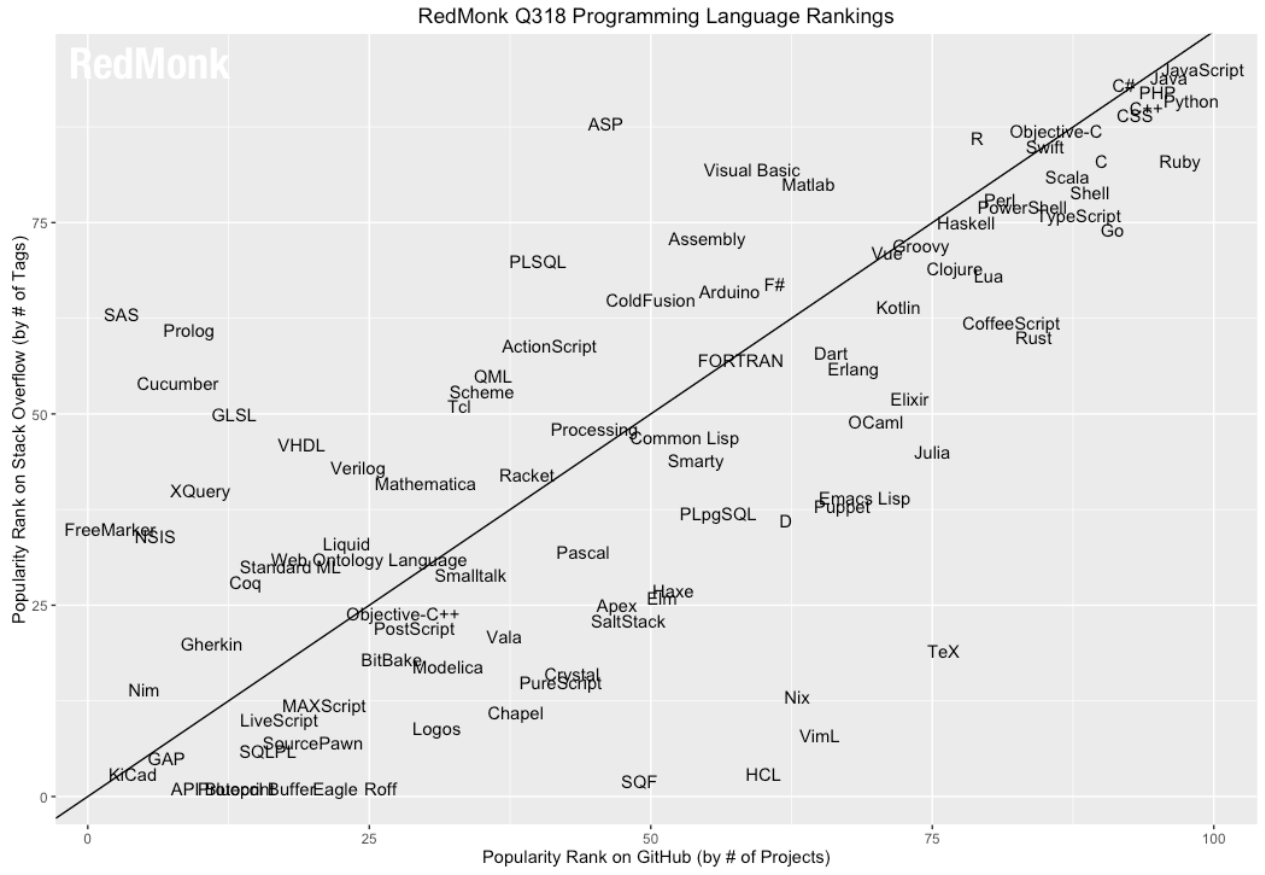
Kotlin для науки

Часть I

Языки и мифология

Больше языков: хороших и разных



- Существуют тысячи языков программирования.
- Из них около сотни активно используется.
- Нет способа определить, какой из них лучше.
- Разные языки хороши для разных задач.
- Программист должен знать несколько языков и иметь понятие об остальных.

- «Рейтингам языков программировании» верить нельзя

Что там с наукой?

- Python – его все знают, он хорош для использования существующих библиотек, но не для разработки.
- Java – самый популярный язык в мире. Есть претензии к многословности.
- C++ - самый гибкий, но в то же время один из самых сложных языков программирования.
- R – специализированный язык для статистических расчетов. Скорее всего будет съеден Python.
- Matlab – специализированный проприетарный язык. Скорее всего будет съеден Python.
- Scala – язык на JVM, разработанный учеными. Много возможностей, но дикая сложность.
- Julia – сравнительно молодой язык. Может служить хорошей альтернативой Python в будущем.
- Haskell – а вы пробовали?
- Kotlin – очень молодой язык на JVM. Полностью совместим с Java – библиотеками и синтактически приятнее Java. Вероятно будет плавно вытеснять другие JVM языки в будущем.

Немного компьютерной мифологии

Для того, чтобы программировать, надо знать
дискретную математику / теорию категорий / другую сложную науку

На самом деле нет.

В подавляющем большинстве задач специальная математика не нужна.
Разве что вы будете писать компилятор.

Для того, чтобы программировать, надо знать алгоритмы

На самом деле далеко не всегда.

Большинство алгоритмов уже написаны в библиотеках.
Те, что вдруг надо будет писать самостоятельно, можно посмотреть в википедии.

Настоящий программист должен работать в VIM

На самом деле нет.

