**[Tproger](https://tproger.ru/)**

[Tproger](https://tproger.ru/)

* [Начинающим](https://tproger.ru/tag/for-beginners/)
* [Алгоритмы](https://tproger.ru/tag/algorithms/)
* [Планы обучения](https://tproger.ru/category/curriculum/)
* [Собеседования](https://tproger.ru/tag/job-interview/)
* [Web](https://tproger.ru/tag/web/)
* [JS](https://tproger.ru/tag/javascript/)
* [Python](https://tproger.ru/tag/python/)
* [С++](https://tproger.ru/tag/cpp/)
* [Java](https://tproger.ru/tag/java/)
* [**Все темы**](https://tproger.ru/articles/javascript-2018-survey/#tagsModal)

[[https://cdn.tproger.ru/wp-images/tg_header.png](tg://resolve/?domain=tproger_official)Го в Телеграм, мы создали](tg://resolve?domain=tproger_official)

StateOfJS: обзор масштабного опроса JavaScript-разработчиков по итогам 2018 года

* 11 января 2019 в 13:42, [Статьи](https://tproger.ru/category/articles/)
* 12 минут
* 2752



[designed by Vecteezy](https://www.vecteezy.com/)

В рамках [State of JavaScript](http://stateofjs.com/" \t "_blank) разработчики рассказывают, чем они пользуются, с чем им нравится работать и что они хотят изучить. В итоге получается уникальная статистика с некоторыми деталями, которые, возможно, помогут вам найти свой путь в экосистему JavaScript.

В прошлом году мы [публиковали](https://tproger.ru/translations/javascript-trends-2018/) результаты опроса программистов, работающих с JavaScript, в нём предсказывались тренды на 2018 год. Если интересно, сравните, что сбылось.

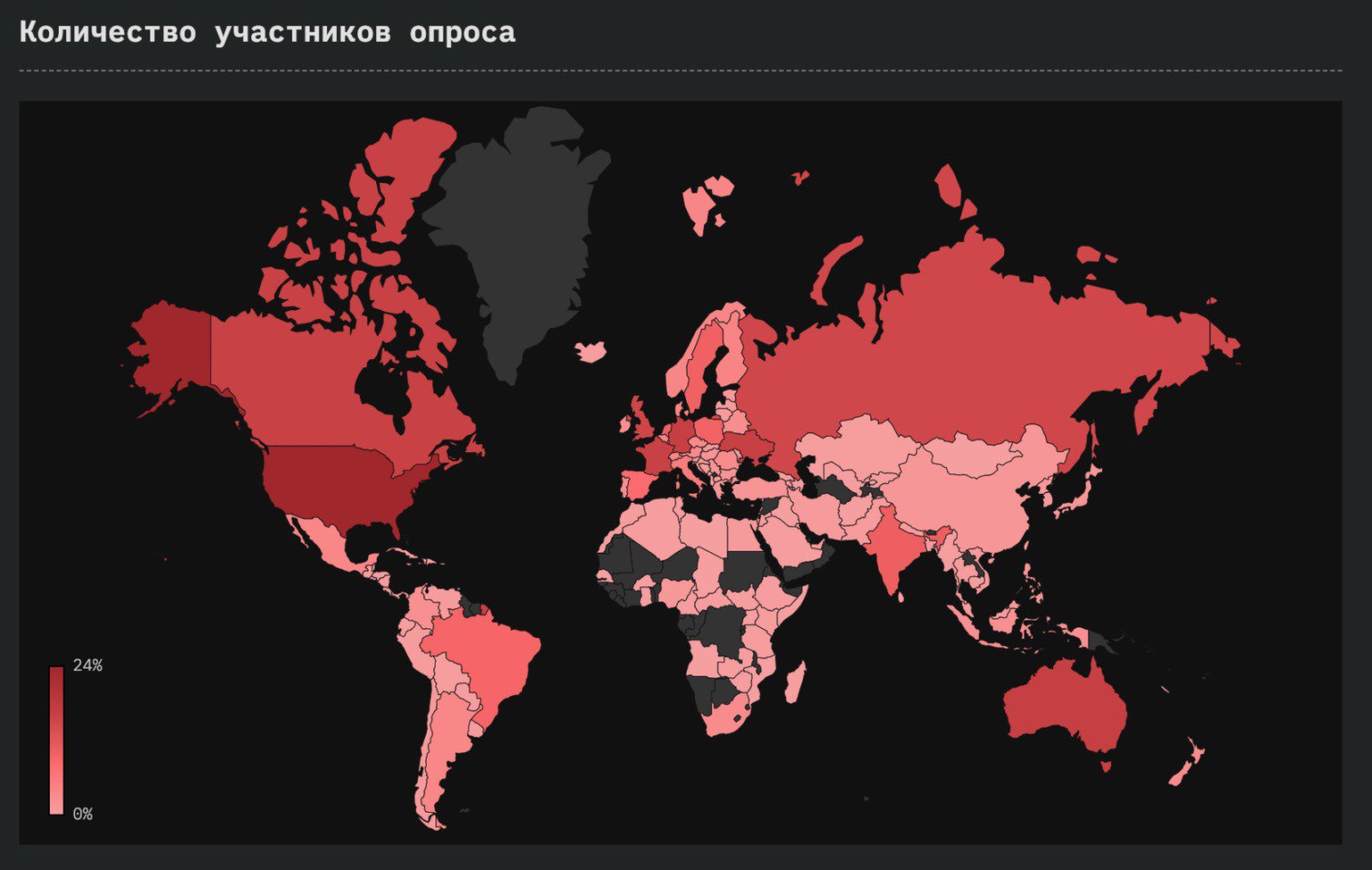
Содержание статьи

1. [Общая информация](https://tproger.ru/articles/javascript-2018-survey/#generalInfo).
2. [«Оттенки» JavaScript](https://tproger.ru/articles/javascript-2018-survey/#JavaScriptFlavors).
3. [Обзор фронтенд-фреймворков](https://tproger.ru/articles/javascript-2018-survey/#frontEndOverview).
4. [Работа с данными и JavaScript](https://tproger.ru/articles/javascript-2018-survey/#DataJS).
5. [Обзор бэкенд-фреймворков](https://tproger.ru/articles/javascript-2018-survey/#backEndOverview).
6. [Тестирование](https://tproger.ru/articles/javascript-2018-survey/#testing).
7. [Обзор фреймворков для создания десктопных и мобильных приложений](https://tproger.ru/articles/javascript-2018-survey/#desktopAndMobile).
8. [Другие инструменты](https://tproger.ru/articles/javascript-2018-survey/#otherTools).
9. [Победители гонки технологий прошедшего года](https://tproger.ru/articles/javascript-2018-survey/#specialAwards).
10. [Вывод](https://tproger.ru/articles/javascript-2018-survey/#theEnd).

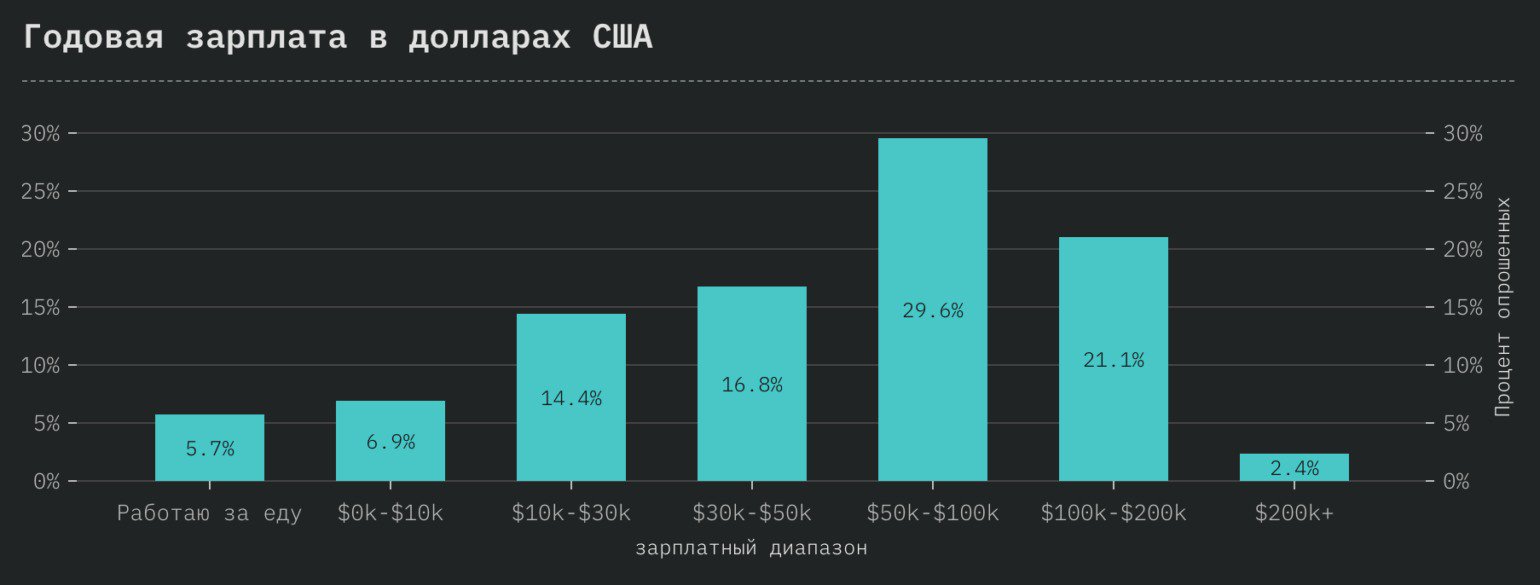
Общая информация

В этот раз опросили**более 20 000 разработчиков из 153 стран**.

Авторы опроса привели данные только тех людей, которые согласились на опрос и прошли его полностью. Реальное количество JS-разработчиков в мире, естественно, будет отличаться. Больше всего откликов на опрос пришло из США, Германии и Австралии. Россия и Украина тоже участвовали достаточно активно.

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

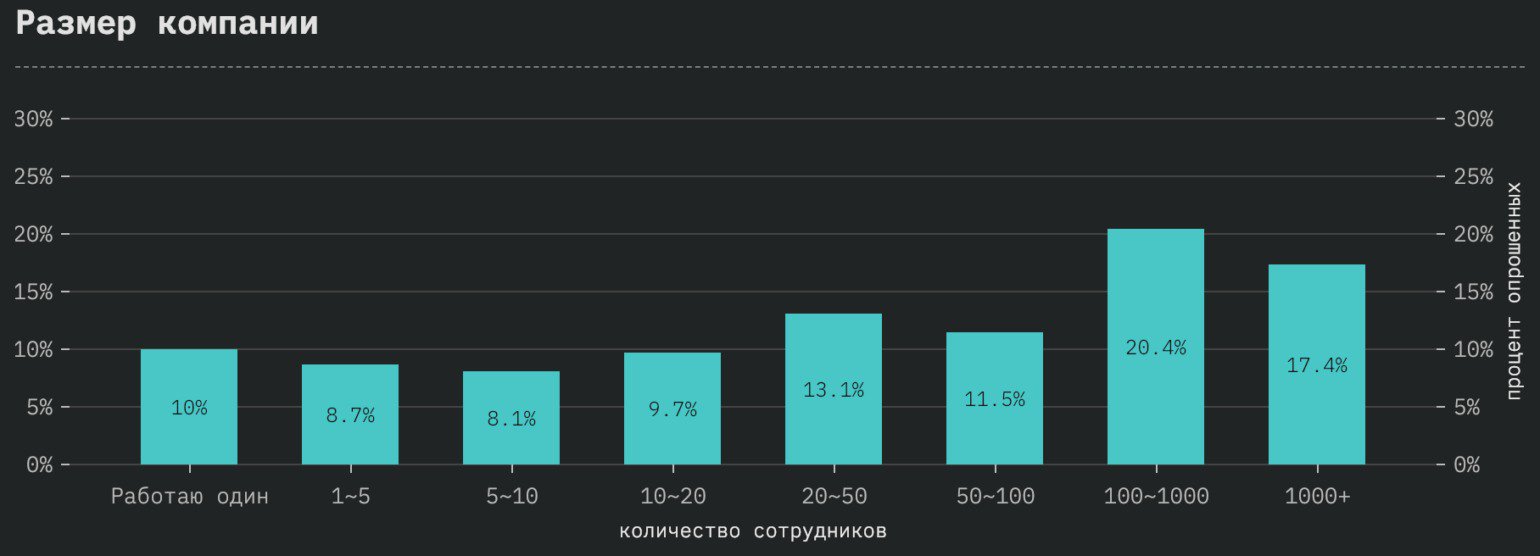
Более 50 % разработчиков зарабатывают от 50 000 $ в год:

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/js-devs-salary.jpg)

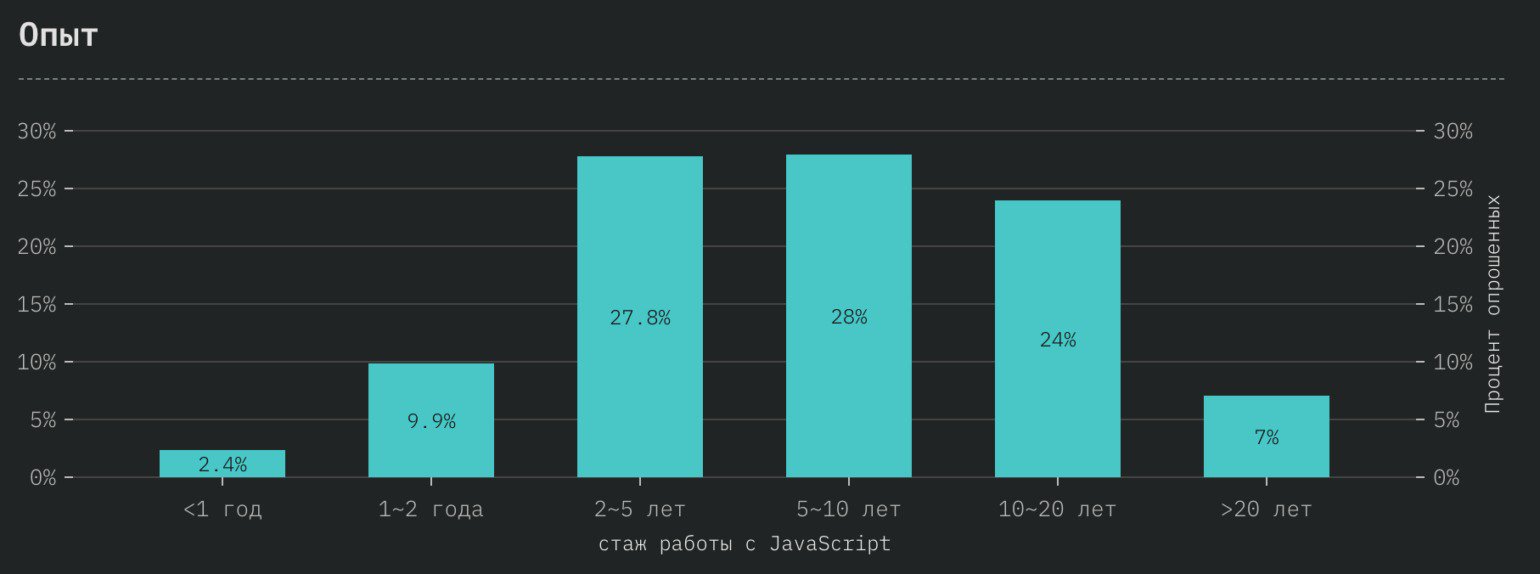
Самые высокие зарплаты получают разработчики из США (116 700 $ в год), Австралии (109 100 $ в год), и Швейцарии (97 300 $ в год). В России этот показатель равен 25 000 $ в год, что ниже, чем во всех других странах бывшего СССР, отмеченных на карте.

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

Большинство разработчиков работает на компании, в которых более 20 сотрудников:

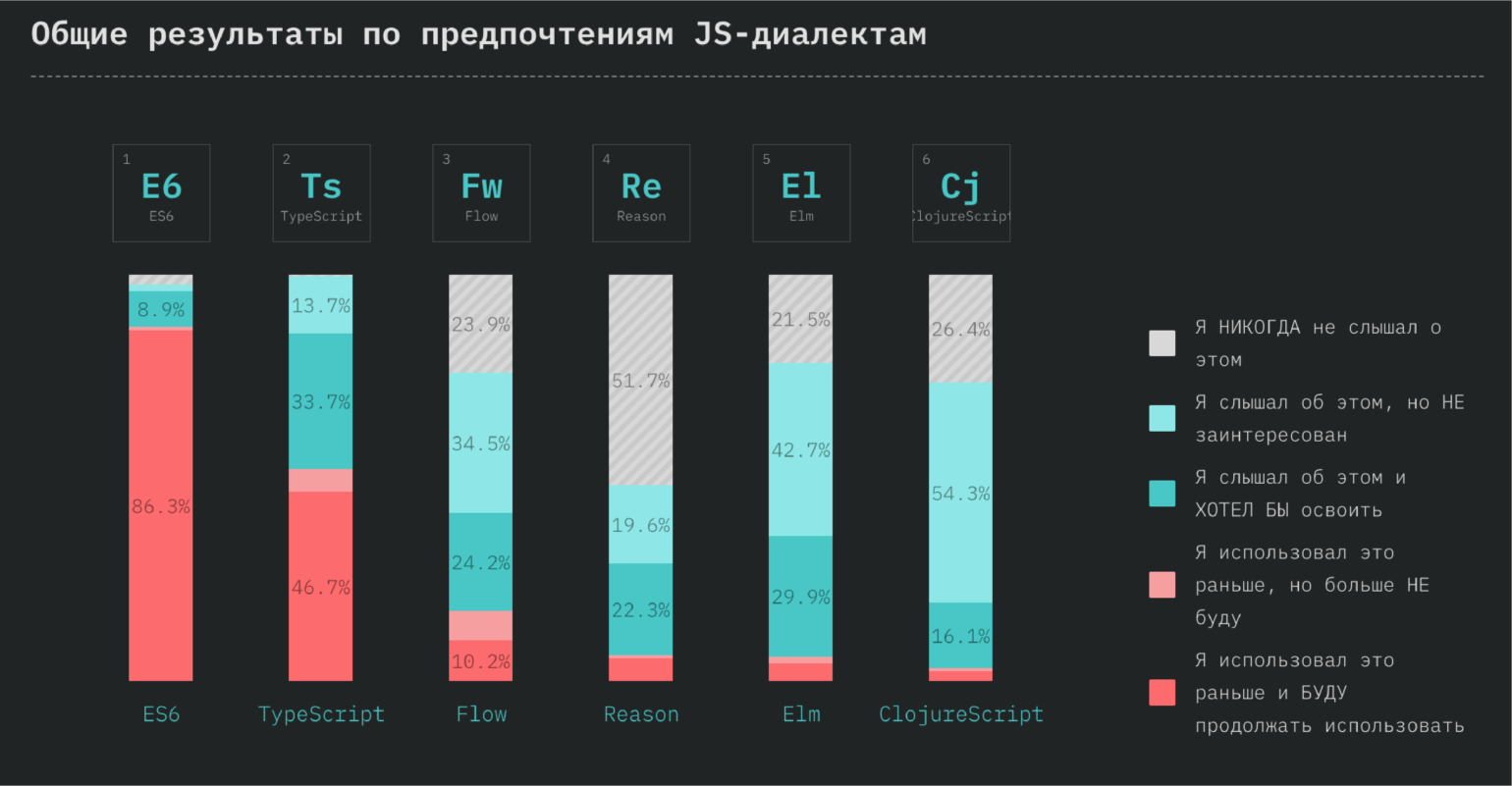
[](https://cdn.tproger.ru/wp-content/uploads/2019/01/Snimok-jekrana-2019-01-05-v-14.05.44.jpg)

Более 50 % участников опроса работают с JavaScript более 5 лет:

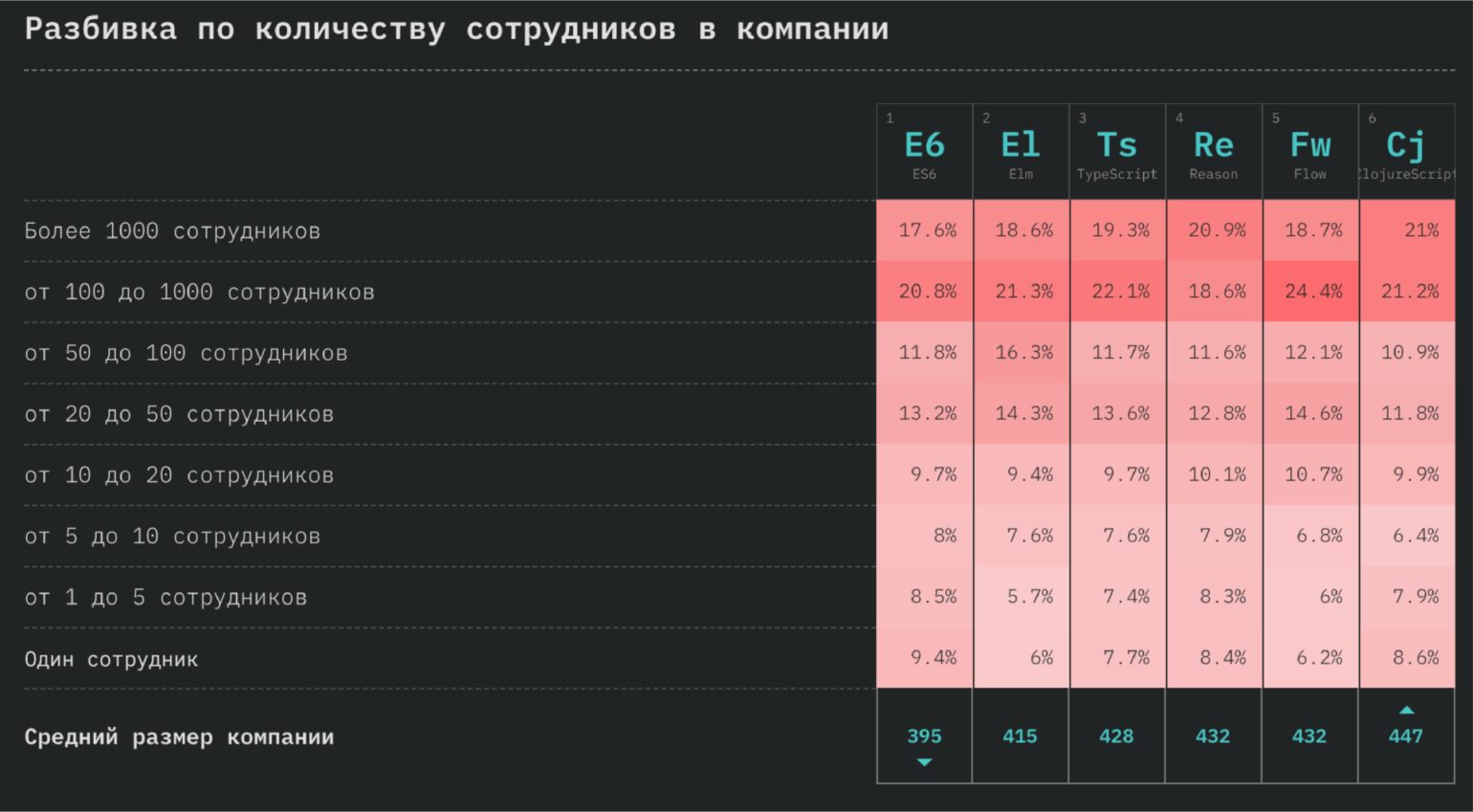
[](https://cdn.tproger.ru/wp-content/uploads/2019/01/js-devs-xp-general.jpg)

«Оттенки» JavaScript

JavaScript становится зрелым, и разработчики выходят за его границы, создавая новые языки и диалекты, которые компилируются в JavaScript. В рамках опроса авторы сравнивают ES6, TypeScript, Flow, Reason, Elm и ClojureScript.

