В помощь веб-разработчику: полезные проекты и инструменты для работы с Chrome DevTools

[Alpár-Etele Méder](https://www.iconfinder.com/pocike" \t "_blank); [Turan Kent](https://www.iconfinder.com/tkportfolio" \t "_blank)

Работу современного веб-разработчика сложно представить без вспомогательных инструментов. Один из самых популярных — Chrome DevTools. Этот набор инструментов помогает тестировать, отлаживать, профилировать, проверять код на соответствие тем или иным стандартам и многое другое.

Подборка содержит ссылки на множество проектов, которые будут полезны тем, кто работает с инструментами разработчика Chrome. Думаете, что чего-то не хватает? Делитесь в комментариях.

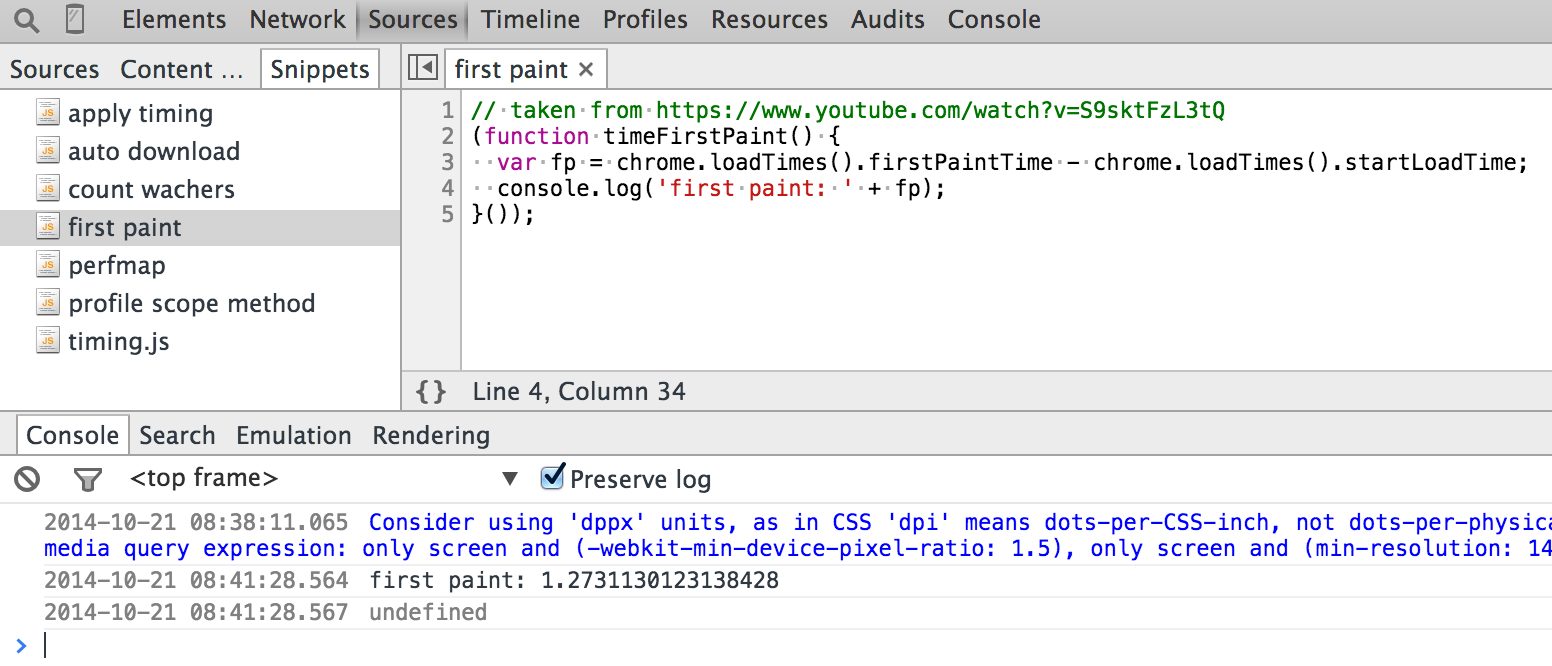
* [Обучение](https://tproger.ru/translations/chrome-devtools/#1)
* [Инструменты разработчика как IDE](https://tproger.ru/translations/chrome-devtools/#2)
* [Инструменты и экосистема](https://tproger.ru/translations/chrome-devtools/#3)
* [Протокол инструментов разработчика Chrome](https://tproger.ru/translations/chrome-devtools/#4)
* [Библиотеки для протокола](https://tproger.ru/translations/chrome-devtools/#5)
* [Разработка с протоколом](https://tproger.ru/translations/chrome-devtools/#6)
* [Адаптеры браузеров](https://tproger.ru/translations/chrome-devtools/#7)
* [Использование инструментов разработчика с другими платформами](https://tproger.ru/translations/chrome-devtools/#8)
* [Расширения инструментов разработчика](https://tproger.ru/translations/chrome-devtools/#9)

