twitch.tv/dancojocar
youtube.com/dancojocar

Mobile Applications

Fall 2023

Prerequisites

- Modern programming language
- Object oriented
- Statically types
- IDE - IntelliJ/Android Studio or Visual Studio Code

What should you know...

- Basics:
  - Object-oriented programming
  - Classes, methods
  - Exception handling

Bonus

- Functional Programming
- Lambdas
- Higher Order Functions
- Reactive Programming

Options

2007

2008

openwebOS

symbian OS

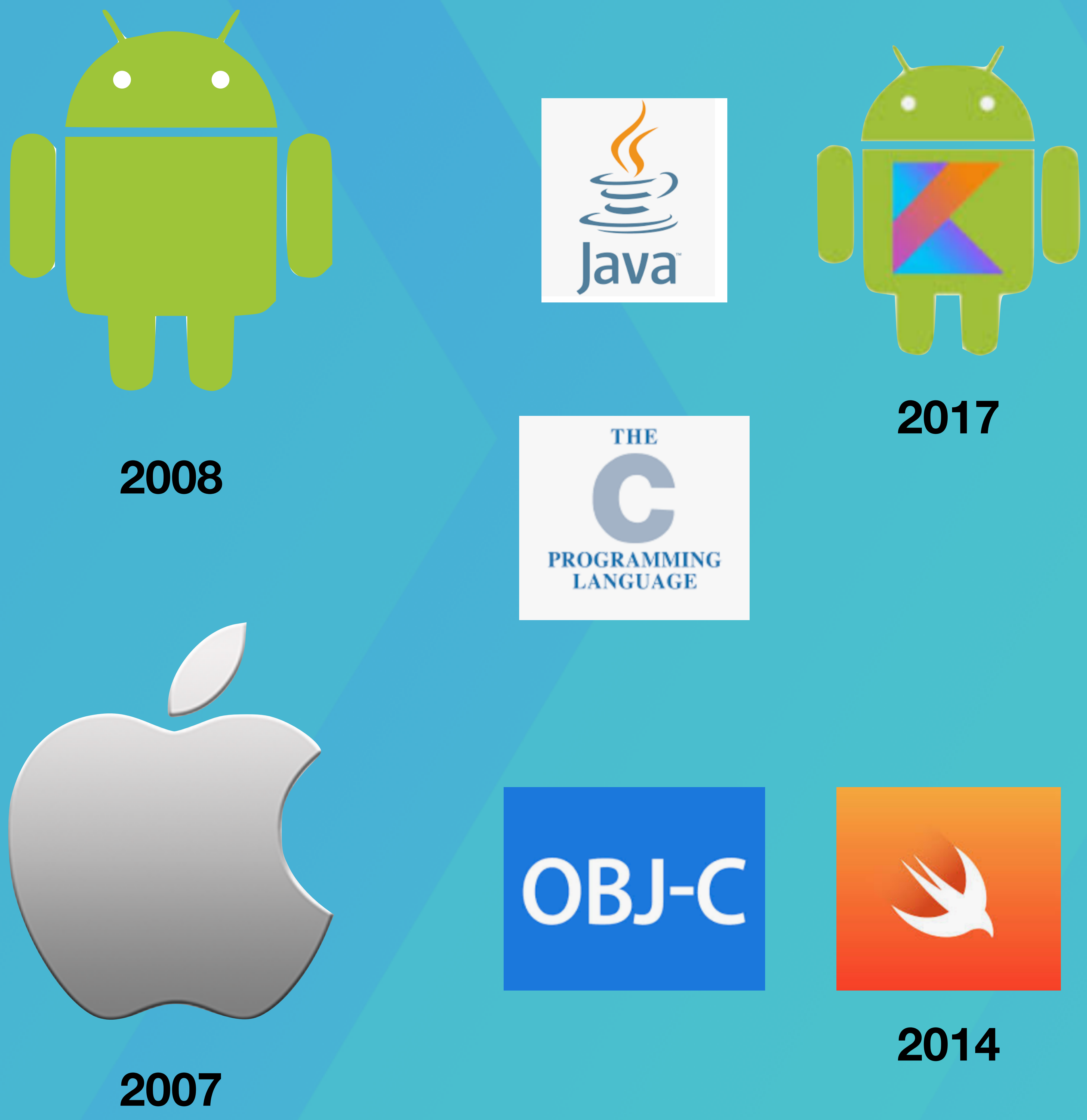
BlackBerry 10

Windows Phone

SAMSUNG bada

2010

Native Options



Non-Native Options

APACHE
CORDOVA™
HTML
CSS
JS
ionic

2013

Hybrid App

Non-Native Options

Compiled App

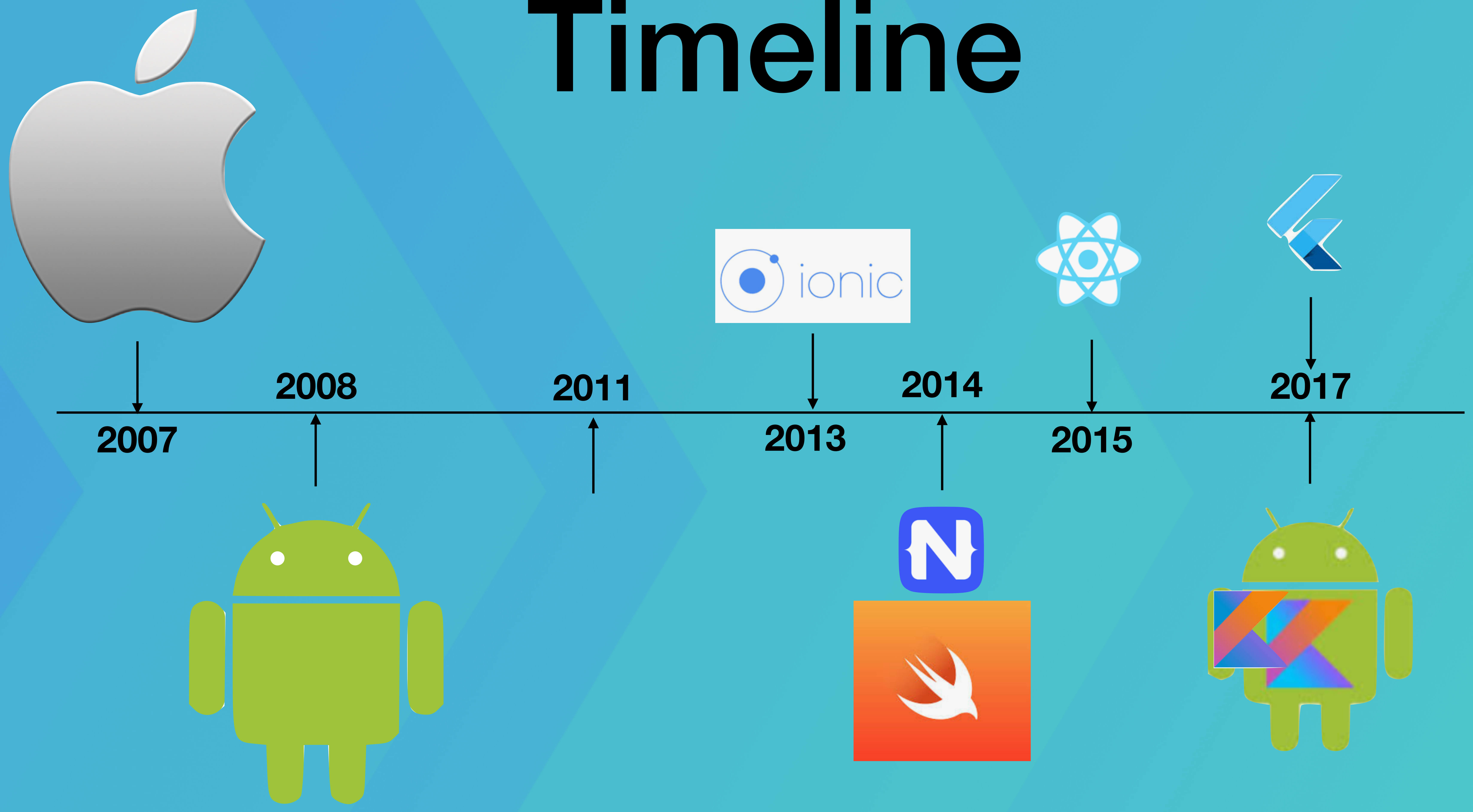
2014

2015

2017

Dart

Timeline



2007

2008

2011

2013

2014

2015

2017

What to learn?

