Programming languages in general, in particular, and at Bloomberg

Bloomberg Frankfurt
March 19, 2019

Sergey Sitov: C++
Matthias Kleiber: Java / JavaScript
Danica Fine: Scala

TechAtBloomberg.com

Bloomberg
Engineering

© 2019 Bloomberg Finance L.P. All rights reserved.

Programming Languages and C++

Bloomberg

Engineering

Bloomberg Frankfurt
March 19, 2019

Sergey Sitov
Tead Lead, Connectivity Infrastructure
s.sitov@Bloomberg.net

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Programming Languages

- How many programming languages are there?

  Active: ~250 (TIOBE)

  Notable: ~700 (Wikipedia, non-markup)

  HOPL: 8,945 (Online Historical Encyclopaedia of Programming Languages)

© 2019 Bloomberg Finance L.P. All rights reserved.

Programming Languages

- Why are there so many?

“We do not need so many languages, but we want them”

We choose to use different tools for different jobs

We choose what we like and know

Variety is beneficial for the industry

© 2019 Bloomberg Finance L.P. All rights reserved.

Programming Languages

- Why there are so many?

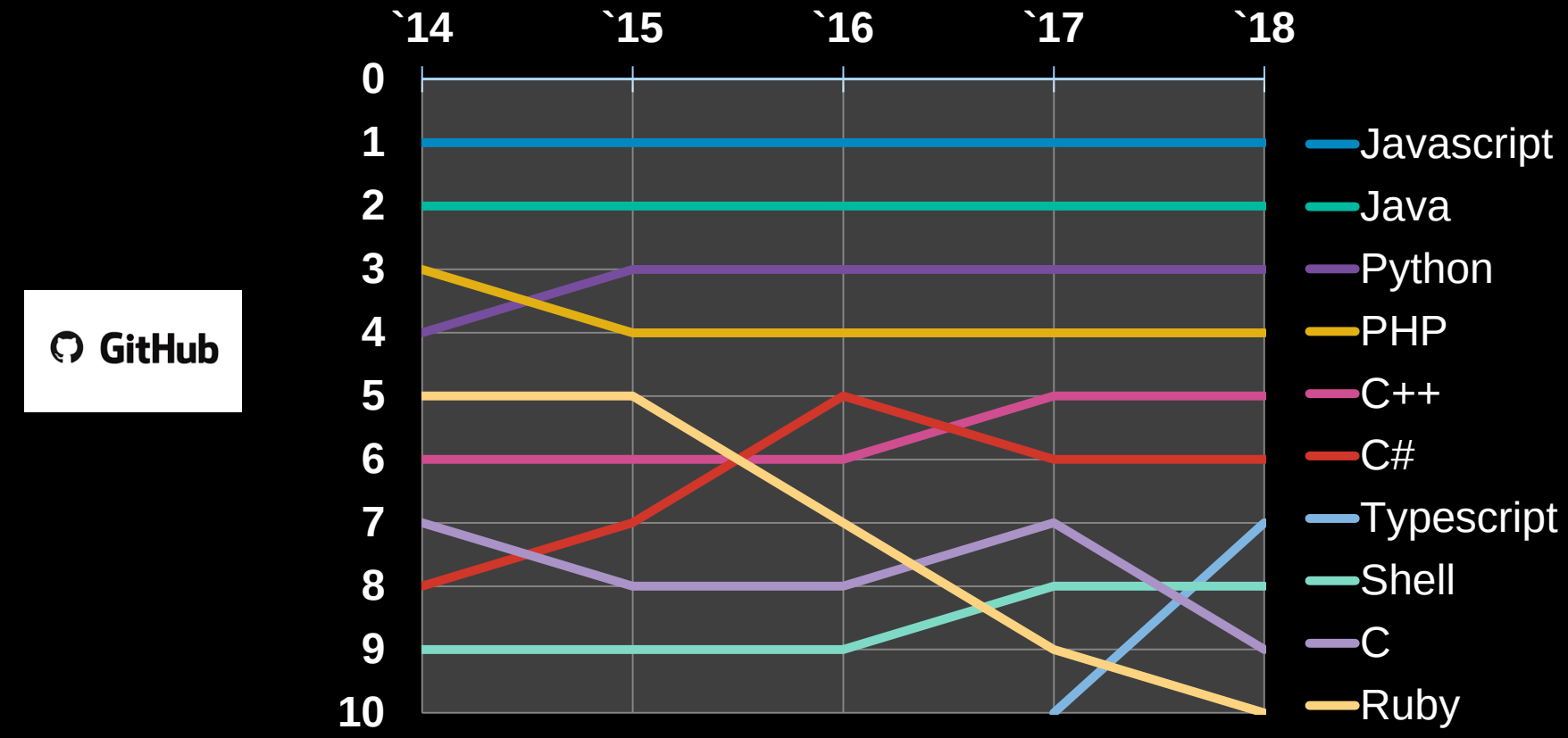
  Because there are:

  - different domains with different requirements
  - different problems to solve
  - different levels of abstraction / paradigms
  - different execution environments / hardware
  - developers with different experience

© 2019 Bloomberg Finance L.P. All rights reserved.