Обучение

* [Dev Tips](https://umaar.com/dev-tips/) — большая коллекция советов в форме гифок.

Chrome DevTools как среда разработки

* [Chrome DevTools App](https://github.com/auchenberg/chrome-devtools-app) — standalone-приложение, которое запускает инструменты разработчика Chrome в отдельном процессе и совершенно независимо от Chrome. Однако обратите внимание, что поддержка проекта завершилась в 2015 году.
* [DevTools Remote](https://github.com/auchenberg/devtools-remote) — удалённая отладка через инструменты разработчика. Позволяет дать разработчику доступ к отдельной вкладке браузера на стороне пользователя.
* [DevTools Snippets](https://github.com/bahmutov/code-snippets) — набор сниппетов для запуска в инструментах разработчика.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/first-paint-code-snippet.jpg)

DevTools Snippets

Инструменты и экосистема

Форматирование объектов

* [immutable-devtools](https://github.com/andrewdavey/immutable-devtools) — настраиваемое форматирование для Immutable.js.

Проверка сети

* [betwixt](https://github.com/kdzwinel/betwixt) — анализ трафика в отдельном окне вне браузера с использованием интерфейса, похожего на вкладку Network инструментов разработчика.

Профилирование процессора

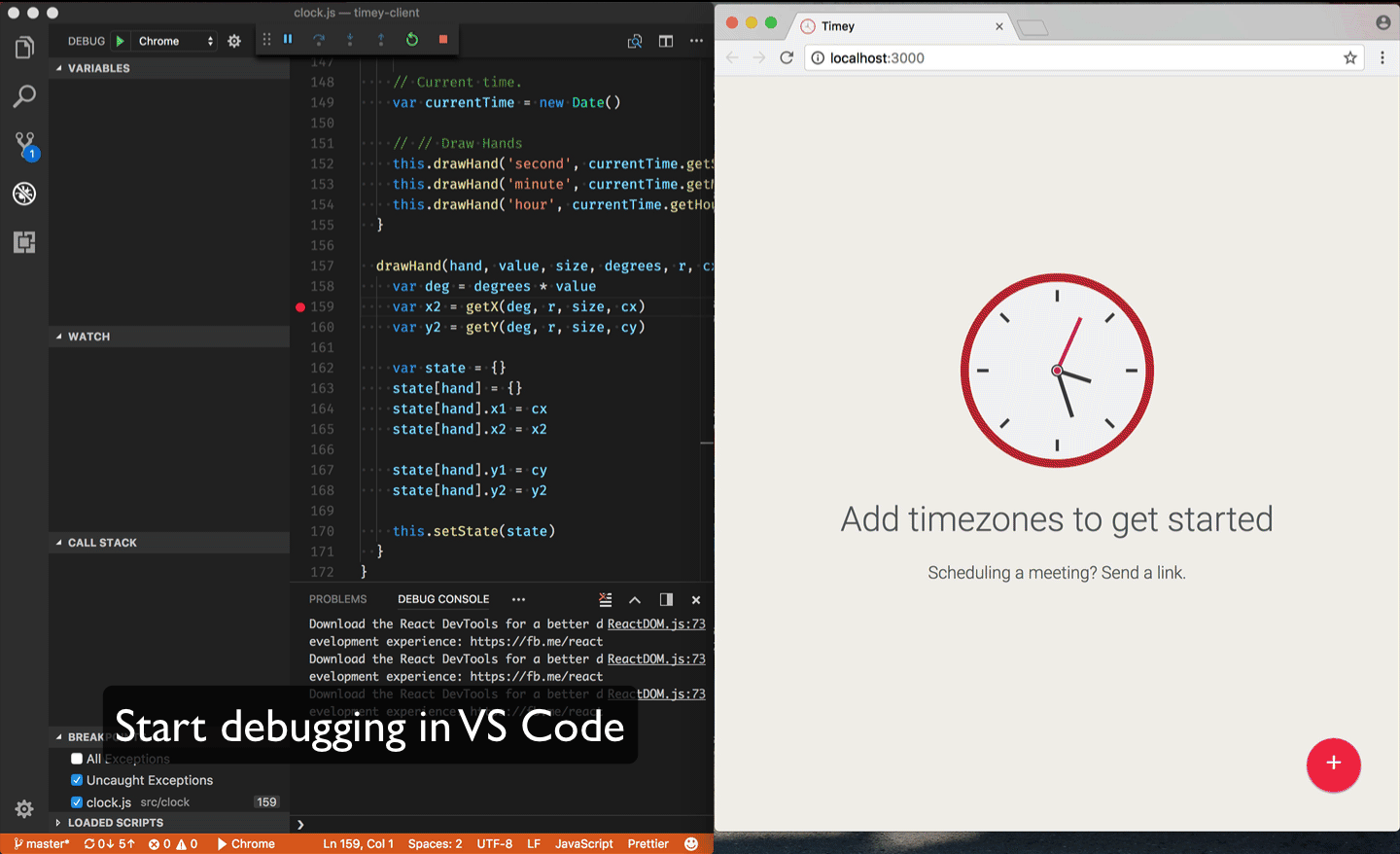
* [call-trace](https://github.com/brendankenny/call-trace) — позволяет записать граф вызовов и (опционально) время, потраченное на выполнение каждой функции JS-файла. Есть возможность генерации файла .cpuprofile.
