

Хабр



КАК СТАТЬ АВТОРОМ



Зарплаты IT-специалистов



Войти



empenoso

6 янв в 05:23

# Как я научился оценивать популярность статей через парсинг показателей сайтов

Простой

7 мин

2.4K

Контент и копирайтинг\*, Управление медиа\*, Node.JS\*, Open source\*

Кейс

Победитель Технотекста 7

Сезон Open source

Я уже давно пишу статьи про различные аспекты IT-технологий, инвестиции, автоматизацию и умные дома на разных площадках: Хабр, Т—Ж, СмартЛаб, Пикабу, VC.ru и других.

За всё время накопилось примерно 250 статей, [которые по итогу свёл в таблицу](#). Но вот задумываться о популярности статей и их реальном эффекте стал относительно недавно.

1	A	B	C	D	E	F	G	H	I
	Название	Тип (задается вручную)	Дата выхода (задается вручную)	Выходные данные (задается вручную)	Автор(ы) (задается вручную)	Тема (задается вручную)	Тираж для печатных изданий	Просмотры	Коментарии
4	Таблицы и скрипты Гугл для бизнеса и анализа	Веб	2024-12-19	<a href="#">https://habr.com/ru/articles/957068/</a>	М.В. Шардин	Скрипты	-	950	
5	Топ-тренды Смартлаба 2024: что читали больше всего?   Смартлаб	Веб	2024-12-19	<a href="#">https://smart-lab.ru/mobiletopic/1096274/</a>	М.В. Шардин	Джава и гугл скрипты	-	4 687	
6	Что читали на Хабре в 2024 году: анализ статей с Node.js, Google Sheets и каплей ChatGPT   Хабр	Веб	2024-12-18	<a href="#">https://habr.com/ru/articles/957068/</a>	М.В. Шардин	Джава и гугл скрипты	-	7 109	
7	С бумажки на цифровую карту: генерация файла из таблицы для импорта на карту и геокодирование адресов с помощью Python   Хабр	Веб	2024-12-12	<a href="#">https://habr.com/ru/articles/956558/</a>	М.В. Шардин	Картография	-	1 974	
8	С бумажки на цифровую карту: генерация файла из таблицы для импорта на карту и геокодирование адресов с помощью Python   Пикабу	Веб	2024-12-12	<a href="#">https://pikabu.ru/story/s_bumazki_na_tsifrovuyu_kartu_generatsiya_fayla_iz_tablitsy_dlya_importa_na_kartu_i_geokodirovaniye_adresov_s_pomoshchyu_python_12126449</a>	М.В. Шардин	Картография	-	4 271	
9	Слабоумие и отага: как найти ликвидные облигации с доходностью до 40% и ежемесячными выплатами   Смартлаб	Веб	2024-12-05	<a href="#">https://smart-lab.ru/mobiletopic/1090879/</a>	М.В. Шардин	Инвестиции	-	5 823	
10	Слабоумие и отага: как найти ликвидные облигации с доходностью до 40% и ежемесячными фиксированными выплатами   Хабр	Веб	2024-12-05	<a href="#">https://habr.com/ru/articles/953752/</a>	М.В. Шардин	Инвестиции	-	7 609	
11	Слабоумие и отага: как найти ликвидные облигации с доходностью до 40% годовых и ежемесячными фиксированными выплатами на Московской бирже   Пикабу	Веб	2024-12-05	<a href="#">https://pikabu.ru/story/slaboumie_i_otaga_kak_nayti_likvidnyye_obligatsii_s_dokhodnoshyu_do_40_godovykh_i_eshcheyezhichnyimi_fiksirovannymi_vypplatami_na_moskovskoy_birzhe_12099410</a>	М.В. Шардин	Инвестиции	-	3 444	
12	Слабоумие и отага: как найти ликвидные облигации с доходностью до 40% годовых и ежемесячными	Веб	2024-12-05	<a href="#">https://nationalanswer.club/post/1087/</a>	М.В. Шардин	Инвестиции	-	нд	

