

# Contributing to Swift

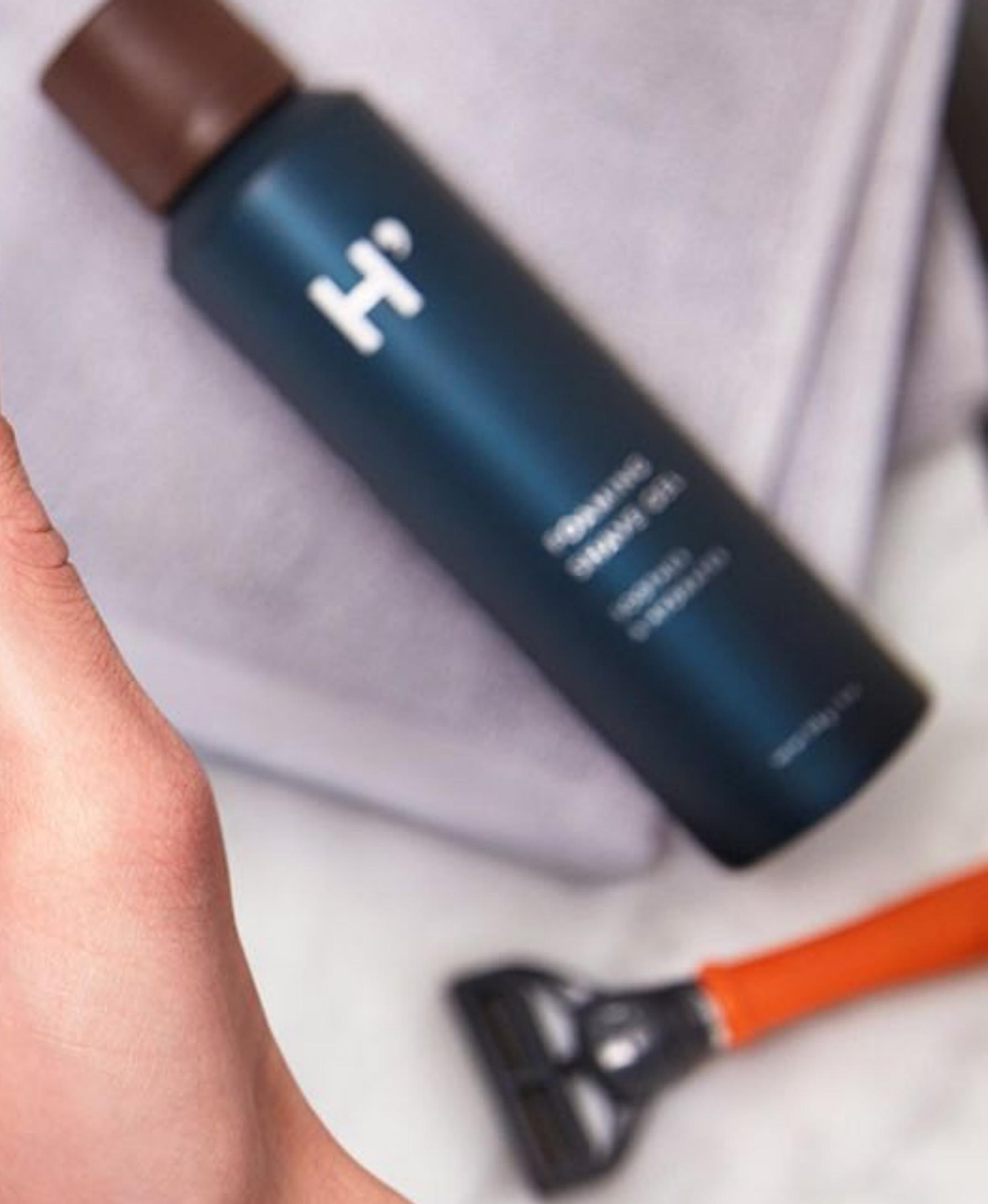
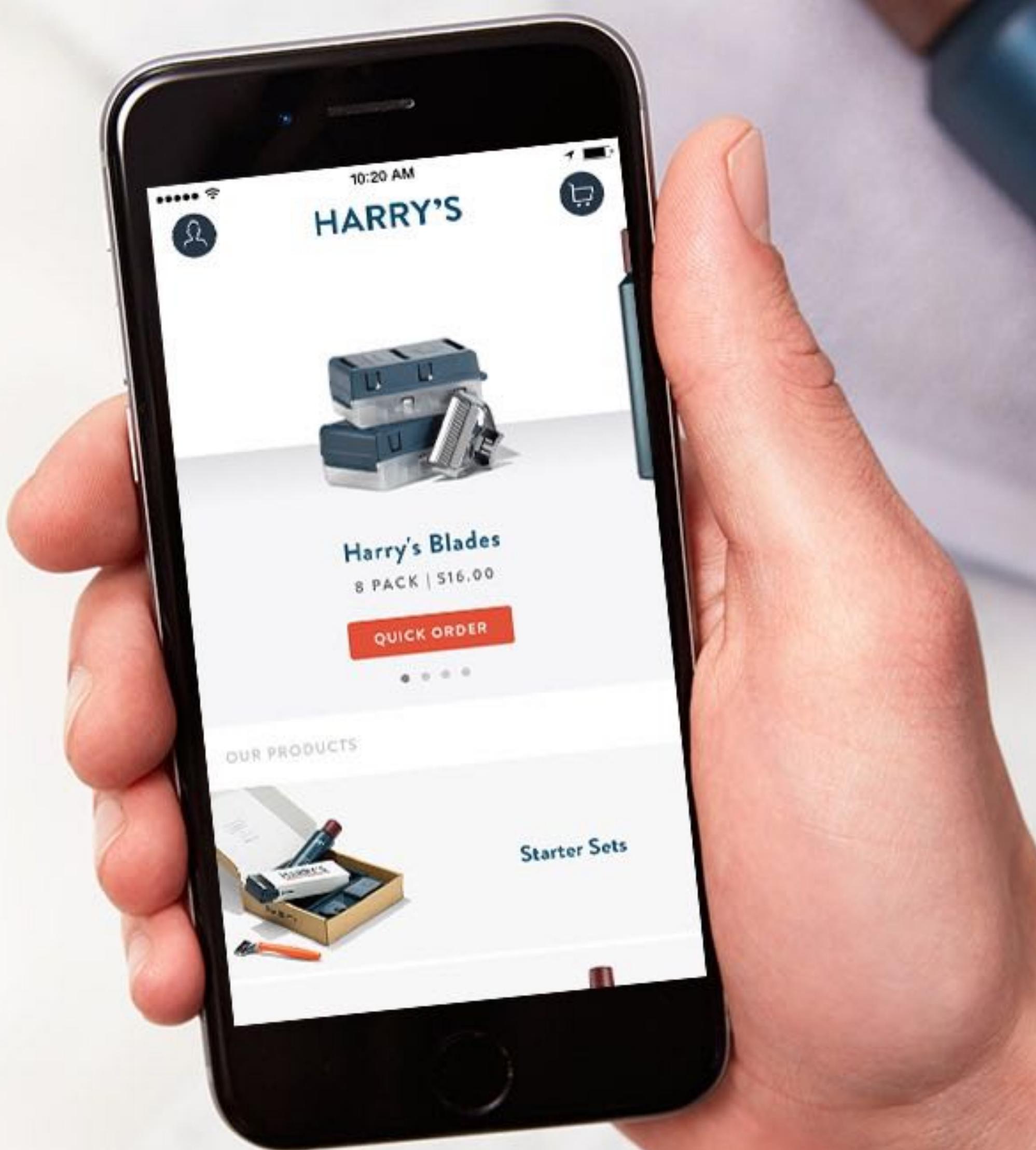
by @neilkimmett



# HARRY'S







```
struct CGPoint {
    var x: Float
    var y: Float
}

struct CGSize {
    var width: Float
    var height: Float
}

struct CGRect {
    var origin: CGPoint
    var size: CGSize
}
```

```
struct CGPoint {  
    static var zero: CGPoint  
}
```