* [cpuprofilify](https://github.com/thlorenz/cpuprofilify) — преобразует выходные данные разных профилировщиков в формат .cpuprofile.

Временные графики, трассировка и профилирование

* [DevTools Timeline Viewer](https://chromedevtools.github.io/timeline-viewer/) — делитесь ссылками на записи временных графиков.
* [snapline](https://github.com/pmdartus/snapline) — преобразует снимки временного графика в gif.

Интеграция отладчика Chrome с IDE

* [VS Code — Debugger for Chrome](https://github.com/Microsoft/vscode-chrome-debug/) — расширение VS Code, позволяющее отлаживать JavaScrip-код в браузере Google Chrome или любом другом приложении, поддерживающим протокол Chrome DevTools.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/code_chrome_devtools.gif)

VS Code — Debugger for Chrome

* [Sublime Web Inspector](https://sokolovstas.github.io/SublimeWebInspector/) — отладка JavaScript прямо в Sublime Text.
* [WebStorm & JetBrains Chrome Extension](https://www.jetbrains.com/help/webstorm/2017.1/configuring-javascript-debugger-and-jetbrains-chrome-extension.html) — позволяет WebStorm отлаживать JavaScript, просматривать дерево DOM и редактировать HTML, CSS и JS на лету.

Протокол Chrome DevTools

Протокол Chrome DevTools позволяет сторонним приложениям отслеживать, профилировать и отлаживать код в Chromium, Chrome и других Blink-based браузерах.

