



Debugging and Xcode Tools

Diving Deeper

Olla Ashour, Senior Software Engineer
olla.ashour@andela.com

March, 2020



Importance?

- **Familiarity with codebase**
- **Be faster, re-compilation is expensive**
- **Edge Cases**
- **Learn new tips/tricks/low-level**



Our GOALS

- **Advanced debugging techniques**
- **Build time enhancement techniques**
- **Xcode tips**

Advanced Debugging

Let's make our life easier :D



Bugs we are fixing*

- ❑ Pull down to refresh **does not** reload new posts.
- ❑ The user is **allowed** to load posts **more than 3** times.
- ❑ The left navigation bar label that indicates how many times the user did load posts is **not** being **updated**.

* Credits to **Fady Derias** /SSE Engineer @ Andela



Advanced Debugging

- ❑ Breakpoints Recap
- ❑ Expressions
- ❑ Watchpoints
- ❑ Symbolic Breakpoints
- ❑ Other Advanced LLDB

Build Time Enhancements

Being more effective :D



Build Time Enhancements

❏ Build with Timing Summary

- ❏ `CompileSwiftSources`
- ❏ `PhaseScriptExecution`

❏ Identify code that compiles slowly

- ❏ `-Xfrontend -warn-long-expression-type-checking=<limit>`
- ❏ `-Xfrontend -warn-long-function-bodies=<limit>`

❏ Swift dependencies



Build Time Enhancements

Type Checking

The screenshot shows the Xcode Build Settings interface. The 'Build Settings' tab is selected. Under the 'Swift Compiler - Custom Flags' section, the 'Other Swift Flags' are configured. The 'Debug' configuration has a tooltip showing the flags: `-Onone -Xfrontend -warn-long-function-bodies=100 -Xfrontend -warn-long-expression-type-checking=100`. The 'Release' configuration is currently empty.

Setting	App
▼ Other Swift Flags	<Multiple values>
Debug	<code>-Onone -Xfrontend -warn-long-function-bodies=100 -Xfrontend -warn-long-expression-type-checking=100</code>
Release	



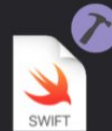
Build Time Enhancements

Swift dependencies

```
struct Point {  
    var x, y: Double  
    init(x: Double, y: Double) {  
        assert(x.isFinite)  
        assert(y.isFinite)  
        self.x = x  
        self.y = y  
    }  
}  
  
struct PathSegment {  
    var start, end: Point  
}
```



```
let point = Point(x: 3.0, y: 4.0)  
  
let distance =  
    sqrt(point.x * point.x +  
        point.y * point.y)
```



Xcode Tips

D: بمزاج Let's enjoy debugging



Xcode Tips

- ❏ Scheme Diagnostics

- ❏ Behaviors

 - ❏ Debug Tab

 - ❏ Custom Behaviour



When debugging, novices **insert corrective code**;
experts **remove defective code**.

— Richard Pattis

Thank YOU

Happy Debugging :D

Brilliance is evenly **distributed**, **opportunity** is not!



Andela

Interested in joining Andela, **Apply here:**
<https://andela.com/JoinAndelaEgypt>



References

- [WWDC 2018](#)
- [Xcode and LLDB Advanced Debugging Tutorial: Part 1 - Fady Derias](#)
- [Advanced Guide Using Breakpoints](#)
- [Behaviors](#)
- [Advanced Debugging with Xcode](#)
- [Analyzing Build Performance](#)
- [Improving Your Build Time in Xcode 10](#)
- [Apple Guide's to LLDB](#)



References

- [Why Debugging is a Learning Opportunity For Developers?](#)
- [Advanced Debugging with Xcode - Headout Engineering](#)
- [facebook/chisel: Chisel is a collection of LLDB commands to assist debugging iOS apps.](#)