



# Java应用技术 课程说明

浙江大学计算机学院

# 课程说明

- 课程名称：Java应用技术
- 主讲老师：鲁伟明
- 联系方式
  - 办公室：玉泉校区数字图书馆教育部工程研究中心（图书馆220）
  - 邮箱：luwm@zju.edu.cn
  - 电话：13575464647
- 上课时间：每周四下午9,10节@曹光彪二期-104;
- 实验课： 周一第9,10节/单周@曹光彪西-503



# 教学目的

- Focus on Java application programming
  - desktop or server
- Able to implement a small software system in Java

通过本课程的学习，掌握Java语言的基础知识及语法，理解和掌握面向对象的设计思想、设计原则、设计模式，初步掌握Java面向对象程序设计的基本方法与技术，提高抽象思维能力，具备应用Java技术求解实际问题的能力。



## □ (1) 知识方面:

- 了解Java语言的特点及应用领域;
- 理解面向对象程序设计的基本概念、基本思想, 掌握面向对象程序设计的基本方法、设计原则、重要模式等;
- 熟练掌握Java语言的基本语法, 包括: 数据类型、表达式、程序控制结构;
- 掌握Java面向对象程序设计的基本编程技术, 包括: 类的定义、对象、继承与多态、抽象类与抽象方法、异常处理等;
- 掌握正确、良好的编程规范及文档注释规范;
- 理解输入/输出流类的基本概念, 掌握输入输出与文件访问的编程方法;
- 理解图形用户界面(GUI)功能的实现机制(事件处理), 掌握简单GUI的编程方法;
- 掌握数据库编程、并发编程、网络编程等能力;
- 了解JVM内存管理。



## □ （2）技能方面：

- 能够在JDK开发环境中，完成Java程序的上机过程；
- 初步掌握一种Java语言的集成开发平台的使用（使用Eclipse）；
- 能够充分利用Java工具及环境所提供的功能，编写规范的程序代码，初步掌握基本的程序编写和调试方法。
- 学会查找Java类库，能够阅读类、构造函数、属性及方法的使用说明。



### □ （3）能力方面：

- 具备基本的Java程序设计能力。
- 具备应用面向对象的思想和方法进行问题求解的能力。
- 具备利用Java语言提供的功能，编写应用型程序、GUI程序、数据库程序、网络程序的能力；

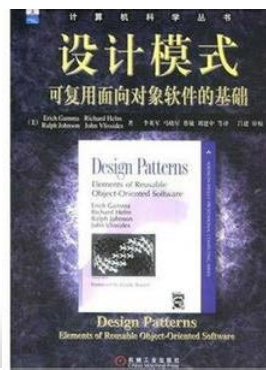
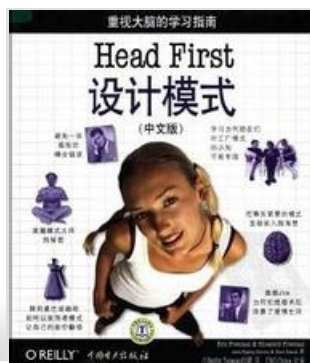
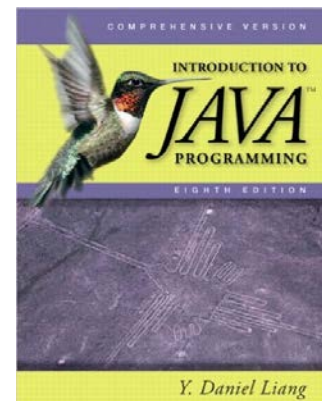
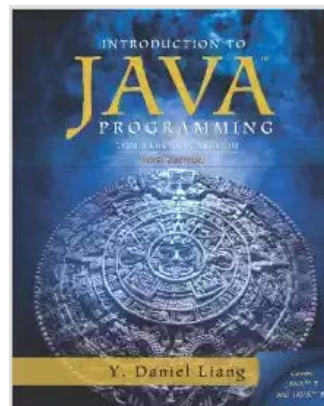
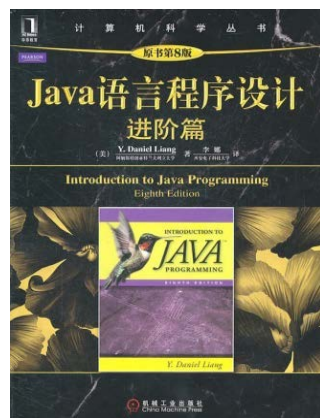
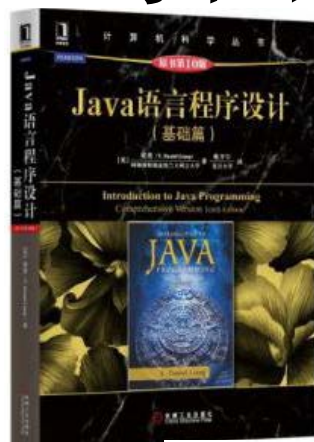


# 教学内容

- Java基础知识
- Java内存模型
- Java的类和对象
- 类库
- 容器
- 泛型
- 异常与RTTI
- 输入输出
- 数据库编程
- 网络编程
- 并发编程
- GUI
- MVC模型
- 设计模式
- 国际化
- .....



# 参考图书





# 联系方式

- QQ群：673126387
- 课间加群



# 课程考核

- 平时成绩（60%）+期末考试（40%）
- 平时成绩
  - 出勤情况
  - 平时作业（小程序或代码分析等）
  - 课堂小测验（PTA）
  - 期中考试(?)
  - .....

PAT测试: <https://www.patest.cn/>

PTA: <https://pintia.cn/>



**PTA**

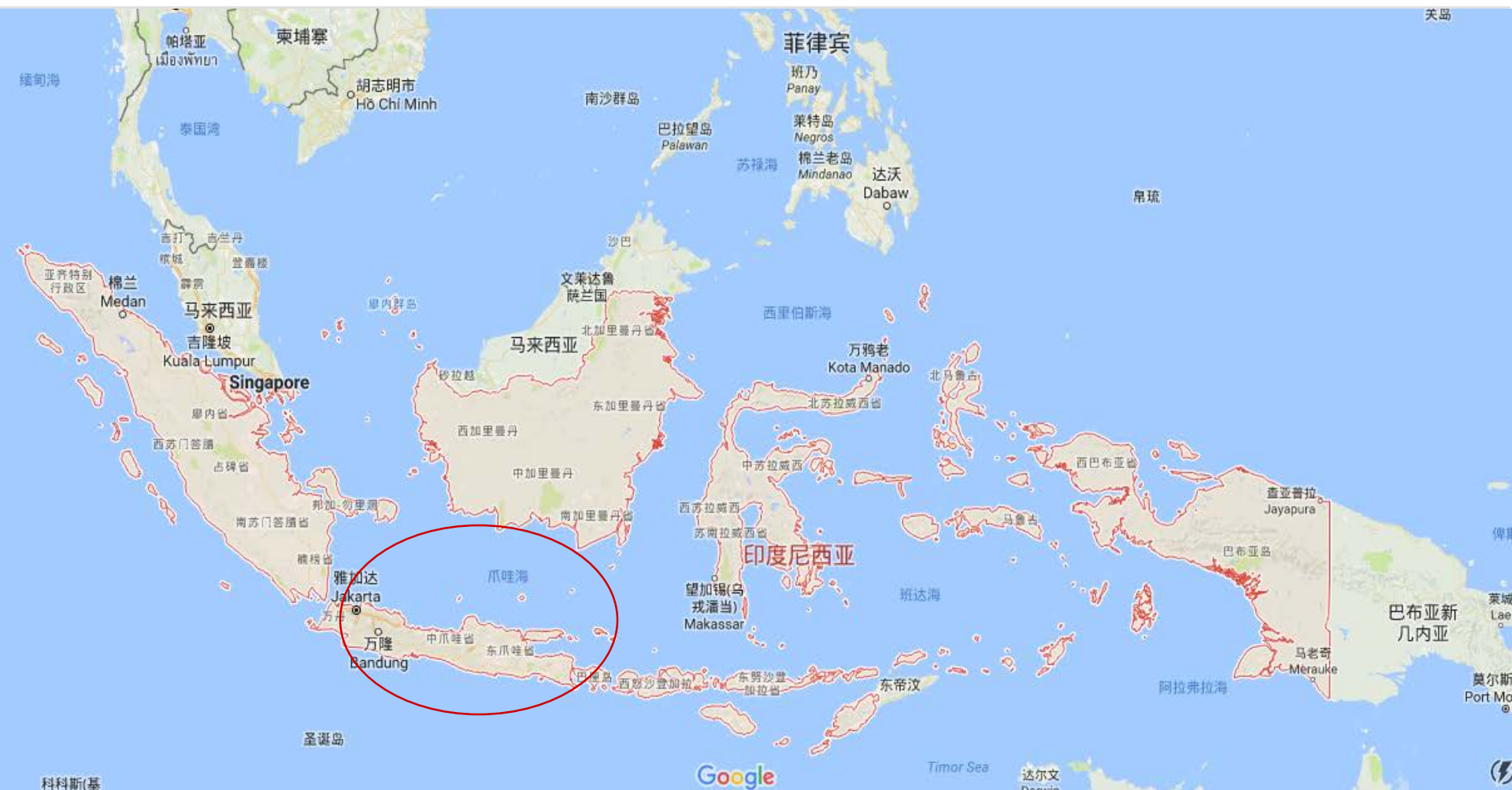
程序设计类实验辅助教学平台  
Programming Teaching Assistant

