```objc
/**
 *  Section3App0AppDelegate.m
 *  Section3App0
 *  Tommy MacWilliam, 2011
 *
 */

#import "Section3App0AppDelegate.h"

#import "Section3App0ViewController.h"

@implementation Section3App0AppDelegate


@synthesize window=_window;

@synthesize viewController=_viewController;

- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    // Override point for customization after application launch.

    self.window.rootViewController = self.viewController;
    [self.window makeKeyAndVisible];
    return YES;
}

- (void)applicationWillResignActive:(UIApplication *)application
{
    /*
     Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions (such as an incoming phone call or SMS message) or when the user quits the application and it begins the transition to the background state.
     Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause the game.
     */
}

- (void)applicationDidEnterBackground:(UIApplication *)application
{
    /*
     Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore your application to its current state in case it is terminated later.
     If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.
     */
}

- (void)applicationWillEnterForeground:(UIApplication *)application
{
    /*
     Called as part of the transition from the background to the inactive state; here you can undo many of the changes made on entering the background.
     */
}

- (void)applicationDidBecomeActive:(UIApplication *)application
{
    /*
     Restart any tasks that were paused (or not yet started) while the application was inactive. If the application was previously in the background, optionally refresh the user interface.
     */
}

- (void)applicationWillTerminate:(UIApplication *)application
{
    /*
     Called when the application is about to terminate.
     Save data if appropriate.
     See also applicationDidEnterBackground:.
     */
}

- (void)dealloc
{
    [_window release];
    [_viewController release];
    [super dealloc];
}

@end
```

```objc
/**
 *  Section3App0ViewController.m
 *  Section3App0
 *  Tommy MacWilliam, 2011
 *
 *  Create a fullscreen MKMapView
 *  Don't forget to add MapKit.framework!
 *
 */

#import "Section3App0ViewController.h"

@implementation Section3App0ViewController

@synthesize mapView=_mapView;

- (void)dealloc
{
    // release ivars
    [_mapView release];

    [super dealloc];
}

- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
}

#pragma mark - View lifecycle


- (void)viewDidUnload
{
    // release properties
    self.mapView = nil;

    [super viewDidUnload];
}

- (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
{
    return (interfaceOrientation == UIInterfaceOrientationPortrait);
}

@end
```

```objc
//
//  main.m
//  Section3App0
//
//  Created by labuser on 4/11/11.
//  Copyright 2011 Harvard University. All rights reserved.
//

#import <UIKit/UIKit.h>

int main(int argc, char *argv[])
{
    NSAutoreleasePool *pool = [[NSAutoreleasePool alloc] init];
    int retVal = UIApplicationMain(argc, argv, nil, nil);
    [pool release];
    return retVal;
}
```

```
/**
 *  Section3App1AppDelegate.m
 *  Section3App1
 *  Tommy MacWilliam, 2011
 *
 */

#import "Section3App1AppDelegate.h"
#import "Section3App1ViewController.h"

@implementation Section3App1AppDelegate

@synthesize window=_window;
@synthesize viewController=_viewController;

- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    // Override point for customization after application launch.

    self.window.rootViewController = self.viewController;
    [self.window makeKeyAndVisible];
    return YES;
}

- (void)applicationWillResignActive:(UIApplication *)application
{
    /*
     Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions (such as an incoming phone call or SMS message) or when the user quits the application and it begins the transition to the background state.
     Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause the game.
     */
}

- (void)applicationDidEnterBackground:(UIApplication *)application
{
    /*
     Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore your application to its current state in case it is terminated later.
     If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.
     */
}

- (void)applicationWillEnterForeground:(UIApplication *)application
{
    /*
     Called as part of the transition from the background to the inactive state; here you can undo many of the changes made on entering the background.
     */
}

- (void)applicationDidBecomeActive:(UIApplication *)application
{
    /*
     Restart any tasks that were paused (or not yet started) while the application was inactive. If the application was previously in the background, optionally refresh the user interface.
     */
}

- (void)applicationWillTerminate:(UIApplication *)application
{
    /*
     Called when the application is about to terminate.
     Save data if appropriate.
     See also applicationDidEnterBackground:.
