



SWCON201  
Opensource & Software  
Development Methods and Tools

# Selecting Right Language

Department of  
Software Convergence



# Contents

---

- Agenda
- Class
- Summary
- Reference
- Homework

**프로그래밍 언어 하나를  
친구에게 추천해 봅시다**

# Question

---

“I’m bored. Let’s learn a foreign language”

“What is the most useful foreign language?”

“Japanese, Chinese, English?”

“I’m bored. Let’s learn a programming language”

“What is the most famous programming language?”

“C, C++, Python, JavaScript?”

“Any programming language you (or I) learn will help?”

“Right?”

# Remind and Recommendation

---

- Remind

- Optimized tools for domain specific problem
  - ☑ Visit Japan? Then learn Japanese!
- Sharp and deep (fine-tuned) tools for the problem
  - ☑ Programming language itself
  - ☑ Libraries and open source softwares
  - ☑ IDEs and development environments

- Recommendation

- Clear problem definition decides the right (domain specific or friendly) programming language

# Agenda

---

- How to select right programming language?
  - Hardware specific problem?
  - Hardware independent problem?

# Agenda

---

- How to select right programming language?
  - Hardware specific (or dependent) problem?
    - Generally C, C++, Assembly
    - Arduino Kit requests C
    - Linux/Unix kernel requests platform supported C/C++
    - MacOS kernel requests Objective-C/Swift
  - Hardware independent problem?
    - Web client and server programming
    - Data analytics
    - Most high level language based approaches, etc.

# What is hardware related features in C/C++?

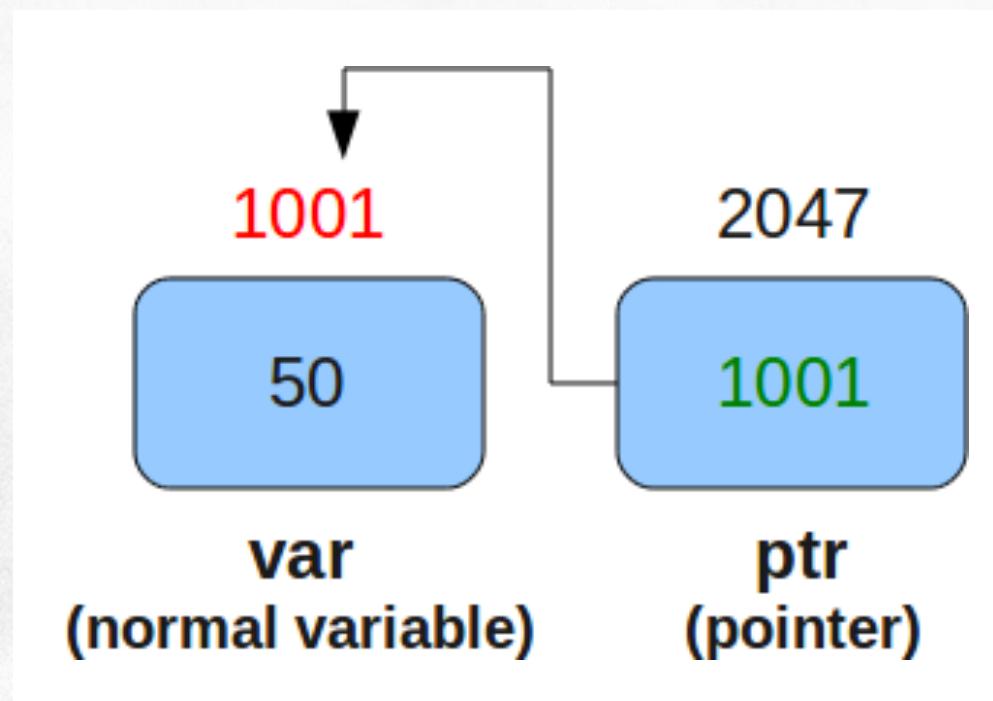
---

- Physical memory management
- Bit operation

# Physical memory management

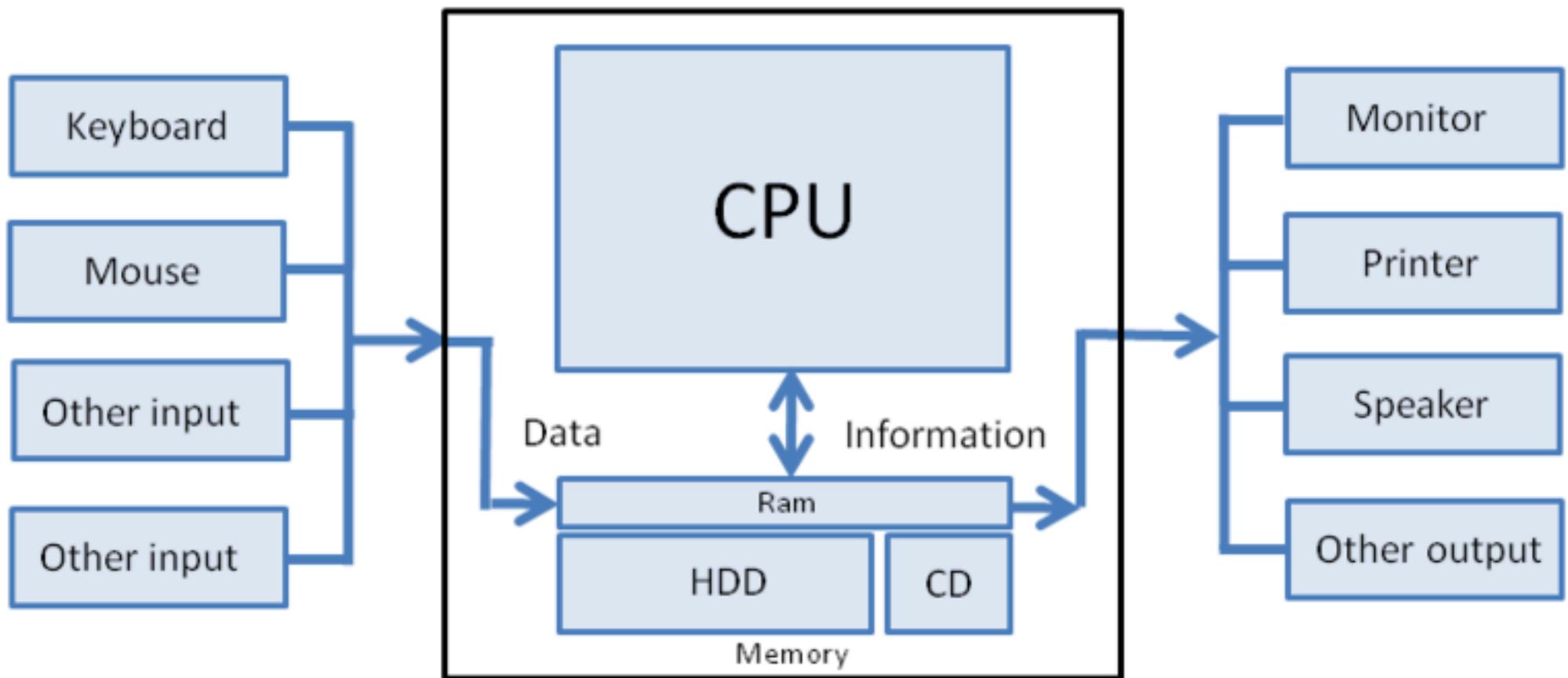
- Pointer in C/C++

- It is a variable whose value is the address of another variable, i.e., direct address of the memory location
- Like any variable or constant, you must declare a pointer before using it to store any variable address



# Physical memory management

- Memory mapped I/O (Input and Output)



# Domain Specific Languages

---

- How can we select programming language in hardware independent problem?

- Domain!!

- Web server (programming language)
- Web server (with dedicated tools)
- Data analytics
- Cloud computing
- Microservice programming
- etc.

# Web server (programming language)

---

- Javascript (JS, <https://developer.mozilla.org/ko/docs/Web/JavaScript>)
  - ▣ December 4, 1995
  - ▣ Initially only implemented client-side in web browsers
  - ▣ JavaScript engines are now embedded in many other types of host software, including server-side in web servers and databases
- Node.js (<https://nodejs.org/ko/>)
  - ▣ May 27, 2009
  - ▣ An open-source, cross-platform JavaScript run-time environment for executing JavaScript code server-side

# Web server (programming language)

- Node.js (<https://nodejs.org/ko/>)
  - ❖ HTTP server example

## Learning by Examples

Our "Show Node.js" tool makes it easy to learn Node.js, it shows both the code and the result.

### Example

```
var http = require('http');

http.createServer(function (req, res) {
    res.writeHead(200, {'Content-Type': 'text/plain'});
    res.end('Hello World!');
}).listen(8080);
```

[Run example »](#)

# Web server (with dedicated tools)

---

- Apache (<https://httpd.apache.org/>)
- Nginx (<https://nginx.org/en/>)
- Wordpress (<https://ko.wordpress.com/>)

# Data analytics (Data Analysis, Data Science)

---

- It is a process of inspecting, cleansing, transforming, and modeling data with the goal of discovering useful information, suggesting conclusions, and supporting decision-making
- Preferred languages : Why?
  - ▣ Python
  - ▣ R

# Data analytics (Data Analysis, Data Science)

---

- It is a process of inspecting, cleansing, transforming, and modeling data with the goal of discovering useful information, suggesting conclusions, and supporting decision-making
- Preferred languages : Why?
  - ▣ Python
  - ▣ R
- Major requirements
  - ▣ Natural language processing, Syntax processing, etc.
  - ▣ Linear algebra, Probability, Statistics

# Data analytics (Data Analysis, Data Science)

---

- Python (<https://www.python.org/>)
  - 20 February, 1991
  - Zen of Python
    - Beautiful is better than ugly
    - Explicit is better than implicit
    - Simple is better than complex
    - Complex is better than complicated
    - Readability counts

# Data analytics (Data Analysis, Data Science)

- R (<https://www.r-project.org/>)

```
# Goal: Show the efficiency of the mean when compared with the median
#       using a large simulation where both estimators are applied on
#       a sample of U(0,1) uniformly distributed random numbers.

one.simulation <- function(N=100) {      # N defaults to 100 if not supplied
  x <- runif(N)
  return(c(mean(x), median(x)))
}

# Simulation --
results <- replicate(100000, one.simulation(20)) # Gives back a 2x100000 matrix

# Two kernel densities --
k1 <- density(results[1,])                      # results[1,] is the 1st row
k2 <- density(results[2,])

# A pretty picture --
xrange <- range(k1$x, k2$x)
plot(k1$x, k1$y, xlim=xrange, type="l", xlab="Estimated value", ylab="")
grid()
lines(k2$x, k2$y, col="red")
abline(v=.5)
legend(x="topleft", bty="n",
       lty=c(1,1),
       col=c("black", "red"),
       legend=c("Mean", "Median"))
```

# Cloud computing

- OpenStack (<https://www.openstack.org/>)
  - Since 2010
  - controls large pools of compute, storage, and networking resources throughout a datacenter

The screenshot shows the official OpenStack website. At the top, there's a dark header bar with the text "An OpenInfra Project" and a grid icon. Below it is a navigation bar with links for "openstack.", "SEARCH", "SOFTWARE", "USE CASES", "EVENTS", "COMMUNITY", "MARKETPLACE", "BLOG", "DOCS", "JOIN", and "LOG IN". The main content area features a large blue title "The Most Widely Deployed Open Source Cloud Software in the World". Below it, a subtext reads "Deployed by thousands. Proven production at scale. OpenStack is a set of software components that provide common services for cloud infrastructure." A red button labeled "BROWSE OPENSTACK COMPONENTS" is visible. To the right, there's a graphic of the OpenStack logo (a red stylized 'E') surrounded by various blue icons representing different cloud services like databases, storage, and networking.

