AN INTRODUCTION TO KOTLIN BY EXAMPLE

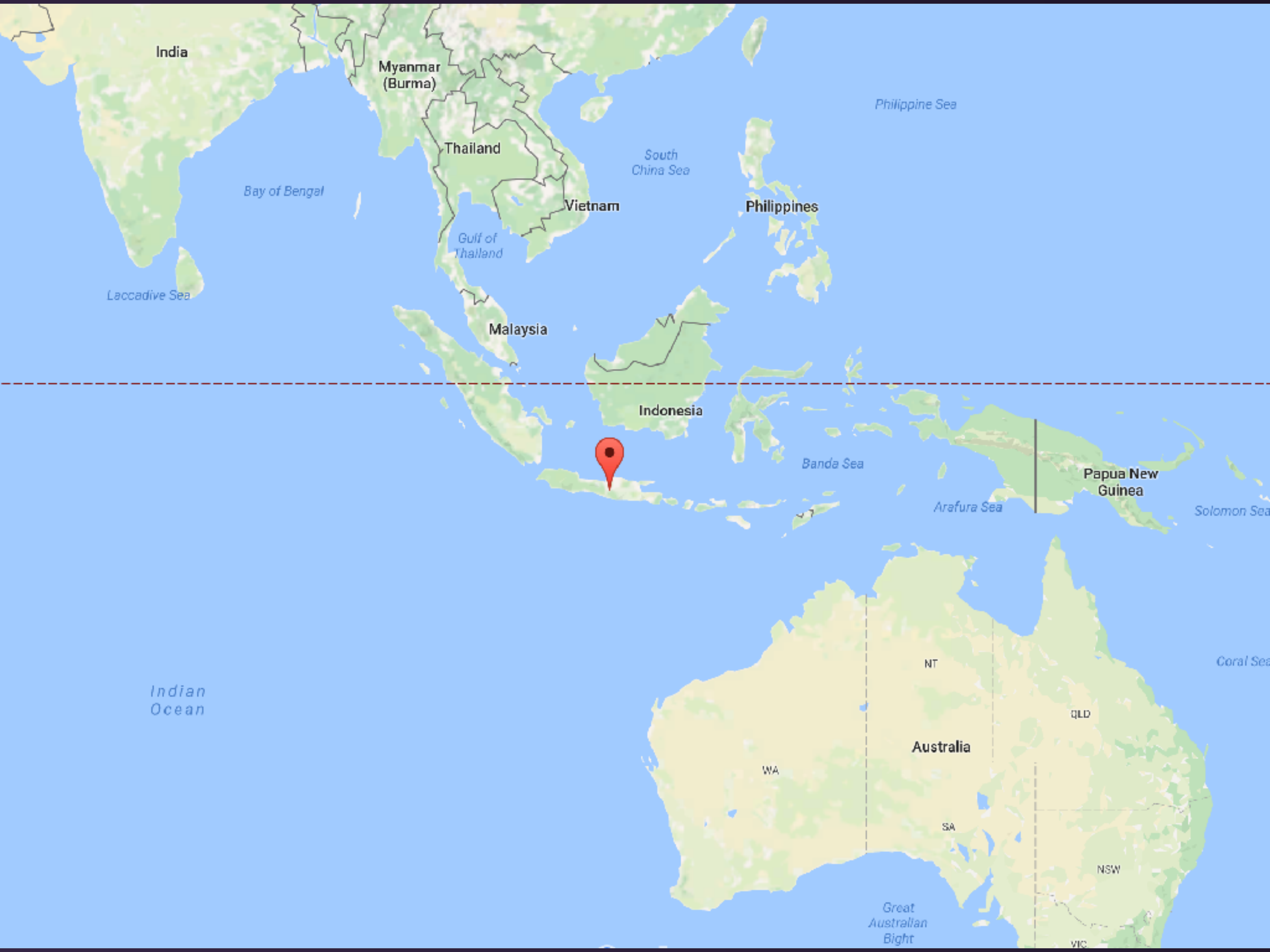
@dmitrykandalov

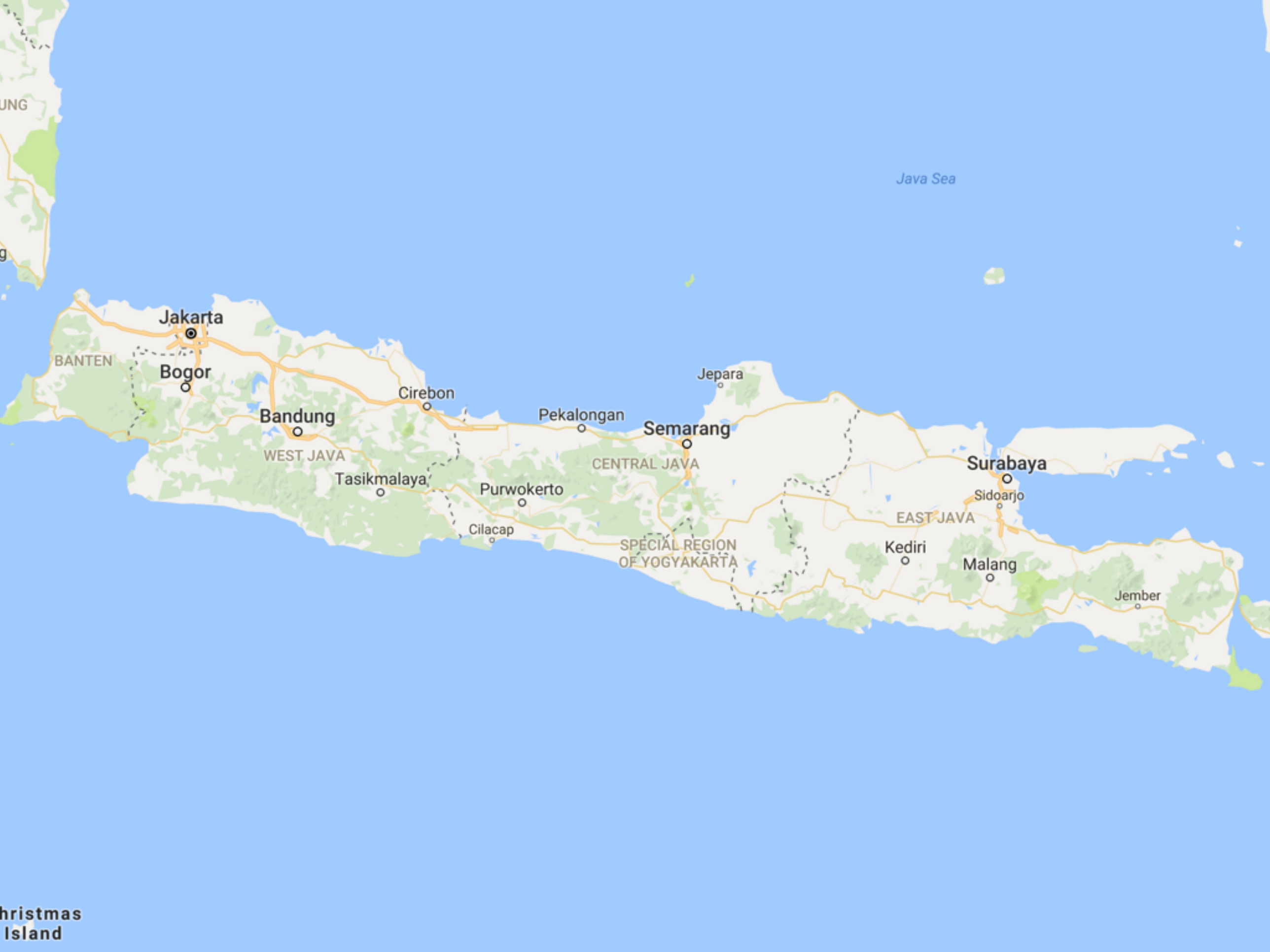
DROPBOX LINK

https://goo.gl/fvT743

JAVA

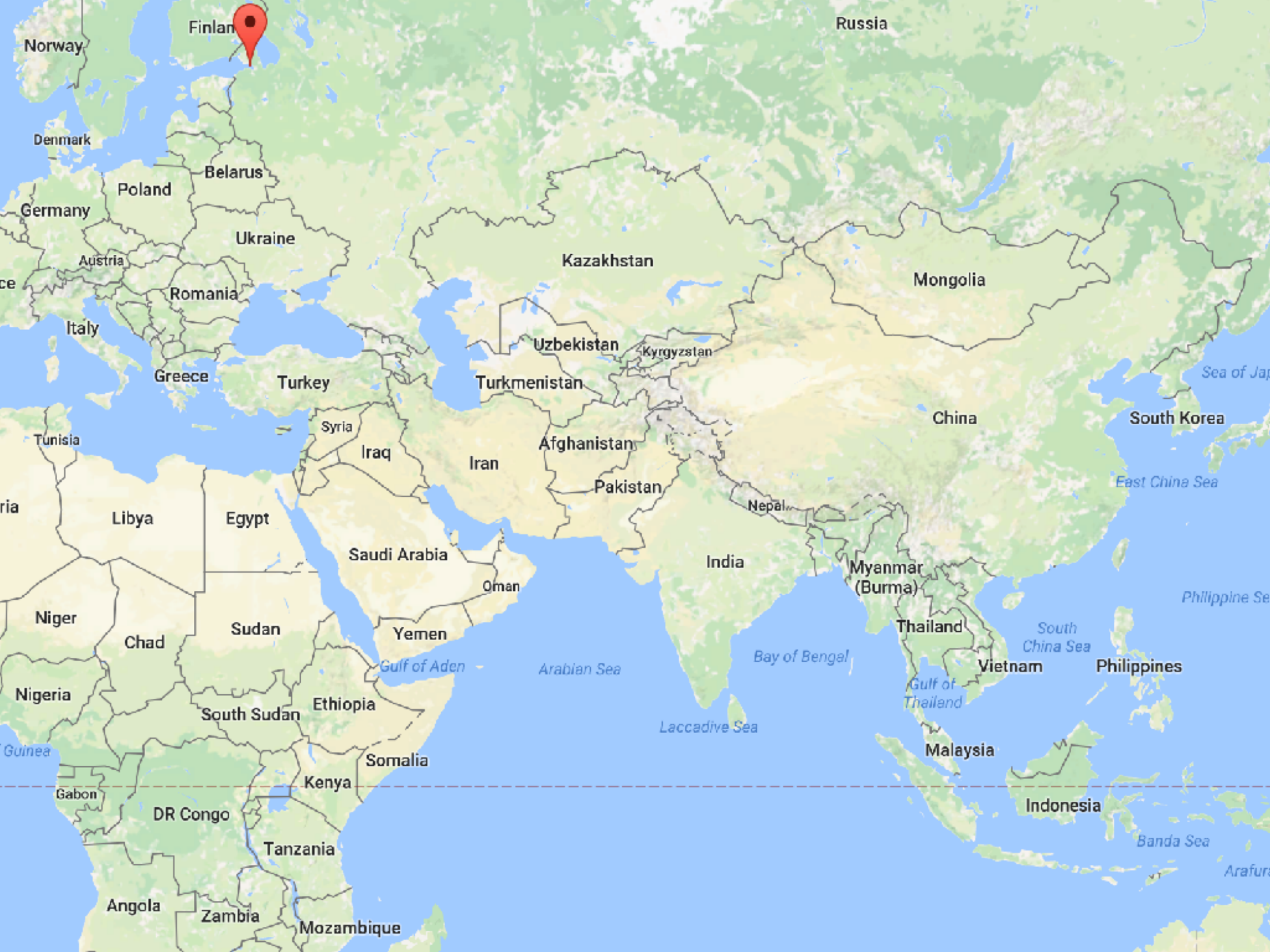

Norway
Finland
Russia
Denmark
Belarus
Poland
Germany
Ukraine
Austria
Romania
Italy
Greece
Turkey
Kazakhstan
Mongolia
Uzbekistan
Kyrgyzstan
Turkmenistan
China
Sea of Japan
South Korea
Japan
Syria
Iraq
Iran
Afghanistan
Pakistan
Nepal
Tunisia
Libya
Egypt
Saudi Arabia
Oman
India
Myanmar
(Burma)
Thailand
Vietnam
Philippines
Philippine Sea
Bay of Bengal
Arabian Sea
Gulf of Aden
Gulf of
Thailand
Laccadive Sea
Malaysia
Niger
Chad
Sudan
Yemen
Nigeria
South Sudan