## Почему я решил собирать статистику публикаций?

РЕКЛАМА



Получи грант за код

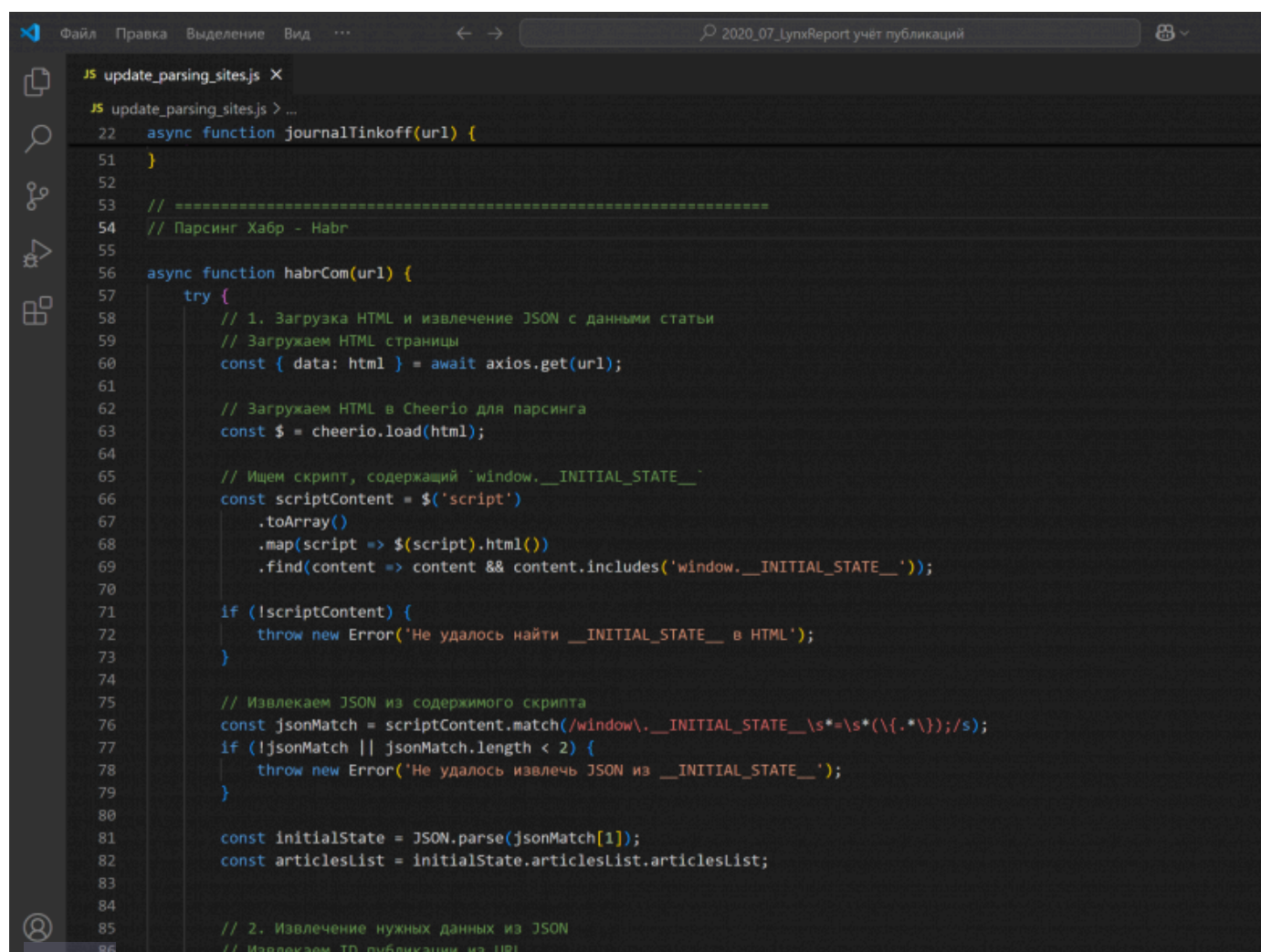
Конкурс open source проектов

на Хабре статьи набирали большой рейтинг при малом количестве комментариев (но это редко).

Однако очевидно одно - статистика заставляет посмотреть на материалы со стороны. Например статья про [то, как я при помощи двух скриптов смог автоматически сгенерировать опись документов для 700 страниц](#) на непрофильном Пикабу собрала три месяца назад почти 75 тысяч просмотров и 80 комментариев, а на Хабре эта же тема была не особо популярна.

Или [статья про то, что читали на Хабре в 2024 году: анализ статей с Node.js, Google Sheets и каплей ChatGPT](#) - собрала на Хабре три недели назад рейтинг +68, но «всего» 7 тысяч просмотров.

