

UIPickerView

Fabrice.Kordon@lip6.fr



As an introduction...

Select values in a list?

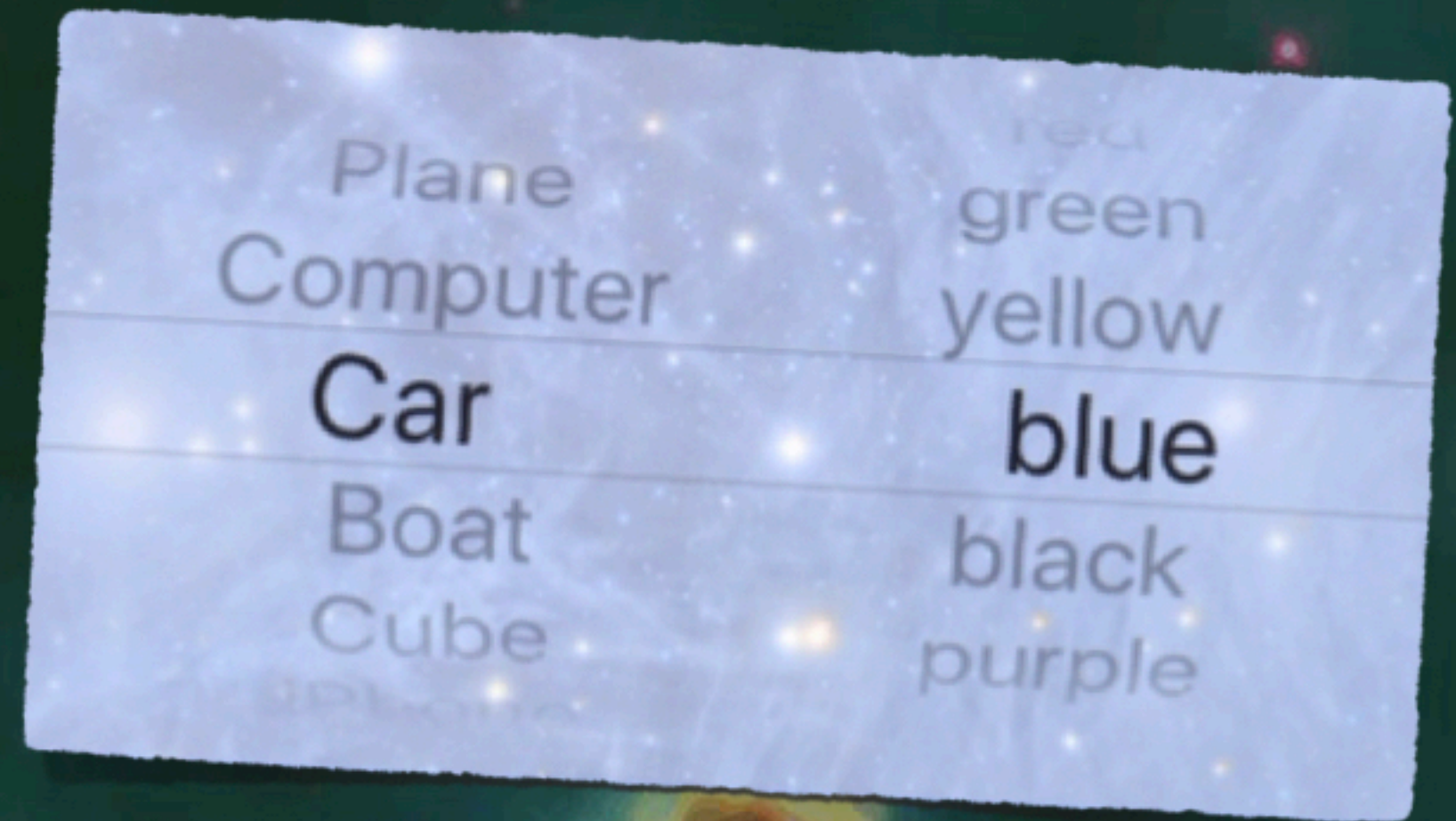
- One or several components

Two associated protocols

- UIPickerViewDataSource
 - ▶ Informations about the picker's content
- UIPickerViewDelegate
 - ▶ Management of the picker itself

Principles

- Creation and customization
- Set-up the two delegates
 - ▶ dataSource property
 - ▶ delegate property



Picker's creation

Default positions

- Default: 0'th value selected for each component
- To change it, use
 - ▶ `selectRow(_:inComponent:animated:)` / `selectRow:inComponent:animated:`

What's in the UIPickerView?

- UIPickerViewDataSource protocol
- Two mandatory methods (returning an integer)
 - ▶ `numberOfComponents(in:)` / `numberOfComponentsInPickerView:`
 - ▶ `pickerView(_:numberOfRowsInComponent:)` / `pickerView:numberOfRowsInComponent:`

Responding to events

UIPickerViewDelegate protocol

- Fetch the value of a component
 - ▶ Returning a String, `pickerView(_:titleForRow:forComponent:)` / `pickerView:titleForRow:forComponent:`
 - ▶ Returning a UIView, `pickerView(_:viewForRow:forComponent:reusing:)` / `pickerView:viewForRow:forComponent:reusingView:`
 - CAUTION, you must avoid creating too many views
- Be informed when something is selected
 - ▶ `pickerView(_:didSelectRow:inComponent:)` / `pickerView:didSelectRow:inComponent:`

The picker may be customized

- Property `showsSelectionIndicator`
- Other UIPickerViewDelegate's methods
 - ▶ `pickerView(_:rowHeightForComponent:)` / `pickerView:rowHeightForComponent:`
 - ▶ `pickerView(_:widthForComponent:)` / `pickerView:widthForComponent:`

As a conclusion...

Another useful object for user interaction

- And two delegate protocols



Very common in applications

- Usually embedded in modal views
 - ▶ appear and disappear
 - ▶ graphical effects
- For large devices?
 - ▶ Better to have it in popovers