Ethiopia
Somalia
Kenya
Gabon
DR Congo
Tanzania
Angola
Zambia
Mozambique
Namibia
Zimbabwe
Botswana
Madagascar
Indian
Ocean
Banda Sea
Arafura Sea
NT
QLD


India
Myanmar
(Burma)
Thailand
Vietnam
Philippines
Philippine Sea
South
China Sea
Bay of Bengal
Gulf of
Thailand
Laccadive Sea
Malaysia
Indonesia
Banda Sea
Arafura Sea
Papua New
Guinea
Solomon Sea
Coral Sea
Indian
Ocean
NT
QLD
Australia
WA
SA
NSW
Great
Australian
Bight
VIC

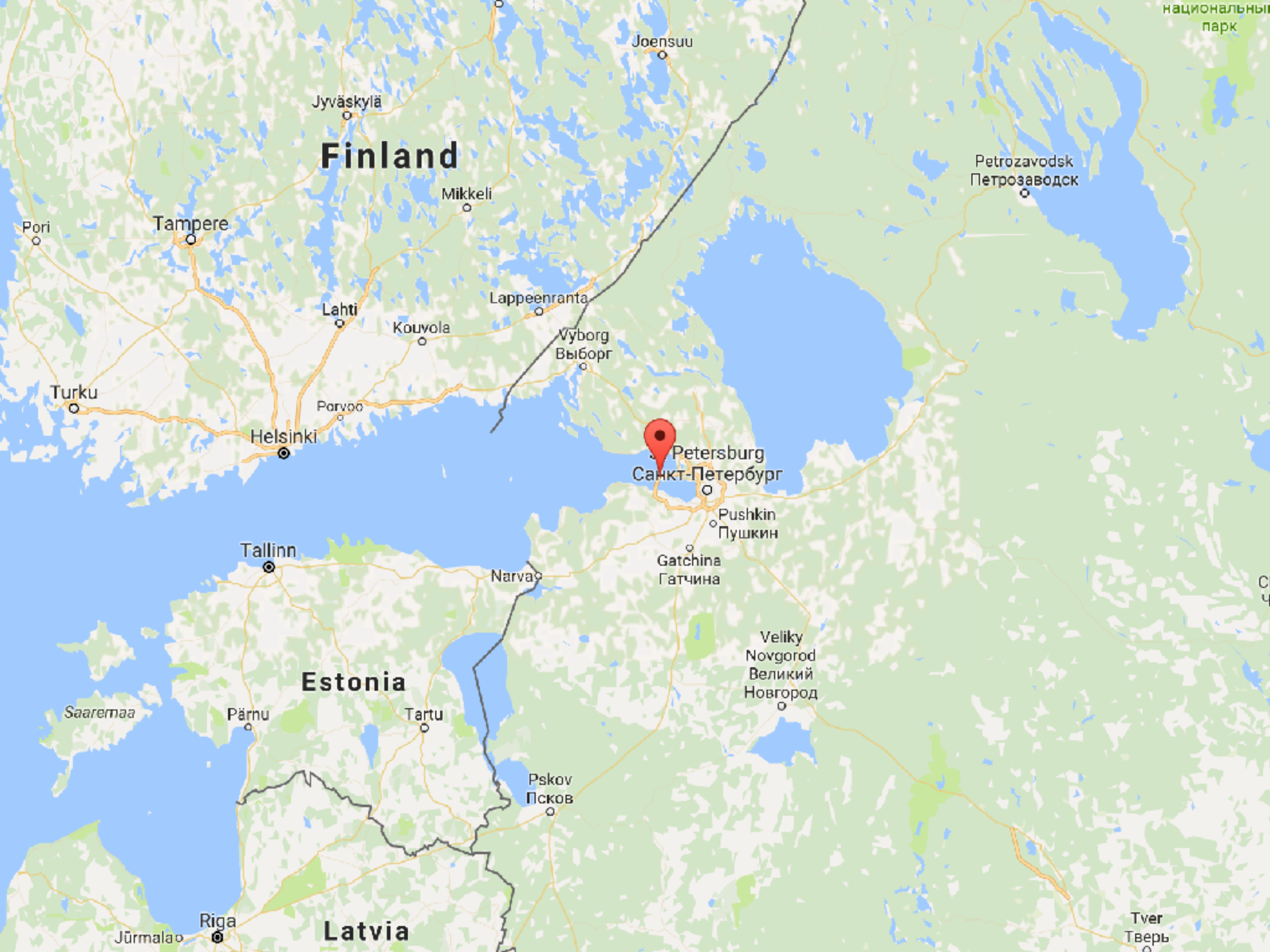

Java Sea
Jakarta
BANTEN
Bogor
Cirebon
Bandung
WEST JAVA
Pekalongan
Jepara
Semarang
CENTRAL JAVA
Tasikmalaya
Purwokerto
Cilacap
SPECIAL REGION
OF YOGYAKARTA
Surabaya
Sidoarjo
EAST JAVA
Kediri
Malang
Jember