ionic

OBJ-C

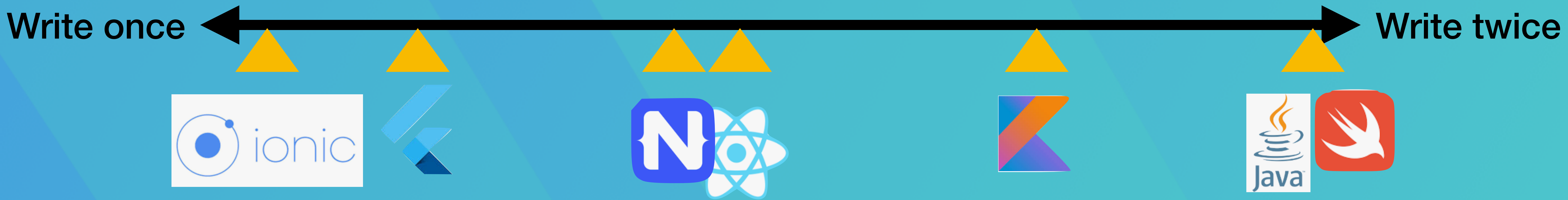
Java

KNOW YOUR RIGHTS

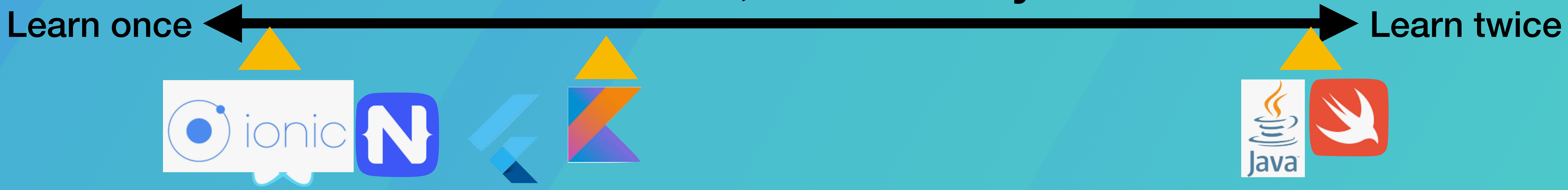
bit.ly/maQuiz2023

Comparison

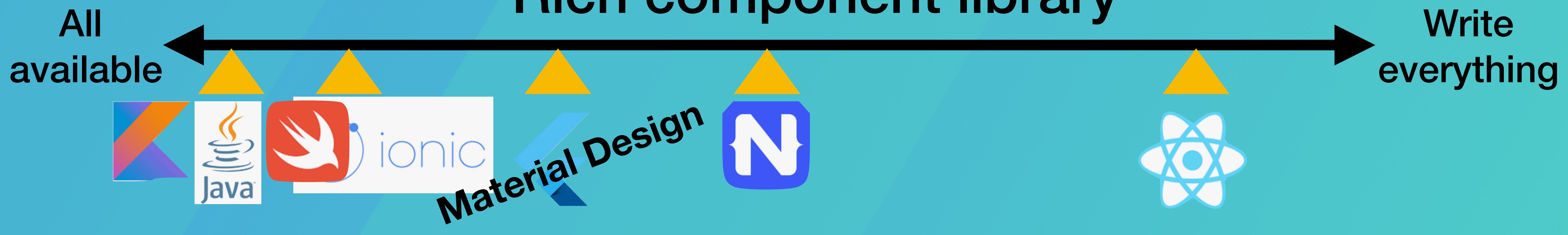
Write once, use everywhere



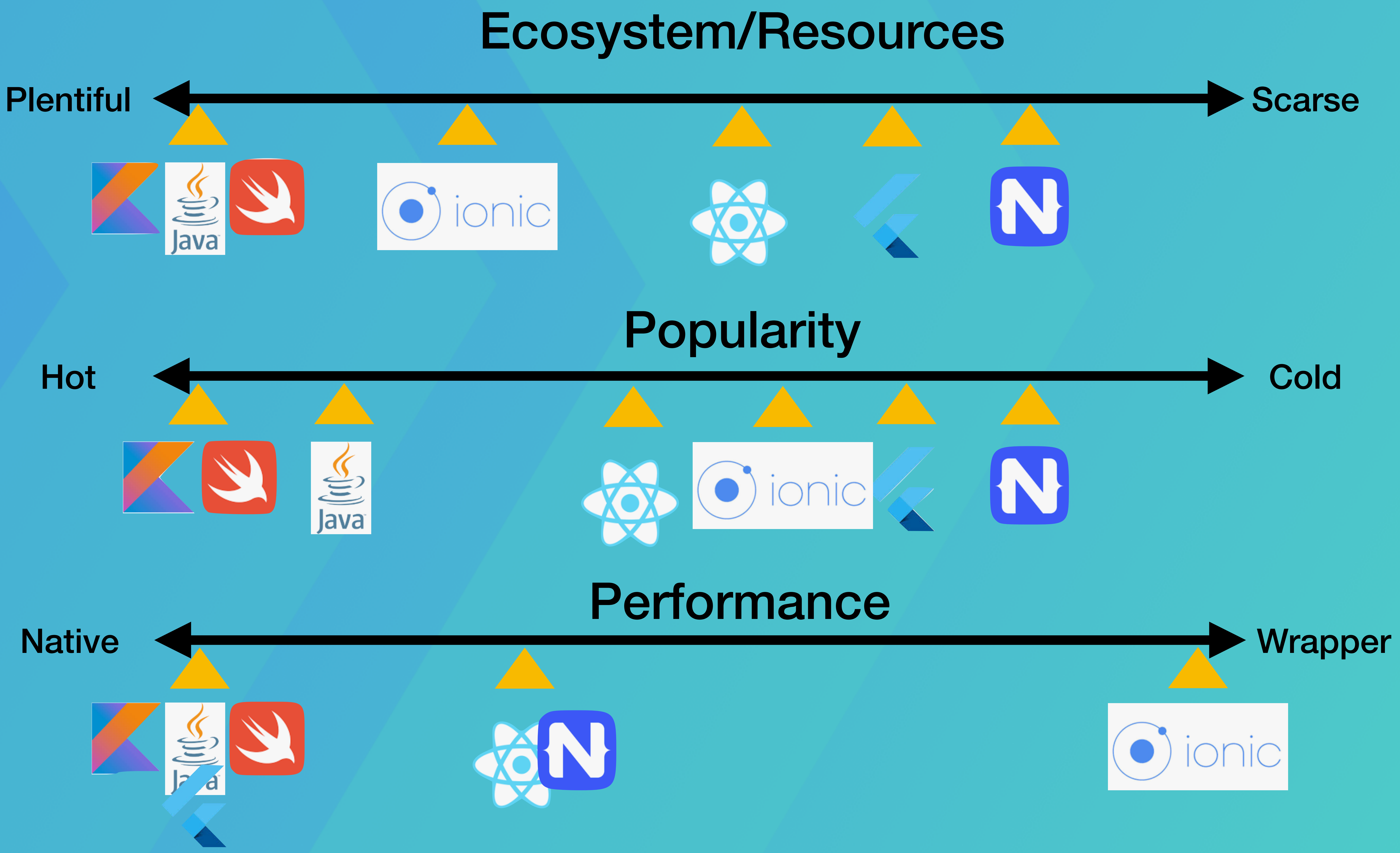
Learn once, write everywhere



Rich component library



Comparison



Ecosystem/Resources

Plentiful Scarse

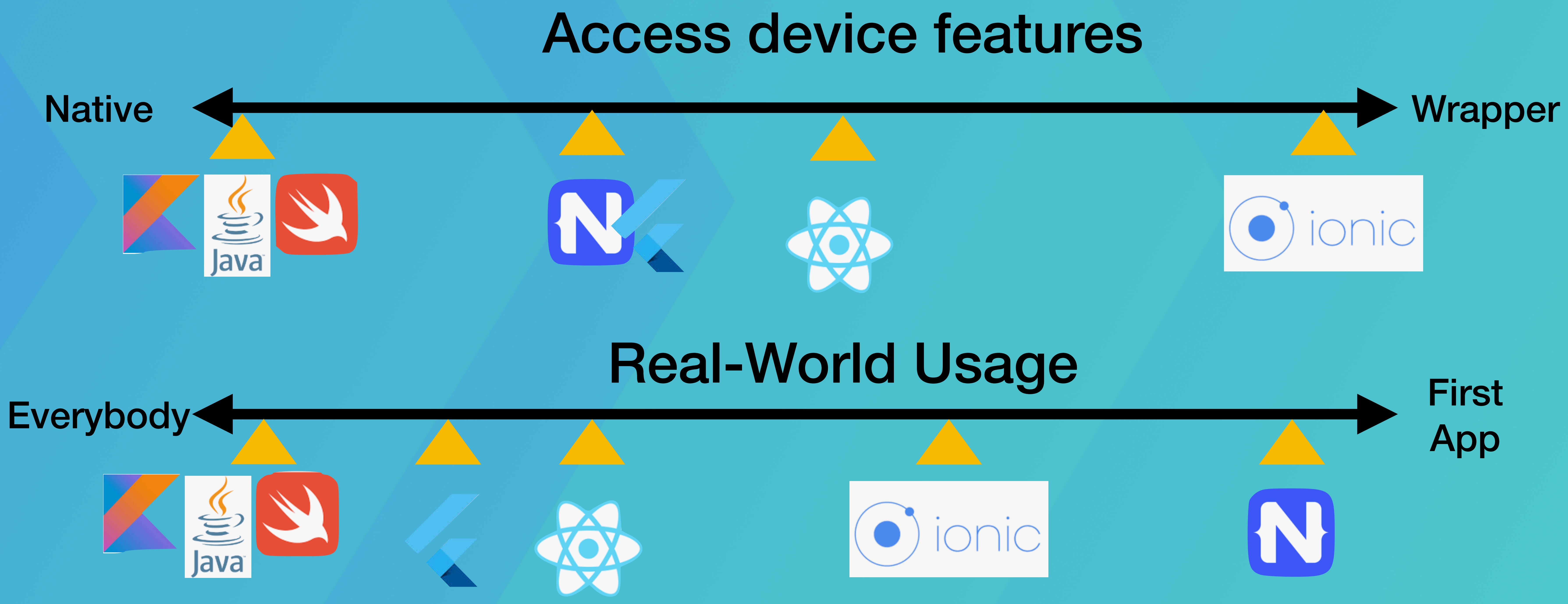
Popularity

Hot Cold

Performance

Native Wrapper

Comparison



Access device features

Native

Wrapper

Real-World Usage

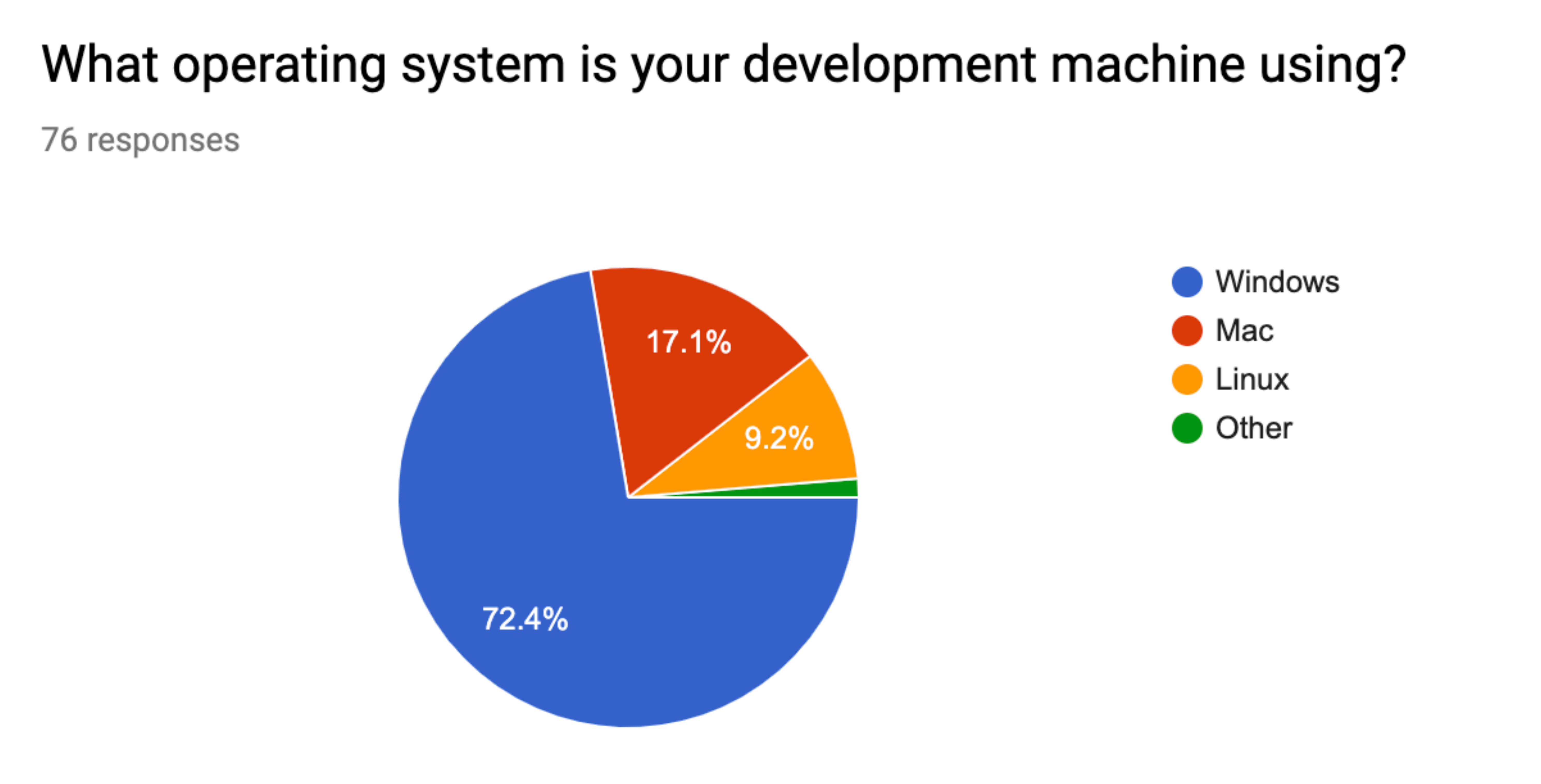
Everybody

First App

Previous years

ionic
OBJ-C
Java

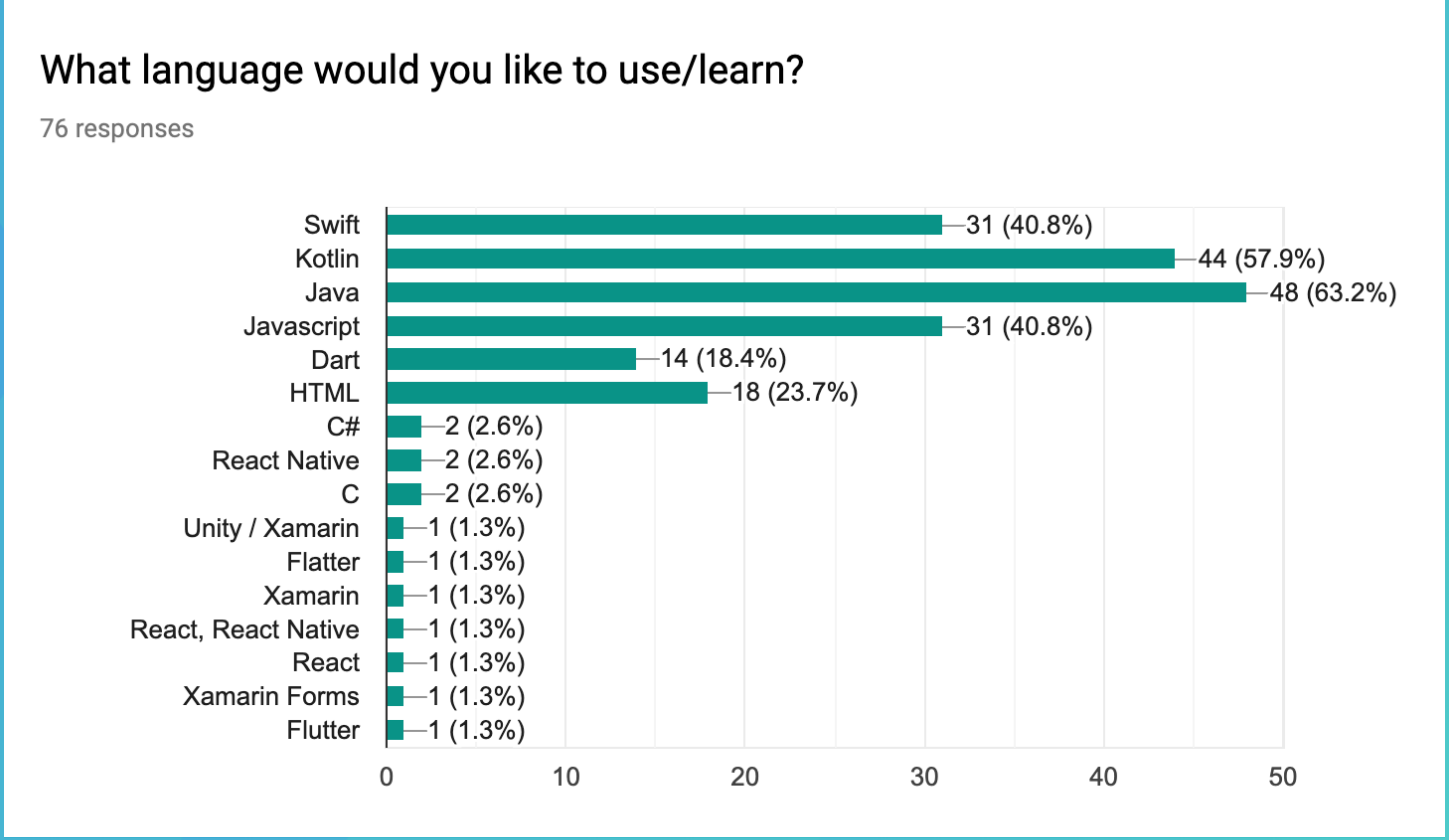
Responses from 2018



What operating system is your development machine using?

76 responses

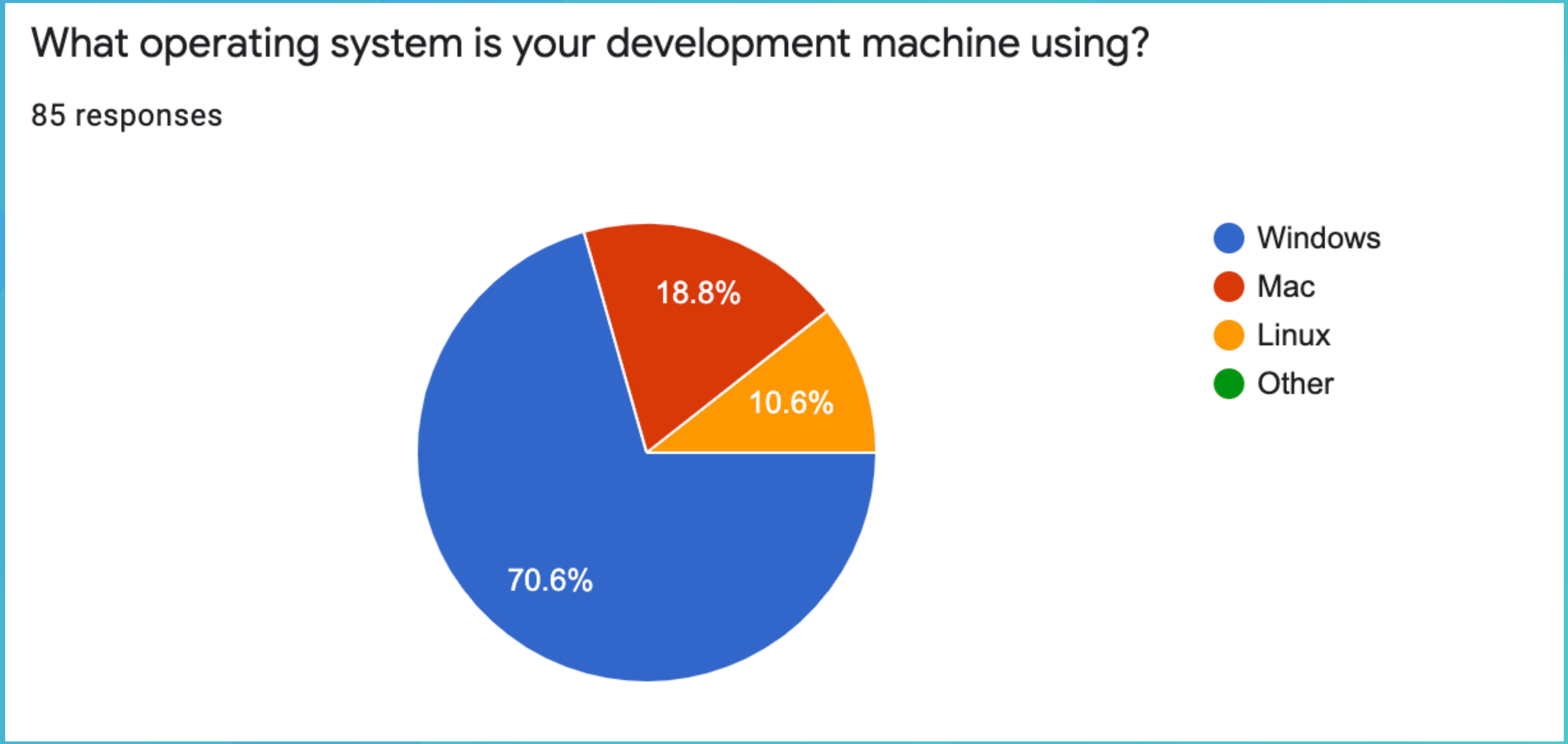
Responses from 2018



What language would you like to use/learn?