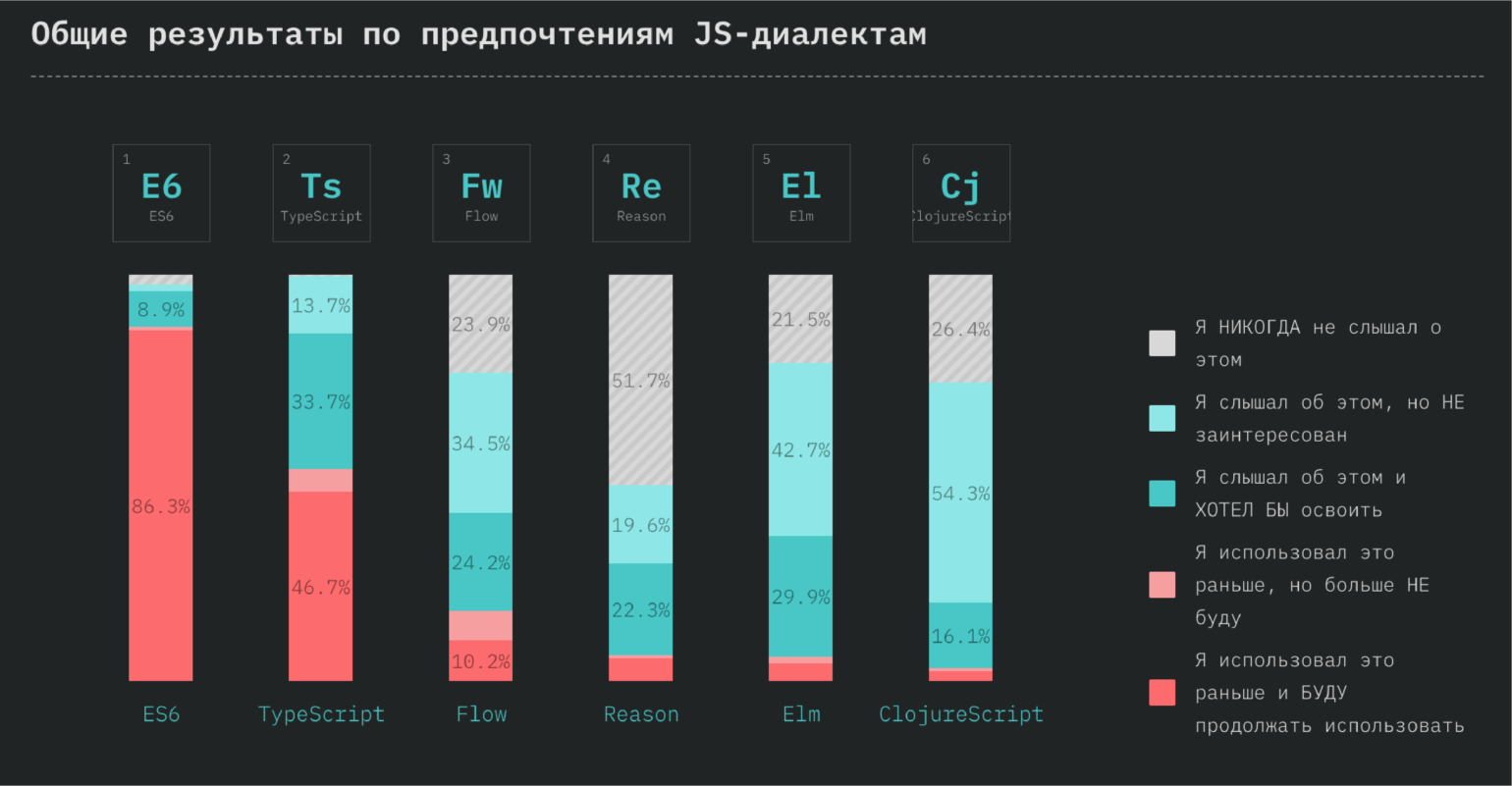
[](https://cdn.tproger.ru/wp-content/uploads/2019/01/overall-results.png)

В компаниях размером около 500 сотрудников активно используют статический анализатор Flow:

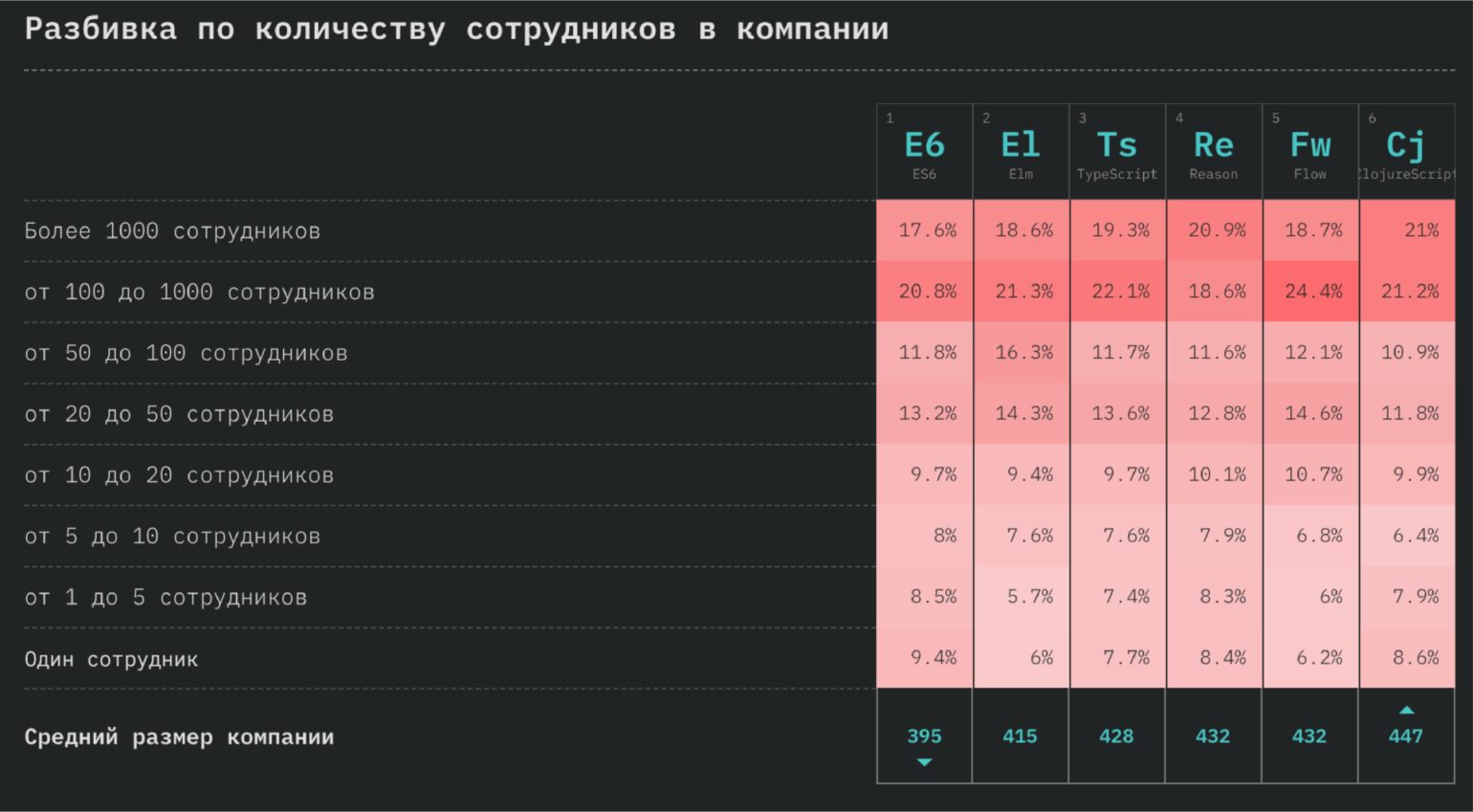
[](https://cdn.tproger.ru/wp-content/uploads/2019/01/headcount-js-flavors-2.png)

## «Оттенки» JavaScript

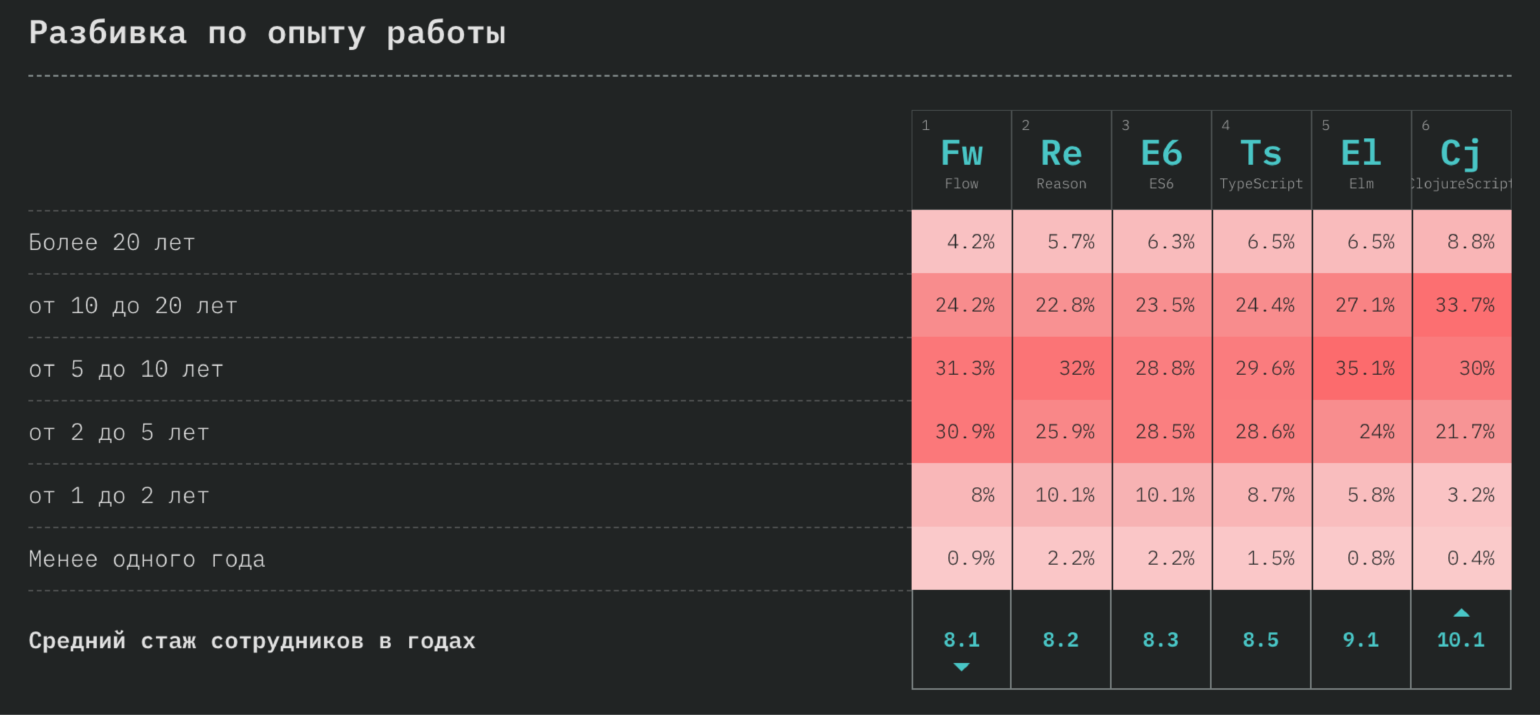
JavaScript становится зрелым, и разработчики выходят за его границы, создавая новые языки и диалекты, которые компилируются в JavaScript. В рамках опроса авторы сравнивают ES6, TypeScript, Flow, Reason, Elm и ClojureScript.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/overall-results.png)

### В компаниях размером около 500 сотрудников активно используют статический анализатор Flow:

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/headcount-js-flavors-2.png)

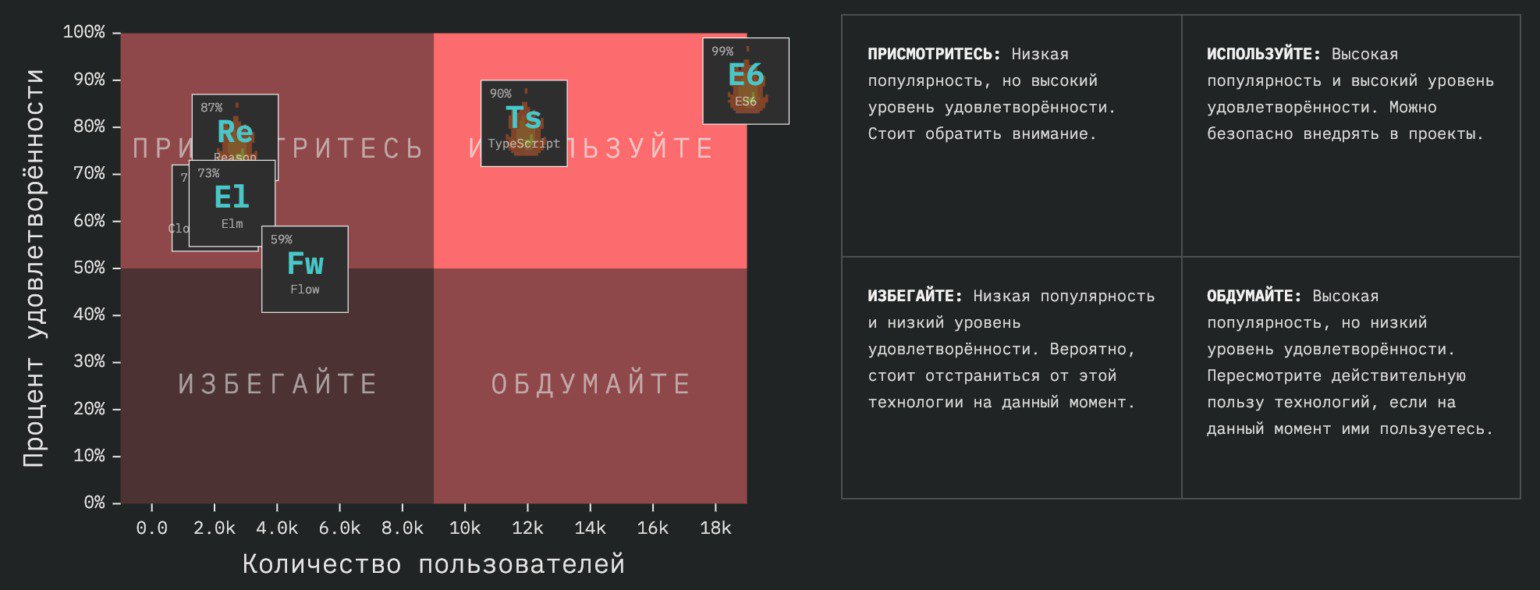
### Разработчики со стажем, который приближается к 10 годам, активно используют Elm — функциональный язык для веб-разработки:

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/opyt-raboty-2.png)

### Что выбирать?

Диаграмма ниже отображает **степень удовлетворённости** технологией в соотношении с её **популярностью**. Кроме того, если технологию хотят изучить более 50 % участников опроса, она помечается огнём.

К Elm, ClojureScript (это он за ним прячется) и Reason имеет смысл присмотреться, а вот Flow на грани фола:

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/Final-chart-js-devs-flavors.jpg)

CoffeeScript долгое время был единственным сторонником подхода компиляции в JavaScript, но сегодня его обогнали ES6 и TypeScript, Flow и даже языки с совсем другим синтаксисом вроде Elm и Reason.

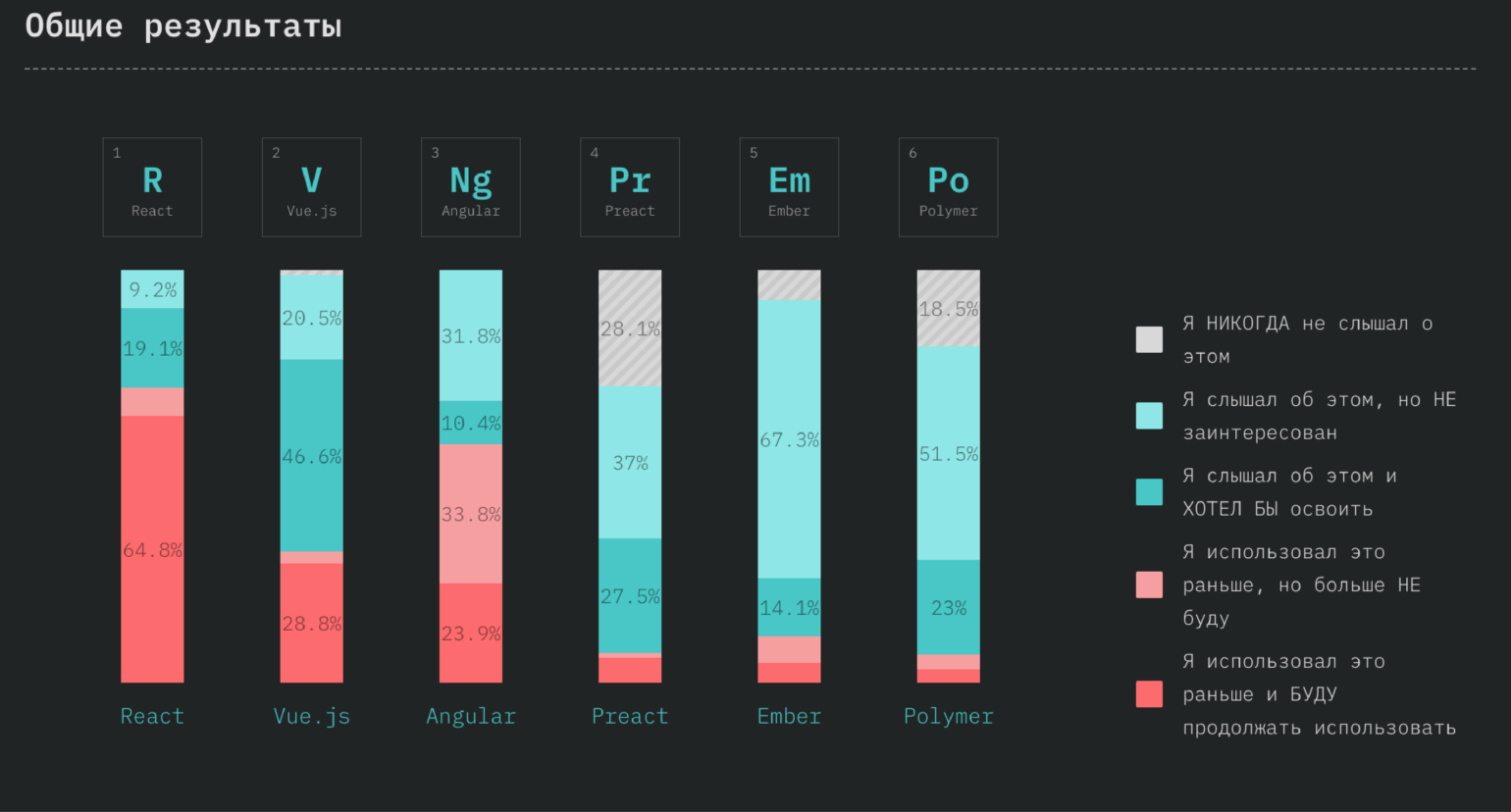
Есть веские основания полагать, что за диалектами и компилируемыми в JavaScript языками стоит его же будущее. С появлением таких проектов, как WebAssembly, написание кода напрямую на JavaScript может скоро показаться странным, так как разработчики активно охватывают такие языки, как Rust.