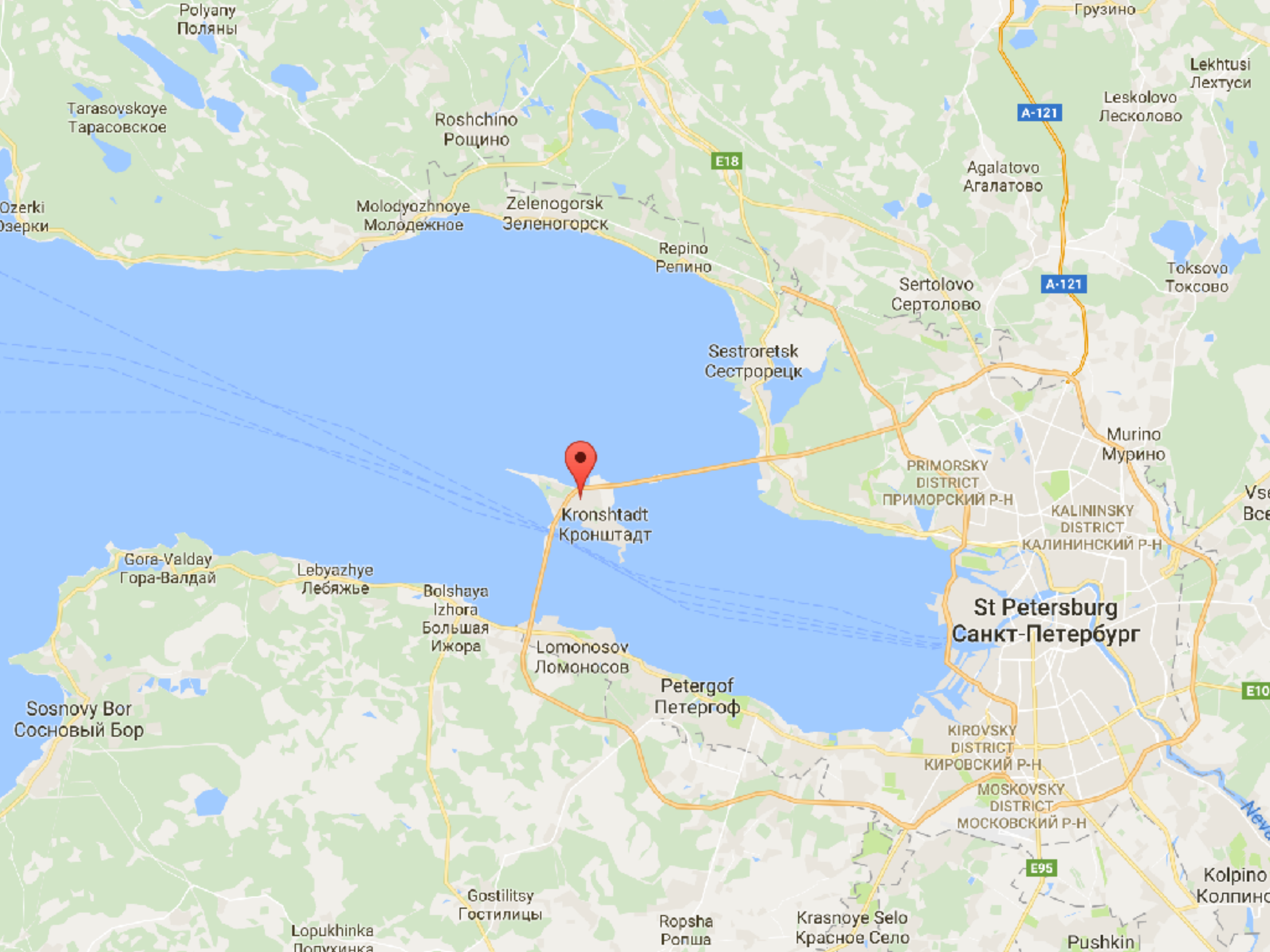

javac
Java Sea
Jakarta
BANTEN
Bogor
Cirebon
Bandung
WEST JAVA
Tasikmalaya
Cilacap
Pekalongan
Jepara
Semarang
CENTRAL JAVA
SPECIAL REGION
OF YOGYAKARTA
Purwokerto
Surabaya
Sidoarjo
EAST JAVA
Kediri
Malang
Jember
hristmas
Island

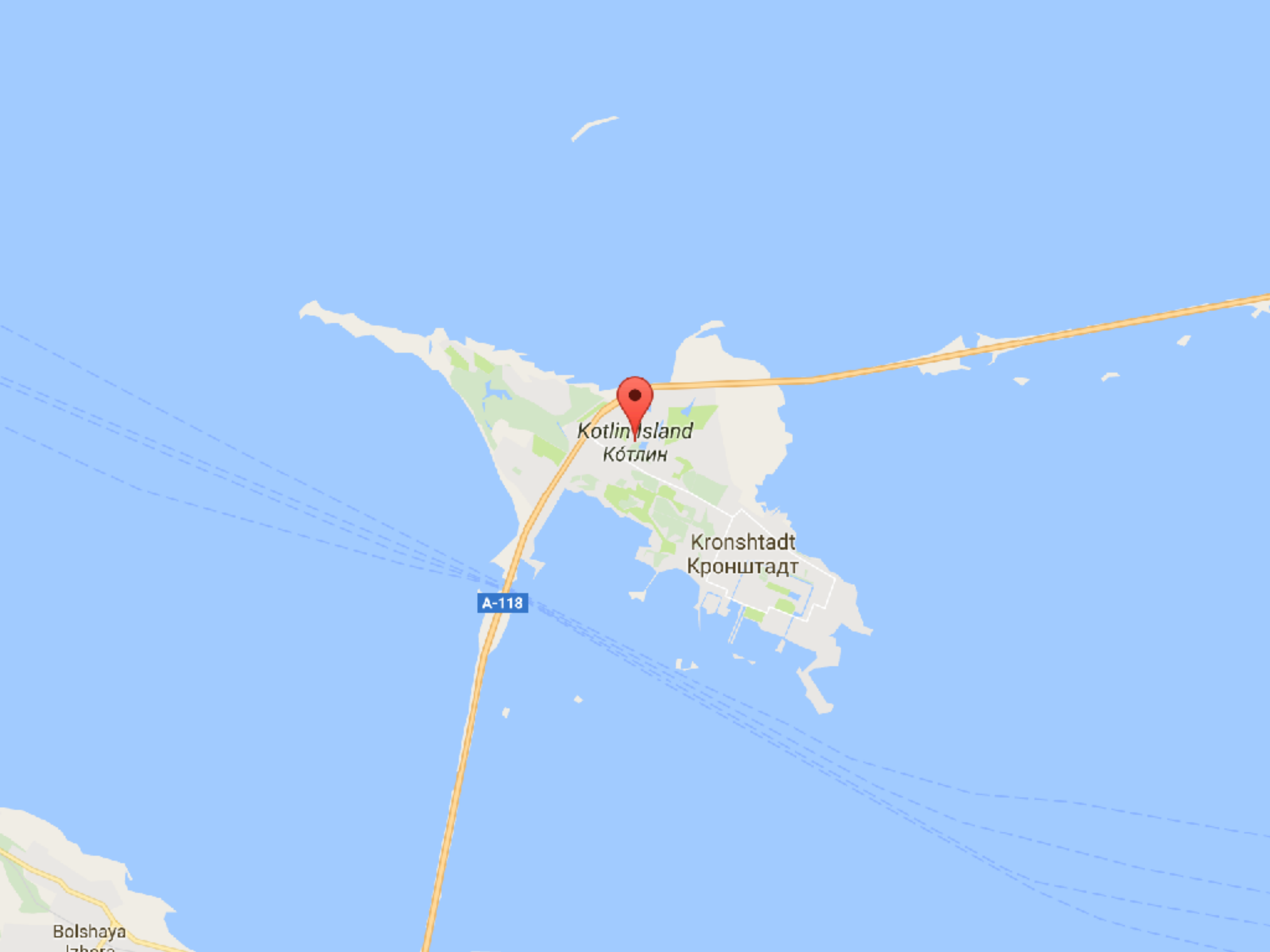
KOTLIN


Finland
Norway
Russia
Denmark
Belarus
Poland
Germany
Ukraine
Austria
Romania
Italy
Kazakhstan
Mongolia
Uzbekistan
Kyrgyzstan
Turkmenistan
Greece
Turkey
Syria
Iraq
Iran
Afghanistan
Pakistan
China
South Korea
Sea of Japan
Tunisia
Libya
Egypt
Saudi Arabia
Oman
Yemen
India
Nepal
Myanmar (Burma)
Thailand
Vietnam
Philippines
Philippine Sea
South China Sea
Bay of Bengal
Arabian Sea
Gulf of Aden
Laccadive Sea
Gulf of Thailand
Malaysia
Indonesia
Banda Sea
East China Sea
Niger
Chad
Sudan
Nigeria
South Sudan
Ethiopia
Somalia
Kenya
Gabon
DR Congo
Tanzania
Angola
Zambia
Mozambique

Sweden
Norway
Finland
Baltic Sea
Estonia
Latvia
Lithuania
Moscow
Москва
Denmark
Poland
Belarus
herlands
Berlin
Germany
Prague
Czech Rep


Finland
Joensuu
Jyväskylä
Mikkeli
Pori
Tampere
Lahti
Kouvola
Lappeenranta
Vyborg
Выборг
Turku
Porvoo
Helsinki
Petrozavodsk
Петрозаводск
национальный
парк
Petersburg
Санкт-Петербург
Pushkin
Пушкин
Gatchina
Гатчина
Narva
Tallinn
Estonia
Saaremaa
Pärnu
Tartu
Pskov
Псков
Veliky
Novgorod
Великий
Новгород
Riga
Jūrmala
Latvia
Tver
Тверь


Polyany
Поляны
Tarasovskoye
Тарасовское
Roshchino
Рощино
E18
Ozerki
Озерки
Molodyozhnoye
Молодежное
Zelenogorsk
Зеленогорск
Repino
Репино
Грузино
Lekhtusi
Лехтуси
Leskolovo
Лесколово
A-121
Agalatovo
Агалатово
Sertolovo
Сертолово
A-121
Toksovo
Токсово
Sestroretsk
Сестрорецк
Murino
Мурино
PRIMORSKY
DISTRICT
ПРИМОРСКИЙ Р-Н
KALININSKY
DISTRICT
КАЛИНИНСКИЙ Р-Н
Kronshtadt
Кронштадт
Gora-Valday
Гора-Валдай
Lebyazhye
Лебяжье
Bolshaya
Izhora
Большая
Ижора
Lomonosov
Ломоносов
St Petersburg
Санкт-Петербург
Sosnovy Bor
Сосновый Бор
Petergof
Петергоф
KIROVSKY
DISTRICT
КИРОВСКИЙ Р-Н
MOSKOVSKY
DISTRICT
МОСКОВСКИЙ Р-Н
E95
Kolpino
Колпино
Gostilitsy
Гостилицы
Lopukhinka
Лопухинка
Ropsha
Ропша
Krasnoye Selo
Красное Село
Pushkin