76 responses

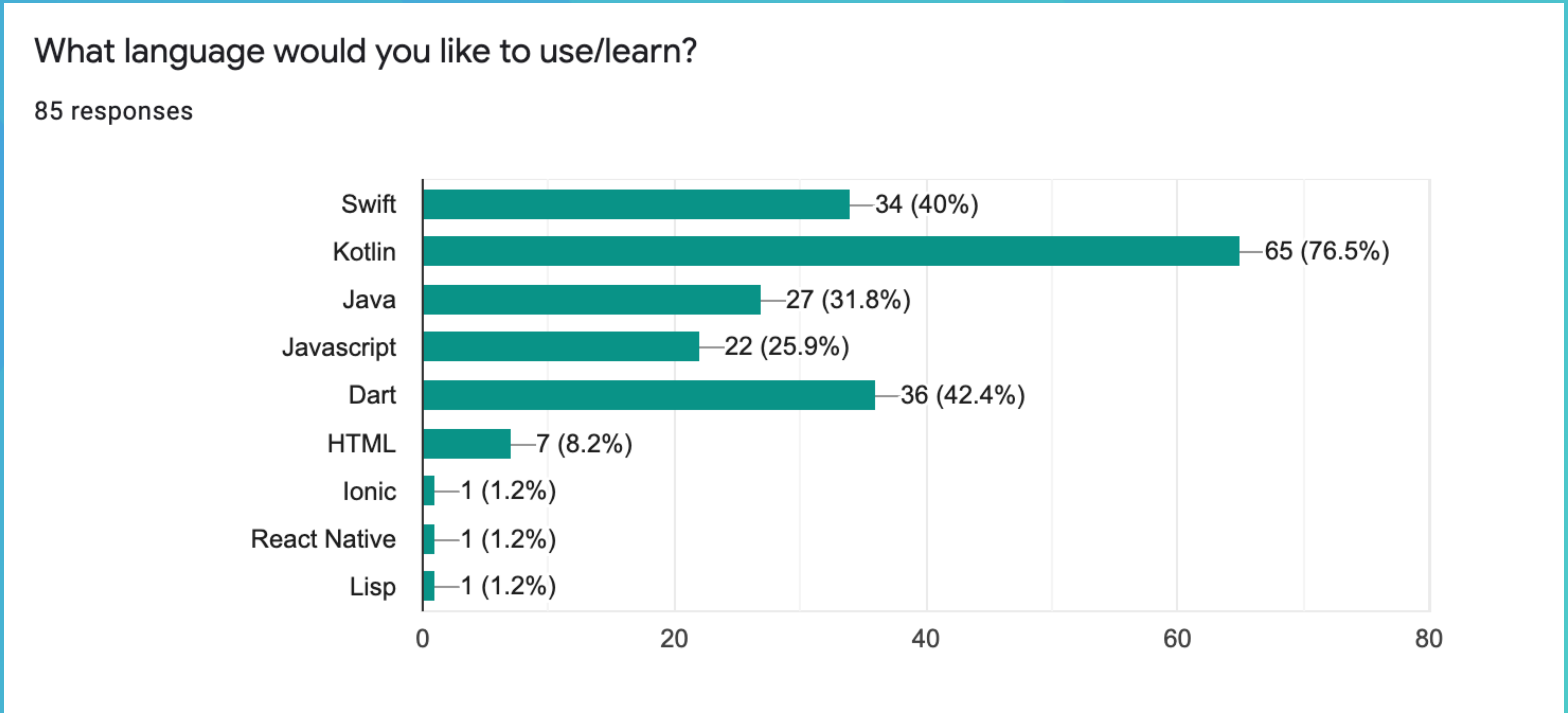
Responses from 2019



What operating system is your development machine using?

85 responses

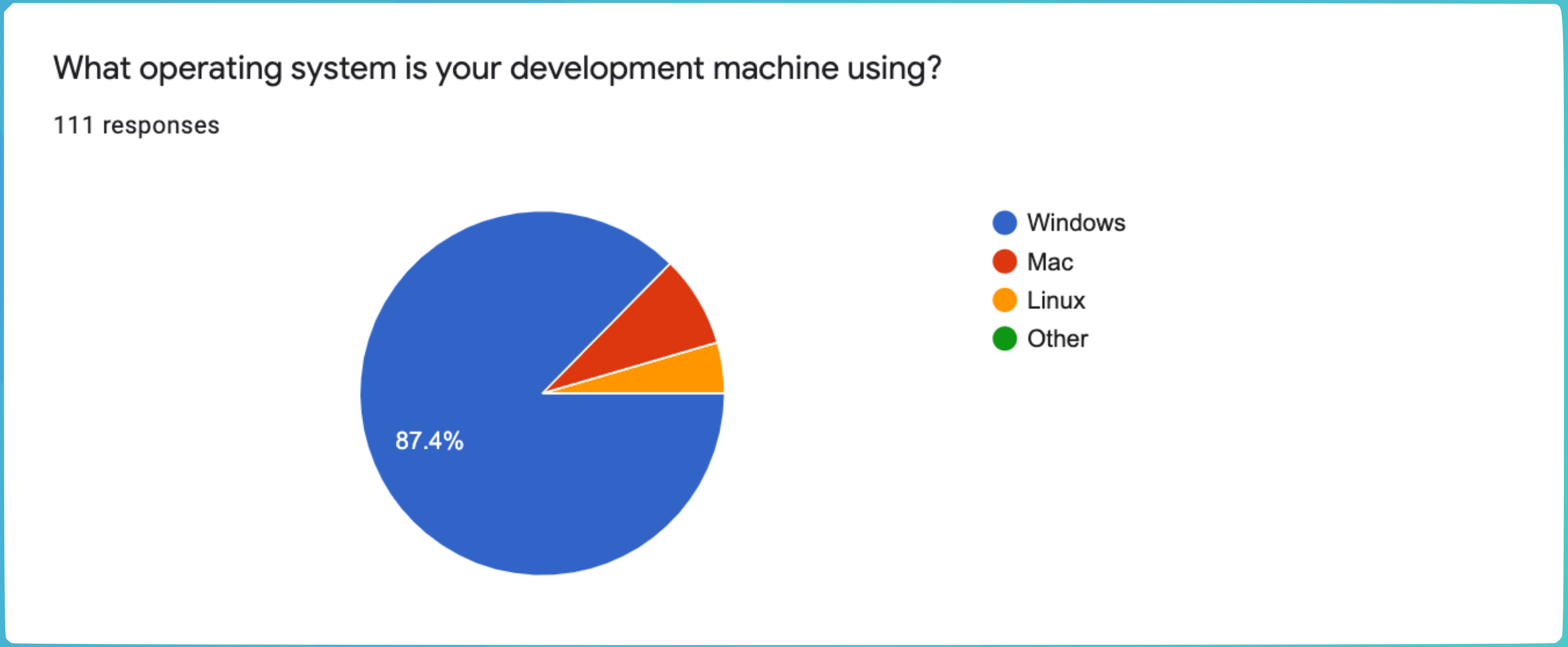
Responses from 2019



What language would you like to use/learn?

85 responses

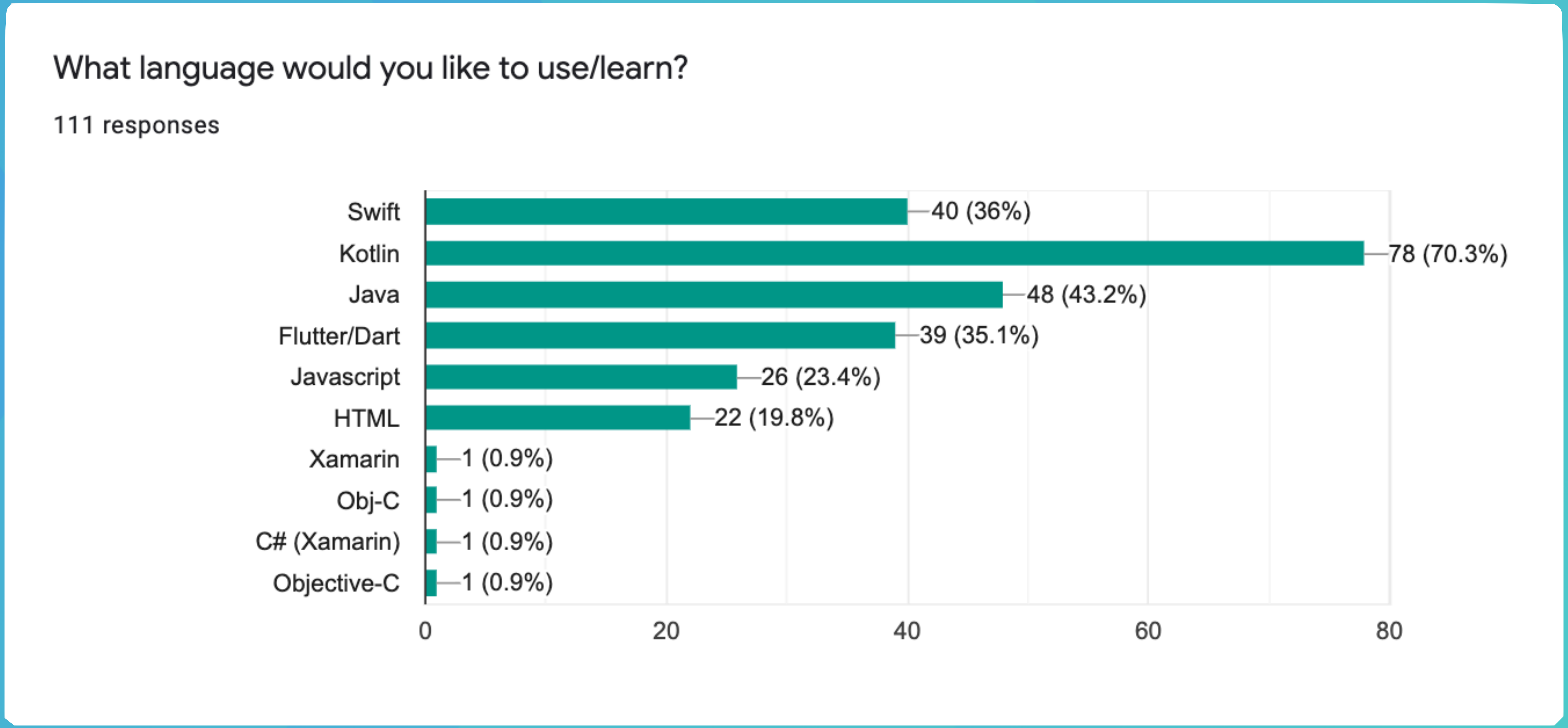
Responses from 2020



What operating system is your development machine using?

111 responses

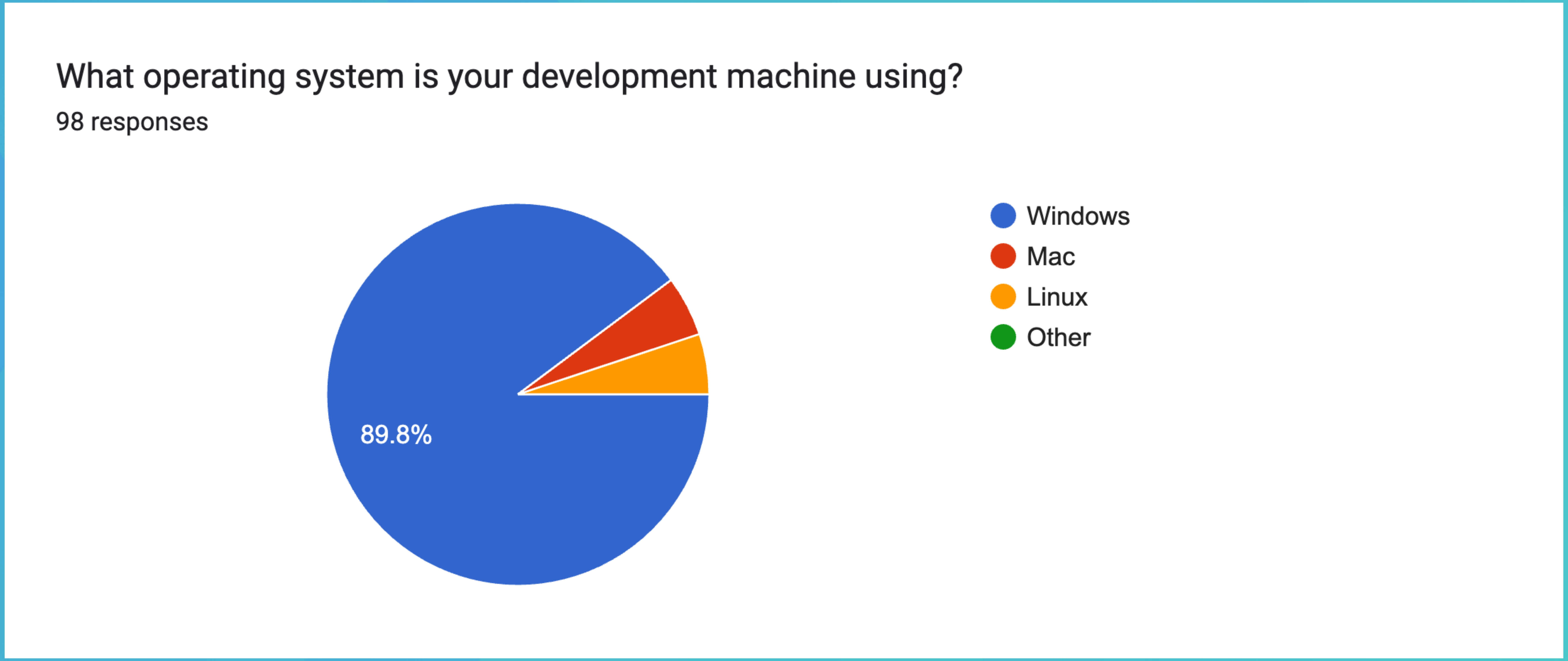
Responses from 2020



What language would you like to use/learn?

