

PickerViews

PickerView Operations

Theory:

- What are pickerviews
- What are examples of where pickerviews used
- What is a delegate and data source?

Code:

- Displaying a pickerview
- Adding data to the pickerview
- Detecting when value in pickerview changes
- Getting the user's selection from the pickerview

PickerViews

PickerViews:

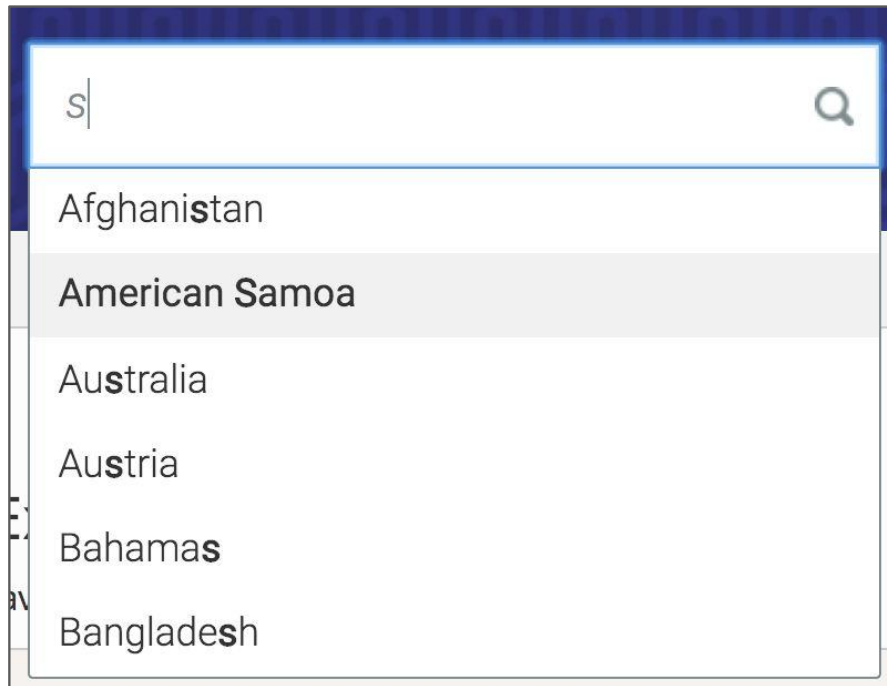
- Used to display data **and** let the user **select** data
- Similar to a “drop down” menu in web
- Spinner in Android

The screenshot shows an iOS 'Add Event' form. At the top, there are three buttons: 'Cancel' (red), 'Add Event' (black), and 'Done' (red). Below these is a 'Location' field. The 'All-day' toggle is currently off. The 'Starts' section shows the date '25 Oct 2013' and time '16:00' in red. Below this is a date picker grid showing days of the week and dates. The 'Ends' section shows the time '17:00'.

Day	Date	Time
Tue	22 Oct	13 57
Wed	23 Oct	14 58
Thu	24 Oct	15 59
Today	16	00
Sat	26 Oct	17 01
Sun	27 Oct	18 02
Mon	28 Oct	19 03

Dropdown menus vs. Pickers

Dropdown menu



A screenshot of a web form featuring a search bar with the letter 's' and a magnifying glass icon. Below the search bar, a dropdown menu is open, displaying a list of countries. The countries listed are Afghanistan, American Samoa, Australia, Austria, Bahamas, and Bangladesh. The text 'American Samoa' is highlighted in a light gray background.

Country
Afghanistan
American Samoa
Australia
Austria
Bahamas
Bangladesh

Picker Menu



A screenshot of a web form with two sections. The first section, titled 'Select Country Name', contains a text input field with the value 'Canada'. The second section, titled 'Select Date', contains a text input field with the value '06/28/2008'. Below these sections, a list of countries is displayed in a light gray background, with 'Canada' highlighted in a darker gray background.

Country
India
United States
United Kingdom
Canada
Australia
UAE
Russia

Pickers vs TableViews

TableViews are another UI control that let user select data.

