

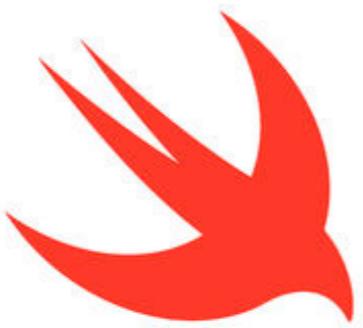
Conclusion

Alex Vollmer
@alexvollmer
<http://alexvollmer.com>



pluralsight 
hardcore dev and IT training



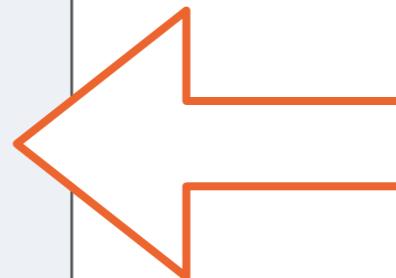
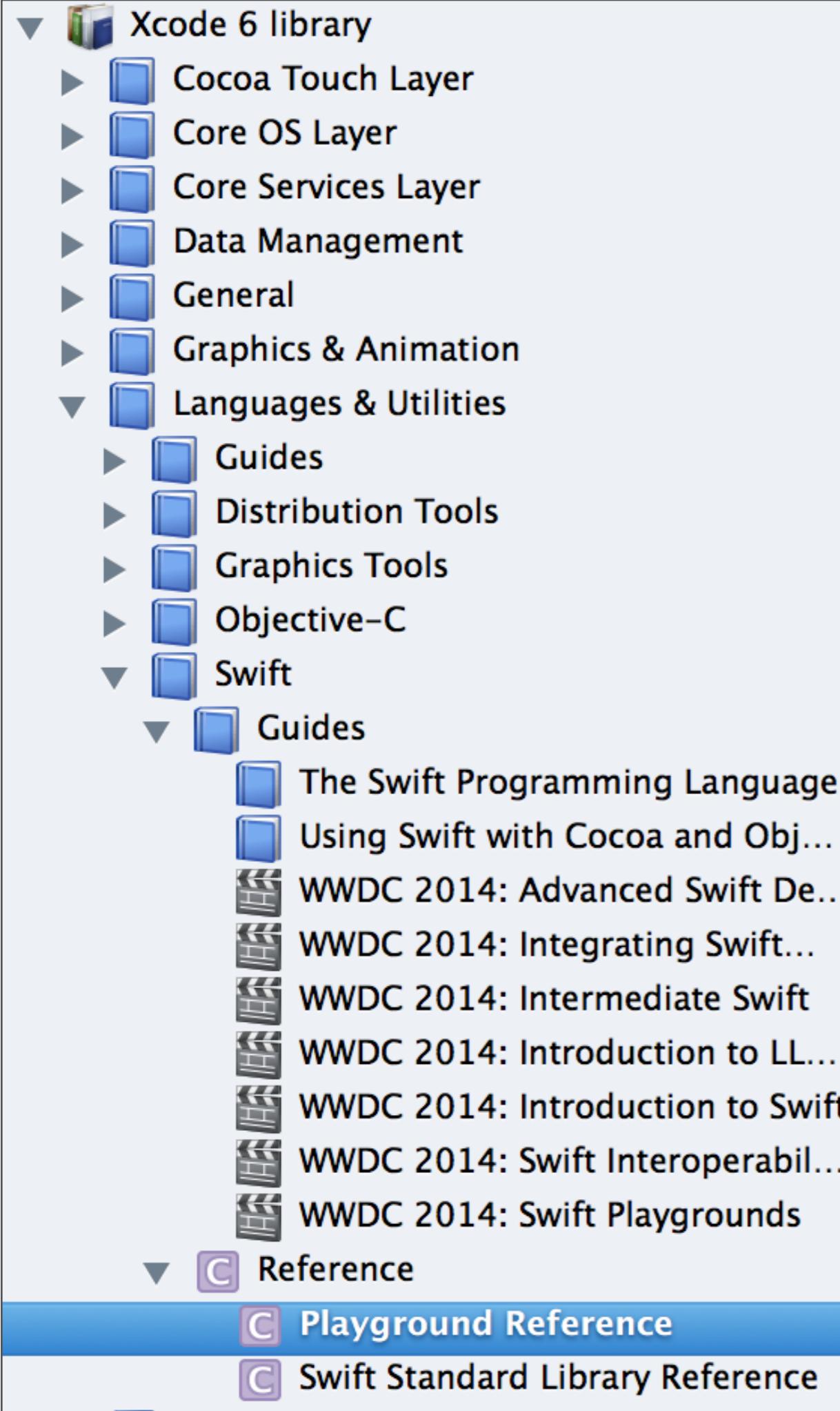