# 课程FTP

- `ftp://java:java2019@10.15.82.27`
  - Slides – 课件及作业
  - Software – 相关软件
  - Homework – 学生上传作业

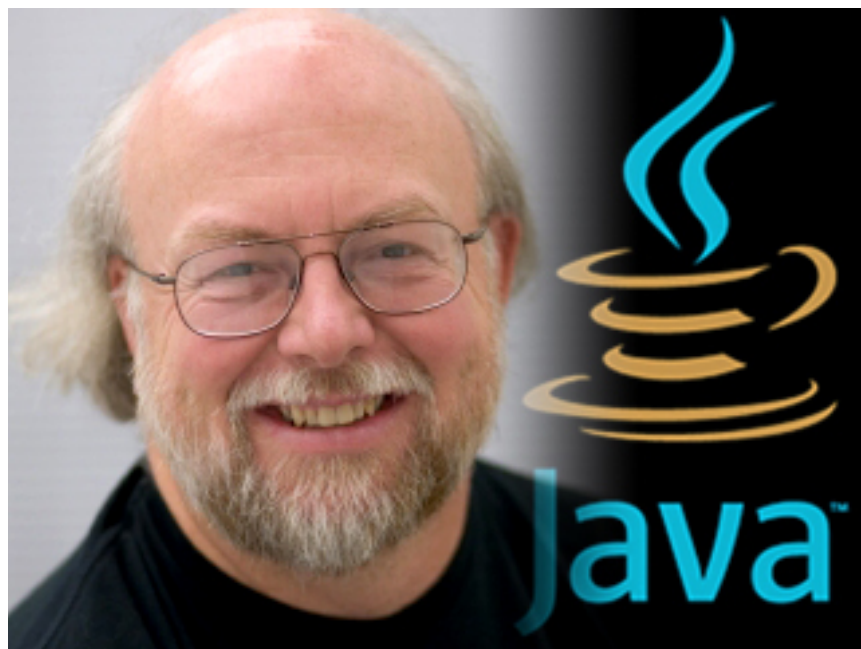


# Java ?



# Java的创始人

## “Java之父”



1955年出生于加拿大，CMU的计算机博士。任职于 Sun Microsystems 期间开发了 Java 编程语言，在Oracle收购 Sun 后辞职，短暂加盟了搜索巨人 Google，随后担任了海洋机器人公司 Liquid Robotics 的首席软件架构师，开发自主驾驶的无人船 Wave Glider。

### James Gosling, Sun副总裁

1990年，与Patrick Naughton和Mike Sheridan等人合作“绿色计划(Green)”，后来发展一套语言叫做“Oak”，后改名为Java。1994年底，James Gosling在硅谷召开的“技术、教育和设计大会”上展示Java程序。2000年，Java成为世界上最流行的编程语言。



- 2013年，Gosling成为美国计算机协会研究员
- 2015年，Gosling获得IEEE颁发的约翰.冯.诺依曼奖章
- 2017年，Gosling 宣布加盟亚马逊 AWS 服务，成为云计算巨头的杰出工程师。



# Java VS. C++

- 1. 指针
- 2. 多重继承
- 3. 数据类型及类（Java是完全面向对象的语言，所有函数和变量都必须是类的一部分）
- 4. 内存管理
- 5. 操作符重载
- 6. 变量与函数（全局）
- 7. goto 语句(*Java*语言中`goto`是保留关键字,但没有`goto`语句)
- 8. 类型转换（自动强制类型转换，如`int x = 1.2`    *java* X）
- 9. 异常处理（*java*异常处理更加严谨、简洁）



# 编程语言排行榜 (TIOBE)

Aug 2019	Aug 2018	Change	Programming Language	Ratings	Change
1	1		Java	16.028%	-0.85%
2	2		C	15.154%	+0.19%
3	4	▲	Python	10.020%	+3.03%
4	3	▼	C++	6.057%	-1.41%
5	6	▲	C#	3.842%	+0.30%
6	5	▼	Visual Basic .NET	3.695%	-1.07%
7	8	▲	JavaScript	2.258%	-0.15%
8	7	▼	PHP	2.075%	-0.85%
9	14	▲▲	Objective-C	1.690%	+0.33%
10	9	▼	SQL	1.625%	-0.69%
11	15	▲▲	Ruby	1.316%	+0.13%
12	13	▲	MATLAB	1.274%	-0.09%
13	44	▲▲	Groovy	1.225%	+1.04%
14	12	▼	Delphi/Object Pascal	1.194%	-0.18%
15	10	▼▼	Assembly language	1.114%	-0.30%
16	19	▲	Visual Basic	1.025%	+0.10%
17	17		Go	0.973%	-0.02%
18	11	▼▼	Swift	0.890%	-0.49%
19	16	▼	Perl	0.860%	-0.31%
20	18	▼	R	0.822%	-0.14%