Kotlin Island
Ко́тлин
Kronshtadt
Кронштадт
A-118
Bolshaya


230

STATICALLY TYPED PRAGMATIC LANGUAGE FOR JVM/JS

JetBrains/kotlin: The Kotlin Pro

GitHub, Inc. [US] https://github.com/JetBrains/kotlin

This repository Search Pull requests Issues Gist

JetBrains / kotlin

Watch 433 Star 6,860 Fork 731

Code Pull requests 23 Projects 0 Pulse Graphs

The Kotlin Programming Language http://kotlinlang.org/

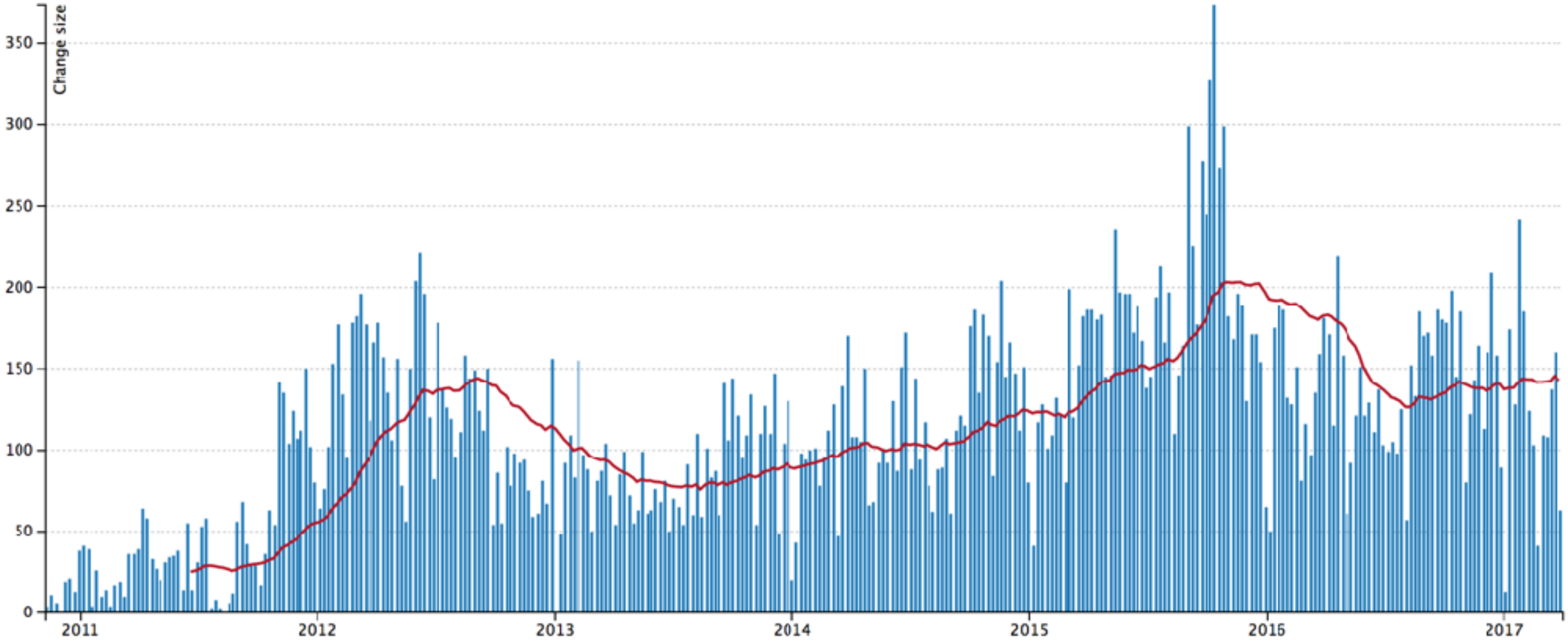
36,655 commits 695 branches 7,237 releases 140 contributors

Branch: master New pull request Create new file Upload files Find file Clone or download

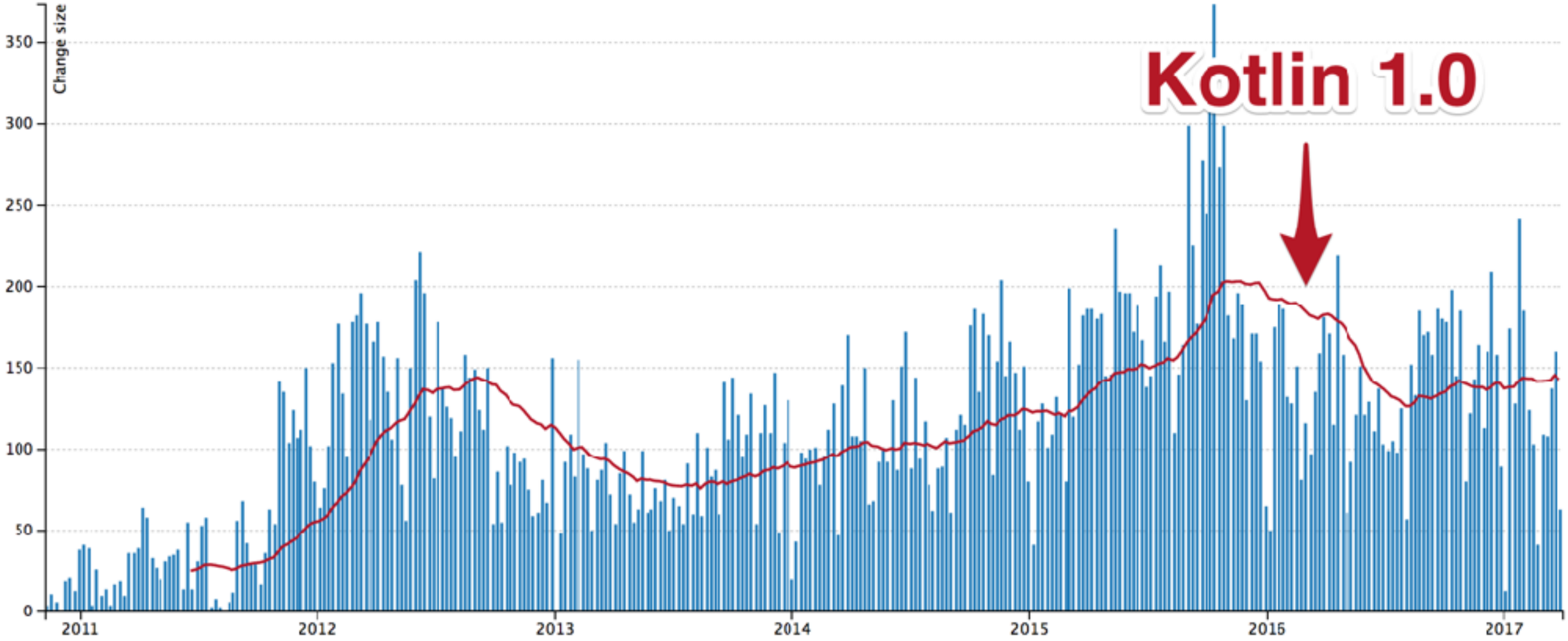
ilya-g Do not generate headers for JS-only functions. Latest commit a429992 an hour ago

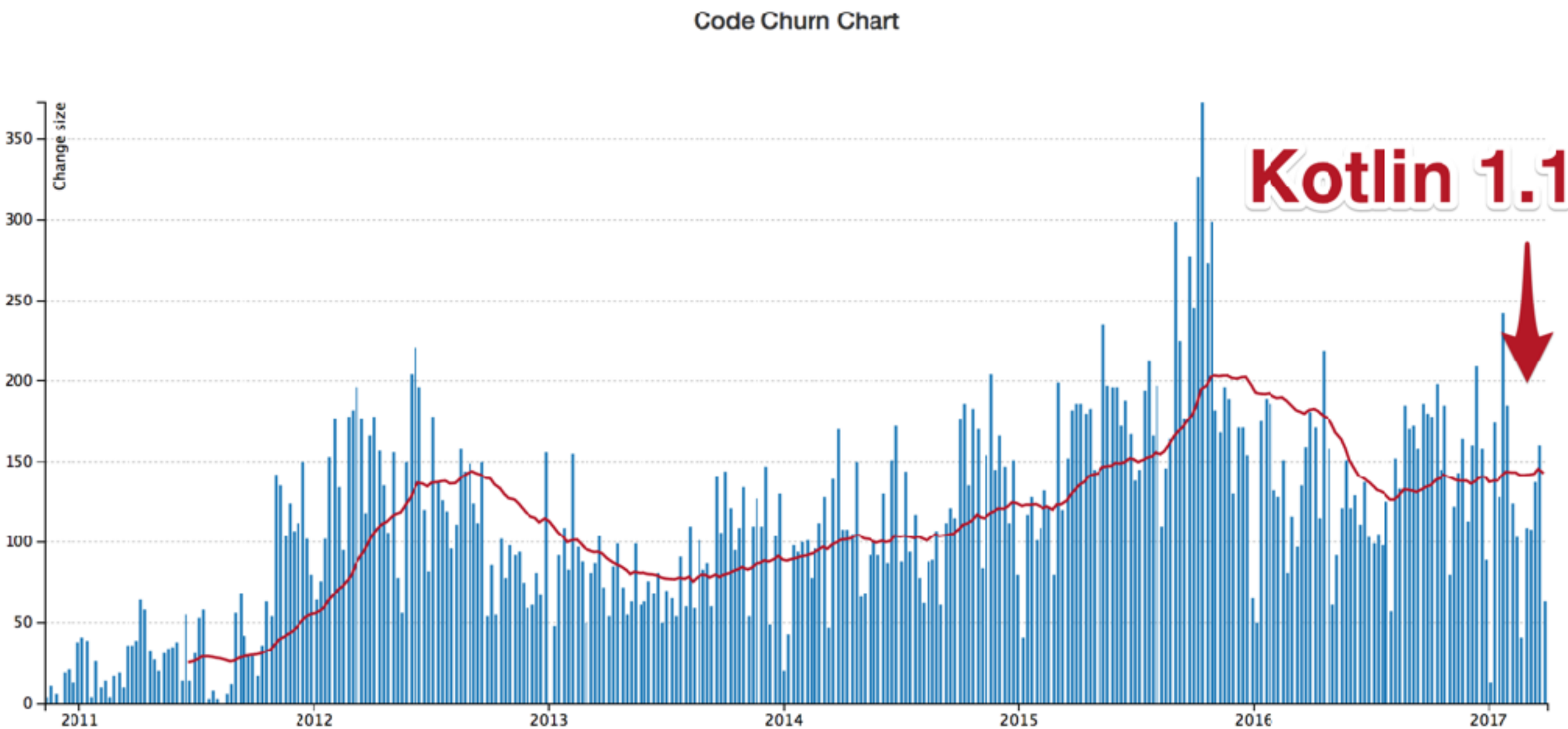
.idea	Fix look up for breakpoint in crossinline lambdas in android (KT-15282)	2 days ago
android-studio	Prepare building plugins modules against 1.8 JDK	10 months ago
annotations	Remove KotlinSignature annotations from project code	a year ago
ant	CLI, Ant: add kotlin-reflect.jar to classpath by default, support "-n...	5 months ago
build-common	Refactor daemon interface	5 days ago
compiler	Fix typo	2 hours ago
core	Do not load PlatformDependent annotated built-in members with lv=1.0	10 hours ago
eval4j	Eval4J: loadClass should return prepared class, otherwise ClassNotPre...	5 months ago
generators	KJS: relocate com.google.dart.compiler -> org.jetbrains.kotlin	9 days ago
gradle/wrapper	add Gradle wrapper to project	2 months ago
grammar	Fix companion object definition in the grammar (KT-8566)	2 months ago
idea-runner	Prepare building plugins modules against 1.8 JDK	10 months ago
idea	Stop using idea isNullOrEmpty function (KT-15786)	5 hours ago