**The Most Widely Deployed Open Source Cloud Software in the World**

Deployed by thousands. Proven production at scale. OpenStack is a set of software components that provide common services for cloud infrastructure.

BROWSE OPENSTACK COMPONENTS

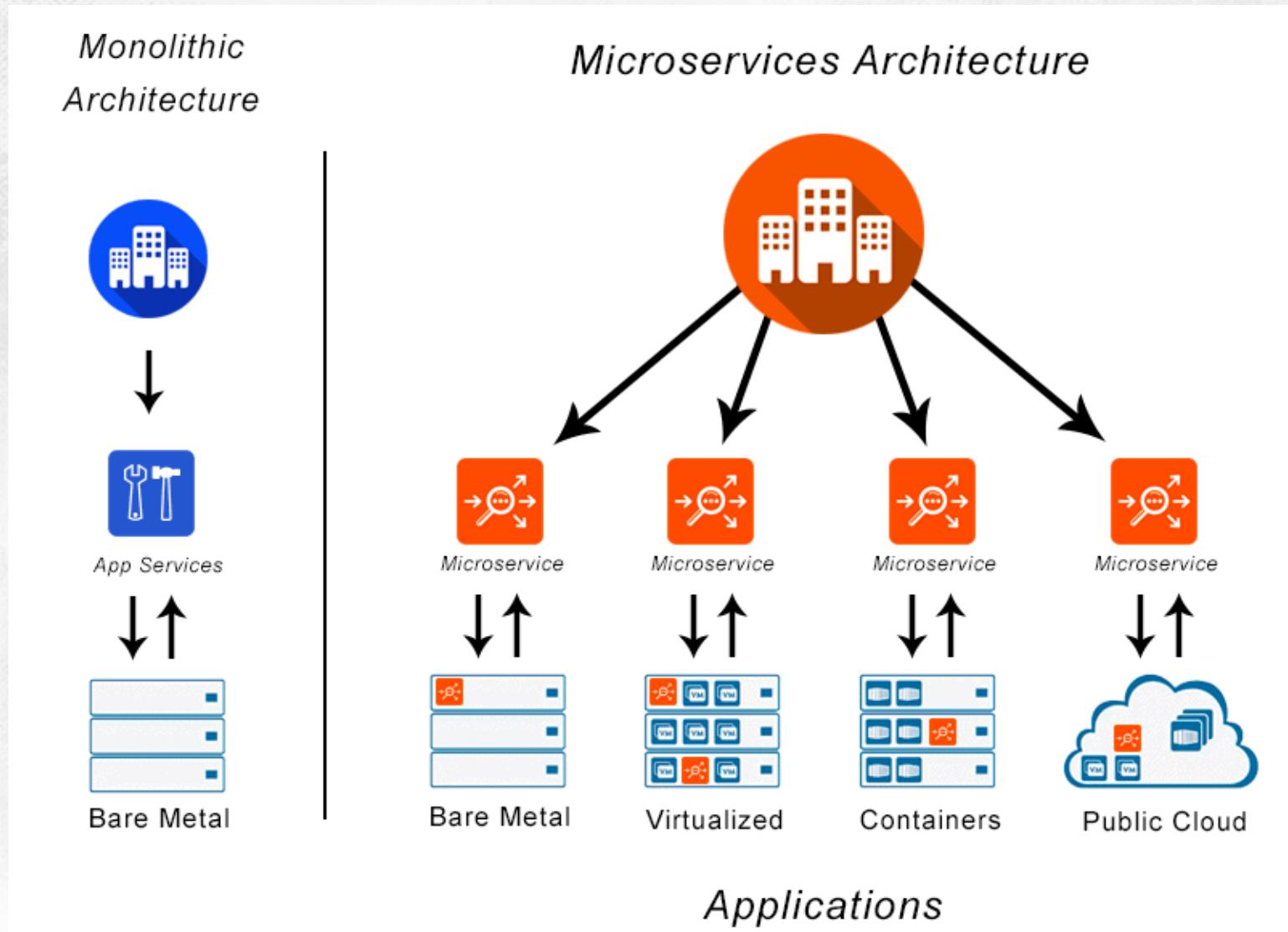
OpenStack is developed by the community. For the community. [Learn how to contribute →](#)

# Cloud computing

	OpenStack	Ceph	KVM	Open vSwitch
시작 년도	2010년	2007년 이전	2007년 이전	2007년 (공식은 2009년)
최초 개발	NASA, RackSpace	Sage Well (박사학위논문)	Kivity	-
관리 주체	OpenStack Foundation	RedHat	Open Virtualization Alliance	<a href="http://openvswitch.org">openvswitch.org</a>
참여 기관	500여사	-	200여사	-
개발 언어	Python	C++, Perl	C	C

