202: Swift (Intro)

Slide: part 1

This talk is about Swift closures, generics, and enums. In other words: the good stuff.

These three features – closures, generics, and enums – are what makes most of Swift possible. They are really what sets Swift apart from Objective-C, and what makes it a truly modern language.

Slide for this?

For example, without enums we wouldn't have optionals. Under the hood, optionals are simply enums and I'll show you how this works in the demo.

Without generics we wouldn't have typed arrays and dictionaries.

Type-safety is a very important feature of Swift, and generics make it possible to have strongly typed arrays. This is unlike NSArray from Objective-C, which doesn't know anything about the type of the objects it holds and is therefore not type-safe at all.

Without closures you wouldn't be able to do any of the functional style of programming that's becoming more and more popular these days.

If you've done any Objective-C programming you're no doubt familiar with blocks. A closure is pretty much the same thing as a block, but with much simpler syntax. So you won't need to visit this site anymore:

goshdarnblocksyntax.com

(You may know this site under a slightly different name.)

Slide: Talk Overview (Tic-Tac-Toe screenshot)

This is what we're going to do in the next hour or so:

In the demo I'll show how closures, generics, and enums work in a playground. So you can follow along and mess about with these things.

In the lab you'll refactor some parts of the Tic-Tac-Toe game using closures and generics.

For those of you who didn’t attend the previous talk, what we learned there was to build a basic Tic-Tac-Toe game using Swift. In this talk you’ll add some new stuff to that game.

In particular, you’ll make your own generic container type. This works like array and dictionary and so on, but is better suited to board games like Tic-Tac-Toe. And because it’s generic, you can reuse it in other games. You’ll also make the win-detection code a lot shorter, and again reusable in other board games.

And finally, in the challenges you’ll replace the game’s delegate protocol with a closure, and you’ll extend the Player enum so that it gives the player an avatar image. This builds on some of the things we’ll do in the playground, so pay good attention. ;-)

Slide: part 2

OK, let’s get started because I have a lot of stuff to show you.

[3:00]