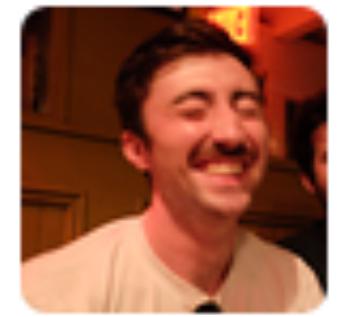
```
struct CGSize {  
    static var zero: CGSize  
}
```

```
struct CGRect {  
    static var zero: CGRect  
}
```

```
let view = UIView()  
  
view.frame.origin = .zero
```

```
struct UIEdgeInsets {  
    var top: Float  
    var left: Float  
    var bottom: Float  
    var right: Float  
}
```

```
let view = UIView()  
view.layoutMargins = .zero
```



**Neil Kimmitt** 2:02 PM



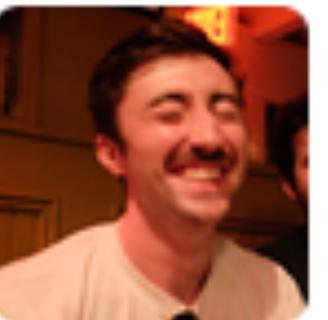
aside: why is there `CGRect.zero` but no `UIEdgeInsets.zero`

2:03 ★ whats that all about



**Soroush Khanlou** 2:03 PM

that's wack



**Neil Kimmett** 2:05 PM

can fix with

```
extension UIEdgeInsets {  
    static var zero: UIEdgeInsets {  
        return UIEdgeInsetsZero  
    }  
}
```