     */
}

- (void)dealloc
{
    [_window release];
    [_viewController release];
    [super dealloc];
}

@end
```

```objc
/**
 *  Section3App1ViewController.m
 *  Section3App1
 *  Tommy MacWilliam, 2011
 *
 *  Create a fullscreen MKMapView centered on a given point
 *
 */

#import "Section3App1ViewController.h"
#import "MapKit/MapKit.h"

@implementation Section3App1ViewController

@synthesize mapView=_mapView;

- (void)dealloc
{
    // release ivars
    [_mapView release];

    [super dealloc];
}

- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
}

#pragma mark - View lifecycle

- (void)viewDidLoad
{
    [super viewDidLoad];

    // define span for map: how much area will be shown
    MKCoordinateSpan span;
    span.latitudeDelta = 0.002;
    span.longitudeDelta = 0.002;

    // define starting point for map
    CLLocationCoordinate2D start;
    start.latitude = 42.36873056998856;
    start.longitude = -71.11504912376404;

    // create region, consisting of span and location
    MKCoordinateRegion region;
    region.span = span;
    region.center = start;

    // move the map to our location
    [self.mapView setRegion:region animated:YES];
}

- (void)viewDidUnload
{
    // release properties
    self.mapView = nil;

    [super viewDidUnload];
}

- (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
{
    return (interfaceOrientation == UIInterfaceOrientationPortrait);
}

@end
```

```
//
//  main.m
//  Section3App1
//
//  Created by labuser on 4/11/11.
//  Copyright 2011 Harvard University. All rights reserved.
//

#import <UIKit/UIKit.h>

int main(int argc, char *argv[])
{
    NSAutoreleasePool *pool = [[NSAutoreleasePool alloc] init];
    int retVal = UIApplicationMain(argc, argv, nil, nil);
    [pool release];
    return retVal;
}
```

```
/**
 *  Section3App2AppDelegate.m
 *  Section3App2
 *  Tommy MacWilliam, 2011
 *
 */

#import "Section3App2AppDelegate.h"
#import "Section3App2ViewController.h"

@implementation Section3App2AppDelegate

@synthesize window=_window;
@synthesize viewController=_viewController;

- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    // Override point for customization after application launch.

    self.window.rootViewController = self.viewController;
    [self.window makeKeyAndVisible];
    return YES;
}

- (void)applicationWillResignActive:(UIApplication *)application
{
    /*
     Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions (such as an incoming phone call or SMS message) or when the user quits the application and it begins the transition to the background state.
     Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause the game.
     */
}

- (void)applicationDidEnterBackground:(UIApplication *)application
{
    /*
     Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore your application to its current state in case it is terminated later.
     If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.
     */
}

- (void)applicationWillEnterForeground:(UIApplication *)application
{
    /*
     Called as part of the transition from the background to the inactive state; here you can undo many of the changes made on entering the background.
     */
}

- (void)applicationDidBecomeActive:(UIApplication *)application
{
    /*
     Restart any tasks that were paused (or not yet started) while the application was inactive. If the application was previously in the background, optionally refresh the user interface.
     */
}

- (void)applicationWillTerminate:(UIApplication *)application
{
    /*
     Called when the application is about to terminate.
     Save data if appropriate.
     See also applicationDidEnterBackground:.
     */
}

- (void)dealloc
{
    [_window release];
    [_viewController release];
    [super dealloc];
}

@end
```

```objc
/**
 *  Section3App2ViewController.m
 *  Section3App2
 *  Tommy MacWilliam, 2011
 *
 *  Create an MKMapView with a single annotation
 *
 */

#import "Section3App2ViewController.h"

@implementation Section3App2ViewController

@synthesize mapView=_mapView;

- (void)dealloc
{
    // release ivars
    [_mapView release];

    [super dealloc];
}

- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
}

#pragma mark - View lifecycle

- (void)viewDidLoad
{
    [super viewDidLoad];

    // define span for map: how much area will be shown
    MKCoordinateSpan span;
    span.latitudeDelta = 0.002;
    span.longitudeDelta = 0.002;

    // define starting point for map
    CLLocationCoordinate2D start;
    start.latitude = 42.36873056998856;
    start.longitude = -71.11504912376404;

    // create region, consisting of span and location
    MKCoordinateRegion region;
    region.span = span;
    region.center = start;

    // move the map to our location
    [self.mapView setRegion:region animated:YES];

    // create new annotation
    MKPointAnnotation *annotation = [[MKPointAnnotation alloc] init];
    annotation.coordinate = start;
    annotation.title = @"Mather House";
    annotation.subtitle = @"The best house";

    // add annotation to map
    [self.mapView addAnnotation:annotation];
    [annotation release];
}

- (void)viewDidUnload
{
    // release properties
    self.mapView = nil;

    [super viewDidUnload];
}

- (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
{
    return (interfaceOrientation == UIInterfaceOrientationPortrait);
}

@end
```

```
//
//  main.m
//  Section3App2
//
//  Created by labuser on 4/11/11.