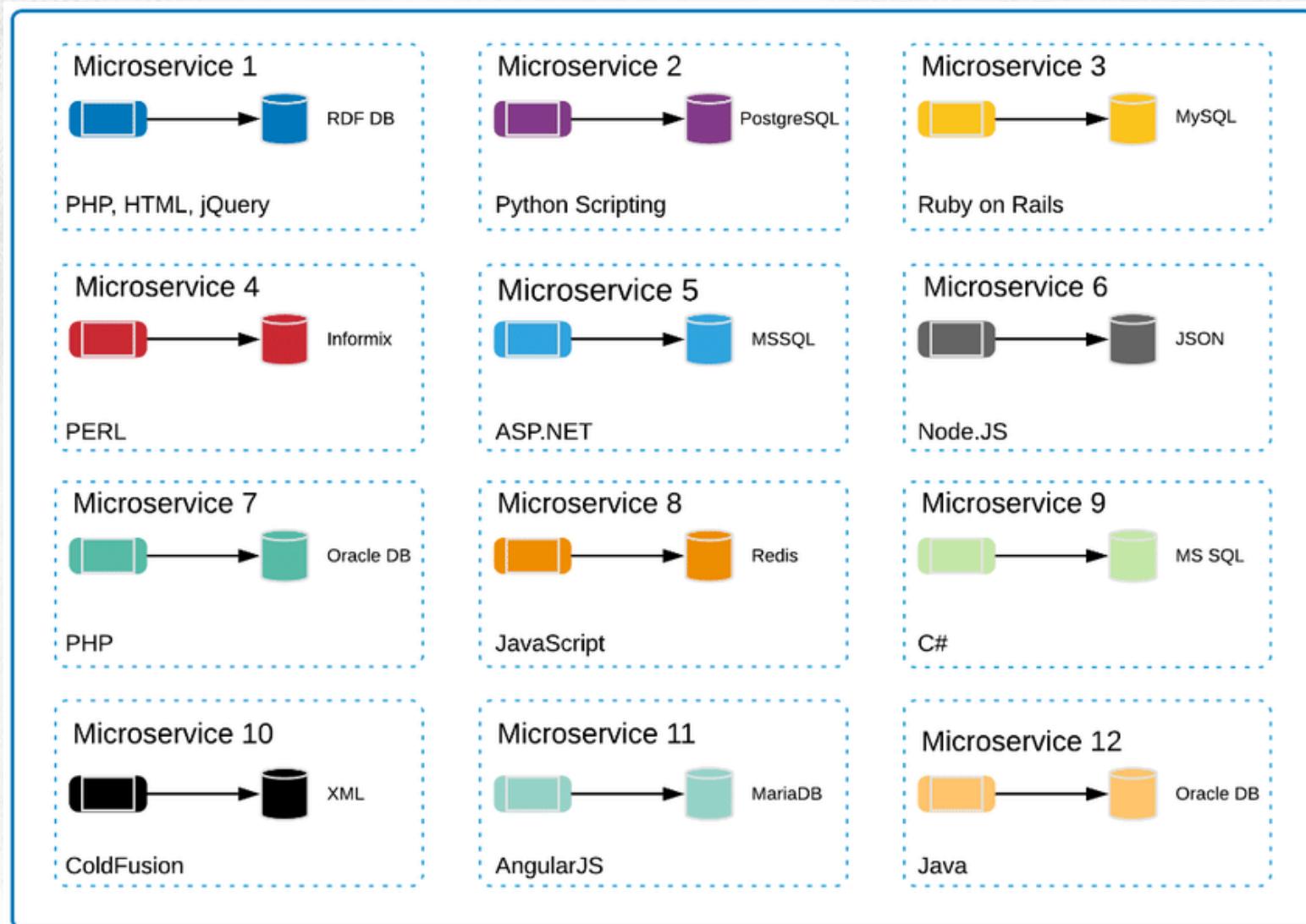
# Microservice programming

## ● Microservice Programming for Datacenter



# Microservice programming

## • **Polyglot** Programming for Datacenter



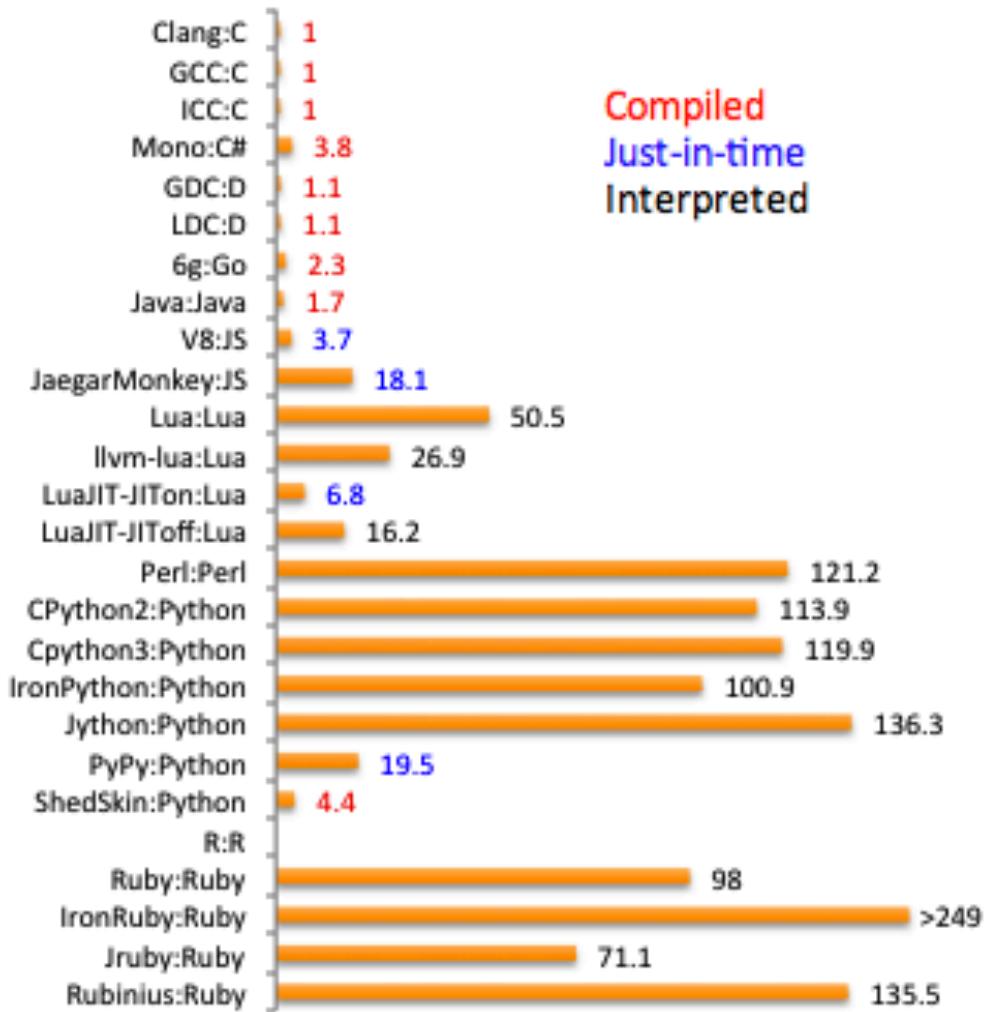
# Microservice programming

● **Polyglot** Programming for Datacenter

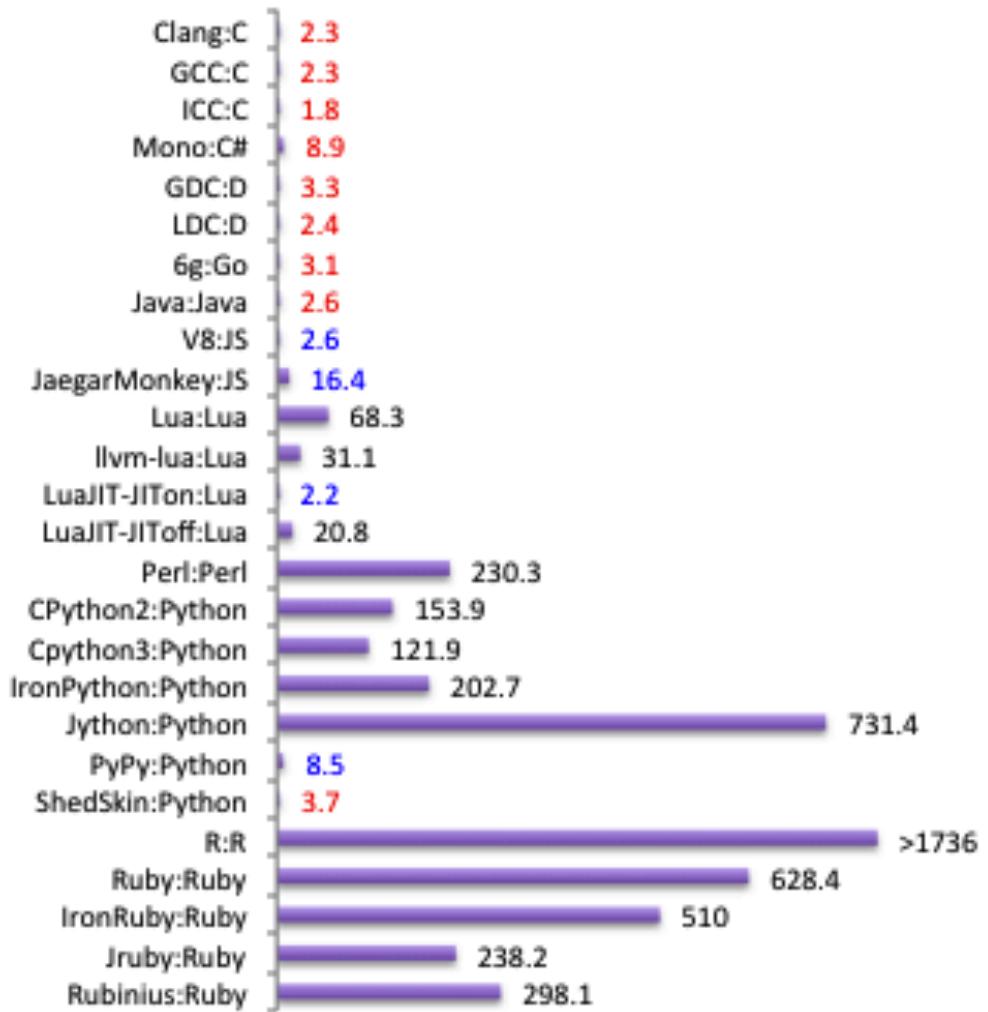


# Selection by Performance (execution time)

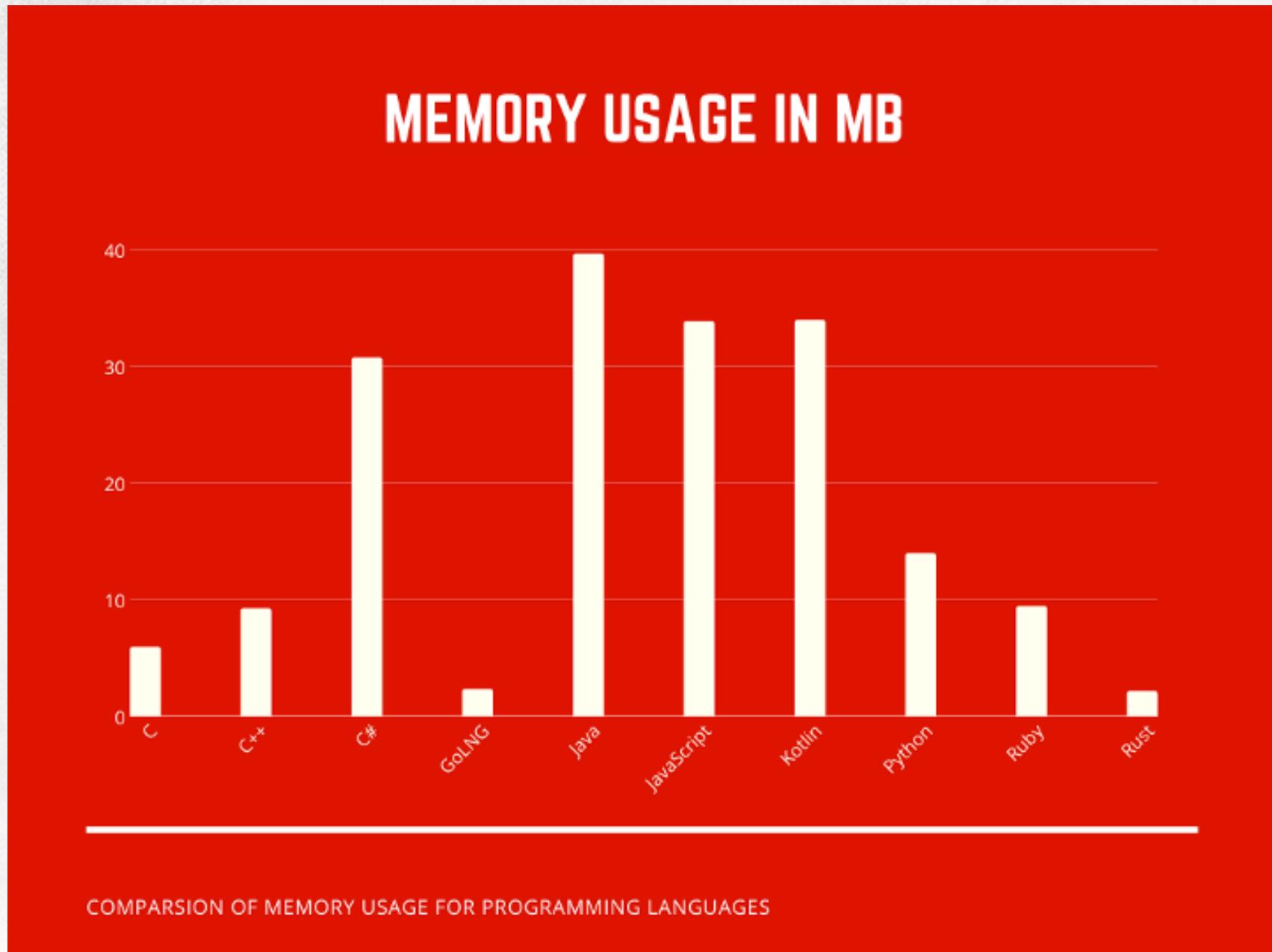
## Sudoku solving (CPU sec)



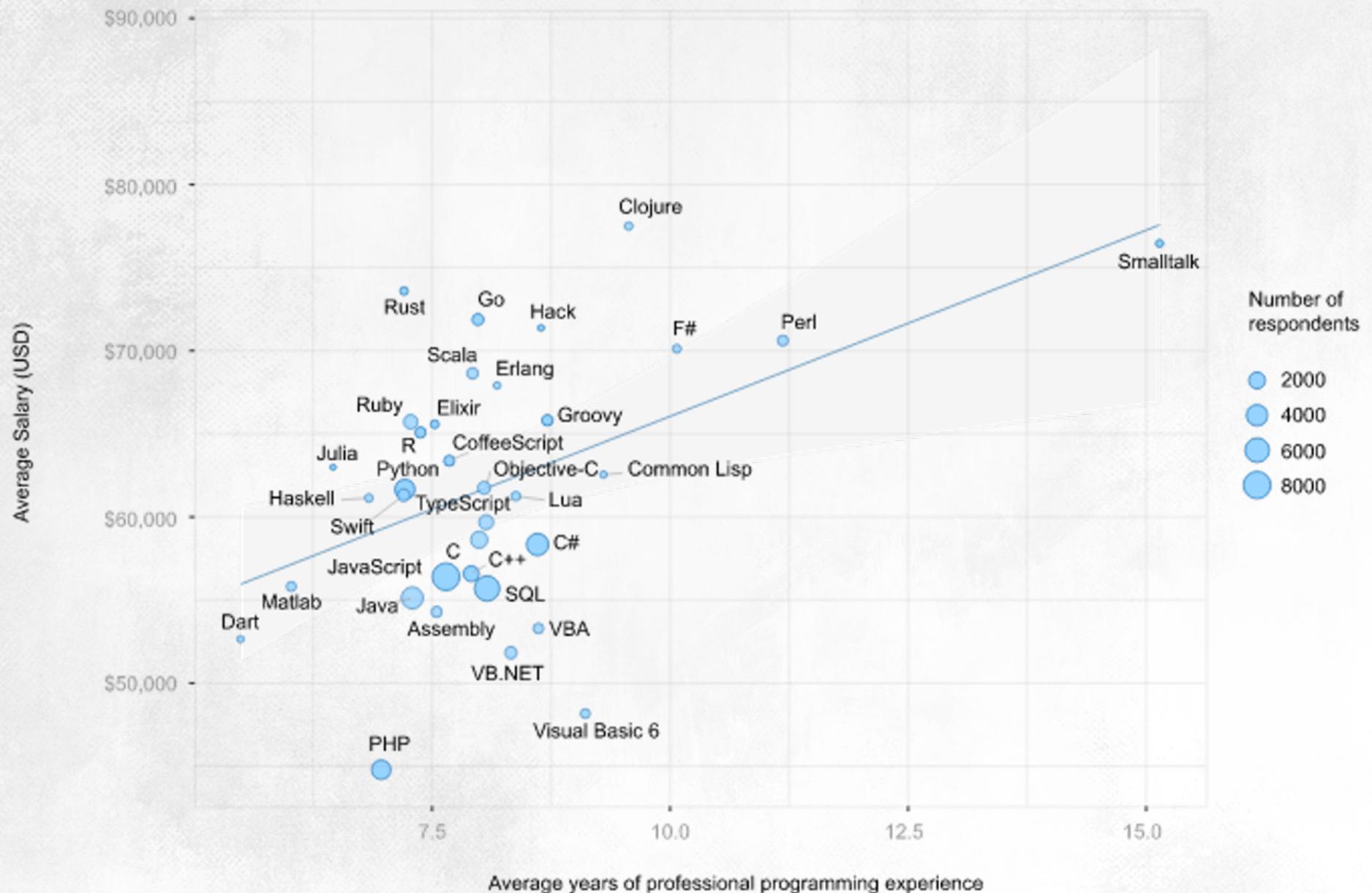
## Matrix multiplication (CPU sec)