* [DevTools Protocol API Docs](https://chromedevtools.github.io/devtools-protocol/) — документация по протоколу.
* [ChromeDevTools/devtools-protocol](https://github.com/chromedevtools/devtools-protocol) — багтрекер для проблем с протоколом.
* [Remote Debug Gateway](https://github.com/RemoteDebug/remotedebug-gateway) — позволяет проводить отладку сразу в нескольких браузерах.
* [DevTools Backend](https://github.com/christian-bromann/devtools-backend) — standalone-реализация бекенда инструментов разработчика Chrome для отладки произвольных веб-платформ вроде приложений HbbTV на Smart TV.
* [RemoteDebug](https://github.com/RemoteDebug) — универсальные протоколы отладки для современных браузеров.
* [ChromeDriver](https://sites.google.com/a/chromium.org/chromedriver/) — официальная реализация Selenium/WebDriver для Chrome, работающая на основе протокола инструментов разработчика.
* [Chrome Protocol Proxy](https://github.com/wendigo/chrome-protocol-proxy) — обратный прокси для отладки с помощью протокола инструментов разработчика.
* [Puppeteer](https://github.com/GoogleChrome/puppeteer/) — Node-библиотека, предоставляющая высокоуровневый API для управления Chrome или Chromium через протокол инструментов разработчика.

Библиотеки на различных языках для работы с протоколом

* JavaScript/Node.js: [chrome-remote-interface](https://github.com/cyrus-and/chrome-remote-interface" \t "_blank) — наиболее используемый JavaScript API для протокола.
* TypeScript/Node.js: [chrome-debugging-client](https://github.com/krisselden/chrome-debugging-client" \t "_blank) — TypeScript-клиент для отладки с поддержкой async/await.
* Java: [cdp4j](https://github.com/webfolderio/cdp4j) — Java-библиотека, которую можно использовать для тестирования веб-страниц. Используется с Chrome/Chromium.
* Java: [karate](https://intuit.github.io/karate/karate-core/" \t "_blank) — фреймворк с Java API для тестирования веб-сервисов, работает с Chrome с помощью протокола инструментов разработчика.
* Python: [PyChromeDevTools](https://github.com/marty90/PyChromeDevTools" \t "_blank) — Python-обёртка для протокола инструментов разработчика Chrome.
* Python: [chromewhip](https://github.com/chuckus/chromewhip" \t "_blank) — сервис, который запускает процесс headless Chrome, обёрнутый в HTTP API.
* Python: [pychrome](https://github.com/fate0/pychrome" \t "_blank) — Python-пакет для протокола инструментов разработчика Chrome.
* Go: [chromedp](https://github.com/chromedp/chromedp" \t "_blank) — простой способ для работы в Go с браузерами, поддерживающими протокол инструментов разработчика Chrome, без нужды во внешних зависимостях вроде Selenium или PhantomJS.
* Go: [cdp](https://github.com/mafredri/cdp" \t "_blank) — библиотека на Go для протокола.
* Go: [gcd](https://github.com/wirepair/gcd" \t "_blank) — неофициальная реализация удалённого отладчика Google Chrome на Go.
* Go: [godet](https://github.com/raff/godet" \t "_blank) — удалённый клиент для инструментов разработчика Chrome.
* C#/dotnet: [chrome-dev-tools](https://github.com/BaristaLabs/chrome-dev-tools-generator" \t "_blank) — C# обёртка на основе акутальной версии протокола. Можно настраивать с помощью шаблонов.
* Ruby: [ChromeRemote](https://github.com/cavalle/chrome_remote/" \t "_blank) — клиентская реализация протокола на Ruby.
* Kotlin: [chrome-reactive-kotlin](https://github.com/wendigo/chrome-reactive-kotlin" \t "_blank) — реактивная (rxjava 2.x) низкоуровневая библиотека на Kotlin.
* Clojure: [clj-chrome-devtools](https://github.com/tatut/clj-chrome-devtools" \t "_blank) — библиотека для протокола на Clojure. Обёртка для API протокола генерируется автоматически и обновляется при его изменении.
* PHP: [chrome-devtools-protocol](https://github.com/jakubkulhan/chrome-devtools-protocol" \t "_blank) — клиент для протокола на PHP.