TIOBE编程社区排名则**每月更新一次**。其参考标准基于主流搜索引擎之上全球各地相关软件工程师、课程及第三方厂商的具体数量。

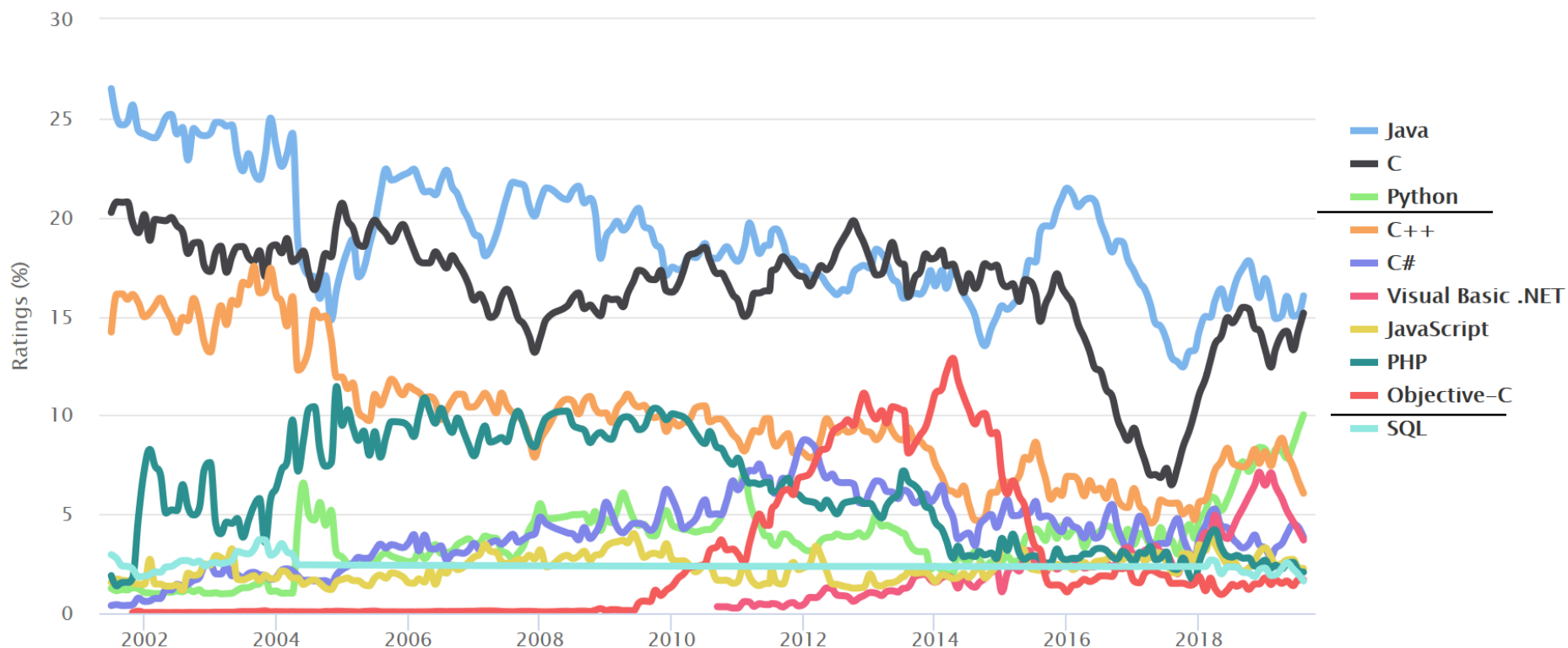


# 编程语言排行榜

## 前 10 名编程语言长期走势图

TIOBE Programming Community Index

Source: [www.tiobe.com](http://www.tiobe.com)



2017年C编程语言的得分为7.382%，为2001年以来新低。这一下降的一个主要原因是C是很难适用于Web开发和移动应用开发蓬勃发展的领域。

21	D	0.807%
22	SAS	0.798%
23	PL/SQL	0.745%
24	Dart	0.715%
25	ABAP	0.498%
26	F#	0.476%
27	Logo	0.465%
28	Rust	0.450%
29	Scratch	0.448%
30	Lua	0.414%
31	Transact-SQL	0.399%
32	COBOL	0.369%
33	Fortran	0.364%
34	Lisp	0.362%
35	TypeScript	0.333%
36	Scala	0.311%
37	Ada	0.296%
38	ActionScript	0.288%
39	Julia	0.279%
40	Scheme	0.278%
41	RPG	0.272%
42	Prolog	0.267%
43	PostScript	0.254%
44	VBScript	0.243%
45	Kotlin	0.225%
46	Awk	0.204%
47	Apex	0.189%
48	Bash	0.187%
49	Haskell	0.174%
50	PowerShell	0.166%



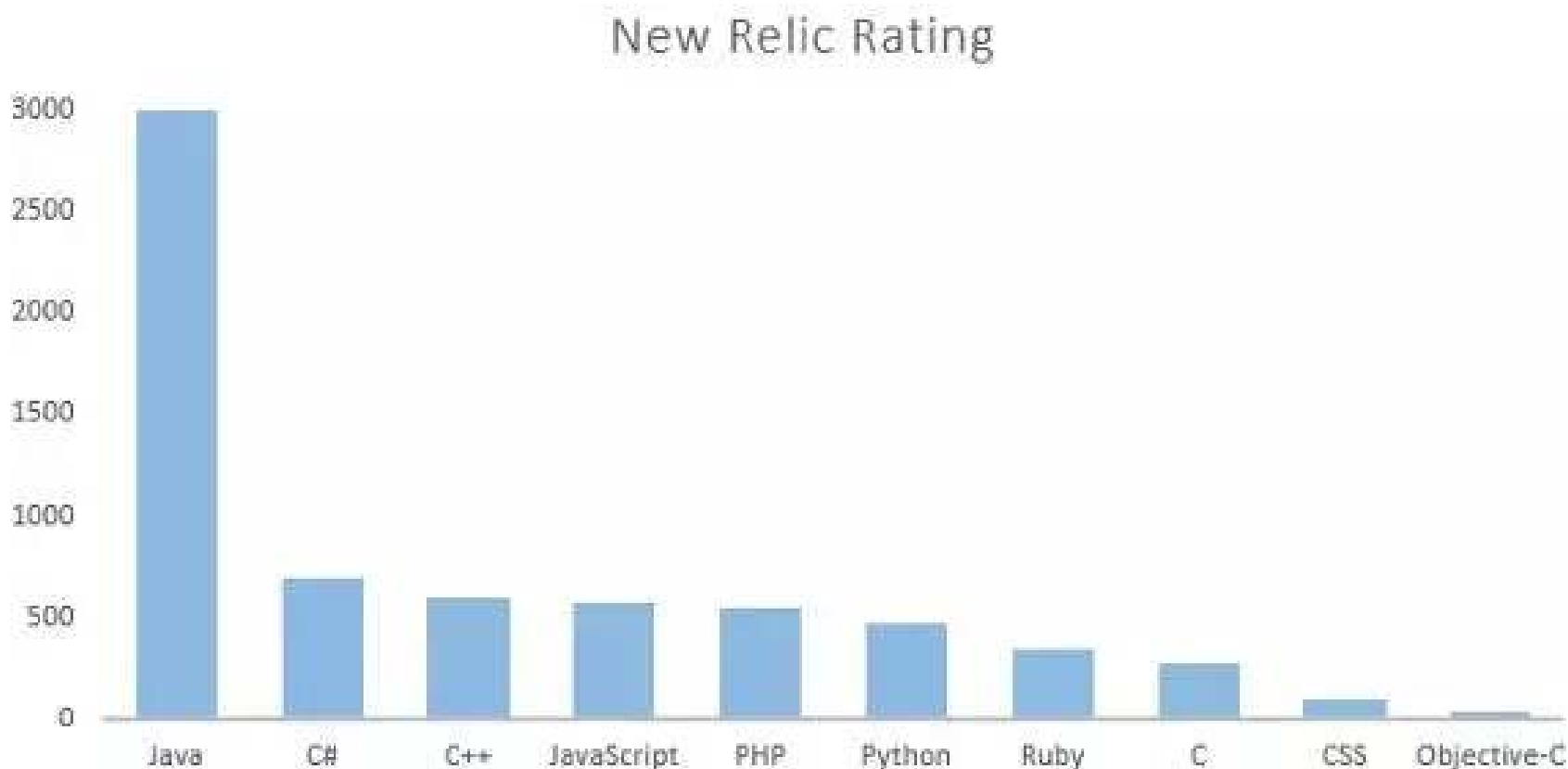
# Very Long Term History

To see the bigger picture, please find below the positions of the top 10 programming languages of many years back. Please note that these are *average* positions for a period of 12 months.

