IS 543 Fall 2023

Mobile Platform Development

Review

Access Control

Assign Project 1

Dr. Stephen W. Liddle
782 TNRB, office hours Tu 9:30-10:30am, Th 2-3pm
is543@byu.edu
801-422-8792 (rings on my cell)
Zoom: https://bit.ly/liddlezoom

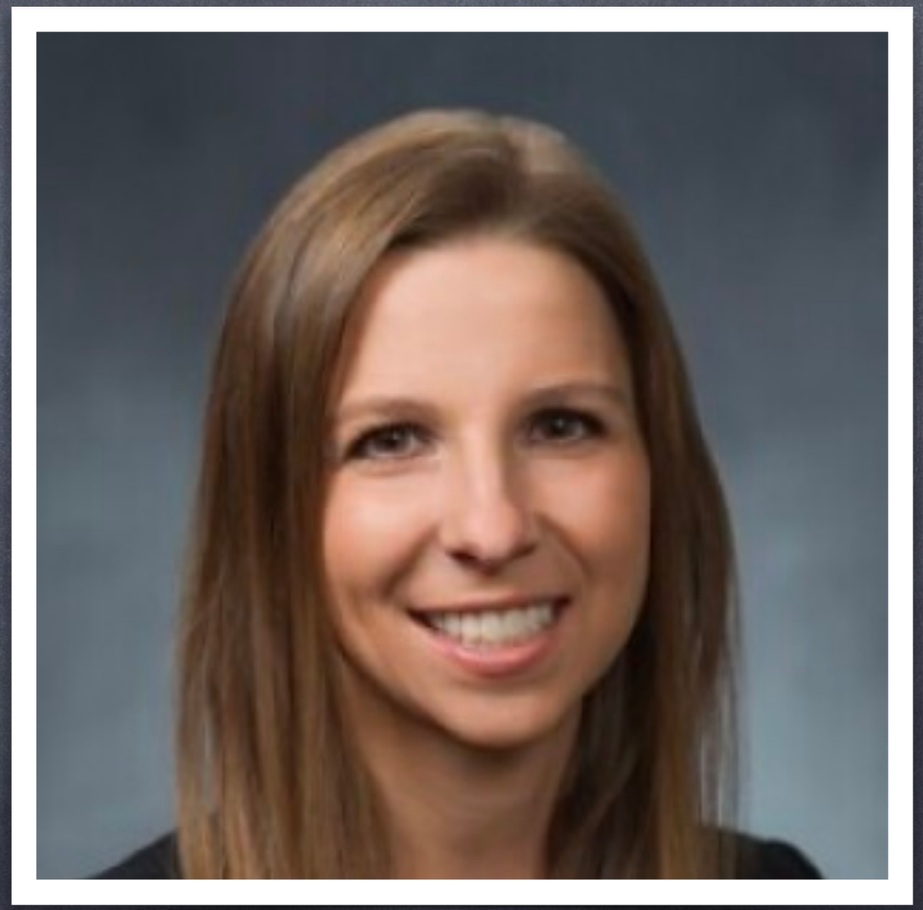
Today

- Debrief HW7
  And do Q&A about the Calculator
- Review the Swift concepts we've explored so far
  And do Q&A about them
- Access Control
- Assign Project 1

Today's Devotional

- Dr. Abigail Allen
  Professor of Accountancy

  I guarantee it will be worthwhile



Fun Facts

- Which signs of obsessive-compulsive personality disorder does Professor Liddle exhibit?

Preoccupation with rules	Some
Preoccupation with orderliness	Some
Preoccupation with control	Some
Over-devotion to work	Some
Not being able to throw things away (even when objects have no value)	Some
Lack of flexibility	A bit
Lack of generosity	Not usually
Not wanting to allow other people to do things	Some
Preoccupation with details, rules, and lists	Some

What I do see easily is anything that's out of place: grammar, punctuation, poorly formatted or inconsistent code, improper English, etc.

HW7 Debrief

- What did you learn from HW7?

  How long did it take you?

  What questions did it generate?

  What main points did you pick up from the Calculator app demo?

  How are you feeling about MVVM now?

Let's Review

- Go to the Swift book and look at the chapter headings

The Basics
Basic Operators
Strings and Characters
Collection Types
Control Flow
Functions
Closures
Enumerations
Structures and Classes
Properties
Methods
Subscripts
Inheritance
Deinitialization
Optional Chaining
Error Handling
Concurrency
Macros
Type Casting
Nested Types
Extensions
Protocols
Generics
Opaque and Boxed Types
Automatic Reference Counting
Memory Safety
Access Control
Advanced Operators

Time for Q&A: which of these concepts do you need to know better?

Access Control

- Five levels of access control in Swift

  `open`: can subclass/override entity in a module that imports this module (only applies to classes)

  `public`: entity can be used in a module that imports this module

  `internal`: entity can only be used within the same module (this is the default access level)

  `fileprivate`: restricts the use of an entity to its own defining source file

  `private`: restricts the use of an entity to the enclosing declaration, and to extensions of that declaration that are in the same file

  (When you create a SwiftUI app, the default is that all your code lives in one "module")

- Common access levels

  Most common is the default, `internal`

  Second most common is `private`

- You can also restrict a property for read only

```
private(set) var sum: Int    // The property's setter is private, but not its getter
fileprivate(set) var x: Int  // The property's setter is fileprivate, not its getter
```

Let's apply these ideas in the Calculator project...

Updating "Counts for Kyle"

- The standard rule is max 50 pushups per day
- I mentioned previously that my neighbor, Richard, agreed that I could count 100 pushups across two days
- I showed you my current algorithm:
  Filter the array to dates that count for Kyle
  Then sum min(50, count) for each date that counts
- How do we modify this algorithm for the new rule?
  Divide into groups of two or three and devise an approach

Assign Project 1

- We're going to implement a game of Set

  First step is to go through the Concentration Game videos I've posted to replace next week's lectures while I'm gone

  It will show you lots of little techniques that will help you on Project 1
  - Like using the custom ".cardify()" ViewModifier
  - And showing how to animate various UI changes (though your animation for Project 1 is simpler than the Concentration game animation)
  - How to draw custom shapes