

# CS193P - Lecture 12

## iPhone Application Development

Web Views

Location & Maps

ウェブ=ビューと  
位置センシングとマップ

# Announcements

- Paparazzi 3 is due next Wednesday at 11:59pm
- Friday section tomorrow at 4 PM, Building 260 Room 113
  - Evan Doll
    - Former CS193p lecturer
    - Giving his thoughts on iPhone and iPad opportunities

# Today's Topics

## 今日のトピックス

- UIWebView

- Loading
- Navigating

- CoreLocation

- MapKit

- MKMapView
- Annotations
- Reverse Geocoding

## ウェブビュー

ウェブ文書のロード  
ナビゲーション

## コア=ロケーション (位置センシング)

## マップ=キット

MKMapView  
アノテーション (付加情報)  
地理情報の逆変換

# UIWebView

ウェブ=ビュー

# Displaying Web Content

- Web content can be displayed with UIWebView ウェブコンテンツを表示する
- Content can be
  - local HTML string ローカルの HTML 文字列
  - local raw data + MIME type ローカルの生データ + MIMEタイプ
  - remote URL リモート URL (外部ウェブ文書)
- Leverages WebKit
  - full WK functionality not currently exposed ロード&ナビのためのシンプルな API
  - simple API for loading & navigating
  - delegate for some control
  - limited JavaScript execution support
    - 5 seconds of execution & 10 MB of memory 制約つきの JS 実行 5秒・10MB 以内

# UIWebView

- UIView subclass, configure in IB or in code
- Feed it data to display 表示するデータの与え方
  - (void)loadHTMLString:(NSString \*)string baseURL:(NSURL \*)baseURL;  
HTML文字列
  - (void)loadData:(NSData \*)data MIMEType:(NSString \*)MIMEType  
textEncodingName:(NSString \*)encodingName  
画像等のデータ baseURL:(NSURL \*)baseURL;
- Or give it a URL request
  - (void)loadRequest:(NSURLRequest \*)request;  
URL でリクエスト
- What's this NSURLRequest?
  - Encapsulates a URL to load and caching policy for fetched data  
NSURL はウェブ文書をロード/キャッシュ

# UIWebView

ウェブ=ビューに期待されるプロパティとアクション

- Properties and actions you'd expect from a web view

@property BOOL loading;	ロード中か？
@property BOOL canGoBack;	「戻る」は可能か？
@property BOOL canGoForward;	「進む」は可能か？
- (void)reload;	再読込
- (void)stopLoading;	停止
- (void)goBack;	戻る
- (void)goForward;	進む

- A couple others that are handy  他の便利な機能

@property BOOL scalesPageToFit;	文書を拡大フィットするか？
@property BOOL detectsPhoneNumbers;	電話番号をリンク化するか？

# UIWebViewDelegate デリゲート (委任/委譲)

- Callbacks for load progress

- (void)webViewDidStartLoad:(UIWebView \*)webView;      ロード開始した
- (void)webViewDidFinishLoad:(UIWebView \*)webView;      ロード終了した

- Error handling

- (void)webView:(UIWebView \*)webView  
    didFailLoadWithError:(NSError \*)error;      ロード失敗

- Navigation management

- (BOOL)webView:(UIWebView \*)webView  
    shouldStartLoadWithRequest:(NSURLRequest \*)request      ロード中にロードを許可するか  
    navigationType:(UIWebViewNavigationType)navigationType;

- navigationType specifies things like link clicked, reload, form submitted, back/forward, or other      ロードの原因 (イベント) を navigationType に入れる



# Demo

## UIWebView

3行で出来るウェブブラウザ  
(ほんとよ)

# Demo

UIWebView - using WebViews for “rich” text

(これは割愛します)

# Core Location

位置情報のセンシング

# What is CoreLocation?

コア=ロケーションとは？

- A framework to manage Location

位置情報を管理するFW

- CLLocation
- CLLocationManager
- CLHeading

緯度・経度・高度  
もろもろの管理  
方角（磁気コンパス）

- No UI

ユーザインタフェースはない

# How Does it KNOW??

どのように位置を把握するのか

- Three tiered approach:

3 階層アプローチ

- GPS
- Wifi
- Cell Network

- GPS
- Wifi 基地局の位置情報
- 携帯基地局の位置情報

- The more accurate the technology, the more power it costs

精度が高いほどパワーを消費する(らしい)

# CLLocation

- An object to represent a point and vector in the real world  
実世界における点とベクトルを表現するオブジェクト

@property CLLocationCoordinate2D coordinate; 座標 (緯度・経度)  
@property CLLocationDistance altitude; 高度

@property CLLocationAccuracy horizontalAccuracy; 水平精度  
@property CLLocationAccuracy verticalAccuracy; 垂直精度

@property CLLocationDirection course; 移動方向 (ベクトル)  
@property CLLocationSpeed speed; 移動速度 m/s

- (NSDate \*)timeStamp; 時刻
- (CLLocationDistance)distanceFromLocation:(CLLocation \*)location  
ある位置からの距離

