

"Введение в язык Kotlin: история, синтаксис и примеры использования"

Буланов Дмитрий





- Предыстория язык Java
- Знакомство с языком Kotlin
- Базовый синтаксис и конструкции языка
- Примеры решения задач

Содержание



Немного предыстории. Язык программирования Java



Java — строго типизированный язык программирования общего назначения. Относится к объектно-ориентированным языкам программирования.

Создатели: Джеймс Гослинг и Sun Microsystems

Появился: 1995 год



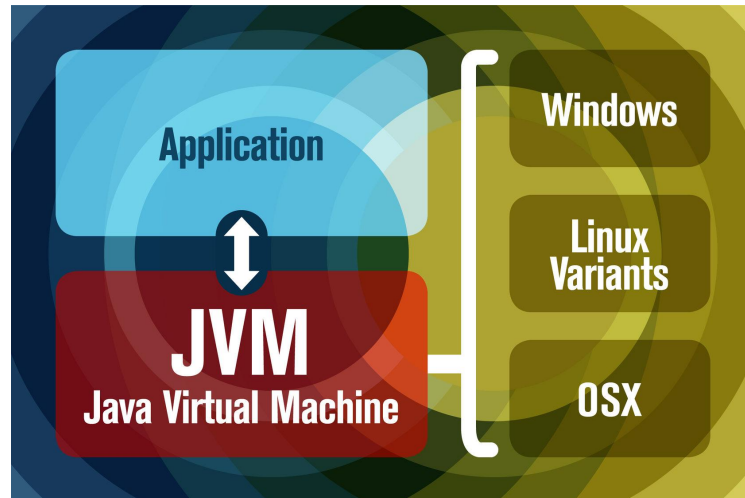
Java. JVM



Создатели реализовали принцип **WORA**:
write once, run anywhere или «пиши один раз, запускай везде»

Это значит, что написанное на Java приложение можно запустить на любой платформе, если на ней установлена среда исполнения Java.

Эта задача решается благодаря компиляции написанного на Java кода в байт-код. Этот формат исполняет **JVM** или виртуальная машина Java.



Все ли так хорошо?



Другие языки
программирования

```
print('Привет!')
```



Java

```
class Praktikum {  
    public static void main(String[] args) {  
        System.out.println("Здравствуй, сэр, рад  
        приветствовать вас!");  
    }  
}
```



Язык программирования Kotlin

Kotlin (Котлин) — объектно-ориентированный язык программирования, работающий поверх Java Virtual Machine и разрабатываемый компанией JetBrains.

Создатели: JetBrains

Появился: 2017 год



Язык программирования Kotlin



Почему такое название?

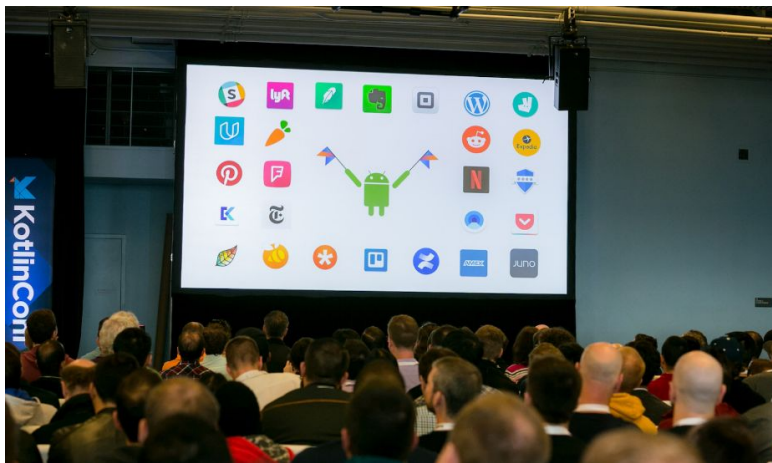
Название языка



Котлин — остров в Финском заливе Балтийского моря, в 30 км западнее центра Санкт-Петербурга.

Сферы использования

- приложения для устройств с операционными системами Android;
- программы под iOS, Windows, Linux, Mac OS;
- серверные и клиентские веб-приложения, десктопное ПО;
- софта для анализа, обработки и представления данных — Data Science.