Разработка с протоколом

* [chrome-remote-interface Wiki](https://github.com/cyrus-and/chrome-remote-interface/wiki) — вики с набором полезных рецептов.
* [Начало работы с Headless Chrome](https://developers.google.com/web/updates/2017/04/headless-chrome).
* [crmux](https://github.com/sidorares/crmux) — мультиплексор для протокола, позволяющий установить несколько подключений инструментов разработчика к одной вкладке.
* [automated-chrome-profiling](https://github.com/paulirish/automated-chrome-profiling#readme) —  автоматизация JavaScript-профилирования в Chrome с помощью Node.js.
* [chrome-devtools-frontend](https://www.npmjs.com/package/chrome-devtools-frontend) — клиентская часть инструментов разработчика Chrome.
* [chrome-timeline-model](https://www.npmjs.com/package/devtools-timeline-model) — npm-пакет для обработки результатов профилирования.
* [headless-devtools](https://github.com/cowchimp/headless-devtools) — позволяет совершать из кода действия, доступные в инструментах разработчика. Использует Headless Chrome и Puppeteer.
* [sloth](https://github.com/denar90/sloth) — расширение Chrome, позволяющее включить троттлинг процессора и сети для выбранных вкладок.
* [crconsole](https://github.com/sidorares/crconsole) — консоль и отладчик Chrome в терминале.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/cdt-terminal.jpg)

Консоль и отладчик Chrome в терминале

Адаптеры браузеров

* [Remote Debug Firefox adapter](https://github.com/RemoteDebug/remotedebug-firefox-adapter) — переводит протокол Firefox в протокол Chrome.
* [ios-webkit-debug-proxy](https://github.com/google/ios-webkit-debug-proxy) — прокси протокола Chrome для iOS-устройств.
* [Remote Debug iOS WebKit adapter](https://github.com/RemoteDebug/remotedebug-ios-webkit-adapter) — основан на предыдущем проекте, переводит API протокола WebKit в протокол Chrome.
* [IE Diagnostics Adapter](https://github.com/Microsoft/IEDiagnosticsAdapter) — адаптер для протокола Microsoft IE 10/11.
* [Edge Diagnostics Adaptor](https://github.com/Microsoft/edge-diagnostics-adaptor) — адаптер протокола, который позволяет использовать инструменты для отладки Edge, используя протокол Chrome.

Использование инструментов разработчика с другими платформами

* [Facebook Stetho](https://github.com/facebook/stetho) — отладка Android с помощью инструментов разработчика Chrome.
* [Dirac](https://github.com/binaryage/dirac) — отладка ClojsureScript.
* [Mare](https://github.com/muzuiget/mare) — отладка Lua.
* [PonyDebugger](https://github.com/square/PonyDebugger) — удалённая отладка нативных iOS-приложений.
* [go-debugger-devtools](https://github.com/allada/gdd) — отладка Go.
* [Debugging Node.js with Chrome DevTools](https://medium.com/@paul_irish/debugging-node-js-nightlies-with-chrome-devtools-7c4a1b95ae27) — гайд по использованию полной поддержки отладки и профилирования в Node v6.3+.
* [devtool](https://github.com/Jam3/devtool) — отладка и профилирование Node.js-приложений с помощью инструментов разработчика и Electron.
* [buggerJS](https://github.com/buggerjs/bugger) — использование возможностей инструментов разработчика в Node.

Расширения инструментов разработчика

Доступность для всех

* [Chromelens](http://chromelens.xyz/) — посмотрите, как ваше приложение увидят люди с проблемами со зрением и как у вас обстоят дела с навигацией для слепых.