Code Churn Chart

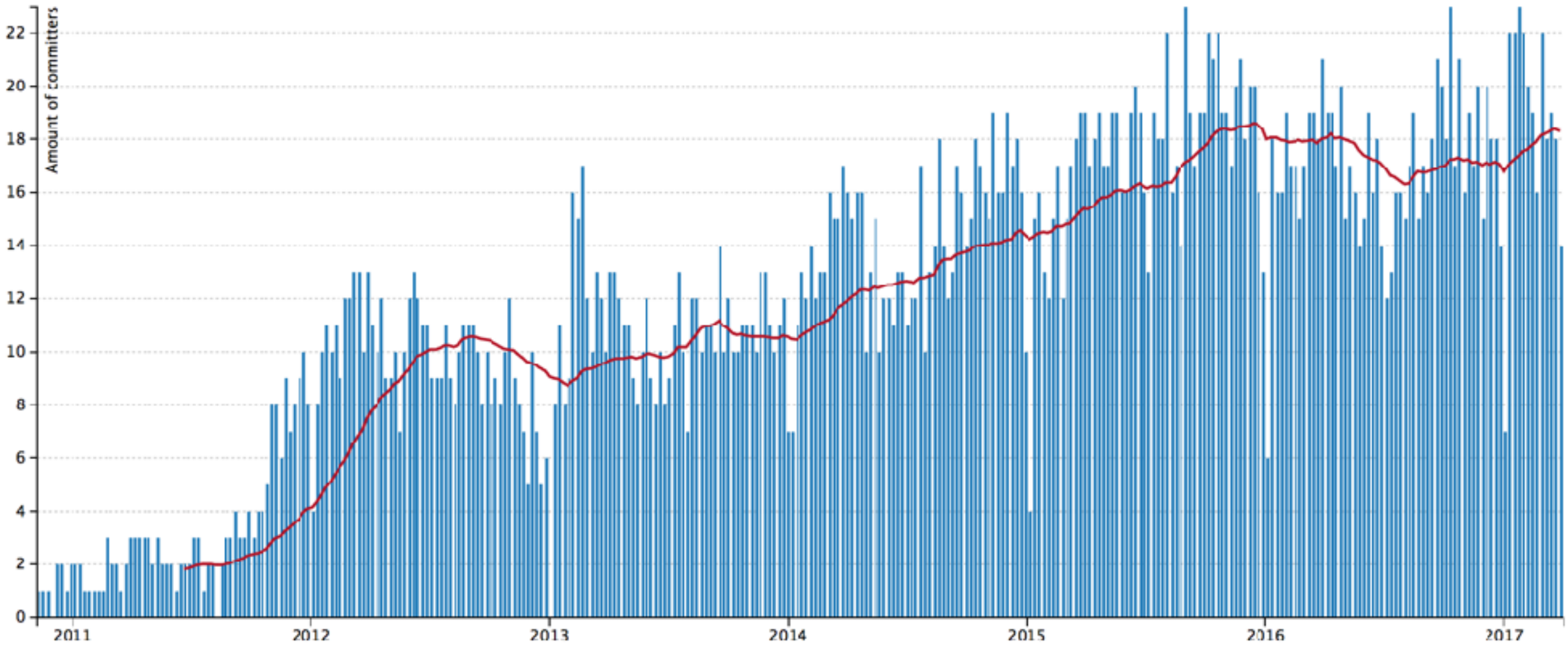


Code Churn Chart




Code Churn Chart
Change size
350
300
250
200
150
100
50
0
2011
2012
2013
2014
2015
2016
2017
Kotlin 1.1

Amount Of Committers Chart



WHY NOT JAVA?

WHY NOT SCALA?

WHY NOT GROOVY?

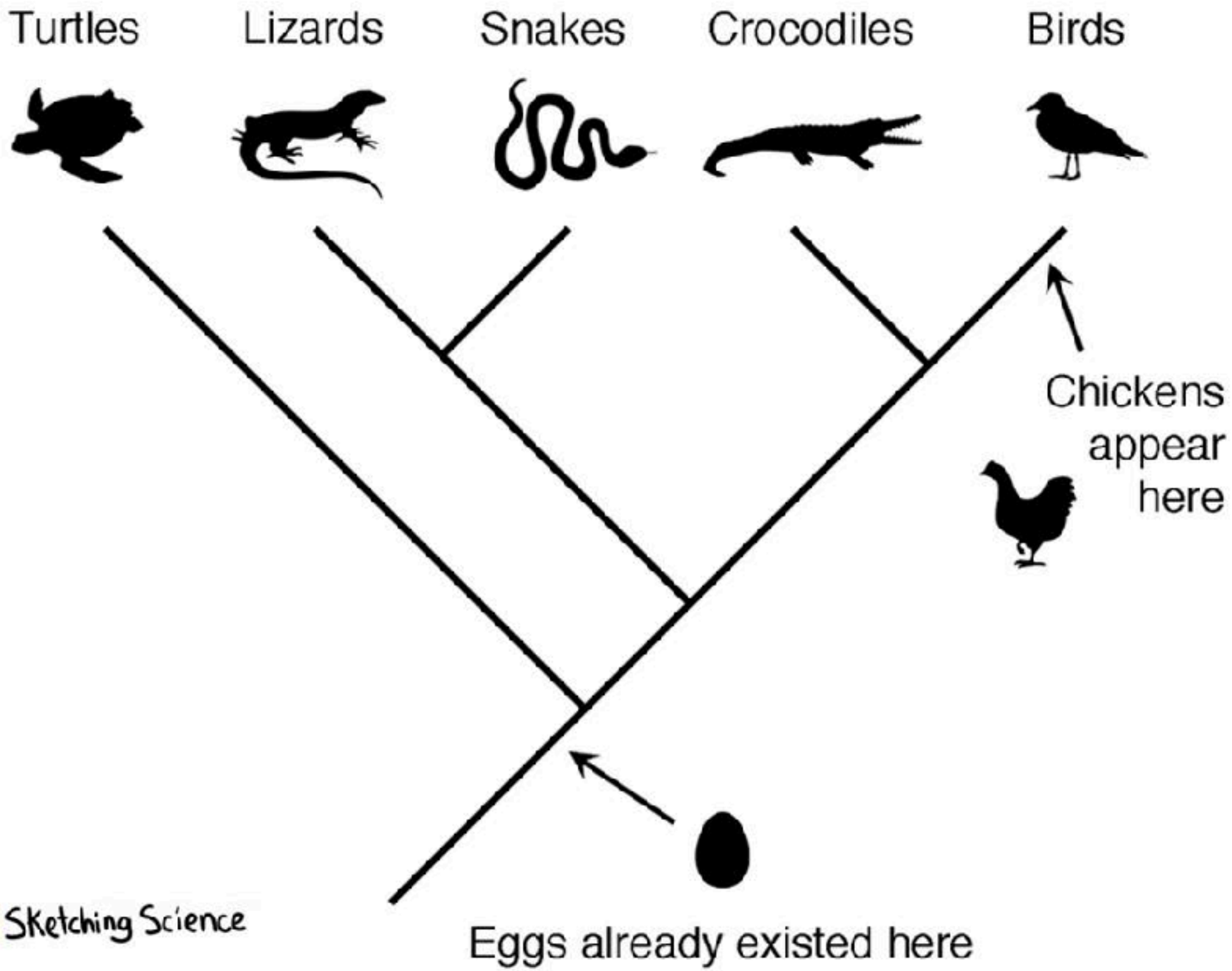
WHY KOTLIN?

EXAMPLES

HELLO WORLD

- BASIC SYNTAX
- STRING ABSTRACTION OVER JAVA/JS
- VAL VS VAR
- NAMED PARAMETERS
- EXTENSION FUNCTIONS
- INTEGRATION WITH JAVA

Which came first, the chicken or the egg?