# Selection by Performance (memory usage)



# Selection by Salary

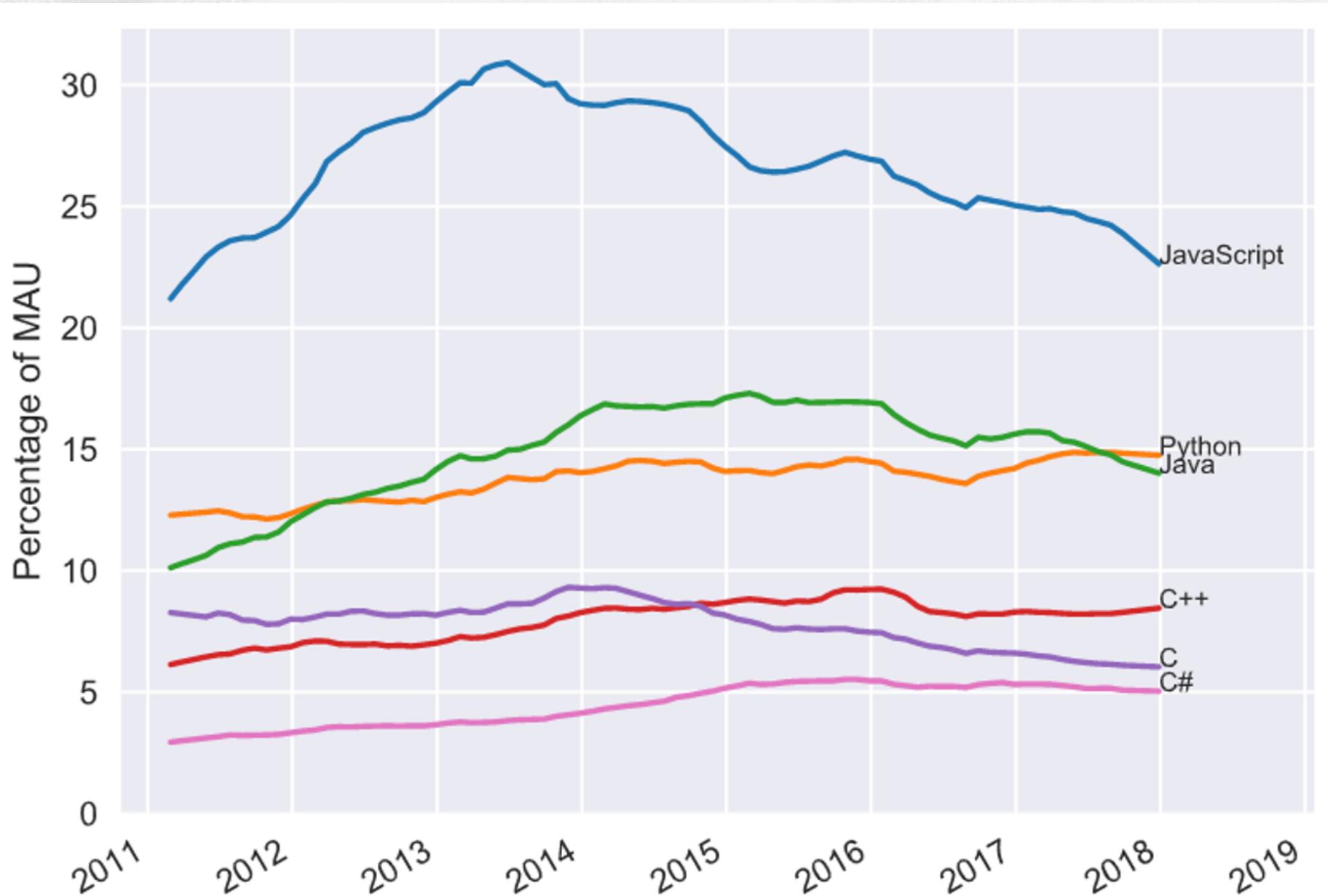


# Selection by GitHub popularity

---

- GitHub is home to open source projects written in **337** unique programming languages

# Selection by GitHub popularity



# Selection by Tiobe index

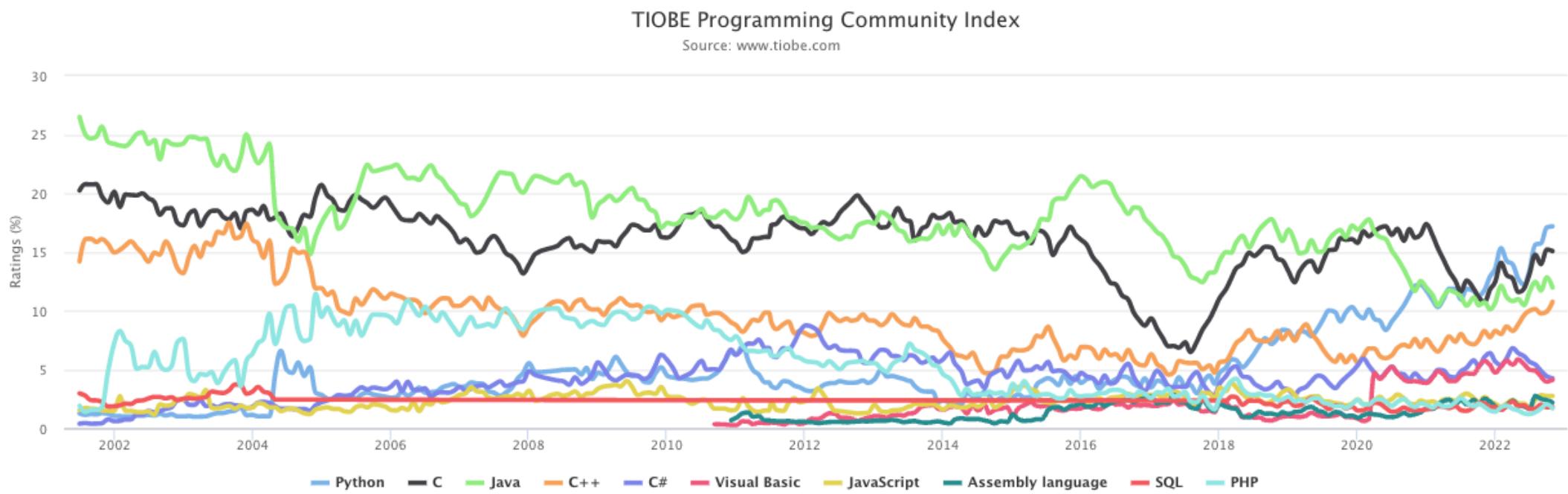
---

- Tiobe index is a measure of popularity of programming languages, created and maintained by the TIOBE Company based in Eindhoven, the Netherlands
- It is calculated from the number of search engine results for queries containing the name of the language
- It covers searches in Google, Google Blogs, MSN, Yahoo!, Baidu, Wikipedia and YouTube

# Selection by Tiobe index

Nov 2022	Nov 2021	Change	Programming Language	Ratings	Change
1	1		 Python	17.18%	+5.41%
2	2		 C	15.08%	+4.35%
3	3		 Java	11.98%	+1.26%
4	4		 C++	10.75%	+2.46%
5	5		 C#	4.25%	-1.81%
6	6		 Visual Basic	4.11%	-1.61%
7	7		 JavaScript	2.74%	+0.08%
8	8		 Assembly language	2.18%	-0.34%
9	9		 SQL	1.82%	-0.30%
10	10		 PHP	1.69%	-0.12%
11	18		 Go	1.15%	-0.06%
12	15		 R	1.14%	-0.14%

# Selection by Tiobe index



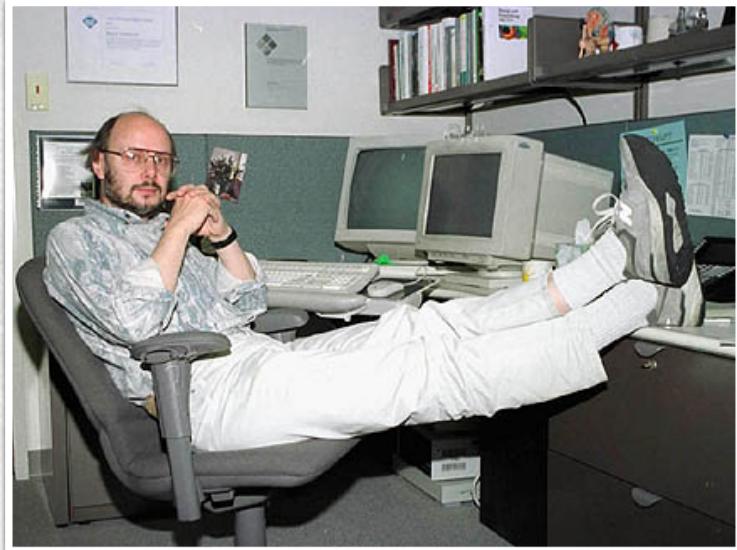
# SWCON's Choice : C++

---

- In 1979, Bjarne Stroustrup, a Danish computer scientist, began work on "C with Classes", the predecessor to C++
- Initially, Stroustrup's "C with Classes" added features to the C compiler, Cpre, including classes, derived classes, strong typing, inlining and default arguments
- In 1983, "C with Classes" was renamed to "C++"
- In 1985, the first edition of The C++ Programming Language was released, which became the definitive reference for the language, as there was not yet an official standard
- In 1989, C++ 2.0 was released, followed by the updated second edition of The C++ Programming Language in 1991
- As of 2017, C++ remains the third most popular programming language, behind Java and C

# SWCON's Choice : C++

- Bjarne Stroustrup



# SWCON's Choice : C++

## ● Bjarne Stroustrup

[home](#) | [C++](#) | [FAQ](#) | [technical FAQ](#) | [publications](#) | [WG21 papers](#) | [TC++PL](#) | [Tour++](#) | [Programming](#) | [D&E](#) | [bio](#) | [interviews](#) | [videos](#) | [quotes](#) | [applications](#) | [guidelines](#) | [compilers](#)

### Welcome to Bjarne Stroustrup's homepage!



I designed and implemented [the C++ programming language](#). To make C++ a stable and up-to-date base for real-world software development, I have stuck with its ISO standards effort for 30+ years (so far). I'm a professor of Computer Science in Columbia University in New York City.