Programming Languages

- What languages are the most popular?



https://github.blog/2018-11-15-state-of-the-octoverse-top-programming-languages/

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Programming Languages

- What languages are the most popular?

Feb 2019	Feb 2018	Change	Programming Language	Ratings	Change
1	1		Java	15.876%	+0.89%
2	2		C	12.424%	+0.57%
3	4	↑	Python	7.574%	+2.41%
4	3	↓	C++	7.444%	+1.72%
5	6	↑	Visual Basic .NET	7.095%	+3.02%
6	8	↑	JavaScript	2.848%	-0.32%
7	5	↓	C#	2.846%	-1.61%
8	7	↓	PHP	2.271%	-1.15%
9	11	↑	SQL	1.900%	-0.46%
10	**20**	⇈	**Objective-C**	**1.447%**	**+0.32%**

https://www.tiobe.com/tiobe-index/

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Programming Languages

- What languages will make you competitive on the market and earn you money?

TechAtBloomberg.com

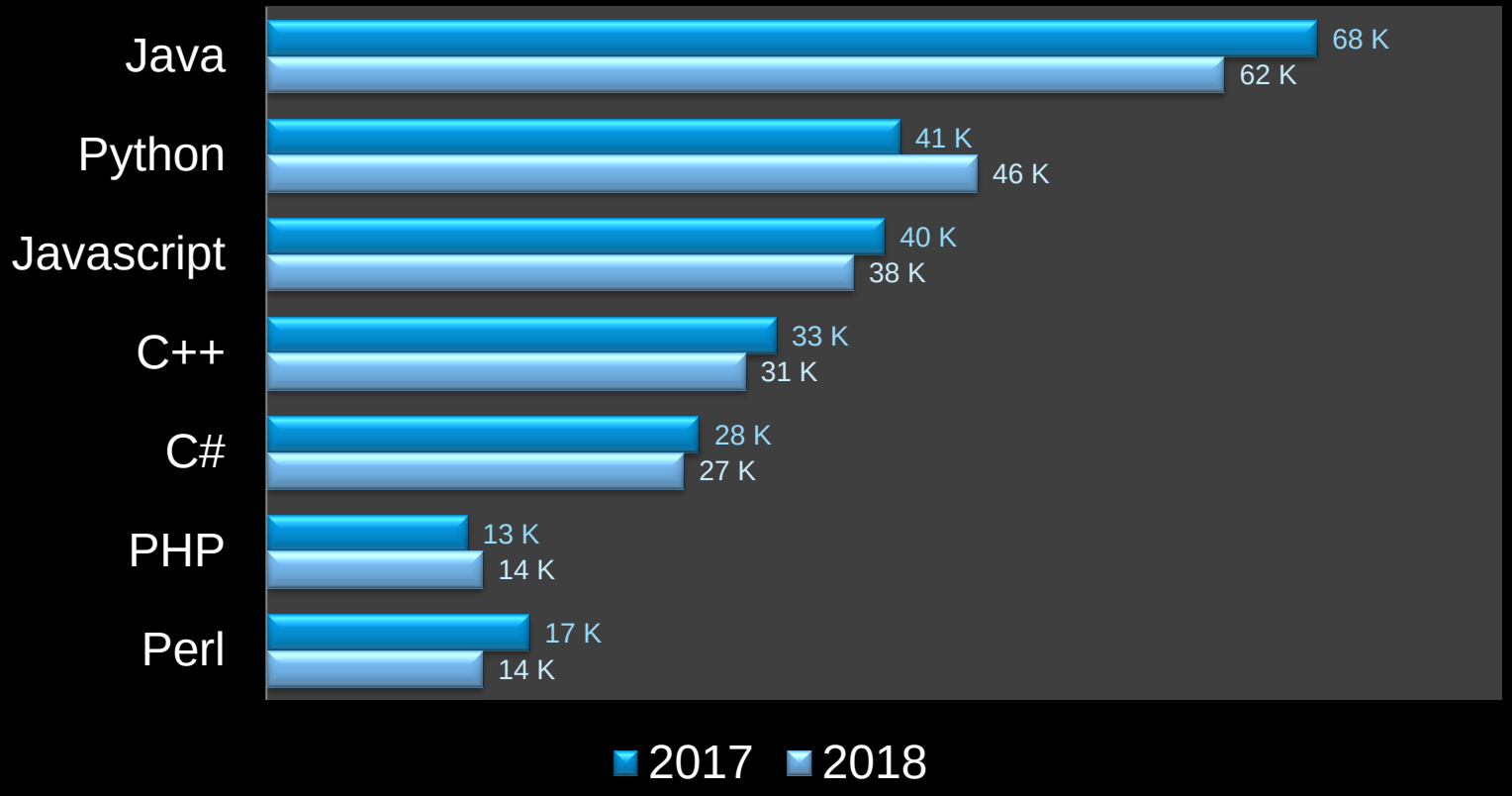
© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Programming Languages