but still

cmon Apple



**Soroush Khanlou** 2:05 PM

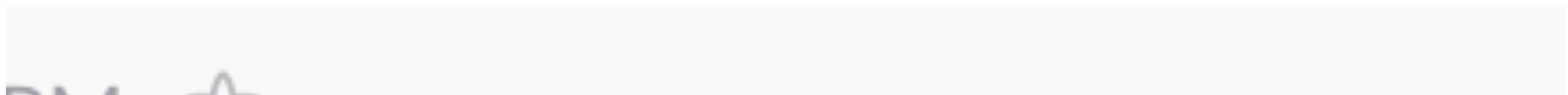
yeah this one's on apple

if you're that kind of person you could make a radar

WPA C07

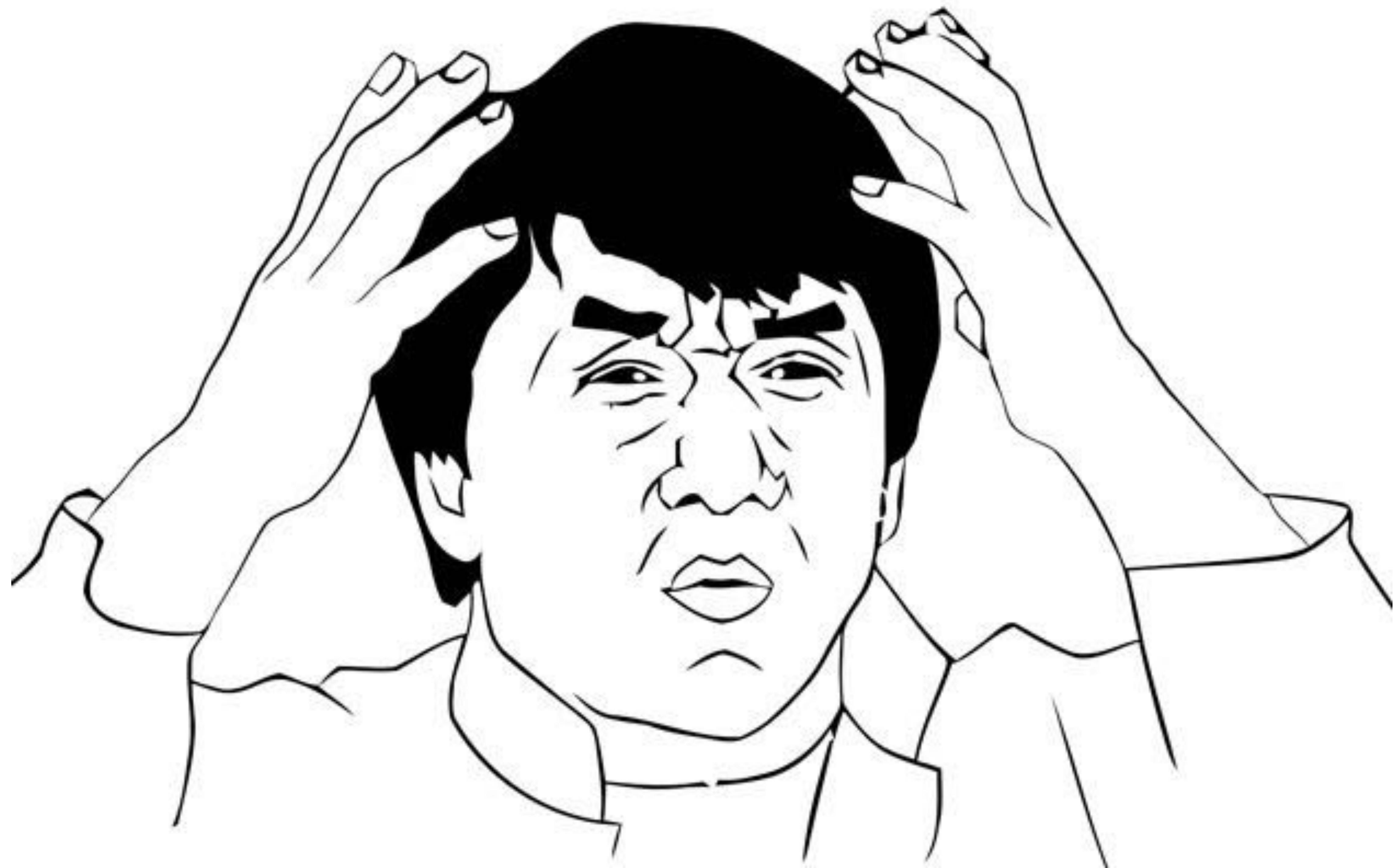
apple

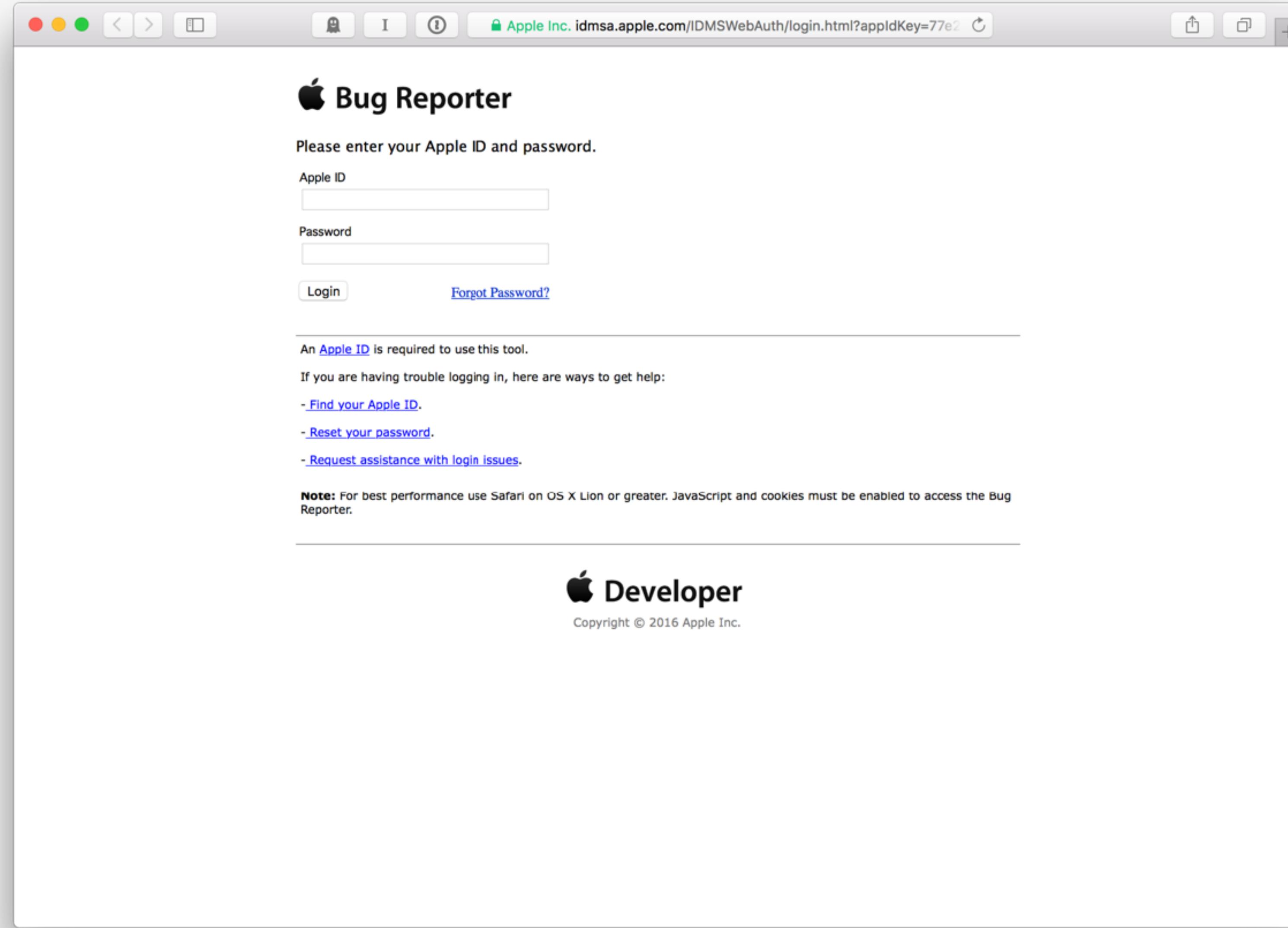
if person you could make a radar



u could make a radar

a radar



A screenshot of a Mac OS X desktop showing a Safari browser window. The window title is "Apple Inc. idmsa.apple.com/IDMSWebAuth/login.html?appIdKey=77e2". The main content is the "Bug Reporter" login page. It features an Apple logo and the text "Bug Reporter". A message says "Please enter your Apple ID and password." Below are two text input fields: "Apple ID" and "Password", each with a corresponding label above it. To the right of the "Apple ID" field is a "Login" button and a "Forgot Password?" link. Below these fields is a horizontal line. Under the line, text states "An [Apple ID](#) is required to use this tool." followed by "If you are having trouble logging in, here are ways to get help:" and three links: "Find your Apple ID.", "Reset your password.", and "Request assistance with login issues.". At the bottom, a note says "Note: For best performance use Safari on OS X Lion or greater. JavaScript and cookies must be enabled to access the Bug Reporter." Another horizontal line follows. At the very bottom is the "Developer" logo with the text "Copyright © 2016 Apple Inc.".

<http://bugreport.apple.com/>

Attention

Open

Hidden

Closed

Archive

Drafts

SEARCH

Results

PRODUCTS

Developer Tools

iOS + SDK

Filter Problem List

Sort by Date

27079603 Xcode suggests

Developer Tools

Rank : No Value

29-Jun-2014

Summary: I use a 3rd party library called PureLayout

18720044 Strikethrough o

iOS + SDK

Rank : No Value

21-Oct-2014

Summary: In order to strikethrough th

16780925 When testing iC

Developer Tools

Rank : No Value

01-May-2014

Summary: This is a duplicate of rdar://16780925. When testing on the iOS Simulator,

## iOS + SDK

[Using Apple Bug Reporter](#)

Please include only one issue per report. All fields are **required** except where noted. While we cannot respond directly to every report, all reports are reviewed by the appropriate engineering teams.

Classification\* Select a type of problem

Reproducibility\* Select how often this problem occurs

Area\*

## Classification

Security

Crash/Hang/Data Loss

Power

Performance

UI/Usability

Serious Bug

Other Bug

Feature (New)

Enhancement

Steps to Reproduce: List the step by step process to reproduce the issue

1. Describe any setup or pre-work (e.g., install Xcode...)
2. Detail the exact steps taken to produce the problem
3. Number each step

Expected Results: Describe what you expected to see

Describe what you expected to happen after completing the steps above.

Actual Results: Describe what you actually saw

Describe what actually happened after completing the steps above.

Applied to whole range



Hide | Revert Save

30-Oct-2014 00:44 AM

and will be closed.

appears in the yellow "Duplicate of XXXXXXXX" section of the bug detail view just under the bug number, title, state, product and rank.

21-Oct-2014 08:04 AM

21-Oct-2014 08:03 AM

21-Oct-2014 08:02 AM

you need to set two attributes on the string.

on the range you want to strikethrough. This bit makes sense.

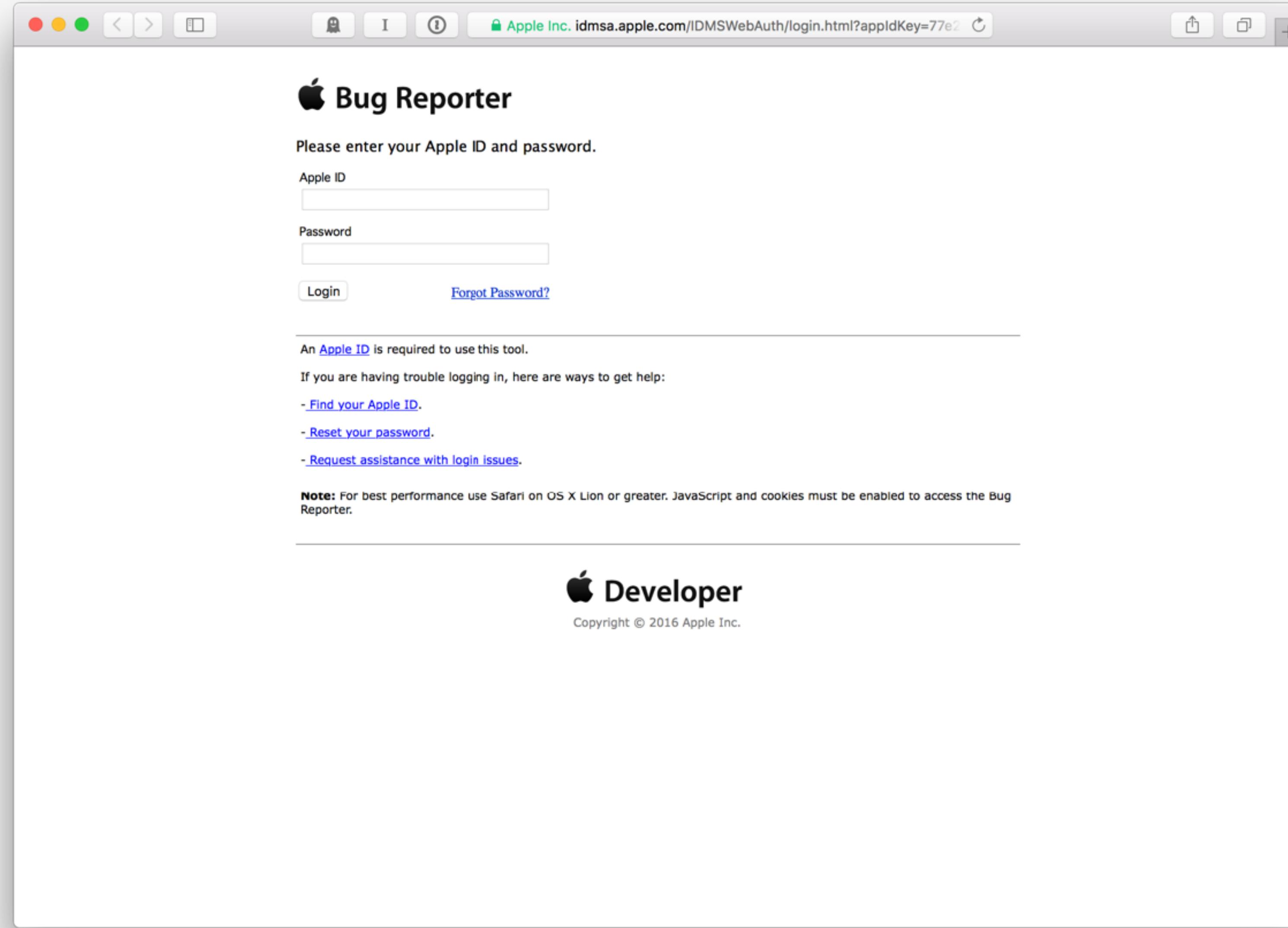
in the range you don't want to strikethrough. This is non-obvious. No I don't want to strikethrough.