SKetching Science

FACTORIAL

Factorial

From Wikipedia, the free encyclopedia

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Wikipedia store

Interaction
Help
About Wikipedia
Community portal
Recent changes
Contact page

Tools
What links here
Related changes
Upload file
Special pages
Permanent link

In mathematics, the **factorial** of a non-negative integer *n*, denoted by *n*!, is the product of all positive integers less than or equal to *n*. For example,

$$5! = 5 \times 4 \times 3 \times 2 \times 1 = 120.$$

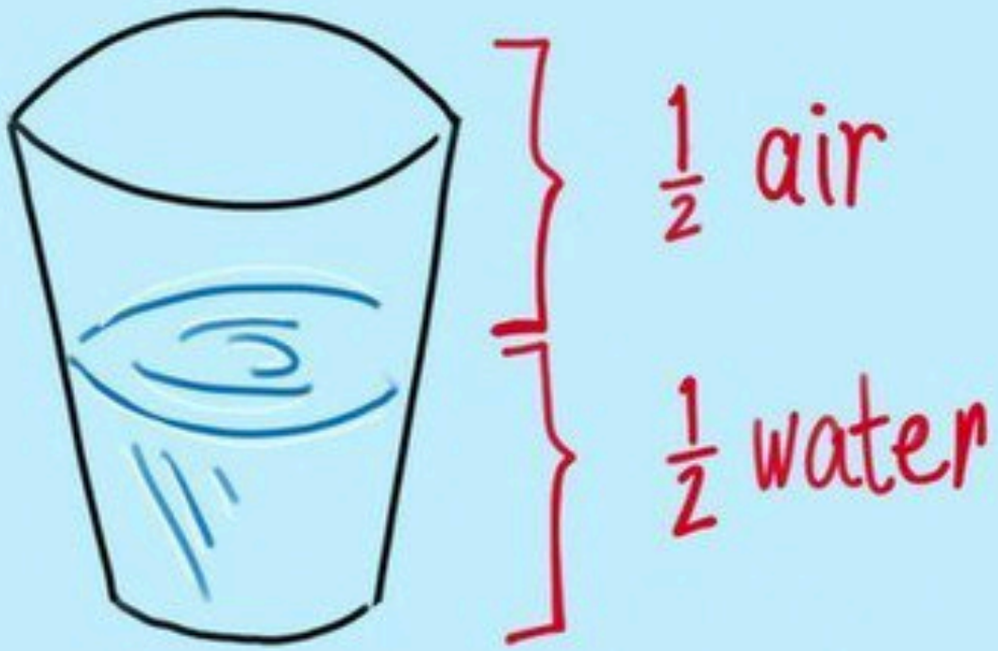
The value of 0! is 1, according to the convention for an empty product.[1]

The factorial operation is encountered in many areas of mathematics, notably in combinatorics, algebra, and mathematical analysis. Its most basic occurrence is the fact that there are *n*! ways to arrange *n* distinct objects into a sequence (i.e., permutations of the set of objects). This fact was known at least as early as the 12th century, to Indian scholars.[2] Fabian Stedman, in 1677, described factorials as applied to change ringing.[3] After describing a recursive approach, Stedman gives a statement of a factorial (using the language of the original):

Selected members of the sequence (sequence A000142 in OEIS); values specified in exponential notation are rounded to 6 significant precisi

n	
0	
1	
2	
3	
4	
5	
6	
7	
8	

- INT/LONG ABSTRACTION OVER JAVA/JS
- ALL NUMBERS ARE OBJECTS
- NO `NEW` KEYWORD
- OPERATOR OVERLOADING
- TAIL CALL OPTIMISATION
- UNIT TESTING



technically,
the glass is always
full.

DATA MUNGING KATA

The Pragmatic Programmer

from journeyman
to master

Andrew Hunt
David Thomas

Foreword by Ward Cunningham

CodeKata

Because experience
is the *only* teacher

CodeKata

(7 Comments)

How do you get to be a great musician? It helps to know the theory, and to understand the mechanics of your instrument. It helps to have talent. But ultimately, greatness comes from practicing; applying the theory over and over again, using feedback to get better every time.

How do you get to be an All-Star sports person? Obviously fitness and talent help. But the great athletes spend hours and hours every day, practicing.

But in the software industry we take developers trained in the theory and throw them straight in to the deep-end, working on a project. It's like taking a group of fit kids and telling them that they have four quarters to beat the Redskins (hey, we manage by objectives,

Recent Posts

CodeKata

CodeKata: How It Started

Kata, Kumite, Koan, and Dreyfus

Kata01: Supermarket Pricing

Kata02: Karate Chop

Kata03: How Big? How Fast?

Kata04: Data Munging

Kata05: Bloom Filters

Kata06: Anagrams

Kata07: How'd I Do?

Kata08: Conflicting Objectives

Kata09: Back to the Checkout

Kata10: Hashes vs. Classes

Kata11: Sorting It Out

Kata12: Best Sellers

Kata04: Data Munging

Martin Fowler gave me a hard time for Kata02, complaining that it was yet another single-function, academic exercise. Which, or course, it was. So this week let's mix things up a bit.

Here's an exercise in three parts to do with real world data. Try hard not to read ahead—do each part in turn.

Part One: Weather Data

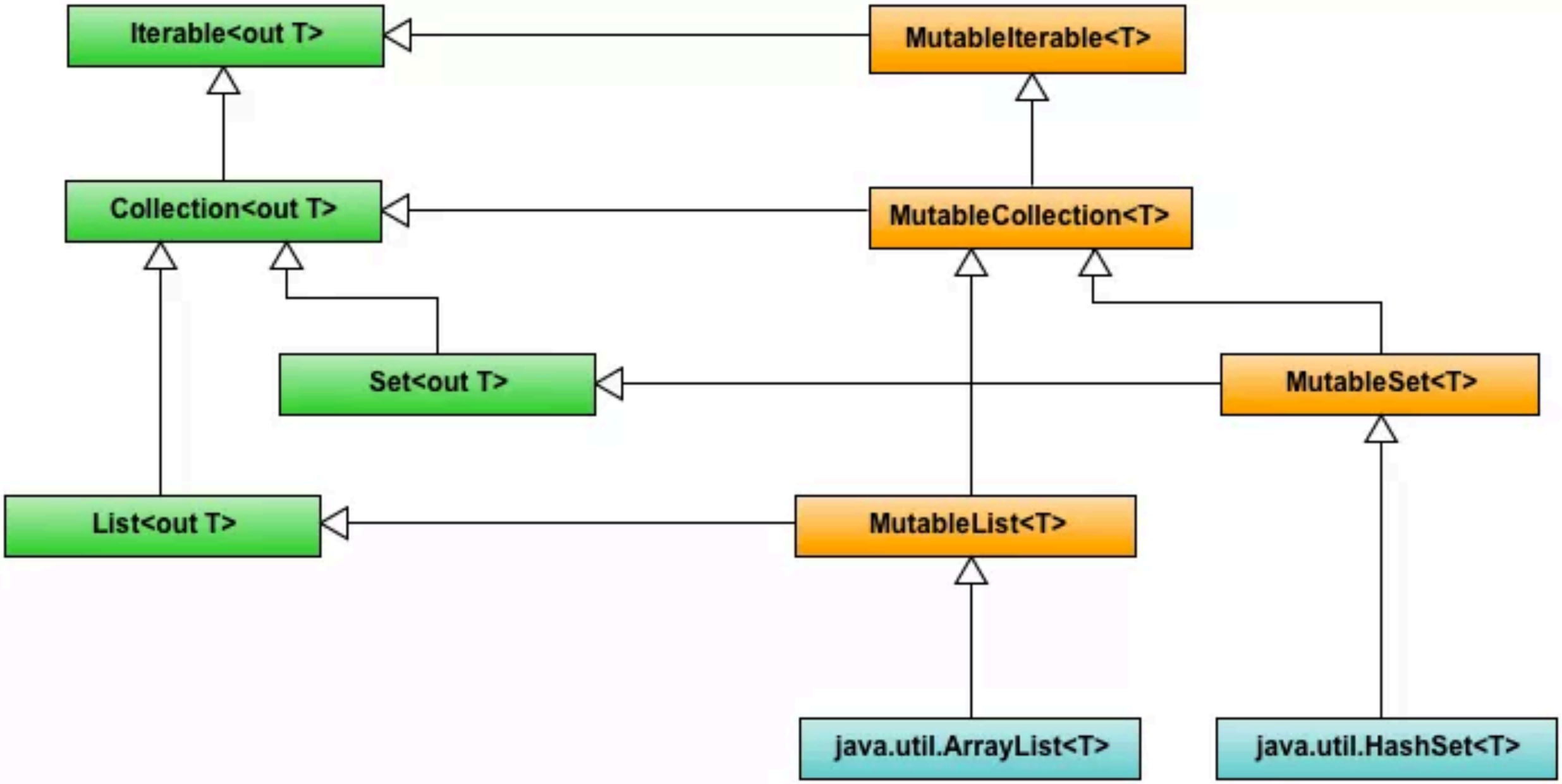
In weather.dat you'll find daily weather data for Morristown, NJ for June 2002. Download this text file, then write a program to output the day number (column one) with the smallest temperature spread (the maximum temperature is the second column, the minimum the third column).

Part Two: Soccer League Table

The file football.dat contains the results from the English Premier League for 2001/2. The columns labeled 'F' and 'A' contain the total number of goals scored for and against each team in that season (so Arsenal scored 79 goals against opponents, and had 36 goals scored against them). Write a program to print the name of the team with the smallest difference in 'for' and 'against' goals.