- In-demand programming languages



https://www.codingdojo.com/blog/7-most-in-demand-programming-languages-of-2018/

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Programming Languages

- The highest paying programming languages
  - Go
  - Python
  - R
  - Java
  - Ruby on rails
  - C++
  - C#
  - Swift
  - Kotlin

© 2019 Bloomberg Finance L.P. All rights reserved.

Programming Languages at Bloomberg

- What do we use at Bloomberg?
  - One of the primary language is C++ (by lines of code in version control)
  - JavaScript is for the Terminal front-end
  - Python for automation
- Also: Scala, Java, Lua, Haskell, Perl, Fortran, C#, Rust, Erlang, Golang, Lisp, PHP, …

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

C++ Programming Language

- Created by Bjarne Stroustrup in 1985
- Standardized only in 1998
- Originally intended to include Object-Oriented paradigm into C language (“C with classes”)
- Now it is a statically-typed, imperative, generic, multi-paradigm, multi-purpose programming language

```
#include <iostream>
int main() {
    std::cout << "Hello, World!";
    return 0;
}
```

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

C++ Programming Language

- At Bloomberg, we have a massive C++ code base

- C++ is mostly used for back-end implementations
- For all mission-critical and performance sensitive applications
- It powers the Bloomberg Terminal

- Maintaining open source **BDE** (Basic Development Environment) (https://github.com/bloomberg/bde)
- Representation on the WG21 (ISO C++ Committee)

- Actively participate at large C++ conferences

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

C++ Programming Language

- Why do engineers want and like to use C++?
  - C++ offers ways of abstraction that have – unlike **all** other languages – no performance overhead at runtime
  - It allows one to write very efficient code that still has a very high abstraction level – compile time vs. runtime trade-offs
  - Direct mapping to hardware features – feeling of coding to `bare-metal` rather than to an interpreter or virtual machine

© 2019 Bloomberg Finance L.P. All rights reserved.

C++ Programming Language

- Generic Programming with C++

- Freedom from data types by templates

```
template <typename T>
const T& max(const T &a, const T &b) {
    return a > b ? a : b;
}
```

- Freedom to manipulate data using comparators/predicates
- Abstraction of access to data structures using iterators or ranges

© 2019 Bloomberg Finance L.P. All rights reserved.

C++ Programming Language

- Generic Programming with C++
- You can write libraries of reusable code with templates, which can be specialized by a user's own types and policies
- Standard Templates Library (STL)
-

- Bloomberg BDE

© 2019 Bloomberg Finance L.P. All rights reserved.

C++ Programming Language

- Generic Programming with C++
- Templates in C++ are Turing complete (meta-programming)
- It is possible to write code executed at compile time (constant expressions)

© 2019 Bloomberg Finance L.P. All rights reserved.

C++ Programming Language

- It is still primarily an Object-Oriented Programming language, but it also now has idioms from functional programming: auto, lambdas, ranges, combinators, filters, …

Generate an infinite list of integers starting at 1, square them, take the first 10, and sum them:

```
using namespace ranges;
auto sum = accumulate(view::ints(1)
                    | view::transform([](int i){return i*i;})
                    | view::take(10), 0);
```


© 2019 Bloomberg Finance L.P. All rights reserved.