Programming Language	2019	2014	2009	2004	1999	1994	1989
Java	1	2	1	1	14	-	-
C	2	1	2	2	1	1	1
Python	3	7	5	7	24	21	-
C++	4	4	3	3	2	2	2
Visual Basic .NET	5	9	-	-	-	-	-
C#	6	5	6	6	19	-	-
JavaScript	7	8	8	8	16	-	-
PHP	8	6	4	5	-	-	-
SQL	9	-	-	89	-	-	-
Objective-C	10	3	31	38	-	-	-
Perl	16	11	7	4	3	10	22
Lisp	32	13	19	13	12	5	3
Pascal	220	16	14	88	6	3	20

# 编程语言排行榜

New Relic对Indeed网站上的招聘信息进行汇总，发现总计1600万个岗位当中Java的对应需求最高。其它高人气语言还包括JavaScript、C#与C++。

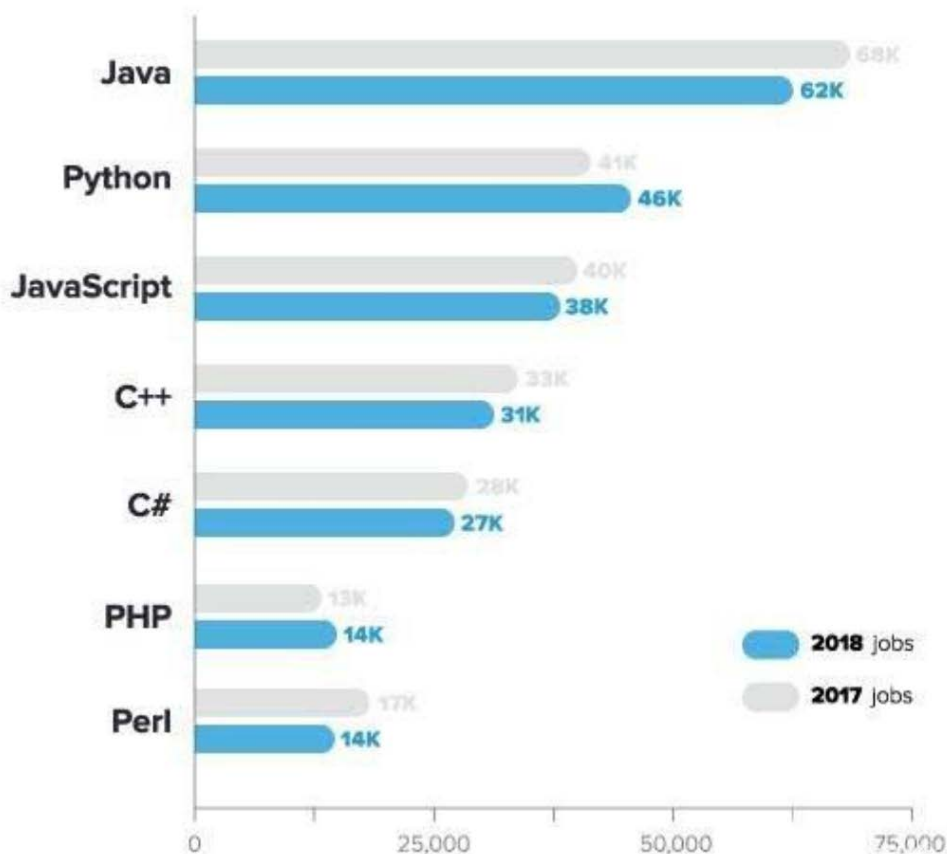


# 编程语言排行榜

- 2019年即将到来，Coding Dojo（编码道场）发布了 2018 最具就业前景的 7 大编程语言。该公司分析了来自 Indeed 的25门编程语言、栈和框架的数据，以找出雇主最需求的七个数据。数据基于每种语言的工作发布数量。

Job postings containing top languages

Indeed.com - November, 17th 2017



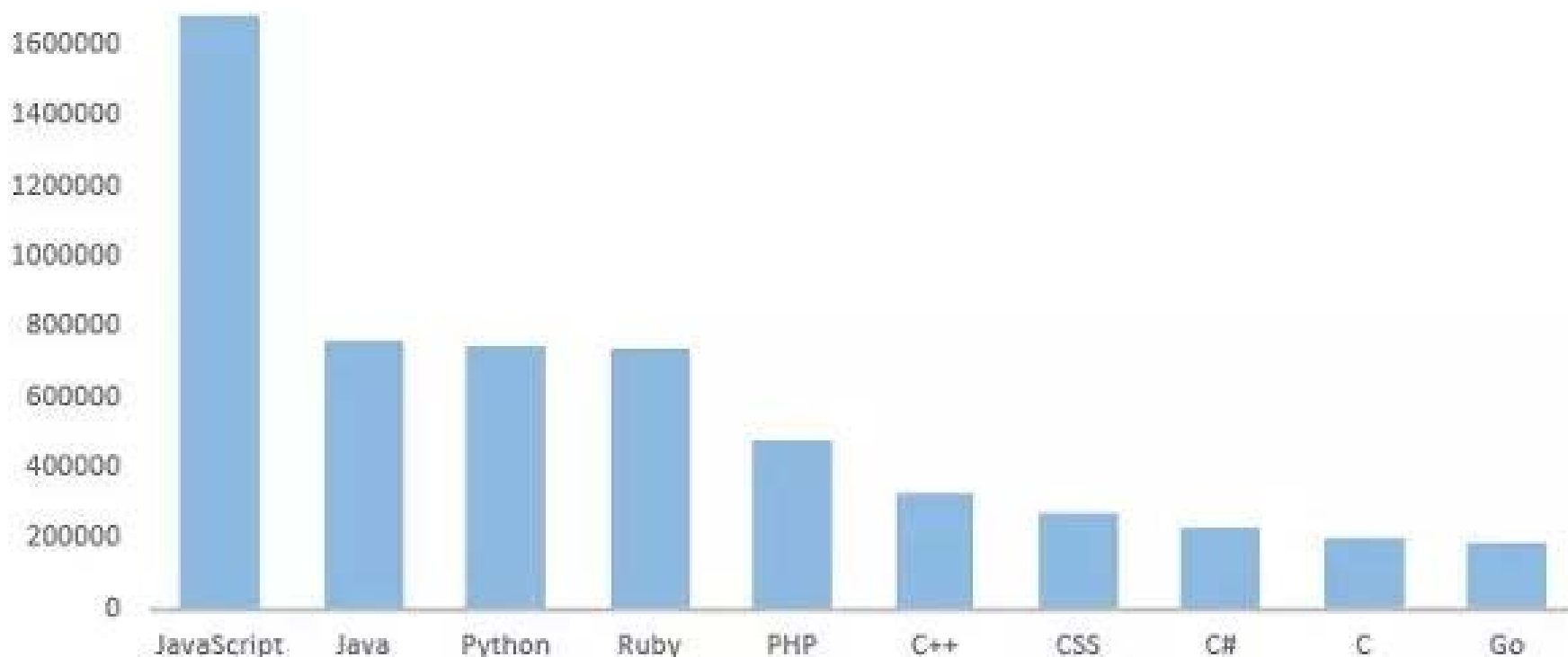
位居前三名的分别为 Java、Python 和 JavaScript，但相比2017年的数据，只有 Python 是岗位有所增长的。而且 Coding Dojo 认为，随着对机器学习开发人员需求的增长，未来对 Python 的需求还将继续增长。