But a picker and a table view are different

Picker vs. TableView - example

Pickerview

Android:
Spinner

Select Country Name

Canada

Select Date

06/28/2008

India

United States

United Kingdom

Canada

Australia

UAE

Russia

TableView


Android:
ListView


[Back](#)


Select a Country


Q Search


A


 Afghanistan


 Aland Islands


 Albania

 Algeria

 American Samoa

 Andorra

 Angola

 Anguilla

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V

Built-in vs. Custom Pickers

Built in PickersViews:

- Date / Time Picker

Custom Picker

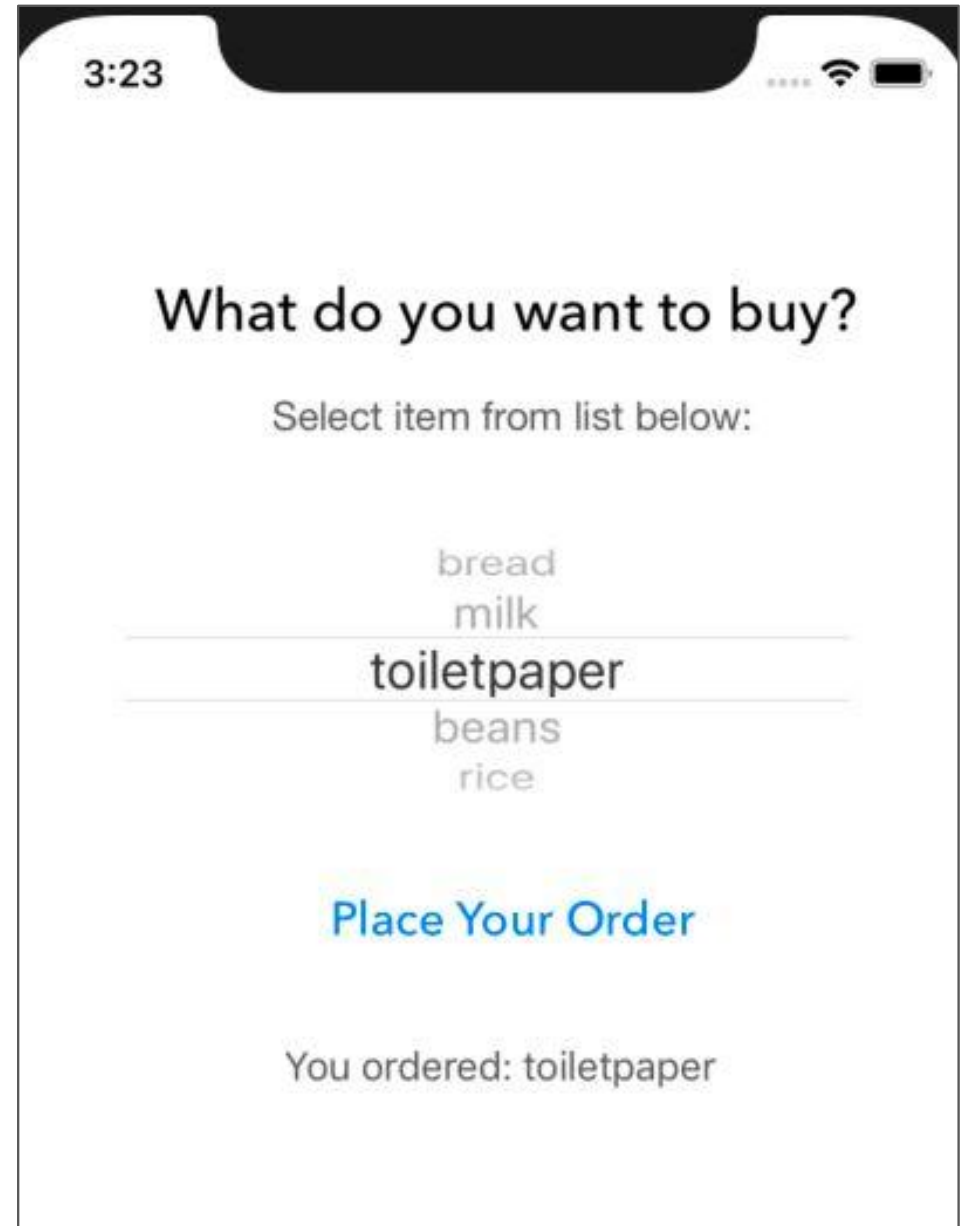
- Uses your own data

Coding a Picker View

Example App

User can select an item from
PickerView

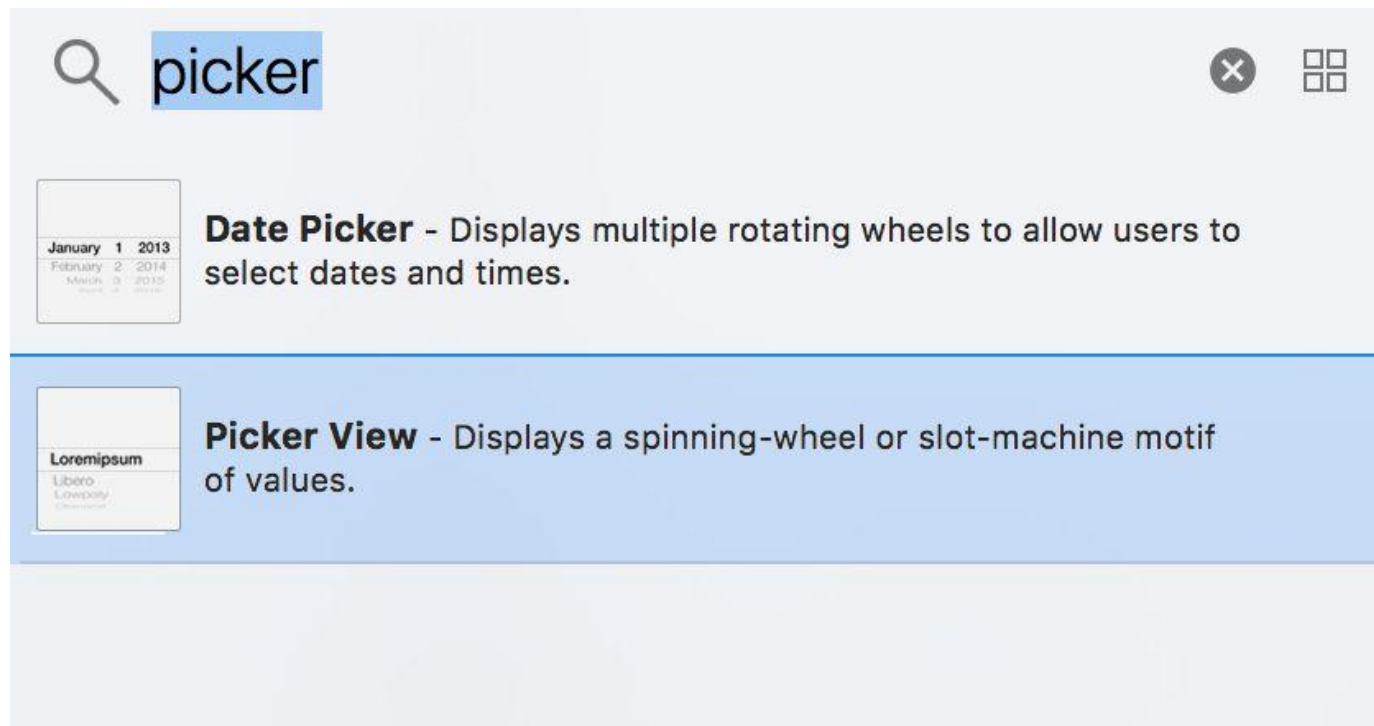
Click button, item is displayed in results
label