Время ES6 и TypeScript ещё не ушло — в них спокойно можно нырять. Следите также за Reason, который может похвастаться одновременно высокой популярностью и высоким уровнем удовлетворённости разработчиков.

## Обзор фронтенд-фреймворков

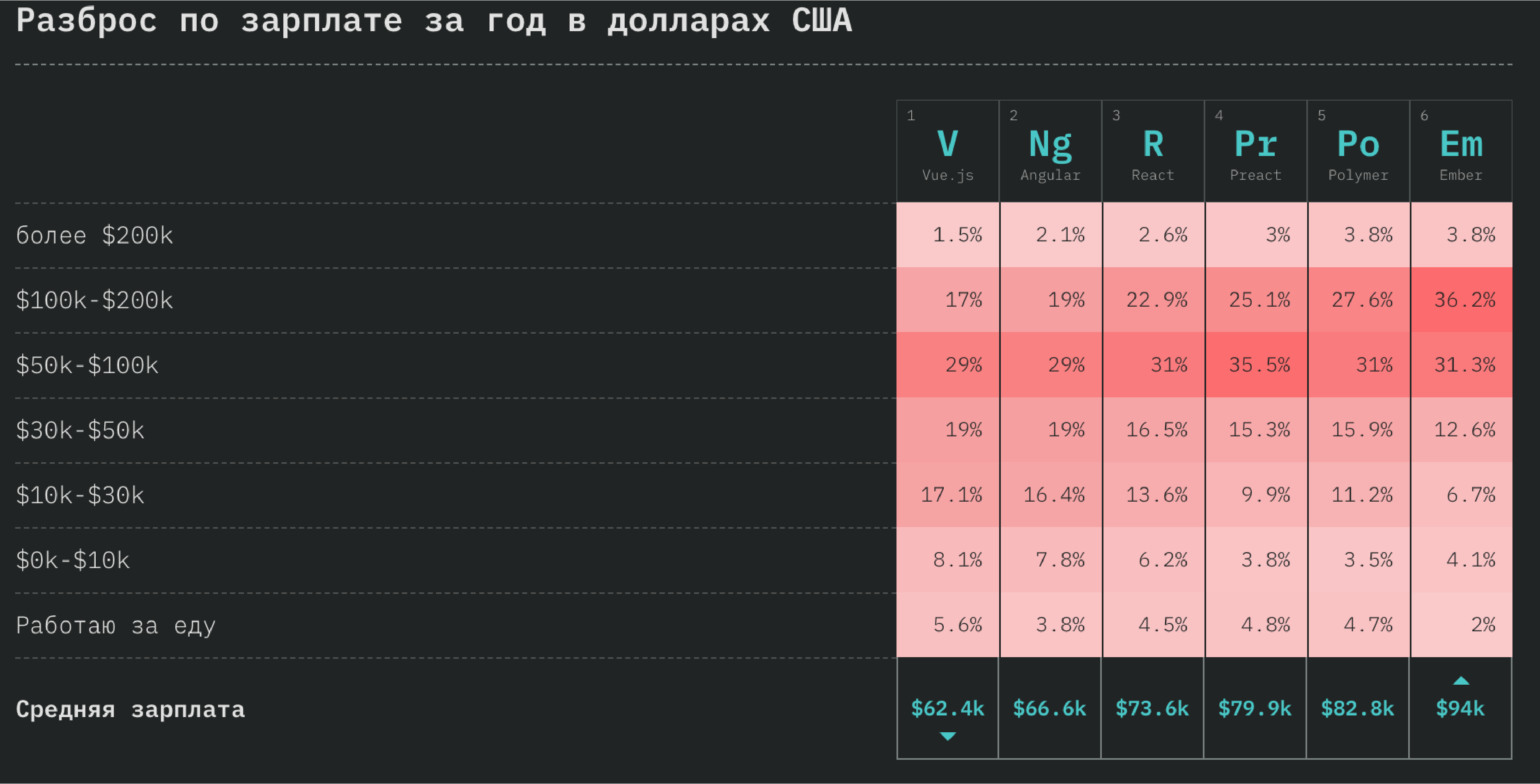
### Введение

Фронтенд остаётся оживлённым полем боя среди фреймворков. Но если присмотреться, то станет понятно, что среди всех есть два лидера, от которых все давно отстали (да-да, речь идёт о React и Vue.js).

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/overall-results-js-devs-fend.png)

При этом можно сделать интересные наблюдения о зарплатах тех, кто постоянно использует какую-то технологию и собирается продолжать это делать.

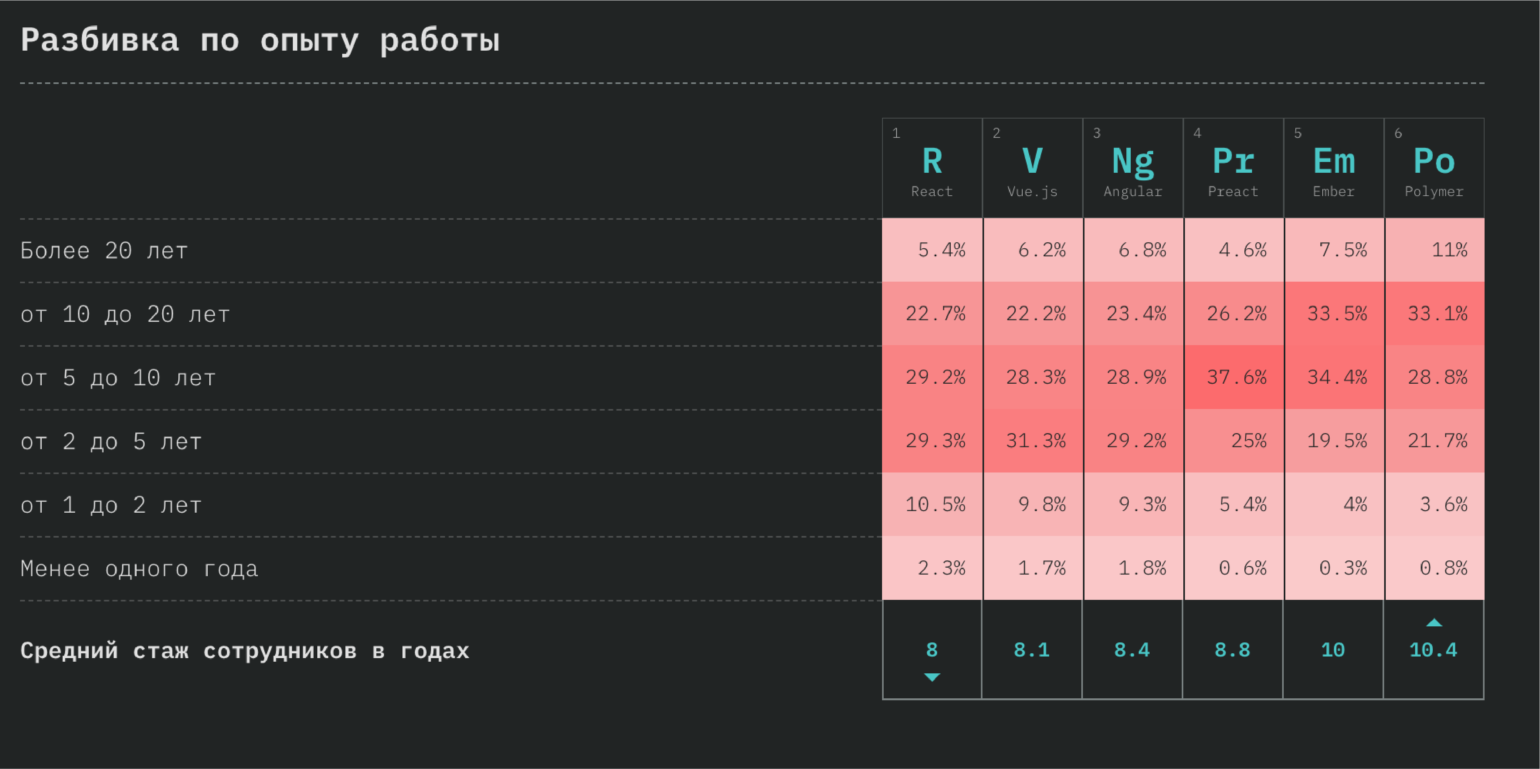
### Среди людей со средним уровнем дохода около 100 000 $ в год популярен фреймворк Ember:

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/salary-js-devs-fend.png)

### В больших компаниях отдают предпочтение Polymer:

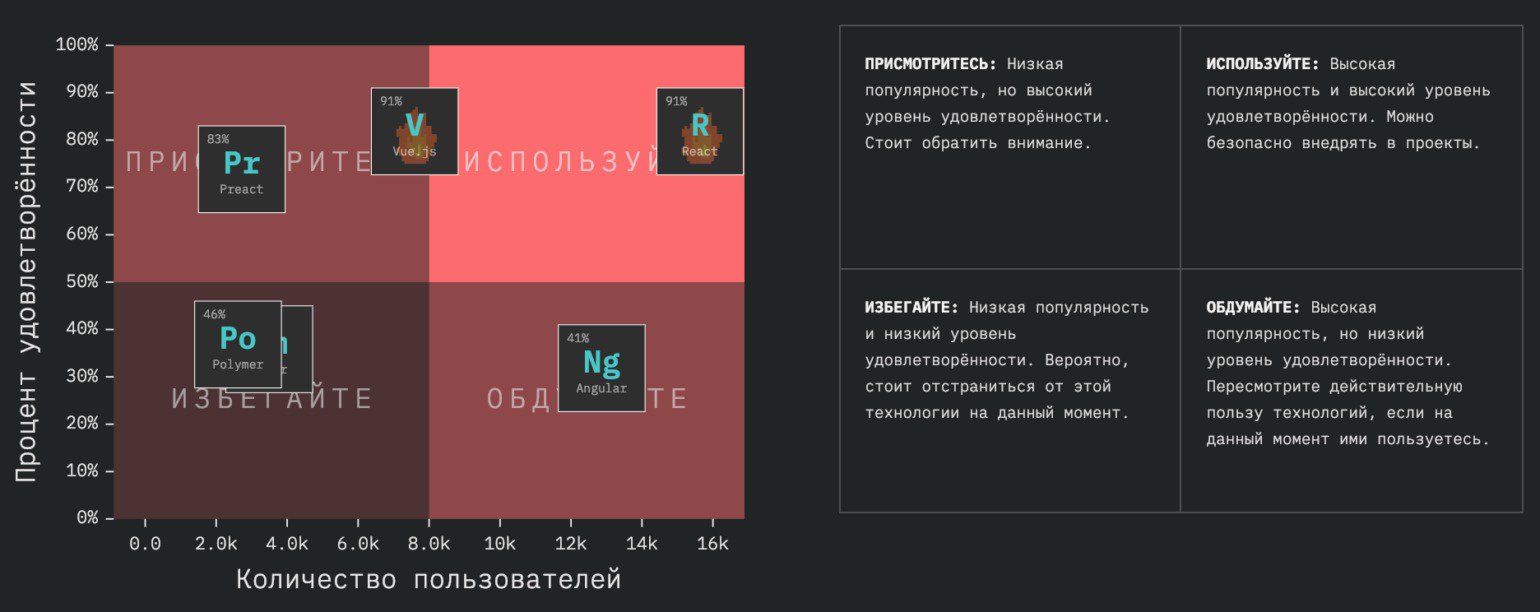
[](https://cdn.tproger.ru/wp-content/uploads/2019/01/fend-js-devs-headcount-1.png)

### Preact популярен среди разработчиков со стажем около 8 лет:

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/xp-js-devs-fend-2.png)

### Давайте лучше выводы — что сейчас в тренде?

Диаграмма ниже отображает **степень удовлетворённости** технологией в соотношении с её **популярностью**. Кроме того, если технологию хотят изучить более 50 % участников опроса, она помечается огнём.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/front-js-devs-final-diagram.jpg)

Как видно по статистике выше, бóльшая часть «власти» во фронтенде принадлежит React и Vue.js. Особенно стоит обратить внимание на Vue.js: ещё два года назад о нём не знали 27% респондентов. В 2018-м их количество сократилось до 1,3 %!

У React пока что по-прежнему устойчивая позиция, и он занимает гораздо бóльшую часть рынка, нежели Vue.js. Но рост последнего мало что может остановить. Более того, Vue уже набрал больше звёзд на GitHub, чем React.

С Angular всё плохо — несмотря на то, что он до сих пор популярен, более 58 % разработчиков отзываются о нём негативно. Тем не менее, он до сих пор остаётся на арене фронтенда из-за своей пользовательской базы. Стоит отметить, что респонденты могли перепутать Angular с AngularJS 🙂

## Работа с данными и JavaScript

### Введение

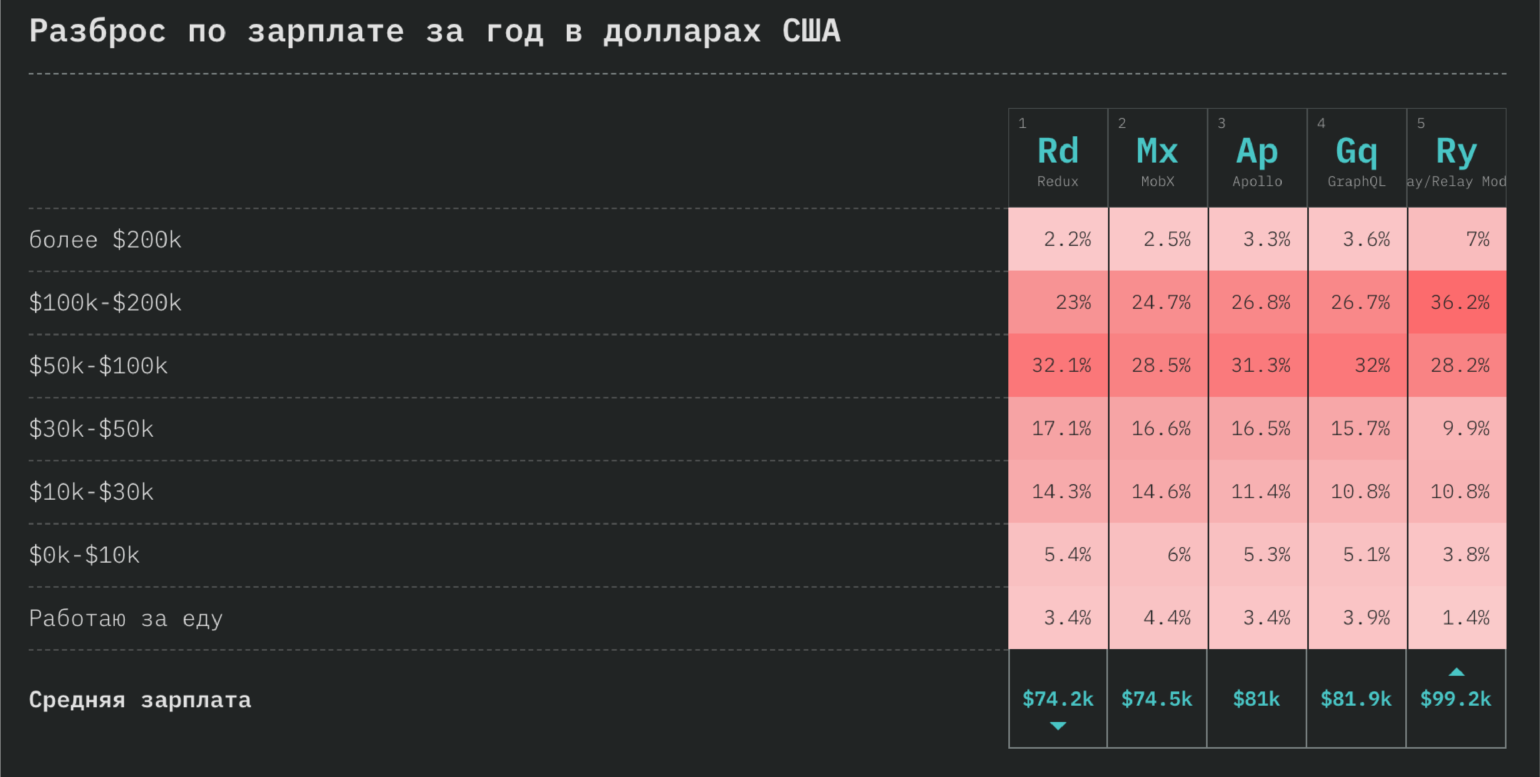
Все технологии, используемые для передачи и управления данными, на этом уровне объединяются. Это обширная категория, внутри которой многие подходы конкурируют в попытке уменьшить возникающие в работе сложности.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/data-js-devs-overall-results.png)

### Зарплаты

Разброс в зарплате разработчиков, выбравших пункт «Я использовал это раньше и буду продолжать использовать».

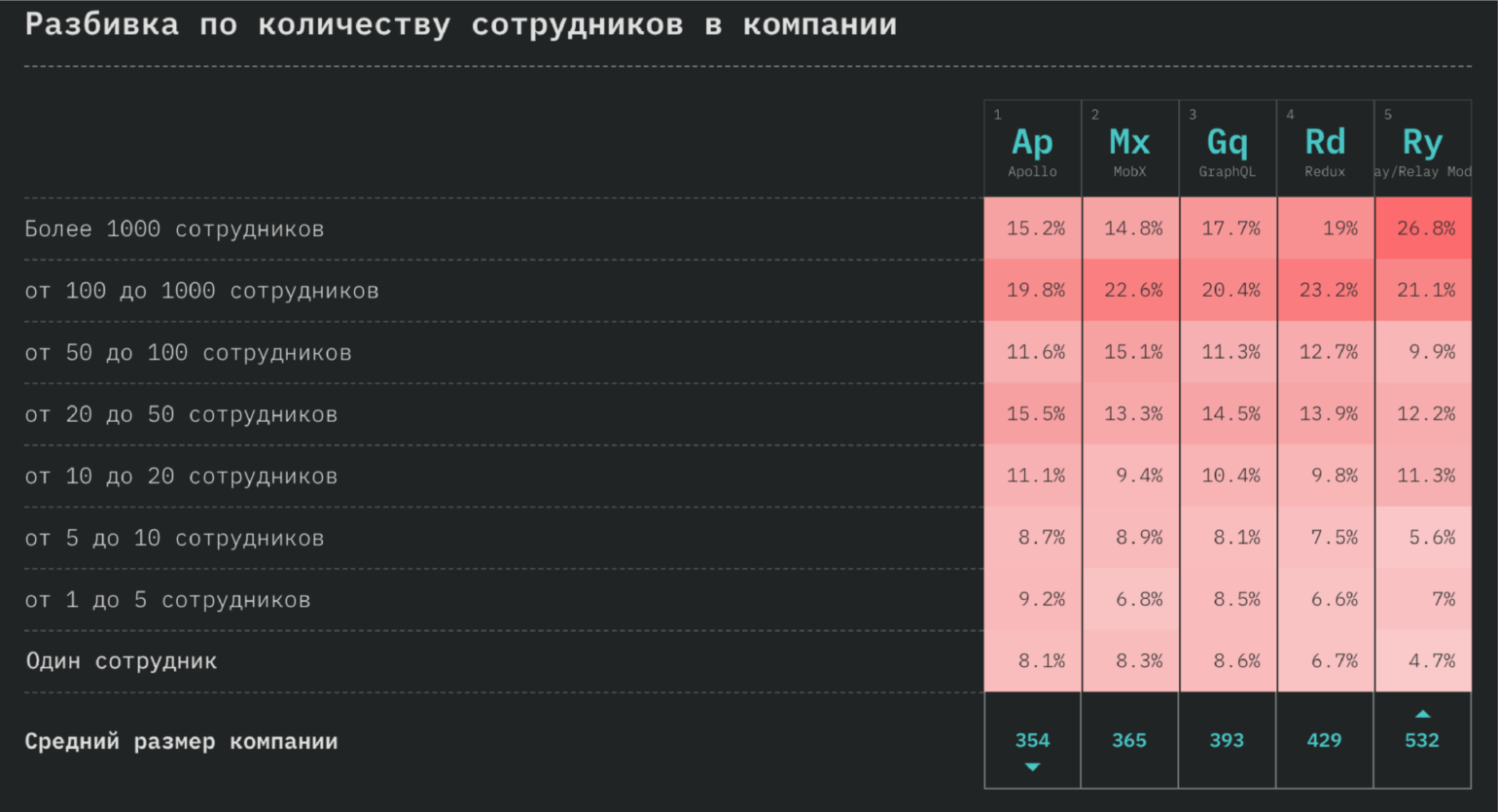
В каждой ячейке отображается процент пользователей в выбранной зарплатной категории. Чем насыщеннее красный, тем больше используется технология в данной категории зарплат.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/data-js-devs-salary.png)

### Размер компании

Разбивка разработчиков, выбравших пункт «Я использовал это раньше и буду продолжать использовать» по размеру компании.