The Swift Programming Language



Using Swift with Cocoa and Objective-C





XCPlayground Module Reference

- XCPCaptureValue
- XCPExecutionShouldContinueIndefinitely
- XCPSetExecutionShouldContinueIndefinitely
- XCPShowView
- XCPSharedDataDirectoryPath

Interactive Learning Document Features

- playground
- sections
- documentation
- code
- timeline

[Revision History](#)

XCPlayground Module Reference

To use this module in a playground, import the `XCPlayground` module as follows:

```
import XCPlayground
```

XCPCaptureValue

`Captures a value to be displayed in the specified value history in the timeline.`

[Revision History](#)

```
func XCPCaptureValue<T>(identifier: String, value: T)
```

`identifier`

The identifier of the value history.

`value`

The value to be captured.

Capturing multiple values with the same value history identifier displays the values on the same timeline item such as a graph. Capturing values with different identifiers displays them on different timeline items.

The identifier is displayed in the timeline as the item's title.

XCPExecutionShouldContinueIndefinitely

Returns a Boolean value indicating whether indefinite execution is enabled.

```
func XCPExecutionShouldContinueIndefinitely() -> Bool
```

Return Value Returns `true` if execution continues after the end of the playground's top-level code is reached; otherwise, `false`.

See also [XCPSetExecutionShouldContinueIndefinitely](#)

Part I: Types

- String
 - Creating a String
 - Querying a String
 - var isEmpty { get }
 - hasPrefix(_ :) -> Bool
 - Declaration
 - Discussion
 - hasSuffix(_ :) -> Bool
 - Converting Strings
 - Operators
 - Array<T>
 - Dictionary<KeyType, ValueType>
 - Numeric Types
- Part II: Protocols
- Part III: Free Functions

String

A String represents an ordered collection of characters.

For a full discussion of String, see [Strings and Characters](#).

Creating a String

init()

Constructs an empty string.

Declaration

```
init()
```

Discussion

Creating a string using this constructor:

```
let emptyString = String()
```

is equivalent to using double-quote convenience syntax:

```
let equivalentString = ""
```

init(count:, repeatedValue:)

Constructs a string with a single character repeated a given number of times.

```
1216     func filter(includeElement: (T) -> Bool) -> [T]
1217 }
1218
1219 extension Array : Reflectable {
1220     func getMirror() -> MirrorType
1221 }
1222
1223 extension Array : Printable, DebugPrintable {
1224     var description: String { get }
1225     var debugDescription: String { get }
1226 }
1227
1228
1229
1230 extension Array {
1231
1232     /// Call body(p), where p is a pointer to the Array's contiguous storage
1233     func withUnsafeBufferPointer<R>(body: (UnsafeBufferPointer<T>) -> R) -> R
1234     mutating func withUnsafeMutableBufferPointer<R>(body: (inout UnsafeMutableBufferPointer<T>) -> R)
1235 }
1236
1237 extension Array {
1238
1239     /// This function "seeds" the ArrayLiteralConvertible protocol
1240     static func convertFromHeapArray(base: Builtin.RawPointer, owner: Builtin.NativeObject, count: B
1241 }
1242
1243 extension Array {
1244     mutating func replaceRange<C : CollectionType where T == T>(subRange: Range<Int>, with newValues:
1245     mutating func splice<S : CollectionType where T == T>(s: S, atIndex i: Int)
1246     mutating func removeRange(subRange: Range<Int>)
1247 }
1248
1249 extension Array {
1250
1251     /// Construct from the given `_CocoaArrayType`. If `noCopy` is `true`,
1252     /// either `source` must be known to be immutable, or the resulting
1253     /// `Array` must not survive across code that could mutate `source`.
1254     init(_fromCocoaArray source: _CocoaArrayType, noCopy: Bool = default)
1255 }
1256
1257
```

Home — Swifter

swifter.natecook.com

Reader

Swifter

Auto-generated documentation for Swift. Command-click no more.