## Зачем я написал Open Source скрипт?



```
JS update_parsing_sites.js X
JS update_parsing_sites.js > ...
22  async function journalTinkoff(url) {
51  }
52
53  // =====
54  // Парсинг Хабр - Habr
55
56  async function habrCom(url) {
57    try {
58      // 1. Загрузка HTML и извлечение JSON с данными статьи
59      // Загружаем HTML страницы
60      const { data: html } = await axios.get(url);
61
62      // Загружаем HTML в Cheerio для парсинга
63      const $ = cheerio.load(html);
64
65      // Ищем скрипт, содержащий 'window.__INITIAL_STATE__'
66      const scriptContent = $('script')
67        .toArray()
68        .map(script => $(script).html())
69        .find(content => content && content.includes('window.__INITIAL_STATE__'));
70
71      if (!scriptContent) {
72        throw new Error('Не удалось найти __INITIAL_STATE__ в HTML');
73      }
74
75      // Извлекаем JSON из содержимого скрипта
76      const jsonMatch = scriptContent.match(/window\.__INITIAL_STATE__\s*=\s*(\{.*\});/s);
77      if (!jsonMatch || jsonMatch.length < 2) {
78        throw new Error('Не удалось извлечь JSON из __INITIAL_STATE__');
79      }
80
81      const initialState = JSON.parse(jsonMatch[1]);
82      const articlesList = initialState.articlesList.articlesList;
83
84
85      // 2. Извлечение нужных данных из JSON
86      // Извлекаем ID публикации из URL
```



Получи грант за код

Конкурс open source проектов

Извлечение просмотров, комментариев, закладок и рейтинга из каждой статьи вручную занимало бы много времени, поэтому я решил пойти путём автоматизации. Написал скрипт, который скачивает эти данные по статьям и помещает сразу в одну удобную таблицу, где я вижу, какие темы стоит развивать дальше и на каких ресурсах.

Это [Open Source скрипт, размещенный на Гитхабе](#), который состоит из Google Apps Script и Node.js частей и обе эти части работают с итоговой сводной Google Таблицей.

Если вы автор или агентство и хотите видеть полную картину популярности ваших материалов, выявлять, какие темы интересуют аудиторию, а какие требуют доработки, автоматизированный инструмент сбора данных - это то, что вам нужно. Я уже испытал это на себе и расскажу подробнее об этом в статье.

## Особенности механизма парсинга сайтов, где размещены публикации

Изначально я использовал Google Apps Script внутри Гугл Таблиц для автоматизации извлечения просмотров, комментариев, закладок, рейтинга. Однако по мере того, как я расширялся на новые платформы, то обнаружил, что некоторые сайты ограничивают доступ к определенным данным, что потребовало от меня перехода на серверный язык JavaScript - Node.js.

**Конкретный пример:** сайт инвесторов и трейдеров Смартлаб имеет скрытое АПИ для получения просмотров статей.

Если обращение идёт через Google Apps Script внутри Гугл Таблиц, то сервер просто не отдаёт просмотры, возвращая пустой ответ, потому что заголовки headers и User-Agent не поддерживаются Google Apps Script:

```
function fetchApiResponse() {  
  const url = 'https://smart-lab.ru/cgi-bin/gcn.fcgi?list=1083556&func=func8422&_=17319  
  
  try {  
    // Выполняем запрос к API с детализированными заголовками  
    const response = UrlFetchApp.fetch(url, {  
      method: 'get', // HTTP метод GET  
      muteHttpExceptions: true, // Не выбрасывать ошибки для HTTP ответов с кодами, отл
```



**Получи грант за код**

Конкурс open source проектов

```
'Accept-Language': 'ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7',
'Cache-Control': 'max-age=0',
'Sec-CH-UA': '"Chromium";v="130", "Google Chrome";v="130", "Not?A_Brand";v="99"',
'Sec-CH-UA-Mobile': '?0',
'Sec-CH-UA-Platform': '"Windows"',
'Sec-Fetch-Dest': 'document',
'Sec-Fetch-Mode': 'navigate',
'Sec-Fetch-Site': 'none',
'Sec-Fetch-User': '?1',
'Upgrade-Insecure-Requests': '1'
}
});

// Проверяем HTTP код ответа
const responseCode = response.getResponseCode();
if (responseCode !== 200) {
  console.error(`Ошибка: Получен HTTP код ответа ${responseCode}`);
  return;
}

// Получаем содержимое в виде обычного текста
const content = response.getContentText();

// Логируем необработанный ответ для отладки
console.log('Необработанный ответ API:', content);

} catch (error) {
  // Обрабатываем ошибки во время выполнения запроса
  console.error('Ошибка при выполнении запроса к API:', error.message);
}
}
```