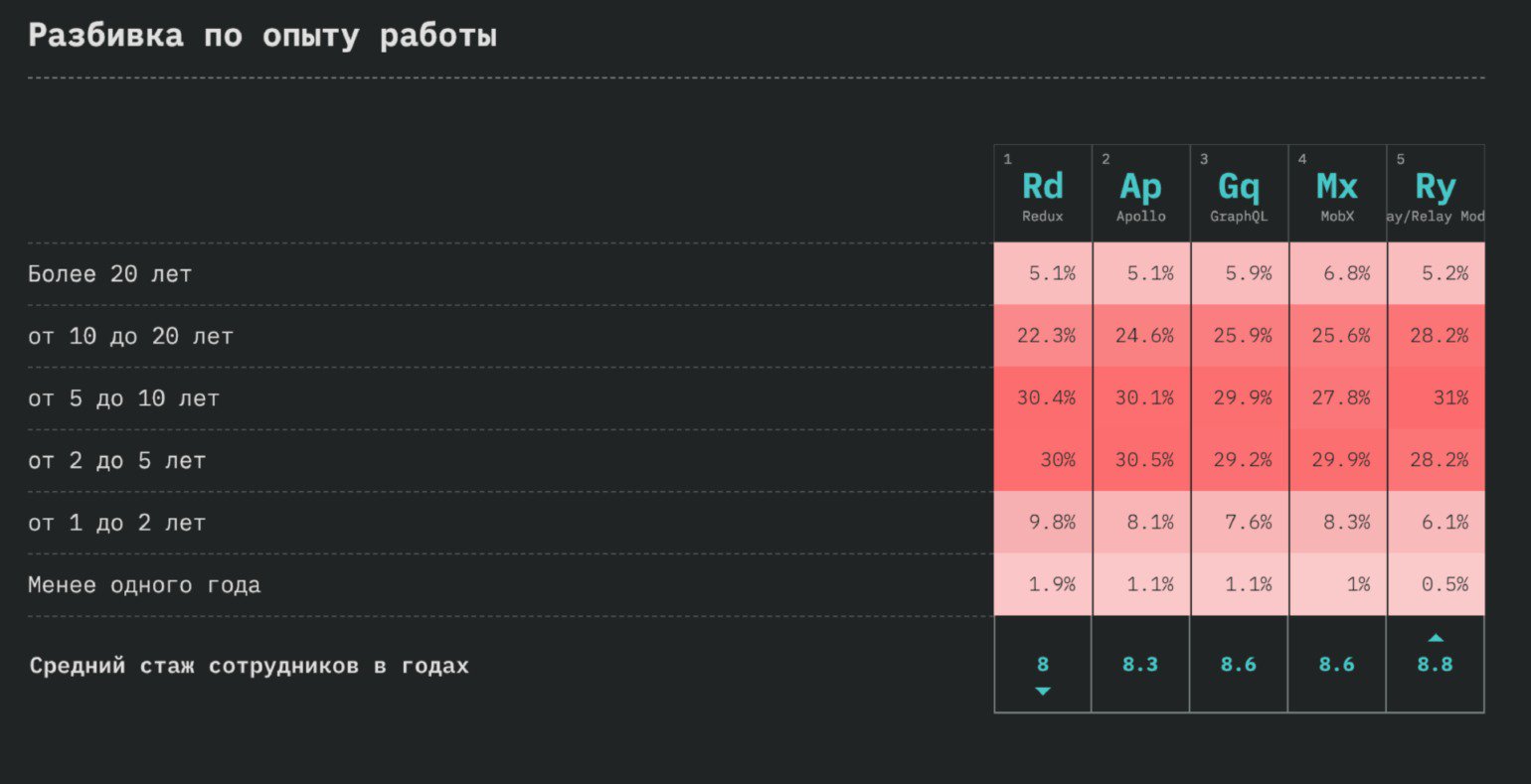
В каждой ячейке отображается процент пользователей в выбранной категории компаний. Чем насыщеннее красный, тем больше используется технология в данной категории.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/data-js-devs-headcount-3.png)

### Опыт

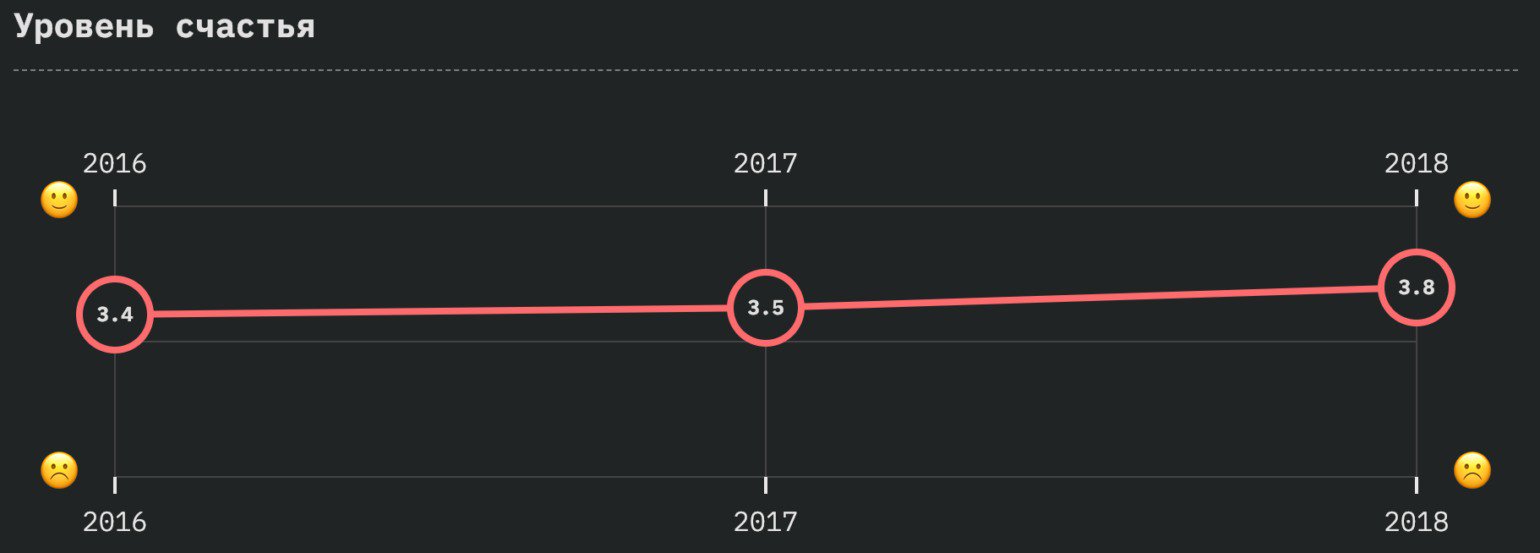
Разбивка по опыту работы для разработчиков, выбравших пункт «Я использовал это раньше и буду продолжать использовать».

В каждой ячейке отображается процент пользователей в выбранной категории стажа. Чем насыщеннее красный, тем популярней технология среди пользователей данной категории стажа.

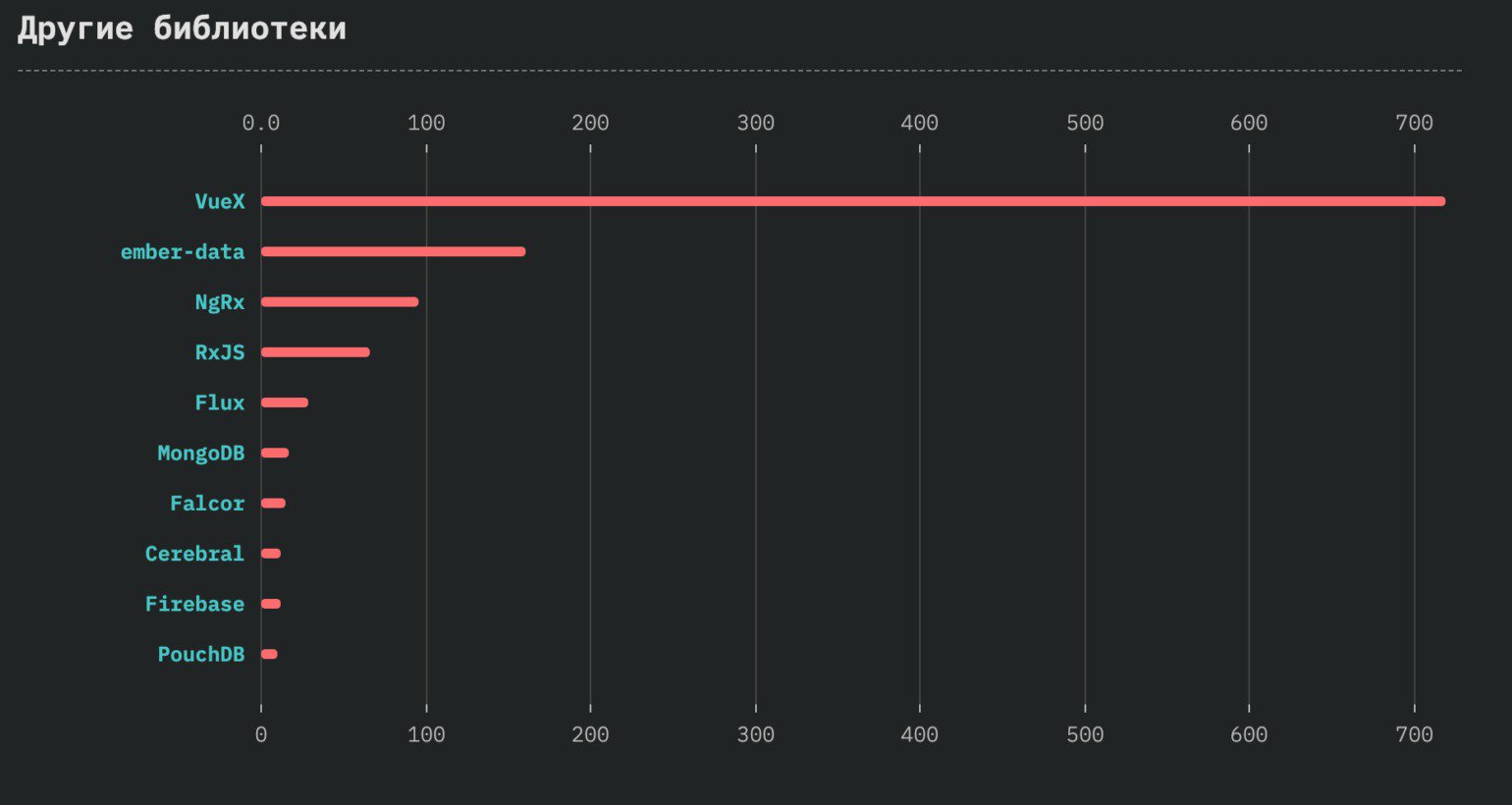
[](https://cdn.tproger.ru/wp-content/uploads/2019/01/data-js-devs-xp-1.jpg)

### Удовлетворённость

Насколько довольны разработчики текущим положением дел в этой категории.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/data-js-devs-happiness.jpg)

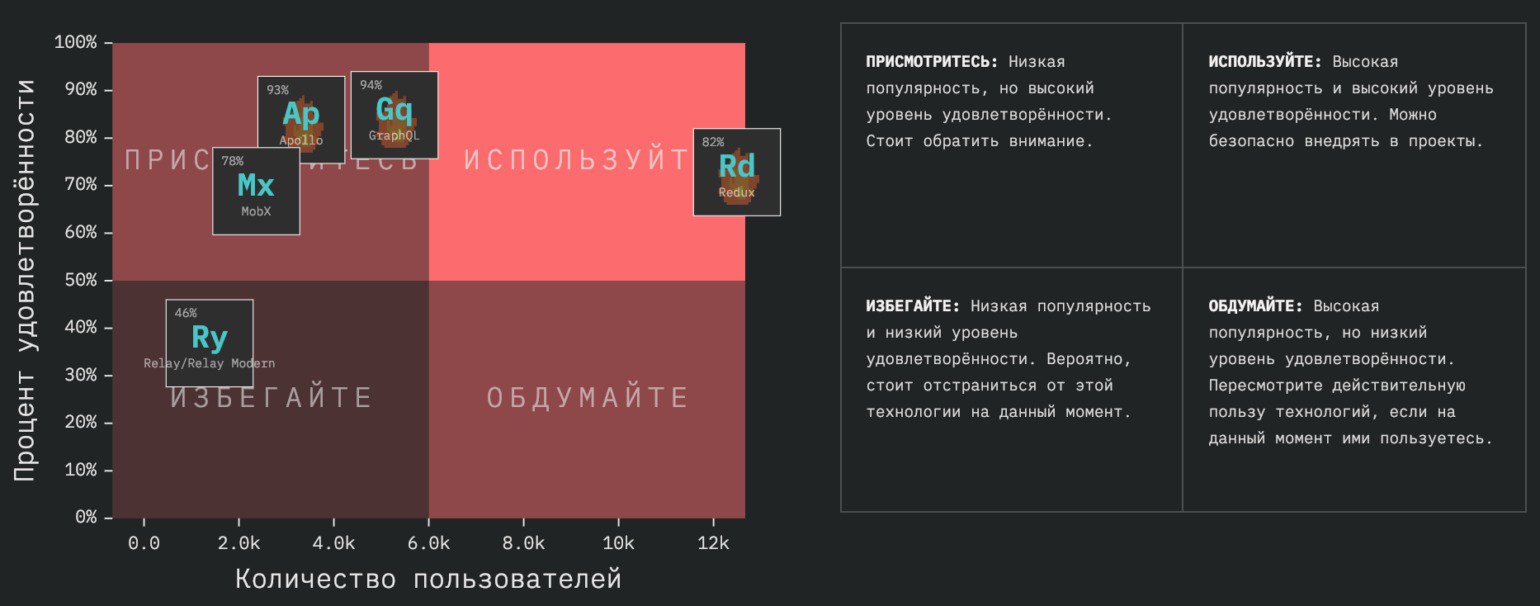
### Список других используемых библиотек

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/data-js-devs-otherLibs.jpg)

### Финальная диаграмма

Диаграмма ниже отображает **степень удовлетворённости** технологией в соотношении с её **популярностью**.

Кроме того, если технологию хотят изучить более 50 % участников опроса, она помечается огнём.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/data-js-devs-final-diagram.jpg)

### Заключение

Раньше всё было проще. Данные находились в базе данных, из которой они извлекались сервером, добавлялись в шаблон и отправлялись клиенту.

Сейчас же всё сложнее. В наше время приложения должны знать, как самостоятельно извлекать данные для обработки собственных шаблонов и компонентов. Так было положено начало целому ряду инструментов, служащих для извлечения и управления данными.

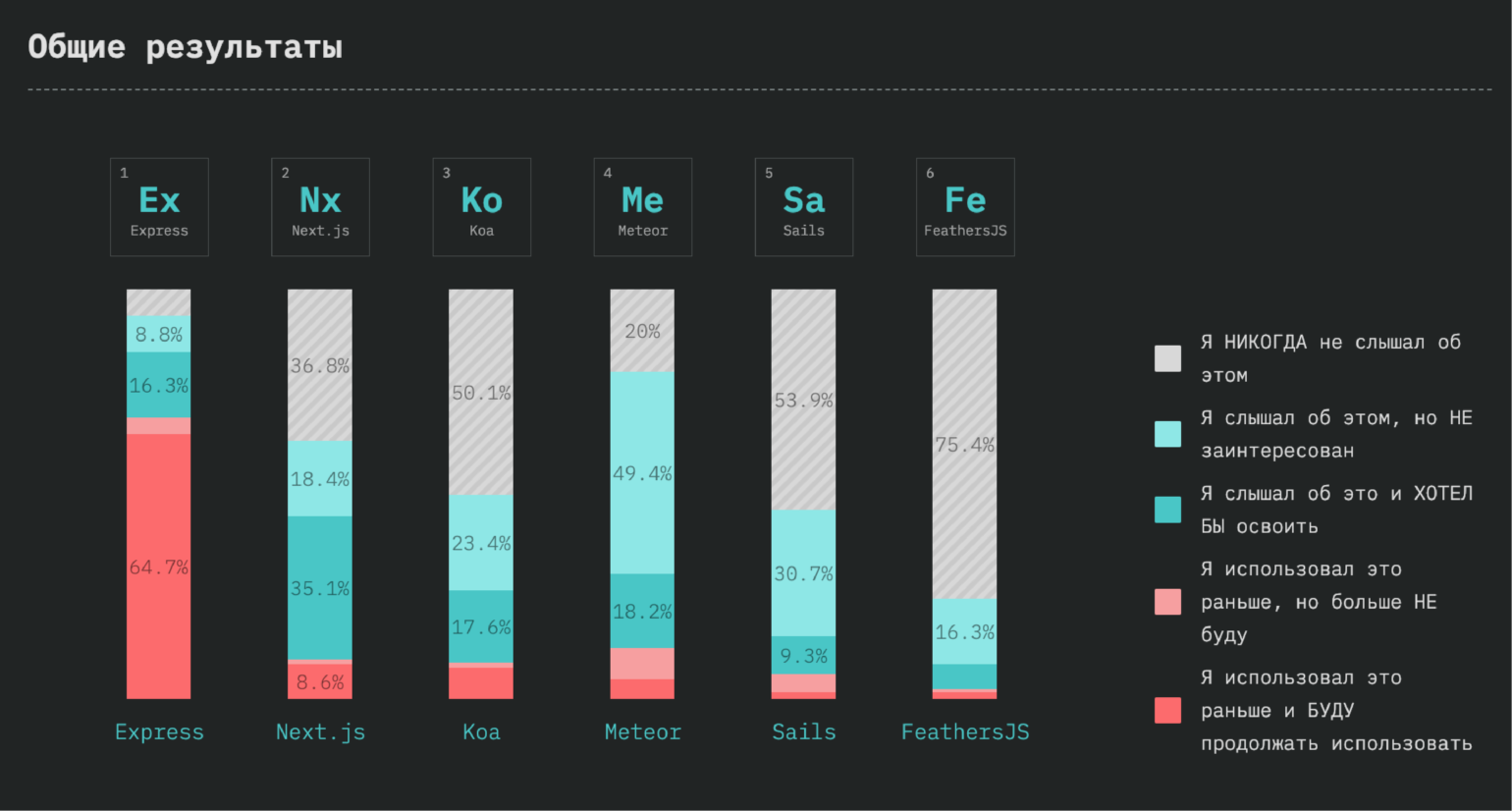
Redux — несомненный лидер среди инструментов для работы с данными: он самый распространённый и самый любимый среди остальных. 82 % разработчиков удовлетворены работой с ним.

Хотя Redux и лидер на данный момент, нельзя не заметить рост количества пользователей GraphQL с 5 % до 20 % за два года. Кроме того, внимание GraphQL, судя по всему, обращено на Apollo — замене Redux при работе с React.