Search 

Versions: [Swift 1.1 b2](#) | [Swift 1.0 GM](#)

Types

Array	MapCollectionView
AssertString	MapSequenceGenerator
AutoreleasingUnsafeMutablePointer	MapSequenceView
BidirectionalReverseView	MirrorDisposition
Bit	ObjectIdentifier
Bool	OnHeap
CFunctionPointer	Optional
COpaquePointer	PermutationGenerator
CVaListPointer	QuickLookObject
Character	RandomAccessReverseView
ClosedInterval	Range

Developer Forums: Swift

https://devforums.apple.com/community/tools/languages/swift

Welcome, Alex Vollmer (Settings | Log out)

Developer Forums

Developer Tools > Languages > Swift

Swift

New Thread Email Updates

Statistics: Messages: 16,408 Threads: 2,438

15 Per Page

Subject	Views	Replies	Last Post
Command-click to see Swift representation of Objective-C API in Xcode	3,316	2	3 months ago SevenTenEleven
README: About the Swift Programming Language	5,992	1	3 months ago Charles Srstka
Stepping through a function produces strange results	40	3	2 minutes ago k8stone
Handling NSExceptions in Swift	49	2	2 hours ago jckarter
Can't Get Keychain Items.	28	1	2 hours ago Zhou Yang
There is something that still REALLY bugs me about swift, coredata and namespaces	92	3	3 hours ago Zac Tolley
Help request with Swift generics	57	4	3 hours ago Vittorio Ficarra

Quick Links

Documentation and Downloads

- iOS Dev Center
- Mac Dev Center
- Safari Dev Center
- App Store Resource Center
- App Review Guidelines
- iTunes Connect
- Licensing and Trademarks

Membership

- Joining a Program
- Account Summary
- Membership Agreements

Help and Support

- Get Forums Help
- Ask Questions
- Forums Guidelines
- Request Technical Support
- Report Bugs / Suggest Features
- Contact Support



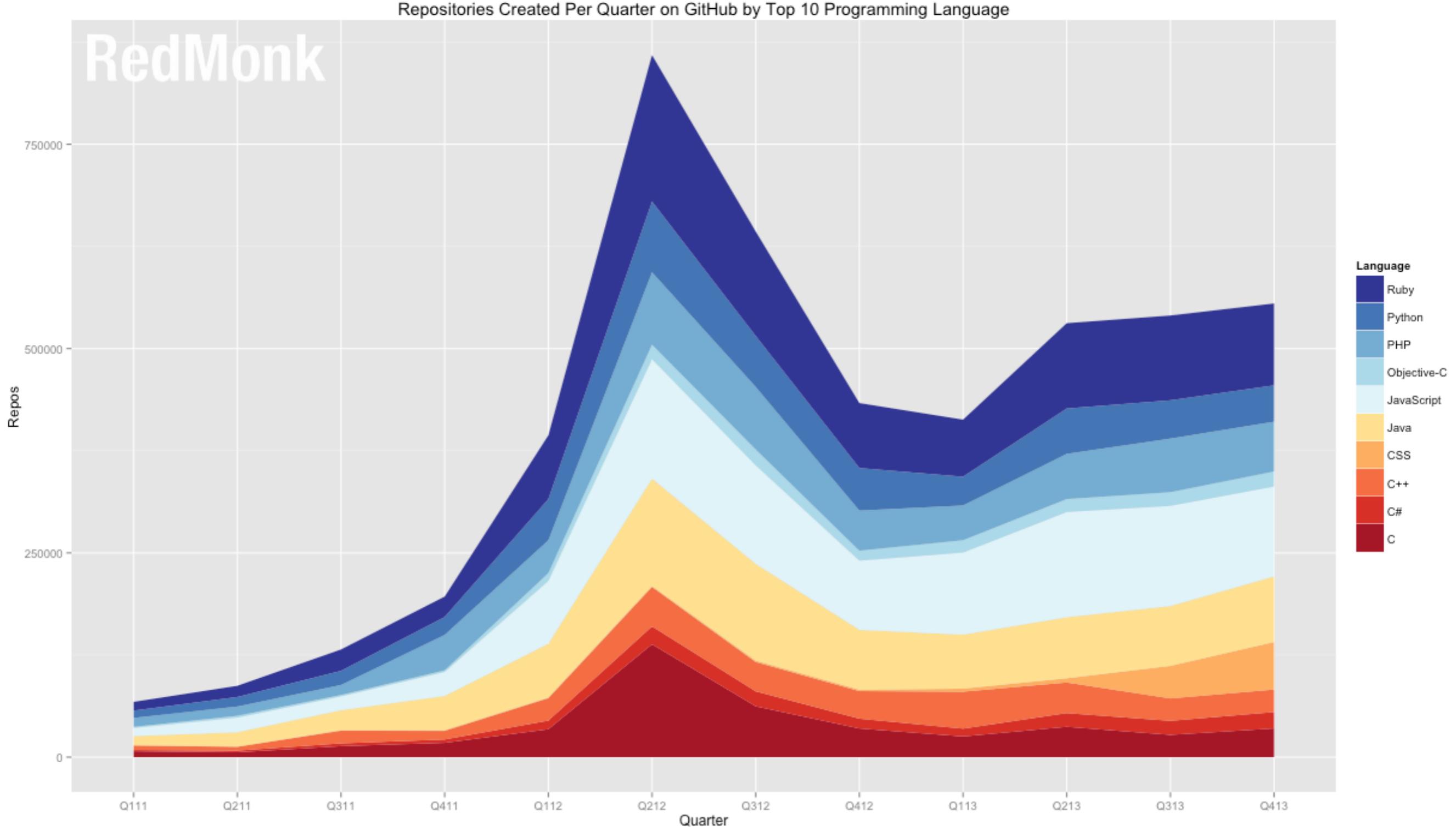
<https://www.flickr.com/photos/magnera/6208260197>