**Получи грант за код**

Конкурс open source проектов

## Google Apps Script

Если же запрос идёт через Node.js и использование User-Agent, то в ответ возвращается число просмотров даже без использования эмуляции браузера:

```
const axios = require('axios');

async function fetchApiResponse() {
  const url = 'https://smart-lab.ru/cgi-bin/gcn.fcgi?list=1083556&func=func8422&_=173';

  try {
    // Установка заголовка User-Agent для имитации браузера Chrome
    const options = {
      headers: {
        'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/53
      }
    };

    // Выполнение запроса к API для получения просмотров
    const apiResponse = await axios.get(url, options);
    const responseText = apiResponse.data;

    // Логируем необработанный ответ для отладки
    console.log('Необработанный ответ API:', responseText);

  } catch (error) {
```

**Получи грант за код**

Конкурс open source проектов

```
}  
}  
  
fetchApiResponse();
```

Node.js

## Проблема оценки популярности статей

Обратная связь от аудитории может сильно варьироваться в зависимости от платформы, формата подачи материала и времени публикации.

Например, статья, выложенная одновременно на разных сайтах, может получить совершенно разные результаты. Уже упоминаемая выше статья "Как я при помощи двух скриптов смог автоматически сгенерировать опись документов для 700 страниц" собрала:

**vc.ru:**

225 / 3 / 2 / 2

Просмотры / Комментарии / Закладки / Рейтинг

**Пикабу:**

74 615 / 76 / 0 / 903

Просмотры / Комментарии / Закладки / Рейтинг

**Получи грант за код**

Конкурс open source проектов

Просмотры / Комментарии / Закладки / Рейтинг

На Пикабу статья вызвала бурное обсуждение, в то время как на vc.ru аудитория осталась почти равнодушной.

Зачем собирать статистику публикаций?

Анализ метрик (просмотры, комментарии, закладки, рейтинг) помогает понять, что действительно интересует вашу аудиторию и какие темы популярны на различных ресурсах.

- **Просмотры:** показывают, сколько людей заинтересовалось темой.
- **Комментарии:** демонстрируют вовлечённость, обсуждения.
- **Закладки:** отражают ценность материала для долгосрочного использования.
- **Рейтинг:** индикатор общего одобрения.

Анализ этих данных помогают адаптировать стиль, темы и платформы для повышения эффективности публикаций.

## Создание Open Source скрипта для сбора данных

За несколько лет написал [Google Apps Script](#), а затем и [Node.js](#), которые предназначены для автоматического сбора показателей публикаций с различных платформ.

Важно, что скрипты приспособлены к конкретной структуре каждой платформы, используя API (Тинькофф Журнал, Пикабу, Смартлаб) или парсинг HTML-кода (Хабр, vc.ru).



**Получи грант за код**

Конкурс open source проектов

## Различие Google Apps Script и Node.js версий

Если очень кратко, то версия на Google Apps Script проста в использовании, запускается прямо в браузере, идеальна для быстрой интеграции с таблицами Google, а Node.js версия мощнее, работает на сервере, подходит для более сложных задач и интеграций.

Google Apps Script – это облачное решение, подходящее для задач, не требующих больших вычислительных ресурсов. Он интегрирован в Google Workspace и позволяет легко работать с веб-страницами, API Google, используя встроенные функции. Однако его функционал ограничен, и он не может работать с ресурсоёмкими задачами или сложным парсингом.

Node.js – это серверная среда, использующая Javascript. Это мощное решение для сложного парсинга, работы с API, обработки больших объёмов данных и реализации сложных логик. Node.js позволяет использовать различные библиотеки (например, Axios, Cheerio, Puppeteer), расширяя возможности скрипта. Однако Node.js требует установки и настройки на сервере или локально.



**Получи грант за код**

Конкурс open source проектов



Публикация скриптов на GitHub в рамках философии Open Source имеет несколько преимуществ. Во-первых, это позволяет другим пользователям использовать и адаптировать скрипт под свои нужды. Во-вторых, открытый исходный код способствует прозрачности и аудиту кода, что помогает обнаруживать и исправлять ошибки. В-третьих, сообщество разработчиков может помочь в улучшении и расширении функциональности скрипта, если захочет :)

## Пример анализа собранной статистики для выбора будущих тем

Мне кажется немного некорректно проводить анализ собственных статей здесь и поэтому я оттолкнусь от анализа двух популярных платформ: [Хабра](#) и [Смартлаба](#) по всем их публикациям за 2024 год.