111 responses

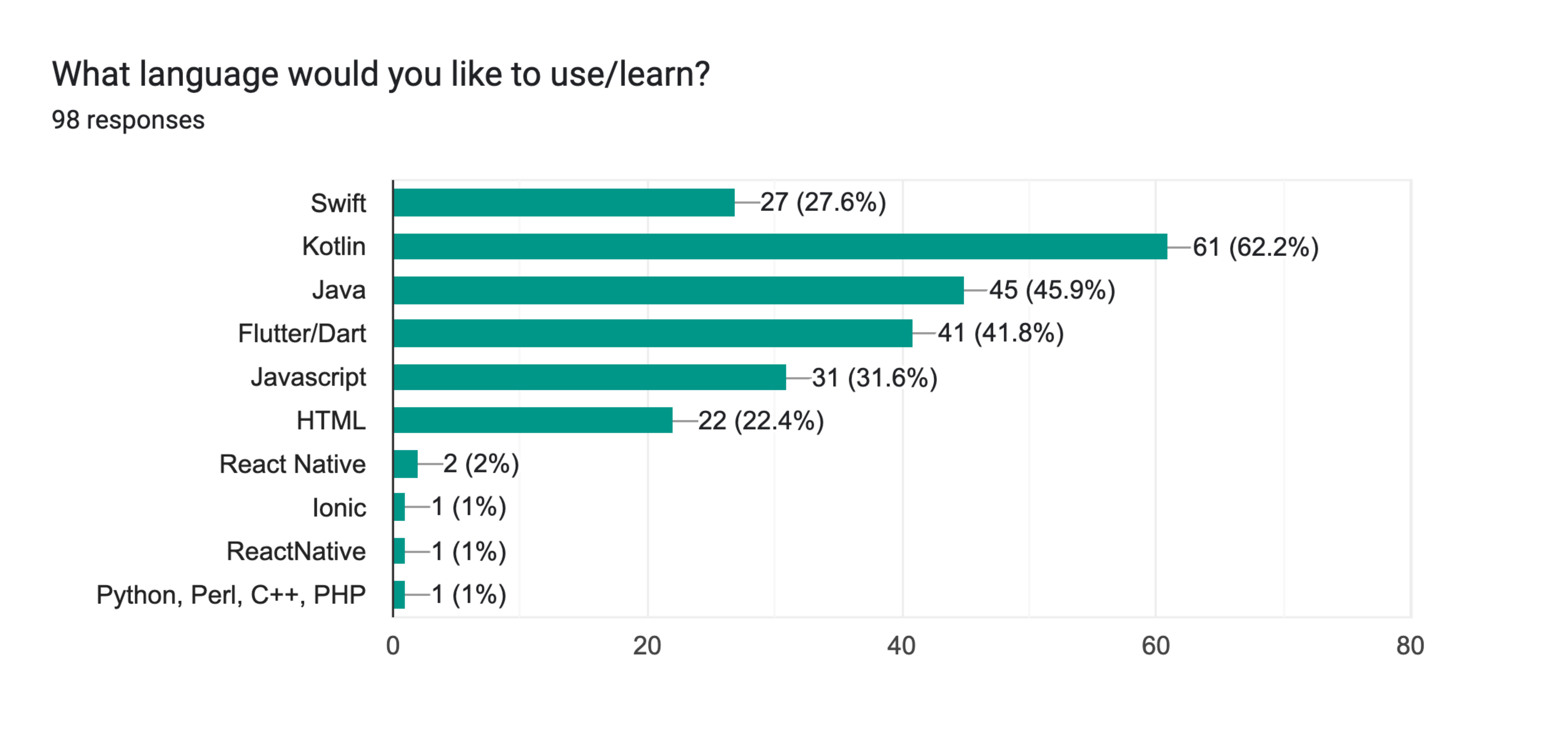
Responses from 2021



What operating system is your development machine using?

98 responses

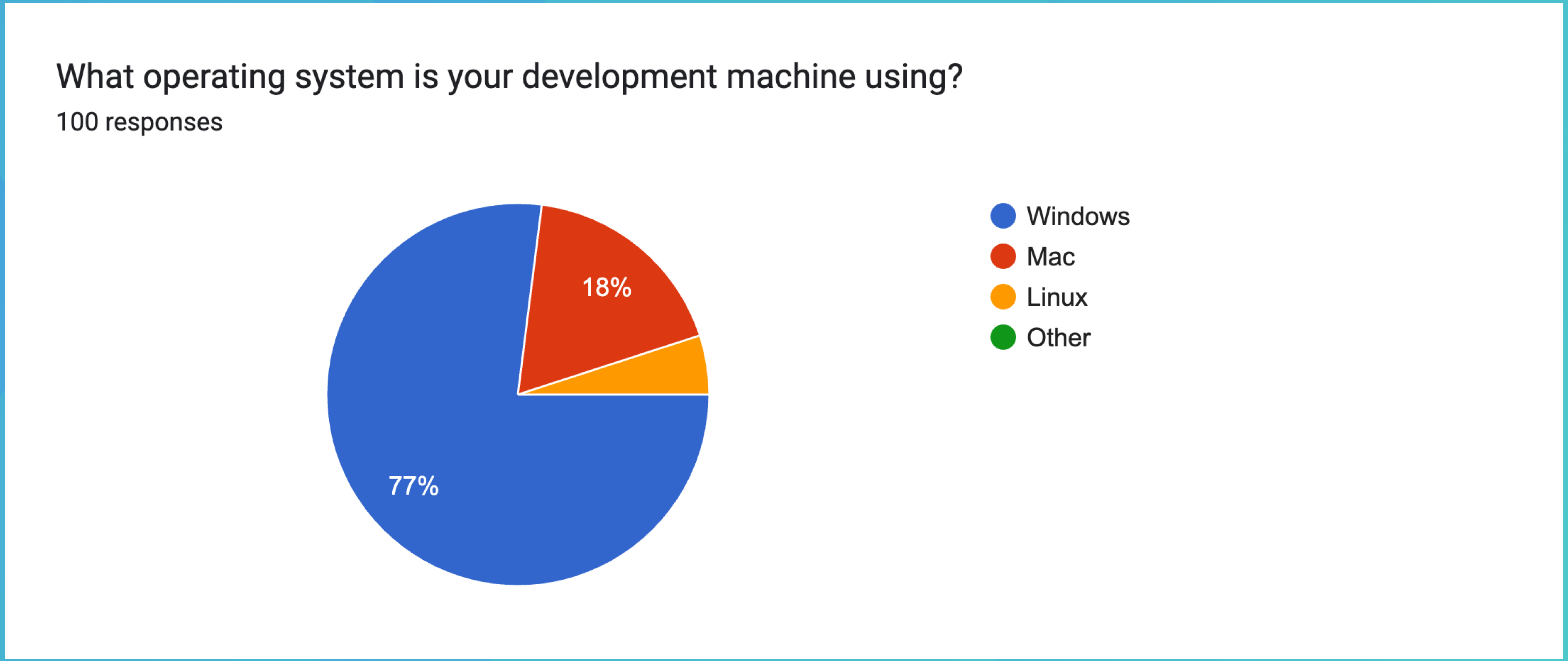
Responses from 2021



What language would you like to use/learn?

98 responses

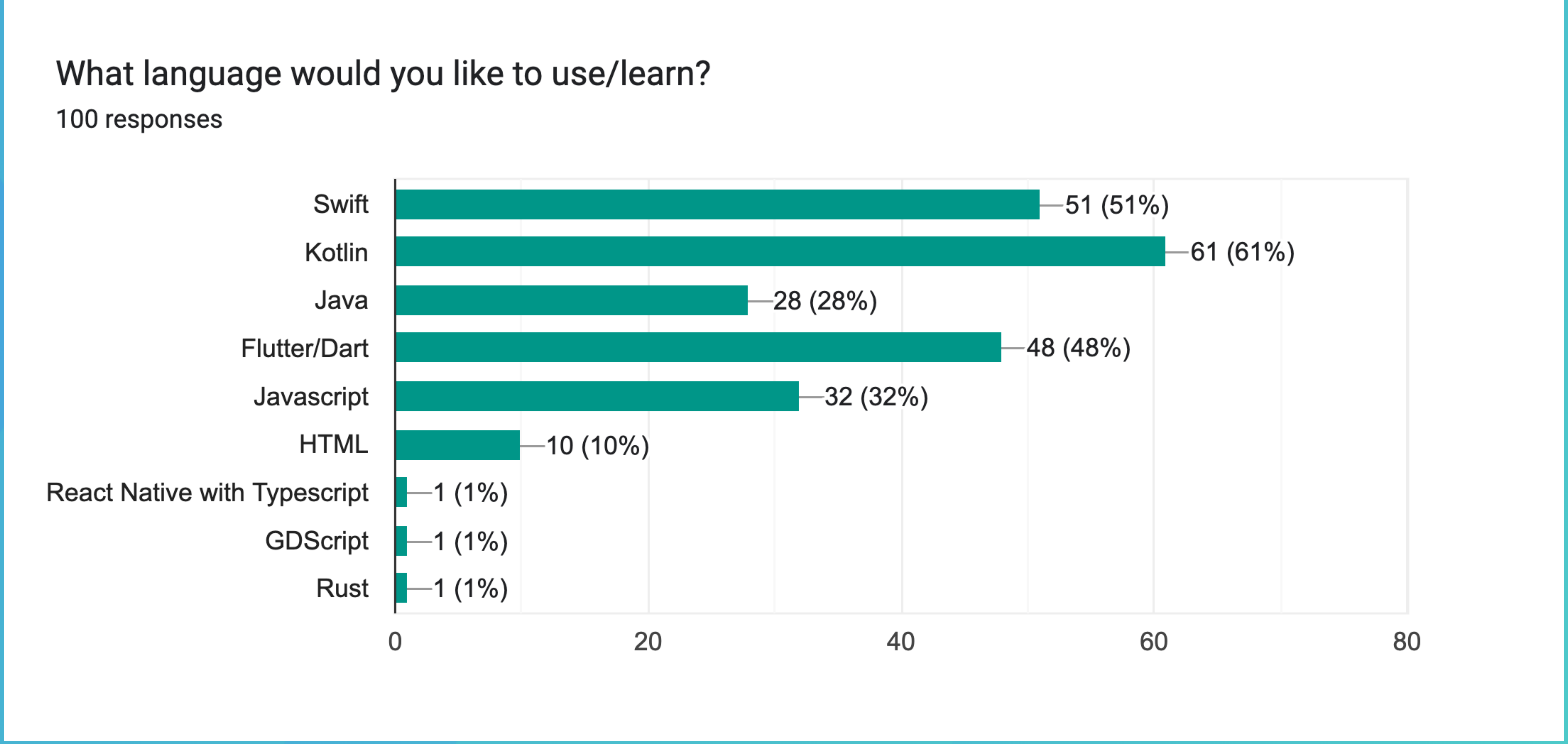
Responses from 2022



What operating system is your development machine using?