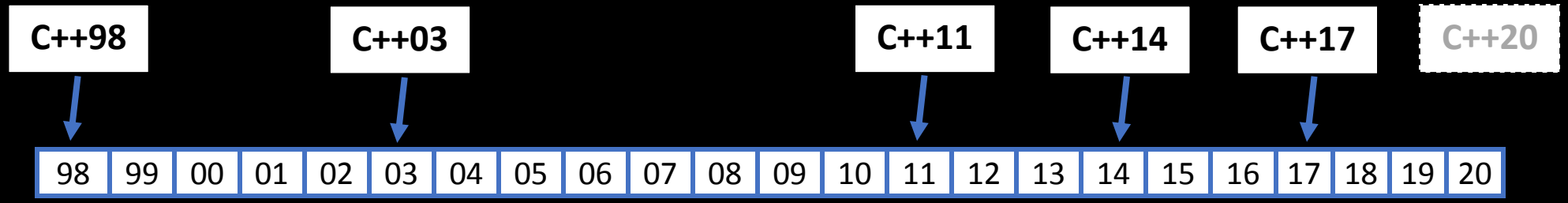
C++ Programming Language

- Low-latency trading with C++
  - Cache-aware programming
  - Lock-free algorithms
  - Bypass calls to kernel
  - Directly use special CPU instruction set (SSE2, SSE3, AVX ...)
  - Use GPU hardware (OpenCL, CUDA)
  - High-performance computing with OpenMP

© 2019 Bloomberg Finance L.P. All rights reserved.

C++ Programming Language

- C++ language is still evolving



- Try the language and all its new features
  - godbolt.org
  - wandbox.org

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Thank you!

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Programmiersprachen im Allgemeinen, im Speziellen und bei Bloomberg: Java/JavaScript

Bloomberg

Engineering

Bloomberg Frankfurt
March 19, 2019

Matthias Kleiber
Software Architect & Developer
m.kleiber@bloomberg.net

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Programmiersprachen: Java

- Warum ist Java NICHT die beste Programmiersprache?
  - Ausschließlich dynamische Funktionsaufrufe (performance)
  - Java-Runtime JVM wird zur Laufzeit benötigt
  - Keine direkten Funktions-Aufrufe zum Betriebssystem
  - Keine Unterstützung für Speicherstrukturen des Betriebssystems
  - Geräte-Treiber sind de facto unmöglich

© 2019 Bloomberg Finance L.P. All rights reserved.

Programmiersprachen: Java

- Muss ich Java sprechen können?
  - Nein, aber lesen!
  - Nein, aber die JVM verstehen
  - Nein, aber Interfaces verstehen
  - Nein, aber warum eigentlich nicht?

© 2019 Bloomberg Finance L.P. All rights reserved.

Programmiersprachen: Java

- OO und EE
  - Klassen sind die Basis...
  - ...aber nicht für alles!
  - Java Enterprise Edition
  - JSF, JAX-RS, JPA, EJB

© 2019 Bloomberg Finance L.P. All rights reserved.

Programmiersprachen: Java

- Wann geht Java in Rente?
  - Wann ging Assembler in Rente?
  - Sind IoT, Social Media und mobile Apps das Ende von Java?
  - Kotlin – mindestens die JVM überlebt

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Programmiersprachen: JavaScript

- Ist JavaScript eine Programmiersprache?
  - Programmierst Du noch oder skriptest Du schon?
  - Warum ein Interpreter weder Compiler, noch VM ist
  - (zu) viel Dynami(k/t)?

© 2019 Bloomberg Finance L.P. All rights reserved.

Programmiersprachen: JavaScript

- Warum spricht das Internet nur JavaScript?
  - Als das Internet Laufen lernte und das erste DOM entstand
  - Die ScriptEngine - der Motor des Internets
  - Transpilieren

© 2019 Bloomberg Finance L.P. All rights reserved.

Programmiersprachen: JavaScript

- Das JavaScript-Engine Oligopol
  - Warum sieht die gleiche Internetseite in jedem Browser anders aus?
  - jQuery – jQuery – jQuery 97.3%
  - Kleine Scripte – große Wirkung

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Programmiersprachen: JavaScript

- Frameworks und kein Ende
  - AngularJS
  - React
  - Vue.js
  - Node.js

© 2019 Bloomberg Finance L.P. All rights reserved.

Danke!

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Scala at Bloomberg and Beyond

Bloomberg Engineering

Bloomberg Frankfurt
March 19, 2019

Danica Fine
Software Engineer
dfine17@bloomberg.net

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Agenda

- What is Scala?
- Why Scala?
  - Functional Features
  - Embracing Functional Programming
- Scala at Bloomberg
- Takeaways
- Questions

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

What is Scala?