lineStyleSingle on some range in the middle of the string (i.e. not

## **Apple Developer Relations**

Engineering has determined that your bug report is a duplicate of another issue and will be closed.

The open or closed status of the original bug report your issue was duplicated to appears in the yellow reporter user interface. This section appears near the top of the right column's bug detail view just und

A screenshot of a Mac OS X desktop showing a Safari browser window. The window title is "Apple Inc. idmsa.apple.com/IDMSWebAuth/login.html?appIdKey=77e2". The main content is the "Bug Reporter" login page. It features an Apple logo and the text "Bug Reporter". A message says "Please enter your Apple ID and password." Below are two input fields: "Apple ID" and "Password", each with a corresponding text entry field. To the right of the "Apple ID" field is a "Login" button and a "Forgot Password?" link. Below these fields is a horizontal line. Under the line, it says "An [Apple ID](#) is required to use this tool." followed by "If you are having trouble logging in, here are ways to get help:" and three links: "Find your Apple ID.", "Reset your password.", and "Request assistance with login issues.". At the bottom, there is a note: "Note: For best performance use Safari on OS X Lion or greater. JavaScript and cookies must be enabled to access the Bug Reporter." Another horizontal line follows. At the very bottom, there is a "Developer" section with the Apple Developer logo and the text "Copyright © 2016 Apple Inc.".

<http://bugreport.apple.com/>



THAT GIVES ME AN IDEA.

A screenshot of a web browser window showing the Swift.org homepage. The address bar at the top reads "Apple Inc. swift.org". The main content area features a large orange header with the word "Swift" and a small orange icon. Below the header, the text "Welcome to Swift.org" is displayed in a large, dark font. A sub-section below it says "Swift is now open source!". To the left of the main content is a vertical sidebar containing several menu items: "ABOUT SWIFT", "BLOG", "DOWNLOAD", "GETTING STARTED", "DOCUMENTATION", "MIGRATING TO SWIFT 3", "SOURCE CODE", "COMMUNITY", "CONTRIBUTING", "CONTINUOUS INTEGRATION", "PROJECTS", "COMPILER AND STANDARD LIBRARY", "PACKAGE MANAGER", "CORE LIBRARIES", and "REPL AND DEBUGGER".

Welcome to Swift.org

Swift is now open source!

We are excited by this new chapter in the story of Swift. After Apple unveiled the Swift programming language, it quickly became one of the fastest growing languages in history. Swift makes it easy to write software that is incredibly fast and safe by design. Now that Swift is open source, you can help make the best general purpose programming language available everywhere.

For students, learning Swift has been a great introduction to modern programming concepts and best practices. And because it is now open, their Swift skills will be able to be applied to an even broader range of platforms, from mobile devices to the desktop to the cloud.

Welcome to the Swift community. Together we are working to build a better programming language for everyone.

*– The Swift Team*

<https://swift.org>

The screenshot shows the GitHub repository page for `apple/swift`. The page includes the repository name, a search bar, navigation links for Pull requests, Issues, and Gist, and a header with Watch (2,223), Star (32,093), Fork (4,418) buttons. Below the header are tabs for Code, Pull requests (99), Pulse, and Graphs. A section titled "The Swift Programming Language" with a link to <https://swift.org/> follows. Key statistics are displayed: 39,327 commits, 21 branches, 36 releases, and 358 contributors. A "Clone or download" button is prominent. The main content area lists recent commits from the `master` branch, starting with a commit from `swift-ci` merged via pull request #3342. Other commits are listed under categories like `.github`, `apinotes`, `benchmark`, etc., with their descriptions and timestamps.

Commit Details	Time Ago
swift-ci committed on GitHub Merge pull request #3342 from practicalswift/typo-fixes-20160705	Latest commit be62dd7 a minute ago
.github Add a missing "the" to the pull request template	8 days ago
apinotes [APINotes] NSPinterFunctionsOptions 0-valued members shouldn't be [].	5 days ago
benchmark [gardening] Use '{let,var} c: C' instead of '{let,var} c : C'	4 days ago
bindings/xml Nesting parameter/returns/throws doc comments for closure parameters	3 months ago
cmake [cmake] Refactor check_working_std_regex into its own cmake file Swif...	8 hours ago
docs [docs] LibraryEvolution: add "let vs. var" as a planned proposal.	4 days ago
include Sema: Explicitly set interface type on all AbstractFunctionDecls	4 hours ago
lib Sema: Explicitly set interface type on all AbstractFunctionDecls	4 hours ago
stdlib Merge pull request #3322 from lucianomarisi/chore/sequence-documentati...	2 days ago
test [gardening] Fix recently introduced typo: "acutall" → "actual"	an hour ago
tools Revert "[cmake] Do not set CMAKE_INCLUDE_CURRENT_DIR."	3 days ago
unittests [SourceKit] Remove dependency of SwiftLang on Core	13 days ago
utils [emacs support] add a missing (require 'compile)	3 days ago

<https://github.com/apple/swift>



[swift](#) / [stdlib](#) / [public](#) / [SDK](#) / [CoreGraphics](#) / **CoreGraphics.swift**

[swift](#) / [stdlib](#) / [public](#) / [SDK](#) / [CoreGraphics](#) / **CoreGraphics.swift**

```
12
13 @_exported import CoreGraphics
14 import Darwin
15
16 //=====
17 // CGGeometry
18 //=====
19
20 public extension CGPoint {
21     static var zero: CGPoint {
22         @_transparent // @fragile
23         get { return CGPoint(x: 0, y: 0) }
24     }
25
26     @_transparent // @fragile
27     init(x: Int, y: Int) {
28         self.init(x: CGFloat(x), y: CGFloat(y))
29     }
30
31     @_transparent // @fragile
32     init(x: Double, y: Double) {
33         self.init(x: CGFloat(x), y: CGFloat(y))
34     }
35
36     @available(*, unavailable, renamed="zero")
37     static var zeroPoint: CGPoint {
38         fatalError("can't retrieve unavailable property")
39     }
40 }
41
```