100 responses

Responses from 2022

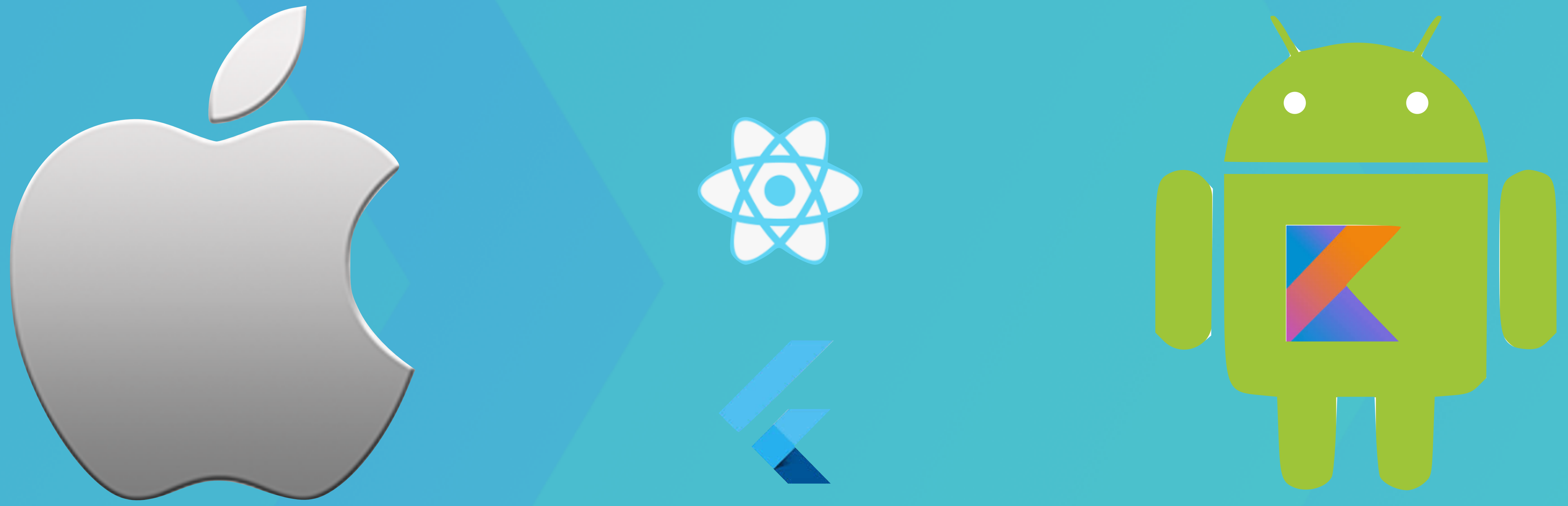


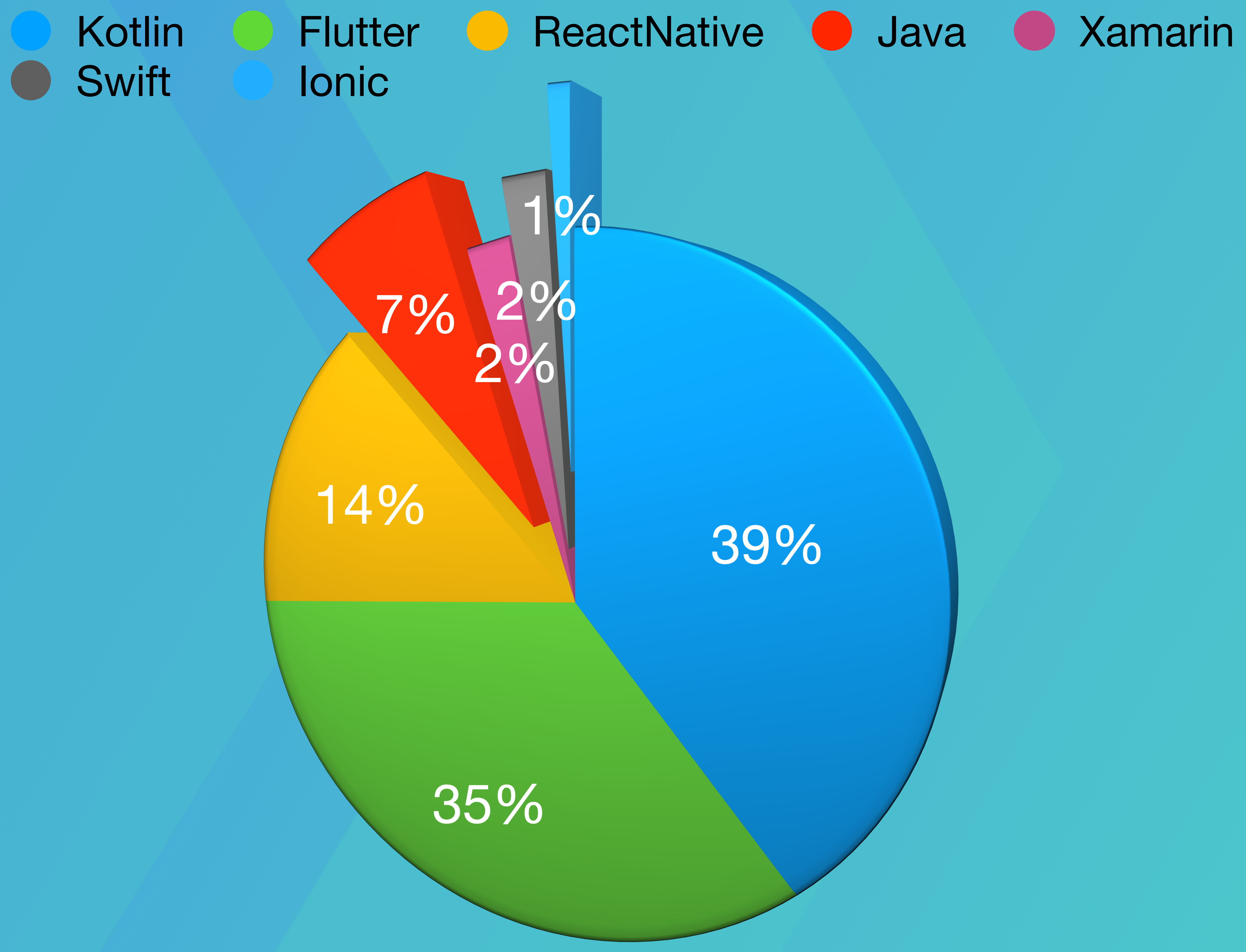
What language would you like to use/learn?