## Обзор бекенд-фреймворков

### Введение

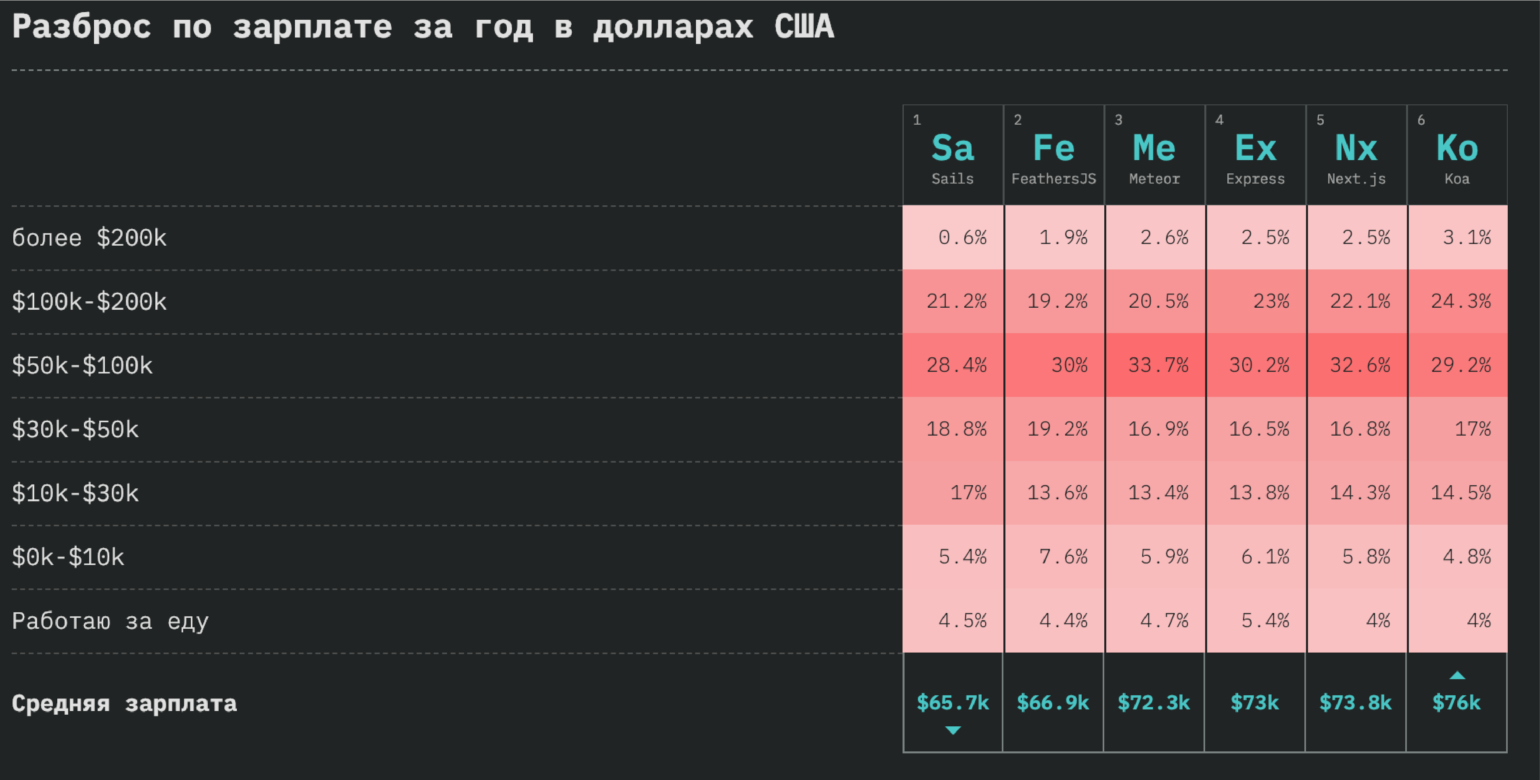
В серверном JavaScript за последние годы не было каких-либо значимых прорывов.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/back-js-devs-overall-results-1.png)

### Зарплаты

Разброс в зарплате разработчиков, выбравших пункт «Я использовал это раньше и буду продолжать использовать».

В каждой ячейке отображается процент пользователей в выбранной зарплатной категории. Чем насыщеннее красный, тем больше используется технология в данной категории зарплат.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/back-js-devs-salary.png)

### Размер компании

Разбивка разработчиков, выбравших пункт «Я использовал это раньше и буду продолжать использовать» по размеру компании.

В каждой ячейке отображается процент пользователей в выбранной категории компаний. Чем насыщеннее красный, тем больше используется технология в данной категории.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/back-js-devs-headcount.jpg)

### Опыт

Разбивка по опыту работы для разработчиков, выбравших пункт «Я использовал это раньше и буду продолжать использовать».

В каждой ячейке отображается процент пользователей в выбранной категории стажа. Чем насыщеннее красный, тем популярней технология среди пользователей данной категории стажа.

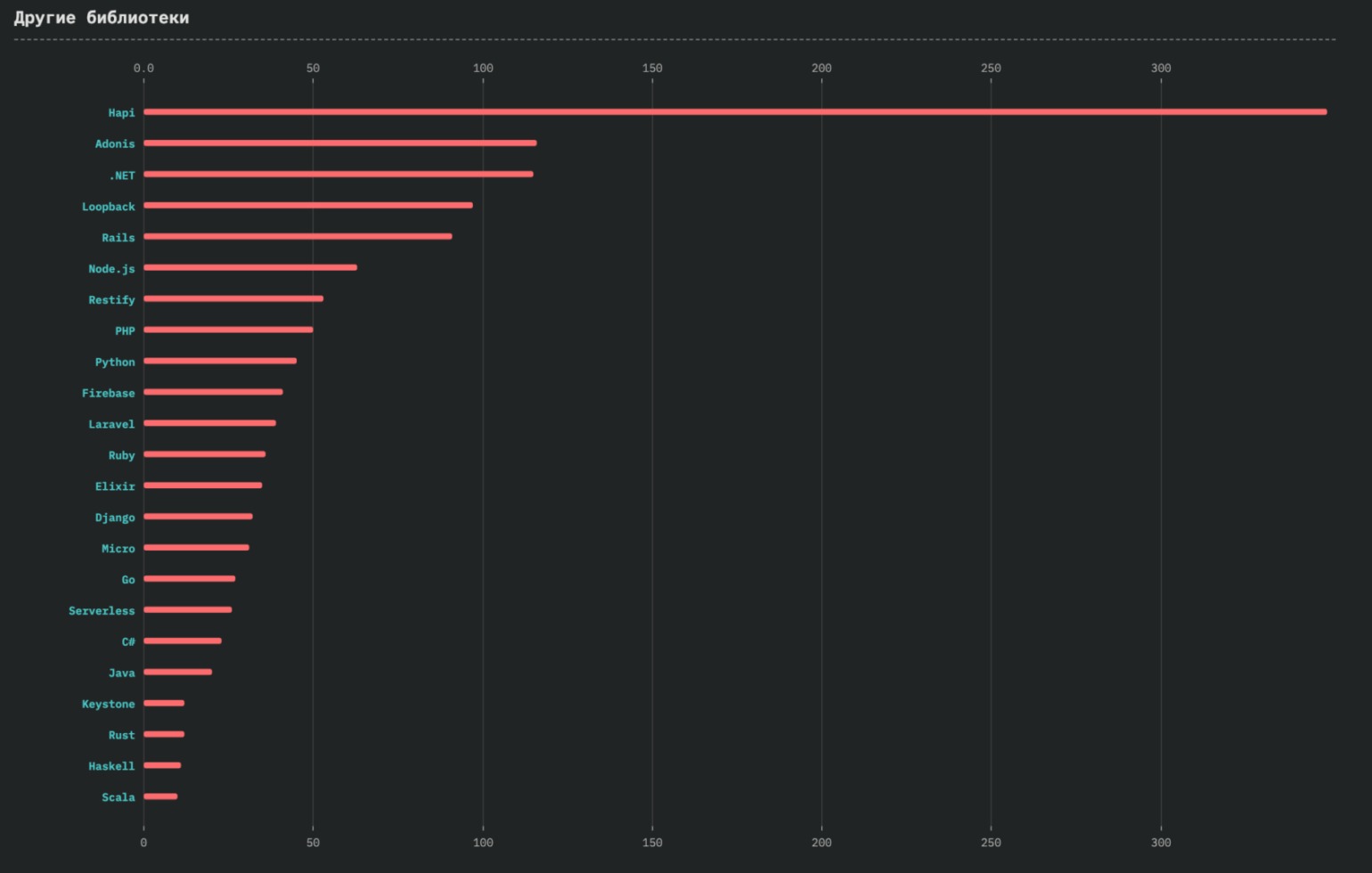
[](https://cdn.tproger.ru/wp-content/uploads/2019/01/back-js-devs-xp-2.jpg)

### Удовлетворённость

Насколько довольны разработчики текущим положением дел в этой категории.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/back-js-devs-happiness.jpg)

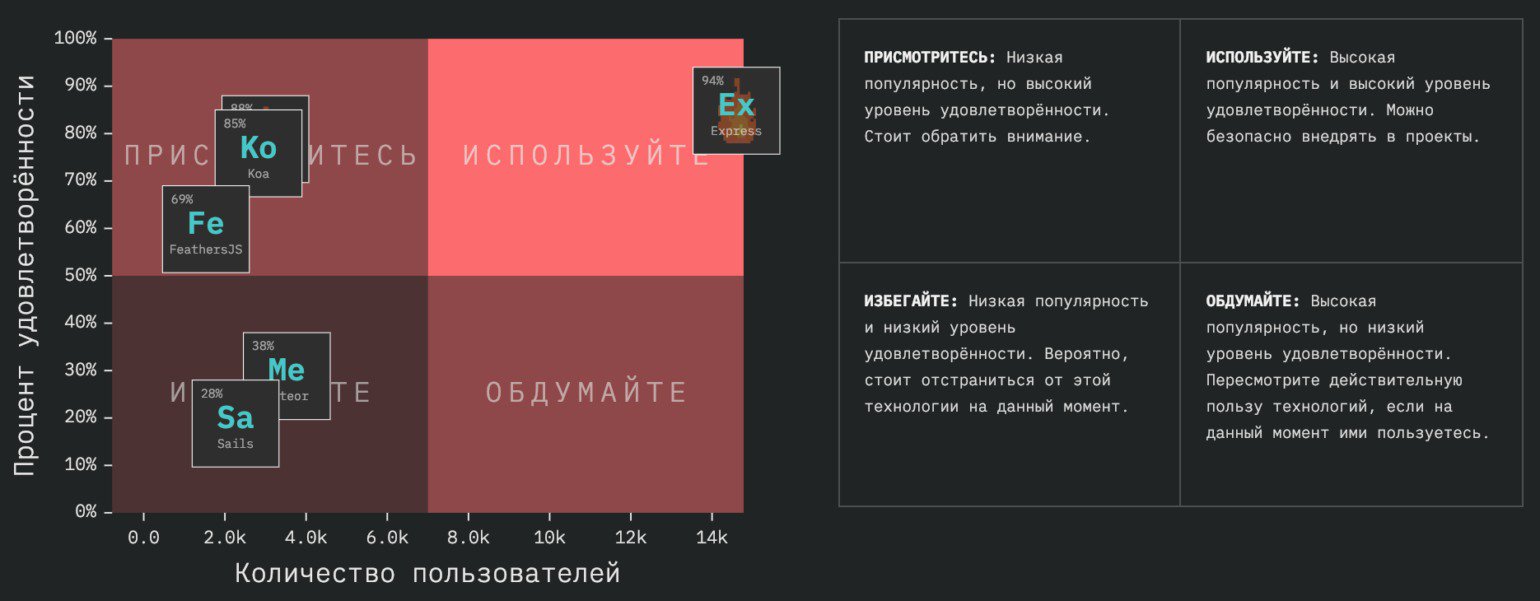
### Список других используемых библиотек

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/back-js-devs-otherLibs.jpg)

### Финальная диаграмма

Диаграмма ниже отображает **степень удовлетворённости** технологией в соотношении с её **популярностью**.

Кроме того, если технологию хотят изучить более 50 % участников опроса, она помечается огнём.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/back-js-devs-final-diagram.jpg)

### Заключение

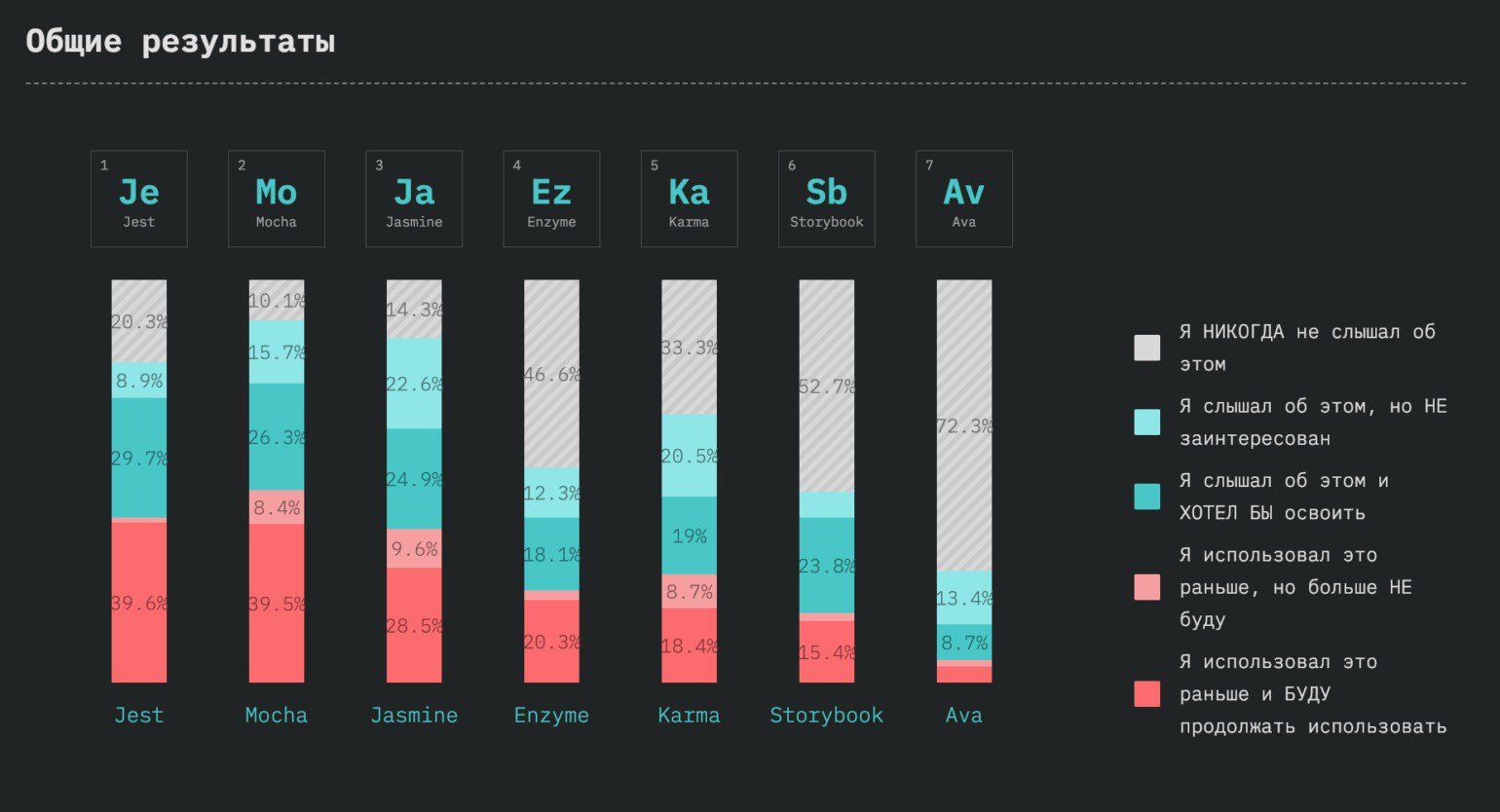
Серверный JavaScript находится в странном состоянии. Несмотря на то, что ежегодно появляется бесчисленное множество фреймворков, только единицам удаётся хоть немного составить конкуренцию Express. Даже Koa, фреймворк, считающийся преемником Express, значительно отстаёт по уровню удовлетворённости пользователей.

С Express конкурировать может разве что Next.js, набирающий популярность. Он завоевал интерес узконаправленным решением проблемы рендеринга со стороны сервера для приложений на React.

Интересно будет понаблюдать за тем, какую роль будут играть бессерверные технологии вроде AWS Lambda. Кто знает, может, знакомый нам в нынешнем виде бекенд скоро останется в прошлом…

## Тестирование

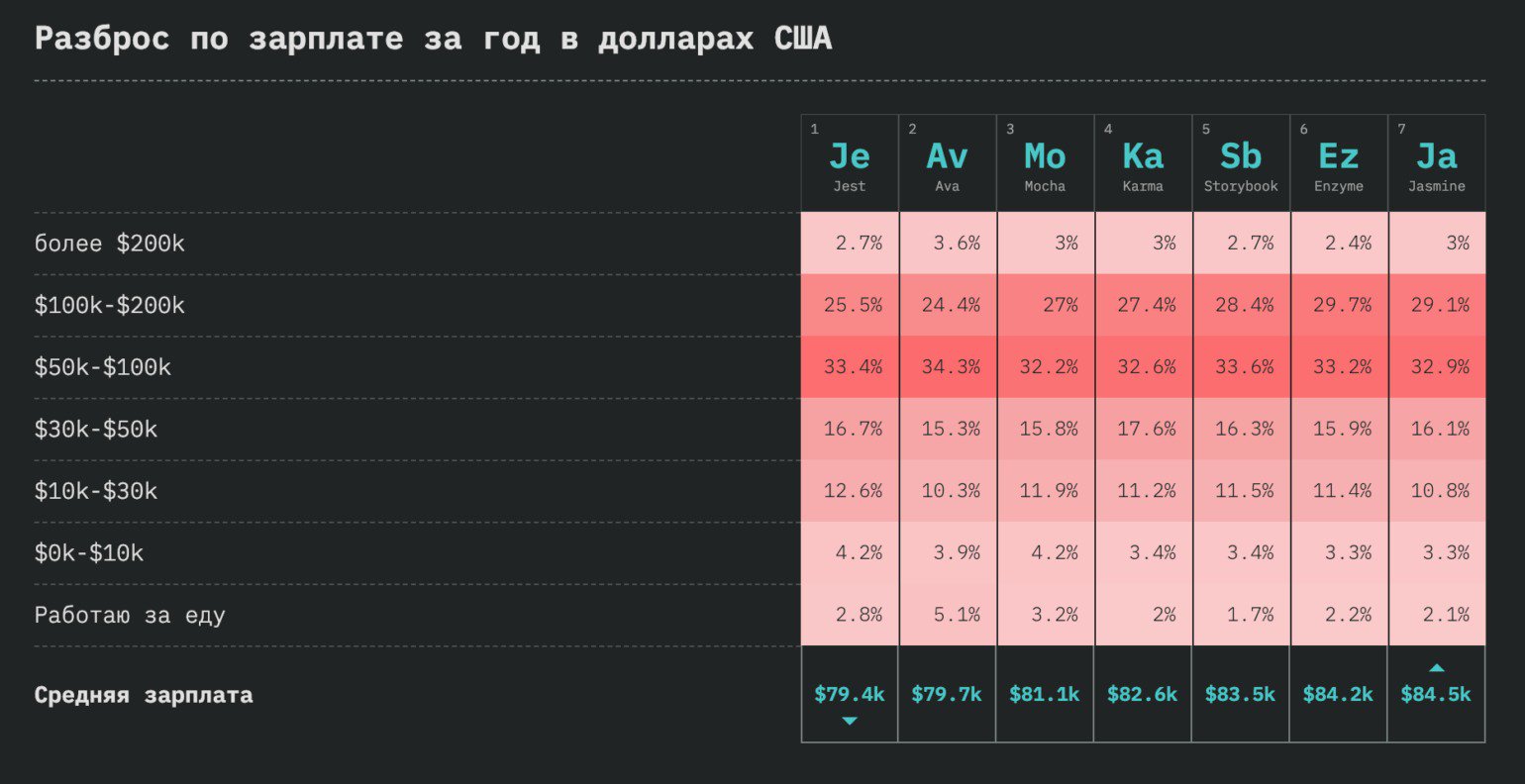
### Обзор

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/testing-js-devs-overall-results.jpg)

### Зарплаты

Разброс в зарплате разработчиков, выбравших пункт «Я использовал это раньше и буду продолжать использовать».

В каждой ячейке отображается процент пользователей в выбранной зарплатной категории. Чем насыщеннее красный, тем больше используется технология в данной категории зарплат.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/testing-js-devs-salary.jpg)

### Размер компании

Разбивка разработчиков, выбравших пункт «Я использовал это раньше и буду продолжать использовать» по размеру компании.