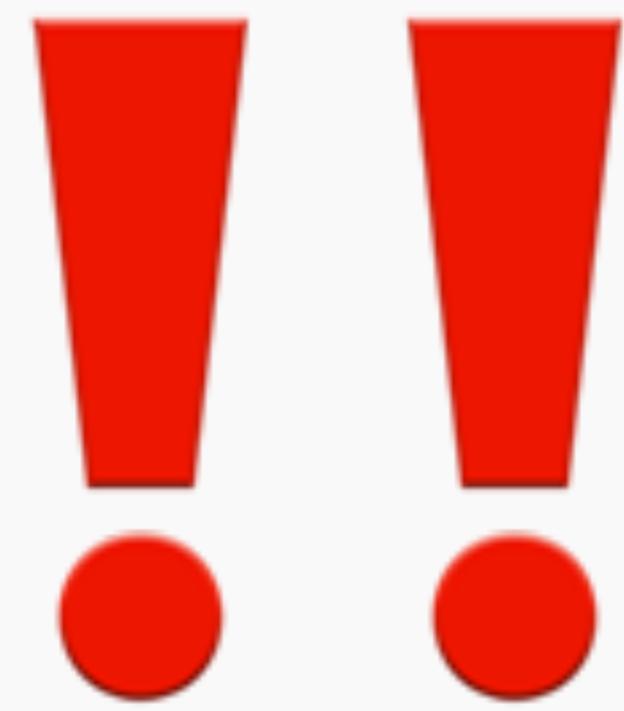
[swift](#) / [stdlib](#) / [public](#) / [SDK](#) / [UIKit](#) / **UIKit.swift**

[swift](#) / [stdlib](#) / [public](#) / [SDK](#) / [UIKit](#) / **UIKit.swift**

```
11 // -----
12
13 import Foundation
14 @_exported import UIKit
15
16
17 //====-
18 // Equatable types.
19 //====-
20
21 @transparent // @fragile
22 @warn_unused_result
23 public func == (lhs: UIEdgeInsets, rhs: UIEdgeInsets) -> Bool {
24     return lhs.top == rhs.top &&
25         lhs.left == rhs.left &&
26         lhs.bottom == rhs.bottom &&
27         lhs.right == rhs.right
28 }
29
30 extension UIEdgeInsets : Equatable {}
31
32 @transparent // @fragile
33 @warn_unused_result
34 public func == (lhs: UIOffset, rhs: UIOffset) -> Bool {
35     return lhs.horizontal == rhs.horizontal &&
36         lhs.vertical == rhs.vertical
37 }
38
39 extension UIOffset : Equatable {}
```

```
13 import Foundation
14 @_exported import UIKit
15
16 +//=====
17 +// UIGeometry
18 +//=====
19 +
20 +public extension UIEdgeInsets {
21 +    static var zero: UIEdgeInsets {
22 +        @_transparent // @fragile
23 +        get { return UIEdgeInsets(top: 0.0, left: 0.0, bottom: 0.0, right: 0.0) }
24 +    }
25 +}
26 +
27 +public extension UIOffset {
28 +    static var zero: UIOffset {
29 +        @_transparent // @fragile
30 +        get { return UIOffset(horizontal: 0.0, vertical: 0.0) }
31 +    }
32 +}
33
```



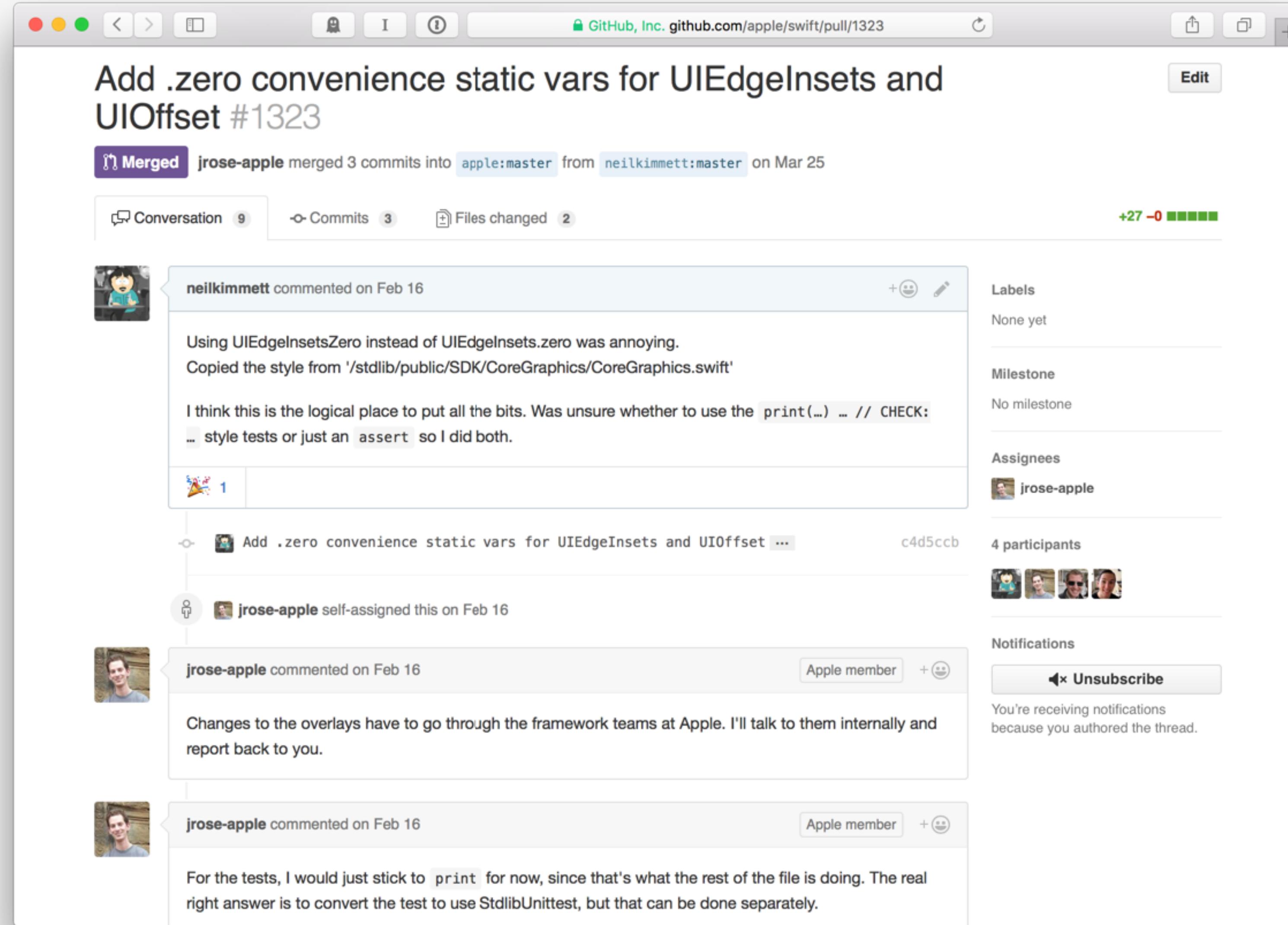


## [swift](#) / [test](#) / [1\\_stdlib](#) / **UIKit.swift**

```
50 var inset1 = UIEdgeInsets(top: 1.0, left: 2.0, bottom: 3.0, right: 4.0)
51 var inset2 = UIEdgeInsets(top: 1.0, left: 2.0, bottom: 3.1, right: 4.0)
52 print("inset1 == inset1: \(inset1 == inset1)")
53 print("inset1 != inset1: \(inset1 != inset1)")
54 print("inset1 == inset2: \(inset1 == inset2)")
55 // CHECK: inset1 == inset1: true
56 // CHECK: inset1 != inset1: false
57 // CHECK: inset1 == inset2: false
58
```

## swift / test / 1\_stdlib / **UIKit.swift**

```
68 +
69 +var inset0 = UIEdgeInsets(top: 0.0, left: 0.0, bottom: 0.0, right: 0.0)
70 +var insetDot0 = UIEdgeInsets.zero
71 +print("inset0 == insetDot0: \(inset0 == insetDot0)")
72 +// CHECK: inset0 == insetDot0: true
73 +
74 +var offset0 = UIOffset(horizontal: 0.0, vertical: 0.0)
75 +var offsetDot0 = UIOffset.zero
76 +print("offset0 == offsetDot0: \(offset0 == offsetDot0)")
77 +// CHECK: offset0 == offsetDot0: true
```

A screenshot of a GitHub pull request interface. The title is "Add .zero convenience static vars for UIEdgeInsets and UIOffset #1323". A purple "Merged" button indicates the pull request has been merged into the `apple:master` branch from the `neilkimmett:master` branch on March 25. The conversation tab shows 9 comments, 3 commits, and 2 files changed. The code review summary shows +27 green and -0 red. The right sidebar displays labels (None yet), milestones (No milestone), assignees (jrose-apple), participants (4), and notifications (Unsubscribe). The main content area shows comments from neilkimmett and jrose-apple discussing the implementation of convenience static vars for UIEdgeInsets and UIOffset.