# 编程语言排行榜

- 2016年秋季，Tech Worm的工作人员通过GitHub整理出其中最具人气的15项编程语言。其中JavaScript位列第一，C#为第八，Go则为第十（较上年增长近一倍）。虽然Swift与TypeScript排名靠后，但二者皆实现了高达3.5倍的惊人需求增长。

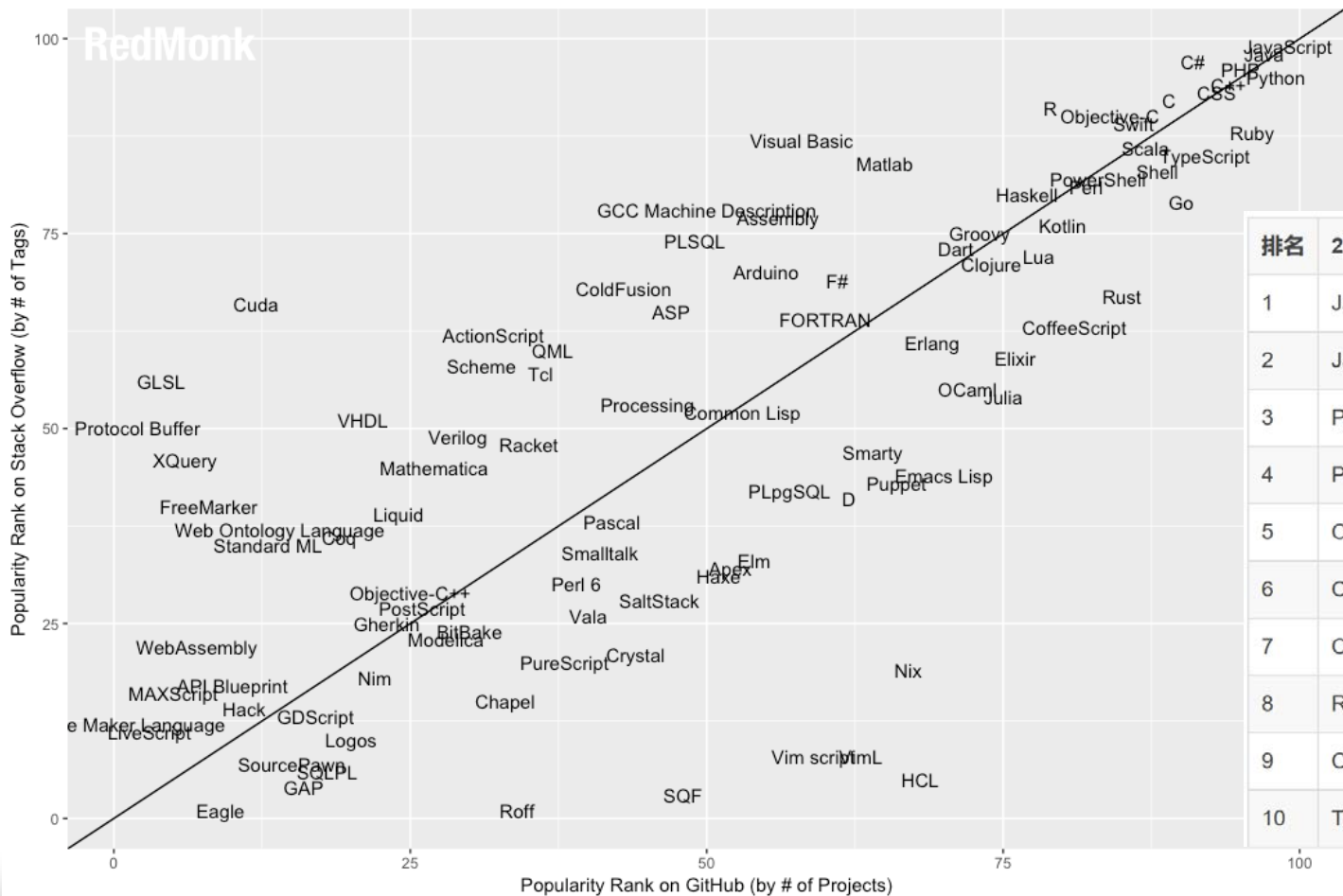
GitHub Rating



# 编程语言排行榜

RedMonk 编程语言排行榜通过追踪编程语言在 GitHub 和 Stack Overflow 上的代码使用情况与讨论数量，统计分析后进行排序，其旨在深入了解潜在的语言采用趋势。

## RedMonk Q319 Programming Language Rankings



排名	2019 年 6 月	2019 年 1 月	2018 年 6 月
1	JavaScript	JavaScript	JavaScript
2	Java	Java	Java
3	Python	Python	Python
4	PHP	PHP	PHP
5	C++	C #	C#
6	C#	C ++	C++
7	CSS	CSS	CSS
8	Ruby	Ruby	Ruby
9	C	C	C, Objective-C
10	TypeScript	Objective-C	

# 编程语言排行榜

- **Business 2 Community** 公布了其给出的最值得掌握的计算机技能。根据结果，去年的十五大语言榜单包含Java、JavaScript、Python、C++与C#，结果在意料之中。另外，作者们表示只要能够掌握其中的任何语言，您即可获得丰厚的薪酬回报。





# 编程语言排行榜

- 2016年，职业规划公司Gooroo通过仔细查看了美国、英国和澳大利亚的超过50万份的IT职位空缺，做了一个编程语言工资和供需的研究报告。

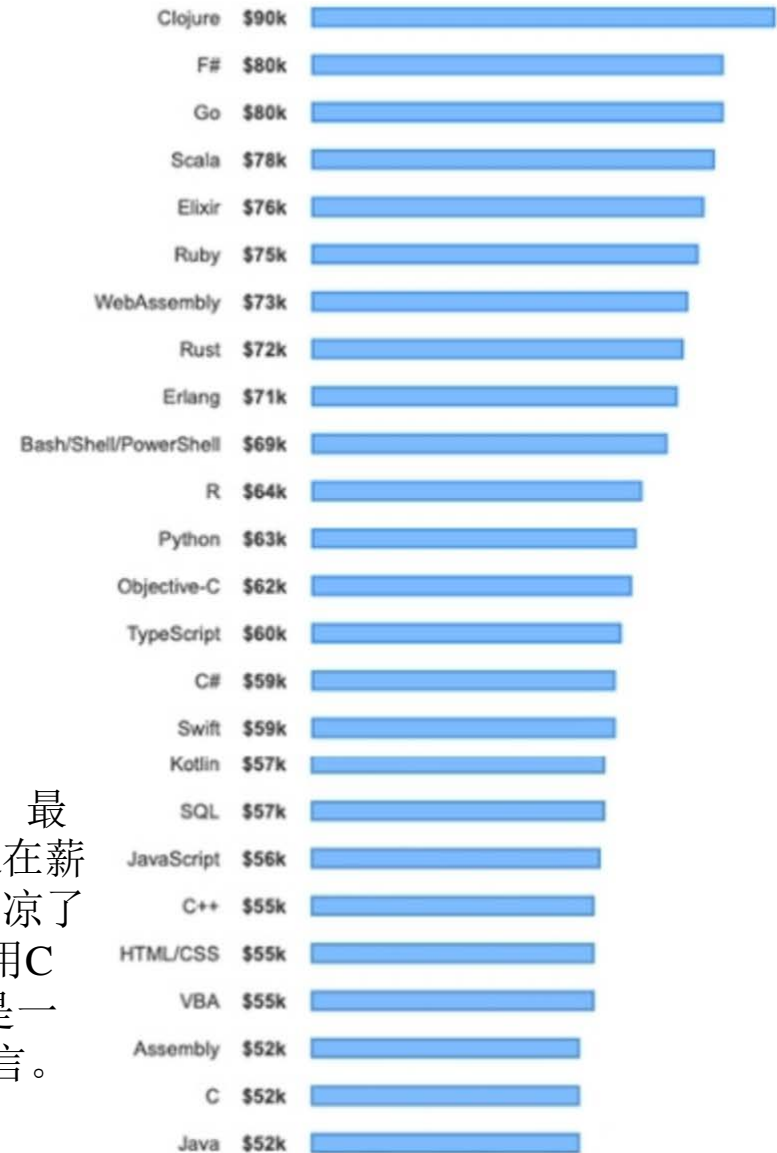
排名	编程语言	职位需求量	平均年薪（美元）
1	Java	18%	100,000
2	JavaScript	17%	90,000
3	C#	16%	85,000
4	C	9%	90,000
5	C++	9%	95,000
6	PHP	7%	75,000
7	Python	5.5%	100,000
8	R	3%	95,000
9	Scheme	3%	65,000
10	Perl	3%	100,000



