

# **Unit 4—Lesson 9:**

## **Building Complex Input Screens**

# Complex input screens

## Hotel Manzana

Cancel

New Guest Registration

Done

First Name

Last Name

Email

Check-In Date

Feb 21, 2017

Check-Out Date

Feb 22, 2017

Adults

2

-

+

Children

0

-

+

Wi-Fi (per day)

\$ 10

Room Type

Not Set >

# Data models

## Registration

First & last name

Email address

Check-in and check-out dates

Number of adults

Number of children

Wi-Fi (per day)

Room type

## Room Type

ID

Name

Short name

Price

# Data models

```
struct Registration {  
    var firstName: String  
    var lastName: String  
    var emailAddress: String  
  
    var checkInDate: Date  
    var checkOutDate: Date  
    var numberOfAdults: Int  
    var numberOfChildren: Int  
  
    var roomType: RoomType  
    var wifi: Bool  
}
```

```
struct RoomType: Equatable {  
    var id: Int  
    var name: String  
    var shortName: String  
    var price: Int  
  
    //Equatable Protocol Implementation for RoomType  
    static func == (lhs: RoomType, rhs: RoomType) -> Bool {  
        return lhs.id == rhs.id  
    }  
}
```

# Converting dates to strings

## Date style

Style	dateStyle	timeStyle
.none		
.short	11/23/37	3:30 PM
.medium	Nov 23, 1937	3:30:32 PM
.long	November 23, 1937	3:30:32 PM PST
.full	Tuesday, April 12, 1952 AD	3:30:42 PM Pacific Standard Time

```
let dateFormatter = DateFormatter()
```

# Converting dates to strings

```
let dateFormatter = DateFormatter()  
dateFormatter.dateStyle = .medium  
  
checkInDateLabel.text = dateFormatter.string(from: checkInDatePicker.date)
```

# Date picker

## UIDatePicker and datePickerMode

November	12	2014
December	13	2015
January	14	2016
<b>February</b>	<b>15</b>	<b>2017</b>
March	16	2018
April	17	2019
May	18	2020

Date  
.date

11	38	
12	39	
1	40	AM
<b>2</b>	<b>41</b>	<b>PM</b>
3	42	
4	43	
5	44	

Time  
.time

Sun Feb 12	11	38	
Mon Feb 13	12	39	
Tue Feb 14	1	40	AM
<b>Today</b>	<b>2</b>	<b>41</b>	<b>PM</b>
Thu Feb 16	3	42	
Fri Feb 17	4	43	
Sat Feb 18	5	44	

Date and time  
.dateAndTime

	58
	59
	0
<b>0 hours</b>	<b>1 min</b>
1	2
2	3
3	4

Countdown timer  
.countDownTimer

# Date picker

## Parameters

`minimumDate` and `maximumDate`

- Minimum date must be earlier than maximum date

`minuteInterval`

- For minute wheel, if displayed
- Must be evenly dividable into 60



# Adjusting cell heights

## tableView(\_:heightForRowAt:)

```
override func tableView(_ tableView: UITableView,  
                        heightForRowAt indexPath: IndexPath) -> CGFloat
```

Allows the delegate to specify rows with varying heights

Use `indexPath` to determine the row in `tableView`

Returned value overrides the value specified for the `rowHeight` property of `UITableView` for the given row

```
override func tableView(_ tableView: UITableView,
                        heightForRowAt indexPath: IndexPath) -> CGFloat {

    switch (indexPath.section, indexPath.row) {
        case (datePickerCellIndexPath.section, datePickerCellIndexPath.row):
            if isDatePickerShown {
                return 216.0
            } else {
                return 0.0
            }
        default:
            return 44.0
    }
}
```

# Unit 4—Lesson 9

## Complex Input Screens



Learn how to incorporate these system view controllers for displaying alerts, sharing content, sending messages, and accessing the camera and photo library on an iOS device.

# Unit 4—Lesson 9

## Lab: Employee Roster



Create a screen that accepts complex user input to build an employee roster that keeps track of employee information