100 responses

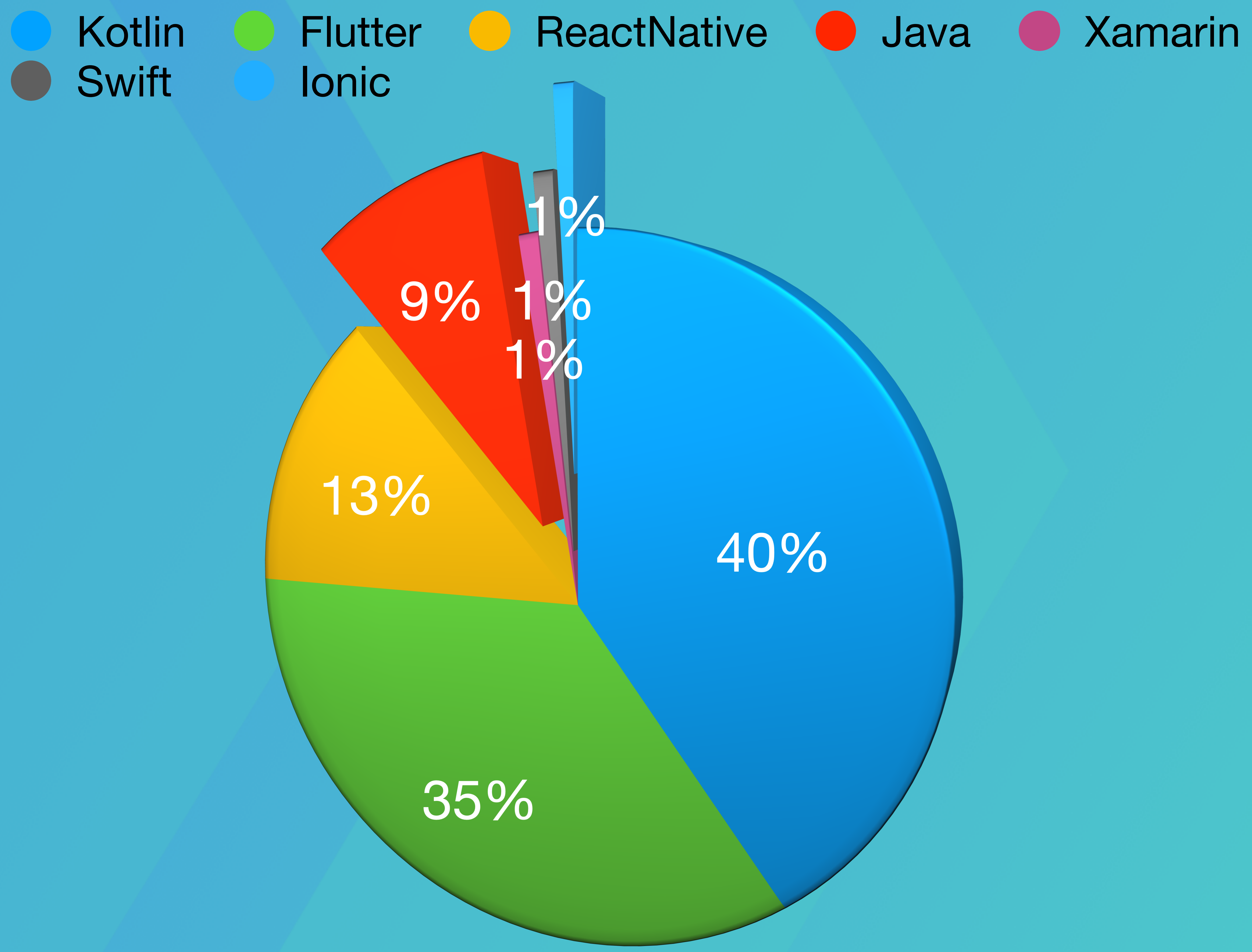
Check the quiz

Previous Year

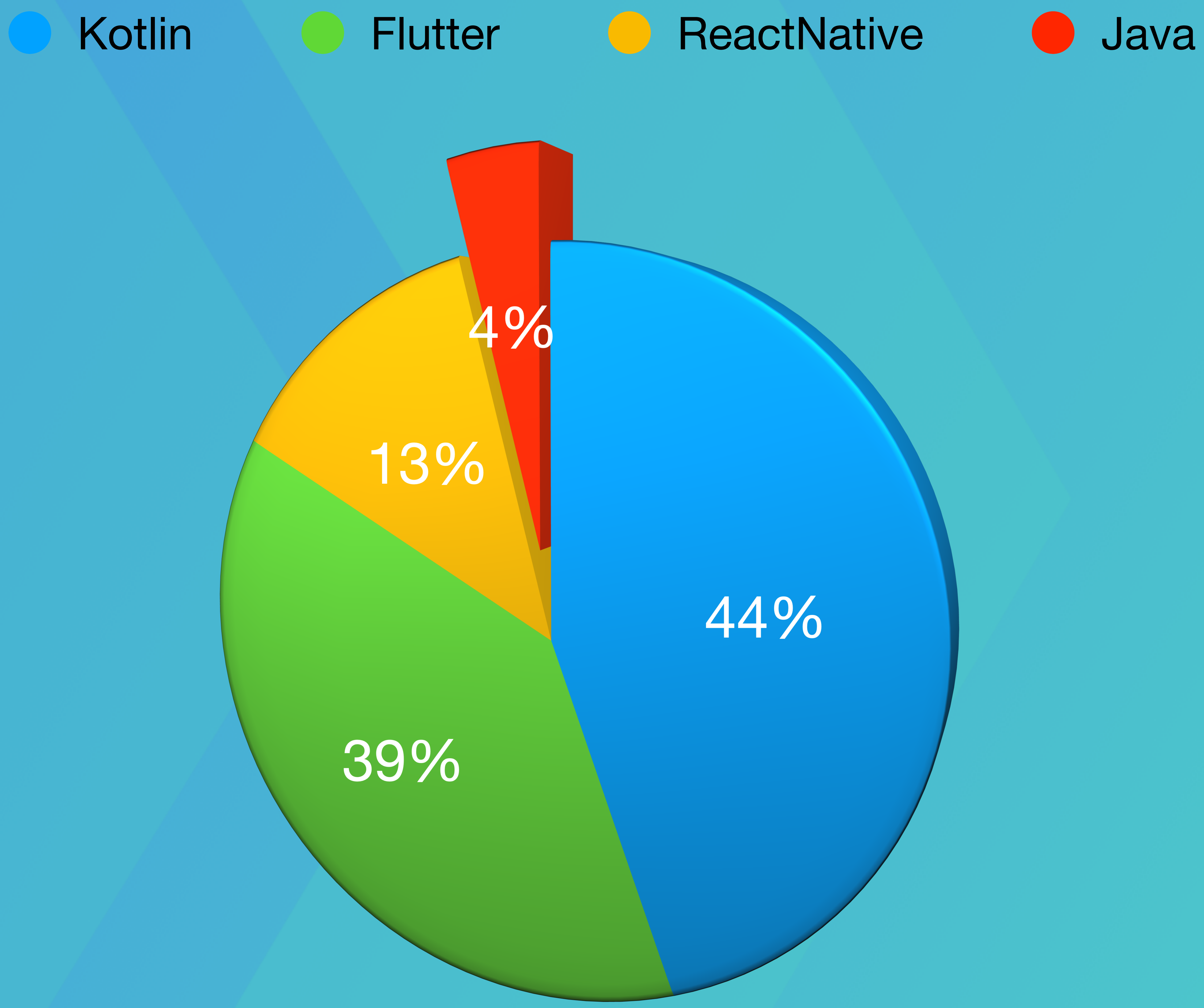



Kotlin
Flutter
ReactNative
Java
Xamarin
Swift
Ionic
1%
2%
7%
2%
14%
39%
35%

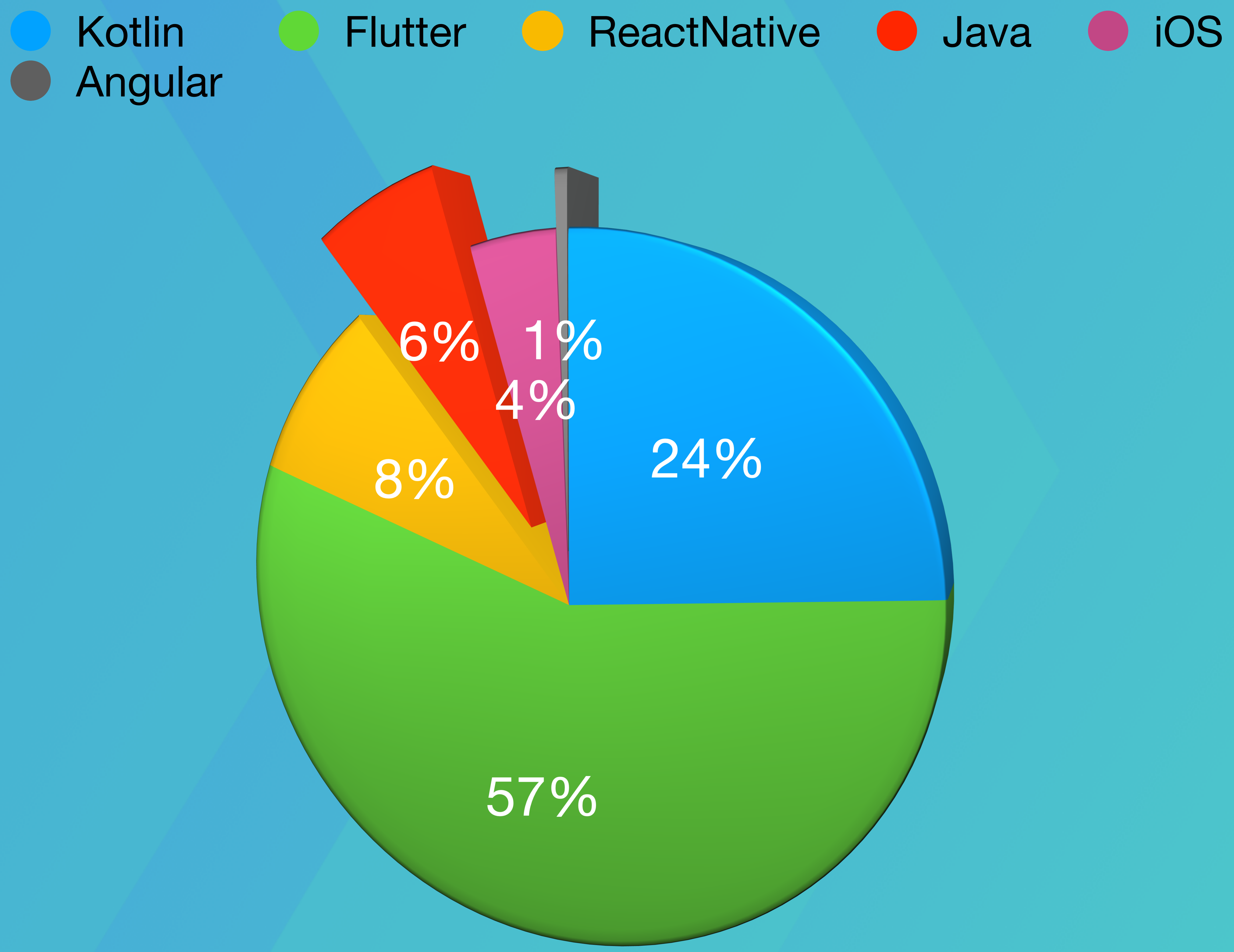
2019


Kotlin
Flutter
ReactNative
Java
Xamarin
Swift
Ionic
1%
9%
1%
1%
13%
40%
35%

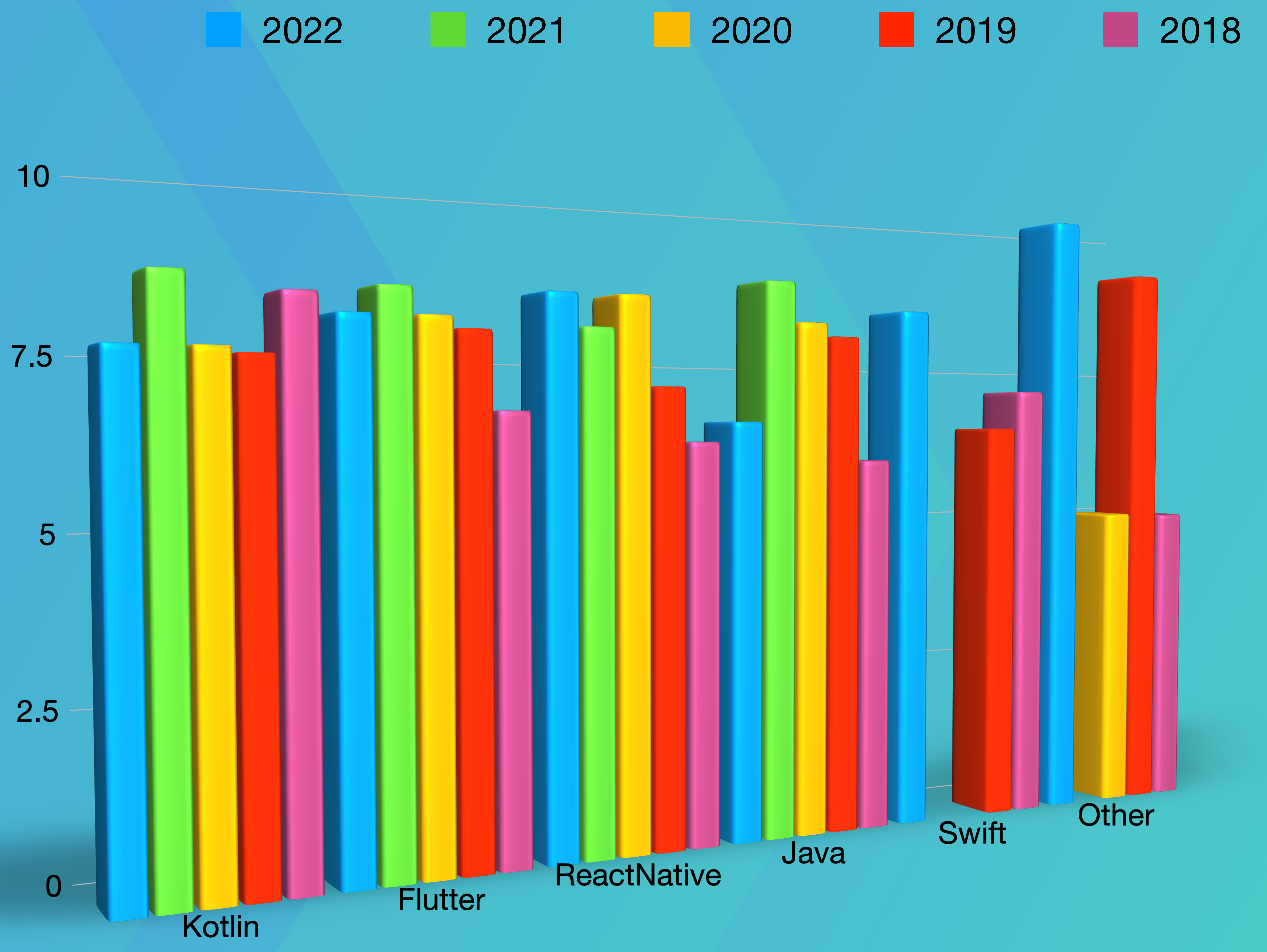
2020



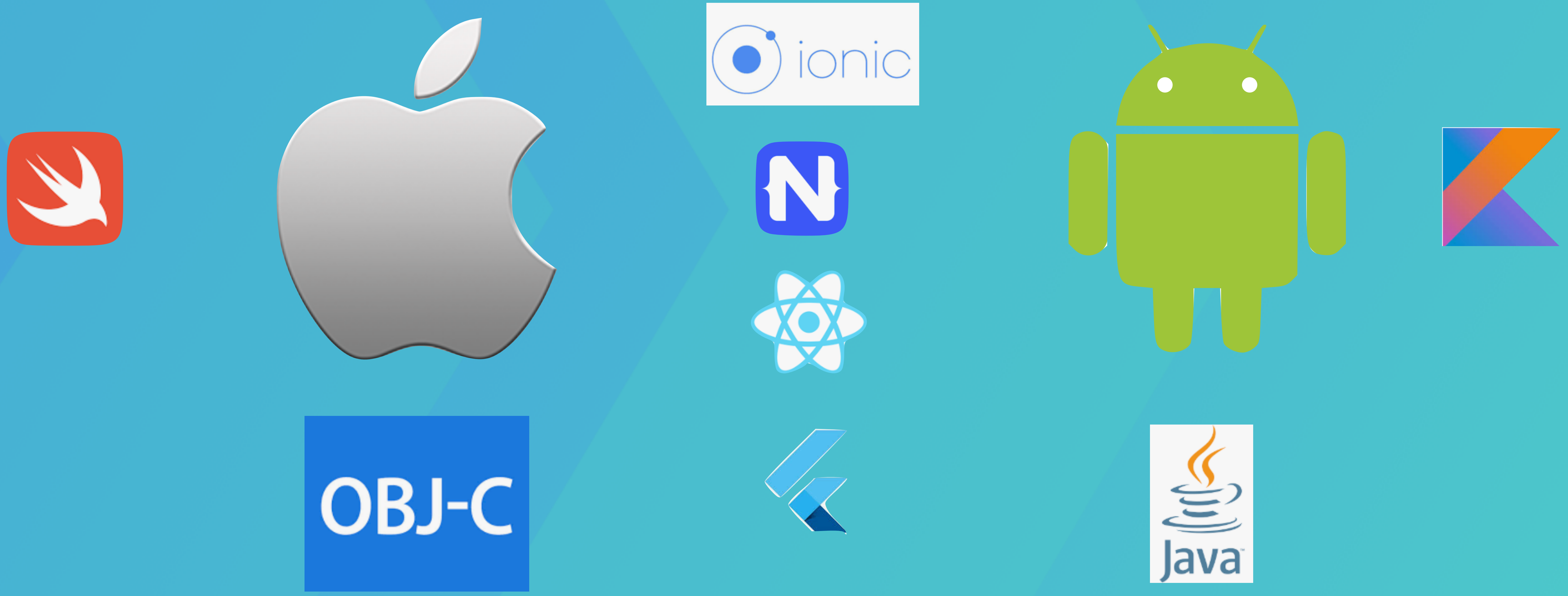
2021



2022


2022
2021
2020
2019
2018
10
7.5
5
2.5
0
Kotlin
Flutter
ReactNative
Java
Swift
Other

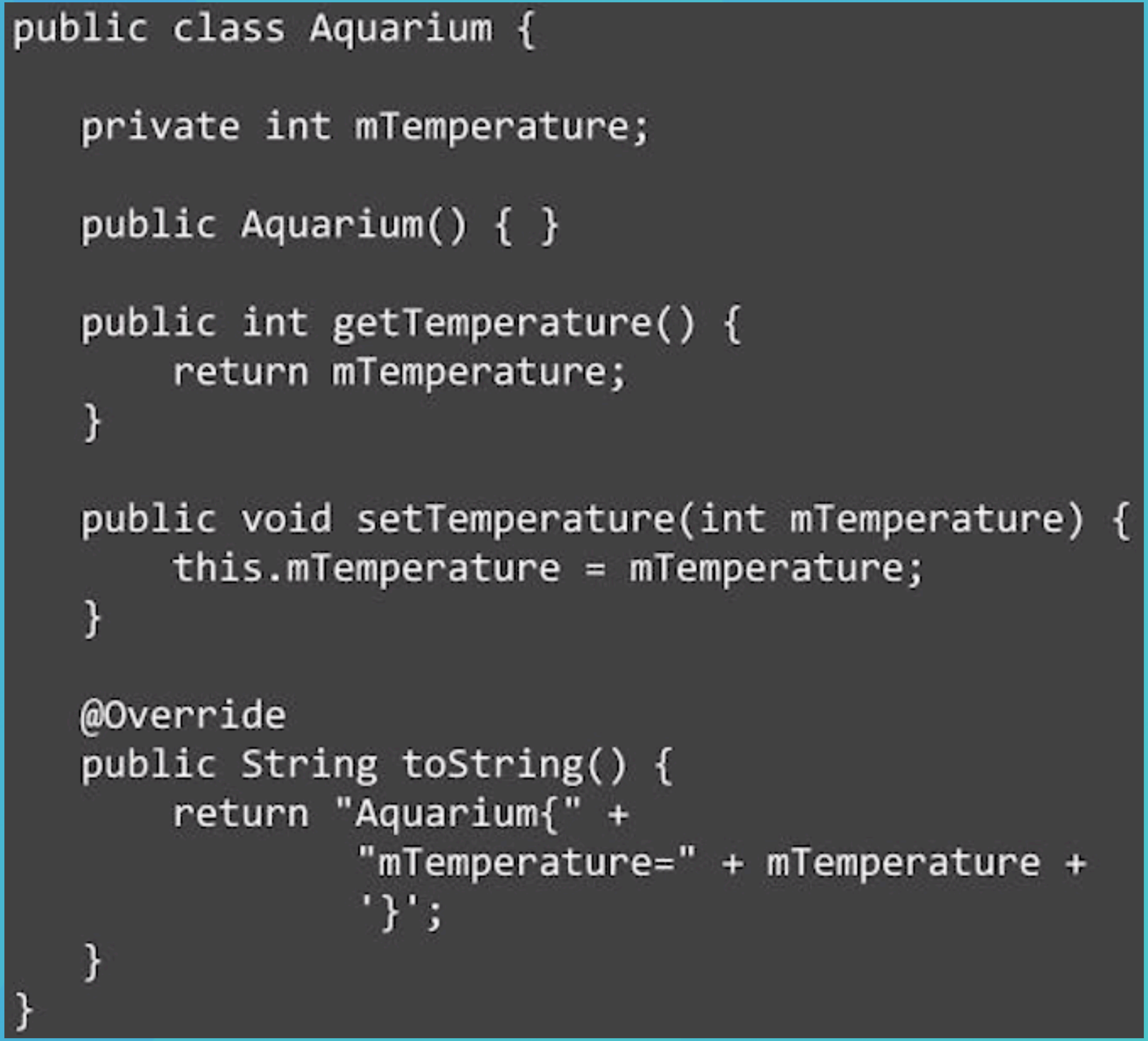
This year



Why Kotlin

- Modern programming language
- Object oriented
- Lambdas, Coroutines, Properties
- Since 2011
- Open Sources 2012
- Official First Class Android Citizen since 2017
- IntelliJ and Android Studio 3.0+

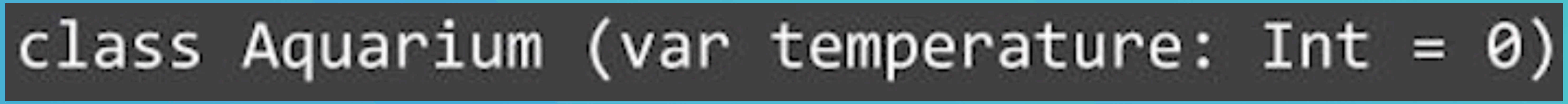
Why Kotlin



```java
public class Aquarium {

    private int mTemperature;

    public Aquarium() { }

    public int getTemperature() {
        return mTemperature;
    }

    public void setTemperature(int mTemperature) {
        this.mTemperature = mTemperature;
    }

    @Override
    public String toString() {
        return "Aquarium{" +
                "mTemperature=" + mTemperature +
                '}';
    }
}
```

Java

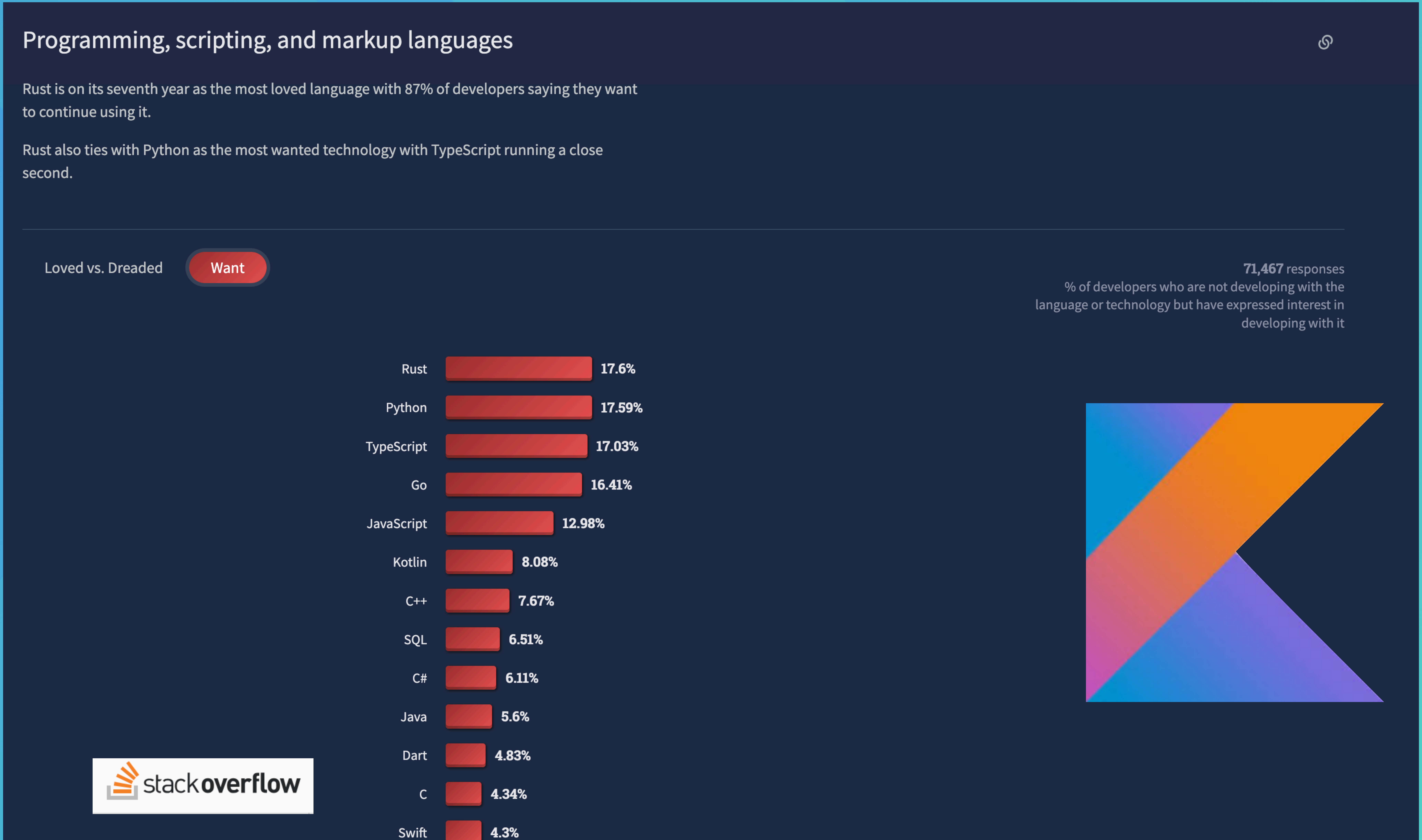
Why Kotlin



```
class Aquarium (var temperature: Int = 0)
```