**Add .zero convenience static vars for UIEdgeInsets and UIOffset #1323**

Merged jrose-apple merged 3 commits into apple:master from neilkimmett:master on Mar 25

Conversation 9 Commits 3 Files changed 2 +27 -0

Labels None yet

Milestone No milestone

Assignees jrose-apple

Participants 4

Notifications Unsubscribe

neilkimmett commented on Feb 16

Using UIEdgeInsetsZero instead of UIEdgeInsets.zero was annoying.  
Copied the style from '/stdlib/public/SDK/CoreGraphics/CoreGraphics.swift'

I think this is the logical place to put all the bits. Was unsure whether to use the `print(...)` ... // CHECK:  
... style tests or just an `assert` so I did both.

jrose-apple self-assigned this on Feb 16

jrose-apple commented on Feb 16

Changes to the overlays have to go through the framework teams at Apple. I'll talk to them internally and report back to you.

jrose-apple commented on Feb 16

For the tests, I would just stick to `print` for now, since that's what the rest of the file is doing. The real right answer is to convert the test to use `StdlibUnittest`, but that can be done separately.

<https://github.com/apple/swift/pull/1323>

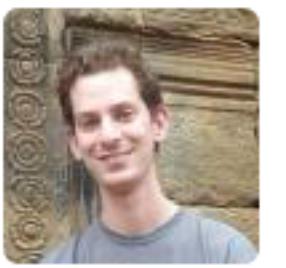


jrose-apple commented on Mar 21

Apple member



Okay, UIKit has approved this change! (Sorry for the turnaround time.) I'll trigger the tests again.



jrose-apple commented on Mar 21

Apple member



@swift-ci Please test



jrose-apple merged commit **d9596ab** into `apple:master` on Mar 25

2 checks passed

[View details](#)

[Revert](#)



jrose-apple commented on Mar 25

Apple member



Thanks, Neil!



```
let scrollView = UIScrollView()  
scrollView.contentInset = UIEdgeInsets.zero
```

UIEdgeInsets zero

What can I do?

Apple Inc. lists.swift.org

## [swift-evolution] Replace throws with Result

Douglas Gregor [dgregor@apple.com](mailto:dgregor@apple.com)  
Wed May 3 15:03:41 CDT 2017

- Previous message: [swift-evolution] Replace throws with Result
- Next message: [swift-evolution] Replace throws with Result
- Messages sorted by: [\[ date \]](#) [\[ thread \]](#) [\[ subject \]](#) [\[ author \]](#)

---

> On May 3, 2017, at 3:31 AM, Jaden Geller via swift-evolution <[swift-evolution@swift.org](mailto:swift-evolution@swift.org)> wrote:  
>  
> To be frank, there's no way this is happening. The rationale for Swift's error handling is documented <<https://github.com/apple/swift/blob/master/docs/ErrorHandling.rst>>, and this system was designed and implemented \*after\* optionals already existed in the language. I don't think there's any evidence that this was a horrible decision, especially one that requires a \*majorly\* source-breaking change to remove.

As a core team member, let me amplify this response a bit: the current error-handling system will not be going away, for the reasons Jaden has stated above.

Maybe we will grow typed throws at some point; maybe Result will get added to the standard library with some nice affordances to map between throwing and Result-producing types; but we're well beyond the point where we can rip out major features that have been generally well-received.

- Doug

> - Jaden  
>  
>> On May 3, 2017, at 3:06 AM, Jose Manuel Sánchez Peñarroja via swift-evolution <[swift-evolution@swift.org](mailto:swift-evolution@swift.org)> wrote:  
>>  
>> There is currently a discussion open about throws, but it seems that the idea would be to add optionals to throws. In my opinion it would be much better to just get rid of all the throw error system.  
>>

<https://lists.swift.org>



A screenshot of a Mac OS X Mail application window. The sidebar on the left shows a list of accounts and mailing lists, with 'swift-users' selected. The main pane displays a list of messages from various users, mostly about Swift development. One message from 'brandonb2019@gmail.com' is highlighted, showing a reply from 'pestov@apple.com'.

Messages listed:

- jeschot@xs4all.nl 5/8/17, 7:39 AM discardable function start? (1)
- kelvin13ma@gmail.com 5/7/17, 1:57 PM Passing value types or members of value types? (10)
- brandonb2019@gmail.com 5/7/17, 8:28 PM Swift build-script Smaller Memory Footprint? (3)
- swift@hpwooten.com 5/5/17, 9:04 AM Compatibility Suite for already failing project (1)
- rmann@latencyzero.com 5/4/17, 7:52 PM Slicing a [UInt8] into a Data without copying? (3)
- dowobeha@gmail.com 5/4/17, 5:12 PM Compiling Swift on Ubuntu to run on Centos (1)
- geojay@gmail.com 5/3/17, 7:15 PM Using .gyb outside of the stdlib (1)

Selected message details:

**From:** brandonb2019@gmail.com **To:** brandonb2019@gmail.com **Date:** May 6, 2017, 10:21:39 PM

Hello,

I'm installing Swift 3 on FreeBSD 11. After installing the necessary dependencies, I now only need to run the build-script.

The problem is that I cannot proceed beyond:

> Linking CXX executable bin/llvm-lto

because I consistently run out of memory on my 16GB server.

(2 direct replies)

**From:** pestov@apple.com **To:** pestov@apple.com **Date:** May 7, 2017, 8:13:10 PM

On Darwin (which uses its own linker and not GNU ld), you can definitely build Swift with less than 16GB of RAM available.

Maybe you could try gold?

Slava

> On May 6, 2017, at 7:21 PM, Brandon B via swift-users<swift-users at swi

<https://stylemac.com/hirundo/>

The screenshot shows a Mac OS X desktop with a browser window open to the Swift Evolution website at [apple.github.io/swift-evolution/](https://apple.github.io/swift-evolution/). The window has a standard OS X title bar with icons for window control, search, and other system functions. The main content area displays the "Swift Evolution" logo (a red square with a white bird icon) and the title "Swift Evolution". Below the title is a search bar with the placeholder "Search" and a magnifying glass icon. To the right of the search bar is a small circular icon with a list symbol. The page lists 176 proposals:

- SE-0176 Enforce Exclusive Access to Memory**  
Author: John McCall  
Review Manager: Ben Cohen  
Status: Active Review  
Scheduled: May 2–8
- SE-0175 Package Manager Revised Dependency Resolution**  
Author: Rick Ballard  
Review Manager: Ankit Aggarwal  
Status: Active Review  
Scheduled: May 2–9
- SE-0174 Change filter to return an associated type**  
Author: Ben Cohen  
Review Manager: Doug Gregor  
Status: Accepted
- SE-0161 Smart KeyPaths: Better Key-Value Coding for Swift**  
Authors: David Smith, Michael LeHew, Joe Groff  
Review Manager: Doug Gregor  
Status: Accepted
- SE-0173 Add MutableCollection.swapAt(\_:\_:)**  
Author: Ben Cohen  
Review Manager: Ted Kremenek  
Status: Implemented  
Implemented In: Swift 4

<https://apple.github.io/swift-evolution/>

A screenshot of a Mac OS X desktop environment showing a web browser window. The window title is "swiftweekly.github.io/issue-68/". The main content of the page is a red header "Swift Weekly Brief". Below it is a paragraph describing the newsletter as a community-driven weekly newsletter about what's happening in the Swift open source projects at Swift.org, curated by Jesse Squires and published for free every Thursday. There are three buttons at the bottom: "Subscribe" (with an envelope icon), "Sponsor" (with a star icon), and "Podcast" (with a microphone icon). A horizontal line separates this from the main content area. In the main content area, the text "Issue #68 04 May 2017" is displayed, followed by "Written by: Bas Broek". Below this, a text block states: "Quite a lot of work has been done to implement recently accepted Swift Evolution proposals, as well as improving their diagnostics and error messages. Interestingly, some of this work has been done by first-time contributors, which is amazing to see!" Further down, there is a link "Interested in sponsoring Swift Weekly Brief? [Learn more here.](#)". Below this, a section titled "Swift Unwrapped" is shown, with a sub-section "Episode 9: [String Manifesto](#)". It mentions that JP Simard and Jesse Squires discuss the String Manifesto. Another section titled "Commits and pull requests" is shown, with a note that Slava Pestov implemented SE-0156: Class and Subtype existentials.