- [A Tour of C++ \(3rd edition\)](#): a brief (254 pages + index, historical information, etc.) tour of the C++ Programming language and its standard library for experienced programmers.
- [The C++ Programming Language \(4th edition\)](#): an exhaustive description of the C++ Programming language, its standard library, and fundamental techniques for experienced programmers.
- [Programming: Principles and Practice using C++ \(2nd edition\)](#): a programming text book aimed at beginners who want eventually to become professionals.
- [The Design and Evolution of C++](#): a book presenting the rationale and design criteria for C++ and its evolution up until 1994.
- [Research and popular papers](#)
- [Technical reports and proposals for the ISO C++ Standard](#)
- [Videos](#)
- [Interviews](#)
- [Biographical material](#),

[home](#) | [C++](#) | [FAQ](#) | [technical FAQ](#) | [publications](#) | [WG21 papers](#) | [TC++PL](#) | [Tour++](#) | [Programming](#) | [D&E](#) | [bio](#) | [interviews](#) | [videos](#) | [quotes](#) | [applications](#) | [guidelines](#) | [compilers](#)

# SWCON's Choice : C++

---

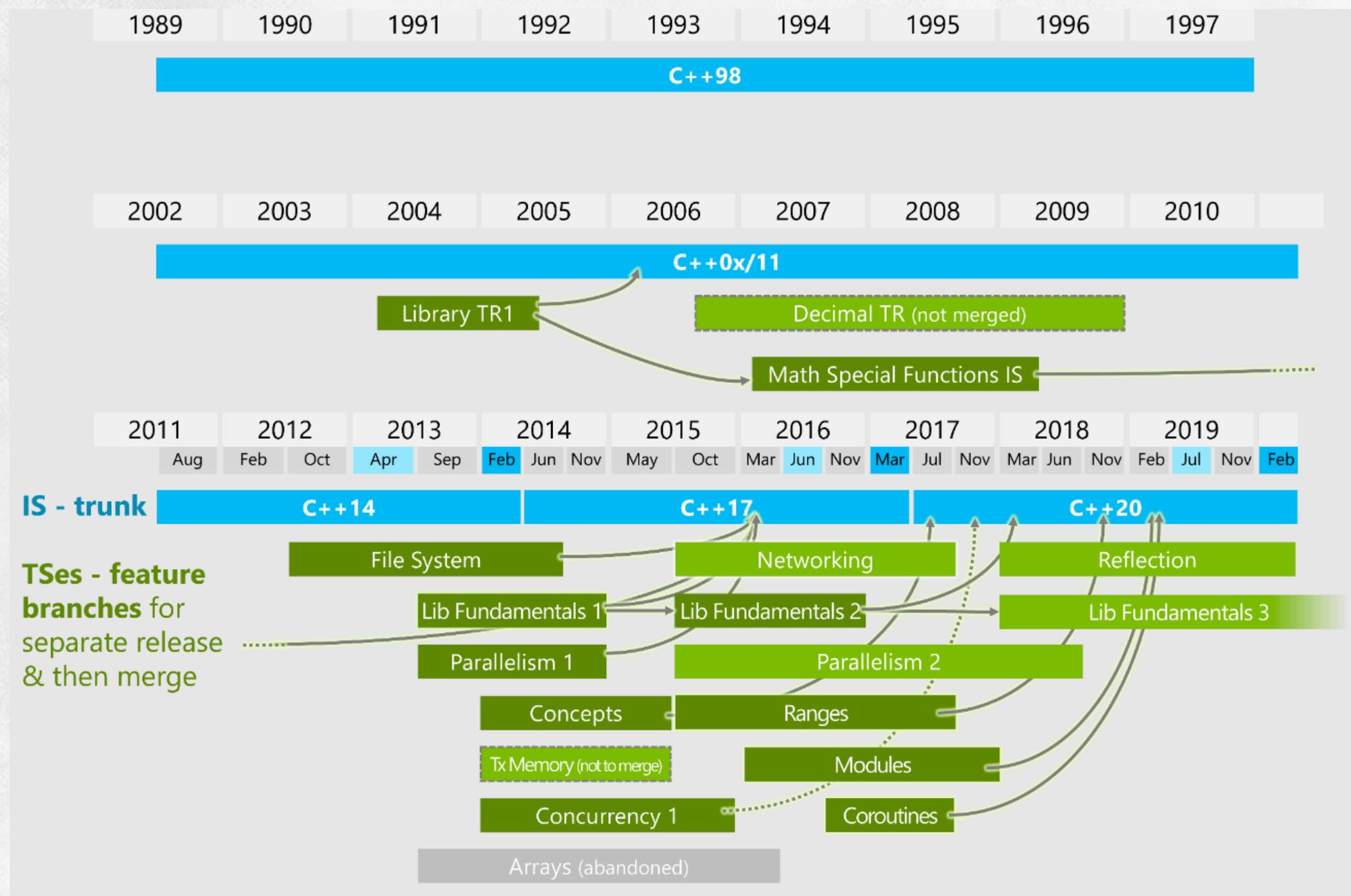
- General-purpose programming language. It has imperative, object-oriented and generic programming features, while also providing facilities for low-level memory manipulation
- Bias toward system programming and embedded, resource-constrained and large systems, with performance, efficiency and flexibility of use as its design highlights
- C++ has also been found useful in many other contexts, with key strengths being software infrastructure and resource-constrained applications, including desktop applications, servers (e.g. e-commerce, web search or SQL servers), and performance-critical applications (e.g. telephone switches or space probes)
- C++ is a compiled language, with implementations of it available on many platforms

# SWCON's Choice : C++

---

- Many vendors provide C++ compilers, including the Free Software Foundation, Microsoft, Intel, and IBM
- Many other programming languages have been influenced by C++, including C#, D, Java, and newer versions of C

# SWCON's Choice : C++



# SWCON's Choice : C++

2020	2021	2022	2023	2024	2025	2026	2027	2028
Jun	Nov	Feb	Jun	Oct	Feb	Jul	Nov	Mar

C++23

C++26

C++29

Lib Fundamentals 3

Tx Memory 2

Concurrency 2



#### FEATURES

Current ISO C++ status

Upcoming ISO C++ meetings

Upcoming C++ conferences

Compiler conformance status

#### TAGS

basics intermediate  
advanced experimental

#### UPCOMING EVENTS

WG 21 autumn meeting  
Nov 7-12, Zoom and Kona, HI, USA

#### MEETING C++ 2022

Nov 17-19, Online and Berlin,  
Germany

#### ACCU 2023

Apr 19-22, Online and Bristol,  
UK

#### FOLLOW US!



## News, Status & Discussion about Standard C++

### Follow All Posts

The home of Standard C++ on the web — news, status and discussion about the C++ standard on all compilers and platforms.

### Recent Highlights

C++ constexpr parlor tricks: Obtain the length of a string at compile time – Raymond Chen

By Blog Staff | Nov 16, 2022 07:37 PM



The Proxy Pattern – Rainer Grimm

By Blog Staff | Nov 16, 2022 07:35 PM

Using final in C++ to improve performance – Niall Cooling

By Blog Staff | Nov 16, 2022 07:27 PM

For Software Performance, the Way Data is Accessed Matters! – Ivica Bogosavljević

By Blog Staff | Nov 12, 2022 06:01 PM

Standard C++ Foundation Annual Report for Fiscal Year 2022

By Blog Staff | Nov 12, 2022 02:48 PM

### Articles & Books

C++ constexpr parlor tricks: Obtain the length of a string at compile time – Raymond Chen

By Blog Staff | Nov 16, 2022 07:37 PM



The Proxy Pattern – Rainer Grimm

By Blog Staff | Nov 16, 2022 07:35 PM

Using final in C++ to improve performance – Niall Cooling

By Blog Staff | Nov 16, 2022 07:27 PM

For Software Performance, the Way Data is Accessed Matters! – Ivica Bogosavljević

### Recent Cpp.Chat Podcasts

The Curse of Backwards Compatibility

Date: Fri, 11 Feb 2022 00:00:00 +0000

### Recent C++ Weekly Podcasts

C++23's move\_only\_function

Date: Mon, 7 Nov 2022 16:29:47 GMT

A Modern C++ Quick Start Tutorial - 90 Topics in 20 Minutes

Date: Mon, 31 Oct 2022 15:29:54 GMT

This PlayStation Jailbreak NEVER SHOULD HAVE HAPPENED

Date: Mon, 24 Oct 2022 15:29:46 GMT

C++23's bind\_back

Date: Mon, 17 Oct 2022 15:24:49 GMT

### Recent 'No Diagnostic Required' Podcasts

Episode #14 - January 2022

Date: Wed, 16 Feb 2022 00:00:00 +0000

### Product News

# C++ Standard Site

<https://isocpp.org/>

# SWCON's Choice : Python

---

- Paradigm : Multi-paradigm: object-oriented, procedural (imperative), functional, structured, reflective
- Designed by : Guido van Rossum
- Developer : Python Software Foundation
- First appeared : 20 February 1991
- Status : Alive
- Typing discipline : Duck, dynamic, strong typing
- OS : Windows, macOS, Linux/UNIX, Android and more
- License : Python Software Foundation License
- Website : [python.org](http://python.org)

# SWCON's Choice : Python

---

- Guido van Rossum began working on Python in the late 1980s as a successor to the ABC programming language and first released it in 1991 as Python 0.9.0.
- Python 2.0 was released in 2000 and introduced new features such as list comprehensions, cycle-detecting garbage collection, reference counting, and Unicode support.
- Python 3.0, released in 2008, was a major revision that is not completely backward-compatible with earlier versions.
- Python 2 was discontinued with version 2.7.18 in 2020.

# SWCON's Choice : Python

## ● Guido van Rossum



### Guido van Rossum - Personal Home Page

["Gawky and proud of it."](#)

#### [Who I Am](#)

Read my ["King's Day Speech"](#) for some inspiration.

