

Swift programming

2019

Programming Language I

- 1951 - Regional Assembly Language
- 1952 - Autocode
- 1954 - FORTRAN **
- 1954 - IPL (LISP的先驅)
- 1955 - FLOW-MATIC (COBOL的先驅)
- 1957 - COMTRAN (COBOL的先驅)
- 1958 - LISP **
- 1958 - ALGOL 58
- 1959 - FACT (COBOL的先驅)
- 1959 - COBOL **
- 1962 - APL
- 1962 - Simula
- 1962 - SNOBOL
- 1963 - CPL (C的先驅)
- 1964 - BASIC
- 1964 - PL/I
- 1967 - BCPL (C的先驅)

**有三個現代程式語言於1950年代被設計出來

這三者所衍生的語言直到今日仍舊廣泛地被採用

Programming Language II

確立了基礎範式

- 1968 - Logo
- 1970 - Pascal
- 1970 - Forth
- 1972 - C語言
- 1972 - Smalltalk
- 1972 - Prolog
- 1973 - ML
- 1975 - Scheme
- 1978 - SQL (起先只是一種查詢語言，擴充之後也具備了程式結構)

Programming Language III

1980年代：增強、模組、效能

- 1980 - **Ada**
 - 1983 - **C++** (就像有類別的C)
 - 1984 - **Common Lisp**
 - 1985 - **Eiffel**
 - 1986 - **Erlang**
 - 1987 - **Perl**
 - 1988 - **Tcl**
 - 1989 - **FL** (Backus)
- C++**合併了物件導向以及系統程式設計

Programming Languages for Internet

- 1990 - Haskell 提升程式設計師的生產力
- 1991 - Python
- 1991 - Visual Basic
- 1993 - Ruby
- 1993 - Lua
- 1994 - CLOS (part of ANSI Common Lisp)
- 1995 - Java
- 1995 - Delphi (Object Pascal)
- 1995 - JavaScript
- 1995 - PHP
- 1997 - REBOL
- 1999 - D

現今的趨勢

- 元件導向(component-oriented)軟體開發
- 更重視分散式及移動式的應用

- 2001 - C#
- 2001 - Visual Basic .NET
- 2002 - F#
- 2003 - Scala
- 2003 - Factor
- 2006 - Windows PowerShell
- 2007 - Clojure
- 2009 - Go
- 2014 - Swift (程式語言)

Mathematics, Statistics and AI computing

- Mathematical and Statistical Softwares
 - C, C++
 - R, SAS and MATLAB
 - Python
 - Swift **
- Mathematical AI and Applications
 - Parallel and distributed computing
 - Medical images
 - Apple App store
 - Integration of databases, mathematical models, neural networks to Apps on iMac and iPhones **

THE SWIFT PROGRAMMING LANGUAGE

SWIFT 5

×



Welcome to Xcode

Version 10.0 (10A255)



Get started with a playground

Explore new ideas quickly and easily.



Create a new Xcode project

Create an app for iPhone, iPad, Mac, Apple Watch, or Apple TV.



Clone an existing project

Start working on something from a Git repository.



Show this window when Xcode launches



swiftBasic

~/Desktop/JiannMingWu/程式設計二SWIFT



demoKmeans

~/Desktop/JiannMingWu



test

~/Documents



LetterCore

...LetterCore36__demo/LetterCore36__demo



deepvisualization

~/Desktop/Swift-AI-master/deepvis-master

Open another project...

```
1 import UIKit
2
3 var myDep = "AM NDHU"
4 let yearFounded = 1996
5 var message = " was founded in "
6 print(myDep+message+String(yearFounded))
```



"AM NDHU"

1996

" was founded i...

"AM NDHU wa...

AM NDHU was founded in 1996


</> Run Swift

```
1 print("Hello World! :-");
```

```
Swift version 4.0
```

```
user:~ $
```

 Save

 Run

2.2

3.1

4.0

Swift is a fantastic way to write software, whether it's for phones, desktops, servers, or anything else that runs code. It's a safe, fast, and interactive programming language that combines the best in modern language thinking with wisdom from the wider Apple engineering culture and the diverse contributions from its open-source community. The compiler is optimized for performance and the language is optimized for development, without compromising on either.

Swift is friendly to new programmers. It's an industrial-quality programming language that's as expressive and enjoyable as a scripting language. Writing Swift code in a playground lets you experiment with code and see the results immediately, without the overhead of building and running an app.