Это все равно, что говорить, что настоящие детали должны быть выточены вручную. Если человек хвастается, что он пишет в VIM, с большой вероятностью для него программирование – хобби.

Настоящий программист должен работать в VIM

На самом деле нет.

Это все равно, что говорить, что настоящие детали должны быть выточены вручную. Если человек хвастается, что он пишет в VIM, с большой вероятностью для него программирование – хобби.

Настоящий программист должен работать на Linux

На самом деле не обязательно.

Работать он может где угодно. Но среди пользователей доля Linux и Mac OS составляет меньше 10%.

Для изучения надо выбрать самый крутой язык

Такой не придумали.

Программисты могут до опупения спорить о том, что язык А лучше языка Б, но единого мнения нет. Каждый язык хорош в своей нише.

Программа для науки может быть написана только на Fortran/C/C++, потому что они быстрые.

Это в корне не верно.

Нет понятия «скорость языка». Есть оптимизация компилятора. Скорость конечной программы зависит от многих факторов. В случае C/Fortran язык играет не положительную роль.

Более подробно разберем это отдельно

Если вы прослушаете этот (или другой) курс лекций, вы научитесь программировать

Неа.

Программирование – это не наука, а навык. Навык можно получить только путем его практического применения.

Java медленная

Не правда.

Во-первых, Java – это язык. Речь идет о конкретной реализации JVM. Современные реализации JVM в смысле оптимизации после разогрева не уступают нативному коду.

Java ест много памяти

Ну вот это возможно.

Реализация ссылок в JVM, а также специфика garbage collection подразумевает использование бОльшего объема памяти.

И все-таки, почему Kotlin?

Как Java, только лучше

- Все многообразие библиотек для Java без дополнительных усилий.
- Существенное расширение возможностей функционального программирования.
- Самый лучший инструментарий.

Широкие возможности compile-time оптимизации

Функции-расширения

Inline-функции

Inline-классы

Крайние меры

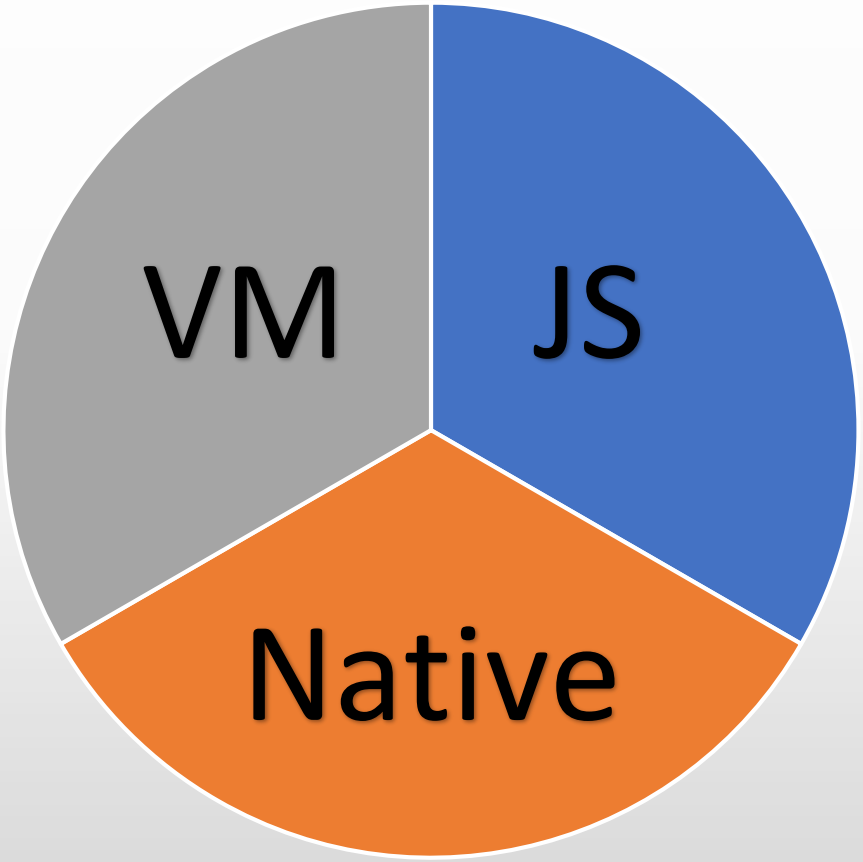
Если не получится с наукой, можно пойти в Android

android

Kotlin

Проблема совместимости

Проблема



- (J)VM используется для фреймворков и параллельных вычислений.
- JavaScript для визуализации данных
- Нативные библиотеки для высоко производительных вычислений.

Код с одной платформы не совместим с кодом с другой.

Kotlin multiplatform

- Компилятор Kotlin позволяет использовать общий код в JVM, JS (начиная с 1.2) и LLVM (начиная с 1.3) приложениях.
- Библиотека состоит из общего кода и специального кода для каждой из платформ.
- Уже есть полная совместимость с JS
- Совместимость с нативными приложениями на уровне C API.
- Планируется создание собственного промежуточного представления.

Полезные полезности

- https://kotlinlang.org/docs/reference/ - документация и справочники
- https://kotlinlang.ru/ - перевод документации на русский
- https://play.kotlinlang.org – онлайн-версия для простых примеров
- https://play.kotlinlang.org/byExample/overview - интерактивные примеры
- https://play.kotlinlang.org/koans - минимальные обучающие примеры
- https://www.jetbrains.com/idea/ - IDE