Announcement

• Lab 4 is due on Monday October 22\textsuperscript{nd} at 11:59 PM

Today’s Topics

• Mapkit

• Core Location
MapKit

What is MapKit?

- API to display Maps
- Classes to translate between CLLocation and human-readable addresses
- Support for “annotations” (pins on a map)
- Reverse Geocoding
MKMapView

- Handles display of map
- “Map” & “Satellite” types
- Panning and Zooming
- Annotations
- Display User Location

Properties in MKMapView

```swift
var region: MKCoordinateRegion
var centerCoordinate: CLLocationCoordinate2D
var userLocation: MKUserLocation
var annotations: [MKAnnotation]
var delegate: MKMapViewDelegate?
MKMapType mapType
```
MKMapViewDelegate

- Callback methods about loading state:
  
  ```swift
  func mapViewWillStartLoadingMap(_ mapView: MKMapView)
  func mapViewDidFinishLoadingMap (_ mapView: MKMapView)
  func mapViewDidFailLoadingMap (_ mapView: MKMapView, withError error: Error)
  ```

- Callback methods about region changes:
  
  ```swift
  func mapView(_ mapView: MKMapView, regionWillChangeAnimated animated: Bool)
  ```
  
  ```swift
  -func mapView(_ mapView: MKMapView, regionDidChangeAnimated :animated Bool)
  ```

- Callback methods to customize and interact with annotations
  
  ```swift
  func mapView (MKMapView, viewFor : MKAnnotation)
  ```
  
  ```swift
  func mapView (MKMapView, didAdd: [MKAnnotationView])
  ```
  
  ```swift
  func mapView (MKMapView, annotationView: MKAnnotationView, calloutAccessoryControlTapped: UIControl)
  ```
MKAnnotation

- A protocol - not a class
- Add to a MapView to plot pins
  var coordinate: CLLocationCoordinate2D

  var title: String?
  var subtitle: String?

MKPlacemark

- Conforms to MKAnnotation protocol
- Convenience for holding human-readable addresses alongside Coordinate

  init(coordinate: CLLocationCoordinate2D, addressDictionary: [String : Any]?)

- Easy to convert between AddressBook addresses and location:
  - thoroughfare, subThoroughfare, locality, subLocality,
    administrativeArea, subAdministrativeArea, postalCode, country,
    countryCode
MKUserLocation

- Special case of an MKAnnotation
- Represents device’s location only
  - You do not create instances of this class directly
  - Retrieve an existing MKUserLocation object from userLocation property of map

var location: CLLocation?
var isUpdating: Bool
var title: String?
var subtitle: String?
Core Location

• What is it?

• Core Location

Activate service  Location ring
Core Location

- Location Technologies

Bootstrap
Crosscheck
Complement
Core Location Framework

• The core classes and protocols
  • Classes
    – CLLocationCoordinate
      • Represents a point and vector in the real world
    – CLLocationManager
      • Allows you to get a CLLocationCoordinate
  • Protocol
    – CLLocationManagerDelegate
Core Location Framework

- CLLocationManagerDelegate protocol

- Several useful optional methods

  ```swift
  func locationManager(CLLocationManager, didUpdateLocations: [CLLocation])
  func locationManager(CLLocationManager, didFailWithError: Error?)
  ```

- Called asynchronously on main thread
- Issues movement-based updates

Getting a Location

- Starting the location service

  ```swift
  let locationManager = CLLocationManager()
  locationManager.delegate = self
  locationManager.requestWhenInUseAuthorization()
  locationManager.startUpdatingLocation()
  ```
Getting user location

- iOS 8 introduced additional requirements to obtain your location
  - Call the requestWhenInUseAuthorization method
  - Add an entry to your plist file to request location
    - NSLocationWhenInUseUsageDescription

Getting a Location – Using Event Data

```swift
func locationManager(manager: CLLocationManager, didUpdateLocations locations: [CLLocation]) {

    let aLocation = locations[0]
    let howRecent = aLocation.timestamp.timeIntervalSinceNow

    if (howRecent < -10) { return }

    if (aLocation.horizontalAccuracy > 100) { return }

    double lat = aLocation.coordinate.latitude
    double long = aLocation.coordinate.longitude

    }
```
### Desired Accuracy

- **Choosing an appropriate accuracy level**
  
  ```
  locationManager.desiredAccuracy = kCLLocationAccuracyBest
  ```

- **Choose an appropriate accuracy level**
  - Higher accuracy impacts power consumption
  - Lower accuracy is “good enough” in most cases

- Can change accuracy setting later if needed

- Actual accuracy reported in CLLocation object

### Distance Filter

- **Choosing an appropriate update threshold**

- **New events delivered when threshold exceeded**
  
  ```
  locationManager.distanceFilter = 3000
  ```
Stopping the Service

locationManager.stopUpdatingLocation()

- Restart the service later as needed
- Also able to pause service and run in background
  - var pausesLocationUpdatesAutomatically: Bool
  - var allowsBackgroundLocationUpdates: Bool

Responding to Errors

- User may deny use of the location service
- Results in a kCLErrorDenied error
- Protects user privacy
- Occurs on a per-application basis
Responding to Errors

- Location may be unavailable
- Results in a kCLErrorLocationUnknown error
- Likely just temporary
- Scan continues in background

Demo
GPS Data
Demo
Geocoding

https://github.com/ooper-shlab/GeocoderDemo-Swift