SPLIT-SCREEN, SEGUES
Programming Handheld Systems—iOS

CMSC 436
Spring 2019

TODAY

• Split-screen Navigator
• Segues
• Demo, showing
  • layout constraints
  • nav controller
  • split-screen controller
  • segues
  • URL class
NAVIGATION

- Your view controller has the following properties:
  - `var viewControllers: [UIViewController]? { get set }
  - // in order (left-to-right (tab), master-detail (split), root etc. (nav)

- References to enclosing view controllers
  - `var tabBarController: UITabBarController? { get }
  - `var splitViewController: UISplitViewController? { get }
  - `var navigationController: UINavigationController? { get }

- Pushing/popping MVC from a UINavigationController
  - func pushViewController(_ vc: UIViewController, animated: Bool)
  - func popViewController(animated: Bool)

- But rarely done. Instead:
  - `seguess

NAVIGATION CONTROLLER

Pushes and pops off of a stack. “Pushed” view controllers are always brand new, and are destroyed as they are popped off.
SEGUES

- For navigation and split-screen controllers, we use **segues**:  
  - show segue: overlays a new MVC  
  - show detail segue: same, but shows detail  
  - modal segue: takes over whole screen, already shown  
  - popover segue: little window  
- **Important: segues**  
  - always create **new instances** of MVC  
  - always thrown away when we go Back (not a segue)
**SplitViewController**

- Side-by-side MVCs:
  - Just show detail in portrait, but
    - can pull out the master for side-to-side
    - None of this works on smaller phones, even Xr/X
      - so we wrap the master in nav controller
      - and then split-view does it’s thing on iPad or iPhone
SPLITSCREEN STORYBOARD

SEGUES

- How do we use the segue identifier?
  - programmatic segues (rare)
    - ctrl-drag from MVC to MVC
    - perform the segue programatically
    - sender can be anything

```swift
func performSegue(withIdentifier: String, sender: Any?) {
    // ...
}
```
SEGUES

• prepare-ing for a segue

```swift
    func prepare(for segue: UIStoryboardSegue, sender: Any?) {
        if let identifier = segue.identifier {
            switch identifier {
            case "moon":
                if let segueVC = segue.destination as? PicController {
                    segueVC.this = "blather"
                    segueVC.this = 42
                }
            case "orion":
                ...
            default: break
            }
        }
    }
```

• Destination is always brand-new VC, `outlets not yet set`
• `sender` is button or control that caused the segue

SEGUES

• Can prevent segue from happening:

```swift
    func shouldPerformSegue(withIdentifier identifier: String?,
        sender: Any?) -> Bool {
        return false
    }
```
URL

- Local through (usually the “main”) bundle
  - URL = Bundle.main.URL(forContent: “pretty”, withExtension: “jpg”)

- Remote through normal URL
  - URL = URL(string: “http://kelehers.me/orion.jpg”)