<https://www.flickr.com/photos/all dread/5441981648>



Do we *really* need another programming language?





Is this language really stable?



[SCNetworkReachabilitySetCallback failing in Swift](#)



[Extension and shared NSUserDefaults, not working when debugging](#)



[Swift - Horrible Performance in Binary Search Recursion](#)



[Swift Framework does not include symbols from extensions to generic structs](#)



[Why doesn't this code work?](#)



[\[AnyObject\]\(\) is crashing my app?](#)



[Is there a plan to support static library target or workarounds?](#)



[Cocoa Touch Framework using Swift fails on device](#)



[Xcode 6 With Playground and Swift \(Execution was interrupted, reason EXC_BAD_INSTRUCTION\)](#)



[Overriding Results in Ambiguous Use Error](#)



[Compiler bug with extension and overridden computed var ?](#)



This is less flexible & dynamic than I'm used to.



json parsing in swift



Web

Videos

News

Shopping

Images

More ▾

Search tools

About 696,000 results (0.35 seconds)

Downloading and parsing json in swift - Stack Overflow

[stack overflow .com/questions/.../downloading-and-parsing-json-in-swift](https://stackoverflow.com/questions/18193545/downloading-and-parsing-json-in-swift) ▾

I'm trying to get the **JSON** from a website and **parse** it before putting it inside of the iOS view. Here's my code; func startConnection(){ let urlPath: String ...

Efficient JSON in Swift with Functional Concepts and Generics

robots.thoughtbot.com/efficient-json-in-swift-with-functional-concepts-a... ▾

Aug 6, 2014 - People were jumping into **Swift** with Xcode Beta1 immediately and it didn't take long to realize that **parsing JSON**, something almost every app ...

Calling APIs & Parsing JSON with Swift – topcoder

www.topcoder.com/blog/calling-apis-parsing-json-with-swift/ ▾ TopCoder ▾

Aug 22, 2014 - **Parsing JSON** using **Swift** is not a happy task. One of the first things I wanted to do in a Playground was call the topcoder API and start playing ...

Swift: JSON Parsing — Swift Programming — Medium

<https://medium.com/swift.../swift-json-parsing-716ea9be1c5b> ▾

Swift Programming David Owens II on Jun 18. 1 min. Info; History. Publish ... **Swift** : **JSON Parsing**. Help me with the errors of my ways. I've moved: **JSON Parsing**.

A safer approach to JSON parsing in Swift - Roadfire Software



Oooh. That's a bit clunky.

```
class Singleton {  
    class var sharedInstance : Singleton {  
        struct Static {  
            static var onceToken : dispatch_once_t = 0  
            static var instance : Singleton? = nil  
        }  
        dispatch_once(&Static.onceToken) {  
            Static.instance = Singleton()  
        }  
        return Static.instance!  
    }  
}
```

The Singleton pattern in Swift



Leave me alone! I know what I'm doing!

! Execution was interrupted, reason: EXC_BAD_INSTRUCTION (code=EXC...