ОС Android

Android — операционная система для смартфонов, планшетов, электронных книг, цифровых проигрывателей, наручных часов, фитнес-браслетов, игровых приставок, ноутбуков, нетбуков, смартбуков, очков Google Glass, телевизоров, проекторов и других устройств

11 июля 2005 года корпорация Google купила компанию Android, Inc. за 130 миллионов долларов





Преимущества Kotlin



1. **Kotlin** полностью совместим с **Java**. Это значит, что можно вызывать функции, объявленные в Java-коде, прямо из кода **Kotlin** и наоборот, и в том числе пользоваться большим количеством уже существующих библиотек на **Java**.
2. **Лаконичность** и **простота** в изучении. Это один из основных факторов, из-за которых все больше разработчиков выбирает именно **Kotlin**.
3. **Безопасность**. На этапе компиляции предусмотрена автоматическая проверка кода. В котлин есть функция **Null Safety**, которая помогает избежать ошибок неопределённости в коде — самых распространённых и самых трудно выявляемых ошибок.
4. **Kotlin** имеет **открытый исходный код**. Код **Kotlin** открыт для программистов, и его внедрение в проект тоже бесплатно. Открытый исходный код упрощает поиск проблем. Разработчики **Kotlin** прислушиваются к сторонним разработчикам и вносят правки, предлагаемые сообществом.

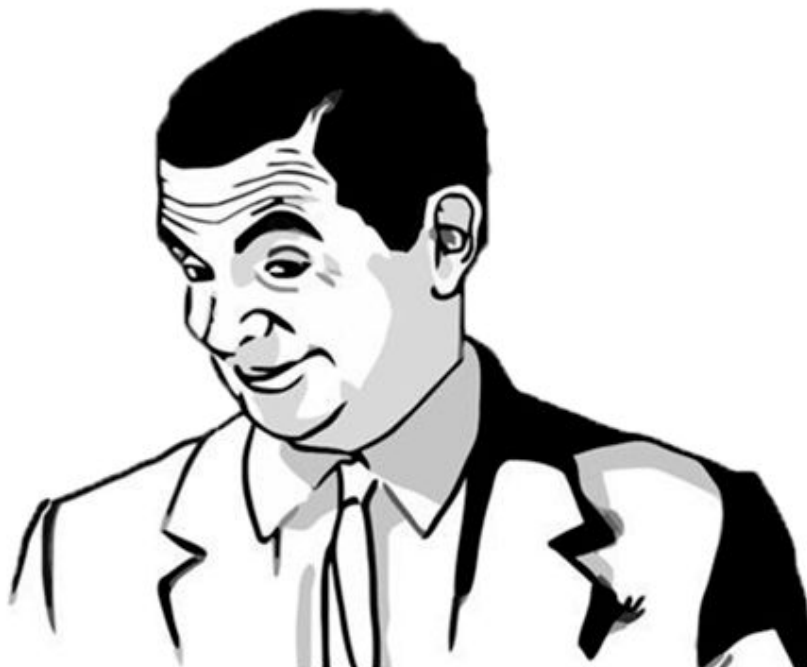


Недостатки Kotlin

1. **Низкая скорость компиляции.** В среднем компиляция на котлин дольше, чем на джава на 30-40 %. Это связано с тем, что компиляция в виртуальной машине JVM предназначена для джава, поэтому с родным языком она работает быстрее.
2. **Мало дополнений.** В силу молодости языка у него намного меньше библиотек и фреймворков, чем у джава.
3. **Малочисленное сообщество.** Официальный релиз языка состоялся в 2017 году. Сообщество разработчиков на языке программирования Kotlin еще молодое и статей по разработке не так много.

