

## Что такое Kotlin

Kotlin — это язык программирования, созданный в компании JetBrains. Его разработали в 2011 году на замену Java, который в компании считали чересчур многословным. Новый язык получился на 40% компактнее предшественника, что помогло ускорить работу над основным продуктом JetBrains — средой разработки IntelliJ IDEA. При этом Kotlin полностью совместим с Java, потому что запускается на его виртуальной машине (JVM).

Если же давать формальное определение, Kotlin — это объектно-ориентированный язык со статической типизацией. Что это значит?

**Объектно-ориентированными** называют языки, в которых все операции происходят с объектами — блоками кода, куда можно «складывать» несколько значений. Все объекты проектируются по специальным шаблонам, которые называются классами.

**Статическая типизация** означает, что типы переменных задаются разработчиком до выполнения программы. Если вы объявили целочисленную переменную, текст в неё поместить уже не получится — компилятор сразу укажет на ошибку. Это замедляет разработку на начальном этапе, но добавляет надёжности.

## Что можно писать на Kotlin

Потенциально Kotlin можно использовать везде, где работает Java — а это и бэкенд, и веб, и десктоп, и куча других задач. Как говорит Андрей Бреслав, создатель языка, «Kotlin — это язык для всех платформ». И в этом большая доля правды.

Тем не менее у каждого языка есть своя ниша. Kotlin пока их две — это серверная и [мобильная](#) разработка. Хотя его всё чаще можно встретить и в других областях — например, в [науке](#) и Data Science.

Давайте рассмотрим все сферы по порядку.

### Приложения для Android

До 2017 года официальным языком для Android-приложений был Java, но потом Google объявил свою мобильную ОС Kotlin-first:

Сегодня на Kotlin написано 80% самых скачиваемых программ в Play Market, включая приложения от Netflix, Twitter, Reddit и самого Google.

### Серверные приложения

Благодаря совместимости с JVM(java virtual machine), Kotlin можно запустить на любых серверах, где работает Java. Поэтому его часто используют для

создания бэкенда. Например, на Kotlin уже работают облачные сервисы Jira и некоторые продукты Adobe.

### **Веб-приложения (фронтенд)**

Помимо бэкенда, есть ещё фронтенд. Почти весь фронтенд пишется на одном языке — JavaScript. А Kotlin как раз умеет компилироваться в JavaScript — и такую его версию можно использовать для создания фронтенда. То есть на нём вполне реально написать веб-приложение целиком.

### **Кросс-платформенные мобильные приложения**

Допустим, нам нужно написать мобильное приложение для аренды самокатов. Оно должно одинаково хорошо работать и на айфонах, и на смартфонах под управлением Android.

В таком случае нам может помочь кросс-платформенная разработка — это когда мы пишем код один раз, а потом адаптируем под несколько систем. В Kotlin есть технология, которая позволяет так делать, — называется Multiplatform. Её используют уже довольно много компаний — например, тот же «Яндекс».

### **Читаемость**

Когда язык Java только появился, он был гораздо проще и лаконичнее конкурентов — тех же C и C++, правда, по современным меркам этот язык уже кажется многословным и несколько старомодным. В Java всё вертится вокруг обратной совместимости, которая так важна для энтерпрайза, в котором «кофейный» язык является стандартом.

Создатели же Kotlin изначально делали ставку на компактность: чем меньше ненужного кода, тем проще работать программисту и тем быстрее идёт разработка. В итоге одной строкой в Kotlin можно описать то, на что в Java уйдёт восемнадцать.

### **Скорость**

Тут шайба, кажется, на стороне Java. Он лучше адаптирован под JVM, поэтому приложения компилируются на 15–20% быстрее в сравнении с Kotlin. Однако существует такая штука, как инкрементальная компиляция — это когда пересобирается не весь проект, а только те участки, где программист внёс изменения. Если подключить к Kotlin специальный плагин, обеспечивающий такую компиляцию, то сборка может проходить даже быстрее, чем на Java. Так что в этом пункте ничья.

### **Безопасность**

В языках со статической типизацией, таких как Java и Kotlin, компилятор следит, чтобы не смешивались несовместимые типы данных — например, строка и число.

В Kotlin вшито несколько функций, которые упрощают работу с типами — например, язык может сам привести переменные к единому типу, если того

требует логика кода. Эта функция называется smart cast, или «умное преобразование».

## Другие киллер-фичи Kotlin

Вот ещё несколько фишек, которые дают Kotlin преимущество в скорости, безопасности и производительности.

### Корутины

Coroutines (корутины) — это средство, которое обеспечивает параллелизм, чтобы программа могла выполнять несколько операций одновременно. Когда возникает необходимость, выполнение одной функции приостанавливается с сохранением данных, и начинает работать другая функция.

Допустим, нам нужно сделать приложение — электронный дневник, чтобы ученик мог зайти в него, узнать свои оценки и посмотреть, что задали.

Если загружать с сервера все данные последовательно, то приложение будет открываться очень долго. Вместо этого ненужные пока данные (например, новости), можно вынести в корутину и поставить их подгрузку на паузу, пока грузится основной интерфейс.

Корутины позволяют расставить приоритеты в работе программы. Этот механизм напоминает многопоточность, но тратит меньше ресурсов процессора — за что их и ценят, например, в мобильной разработке.