I am the author of the [Python](#) programming language. See also my [resume](#) and my [publications list](#), a [brief bio](#), assorted [writings](#), [presentations](#) and [interviews](#) (all about Python), some [pictures of me](#), [my new blog](#), and my [old blog](#) on Artima.com. I am [@gvanrossum](#) on Twitter.

I am currently a Distinguished Engineer at Microsoft. I have worked for Dropbox, Google, Elemental Security, Zope Corporation, BeOpen.com, CNRI, CWI, and SARA. (See my [resume](#).) I created Python while at CWI.

#### [How to Reach Me](#)

You can send email for me to [guido \(at\) python.org](mailto:guido@python.org). I read everything sent there, but I receive too much email to respond to everything.

#### [My Name](#)

My name often poses difficulties for Americans.

**Pronunciation:** in Dutch, the "G" in Guido is a hard G, pronounced roughly like the "ch" in Scottish "loch". (Listen to the [sound clip](#).) However, if you're American, you may also pronounce it as the Italian "Guido". I'm not too worried about the associations with mob assassins that some people have. :-)

**Spelling:** my last name is two words, and I'd like to keep it that way, the spelling on some of my credit cards notwithstanding. Dutch spelling rules dictate that when used in combination with my first name, "van" is not capitalized: "Guido van Rossum". But when my last name is used alone to refer to me, it is capitalized, for example: "As usual, Van Rossum was right."

**Alphabetization:** in America, I show up in the alphabet under "V". But in Europe, I show up under "R". And some of my friends put me under "G" in their address book...

#### [More Hyperlinks](#)

- Here's a collection of [essays](#) relating to Python that I've written, including the foreword I wrote for Mark Lutz' book "Programming Python".
- I own the official  [Python license](#).

#### [The Audio File Formats FAQ](#)

I was the original creator and maintainer of the Audio File Formats FAQ. It is now maintained by Chris Bagwell at <http://www.cnpbagwell.com/audio-faq>. And here is a link to [SOX](#), to which I contributed some early code.

["On the Internet, nobody knows you're a dog."](#)

# SWCON's Choice : Python

The screenshot shows the official Python website. At the top is a dark navigation bar with tabs: Python (selected), PSF, Docs, PyPI, Jobs, and Community. Below the bar is the Python logo and a search bar with 'Search' and 'GO' buttons. A yellow 'Donate' button is also visible. The main content area features a blue navigation bar with links: About, Downloads, Documentation (selected), Community, Success Stories, News, and Events. To the left, there's a code editor window displaying Python code. To the right, there's a section titled 'Quick & Easy to Learn' with a sub-section about Python's syntax and a 'Learn More' link.

# Simple output (with Unicode)  
>>> print(\"Hello, I'm Python!\")  
Hello, I'm Python!  
# Input, assignment  
>>> name = input('What is your name?\n')  
What is your name?  
Python  
>>> print(f'Hi, {name}.')  
Hi, Python.

**Quick & Easy to Learn**

Experienced programmers in any other language can pick up Python very quickly, and beginners find the clean syntax and indentation structure easy to learn. [Whet your appetite](#) with our Python 3 overview.

1 2 3 4 5

Python is a programming language that lets you work quickly and integrate systems more effectively. [» Learn More](#)

Python Official Site : <https://www.python.org/>

# SWCON's Choice : Python

## Get Started

Whether you're new to programming or an experienced developer, it's easy to learn and use Python.

[Start with our Beginner's Guide](#)

## Download

Python source code and installers are available for download for all versions!

Latest: [Python 3.11.0](#)

## Docs

Documentation for Python's standard library, along with tutorials and guides, are available online.

[docs.python.org](#)

## Jobs

Looking for work or have a Python related position that you're trying to hire for? Our **relaunched community-run job board** is the place to go.

[jobs.python.org](#)

## Latest News

[»» More](#)

- 2022-11-15    [Python 3.12.0 alpha 2 released](#)
- 2022-11-07    [It's time for our annual year-end PSF fundraiser and membership drive!](#)
- 2022-11-01    [Thank You for Making PyCon US amazing, Jackie!](#)
- 2022-10-26    [Announcing Python Software Foundation Fellow Members for Q3 2022!](#)
- 2022-10-25    [Python Software Foundation - October 2022 Newsletter](#)

## Upcoming Events

[»» More](#)

- 2022-11-23    [NZPUG-Auckland Coding Challenge "Office Hours"](#)
- 2022-11-26    [Pyjamas Conf 2022](#)
- 2022-11-26    [Django Girls Groningen](#)
- 2022-11-29    [Getting back to Python land - Stockholm Python User Group](#)
- 2022-11-30    [NZPUG-Auckland Coding Challenge "Office Hours"](#)

## Success Stories

[»» More](#)

*Python and its broad variety of libraries are very well suited to develop customized machine learning tools which tackle the complex challenges posed by financial time series.*

Python for Financial Machine Learning at Union Investment by Dr. Christian Mandery and Nikolas Gerlich

## Use Python for...

[»» More](#)

**Web Development:** [Django](#), [Pyramid](#), [Bottle](#), [Tornado](#), [Flask](#), [web2py](#)

**GUI Development:** [tkinter](#), [PyGObject](#), [PyQt](#), [PySide](#), [Kivy](#), [wxPython](#)

**Scientific and Numeric:** [SciPy](#), [Pandas](#), [IPython](#)

**Software Development:** [Buildbot](#), [Trac](#), [Roundup](#)

**System Administration:** [Ansible](#), [Salt](#), [OpenStack](#), [xonsh](#)

# SWCON's Choice : Python

»» Python Enhancement Proposals: The future of Python



## »» Python Software Foundation

The mission of the Python Software Foundation is to promote, protect, and advance the Python programming language, and to support and facilitate the growth of a diverse and international community of Python programmers. »» Learn more

[Become a Member](#)

[Donate to the PSF](#)



About	Downloads	Documentation	Community	Success Stories	News
Applications	All releases	Docs	Diversity	Arts	Python News
Quotes	Source code	Audio/Visual Talks	Mailing Lists	Business	PSF Newsletter
Getting Started	Windows	Beginner's Guide	IRC	Education	Community News
Help	macOS	Developer's Guide	Forums	Engineering	PSF News
Python Brochure	Other Platforms	FAQ	PSF Annual Impact Report	Government	PyCon News
	License	Non-English Docs	Python Conferences	Scientific	
Events	Alternative Implementations	PEP Index	Special Interest Groups	Software Development	<a href="#">Contributing</a>
Python Events		Python Books	Python Logo		Developer's Guide
User Group Events		Python Essays	Python Wiki		Issue Tracker
Python Events Archive			Merchandise		python-dev list
User Group Events Archive			Community Awards		Core Mentorship
Submit an Event			Code of Conduct		Report a Security Issue
			Get Involved		
			Shared Stories		

## What is Python? Executive Summary

### What is Python? Executive Summary

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

## Official Speech of Python

# SWCON's Choice : Python

Often, programmers fall in love with Python because of the increased productivity it provides. Since there is no compilation step, the edit-test-debug cycle is incredibly fast. Debugging Python programs is easy: a bug or bad input will never cause a segmentation fault. Instead, when the interpreter discovers an error, it raises an exception. When the program doesn't catch the exception, the interpreter prints a stack trace. A source level debugger allows inspection of local and global variables, evaluation of arbitrary expressions, setting breakpoints, stepping through the code a line at a time, and so on. The debugger is written in Python itself, testifying to Python's introspective power. On the other hand, often the quickest way to debug a program is to add a few print statements to the source: the fast edit-test-debug cycle makes this simple approach very effective.

See also some [comparisons](#) between Python and other languages.

## Official Speech of Python

## Comparing Python to Other Languages

### Comparing Python to Other Languages

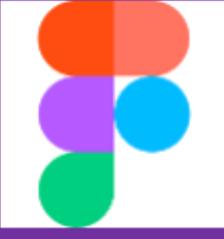
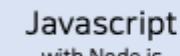
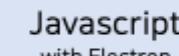
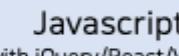
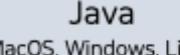
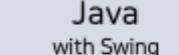
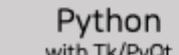
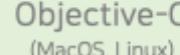
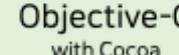
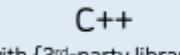
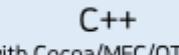
---

***Disclaimer:** This essay was written sometime in 1997. It shows its age. It is retained here merely as a historical artifact. --Guido van Rossum*

#### C++

Almost everything said for Java also applies for C++, just more so: where Python code is typically 3-5 times shorter than equivalent Java code, it is often 5-10 times shorter than equivalent C++ code! Anecdotal evidence suggests that one Python programmer can finish in two months what two C++ programmers can't complete in a year. Python shines as a glue language, used to combine components written in C++.

## Official Speech of Python

UI/UX	Mobile Application (Front-End)	Server Application (HTTP based Back-End)	Desktop (with GUI)	Web
 <b>Adobe XD</b>  <b>Figma</b>	 <b>Swift</b> with Cocoa (iOS)	 <b>Swift</b> (MacOS, Linux)	 <b>Swift</b> with Cocoa (MacOS)	
	 <b>Dart</b> with Flutter (iOS, Android)	 <b>Dart</b> with Flutter (MacOS, Windows, Linux)	 <b>Dart</b> with Flutter (MacOS, Windows, Linux)	 <b>Dart</b> with Flutter
	 <b>Rust</b> (MacOS, Windows, Linux)		 <b>Rust</b> with {plugins}	
	 <b>Go</b> with gomobile (iOS, Android)	 <b>Go</b> (MacOS, Windows, Linux)	 <b>Go</b> with {plugins} {Case by Case}	
	 <b>C#</b> with Unity/Xamarin (iOS, Android)	 <b>C#</b> with .Net (Windows)	 <b>C#</b> with Unity/Xamarin/.Net (MacOS, Windows)	
	 <b>Javascript</b> with jQuery/React/Cordova/Ionic (iOS, Android)	 <b>Javascript</b> with Node.js (MacOS, Windows, Linux)	 <b>Javascript</b> with Electron (MacOS, Windows, Linux)	 <b>Javascript</b> with jQuery/React/Vue.js
	 <b>Java</b> with Android SDK (Android)	 <b>Java</b> (MacOS, Windows, Linux)	 <b>Java</b> with Swing (MacOS, Windows, Linux)	
	 <b>Python</b> with kivy (iOS, Android)	 <b>Python</b> (MacOS, Windows, Linux)	 <b>Python</b> with Tk/PyQt (MacOS, Windows, Linux)	
	 <b>Objective-C</b> with Cocoa (iOS)	 <b>Objective-C</b> (MacOS, Linux)	 <b>Objective-C</b> with Cocoa (MacOS)	
	 <b>C++</b> with MS Cross-Platform (iOS, Android)	 <b>C++</b> with {3rd-party libraries} (MacOS, Windows, Linux)	 <b>C++</b> with Cocoa/MFC/QT/GTK (MacOS, Windows, Linux)	