! Class 'Person' has no initializers

! Property 'self.firstName' not initialized

! Immutable value of type '[String]' only has mutating members named 'append'

! Convenience initializer for 'Person' must delegate (with 'self.init')

! Type 'String' does not conform to protocol 'IntegerLiteralConvertible'

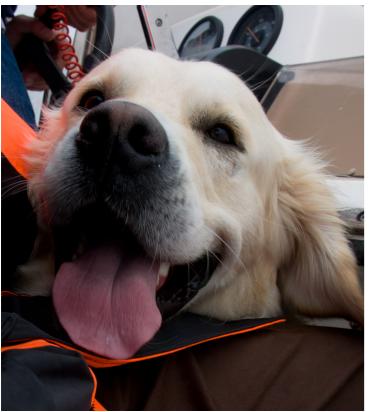
! Deinitializer does not have a parameter clause

3

! Static properties are only allowed within structs and enums; use 'class' to declare a class property

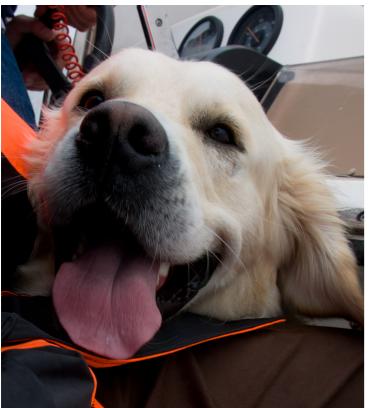
! Type 'Person' does not conform to protocol 'Equatable'





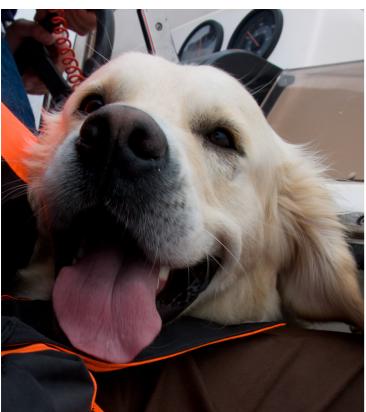
Hey guys! No more C!

C



Less typing makes me happy!

```
NSMutableArray *fileData = [NSMutableArray arrayWithCapacity:files.count];
for(NSFileHandle *handle in files) {
    let filteredData = files.map(){ $0.availableData };
    [fileData addObject:handle.availableData];
}
```

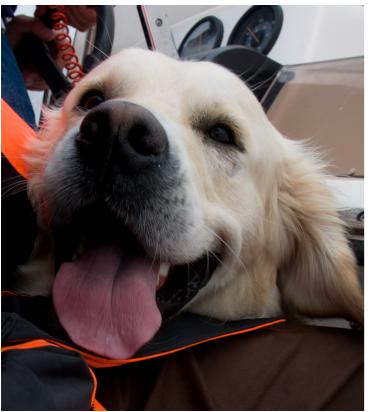


Let's go play!

```
Apple Xcode File Edit View Find Navigate Editor Product Debug Source Control Window Help ... Demo Q ⌂

Swift | sort(_)

5629 /// Return an Array containing the elements of source in reverse
5630 /// order.
5631 func reverse<C : CollectionType where C.Index : BidirectionalIndexType>(source: C) -> [C.Generator.Element]
5632
5633 func sizeof<T>(_: T.Type) -> Int
5634
5635 func sizeofValue<T>(_: T) -> Int
5636
5637 func sort<T>(inout array: [T], isOrderedBefore: (T, T) -> Bool)
5638
5639
5640 /// Sort `collection` in-place.
5641 /// Requires: The less-than operator ('func <') defined in the Comparable
5642 /// conformance is a `strict weak ordering
5643 /// <http://en.wikipedia.org/wiki/Strict\_weak\_order#Strict\_weak\_orderings>` __
5644 /// over `elements`.
5645 func sort<C : MutableCollectionType where C.Index : RandomAccessIndexType, C.Generator.Element : Comparable>(inout collection: C)
5646
5647 func sort<T : Comparable>(inout array: [T])
5648
5649
5650 /// Sort `collection` in-place according to 'isOrderedBefore'.
5651 /// Requires: `isOrderedBefore` is a `strict weak ordering
5652 /// <http://en.wikipedia.org/wiki/Strict\_weak\_order#Strict\_weak\_orderings>` __
5653 /// over `elements`.
5654 func sort<C : MutableCollectionType where C.Index : RandomAccessIndexType>(inout collection: C, isOrderedBefore: (C.Generator.Element, C.Generator.Element) -> Bool)
```