Рабочий процесс

* [Clockwork](https://chrome.google.com/webstore/detail/clockwork/dmggabnehkmmfmdffgajcflpdjlnoemp?hl=en) — просмотр данных профилирования PHP-приложения.
* [Emulated Device Lab](https://chrome.google.com/webstore/detail/emulated-device-lab/oaonfodocibcdobdeelbbfggjombamff) — тестирование сразу на нескольких эмулированных устройствах.
* [RailsPanel](https://chrome.google.com/webstore/detail/railspanel/gjpfobpafnhjhbajcjgccbbdofdckggg?hl=en-US) — просмотр данных профилирования приложения на Ruby on Rails.
* [React Developer Tools](https://chrome.google.com/webstore/detail/react-developer-tools/fmkadmapgofadopljbjfkapdkoienihi) — просмотр иерархий компонентов React.
* [EmberJS Inspector](https://chrome.google.com/webstore/detail/ember-inspector/bmdblncegkenkacieihfhpjfppoconhi) — позволяет просматривать объекты EmberJS.
* [VueJS Developer Tools](https://github.com/vuejs/vue-devtools) — просмотр компонентов VueJS и управление их данными.
* [Angular Batarang](https://chrome.google.com/webstore/detail/angularjs-batarang/ighdmehidhipcmcojjgiloacoafjmpfk) — просмотр области видимости Angular-приложения и профилирование его данных.
* [Augury](https://augury.angular.io/) — отладка и профилирование приложений на Angular 2.
* [Marionette Inspector](https://chrome.google.com/webstore/detail/marionette-inspector/fbgfjlockdhidoaempmjcddibjklhpka) — просмотр отображений, событий и данных приложения на Marionette.
* [Backbone Debugger](https://chrome.google.com/webstore/detail/backbone-debugger/bhljhndlimiafopmmhjlgfpnnchjjbhd) — просмотр отображений, событий и моделей приложения на Backbone.
* [App Inspector for Sencha](https://chrome.google.com/webstore/detail/app-inspector-for-sencha/pbeapidedgdpniokbedbfbaacglkceae) — просмотр дерева компонентов, хранилищ данных, событий и разметки приложений на Sencha ExtJS/Touch.
* [Redux Devtools](https://chrome.google.com/webstore/detail/redux-devtools/lmhkpmbekcpmknklioeibfkpmmfibljd) — инструменты разработчика для отслеживания изменения состояний приложения.
* [Three.js](https://chrome.google.com/webstore/detail/threejs-editor-extension/fbgbekpggeldiacgjkacbkkcbjhmakea/) — редактирование проектов на three.js.
* [Insight](https://github.com/3Dparallax/insight/) — расширение для WebGL-разработки.
* [Metal.js Developer Tools](https://chrome.google.com/webstore/detail/metaljs-developer-tools/fagnjmppkokolnbloalifcmcooldhiik) — просмотр иерархии компонентов Metal.

UX

* [DevTools Author](https://chrome.google.com/webstore/detail/devtools-author/egfhcfdfnajldliefpdoaojgahefjhhi) — подборка тем для кастомизации инструментов разработчика.
* [Zero Dark Matrix](https://chrome.google.com/webstore/detail/devtools-theme-zero-dark/bomhdjeadceaggdgfoefmpeafkjhegbo) — тёмная тема для инструментов разработчика.

Производительность

* [Chrome React Perf](https://chrome.google.com/webstore/detail/react-perf/hacmcodfllhbnekmghgdlplbdnahmhmm) — интерфейс для пакета react-addons-perf.

