UnitConverter
Lesson 1

Description

Use Interface Builder to create the user interface with a label and picker view.

Learning Outcomes

- Predict the amount of user activity based on the choice of user interface components, and compare different mechanisms of user input.
- Apply Interface Builder to create a user interface.
- Practice using Auto Layout constraints to create adaptable user interfaces.
- Recognize the picker view interface component and plan user interaction with the interface.

Vocabulary

user experience	usability	Interface Builder
IB Object Library	Attributes Inspector	constratin
Pin control	Align control	Picker View
Assistant Editor		

Materials

- **UnitConverter Lesson 1** Xcode project

Opening

The user of our app needs to provide a temperature. How many taps would be involved if our interface consisted of a text field for input?

Agenda

- Discuss the main requirement of the application: for the user to input a temperature in Celsius, and to convert the value to a temperature in Fahrenheit.
- Discuss the number of taps that would be necessary if the interface consisted of a text field and button, including tapping the field, typing a number, and tapping a button to execute the temperature value conversion.
- Discuss the improved user experience of flicking a picker element and converting the temperature value when a temperature is selected.
- Using Interface Builder and the Object Library (⌥⌘L), add a text label for the converted temperature.
- Use the Attributes Inspector (⌥⌘4) to adjust the label size and typeface.
- Use upward Control-dragging to set the label's top spacing, the Align control (⫿) to center it horizontally, and the Resolve Auto Layout Issues control (⊦△⊦) to update the frame(⌥⌘=).
- Using Interface Builder and the Object Library (⌥⌘L), add a Picker View to the bottom of the interface.
- Use downward Control-dragging to set the Picker View bottom spacing, the Align control (⫿) to center it horizontally, and the Resolve Auto Layout Issues control (⊦△·) to update the frame(⌥⌘=).
- Use the Assistant Editor (⌥⌘↩) Preview to observe the interface on different devices and orientations.
- Discuss the positioning of the picker view, and how placing it at the bottom of the interface relates to how the user might use the app running on a device in the hand.
- Run the app (⌘R) and attempt to use the picker.

Closing

Why do you think the picker was unpopulated with the California city names when you ran the app?

Modifications and Extensions

- Create the temperature label with code inside the controller `viewDidLoad` method, and critique the benefits and drawbacks of creating the interface procedurally.

Resources

Xcode Overview: Build a User Interface https://developer.apple.com/library/ios/documentation/ToolsLanguages/Conceptual/Xcode_Overview/edit_user_interface.html

Interface Builder Object and Media Help: Adding an Object to Your Interface https://developer.apple.com/library/ios/recipes/xcode_help-IB_objects_media/Chapters/AddingObject.html

Auto Layout Guide: Working with Constraints in Interface Builder https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/AutolayoutPG/WorkingwithConstraints/WorkingwithConstraints.html