# CLLocationManager

- Your entry point to the location service 位置情報サービスへの入口

```
@property CLLocation *location;
```

```
@property id <CLLocationManagerDelegate> delegate;
```

```
@property CLLocationDistance distanceFilter; 距離フィルタ  
@property CLLocationAccuracy verticalAccuracy; 垂直精度
```

- (void)startUpdatingLocation 位置更新スタート
- (void)stopUpdatingLocation ストップ
- (void)startUpdatingHeading 方向更新スタート
- (void)stopUpdatingHeading ストップ

# CLLocationManagerDelegate 委任/委譲

- Callbacks for location change 位置変化時に呼び出されるメソッド

- (void)locationManager:(CLLocationManager \*)manager  
didUpdateToLocation:(CLLocation \*)newLocation 新しい位置  
fromLocation:(CLLocation \*)oldLocation; 古い位置

## Callbacks for heading change 方角変化時に呼び出されるメソッド

- (void)locationManager:(CLLocationManager \*)manager  
didUpdateHeading:(CLHeading \*)newHeading; 新しい方角

## Error handling

- (void)locationManager:(CLLocationManager \*)manager  
didFailLoadWithError:(NSError \*)error; エラーのとき



# MapKit

# What is MapKit?

マップ=キットとは？

- API to display Maps      地図を表示する API
- Classes to translate between CLLocation and human-readable addresses      住所（アドレス）      位置情報
- Support for “**annotations**” (pins on a map)      付加情報（ピン）
- Reverse Geocoding      位置情報      住所（アドレス）

# MKMapView

- Handles display of map マップ表示
- “Map” & “Satellite” types 地図/衛星
- Panning and Zooming 移動とズーム
- Annotations 付加情報
- Display User Location ユーザ位置の表示



# MKMapView

- Properties in MKMapView プロパティ (変数)

@property MKCoordinateRegion region; 地図上の長方形領域  
@property CLLocationCoordinate2D centerCoordinate; その中央

@property MKMapType mapType; 標準・衛星・両方など

@property NSArray \*annotations; 付加情報 (ピン等)  
@property MKUserLocation userLocation; ユーザ位置 (読み出しのみ)

@property id <MKMapViewDelegate> delegate;

# MKMapViewDelegate

- Callback methods about loading state: マップロードに伴って呼び出されるメソッド
  - (void)mapViewWillStartLoadingMap:(MKMapView \*)mapView;
  - (void)mapViewDidFinishLoadingMap:(MKMapView \*)mapView;
  - (void)mapViewDidFailLoadingMap:(MKMapView \*)mapView  
withError:(NSError \*)error;
- Callback methods about region changes: リージョン変更 (ユーザによるドラッグ/ズーム)に伴って呼び出されるメソッド
  - (void)mapView:(MKMapView \*)mapView  
regionWillChangeAnimated:(BOOL)animated;
  - (void)mapView:(MKMapView \*)mapView  
regionDidChangeAnimated:(BOOL)animated;

# MKMapViewDelegate

- Callback methods to customize and interact with “annotations”:

## 付加情報（ピン等）の操作

- (MKAnnotationView \*)mapView:(MKMapView \*)mapView  
viewForAnnotation:(id <MKAnnotation>)annotation;
- (void)mapView:(MKMapView \*)mapView  
didAddAnnotationViews:(NSArray \*)views;
- (void)mapView:(MKMapView \*)mapView  
annotationView:(MKAnnotationView \*)view  
calloutAccessoryControlTapped:(UIControl \*)control;

# MKAnnotation

- A @protocol - not a @class
- Add to a MapView to plot pins

```
@property CLLocationCoordinate2D coordinate;
```

```
@property NSString *title;
```

```
@property NSString *subtitle;
```

付加情報（ピン）は  
位置だけでなく  
タイトルとサブタイトルをもつ

# MKPlacemark

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

```
- (void)initWithCoordinate:(CLLocationCoordinate2D *)coordinate  
    addressDictionary:(NSDictionary *)dictionary;
```

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

住所アドレスとの変換



# MKUserLocation

- Special case of an MKAnnotation
- Represents device's location only    **デバイスの位置を表わす**

```
@property BOOL updating (getter = isUpdating);  
@property CLLocation *location;
```

```
@property NSString *title;  
@property NSString *subtitle;
```

# MKReverseGeocoder

- Given a location, what's the human-readable address?

位置情報からアドレスを得るには？

- (void)`initWithCoordinate:(CLLocationCoordinate2D)coordinate;`  
@property id <MKReverseGeocoderDelegate> delegate;
- (void)`start;`
- (void)`cancel;`

- Delegate callbacks:

- (void)`reverseGeocoder:(MKReverseGeocoder *)geocoder`  
`didFindPlacemark:(MKPlacemark *)placemark;`
- (void)`reverseGeocoder:(MKReverseGeocoder *)geocoder`  
`didFailWithError:(NSError *)error;`

# Demo

## MKMapView and friends

3行で出来る地図アプリ（ほんとよ）

# Questions?