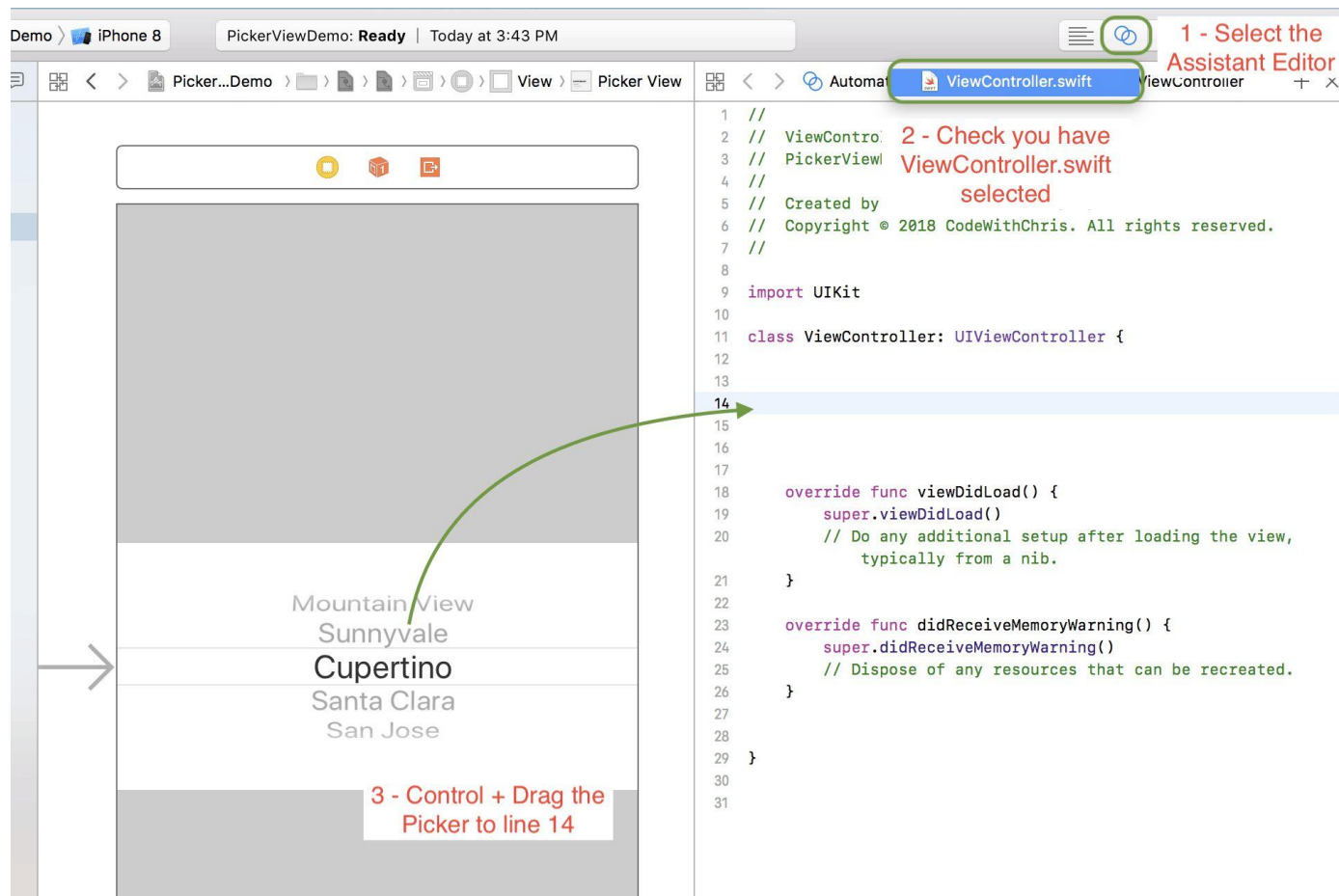
Recipe Alert: How to Add a UIPickerView

1. Choose a Picker type (Date vs. Custom)
2. In Main.storyboard, add the Picker to your layout
3. In ViewController.swift:
 - a. Add outlet for the picker view
 - b. Setup class to conform to UIPickerView **protocols**
 - c. Set up the **mandatory UIPickerView API functions**
 - i. `numberOfComponents`
 - ii. `numberOfRowsInComponent`
 - iii. `titleForRow row:`
 - d. Define what data will go in the UIPickerView
 - e. Connect data to the UIPickerView
4. Done!