# Swift Weekly Brief

A community-driven weekly newsletter about what's happening in the Swift open source projects at [Swift.org](#). Curated by [Jesse Squires](#) and published for free every Thursday.

[Subscribe](#)  [Sponsor](#)  [Podcast](#)

---

Issue #68 04 May 2017

Written by: [Bas Broek](#)

Quite a lot of work has been done to implement recently accepted Swift Evolution proposals, as well as improving their diagnostics and error messages. Interestingly, some of this work has been done by first-time contributors, which is amazing to see!

Interested in sponsoring Swift Weekly Brief? [Learn more here.](#)

## Swift Unwrapped

Episode 9: [String Manifesto](#)

This week JP Simard and Jesse Squires discuss the String Manifesto!

## Commits and pull requests

Slava Pestov [implemented SE-0156: Class and Subtype existentials](#).

<https://swiftweekly.github.io>

The screenshot shows a web browser window with the URL [spec.fm/podcasts/swift-unwrapped](https://spec.fm/podcasts/swift-unwrapped) in the address bar. The page is titled "Swift Unwrapped" and describes it as "A 30-minute spin off of Swift Weekly Brief and other Swifty news." It features a search bar and navigation links for "About" and "Sponsors". Below the title, there's a thumbnail for the episode "09: String Manifesto" (May 1, 2017), which discusses Swift's String Manifesto. Another thumbnail for "08: Archival Serialization & Swift Encoders" (April 24, 2017) is shown, mentioning SE proposals on serialization & encoding. A third thumbnail for "07: Access Control - The Saga Continues" (April 17, 2017) discusses access control in Swift. At the bottom, there are thumbnails for episodes 06, 05, and 04, though their descriptions are partially cut off.

**Swift Unwrapped**  
A 30-minute spin off of Swift Weekly Brief and other Swifty news.

[Subscribe](#) • [Follow](#) • [Sponsor](#)

**09: String Manifesto**  
May 1, 2017  
We go in way over our heads into Swift's String Manifesto.

[Listen to this Episode](#) • [Download](#)

**08: Archival Serialization & Swift Encoders**  
April 24, 2017  
We unpack the recent SE proposals on serialization & encoding.

[Listen to this Episode](#) • [Download](#)

**07: Access Control - The Saga Continues**  
April 17, 2017  
There has been a ton of debate on the Swift Evolution mailing lists about access control in Swift. We share our thoughts on the situation.

[Listen to this Episode](#) • [Download](#)

**06: Swift 3.1 Release & SwiftPM Improvements**  
April 10, 2017

**05: Objective-C Interoperability**  
April 3, 2017

**04: Bridging with Objective-C**  
March 27, 2017

<https://spec.fm/podcasts/swift-unwrapped>



Joe Groff

**@jckarter**

Jordan Rose

**@UINT\_MIN**

Doug Gregor

**@dgregor79**

Ted Kremenek

**@tkremenek**

Erica Sadun

**@ericasadun**

Brian Gesiak

**@modocache**



**Brian Gesiak**

@modocache

To become one of the top 100 most prolific committers to Swift, you need 15 commits. For LLVM, you'd need 130.

---

RETWEETS

4

LIKES

24



---

5:56 PM - 6 May 2017

Swift Dashboards Projects Issues

FILTERS <>

New filter

Find filters

My Open Issues

Reported by Me

Recently Viewed

All Issues

FAVORITE FILTERS

You must be [logged in](#) to view favorite filters.

All starter tasks Details

Swift Type: All Open, Reopened Unassigned Contains text More Advanced

Label: StarterBug

Order by Created ↓

- SR-4830 Propagate colors from compiler output
- SR-4824 Add tests to check Collection constraint...
- SR-4793 SwiftPM Gives three different error mes...
- SR-4747 Implement binary search for inserting e...
- SR-4716 Better diagnostics for multifile executab...
- SR-4595 Tests generate lots of warnings
- SR-4589 Add --filter option to swift test
- SR-4411 Make SwiftPM store relative paths in pi...
- SR-4405 Add @escaping to RecoverableError.at...

1 of 82 ▲ ▼

Export

SR-4830 Propagate colors from compiler output

Details

Type: Bug Status: OPEN Priority: Medium Resolution: Unresolved Component/s: lldbuild, ... (1) Labels: StarterBug

People

Assignee: Unassigned Reporter: Ankit Aggarwal Votes: 1 Vote for this issue Watchers: 2 Start watching this issue

Dates

Created: 11 hours ago Updated: 11 hours ago

Description

Currently, colors are not shown in compiler output. We probably need to detect if stdout is tty and then force colors out of compiler using some driver flag.

Activity

All Comments History Activity

There are no comments yet on this issue.

<https://bugs.swift.org>

The screenshot shows the GitHub repository page for `apple/swift`. The page includes the repository name, a search bar, navigation tabs for Pull requests, Issues, and Gist, and a header with Watch (2,223), Star (32,093), Fork (4,418) buttons. Below the header are tabs for Code, Pull requests (99), Pulse, and Graphs. A banner at the top says "The Swift Programming Language <https://swift.org/>". Key statistics are displayed: 39,327 commits, 21 branches, 36 releases, and 358 contributors. A "Clone or download" button is prominent. The main content area lists recent commits from various contributors, such as `swift-ci`, `practicalswift`, and `lucianomarisi`, with details like commit messages and timestamps.

Commit Details	Time Ago
<code>swift-ci</code> committed on GitHub Merge pull request #3342 from practicalswift/typo-fixes-20160705	Latest commit be62dd7 a minute ago
<code>.github</code> Add a missing "the" to the pull request template	8 days ago
<code>apinotes</code> [APINotes] NSPointerFunctionsOptions 0-valued members shouldn't be [].	5 days ago
<code>benchmark</code> [gardening] Use '{let,var} c: C' instead of '{let,var} c : C'	4 days ago
<code>bindings/xml</code> Nesting parameter/returns/throws doc comments for closure parameters	3 months ago
<code>cmake</code> [cmake] Refactor check_working_std_regex into its own cmake file Swif...	8 hours ago
<code>docs</code> [docs] LibraryEvolution: add "let vs. var" as a planned proposal.	4 days ago
<code>include</code> Sema: Explicitly set interface type on all AbstractFunctionDecls	4 hours ago
<code>lib</code> Sema: Explicitly set interface type on all AbstractFunctionDecls	4 hours ago
<code>stdlib</code> Merge pull request #3322 from lucianomarisi/chore/sequence-documentati...	2 days ago
<code>test</code> [gardening] Fix recently introduced typo: "acutall" → "actual"	an hour ago
<code>tools</code> Revert "[cmake] Do not set CMAKE_INCLUDE_CURRENT_DIR."	3 days ago
<code>unittests</code> [SourceKit] Remove dependency of SwiftLang on Core	13 days ago
<code>utils</code> [emacs support] add a missing (require 'compile)	3 days ago