Part Three: DRY Fusion

Take the two programs written previously and factor out as much common code as possible, leaving you with two smaller programs and some kind of shared functionality.


Iterable<out T>
MutableIterable<T>
Collection<out T>
MutableCollection<T>
Set<out T>
MutableSet<T>
List<out T>
MutableList<T>
java.util.ArrayList<T>
java.util.HashSet<T>

- KOTLIN.COLLECTIONS
- LAMBDAS, HIGHER-ORDER FUNCTIONS
- NO CHECKED EXCEPTIONS
- DATA CLASSES
- NULLABLE TYPES
- KOTLINTEST


ZU 0666', 0, 0); DROP DATABASE TABLICE;

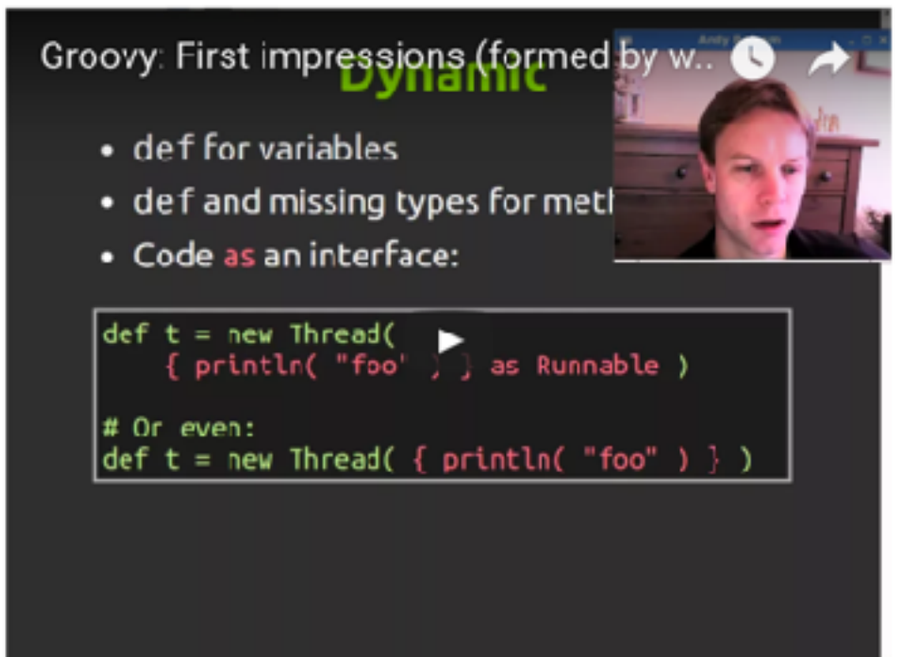
SNAKE

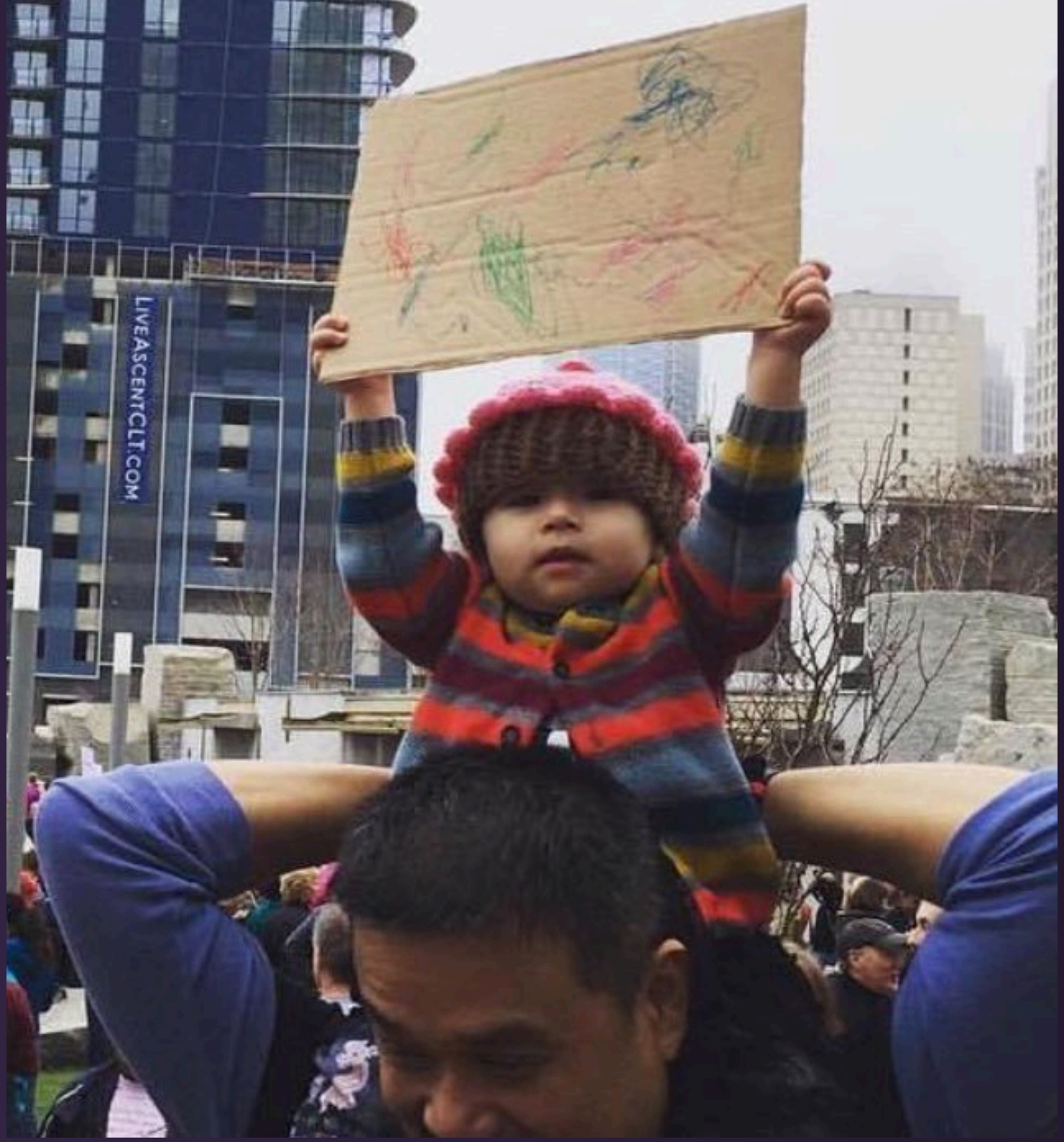
> Snake in Groovy

Series: Groovy, Ruby, BASIC, Dart, Elm, Python3+Qt5

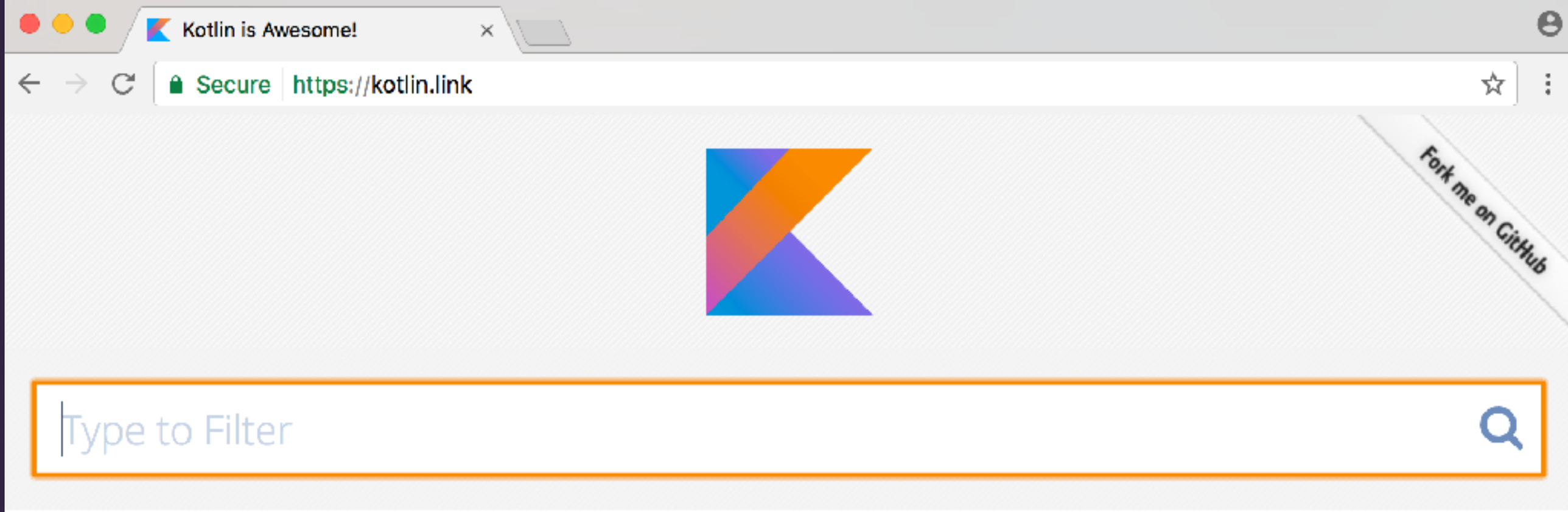
I'm starting a series where I write the game Snake in lots of programming languages.

I almost always use writing Snake as my way in to understand a new language, so I'll share my thoughts about each language as I go.




