

swift语言学习总结

与其他语言对比学习

WeGene移动技术团队——邓积艺

swift语言的发展历史

- 由Chris Lattner主导开发（他同时也是LLVM的最初开发者），主要用于iOS, macOS平台。支持多平台开发，也可用于服务器端开发；已开源
- 2014年发布1.0版本
- 之后几年时间发布了2.0, 3.0, 4.0, 5.0版本
- 在5.0之前，几乎每一个大版本发生重大改变，被戏称每个版本是不同的语言
- 从5.0版本开始，ABI（application binary Interface）开始稳定
- 最新的版本是5.2

swift语言特点

- swift是苹果公司开发的一门编程语言，集合多门语言的优点（objective-c, c++, Ruby, Haskell等），具有类型安全，简单易用，灵活高效的特点。
- 集众所长，既有高级语言灵活的特性，又有类型安全和高效特点
- 还能和Objective-C混编，对OC程序员很友好

安全

静态语言，类型安全

- swift是强类型的编译型语言，支持类型安全和类型推断，类型转换需显式转换（Java的语言类型，高级语言类型）
- 常量let，变量var，常量不可更改（对比其他语言的final，const）。
- 引入可选类型，强大的编译期帮你判断类型安全（有效避免空指针异常，kotlin也有）
- 泛型支持和类型约束（c++和Java有，OC缺乏）

方便易用

各种语法糖，类型推断

- 表达式后面可以省略分号，if let 语法，guard语法等
- 支持全局变量，全局函数，简单宏定义，别名定义typealias（很像c++）
- named parameter（标签化参数或者叫命名参数）：可以给函数参数添加外部标签名，也可以忽略标签名。标签化参数可以增加函数可读性和可理解性，是从OC上保留下的优点
- default parameter（函数参数可以附带默认值）（跟c++差不多）
- tuple和多返回值
- 闭包和函数回调无处不在

灵活

函数式编程，闭包

- 函数式编程和闭包（高阶函数）
- 扩展extension（很方便给类增加功能，从OC保留下来的优点）
- 强大的枚举和switch语法（枚举很强大，结合switch可以写出很好的代码和框架）
- 面向协议编程protocol（面向接口编程）（对比Java的接口）

高效

强大的编译器

- 值类型struct与引用类型class（对比其他语言的基本数据类型和引用类型）
- 自动引用计数ARC（对比Java的垃圾回收机制）
- 内存模型（反射机制和内存操作）
- String字符串优化（类比Java语言的字符串存储）

兼容性

- 与Objective-C混编，支持Objective-C接口互相调用（有点类似Java和Kotlin的相互调用，但是本质不同）
- 通过Objective-C的接口进行c++的混合编程
- NSObject是桥梁

参考文献

- 《The Swift Programming Language》 <https://swiftgg.gitbook.io/swift/>
- Swift 对象内存模型探究（一） <https://mp.weixin.qq.com/s/zIkB9KnAt1YPWG0OwyqY3Q>
- 汇编窥探Swift String的底层 <https://www.cnblogs.com/mjios/p/11799136.html>
- Swift指针UnsafePointer <https://www.jianshu.com/p/e90393ba2aea>
- [HandyJSON] 设计思路简析 <https://www.jianshu.com/p/eac4a92b44ef>