Недостатки Kotlin

Maybe we need back to **Java**



Среда разработки

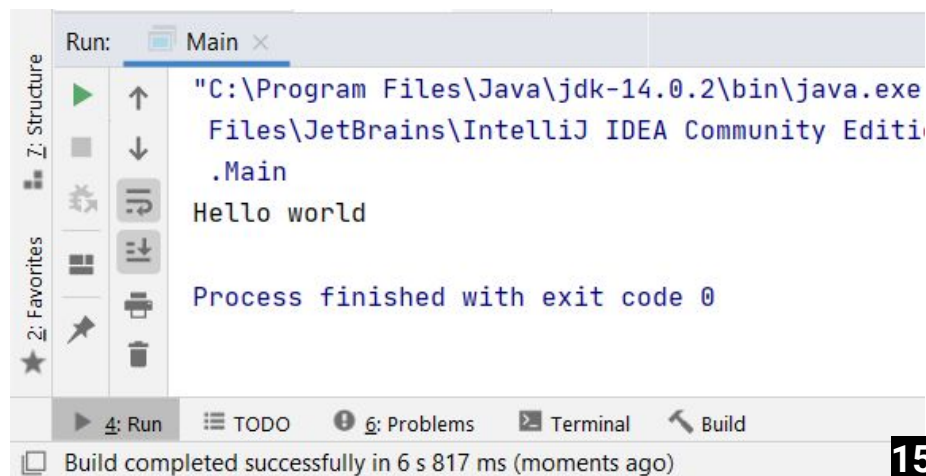
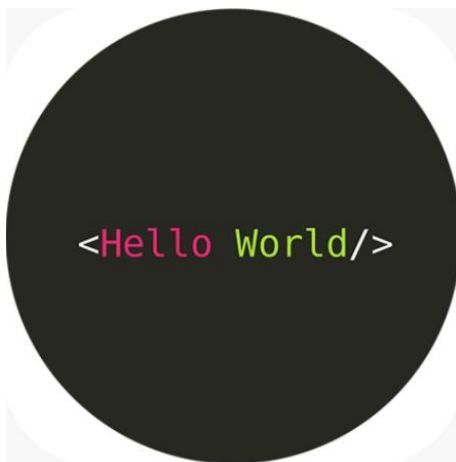


- **IntelliJ IDEA** - интегрированная среда разработки программного обеспечения для многих языков программирования, в частности Java, JavaScript, Python, разработанная компанией JetBrains
- **Eclipse** - свободная интегрированная среда разработки модульных кроссплатформенных приложений.
- **Android studio** - интегрированная среда разработки (IDE) для работы с платформой Android, анонсированная 16 мая 2013 года на конференции Google I/O.



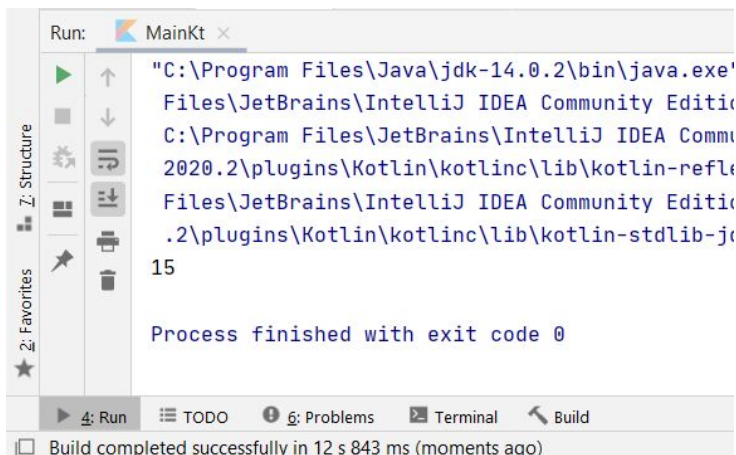
Программа Hello world

```
fun main(args: Array<String>) {  
    println("Hello world")  
}
```



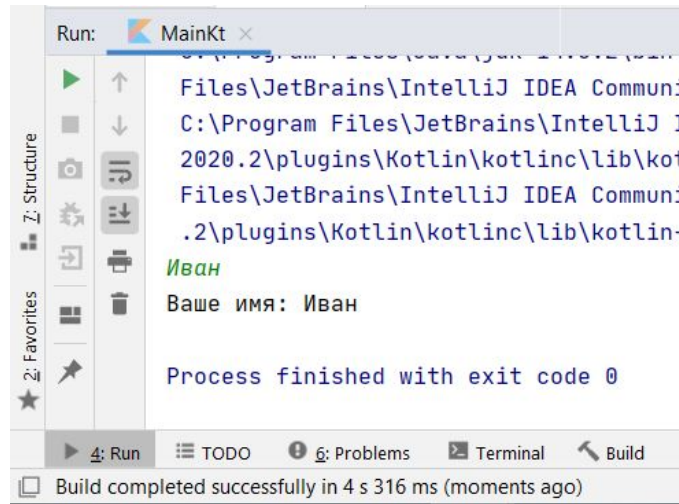
Пример сложения двух чисел

```
fun main() {  
    var a: Int = 5  
    var b = 10  
    println(a + b) // 15  
}
```