# 编程语言排行榜

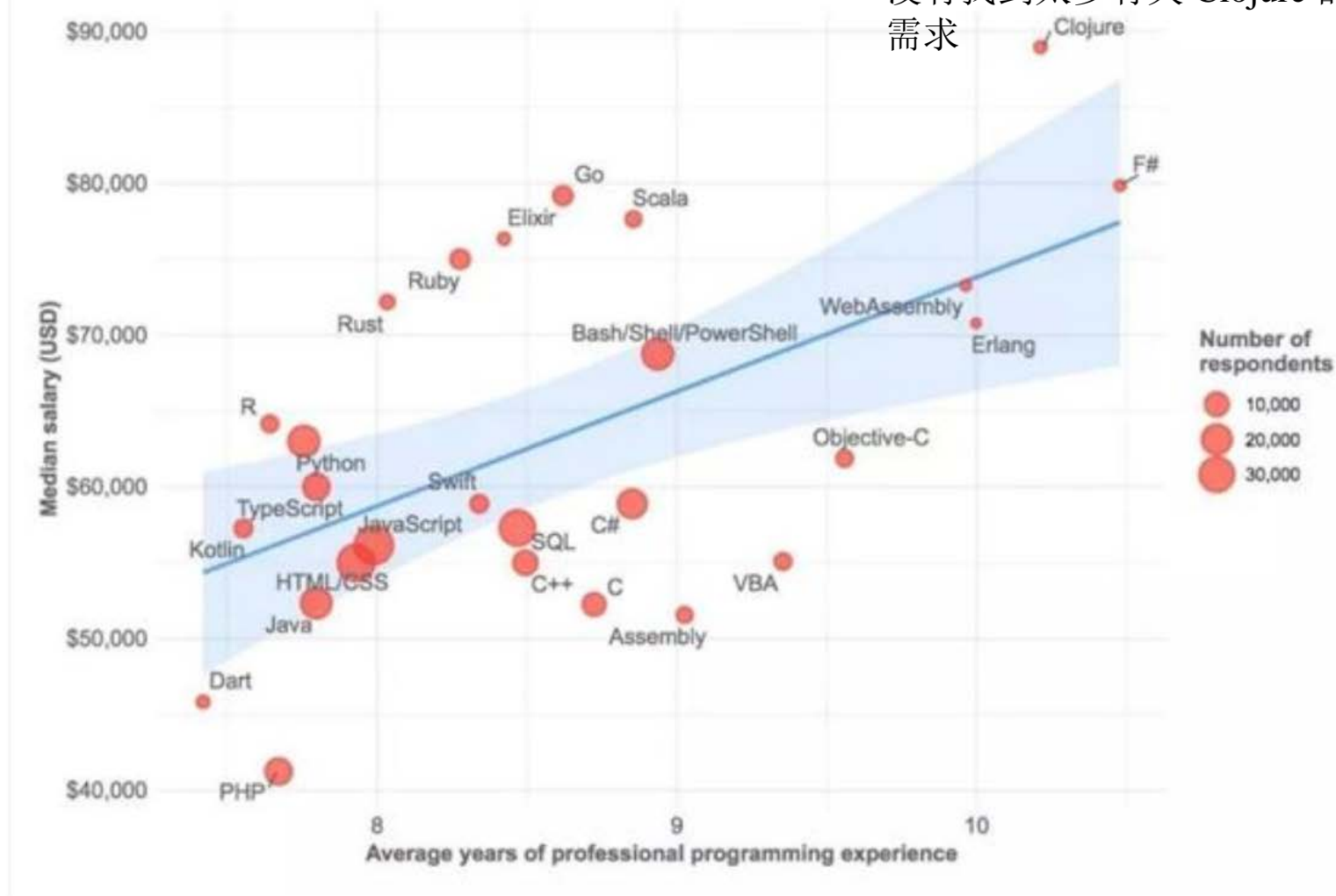
- 国外著名开发者社区Stack Overflow 对全球近 90,000 名开编程语言开发者发起调查，内容包括开发者的开发语言、薪资、工作经验（从业年长）、工作地区等信息，最后发布了2019年各编程语言从业者工资的整体水平报告。

从榜单来上看，大众语言全面垫底，小众语言独领风骚。最让大家意外的是常年占据编程语言排行版前三的C和Java在薪资排行版上居然是垫底的，但这不意味着C和Java就要凉凉了，也许正是因为热度高，所以吸引了大部分人学习和采用C和Java开发，从而拉低了整体的工资水平。同样意外的是一些小众语言异军突起，榜单前三是Clojure、F# 和 Go 语言。