“Scala combines object-oriented and functional programming in one concise, high-level language. Scala's static types help avoid bugs in complex applications, and its JVM and JavaScript runtimes let you build high-performance systems with easy access to huge ecosystems of libraries.”

www.scala-lang.org

TechAtBloomberg.com

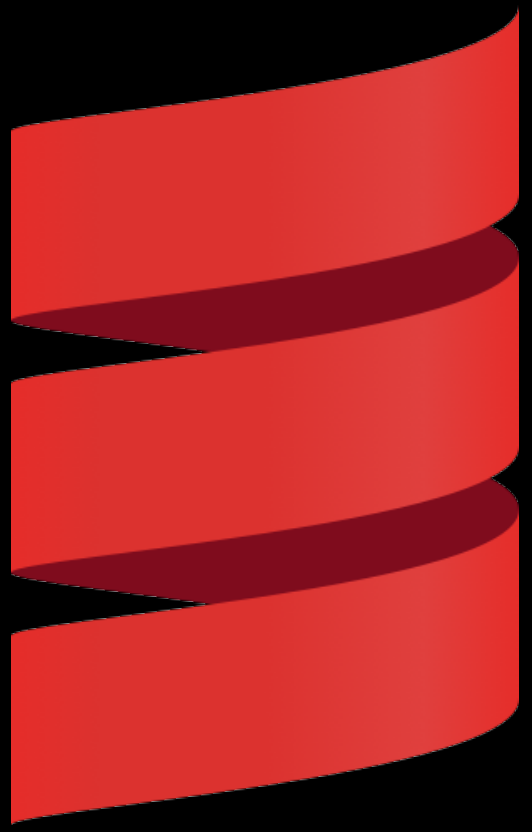
© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Scala Overview

- “Scalable Language”
- Blend of functional and object-oriented paradigms
- More specifically…
  - Type-safe
  - Built on the JVM
  - Wide support of Java collections and other popular libraries



TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Why Functional?

- Higher predictability
- Compile-time optimizations
- Avoid concurrency problems
- Reduce verbosity (in most cases)

Type fewer characters!

“You only have a finite number of keystrokes in your lifetime. Don’t waste them on unnecessarily verbose code.”

- Afzal Mazhar

© 2019 Bloomberg Finance L.P. All rights reserved.

Other Features

To convince you to love Scala…

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Type Inference

- Compiler infers types of expressions

```
>> val mult = (x: Int, y:Int) => x * y
   mult: (Int, Int) => Int
>> val bbg = "Bloomberg"
   bbg: String = Bloomberg
```

- Safer Code

```
def userExistsIn(userIds: Set[Long]): Boolean = {
  var userId = getCurrentUserId() // returns Long
  userIds.contains(userId)
}
```

- Convenient, less verbose

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Functions as First-Class Citizens

- Values can be a function type

```
>> val f1 = (s: String) => s.length
   f1: String => Int = $$Lambda$162/1654@4a7fd0c9
```

- Passed as arguments

```
>> val l1 = Seq("a", "ab", "abc")
   l1: Seq[String] = List(a, ab, abc)
>> l1.map(f1)
   res0: Seq[Int] = List(1, 2, 3)
```


© 2019 Bloomberg Finance L.P. All rights reserved.

Functions as First-Class Citizens (cont.)

- Returned as results

```
>> def add(x: Int) = (y: Int) => x + y
   add: (x: Int)Int => Int
>> val addTwo = add(2)
   addTwo: Int => Int = $$Lambda$107/7037@7d401cdc
>> addTwo(3)
   res0: Int = 5
```

- Partial application/currying

```
>> val l = Seq(1, 2, 3)
>> val init0 = l.foldLeft(0)(_)
   init0: ((Int, Int) => Int) => Int = $$Lambda$17/89@77f3c1
```


© 2019 Bloomberg Finance L.P. All rights reserved.

Option Type

- Handling objects that may or may not exist
- Compiler forces programmer to deal with this possibility

Offers a solution to the “Billion-dollar mistake!”

© 2019 Bloomberg Finance L.P. All rights reserved.

Option Type (cont.)

```
def toInt(s: String): Option[Int] = {
  try {
    Some(Integer.parseInt(s.trim))
  } catch {
    case e: NumberFormatException => None
  }
}
```

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Case Classes

- Good for modelling immutable data
- Similar to regular classes, but with some extras
  - Built-in “apply” method for instantiation
  - Shallow copying ability
  - Structure comparison
- Boilerplate code done for you

```
>> case class User(uuid: Int, name: String, . . .)
   defined class User
```