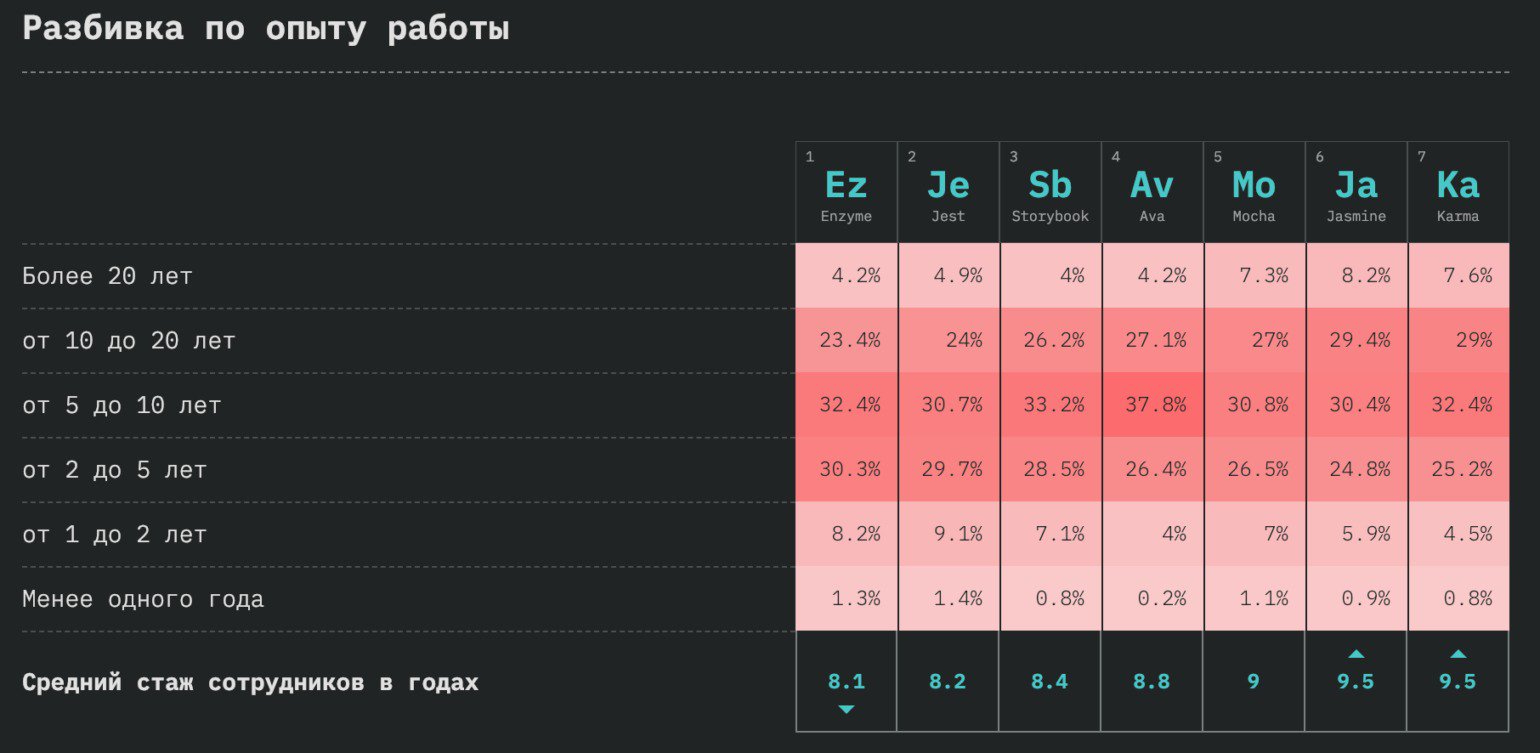
В каждой ячейке отображается процент пользователей в выбранной категории компаний. Чем насыщеннее красный, тем больше используется технология в данной категории.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/testing-js-devs-headcount.jpg)

### Опыт

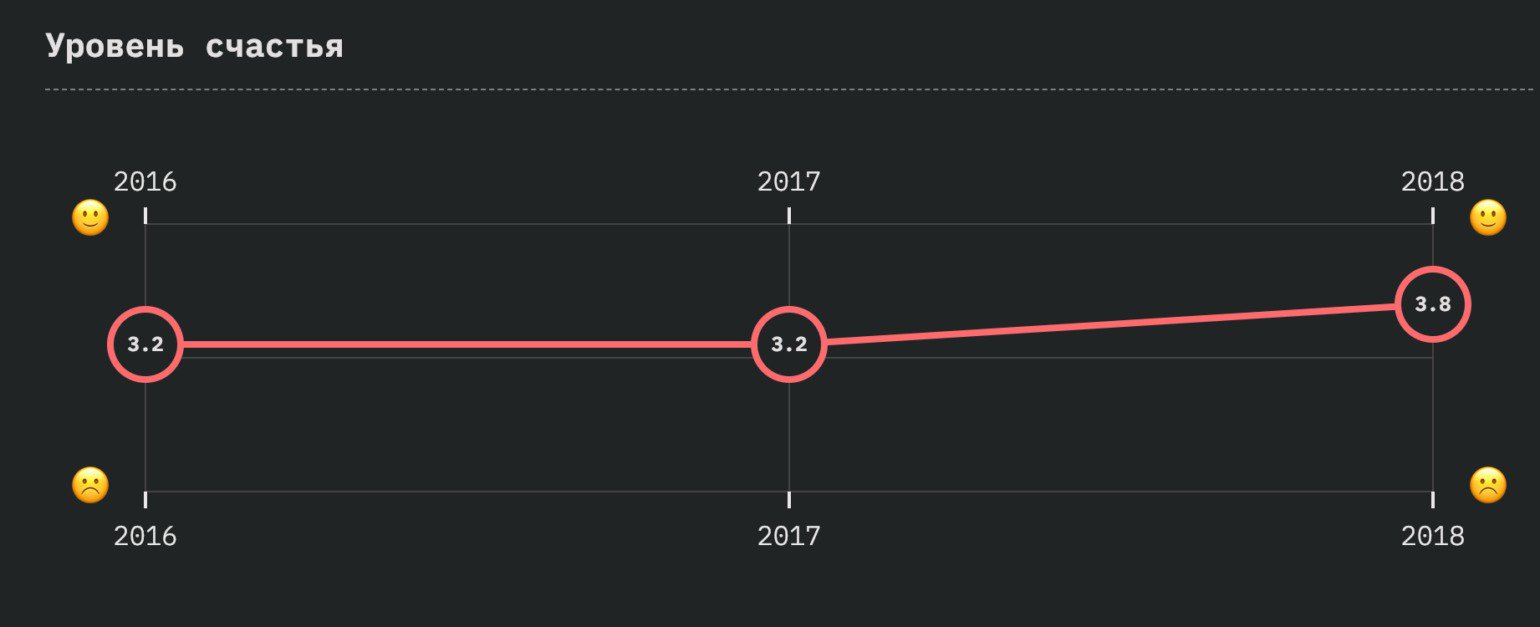
Разбивка по опыту работы для разработчиков, выбравших пункт «Я использовал это раньше и буду продолжать использовать».

В каждой ячейке отображается процент пользователей в выбранной категории стажа. Чем насыщеннее красный, тем популярней технология среди пользователей данной категории стажа.

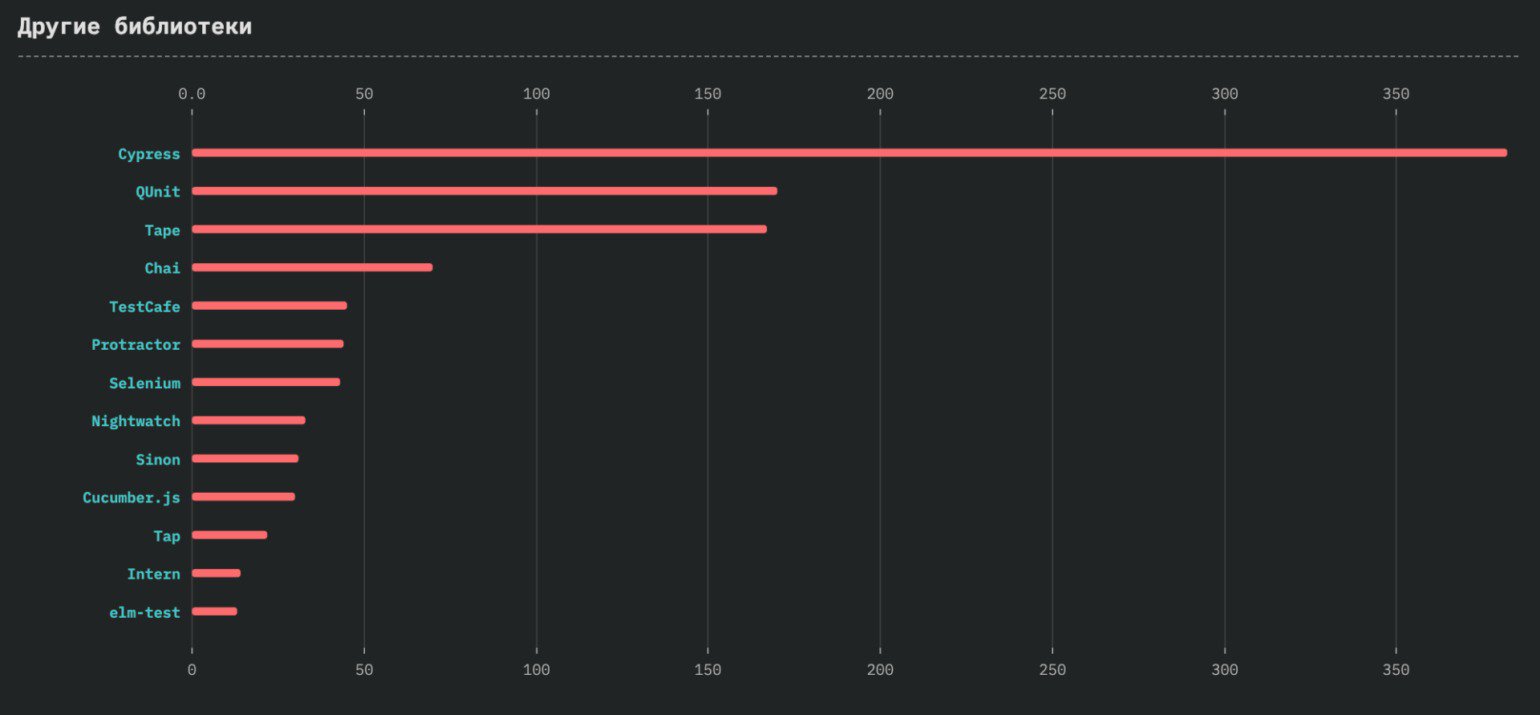
[](https://cdn.tproger.ru/wp-content/uploads/2019/01/testing-js-devs-xp-3.jpg)

### Удовлетворённость

Насколько довольны разработчики текущим положением дел в этой категории.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/testing-js-devs-happiness.jpg)

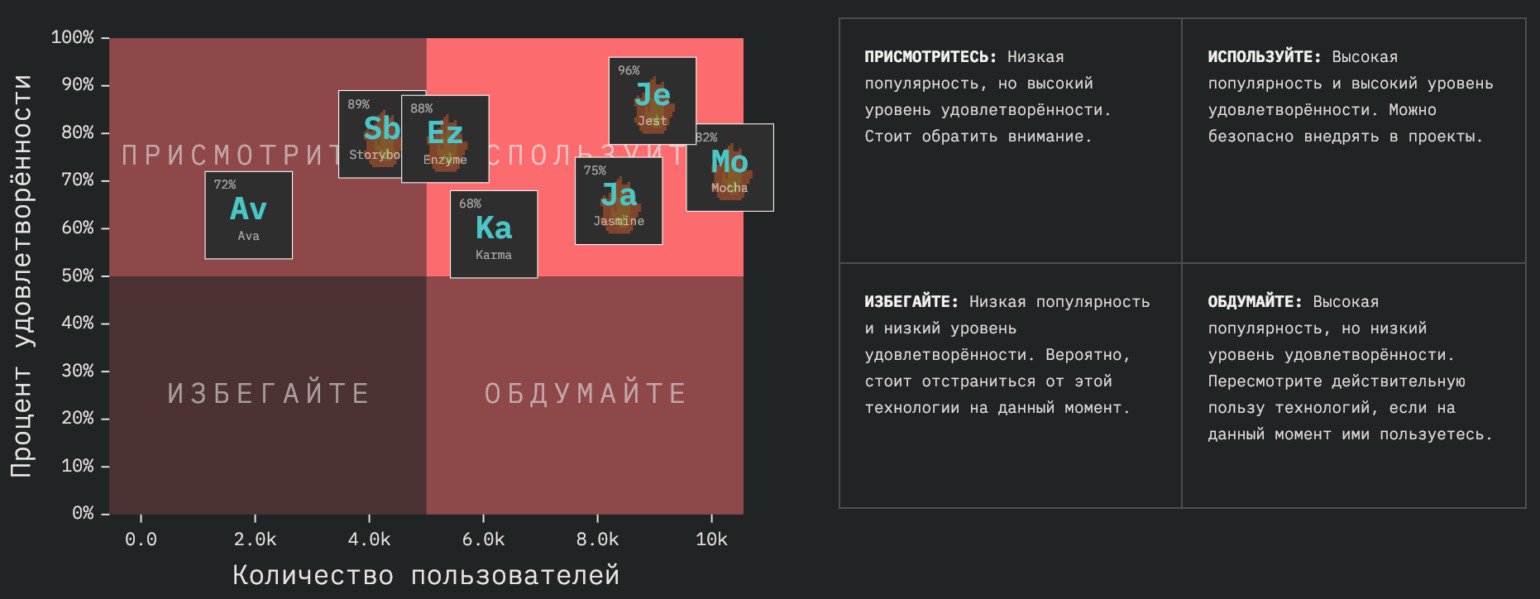
### Список других используемых библиотек

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/testing-js-devs-otherLibs.jpg)

### Финальная диаграмма

Диаграмма ниже отображает **степень удовлетворённости** технологией в соотношении с её **популярностью**.

Кроме того, если технологию хотят изучить более 50 % участников опроса, она помечается огнём.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/testing-js-devs-final-diagram.jpg)

### Заключение

Поле тестирования остаётся в состоянии некоторой неопределённости. В то время как другие части (фронтенд- и бекенд-фреймворки и т.д.) экосистемы JavaScript медленно распределяются вокруг нескольких доминирующих решений, сфера тестирования остаётся довольно раздробленной. Множество разных и дополняющих друг друга инструментов разделяют влияние между собой. Но даже несмотря на это разработчики в целом удовлетворены своими технологиями для тестирования. Самая нижняя планка удовлетворённости равна 6 %.

Результаты опроса подтверждают, что Mocha по-прежнему является наиболее используемой платформой для юнит-тестирования, насчитывая более 10 000 активных пользователей. Более того, большинство разработчиков Node.js знакомы с этим инструментом.

Jest не отстаёт от Mocha, а в плане пользовательской удовлетворённости даже обгоняет: 96 % против 82 %. Более того, процент пользовательской удовлетворённости во всём опросе выше только у ES6.

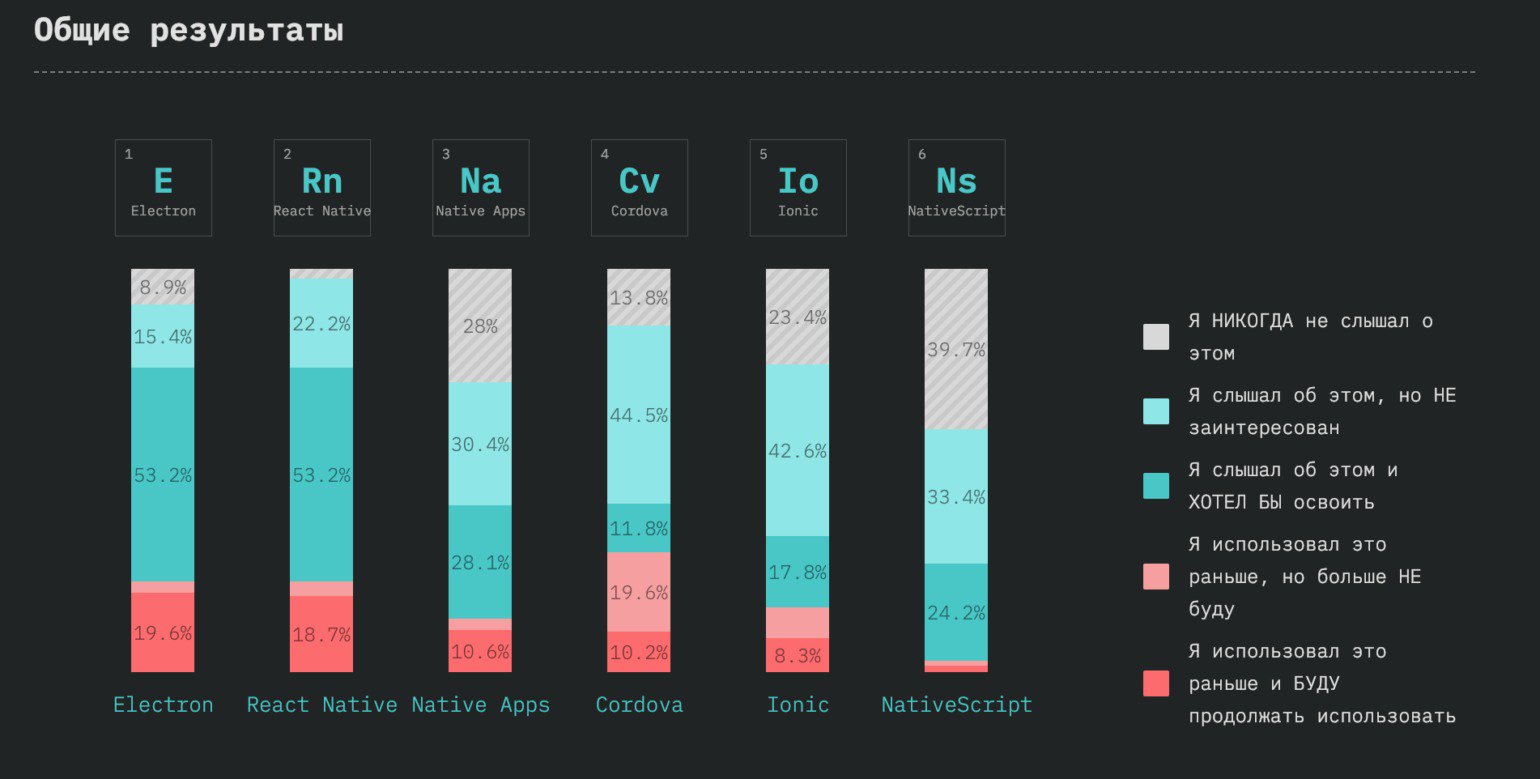
Популярность Jest показывает, что разработчики действительно ценят попытки Facebook создать полнофункциональную среду для тестирования, которая может применяться как для фронтенда, так и для бекенда.

Так называемые «одностраничные приложения» становятся всё сложнее, реализуя ещё больше логики на клиентской стороне. Результаты опроса показывают, что разработчики используют много инструментов для тестирования своих приложений.

## Обзор фреймворков для создания десктопных и мобильных приложений

### Введение

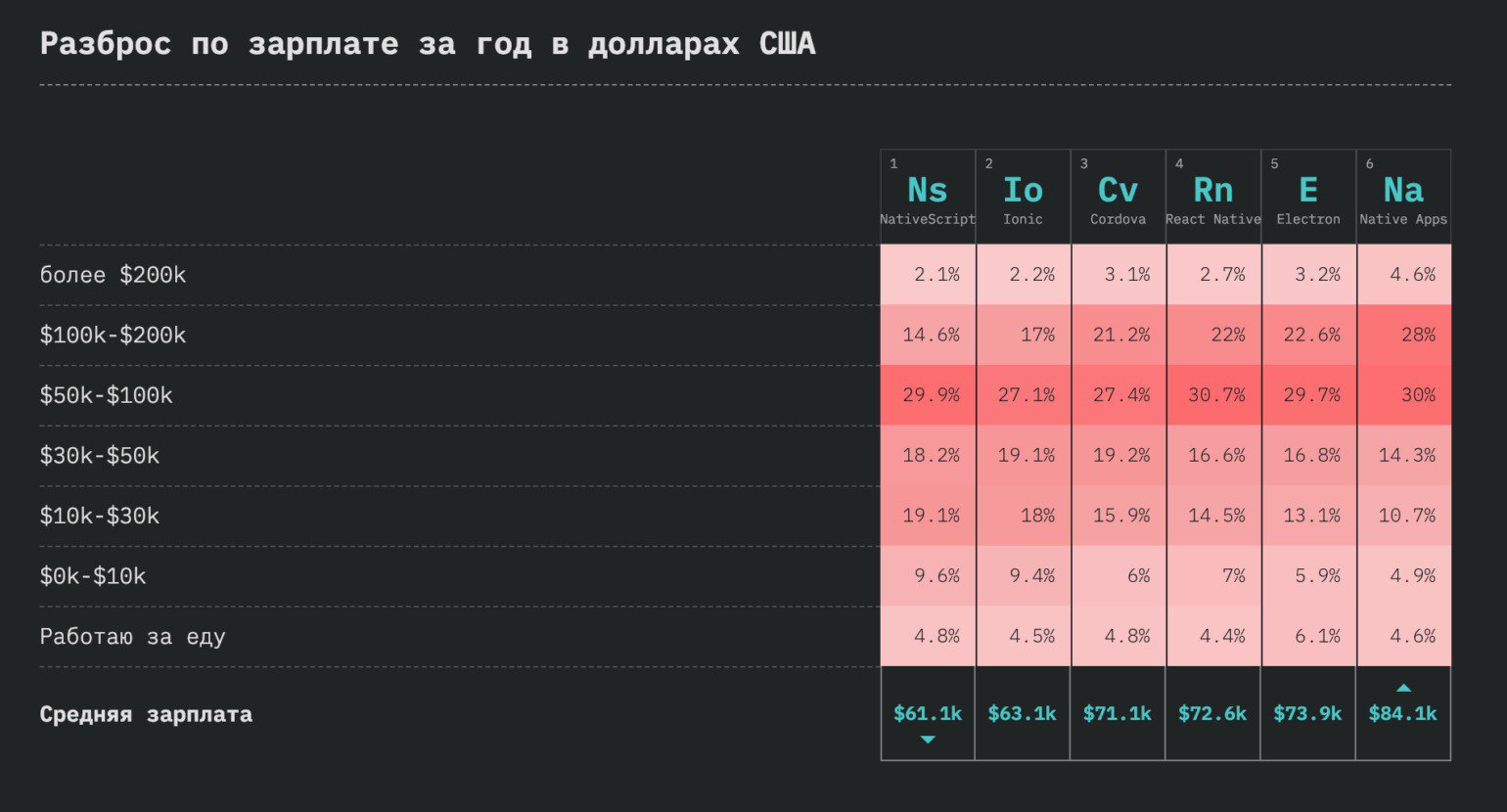
Вы только посмотрите, насколько далеко JavaScript вышел за пределы браузера. А вы всё ещё не верите в тостеры с багами на JS…

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/mobil-desktop-js-devs-overall-results.jpg)

### Зарплаты

Разброс в зарплате разработчиков, выбравших пункт «Я использовал это раньше и буду продолжать использовать».