This looks familiar...

```
class Person {  
    var firstName: String  
    var lastName: String  
  
    init(firstName: String, lastName: String) {  
        self.firstName = firstName  
        self.lastName = lastName  
    }  
}
```

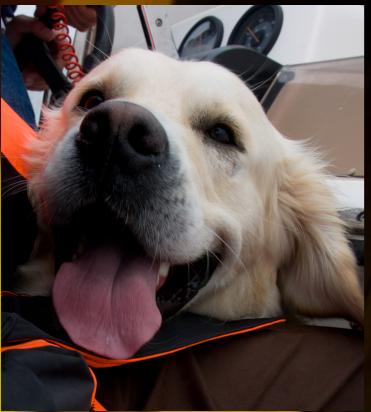
Swift

```
class Person {  
    public String firstName;  
    public String lastName;  
  
    public Person(String firstName, String lastName) {  
        this.firstName = firstName;  
        this.lastName = lastName;  
    }  
}
```

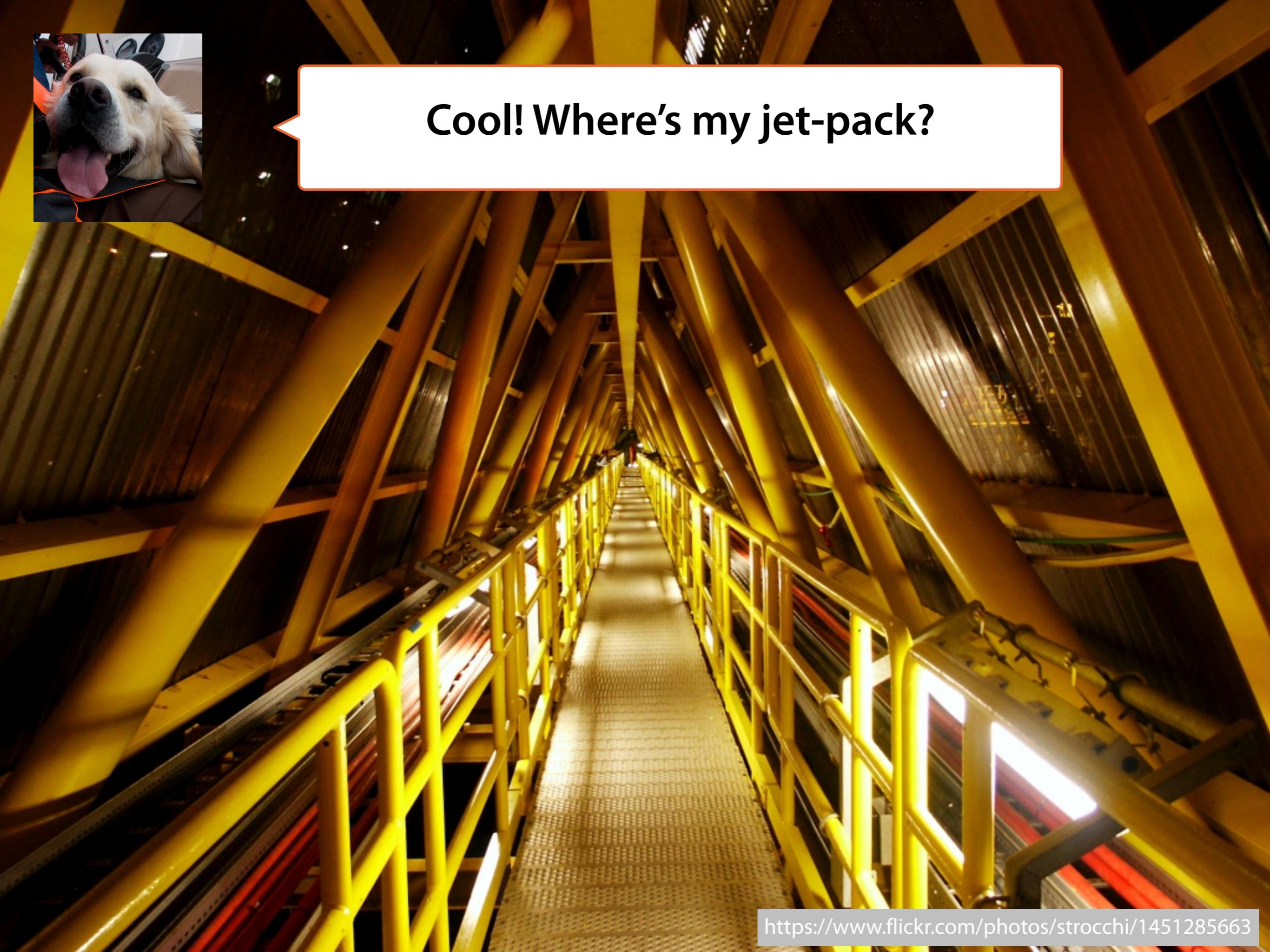
C#

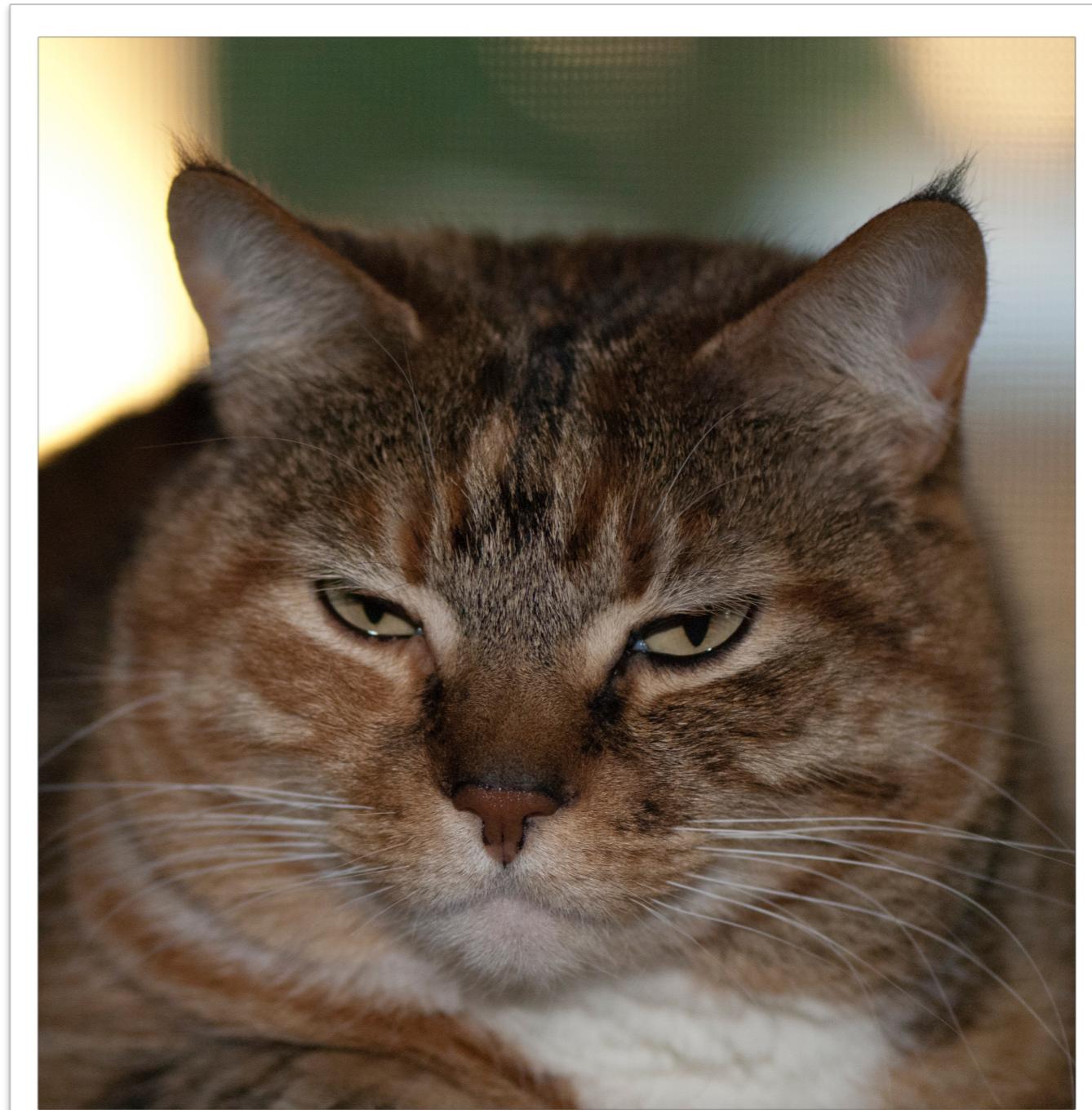
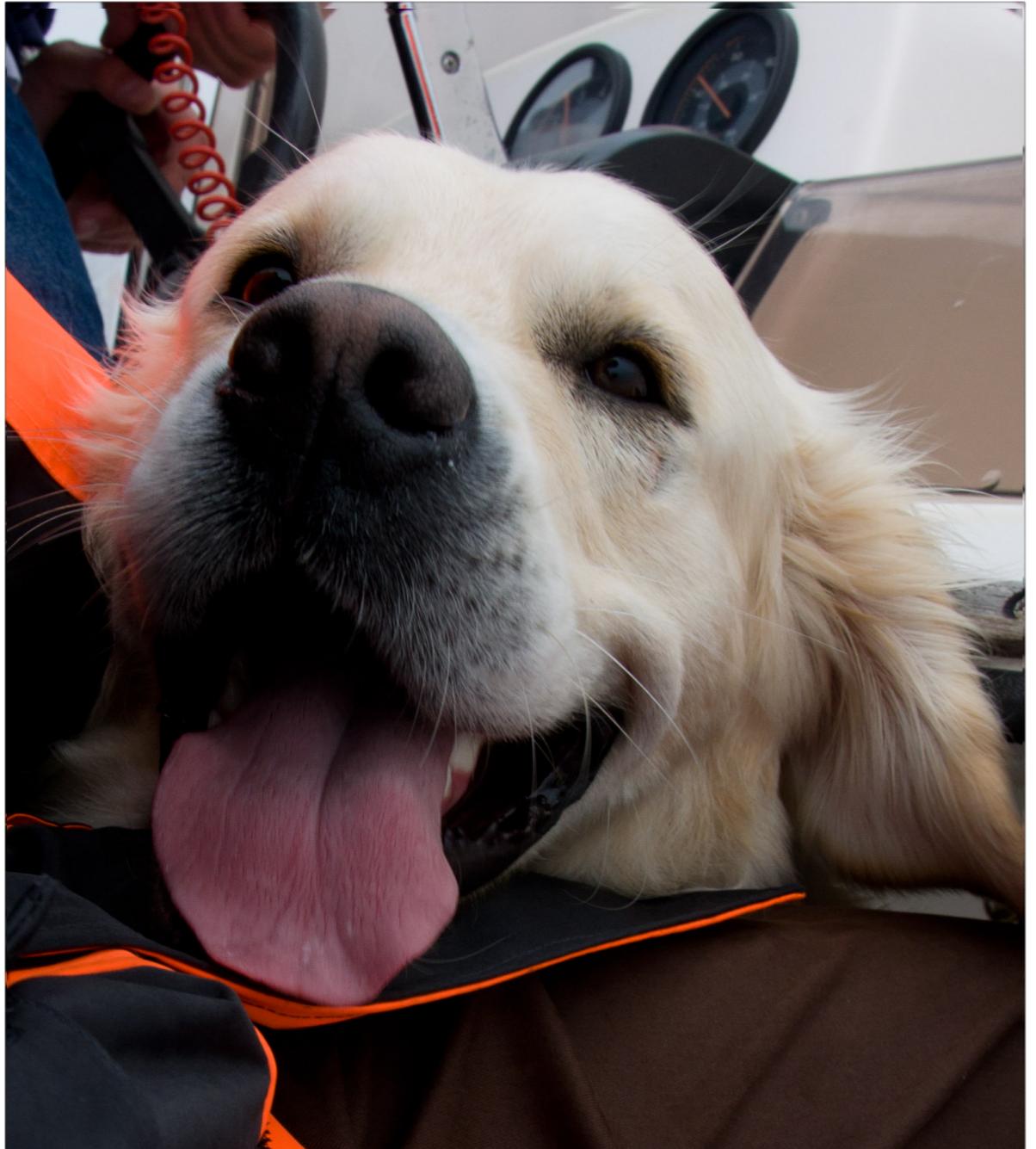
```
public class Person {  
    private String firstName;  
    private String lastName;  
  
    public Person(String firstName, String lastName) {  
        this.firstName = firstName;  
        this.lastName = lastName;  
    }  
  
    public String getFirstName() {  
        return self.firstName;  
    }  
  
    public String getLastname() {  
        return self.lastName;  
    }  
}
```

Java



Cool! Where's my jet-pack?









What's Left Unsaid



Access control

Flow-control

Custom operators

Interface Builder

Optional Chaining

Deeper Obj-C Interop.