**Получи грант за код**

Конкурс open source проектов

Исходные данные были получены мной путем парсинга двух ресурсов. Использовался автоматизированный сбор данных с использованием Node.js для парсинга HTML-страниц и последующего извлечения информации о каждой статье. Данные о просмотрах, комментариях, рейтинге и закладках позволяли оценить популярность и вовлеченность аудитории.

Фильтр по минимальному количеству просмотров (5 000 на Смартлабе, 30 000 на Хабре) и рейтингу (более 30 на Хабре) позволил отфильтровать статьи, которые не соответствовали критериям популярности.

**Определение трендов и предпочтений аудитории для разных платформ: Хабр и Смартлаб.**



**Получи грант за код**

Конкурс open source проектов

Хабр более ориентирован на технические темы, такие как настройка программного обеспечения, обходы блокировок, разработка и советы по программированию. Наиболее популярные темы за 2024 год на Хабре - проблемы блокировки YouTube, решение проблем с GoodbyeDPI, уязвимости безопасности.

На Смартлабе, наоборот, доминируют темы, связанные с инвестициями, экономикой и геополитикой. Трендовые темы Смартлаба - санкции, рубль, ключевая ставка, дедолларизация, анализ рынка акций и облигаций.

### **Оптимизация публикаций под разные платформы: про что писать и на чём делать акценты.**

Для эффективной публикации на каждой платформе важно учитывать интересы ее аудитории. На Хабре следует фокусироваться на конкретных технических проблемах и их решениях, практических советах и "человеческом" стиле. Статьи с интересными заголовками, посвященные узким, конкретным темам, вероятнее всего, получат большую вовлеченность.

На Смартлабе, ключевым является анализ геополитических и экономических трендов, связанных с финансами. Оптимально создавать качественный аналитический контент, предлагающий рекомендации по инвестированию, с подкреплением фактами и экспертизой.

## **Выводы**

Мой Open Source скрипт для автоматического сбора статистики (просмотры, комментарии, закладки, рейтинг) позволяет анализировать популярность статей на разных платформах, выявлять тренды и понять что разные платформы могут требовать разного контента.

Работа с Open Source инструментом делает этот процесс масштабируемым и полезным для других авторов и агентств.

**Автор:** Михаил Шардин

 [Моя онлайн-визитка](#)

 [Telegram «Умный Дом Инвестора»](#)



**Получи грант за код**

Конкурс open source проектов

**Хабы:** [Контент и копирайтинг](#), [Управление медиа](#), [Node.JS](#), [Open source](#)

## Редакторский дайджест



Присылаем лучшие статьи раз в месяц



Оставляя свою почту, я принимаю [Политику конфиденциальности](#) и даю согласие на получение рассылок

**212**

Карма

**63.3**

Рейтинг

**Михаил Шардин** [@empenoso](#)

[Автоматизация](#) / [Данные](#) / [Финансы](#) / [Умные дома](#)

[Подписаться](#)

[Сайт](#) [Сайт](#) [GitHub](#)