## 工作经验与薪水

在国内几个招聘网站进行搜索，都没有找到太多有关 Clojure 的招聘需求

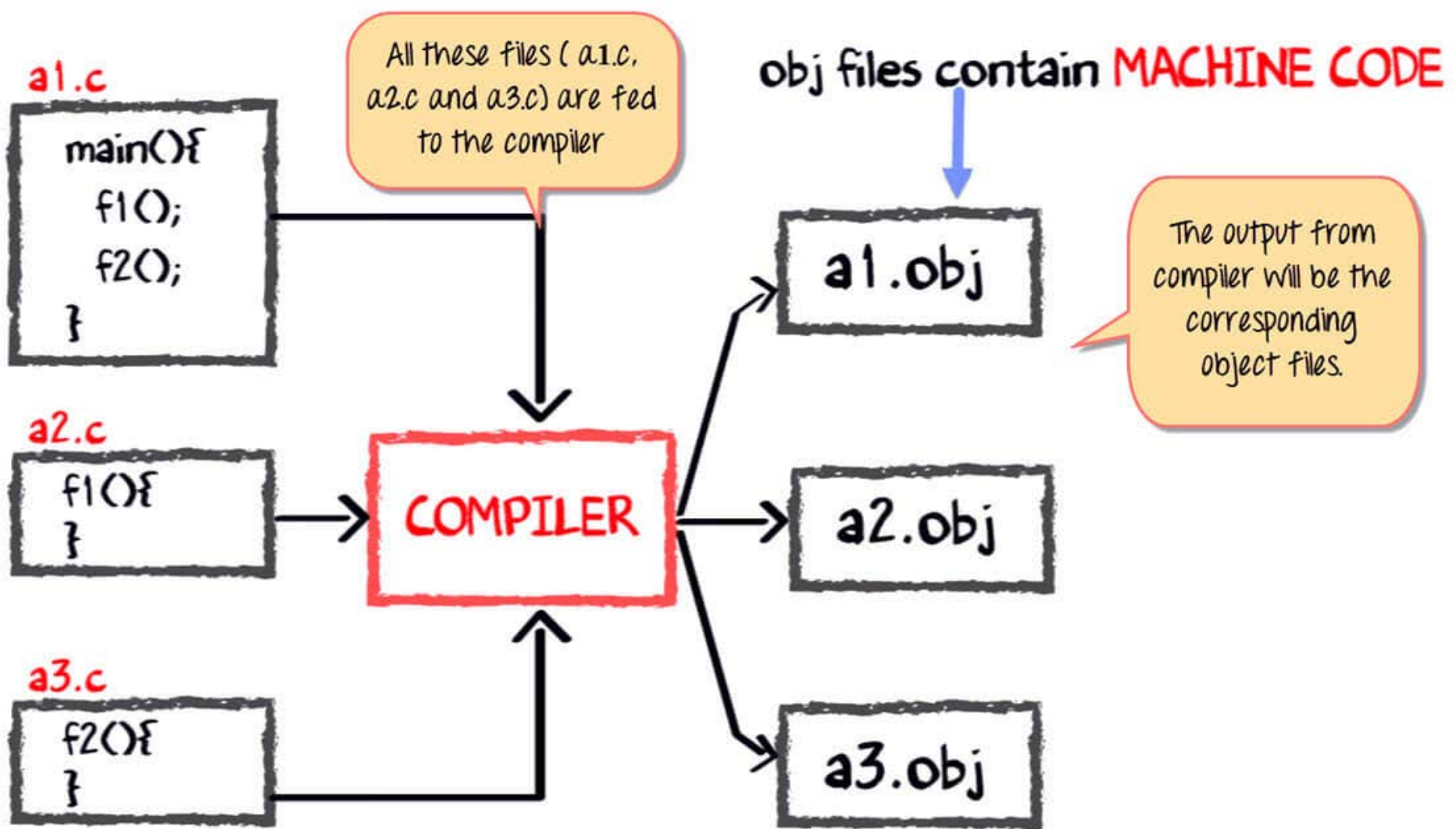


如上图，红圈的大小不同代表着受访者人数的范围不同，横轴代表着工作经验(年数)，纵轴代表着薪资中位数。

# Compiled versus Interpreted

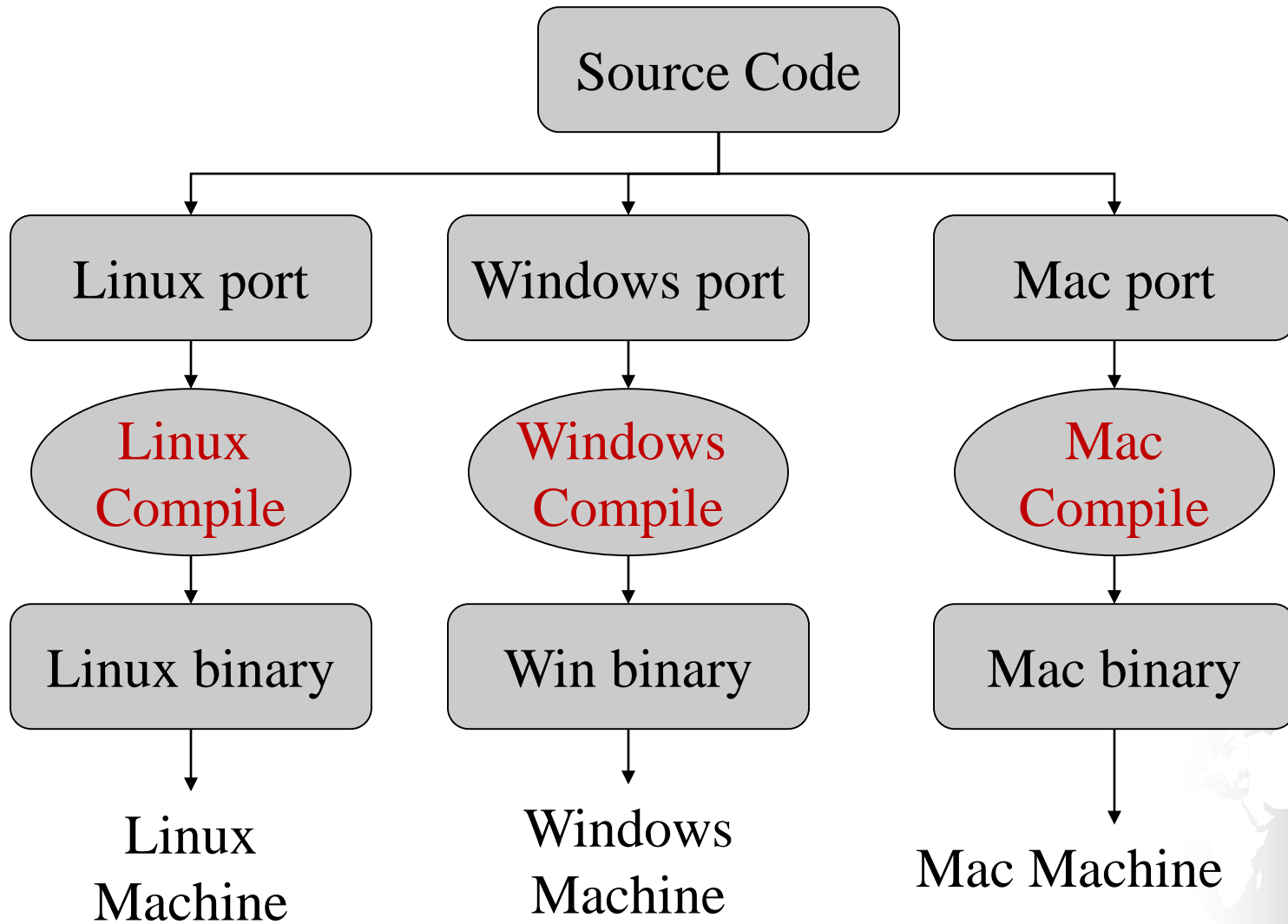
- ❑ Interpreters run source program, "interpreting" the source code on the fly.
- ❑ Compilers translate the source program to machine language.
  - Well, actually, they usually translate it to object code.
  - The object code is "linked" with other code by a linker.
- ❑ Speed
  - Compilers win
  - Optimizing compilers really win
  - (In Java automatic garbage collector runs as a low-priority background thread *e.g.* improves response.)
- ❑ Security, viruses etc.
  - Fully interpreted languages win
- ❑ Java is interpreted.
  - Well, actually, it's compiled to "bytecodes".
  - Bytecodes are interpreted by a virtual machine.
  - The virtual machine is emulated by the Web browser.
- ❑ **Java is a trade off between speed and security**