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Case Classes (cont.)

```
>> case class User(uuid: Int, name: String)
   defined class User
>> val bob = User(0, "Bob")
   bob: User = User(0, "Bob")
>> bob.name
   res1: String = Bob
>> bob.name = "New Bob”
   <console>:12: error: reassignment to val
>> val newBob = bob.copy(name = "New Bob")
   newBob: User = User(0, "New Bob")
>> bob == newBob
   res3: Boolean = false
```

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Pattern Matching

- Used to deconstruct values that match against a pattern
- Patterns include:
  - Values

```
(l1.size, l2.size) match {
  case (0,0) => “Both are empty.”
  case (0,_) => “First is empty.”
  case (_,0) => “Second is empty.”
  case (a,b) => “Total ” + (a + b)
}
```

  - Types

```
v match {
  case (s: String) => . . .
  case (i: Int)    => . . .
  case _           => . . .
}
```

  - Case Classes

```
v match {
  case User(i, _) if (i < 10) => . . .
  case User(i, n) => . . .
}

v match {
  case Some(s) => . . .
  case None    => . . .
}
```

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Tips to Embracing the Functional Mindset

- Avoid mutating variables; opt for immutable values instead
- Take advantage of collections and higher-order functions (e.g., map, filter, and fold)
- Pass all required data as arguments; avoid relying state

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Scala at Bloomberg

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Derivatives Data Use Case

- Low-latency application pipeline
- Streaming infrastructure
- Scala is a player in big data processing (Spark)

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Other Projects within the Company

- Apache Spark
- Support Scala code generation for our in-house application layer

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Things to Consider

According to Stack Overflow's 2018 Developer Survey…

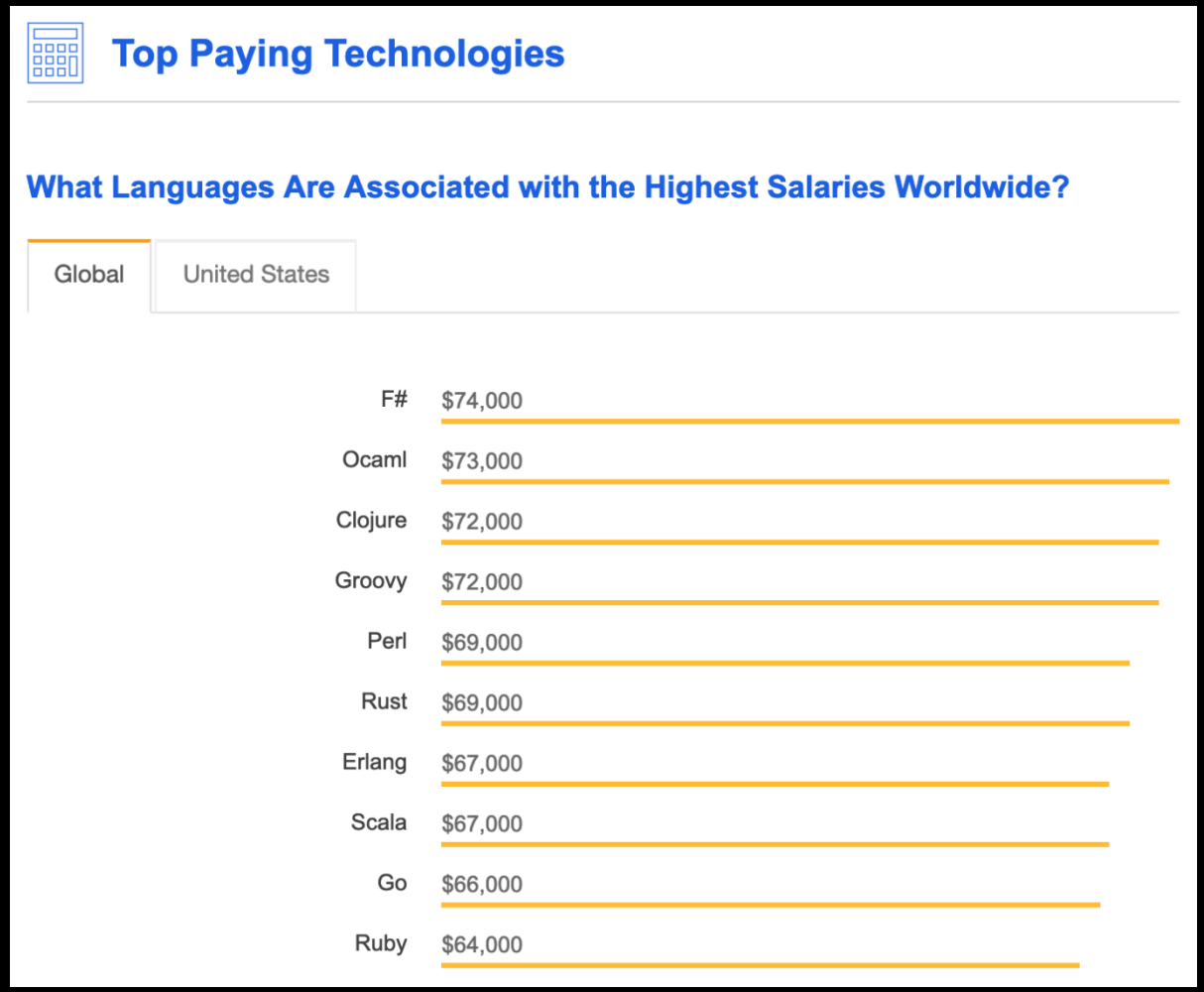
https://insights.stackoverflow.com/survey/2018