Swift defines away large classes of common programming errors by adopting modern programming patterns:

- Variables are always initialized before use.
- Array indices are checked for out-of-bounds errors.
- Integers are checked for overflow.
- Optionals ensure that `nil` values are handled explicitly.
- Memory is managed automatically.
- Error handling allows controlled recovery from unexpected failures.

Swift code is compiled and optimized to get the most out of modern hardware. The syntax and standard library have been designed based on the guiding principle that the obvious way to write your code should also perform the best. Its combination of safety and speed make Swift an excellent choice for everything from “Hello, world!” to an entire operating system.

Swift combines powerful type inference and pattern matching with a modern, lightweight syntax, allowing complex ideas to be expressed in a clear and concise manner. As a result, code is not just easier to write, but easier to read and maintain as well.

Swift has been years in the making, and it continues to evolve with new features and capabilities. Our goals for Swift are ambitious. We can't wait to see what you create with it.

A Swift Tour

Tradition suggests that the first program in a new language should print the words “Hello, world!” on the screen. In Swift, this can be done in a single line:

```
5 print("Hello, world!")  
6
```

```
15 var myVariable = 42      42
16 myVariable = 50        50
17 let myConstant = 42    42
18
```

A constant or variable must have the same type as the value you want to assign to it. However, you don't always have to write the type explicitly. Providing a value when


```
Hello, world!
["catfish", "bottle of water", "tulips", "blue
```


</> Run Swift

```
1 print("Hello World! :-");
```

```
Swift version 4.0
```

```
user:~ $
```

 Save

 Run

2.2

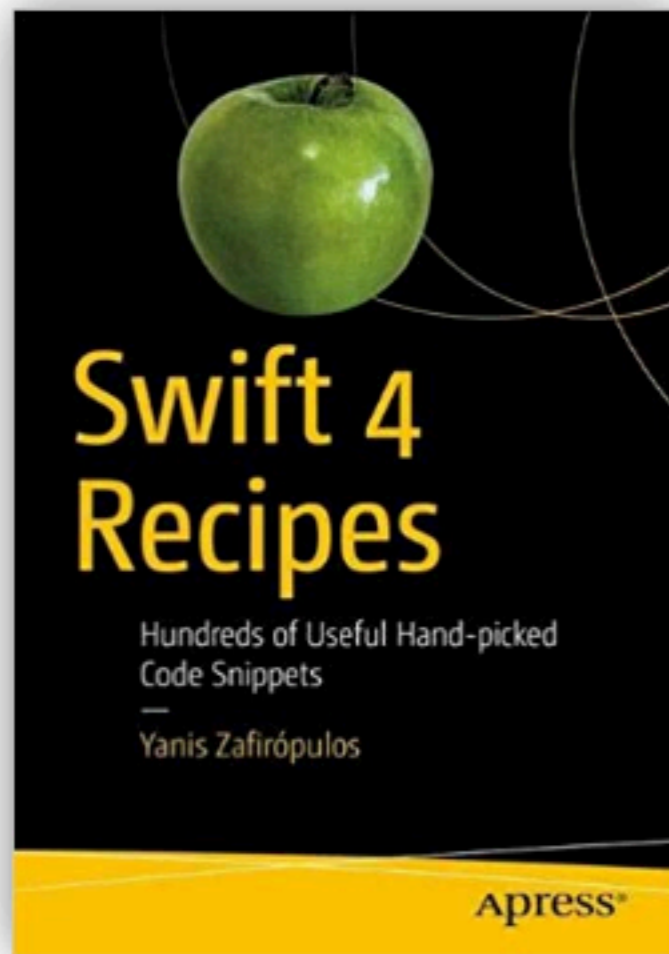
3.1

4.0

<https://iswift.org/playground>

The new Swift 4 cookbook is out!

★ NEW



Swift 4 Recipes

Hundreds of Useful Hand-picked Code Snippets

Get the most out of Swift 4 with this carefully compiled collection of select code snippets designed to solve everyday coding problems. This book features the Apress easy-to-use recipe format, with step-by-step instructions, and a no-fuss approach. You'll explore a wide range of topics, all neatly organized according to the language's own core elements and building blocks. You'll review common topics such as conditionals, loops, functions, classes, closures, and arrays. This book also includes recipes to some more advanced problems found in files, system programming, and algorithms. With Swift 4 Recipes, your programming problems are easily resolved, without wading through paragraphs of text.