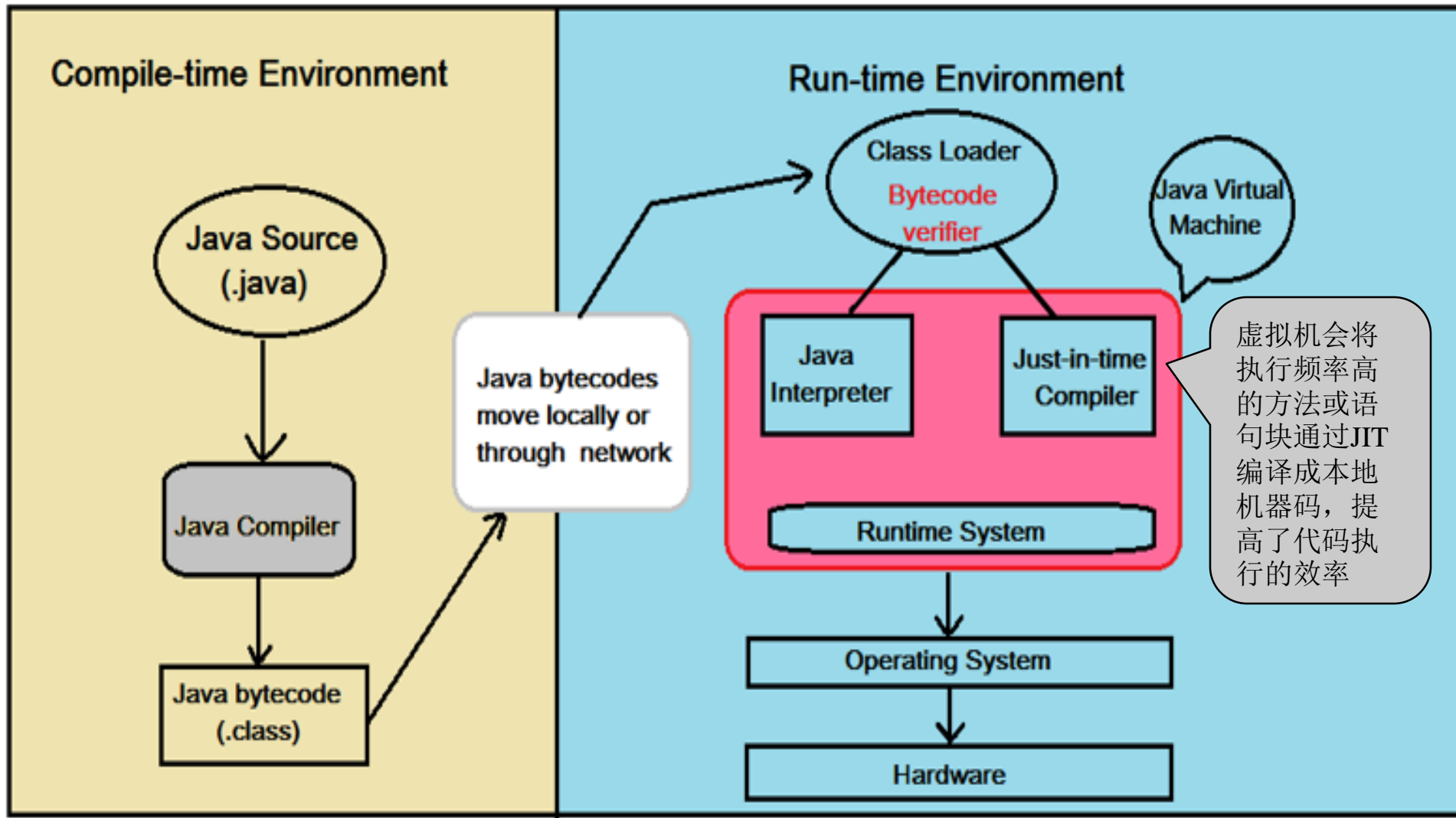


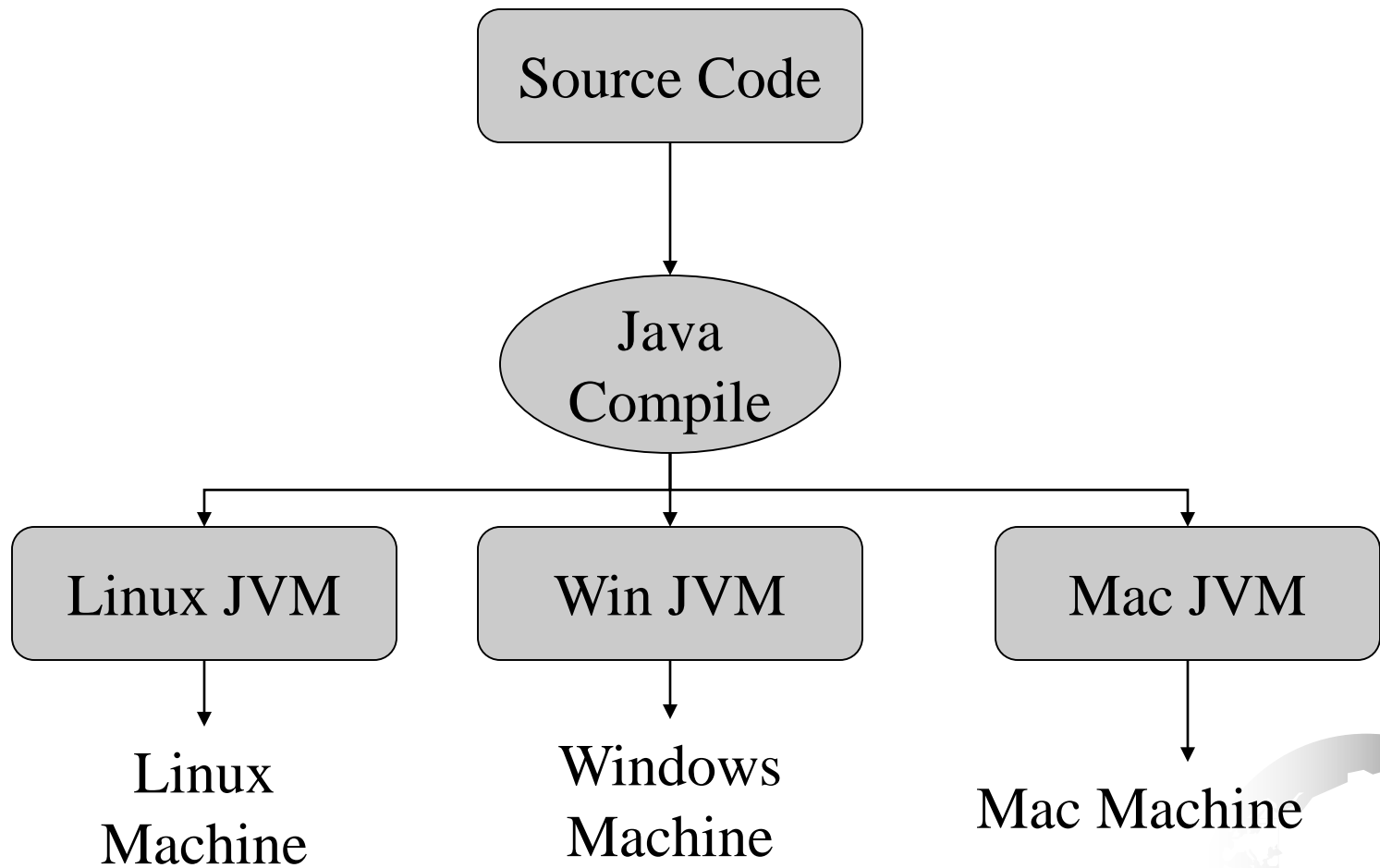
- Source code is written
- Source code is ported to every different platform
- Source code is compiled into platform specific **machine code (or binaries)**
- Binaries execute on a single platform





# How Java Does it





**Write once, compile, run anywhere!**



# Advantages of Bytecode

- Bytecode is **architecture independent** (and writing a VM is easier than rewriting a compiler for every architecture)
- VMs can enforce **different levels of security** automatically (applets versus applications)
- **Just In-Time (JIT) compiling** helps achieve same or better speed than traditional compiled code

在执行时JIT会把翻译过的机器码保存起来，以备下次使用，因此从理论上来说，采用该JIT技术，能够接近曾经纯编译技术。



Thanks!

