настроек и нажатия кнопки Run панель аудита анализирует как загружается страница и затем предоставляет предложения по оптимизации для уменьшения времени загрузки страницы и увеличению ее отзывчивости.

Анализируются такие параметры как: кэширование ресурсов, gzip сжатие, наличие неиспользуемых частей JS кода и CSS правил и много других параметров. Далее пользователю выводится сгруппированных список рекомендаций за счет выполнения которых можно существенно оптимизировать скорость загрузки и отзывчивости страницы.