1. Choose a Picker Type



2. Add UIPickerView and make outlets



3. Check that class conforms to protocols

Required protocols

```
class ViewController: UIViewController, UIPickerViewDelegate, UIPickerViewDataSource
{
    @IBOutlet weak var picker: UIPickerView!

    override func viewDidLoad() {
        super.viewDidLoad()

        // Do any additional setup after loading the view, typically from a nib.
        self.picker.delegate = self
        self.picker.dataSource = self
    }
}
```

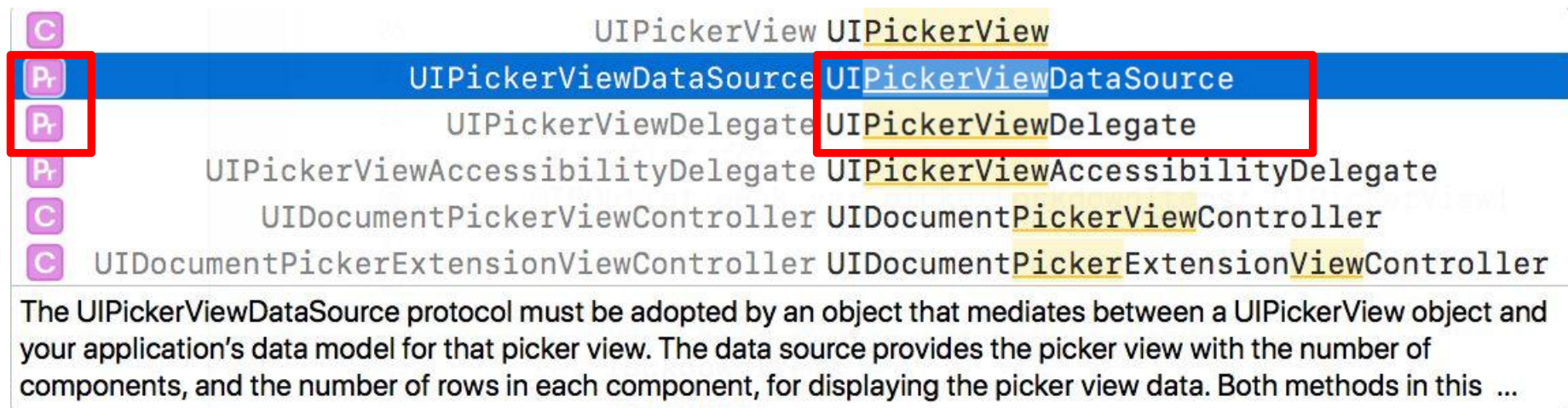
Pickerview outlet

required for protocols to work properly

3. pickerView Protocols

Any screen that includes a UIPickerView must conform to the UIPickerViewDataSource and UIPickerViewDelegate protocols.

These can be selected using XCode's autocomplete feature:



4. Add mandatory pickerView functions

```
// Number of columns of data you want to show in the pickerView
func numberOfComponents(in pickerView: UIPickerView) -> Int {
    return 1
}
```

```
// The total number of items in your pickerView
func pickerView(_ pickerView: UIPickerView, numberOfRowsInComponent component:
Int) -> Int {
    return 1;
}
```

```
// What data should be displayed in each row of the picker
func pickerView(_ pickerView: UIPickerView, titleForRow row: Int, forComponent
component: Int) -> String? {
    return "abcd"
}
```

5. Create a data source for your UIPickerView



1. If this is what you want in the UIPickerView then....

```
let countries = [  
    "United States",  
    "United Kingdom",  
    "Canada",  
    "Australia",  
    "UAE",  
    "Russia"  
]
```

2. This is what your data source looks like

6. Add mandatory UIPickerView functions

```
let countries = ["United States", "United Kingdom", "Canada", "Australia", "UAE"]
```

```
// Number of columns of data you want to show in the pickerview
```

```
func numberOfComponents(in pickerView: UIPickerView) -> Int {  
    return 1  
}
```

Change this if you want to have a multicolumn pickerview

```
// The total number of items in your UIPickerView
```

```
func pickerView(_ pickerView: UIPickerView, numberOfRowsInComponent component:  
Int) -> Int {  
    return countries.length;  
}
```

Updated to be same number as items in your data source

```
// What data should be displayed in each row of the picker
```

```
func pickerView(_ pickerView: UIPickerView, titleForRow row: Int, forComponent  
component: Int) -> String? {  
    return countries[row]  
}
```

Updated code to show the current item in the data source