LIVEASCENTCLT.COM

KOTLIN LINKS



LINKS

Official Links

JetBrains/kotlin 6822 ★
Last update: Jan 15, 2017

Home Page

Language Reference

Slack (5500+ users)

Resources

/r/Kotlin

Stackoverflow Documentation on Kotlin

Quora Kotlin

Trending Kotlin on Github

Kotlin Programming Language

kotlinlang.org

Kotlin

LEARN COMMUNITY TRY ONLINE

Statically typed programming language for the JVM, Android and the browser

100% interoperable with Java™

TRY KOTLIN

Kotlin 1.1 Beta Is Here! read more

Why Kotlin?

Shortcuts | Convert from Java | Fullscreen

Kotlin Koans / Introduction / Hello, world! / Task.kt

- Examples
- Kotlin Koans 0/42
  - Introduction
    - Hello, world!
      - Task.kt
      - Test.kt
    - Java to Kotlin conversion
    - Named arguments
    - Default arguments
    - Lambdas
    - Strings
    - Data classes
    - Nullable types
    - Smart casts
    - Extension functions
    - Object expressions
    - SAM conversions
    - Extensions on collections
  - Conventions

Save | Save as | Arguments | JUnit | Run

Program arguments

Simple Functions

Take a look at function syntax and make the function `start` return the string `"OK"`.

In the tasks the function `TODO()` is used that throws an exception. Your job during the koans will be to replace this function invocation with a meaningful code according to the problem.

Check | Revert | Jamsue Moys

```
1 fun start(): String = "What about using static import for assertEquals in test code?"
```

Compilation completed successfully | On-the-fly type checking

- TestStart
  - testOk

Failed: 1 of 1 in 0.003s

```
org.junit.ComparisonFailure:

Expected: OK
Actual: What about using static import for assertEquals in test code?
    <click to see a difference>

    at org.junit.Assert.assertEquals(Assert.java:115)
    at org.junit.Assert.assertEquals(Assert.java:144)
    at TestStart.testOk(Test.kt:6)
```

Problems view | Console | Generated classfiles

This demo is running on Kotlin v. 1.0.6

Kotlin / **kotlin-koans**

Watch 52 | Unstar 579 | Fork 384

Code | Issues 1 | Pull requests 1 | Projects 0 | Pulse | Graphs

Kotlin workshop

147 commits | **3** branches | **0** releases | **26** contributors | MIT

Branch: **master** | **New pull request** | **Create new file** | **Upload files** | **Find file** | **Clone or download**

svtk Fixed an error in the example — Latest commit `a20601d` on 15 Dec 2016

.idea	Minor: idea configuration changes	4 months ago
gradle/wrapper	Minor: idea configuration changes	4 months ago
lib	Removed 'annotations.jar'	a year ago
src	Fixed an error in the example	2 months ago
test	Catch partial regexes in test cases	4 months ago
.classpath	Add Eclipse project files to allow easy import into Eclipse workspace	a year ago
.gitignore	Build project using Gradle	4 months ago
.project	Place Kotlin buider before Java builder for Eclipse settings	a year ago
LICENSE.txt	add MIT license	2 years ago
README.md	Added a reference to the documentation	4 months ago
build.gradle	upgrade to kotlin 1.0.5	3 months ago
gradlew	Build project using Gradle	4 months ago
gradlew.bat	Build project using Gradle	4 months ago

Ninety-Nine Kotlin Problems

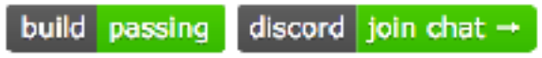


Table of Contents

Introduction

This an adaptation of Ninety-Nine Scala Problems by Phil Gold which itself is an adaptation of the Ninety-Nine Prolog Problems written by Werner Hett at the Berne University of Applied Sciences in Berne, Switzerland. Some problems have been altered to be more amenable to programming in Kotlin.

You might want to do these problems if you want to learn Kotlin, are interested in the problems described below, or both. The main reason to prefer this to using websites like hackerrank.com and codewars.com is that there is no vendor lock-in and no hidden agenda pursued by the website owner.

Suggested workflow is to solve a problem yourself and then compare solution to the one provided.
Solutions are available by clicking on the link at the beginning of the problem description. Your goal should be to find the most elegant solution to the given problems. Efficiency is important, but clarity is even more crucial. Some of the

TWEETS 685 | FOLLOWING 1 | FOLLOWERS 11.8K | LIKES 195

Follow

Kotlin

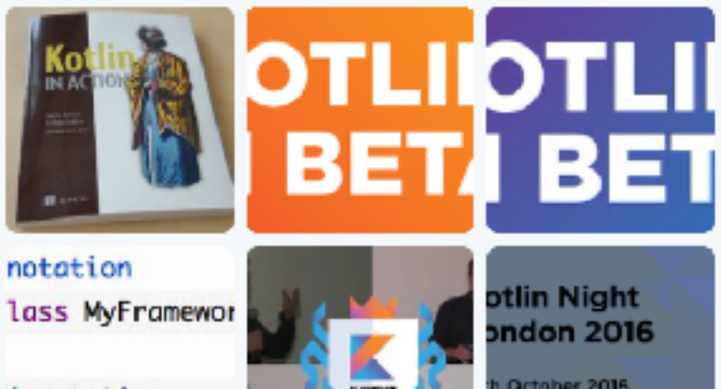
@kotlin

Statically typed Programming Language targeting JVM and JavaScript - Sponsored and Developed by @JetBrains Join us on Slack: kotlinslackin.herokuapp.com

kotlinlang.org

Joined July 2011

90 Photos and videos



Tweets | Tweets & replies | Media

Pinned Tweet

Kotlin @kotlin · 14h

Kotlin 1.1 RC is here, and it's high time to give it a try. We need your feedback!
blog.jetbrains.com/kotlin/2017/02…

← Kotlin 1.1 Beta 2 is here

Kotlin 1.1 Release Candidate is Here →

Our first book about Kotlin is out

Posted on February 10, 2017 by Dmitry Jemerov

We're happy to announce that Kotlin in Action – a book about Kotlin written by the members of the Kotlin team – is now out, as both a eBook and a printed book. The book is written for experienced Java developers and covers all aspects of the language, without focusing on any specific problem domain. We received a lot of positive feedback about the book during Manning's Early Access Preview program, so we hope that you'll enjoy it too!

Kotlin Discussions

Secure https://discuss.kotlinlang.org

Kotlin

all categories | Latest | New (3) | Unread | Top | Categories | + New Topic

Topic	Category	Users	Replies	Views	Activity
Welcome to the Kotlin Discussions powered by Discourse Welcome to the community of Kotlin users, contributors and developers. [image]			1	1.3k	Dec '15
Why Kotlin Native? • new	Language Design		0	16	5h
Nameof to get names of program elements	Language Design		4	415	6h
How is Kotlin intended to be used with Node.js?			3	1.8k	17h
Generated Javascript bindings for Kotlin • new	JavaScript		0	30	1d
Kotlin compiler isn't happy about class and package name clash in a jar • new			1	36	1d
Full Stack Web App Example With Kotlin 1.1	Web Development		5	137	2d
Cross Platform Interop Between Kotlin JS And Kotlin JVM	Support		4	69	2d
Es6 moduleKind support?	JavaScript		7	50	2d
Nulls and generics	Language Design		6	301	3d

general | kotlinlang Slack

Secure https://kotlinlang.slack.com/messages/general/

kotlinlang

Dmitry Kandalov

All Threads

CHANNELS (119)

- # codereview
- # **coroutines**
- # eap
- # general
- # **javascript**
- # kotlinlondon
- # **ktor**
- # random
- # scripting
- # speaking
- # **stdlib**

DIRECT MESSAGES

- slackbot
- Dmitry Kandalov (you)

#general

5,723 | 8 | Welcome, have fun! Slack archive at http://kotlinla...

Search

Today

Yeah, using `TypedArray` always felt like going back ...

@sreich You cannot. You can skip `a.` from `start()` call inside the block tho.

Shaun Reich 16:41
Yeah, thankfully extensions make all of that custom stuff pretty easy and nice

Maciej Górski 16:42
They are indeed awesome until you try something like adding extension function to function type that takes function as param and returns function. //CC @kasper.kondzielski

Victor Tong 16:45
like funktionale....👍

Victor Tong 16:59
added this Plain Text snippet: **I think this is better version of it**

```
1  fun using(obj: Closeable, block: () -> Unit) {
2    try {
3      block()
4    } finally {
5      obj.close()
```

Victor Tong 17:00
@sreich I think this is a standard version of "Finally" under the language construct, Is it what you looking for?

Maciej Górski 17:54
@victortong Kotlin's `use` is actually implemented with what could be said to be correct handling of some edge-case compared to your version. Take a look at the source.

Victor Tong 18:00
@mg6maciej Thanks, it actually in the STD

@mg6maciej As a readWrite kt, Can you give out example when u have the time? Thanks

Krzysztof Kot 18:27
joined #general. Also, @dnipser joined.

Message #general

KOTLIN FUTURE

1.X RELEASES

CO-ROUTINES

KOTLIN NATIVE

THANK YOU!

PLEASE GIVE SOME FEEDBACK

SURVEYMONKEY.CO.UK/R/3HT3GYX

QUICK SORT

```
public static void quicksort(char[] items, int left, int right)
{
   int i, j;
   char x, y;

   i = left; j = right;
   x = items[(left + right) / 2];

   do
   {
      while ((items[i] < x) && (i < right)) i++;
      while ((x < items[j]) && (j > left)) j--;

      if (i <= j)
      {
         y = items[i];
         items[i] = items[j];
         items[j] = y;
         i++; j--;
      }
   } while (i <= j);

   if (left < j) quicksort(items, left, j);
   if (i < right) quicksort(items, i, right);
}
```