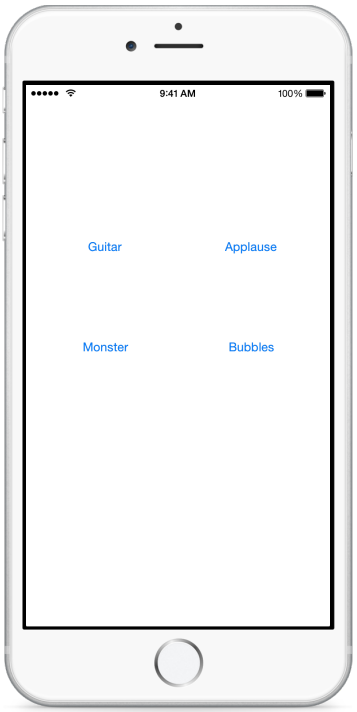
NoiseMaker
Lesson 9

Description

Assign values to each button **Tag** attribute, and bind each button to a single controller action.



Learning Outcomes

- Compare different solutions to a particular programming problem.
- Practice using the Attributes Inspector and Connections Inspector to modify view attributes and connections.
- Discover the **Tag** attribute of view elements, and apply the tag property to distinguish one interface element from another in code.

Vocabulary

`UIButton`	`UIView`	`tag` property
Attributes Inspector	Tag attribute	controller action
Connections Inspector	parameter	argument

Materials

- **NoiseMaker Lesson 9** Xcode project

Opening

Can you think of a way we can connect each button to a single controller action that plays a different sound depending on which button is tapped?

Agenda

- Discuss how one might connect the four buttons to one `ViewController` method that uses an `if-else` statement to determine which button was tapped.
- Discuss how each `ViewController` action `sender` argument could be compared to `UIButton` objects that are `ViewController` outlet properties.
- Discuss how the above approaches would increase the amount of code, yielding little benefit.
- Using the Xcode Documentation and API Reference (⇧⌘0), view the `UIButton` class reference and notice that it descends from `UIView`.
- Using the Xcode Documentation and API Reference (⇧⌘0), view the `UIView` class reference and observe the `tag` property.
- Using Interface Builder, select a button and view the Attributes Inspector (⌥⌘4).
- Discuss the **Tag** attribute in the **View** section of the Attributes Inspector (⌥⌘4).
- Using the Attributes Inspector (⌥⌘4), assign each button a **Tag** value that corresponds to the `AVAudioPlayer` indices in the model `players` array property (e.g., Guitar is **0**, Applause is **1**, Monster is **2**, Bubbles is **3**).
- Using the Connections Inspector (⌥⌘6), delete each button's connection to the respective controller action.
- Using the Assistant Editor (⌥⌘↩), replace the four controller actions with one new `playSound:` action.

```
@IBAction func playSound(sender: UIButton) {
   // play the right sound
}
```

- Using Interface Builder and the Assistant Editor (⌥⌘↩), Control-drag from each button to the single `playSound:` method to establish action connections.
- Implement the `playSound:` method to call the `NoiseMaker` model `play:` method, passing the `sender` parameter `tag` property value as the argument.

```
@IBAction func playSound(sender: UIButton) {
   noiseMaker.play(sender.tag)
}
```

- Explain that the value entered for the **Tag** attribute in Interface Builder is accessible via the `tag` property.
- Run the app (⌘R), tap the buttons and verify that the sounds still play.
- Discuss the significant reduction of code in both the model and controller.

Closing

How else might we customize our button attributes, especially to make them look more fun?

Modifications and Extensions

- Analyze the changes necessary if one were to add an additional button and sound to the app. Add another button and audio file to the project. Describe the subtle dependencies between the user interface and the model, and summarize what code needed to change.
- Consider how the size of the model's `AVAudioPlayer` array is coupled to the number of buttons and the corresponding `tag` values. Take a programmatic approach to creating the view, generating buttons depending on the size of the `NoiseMaker players` array.

Resources

Managing a User Interface Object's Connections https://developer.apple.com/library/ios/recipes/xcode_help-IB_connections/chapters/Connections.html

UIKit User Interface Catalog: Buttons https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/UIKitUICatalog/UIButton.html

UIButton Class Reference https://developer.apple.com/library/ios/documentation/UIKit/Reference/UIButton_Class/index.html

UIView Class Reference https://developer.apple.com/library/ios/documentation/UIKit/Reference/UIView_Class/index.html

Creating an Action Connection https://developer.apple.com/library/ios/recipes/xcode_help-IB_connections/chapters/CreatingAction.html

Interface Builder Help: Configuring Object Attributes https://developer.apple.com/library/ios/recipes/xcode_help-IB_objects_media/Chapters/ObjectAttributes.html