Kotlin equivalent

Why Kotlin



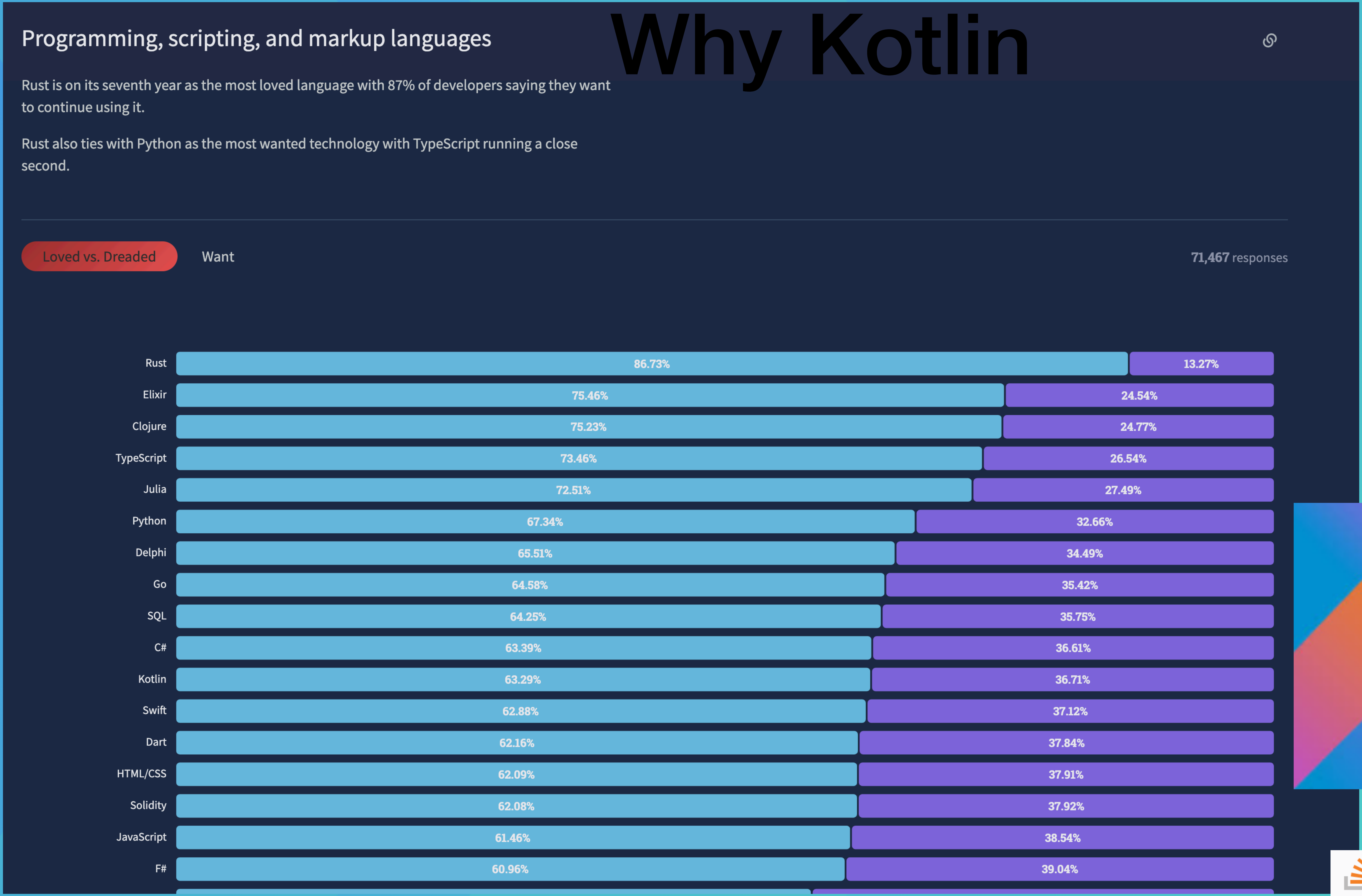
Programming, scripting, and markup languages

Rust is on its seventh year as the most loved language with 87% of developers saying they want to continue using it.

Rust also ties with Python as the most wanted technology with TypeScript running a close second.

71,467 responses
% of developers who are not developing with the language or technology but have expressed interest in developing with it

Why Kotlin

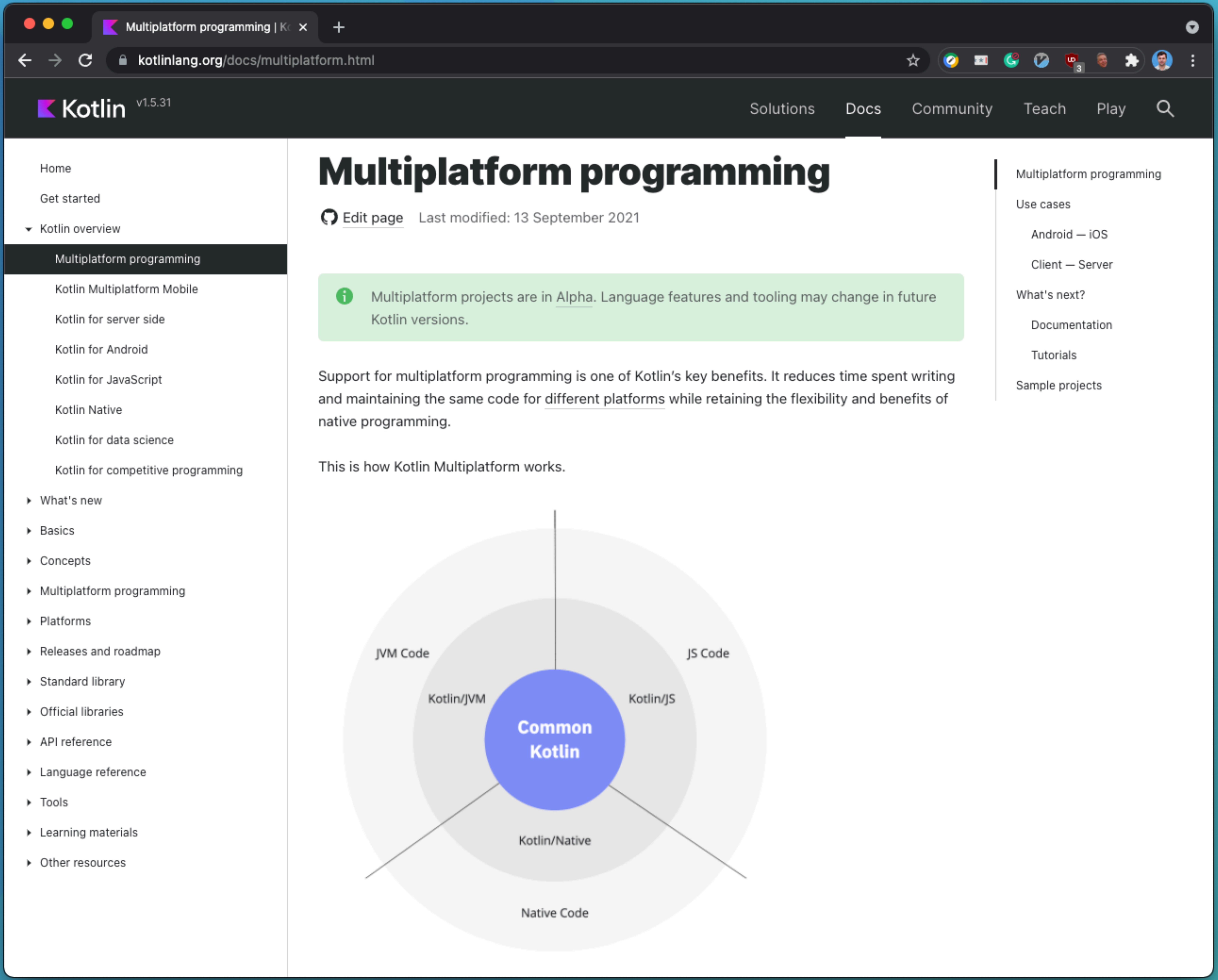


Programming, scripting, and markup languages

Rust is on its seventh year as the most loved language with 87% of developers saying they want to continue using it.

Rust also ties with Python as the most wanted technology with TypeScript running a close second.

Why Kotlin



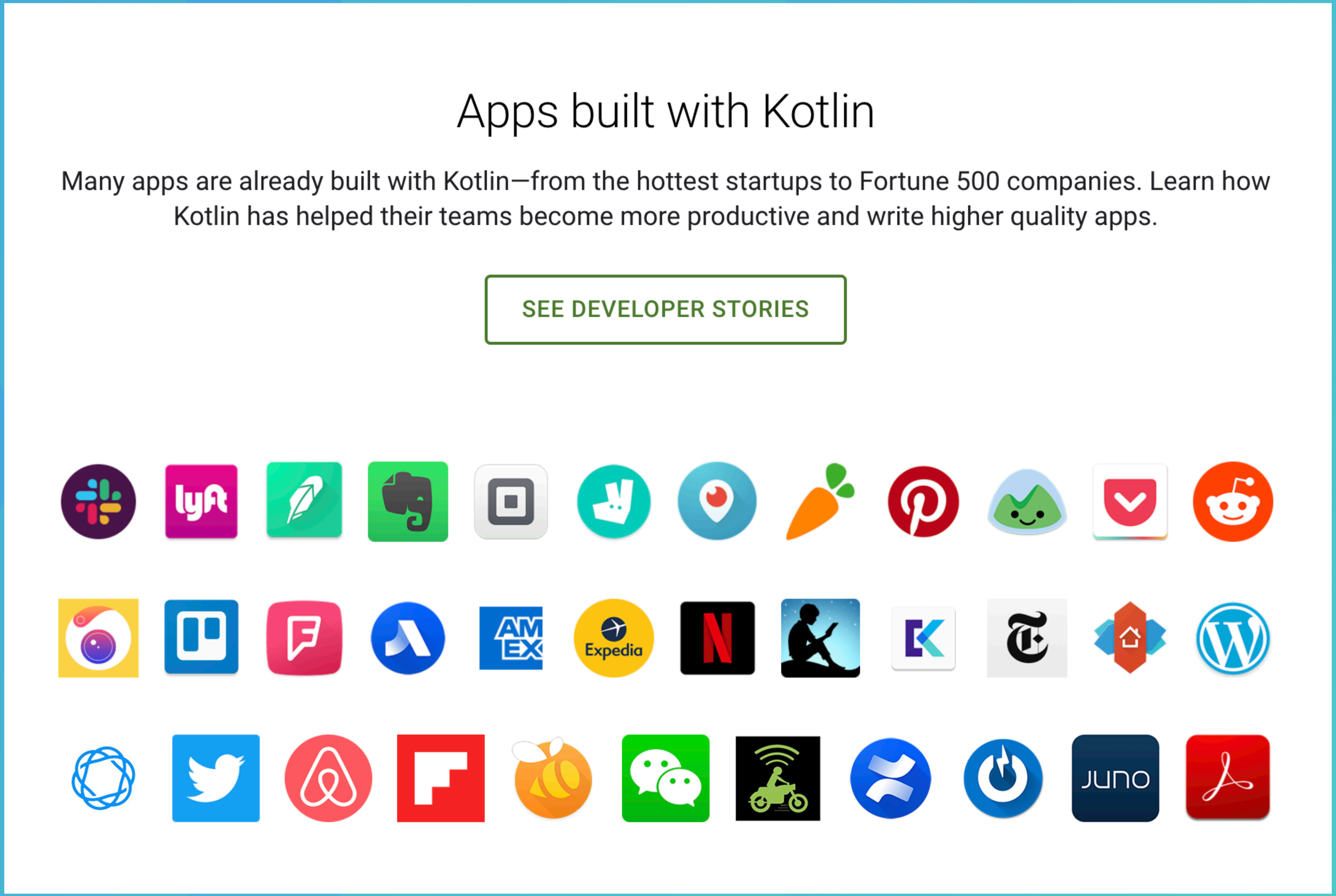
Multiplatform programming

Edit page Last modified: 13 September 2021

Multiplatform projects are in Alpha. Language features and tooling may change in future Kotlin versions.

Support for multiplatform programming is one of Kotlin's key benefits. It reduces time spent writing and maintaining the same code for different platforms while retaining the flexibility and benefits of native programming.

This is how Kotlin Multiplatform works.

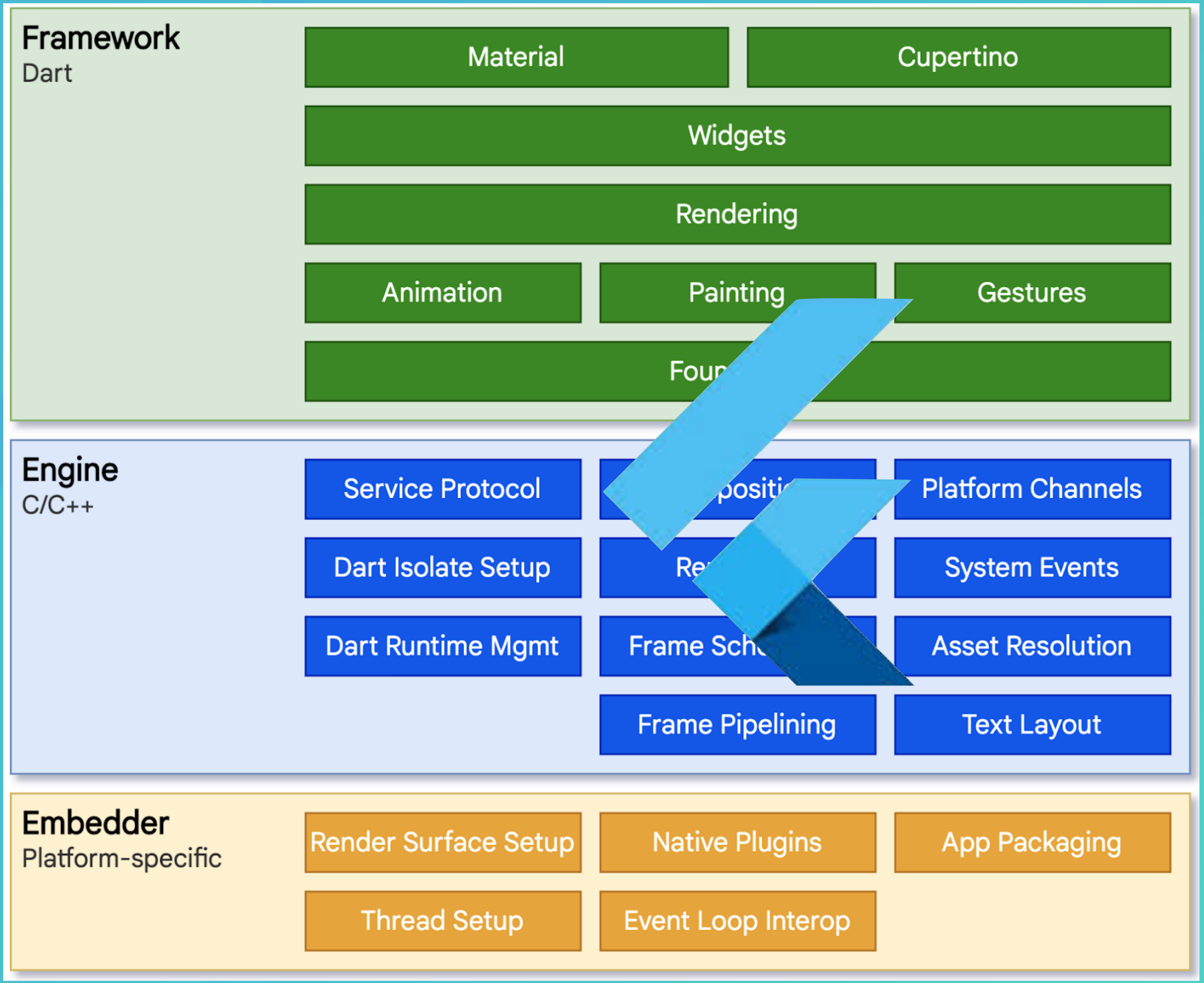


Apps built with Kotlin

Many apps are already built with Kotlin—from the hottest startups to Fortune 500 companies. Learn how Kotlin has helped their teams become more productive and write higher quality apps.