Bloomberg

Engineering

Salary

- Scala is associated with high-earning jobs
- ~€60,000
- Half of the top 10 highest earning programming jobs use functional languages



TechAtBloomberg.com

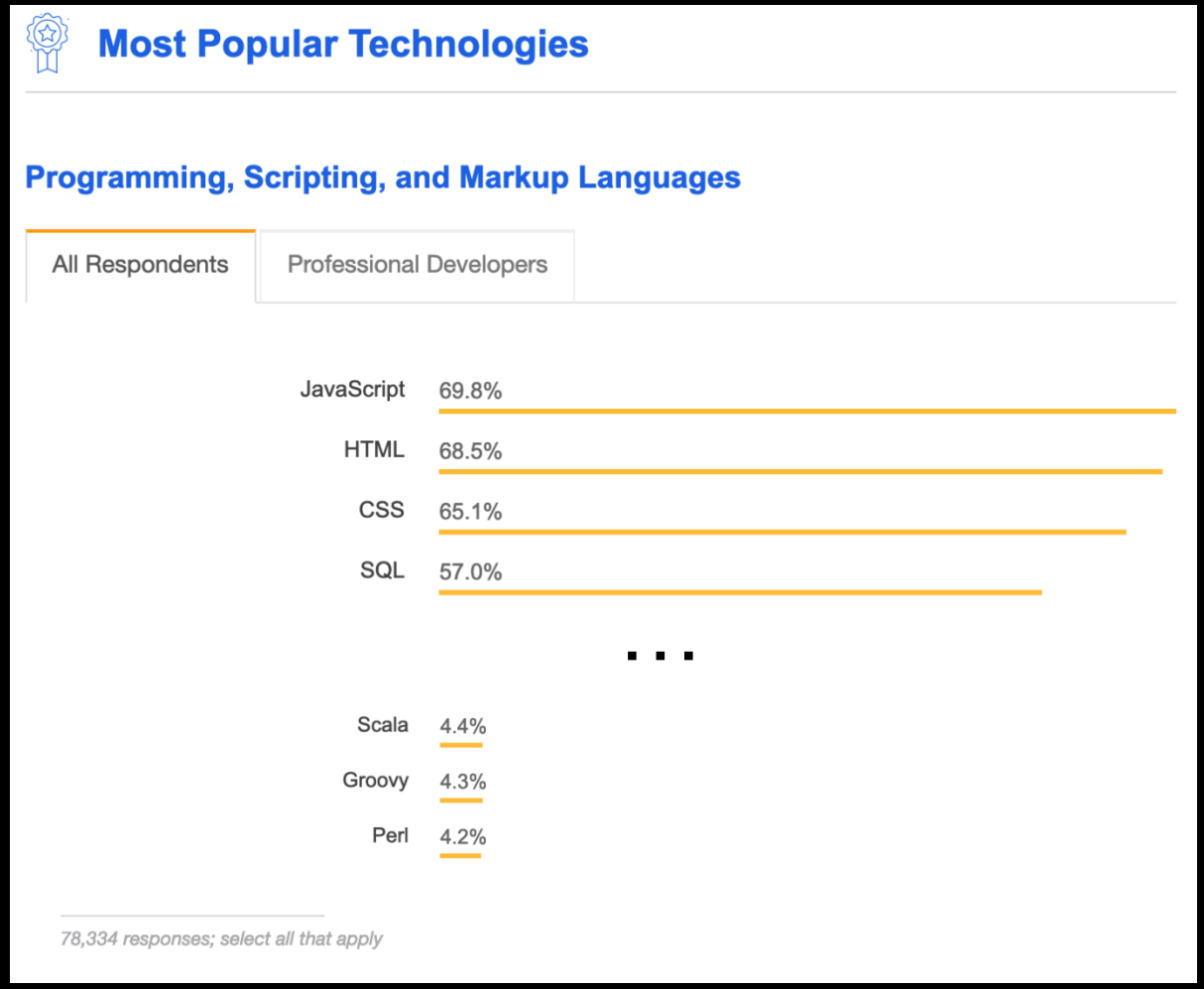
© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