# IOS dependent development platforms

---

- Microsoft Visual Studio C++

<https://www.visualstudio.com/ko/vs/cplusplus/>

- Apple Xcode

<https://developer.apple.com/kr/xcode/>

# IOS independent development platforms

---

- Microsoft Visual Studio Code

<https://code.visualstudio.com/>

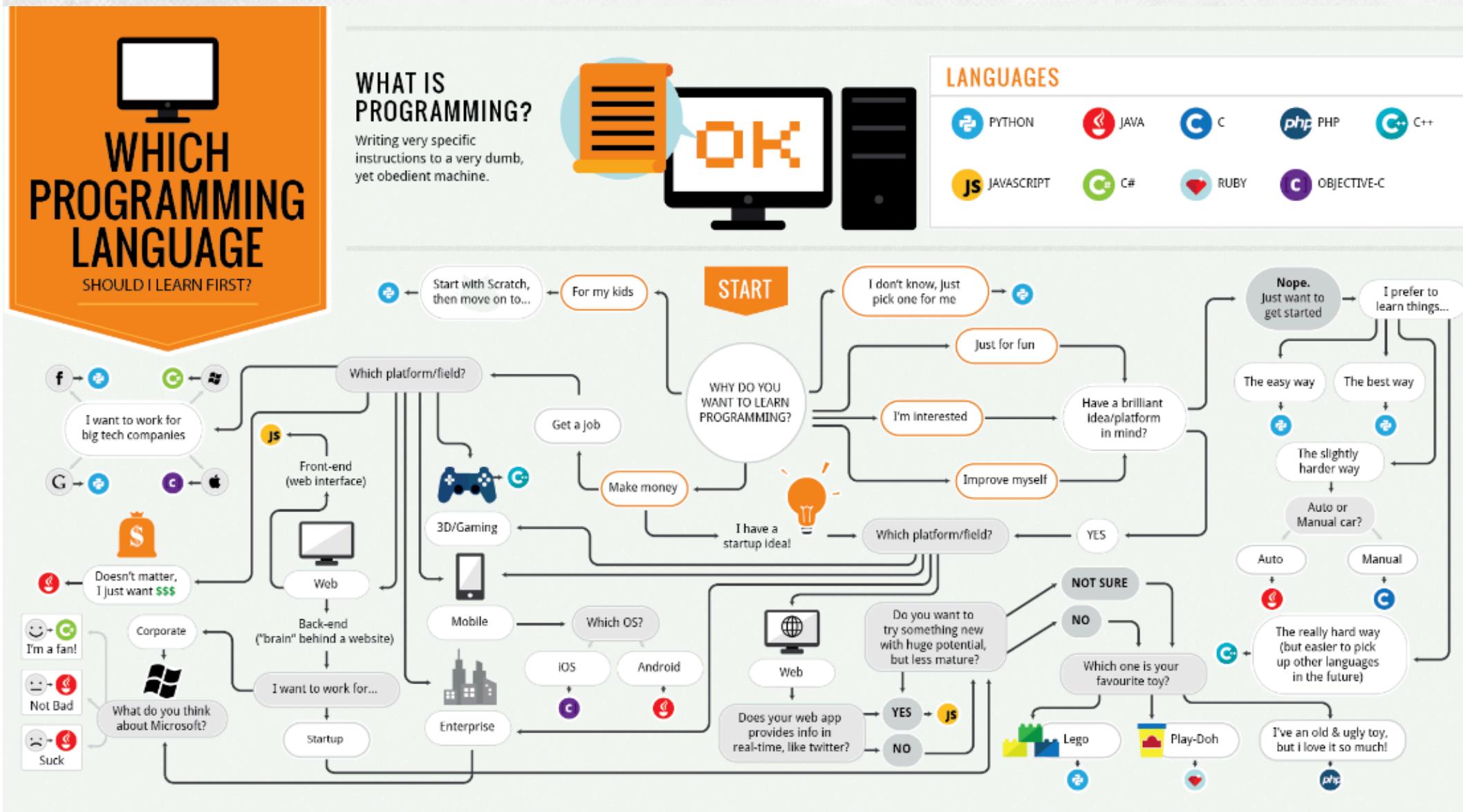
- Atom

<https://atom.io/>

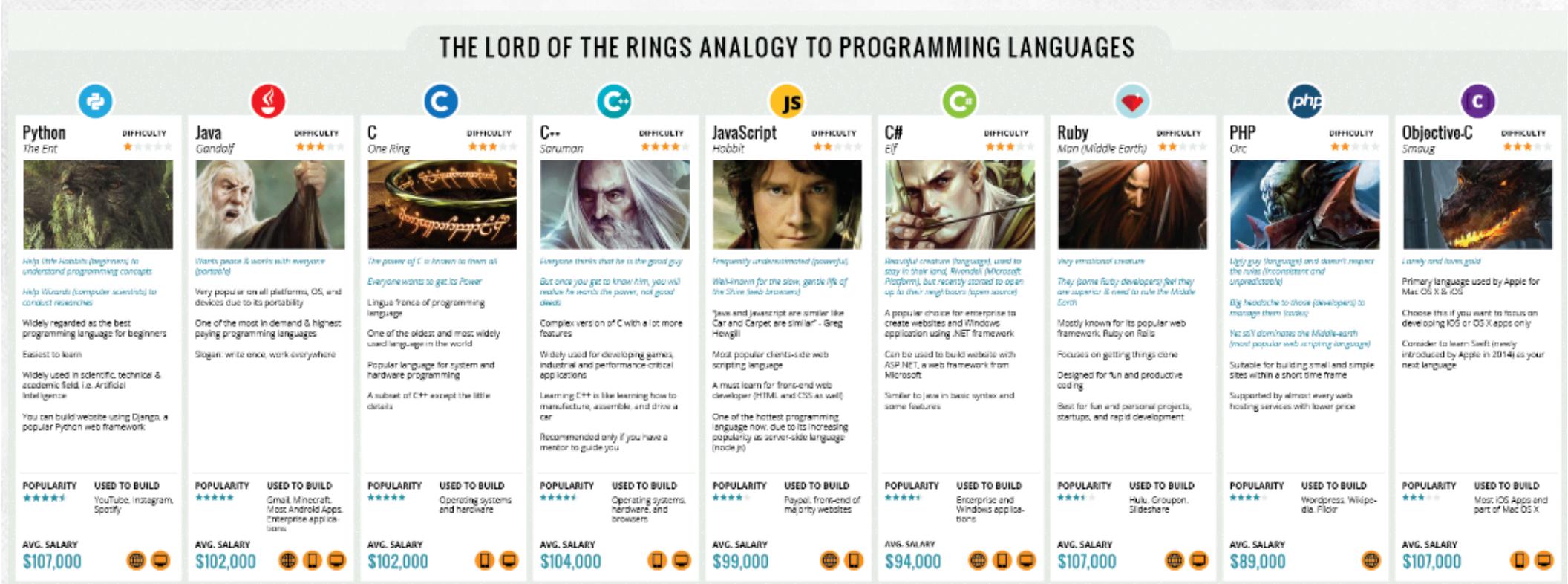
- Eclipse

<https://www.eclipse.org/>

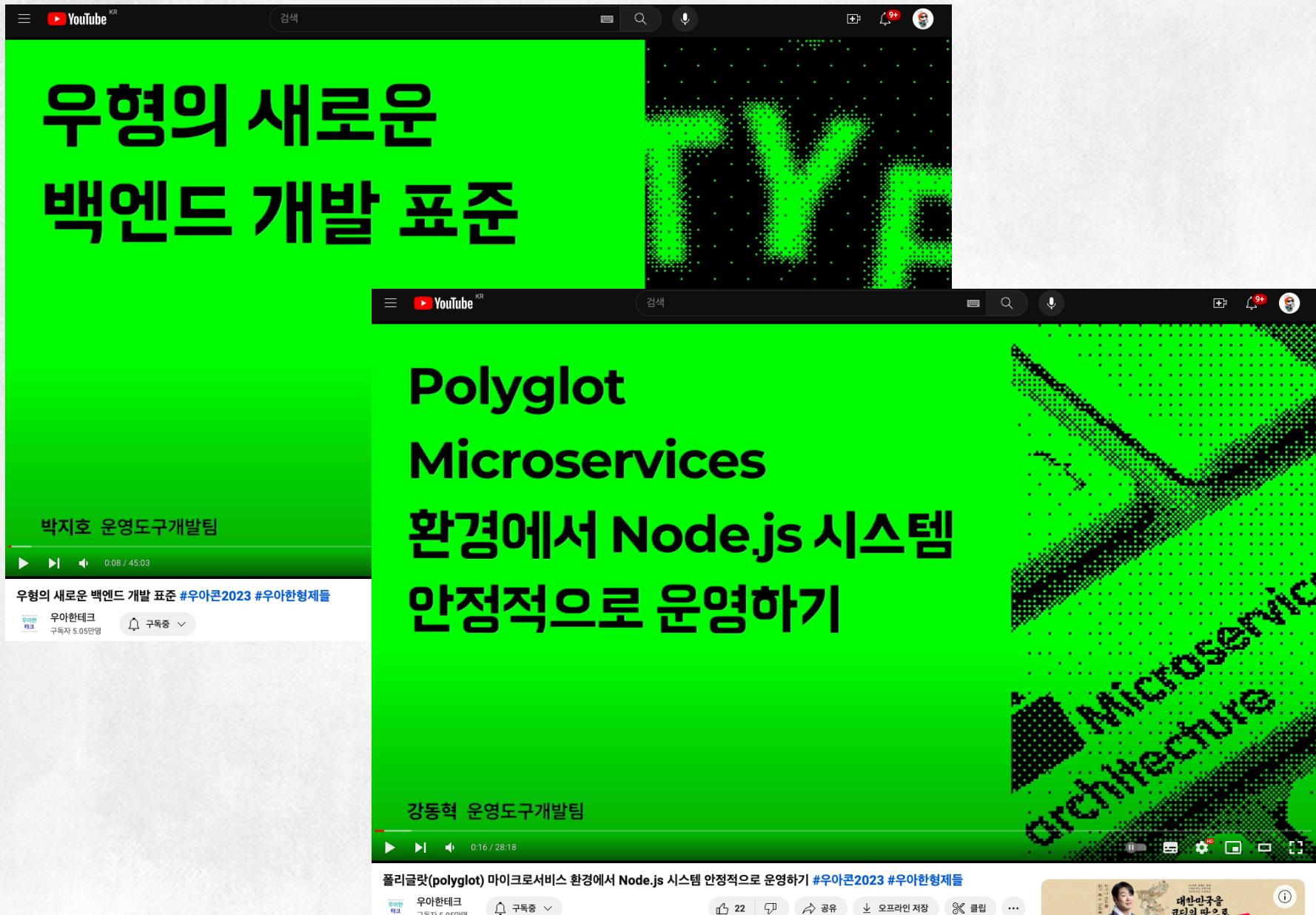
# Others



# Others



# [REF] 우아한 형제들 (서버) : JavaScript



# [REF] 요기요 (서버) : Python

[10년차 시리즈] 개발언어 때문에 탄생한 요기요 탄생 비하인드  
2021-01-15

전화 주문이 배달앱 주문으로 변화하기까지, 지난 10년간 음식 배달 시장은 역동적으로 변화하며 한국인의 일상을 바꾸어 놓았습니다. 국내 대표 배달앱 요기요 역시 급변하는 배달 시장의 변화에 맞춰 서비스부터 앱의 디자인까지 다양한 면에서 진화를 거듭하고 있습니다. 올해 서비스 론칭 10년 차를 맞아 지난 요기요의 역사를 알아봅니다.