**Получи грант за код**

Конкурс open source проектов

 Комментарии 11

## Публикации

ЛУЧШИЕ ЗА СУТКИ

ПОХОЖИЕ

**Tirarex**

14 часов назад

### Как я делал сеть на 2,5 гигабита с минимальным бюджетом — апгрейд, доступный каждому



Простой



9 мин



11K

Тutorial



+41



83



34

**Erwinmal**

18 часов назад

### Сэндвич, сэр? История британских бутербродов от аристократических салонов до вокзальных буфетов



Простой



13 мин



3K

Ретроспектива



+41



20



2

**oneastok**

19 часов назад

### Умное зеркало на Raspberry Pi: пошаговое руководство

**Получи грант за код**

Конкурс open source проектов

 +22 56 14**iLushkersky**

14 часов назад

## Жизнь на Марсе? (снова)

**Простой**

3 мин



2.3K

 +17 5 10**TrexSelectel**

16 часов назад

## Nintendo Virtual Boy: неожиданное возрождение виртуальной реальности из 90-х



5 мин



1.1K

 +14 3 3**mio\_anni**

19 часов назад

## От мини-ЭВМ и перфокарт к IDE и фреймворкам. Как поменялось программирование за 50 лет — взгляд изнутри



12 мин



1.9K

 +12 15 35**RED\_OS\_M**

18 часов назад

## Станислав Петров: «Ключевые отличия РЕД ОС М от Android – вовсе не в интерфейсе»

**Средний**

8 мин



6.5K

**Интервью** +10 10 43**Получи грант за код**

Конкурс open source проектов

## Миф о быстром и медленном пути выполнения программы

 Средний  11 мин  1.5K

Обзор

Перевод

 +9

 16

 0



**kilokanat**

5 часов назад

## Механическая клавиатура LARKeyboard

 Простой  5 мин  518

Тutorial

 +8

 4

 2



**beeline\_cloud**

10 часов назад

## Научный «дипфейк»? Как галлюцинации нейросетей — и другие проблемы — просачиваются в академические статьи

 Простой  8 мин  886

Аналитика

 +8

 11

 2

## У нас было два админа, одна консоль, новый NGFW и более 50 сценариев тестирования

Турбо

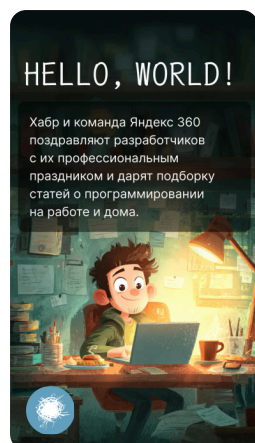
Показать еще

### ИСТОРИИ

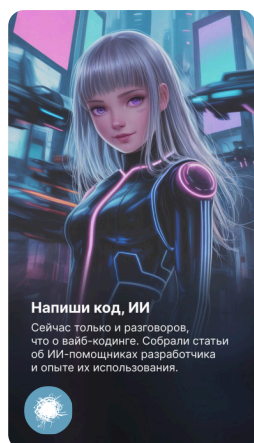


**Получи грант за код**

Конкурс open source проектов



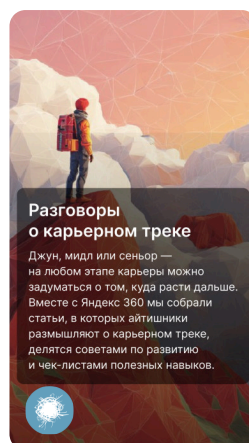
**Чай, торт и код: с Днём программиста!**



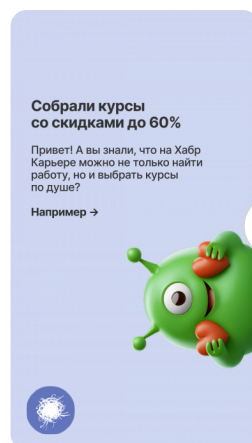
**Made in AI**



**Чего хотят лиды в бигтехе?**



**Как расти в IT: советы, гайды и опыт сеньоров**

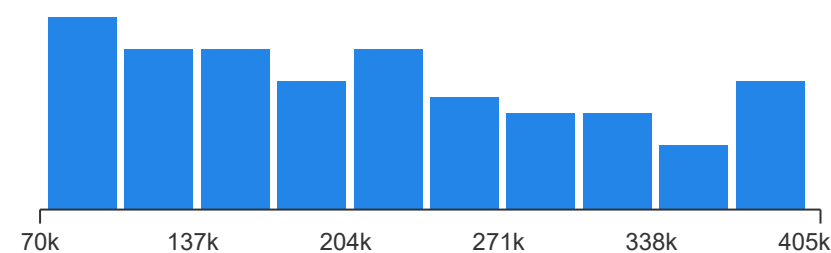


**Курсы со скидками до 60%**

## СРЕДНЯЯ ЗАРПЛАТА В IT

**214 724** ₽/мес.

— средняя зарплата во всех IT-специализациях по данным из 27 443 анкет, за 2-ое пол. 2025 года. Проверьте «в рынке» ли ваша зарплата или нет!