Пример ввода данных

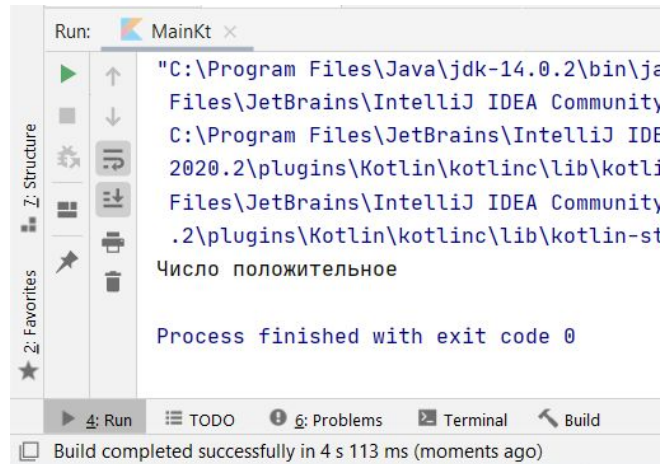
```
fun main() {  
    var name = readLine()  
    println("Ваше имя: $name") // Ваше имя: Иван  
}
```



Пример использования условных конструкций

Условные конструкции позволяют направить выполнение программы по одному из путей в зависимости от условия.

```
fun main() {  
    val a = 10  
    if (a >= 0)  
        println("Число положительное ")  
    else  
        println("Число отрицательное")  
}
```



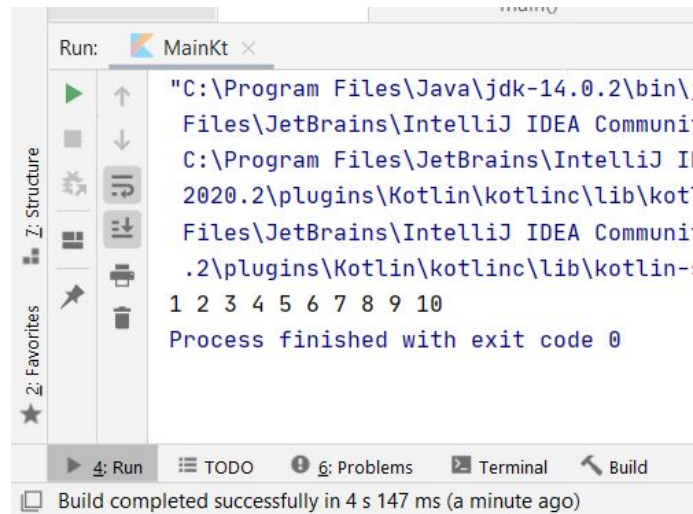
Пример работы с циклами

```
fun main() {  
    for(i in 1..10)  
        print("$i ")  
}
```

Циклы представляют вид управляющих конструкций, которые позволяют в зависимости от определенных условий выполнять некоторое действие множество раз.



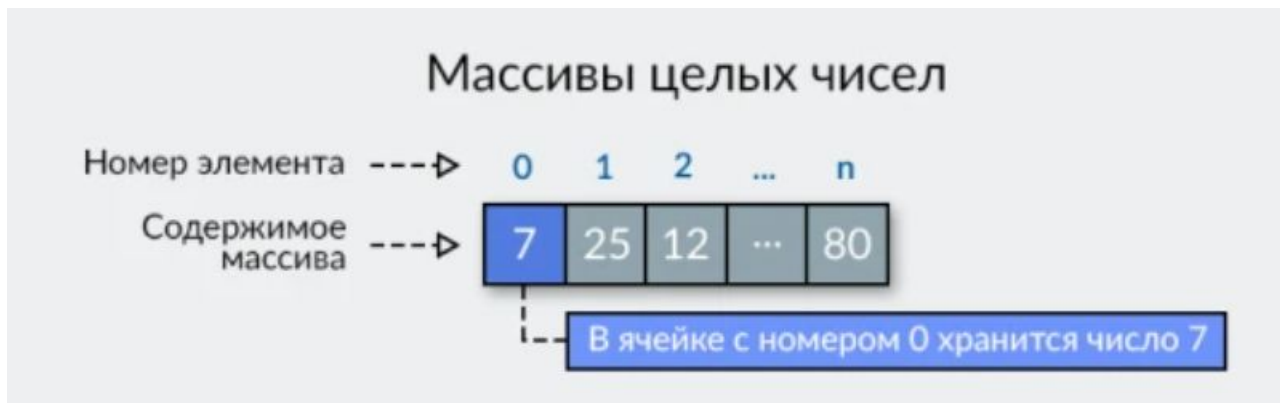
Kotlin



Что такое массив ?

