GENERAL ASSEMBLY

MOBILE DEVELOPMENT

PRESENT VIEWS IN CODE, ARRAYS, TABLE VIEWS

Tedi Konda
Executive Director, Technology, Unison

Learning Objectives

- Present views programmatically
- Identify arrays in Swift
- Implement gestures programmatically
- Explore table views and add data programmatically
- Explore dictionaries and compare arrays (if there is time, otherwise TBC)

PRESENT VIEWS IN CODE, ARRAYS, TABLE VIEWS

GESTURES

GESTURES

- Tap
- Swipe
- Pinch
- Pan
- Edge pan
- Long press
- Rotate
- Can add in interface builder or code

SHOW VIEW CONTROLLERS WITH LOGIC

- Create a segue between two controllers
- Give it an identifier
- At some point in your view controller, call performSegueWithIdentifier using that identifier

CODE DEMO: PRESENT VIEWS IN CODE

PAIR EXERCISE

- Complete to-dos 1-4 on ViewController.swift (located under Assessments -> Week 4 -> Segues and Tables

ARRAYS

ARRAYS

- Arrays have a few interesting properties
  - They contain things (we'll call them **elements**)
    - Arrays can also be empty
  - Each element has an index
    - Indexes start at 0
  - The array has a **count** of elements
  - Arrays have order, and can be iterated over in order
  - Looking up an element by index is **fast**

ARRAYS SYNTAX

- Creating an array
  - var array = [1, 2, 3] // Type is inferred if the array is populated
  - var array: [Int] = [] // Must declare type if array is empty
  - let array = [1, 2, 3] // Array constants cannot be modified
- Accessing an array
  - for i in [1, 2, 3] { /* This loops three times. i is first 1, then 2, then 3. */ }
  - for (index, element) in enumerate([“hi”, “there”, “class!”]) /* Loops three times, index is 0, 1 then 2.

    Element is “hi”, “there” then “class!” */
  - let firstElement = array[1] // We can access elements by index using this syntax

TABLE VIEWS

TABLE VIEWS

- Table views are a one dimensional list
  - Vocabulary:
    - Section: All table views contain multiple sections
    - Row: Every section has a number of rows, which are entries in that section
    - Index path: The combination of a section and row that is a unique entry in a table view
    - Cell: The view that is displayed for an index path (the class UITableViewCell is a subclass of UIView)
- Table views must have a number of sections, a number of cells in each section, and (optionally), the cells themselves
- Table views have a data source and a delegate
  - Data source: Provides cells, number of cells and sections
  - Delegate: Gets called when things happen to the table view, provides some views (e.g. header and footer)

TABLE VIEWS DEMO

PAIR EXERCISE

- Complete to-do 5 and 6 on ArrayTableViewController.swift (located under Assessments -> Week 4 -> Segues and Tables

PRESENT VIEWS IN CODE, ARRAYS, TABLE VIEWS

DICITONARIES

WHAT IS A DICTIONARY?

A dictionary has a unique set of **keys**. Each of those keys is unique in the dictionary

- Each key has a **value**, which can be quickly referenced if you have the **key**
  - Values do not have to be unique in the dictionary
- Storage: ages[“tedi”] = 30
- Retrieval: if let tediAge = ages[“tedi”] {/* if ages[“tedi”] exists, this is run */}
- Also referred to as **maps**

WHAT IS A DICTIONARY?

- We use dictionaries when there is an association between one thing and another
- You r**eally really should** query a dictionary for a value when you already have

the key

- Looking up values for keys in dictionaries is **fast**

DICTIONARY SYNTAX

- Creating a dictionary with values: var ages = [“tedi”:30] // Type is [String: Int]
- Creating an empty dictionary: var ages: [String: Int] = [:]
- Creating a constant: let ages = [“tedi”:30]
- Accessing: let tediAge = ages[“tedi”] // tediAge is an Int? with value 30
  - Hint: This is a great chance to use ‘if let’!
- Setting: ages[“thomas”] = 43