[Проверить свою зарплату](#)

## МИНУТОЧКУ ВНИМАНИЯ



**Получи грант за код**

Конкурс open source проектов

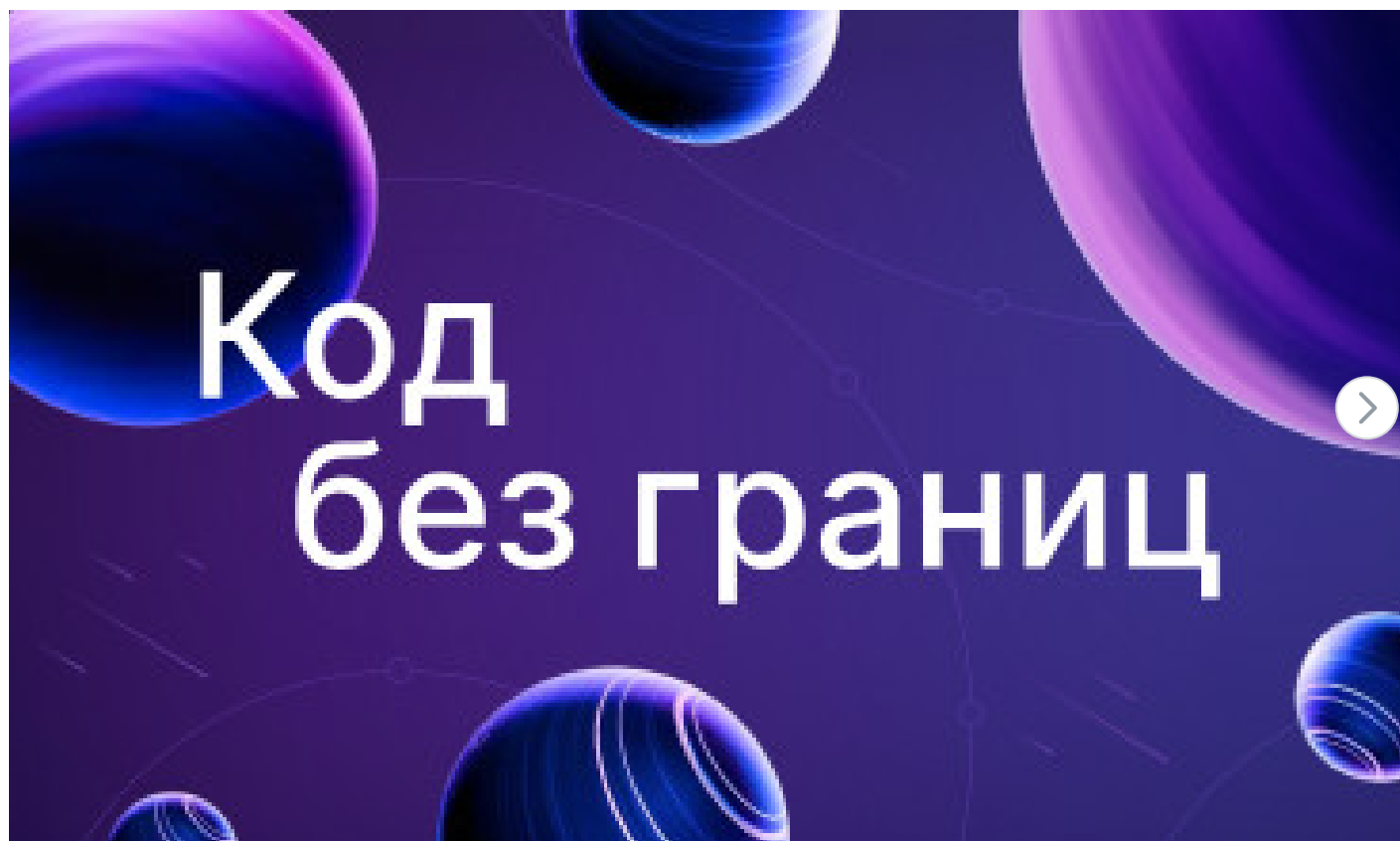


Bluetooth против плохой связи:  
кейс каршеринга

2 пилота и 50 сценариев: ИБ-  
команда тестирует NGFW

Цифровизация на максималках:  
чем живёт ИТ-пром

## БЛИЖАЙШИЕ СОБЫТИЯ



3 сентября – 31 октября

### Программа грантов для развития open source проектов «Код без границ»

Онлайн

Разработка

[Больше событий в календаре](#)



**Получи грант за код**

Конкурс open source проектов



 [Настройка языка](#)

[Техническая поддержка](#)

© 2006–2025, Habr



**Получи грант за код**

Конкурс open source проектов