Popularity

- Not at the bottom of the list…



TechAtBloomberg.com

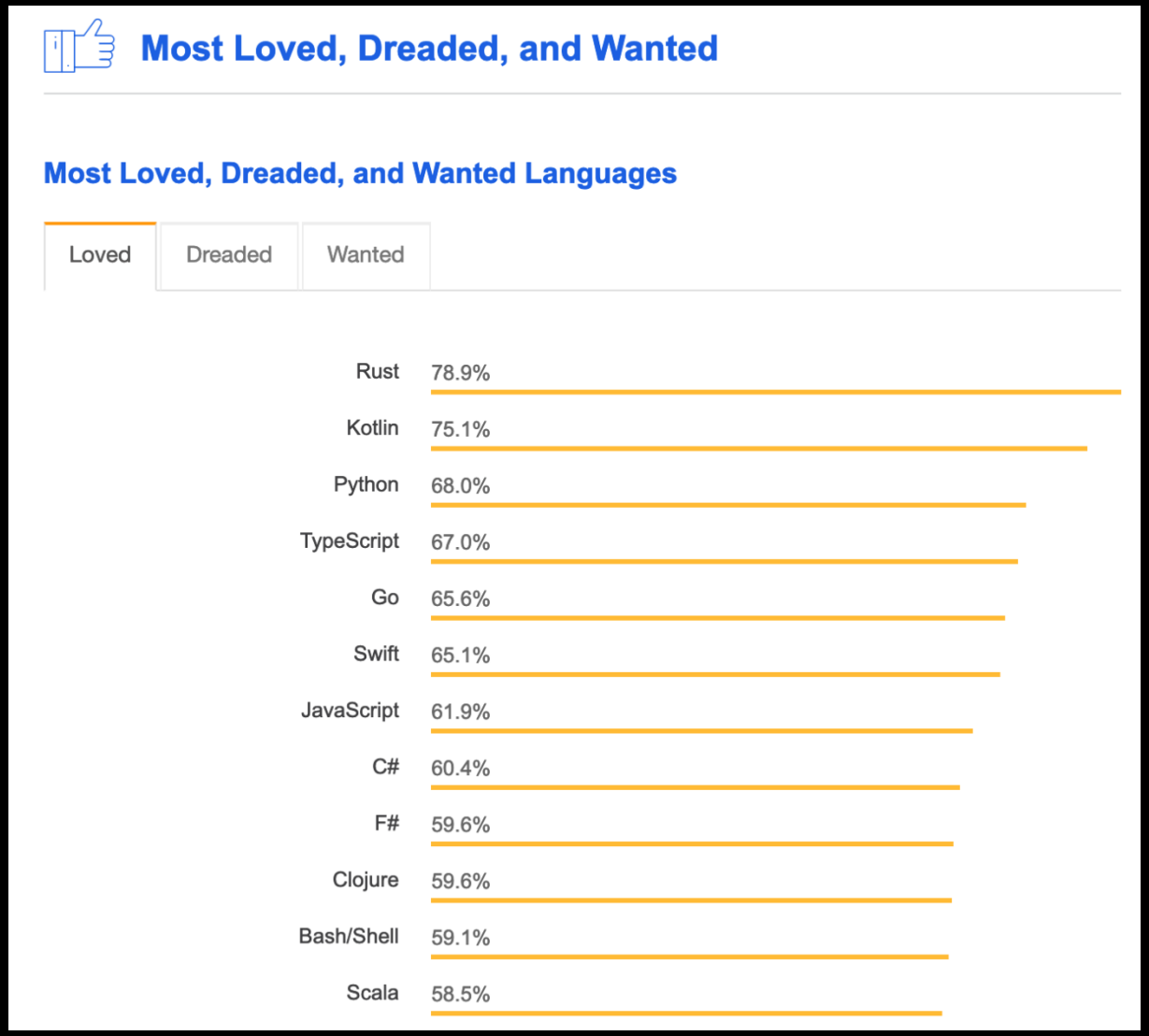
© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

A Loved Language

- 12th most loved!
- Did not make the “Most Dreaded” list



TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.

Bloomberg

Engineering

We are hiring!

https://www.bloomberg.com/careers

Bloomberg

Engineering

Questions?

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.