В каждой ячейке отображается процент пользователей в выбранной зарплатной категории. Чем насыщеннее красный, тем больше используется технология в данной категории зарплат.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/mobile-desktop-js-devs-salary.jpg)

### Размер компании

Разбивка разработчиков, выбравших пункт «Я использовал это раньше и буду продолжать использовать» по размеру компании.

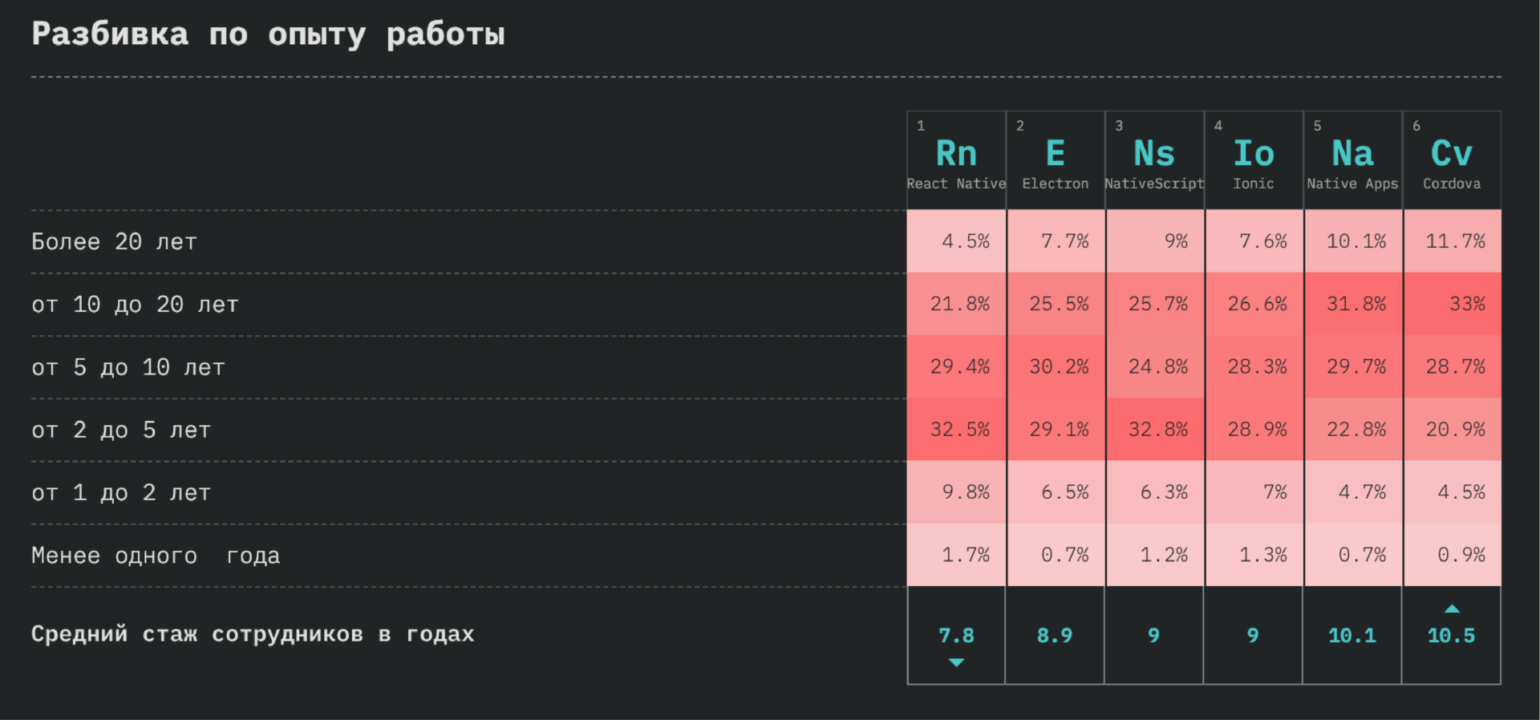
В каждой ячейке отображается процент пользователей в выбранной категории компаний. Чем насыщеннее красный, тем больше используется технология в данной категории.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/mobile-desktop-headcount.jpg)

### Опыт

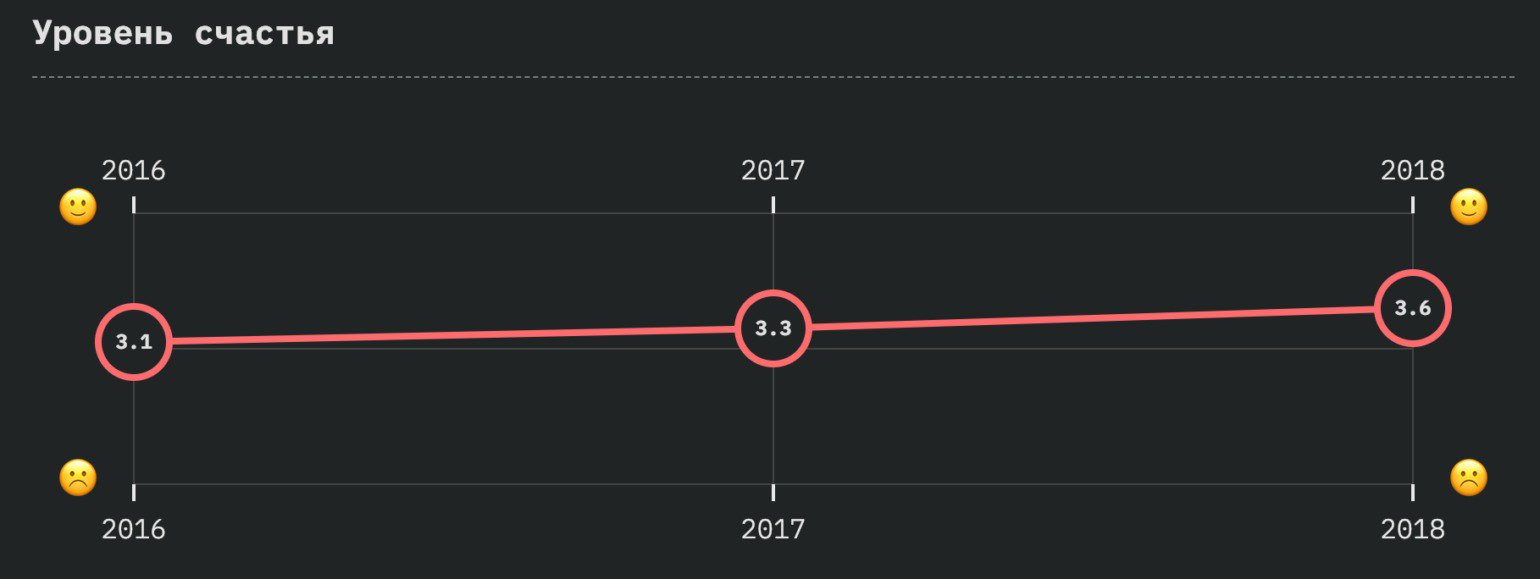
Разбивка по опыту работы для разработчиков, выбравших пункт «Я использовал это раньше и буду продолжать использовать».

В каждой ячейке отображается процент пользователей в выбранной категории стажа. Чем насыщеннее красный, тем популярней технология среди пользователей данной категории стажа.

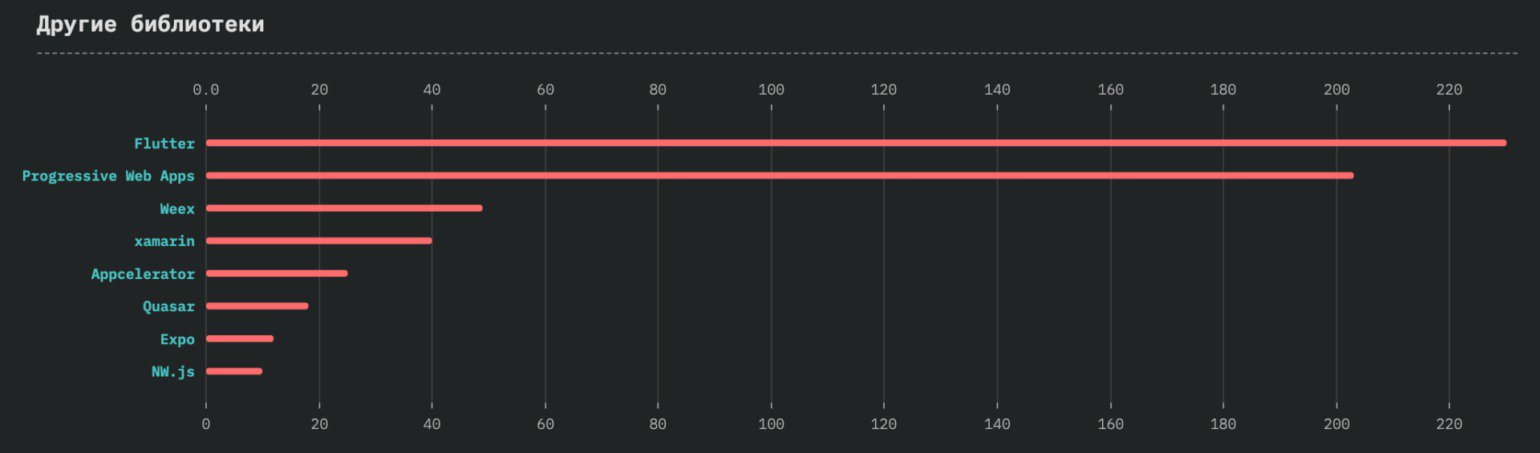
[](https://cdn.tproger.ru/wp-content/uploads/2019/01/mobile-desktop-js-devs-xp-1.png)

### Удовлетворённость

Насколько довольны разработчики текущим положением дел в этой категории.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/mobile-testing-js-devs-happiness.jpg)

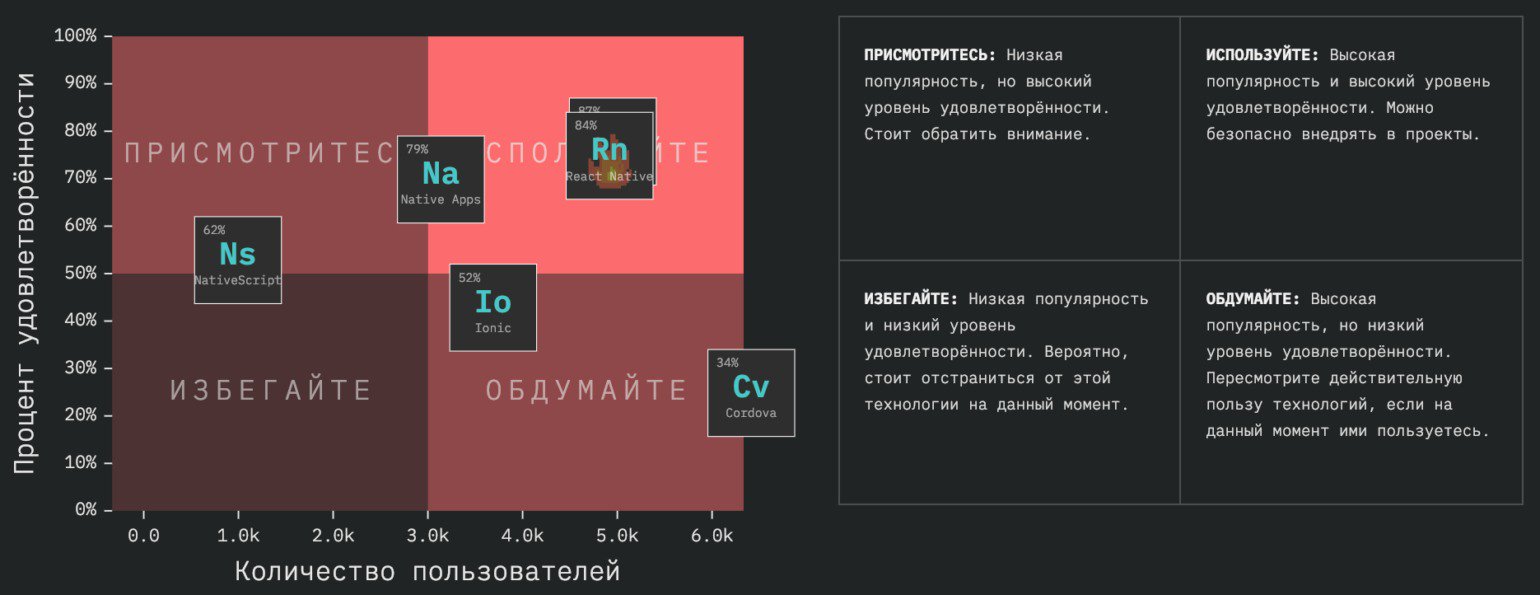
### Список других используемых библиотек

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/mobile-desktop-js-devs-otherLibs.jpg)

### Финальная диаграмма

Диаграмма ниже отображает **степень удовлетворённости** технологией в соотношении с её **популярностью**.

Кроме того, если технологию хотят изучить более 50 % участников опроса, она помечается огнём.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/mobile-desktop-js-devs-final-diagram.jpg)

### Заключение

React Native и Electron занимают лидирующие позиции для создания как десктопных, так и мобильных приложений. Обе технологии показывают одинаковый уровень удовлетворённости и равное количество пользователей. Вот это совпадение.

Универсальность Electron (может работать с любым UI-фреймворком, несмотря на то, что часто ассоциируется с React или Vue.js), объясняет причину высокой удовлетворённости в своей категории.

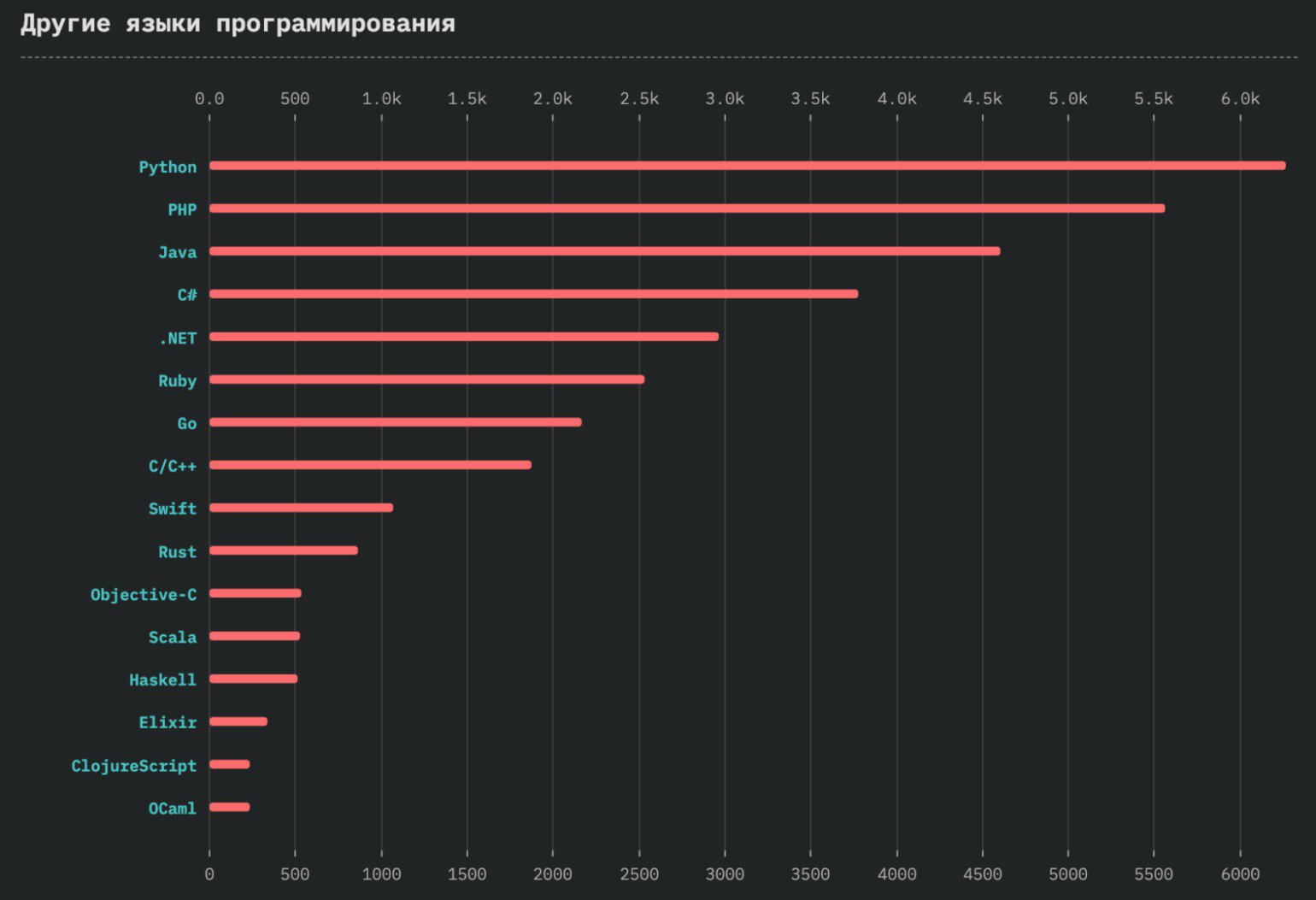
А вот Airbnb недавно [опубликовала](https://medium.com/airbnb-engineering/react-native-at-airbnb-f95aa460be1c) серию статей с объяснением причин отказа от React Native в пользу нативных приложений.