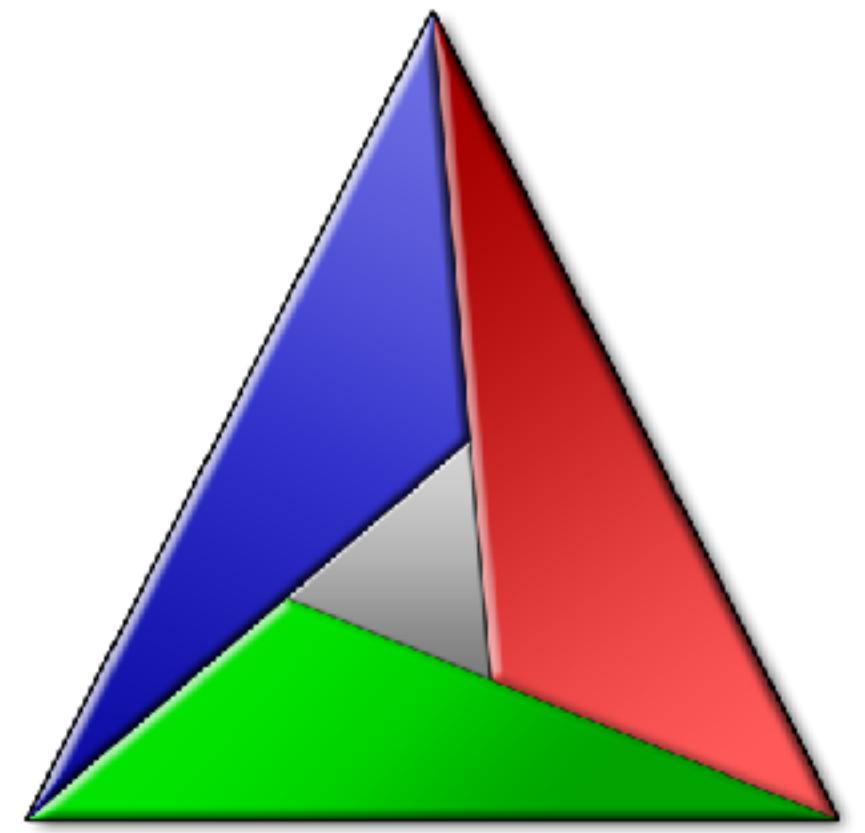
<https://github.com/apple/swift>

```
mkdir swift-source
```

```
cd swift-source
```

```
git clone https://github.com/apple/swift.git
```

```
./swift/utils/update-checkout --clone
```



**CMake**  
*Cross-platform Make*

Ninja



```
brew install cmake ninja
```

utils/build-script -h

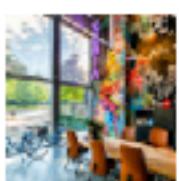


A woman with dark hair tied back is speaking on stage. She is wearing a black long-sleeved shirt and a grey wristband on her left wrist. She is gesturing with her right hand raised. The background is dark.

▶ ▶ | 16:30 / 38:26

CC HD M Q E

### Do iOS 2016 - Ayaka Nonaka, Contributing to Swift



Xebia Group International



326

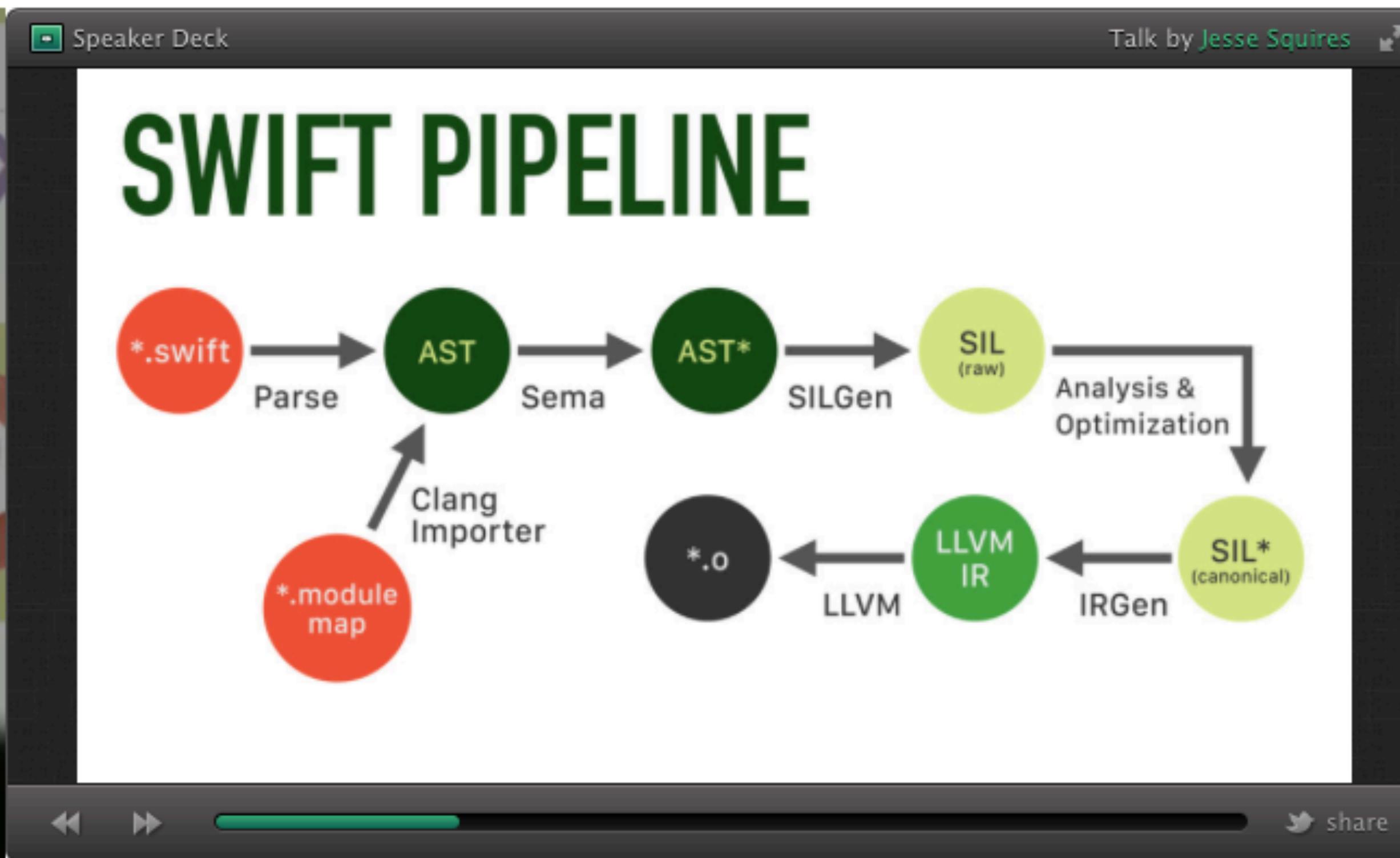
659 views



Add to Share More

17 0

<https://www.youtube.com/watch?v=GnT2ZeHVJe4>



## Jesse Squires - Contributing to Open Source Swift

<https://news.realm.io/news/tryswift-jesse-squires-contributing-open-source-swift/>



Speaker Deck

Talk by Realm

# CONTRIBUTING TO SWIFT

Russ Bishop  
My views are my own and are not endorsed by my employer in any way.

share

A screenshot of a presentation slide titled "CONTRIBUTING TO SWIFT". The slide has a blue background. On the left, there is a portrait of Russ Bishop, a man with glasses and a red and blue checkered shirt, speaking into a microphone. On the right, the title is displayed in large yellow capital letters. Below the title, there is a bio for Russ Bishop, stating "My views are my own and are not endorsed by my employer in any way." At the bottom, there are standard presentation controls for navigating through the deck.

## Russ Bishop - Contributing to Swift: From Proposal to Shipped

<https://news.realm.io/news/slug-russ-bishop-contributing-open-source-swift-proposal/>

No-one has any idea  
what they're doing





# Thanks!

from @neilkimmett