Why Flutter

- Platform-agnostic
- Simplifies and speeds application development
- Easy to learn and easy to use
- Scales well
- Offer an excellent user experience



Framework
Dart

Material | Cupertino
Widgets
Rendering
Animation | Painting | Gestures
Foun

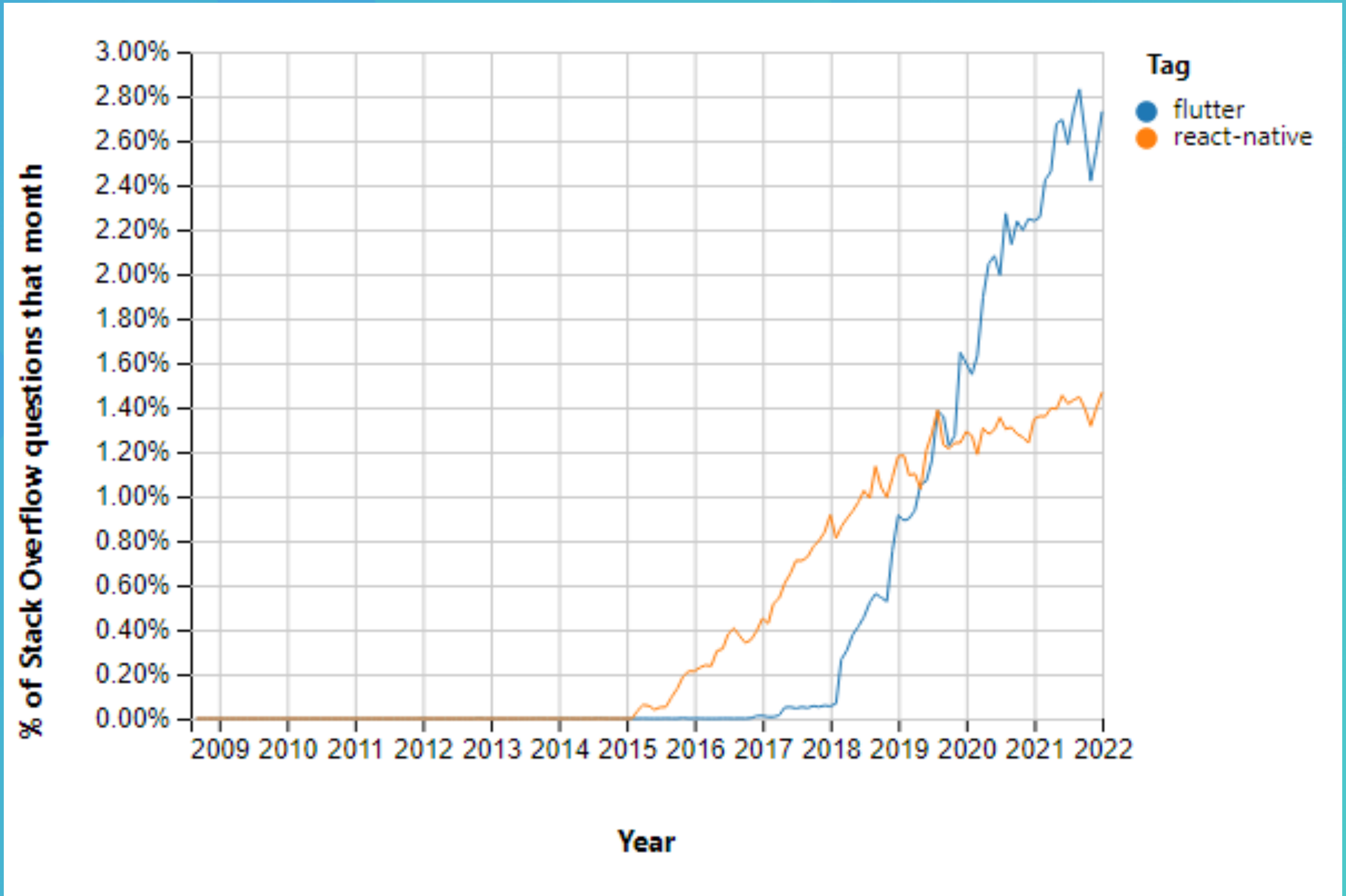
Engine
C/C++

Service Protocol | positi | Platform Channels
Dart Isolate Setup | Re | System Events
Dart Runtime Mgmt | Frame Sch | Asset Resolution
Frame Pipelining | Text Layout

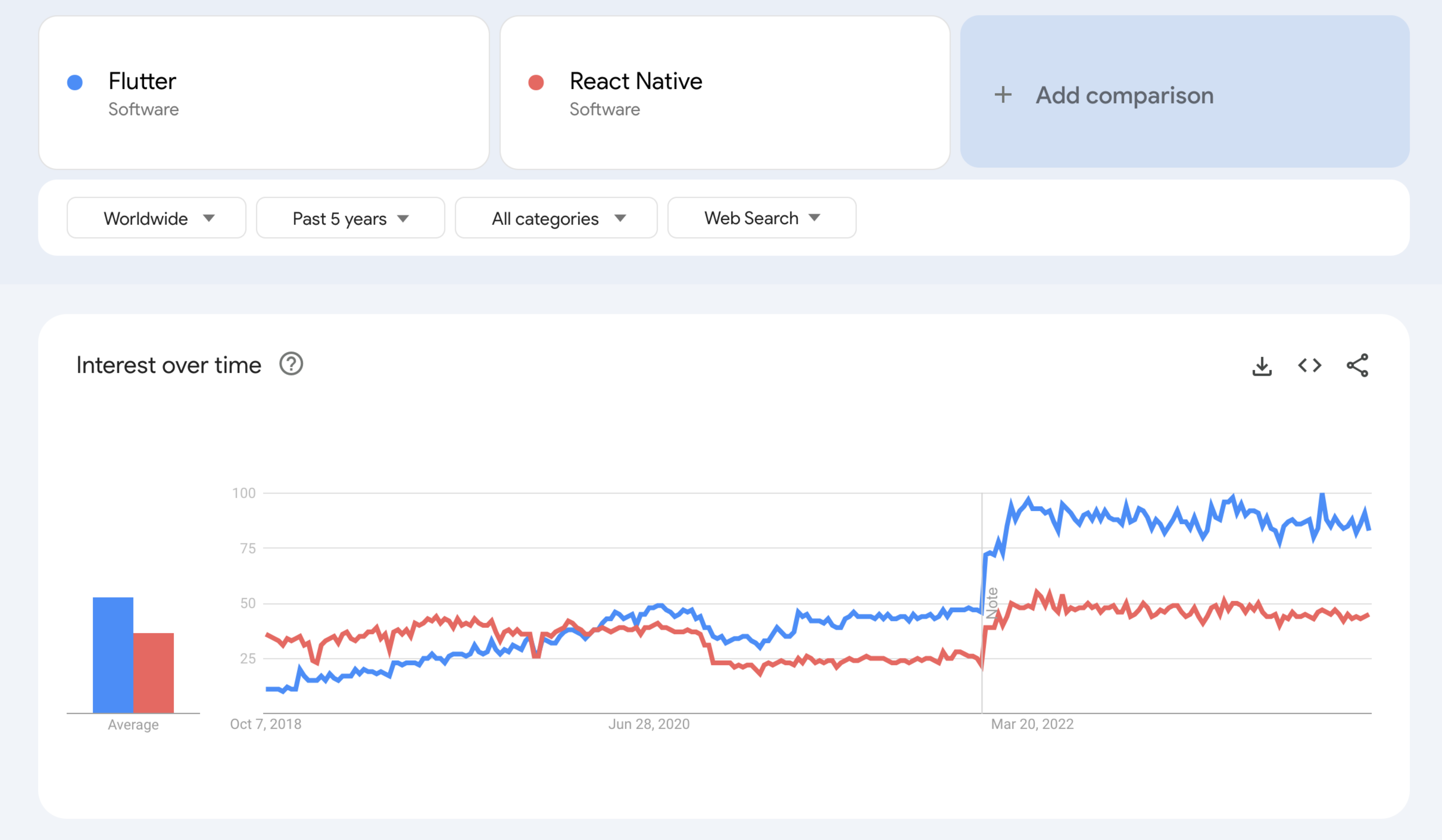
Embedder
Platform-specific

Render Surface Setup | Native Plugins | App Packaging
Thread Setup | Event Loop Interop

Why Flutter


% of Stack Overflow questions that month
3.00%
2.80%
2.60%
2.40%
2.20%
2.00%
1.80%
1.60%
1.40%
1.20%
1.00%
0.80%
0.60%
0.40%
0.20%
0.00%
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
Year
Tag
● flutter
● react-native

Why Flutter



● Flutter
Software

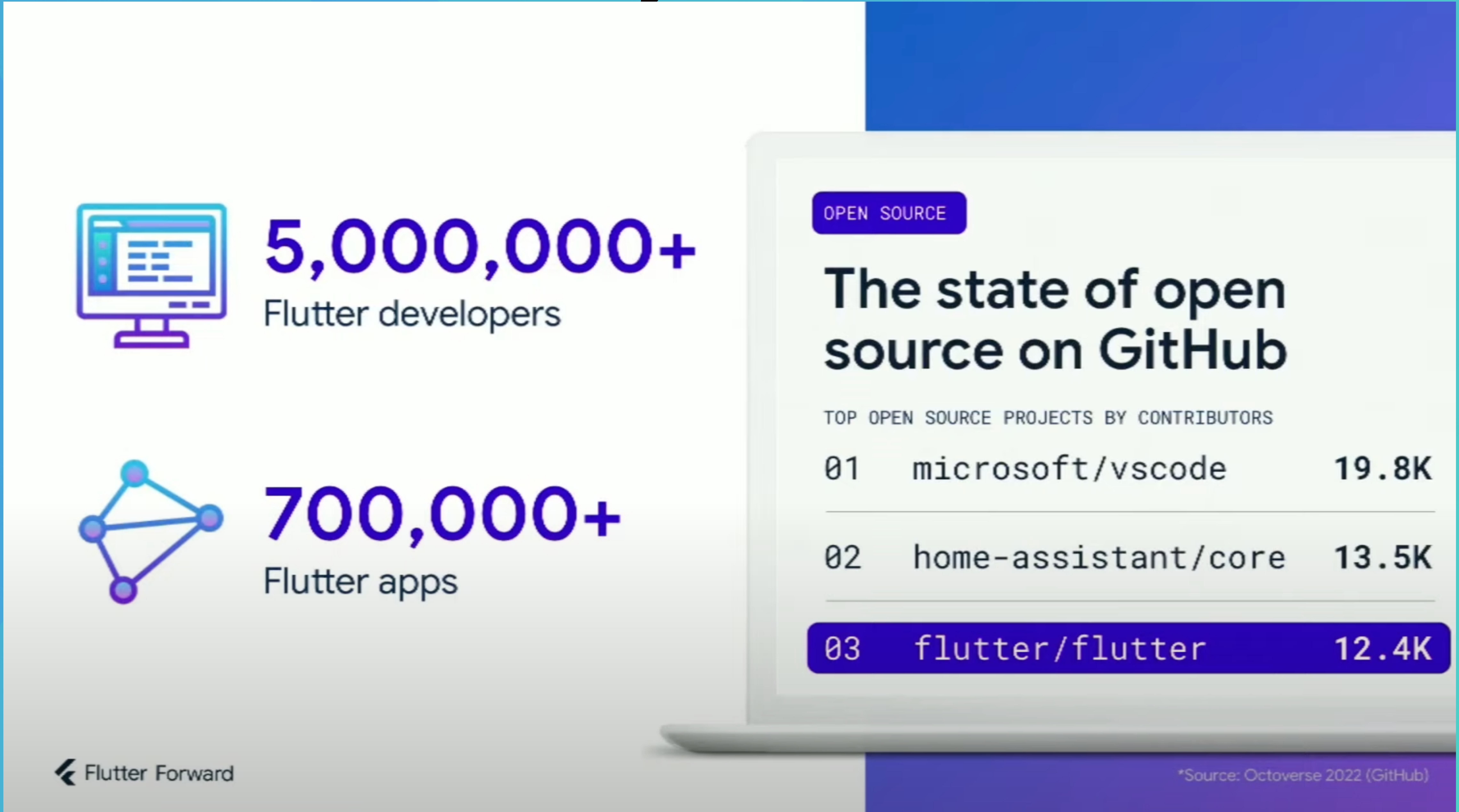
● React Native
Software

+ Add comparison

Worldwide ▾ Past 5 years ▾ All categories ▾ Web Search ▾

Interest over time

Why Flutter



5,000,000+

Flutter developers

700,000+

Flutter apps

OPEN SOURCE

The state of open source on GitHub

TOP OPEN SOURCE PROJECTS BY CONTRIBUTORS

01	microsoft/vscode	19.8K
02	home-assistant/core	13.5K
03	**flutter/flutter**	**12.4K**

Flutter Forward

*Source: Octoverse 2022 (GitHub)

Course Goals

- Knowledge of key base concepts for developing mobile applications.

- Learn the Android platform.

- Learn a framework to develop multi-platform applications (Android&iOS)

Lecture outcomes

- Understand the generated artifacts
- Lifecycle of applications, activities and fragments.
- Use logs to debug and study the behavior.