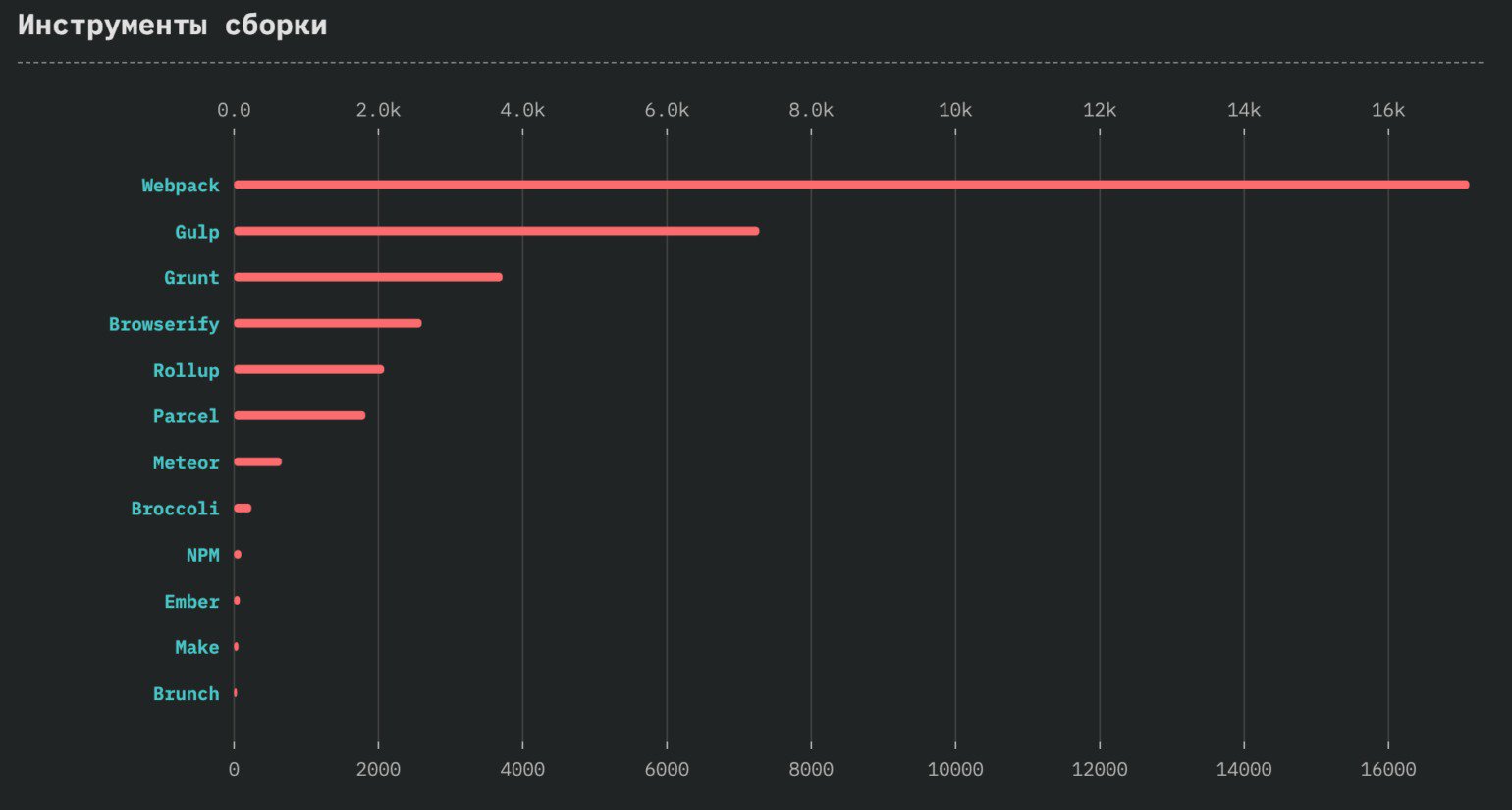
Google тоже не осталась в стороне. Знакомьтесь с [Carlo](https://github.com/GoogleChromeLabs/carlo" \t "_blank) — совершенно новой оболочкой для веб-обработки приложений на Node.js, созданной на основе Puppeteer. Но Carlo компилирует код в Dart, в отличие от React Native. Именно поэтому последний всё ещё будет занимать важную позицию в сообществе разработчиков, знакомых с React.

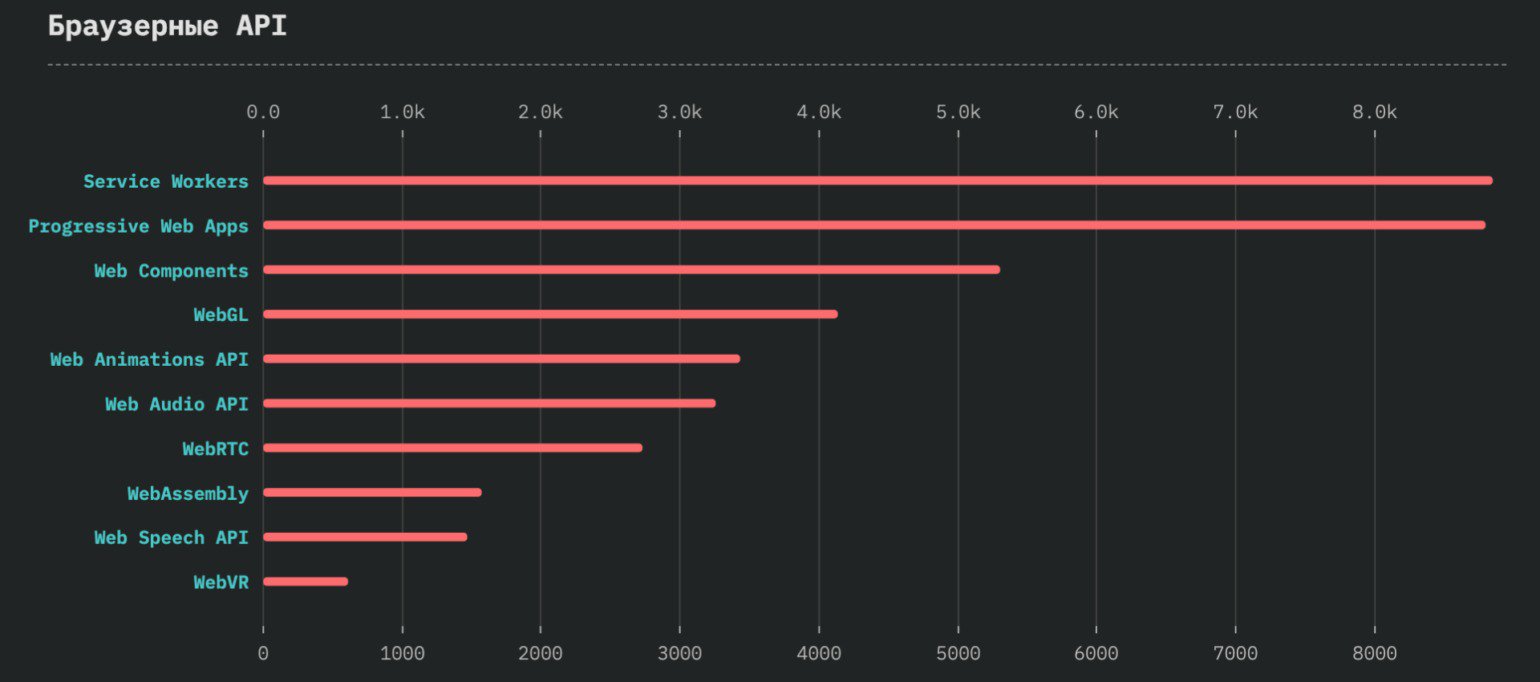
### Другие инструменты

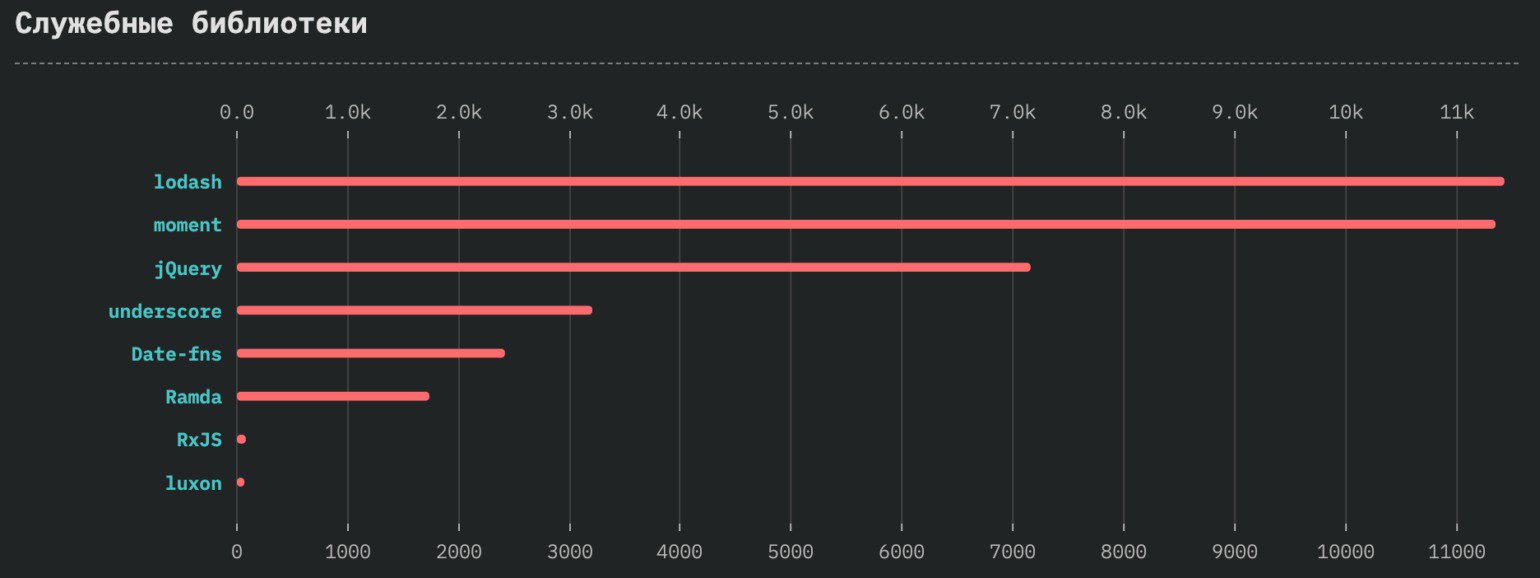
Как бы странно это ни звучало, но среди разработчиков JavaScript есть те, кто работает не только с разными утилитами и упаковщиками для самого же JavaScript, но и те, кто использует в проектах другие языки программирования!

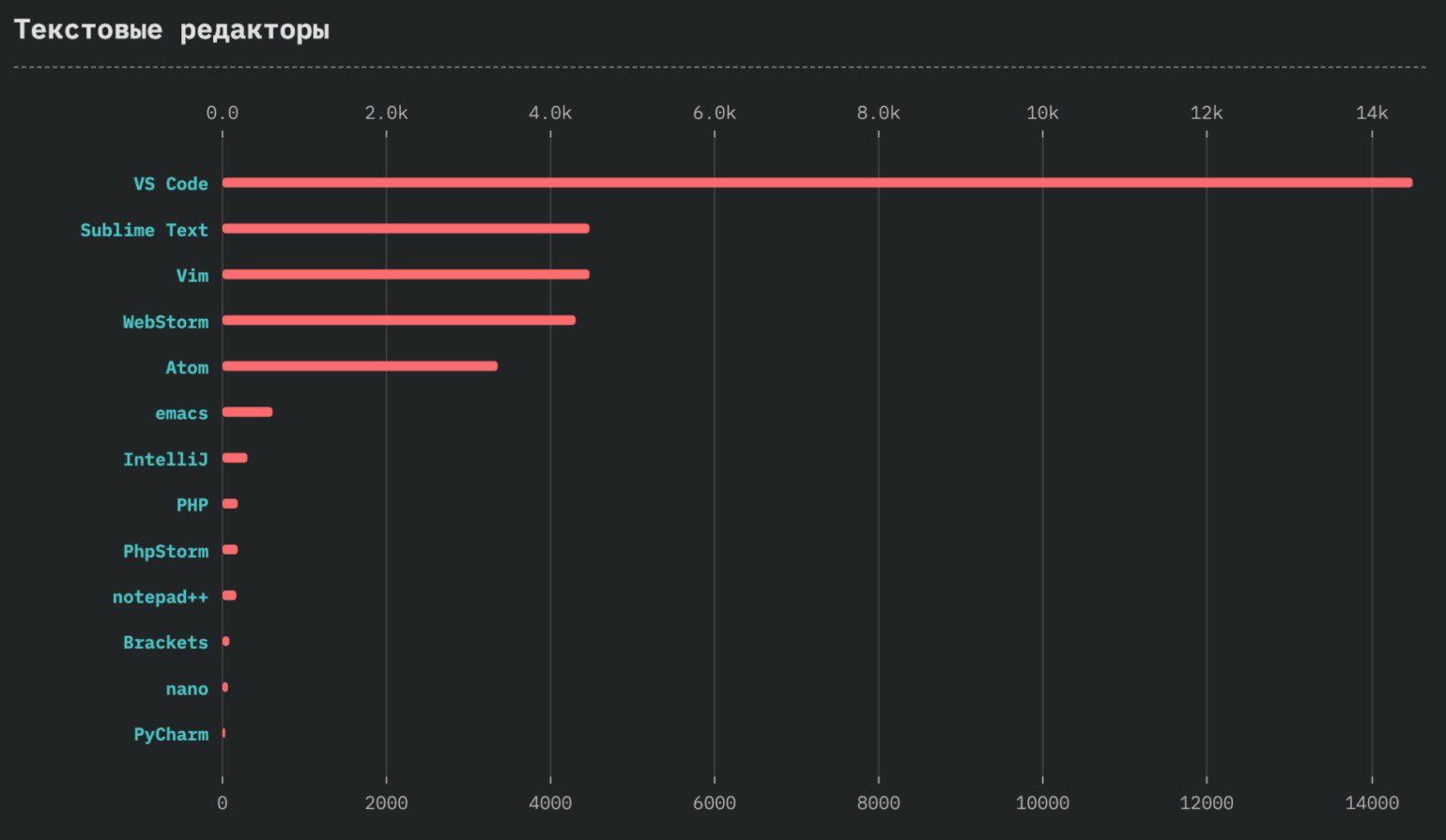
Python является самым популярным языком программирования среди любителей JavaScript.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/Other-programming-langs-js-devs.jpg)

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/Build-tools-js-devs.jpg)

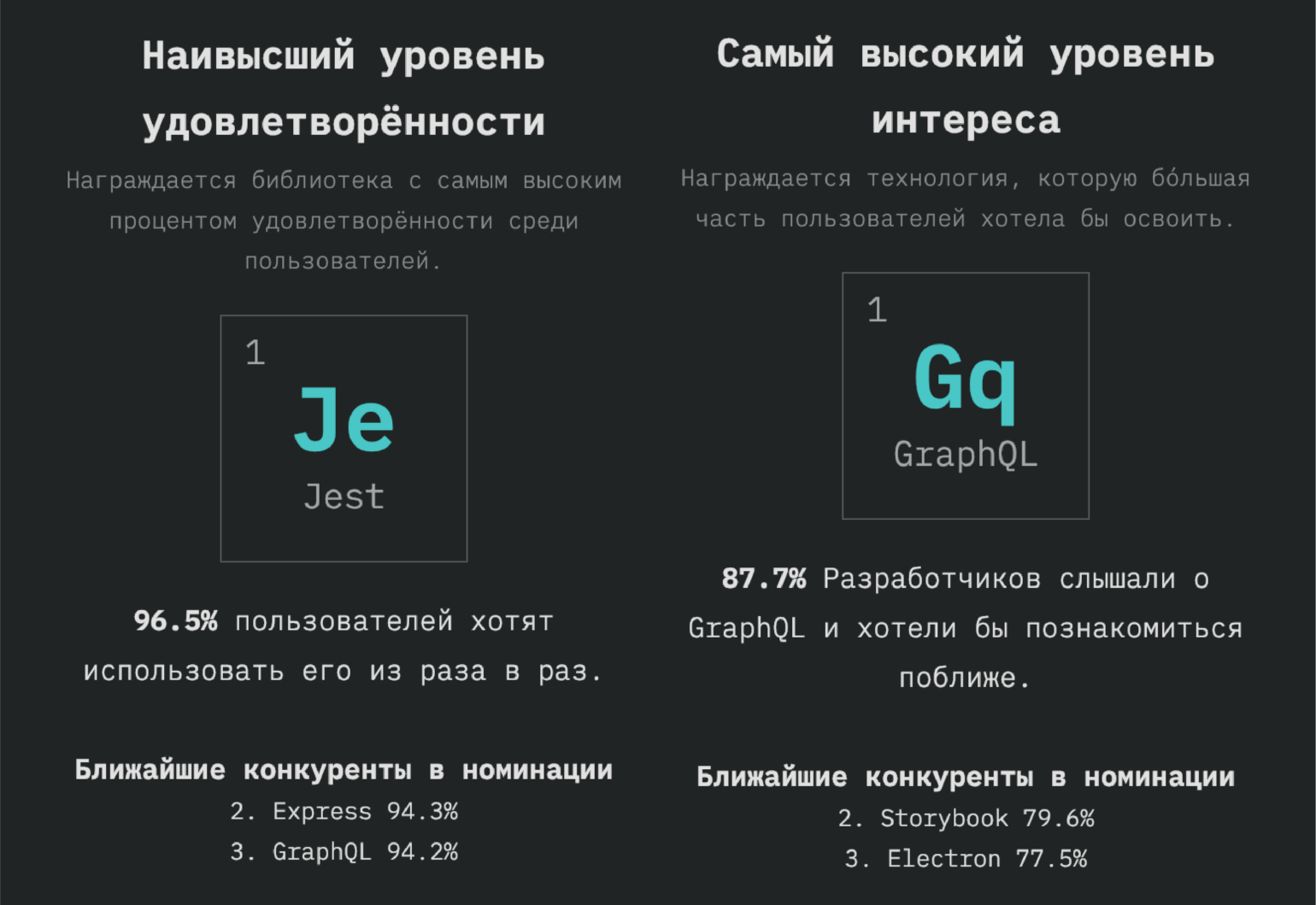
[](https://cdn.tproger.ru/wp-content/uploads/2019/01/browser-apis-js-devs.jpg)

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/utility-libs-js-devs.jpg)

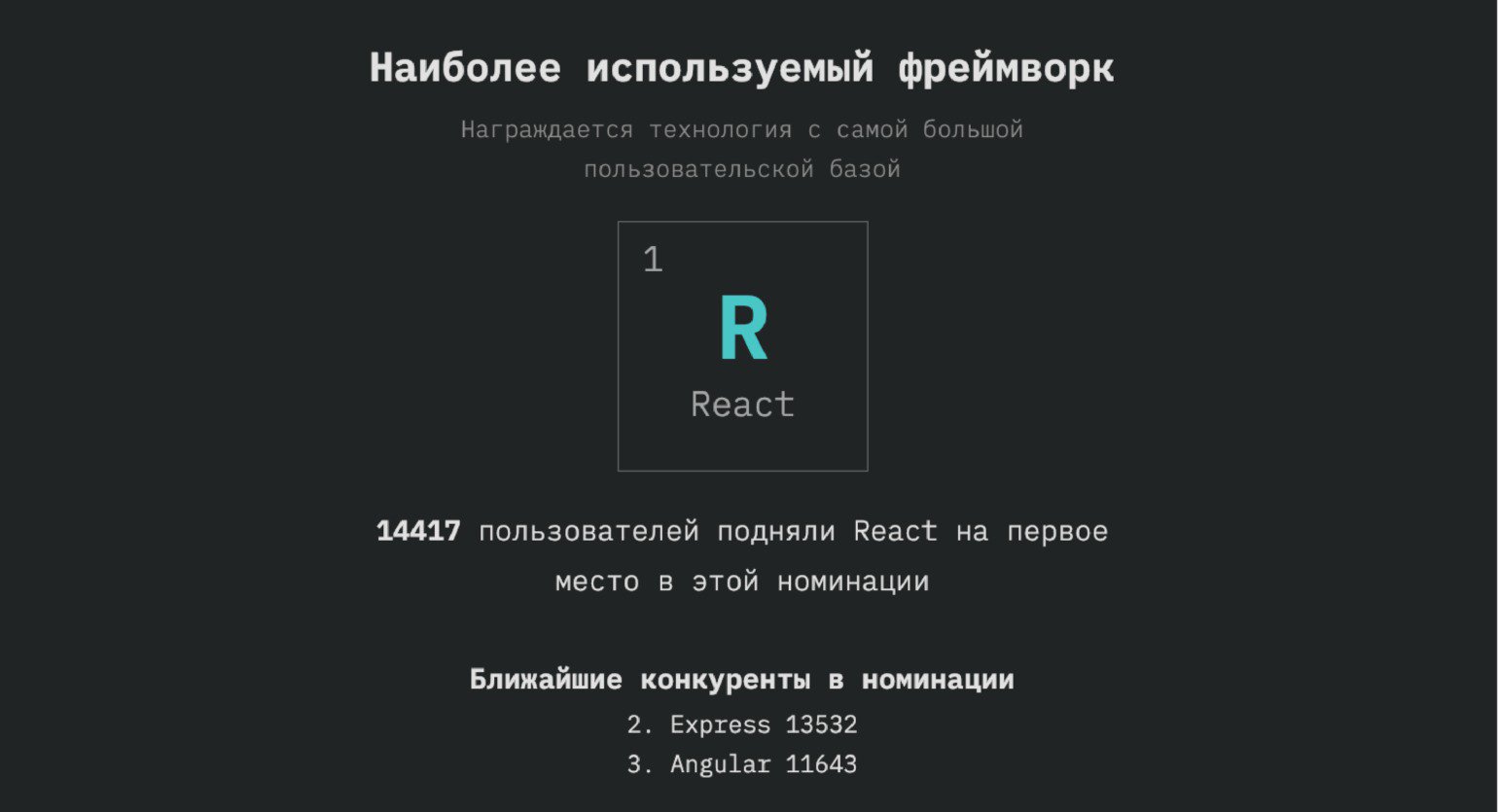
[](https://cdn.tproger.ru/wp-content/uploads/2019/01/text-editors-js-devs.jpg)

### Победители гонки технологий прошедшего года

**Примечание** ES6 не принимал участие в этом соревновании, он слишком силён и важен для подобных затей.

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/awards-1-js-devs.png)

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/awards-2-devs-js.png)

[](https://cdn.tproger.ru/wp-content/uploads/2019/01/awards-3-js-devs.jpg)

## Вывод

Похоже, что 2018 год не принёс чего-то кардинально нового — в основном продолжают развиваться устоявшиеся тренды. И это здорово, потому что разработчики наконец-то могут заняться чем-то более полезным, чем изучением новых фреймворков 🙂