Смотрите также: [Удобные возможности консоли разработчика Chrome, о которых вы можете не знать](https://tproger.ru/translations/chrome-command-line-api-cheatsheet/)

На основе [Awesome chrome-devtools](https://github.com/ChromeDevTools/awesome-chrome-devtools" \t "_blank)

[Никита Прияцелюк](https://tproger.ru/author/NickPrice), последний центурион

* [Google Chrome](https://tproger.ru/tag/google-chrome/), [Веб-разработка](https://tproger.ru/tag/web/), [Инструменты](https://tproger.ru/tag/tools/)

Вакансии в тему:

Лого компании «Holding BR»

[Go/Golang Developer / Разработчик Golang (Senior/Lead Developer)](https://tproger.ru/jobs/go-golang-developer-holding-br/?utm_source=tproger_ru_after_main_page_banner" \o "Go/Golang Developer / Разработчик Golang (Senior/Lead Developer))

Holding BR, Москва, можно удалённо, от 160 000 ₽

Лого компании «CrmBus»

[Web разработчик](https://tproger.ru/jobs/web-dev-crmbus/?utm_source=tproger_ru_after_main_page_banner" \o "Web разработчик)

CrmBus, удаленно, от 70 000 ₽

[**Все вакансии**](https://tproger.ru/jobs/?utm_source=tproger_ru_after_article_banner)

[Бесплатное обучение ФорексуMTBankFX обучает настоящему**Форексу**. Профессиональные тренеры. Запишитесь бесплатно!MTBankFX обучает настоящему **Форексу**. Профессиональные тренеры. Запишитесь бесплатно!mtbank.byМинскПерейтиЯндекс.ДиректСодействие в подборе финансовых услуг/организацийСодействие в подборе финансовых услуг/организаций](https://an.yandex.ru/count/NAvj390HWxW502i2CKNfG5m00000EFow4K02I09Wl0Xe173CZBBG1801kjQpwz76h-RF0OW1-8Zr-IUG0Shwp-unc06-ZjgWCw01oFhFxZ6e0RIEsg0pk07IzEov5S010jW1tDlp3-01i8RL2-W1IFW10OW21A02pEUa1ha2gWIT6TEEmwVm0l3UxiG4-0A2W820WF0TW0Ek-y642eW3slFpj0-O0xJG2x031B040RW4_m7e1CS9-0IWh3w81Q2iFf05gAP1e0MyGQW5l46m1Rn1k0N9HC05Yeeqo0NODj05syC7u0Lby0K1c0Q2qApp3g06kWAe1he2oGPEUFpZJ3ni7D4651MU8ClyO0VP1W000A280000a0Vu0Qa7ytM4aaKgLF-m1u20a0p01y3tZjGAq0S2u0VA3e081j08keg0WS2GW0BW2EQpaWE02W712W0000000F0_s0e2u0g0YNg82mYg2n067LUmytK00D3QStUjVmK0m0k0emN82u3Kam7P2_DrX995AbJ_w0kWh3xm2mQ8383lthu1w0mRc0soXCG1u0q2YGu00000003mFv0Em8Gzc0xEyF6VpecFjXoW3i24FR0E0Q4F00000000y3-e3_FhaTIno8gg_m7P3sXrGgsn3_G_u0y1W13BqeKEa13Awuo3Wf2gaWsQ40aH00000000y3_840Ju40604Vh__________m684G6G4GAX4G0000000F0_?stat-id=4&test-tag=131392050605057&format-type=43&actual-format=39&banner-test-tags=eyI2NjkyNjY3MDQ1IjoiMTMxMzkxNjM5NTY4Mzg0In0%3D&)

Также рекомендуем:

Начало формы

Рассылка «Аргументы и функции»

Только самые важные IT-новости

Подписаться

Конец формы

[О проекте](https://tproger.ru/about) [Реклама](https://tproger.ru/ad)[Мобильная версия](https://tproger.ru/translations/chrome-devtools/amp/)

[Пользовательское соглашение](https://tproger.ru/wp-content/uploads/custom/terms_v1.pdf)  
[Политика конфиденциальности](https://tproger.ru/wp-content/uploads/custom/privacy_v1.pdf)

[«Аргументы и функции» — рассылка новостей](https://tproger.ru/newsletter/)

[Включить уведомления](https://tproger.ru/translations/chrome-devtools/)

Нашли опечатку? Выделите фрагмент и отправьте нажатием Ctrl+Enter.

Спасибо за внимательность.  
Опечатка уже отправлена нашим редакторам.