**요기 [명사]**: 시장기를 겨우 면할 정도로 조금 먹음.  
**여기요 [감탄사]**: 주문 따위를 하기 위해 종업원을 부를 때 쓰는 말.

배고픔을 면하는 말인 '요기'와 레스토랑에서 음식을 주문할 때 사용하는 흔한 말인 '여기요'를 합쳐 배고픔을 느끼는 고객들이 가장 먼저 떠올릴 수 있는 서비스가 되기 위해 탄생된 '요기요'.

'요기요'는 지난 2012년부터 무려 10년 동안 국내 소비자들에게 맛있는 일상의 즐거움을 선사하며 국내 대표 배달앱으로 성장했다. 하지만 '요기요'가 탄생하기까지 그 과정은 결코 순탄치만은 않았다는데, '요기요' 탄생에 대한 비하인드 스토리를 요기.서.부터 알아보자.

■ '요기요'의 탄생 미션! "앱 내에서 주문부터 결제까지 가능케하라"

'요기요'는 실수로 탄생한 세계적 빅뱅품처럼 원래는 전혀 탄생할 계획이 없었던 브랜드이자 서비스였다. 원래 계획대로라면 딜리버리히어로는 2010년 4월에 론칭한 대한민국 배달앱의 원조 '배달통'을 시작으로 국내 서비스를 시작할 계획이었다. 하지만 요기요 탄생의 비화가 있었으니 바로 '파이썬(Python)'이란 개발 언어 때문이다.

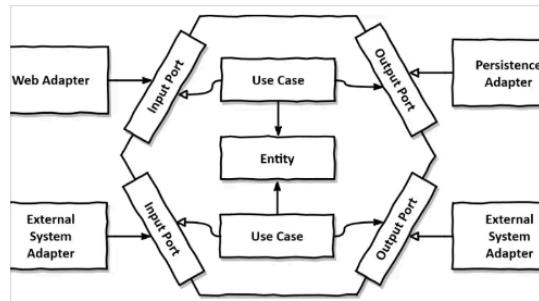
당시 주문 중개 플랫폼의 성패는 결제방법에 달려있었다. 어떻게 결제를 손쉽게 하느냐가 배달앱을 막 이용하기 시작하는 소비자들의 주문습관을 결정짓는 핵심 열쇠였기 때문이다. 경쟁사보다 날빠르게 온라인 결제 기능을 구현하기 위해 독일 딜리버리히어로가 가진 간편한 결제 기술을 적용하여 '요기요'는 그렇게 대한민국 배달앱 시장에 태어나게 됐다. 그렇게 '요기요'는 기존 배달앱들처럼 전단지를 한 번에 모아 볼수 있도록 한 기능을 넘어 앱을 통해 주변 맛집 정보를 알고 먹고 싶은 메뉴 선택 후 결제하기 버튼만 누르면 바로 주문이 되는 편리한 주문환경을 만들기 시작해 소비자들의 마음을 사로잡았다.

지금은 모바일 결제가 누구에게나 패 익숙해졌지만, 당시만 하더라도 '요기요'가 업계 최초로 선보인 '요기서 1초결제'는 큰 센세이션을 일으켰다. 이후 기술 공유를 통해 프로그램의 안정화가 이뤄지면서 '요기요'의 기능들이 '배달통'에도 차츰 적용되기 시작했다. 두 플랫폼은 그렇게 사이좋게 발맞춰 다양한 기능을 구현해 내기 시작했고, 소비자들의 큰 사랑을 받으며 배달앱 시장의 성장을 견인하는 주요 플레이어이자 서비스로 자리잡았다.

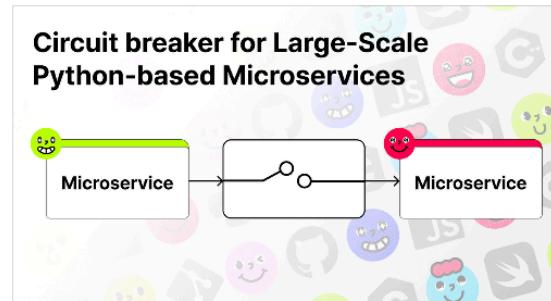
그동안 왜 딜리버리히어로 코리아(구. 알지피코리아)는 여러 개의 앱을 론칭해 운영할까 의문을 품는 사람들이 많았다. 같은 듯 다른 서비스이자 매력을 가진 두 앱이 오랫동안 함께하고 있는 이유는 바로 이런 탄생의 비화가 있었기 때문이다. 앞으로도 요기요와 배달통이 우리에게 선사할 맛있는 즐거움의 시간을 기대해달라.

PYTHON POSTS CULTURE INTERVIEW CAREER

## 요기요 X 파이썬



광고 시스템 퍼포먼스 튜닝 회고록  
신규 광고 시스템 운영 과정에서 발생한 퍼포먼스 이슈와 그에 대한 해결 사례  
Youngsik Choo Jul 18, 2023 · 11 min read



파이썬 기반의 대규모 마이크로 서비스에서 Circuit Breaker 도입 예정  
Circuit breaker for Large-Scale Python-based Microservices  
Edward J. Yoon Mar 31, 2023 · 7 min read



2022 요기요 신입채용 '위대한 ROOKIES 5기' 모집  
10. 01일 ~ 10. 19(수) 17:00 까지  
모집부분 : Backend Developer  
국내 최대 파이썬 주력 요기요에서  
위대한 어첨을 시작하세요.

한 방! 온보딩 스크립트 만들기  
개발자가 입사 후 맨 처음 하는 일. 자동화 할 수 없을까?  
Seungwon Jeong



제목은 메리 크리스마스로 하겠습니다. 그런데 이제 프로그래밍을 결들이  
25일 간의 Advent of Code 참여 후기

Reference: <https://techblog.yogiyo.co.kr/yogiyo-python/home>  
<https://www.wesang.com/mobile/newsroom/?nid=161>  
<https://www.etnews.com/20220927000104>

# Summary

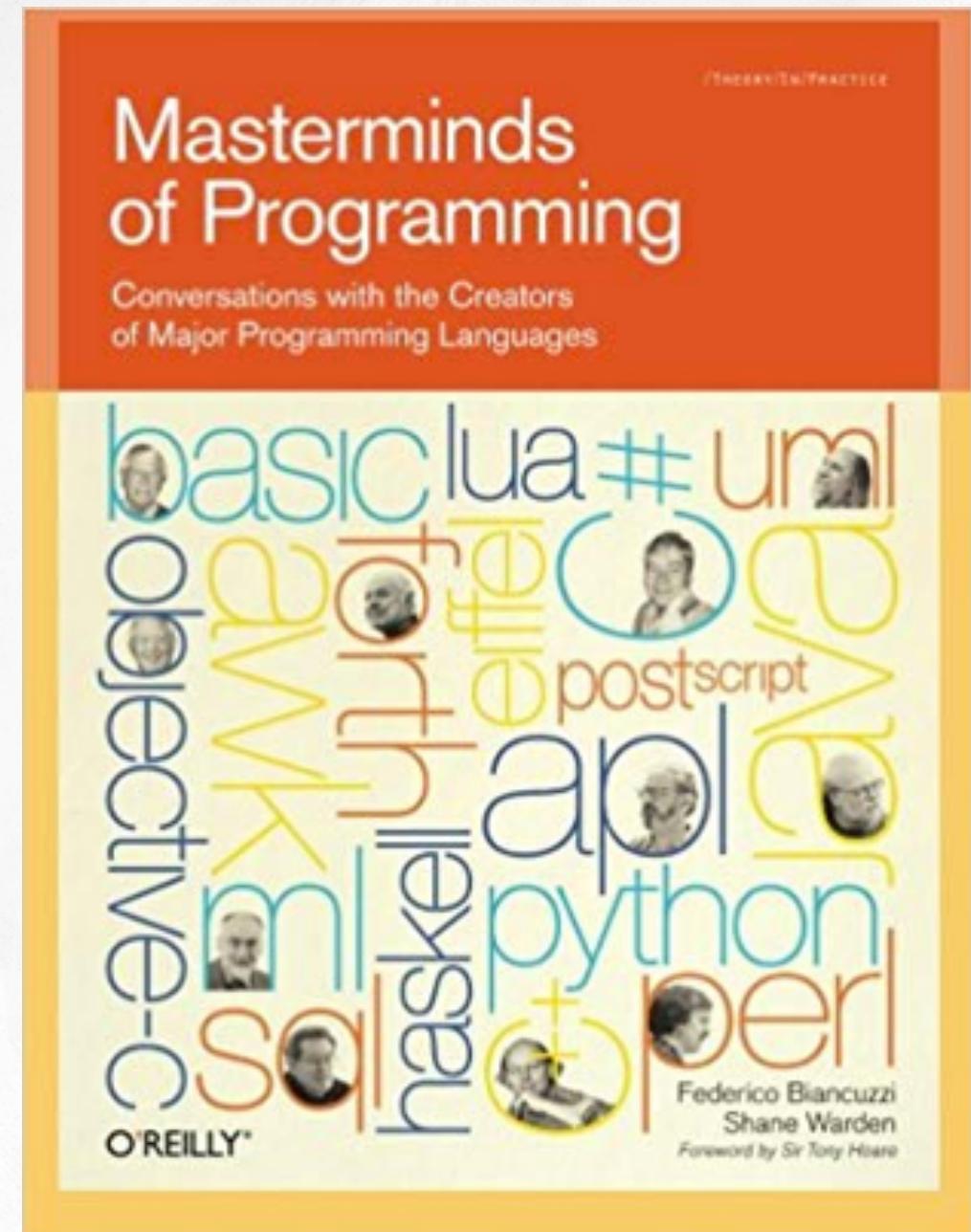
---

- Hardware dependent programming requests specific development language and environments (in general).
- Hardware independent programming strongly related to domain specific features of a language, libraries, and tools.
- Every programming language has authors, reason and recommended use cases.

# Reference

---

- Masterminds of Programming  
O'Reilly Media, 2009



# Homework

---

- Survey a specific programming language
  - Define your problem (domain)
  - Review widely used domain specific languages
  - Select an appropriate language and study:
    - Language author
    - When and Why author made the language
    - Pros and Cons of the language
    - Standard and/or Roadmap of the language
    - Famous open source softwares using the language



**Thank you**