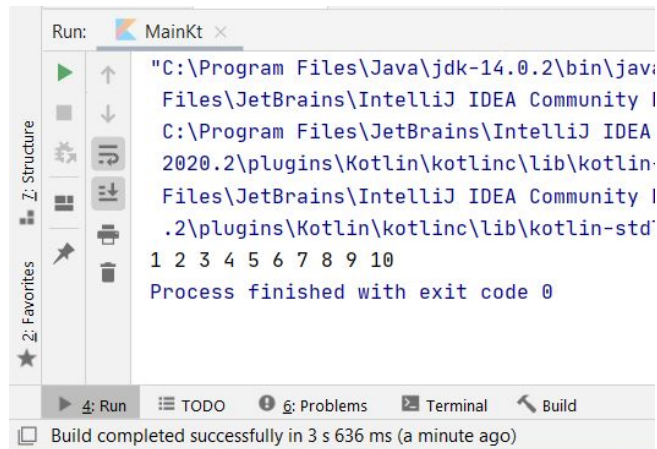


Массив — это структура данных, в которой хранятся элементы одного типа. Его можно представить, как набор пронумерованных ячеек, в каждую из которых можно поместить какие-то данные. Доступ к конкретной ячейке осуществляется через её номер (индекс)



Пример работы с массивами

```
fun main() {  
    var arr = arrayOf(1,2,3,4,5,6,7,8,9,10)  
    for(i in 0..arr.size-1)  
        print("${arr[i]} ")  
}
```



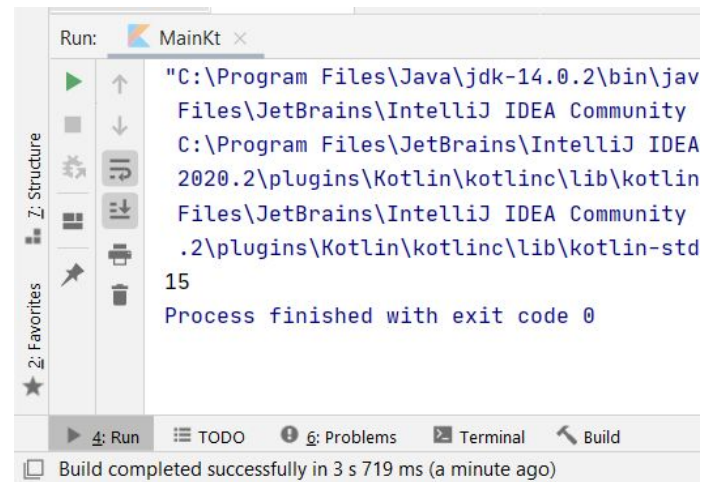
Пример функций



Функции являются основой многих языков программирования. Проще говоря, функция позволяет определить блок кода, который выполняет определенную задачу.

```
fun sumElems(a:Int, b:Int):Int {  
    return a + b;  
}
```

```
fun main() {  
    print(sumElems(5, 10)) // 15  
}
```

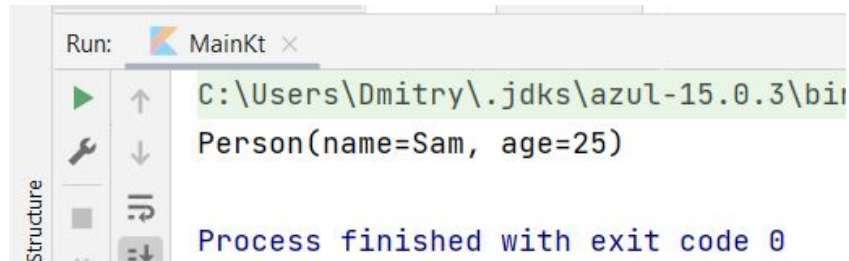


Пример data класса

Kotlin поддерживает объектно-ориентированную парадигму программирования, а это значит, что программу на данном языке можно представить в виде взаимодействующих между собой объектов.

```
data class Person(var name: String, var age: Int)
```

```
fun main() {  
    var sam: Person = Person("Sam", 25)  
    println(sam.toString())  
}
```



Решение задач

1. Дано число N - количество целочисленных элементов массива A . Заполнить массив случайными числами от -10 до 10. Вывести на экран. Затем посчитать количество положительных элементов.
2. Написать мобильное приложения счетчик нажатий на кнопку под Андроид на языке Котлин. Также по нажатию на кнопки необходимо менять цвет экрана в зависимости от нажатой кнопки. (Пример рис 1.)



рис 1.

Полезные ссылки

1. <https://kotlinlang.org>
2. <https://developer.android.com/kotlin>

Ссылка на презентацию и приложение:

<https://github.com/dmitriybulanov/ItFestMobDevKotlin>

Почта: *bulanov.dmitry@itcube61.ru*