//  Copyright 2011 Harvard University. All rights reserved.
//

#import <UIKit/UIKit.h>

int main(int argc, char *argv[])
{
    NSAutoreleasePool *pool = [[NSAutoreleasePool alloc] init];
    int retVal = UIApplicationMain(argc, argv, nil, nil);
    [pool release];
    return retVal;
}
```

```
/**
 *  Marker.m
 *  Section3App3
 *  Tommy MacWilliam, 2011
 *
 */

#import "CustomAnnotation.h"

@implementation CustomAnnotation

@synthesize coordinate=_coordinate, title=_title, subtitle=_subtitle;

/**
 * Create a new annotation from a given coordinate
 *
 */
- (id)initWithCoordinate:(CLLocationCoordinate2D)coordinate
{
    self = [super init];

    if (self != nil) {
        self.coordinate = coordinate;
    }

    return self;
}

- (void)dealloc
{
    // release properties
    self.title = nil;
    self.subtitle = nil;

    [super dealloc];
}

@end
```

```objc
/**
 *  Section3App3AppDelegate.m
 *  Section3App3
 *  Tommy MacWilliam, 2011
 *
 */

#import "Section3App3AppDelegate.h"
#import "Section3App3ViewController.h"

@implementation Section3App3AppDelegate

@synthesize window=_window;
@synthesize viewController=_viewController;

- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    // Override point for customization after application launch.

    self.window.rootViewController = self.viewController;
    [self.window makeKeyAndVisible];
    return YES;
}

- (void)applicationWillResignActive:(UIApplication *)application
{
    /*
     Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions (such as an incoming phone call or SMS message) or when the user quits the application and it begins the transition to the background state.
     Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause the game.
     */
}

- (void)applicationDidEnterBackground:(UIApplication *)application
{
    /*
     Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore your application to its current state in case it is terminated later.
     If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.
     */
}

- (void)applicationWillEnterForeground:(UIApplication *)application
{
    /*
     Called as part of the transition from the background to the inactive state; here you can undo many of the changes made on entering the background.
     */
}

- (void)applicationDidBecomeActive:(UIApplication *)application
{
    /*
     Restart any tasks that were paused (or not yet started) while the application was inactive. If the application was previously in the background, optionally refresh the user interface.
     */
}

- (void)applicationWillTerminate:(UIApplication *)application
{
    /*
     Called when the application is about to terminate.
     Save data if appropriate.
     See also applicationDidEnterBackground:.
     */
}

- (void)dealloc
{
    [_window release];
    [_viewController release];
    [super dealloc];
}

@end
```

```
/**
 *  Section3App3ViewController.m
 *  Section3App3
 *  Tommy MacWilliam, 2011
 *
 *  Create an MKMapView with several customized annotations
 *
 */

#import "Section3App3ViewController.h"
#import "MapKit/MapKit.h"
#import "CustomAnnotation.h"

@implementation Section3App3ViewController

@synthesize mapView=_mapView;

- (void)dealloc
{
    // release ivars
    [_mapView release];

    [super dealloc];
}

- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
}

#pragma mark - View lifecycle

- (void)viewDidLoad
{
    [super viewDidLoad];

    // define span for map: how much area will be shown
    MKCoordinateSpan span;
    span.latitudeDelta = 0.002;
    span.longitudeDelta = 0.002;

    // define starting point for map
    CLLocationCoordinate2D start;
    start.latitude = 42.36873056998856;
    start.longitude = -71.11504912376404;

    // create region, consisting of span and location
    MKCoordinateRegion region;
    region.span = span;
    region.center = start;

    // move the map to our location
    [self.mapView setRegion:region animated:YES];

    // create annotation for mather
    CustomAnnotation *mather = [[CustomAnnotation alloc] initWithCoordinate:start];
    mather.title = @"Mather House";
    mather.subtitle = @"The best house";

    // create location for dunster
    CLLocationCoordinate2D dunsterLocation;
    dunsterLocation.latitude = 42.36846289215954;
    dunsterLocation.longitude = -71.11598941345215;

    // create annotation for dunster
    CustomAnnotation *dunster = [[CustomAnnotation alloc] initWithCoordinate:dunsterLocation];
    dunster.title = @"Dunster House";
    dunster.subtitle = @"The worst house";

    // add annotations to map
    [self.mapView addAnnotation:mather];
    [self.mapView addAnnotation:dunster];

    // release annotations
    [mather release];
    [dunster release];

}

/**
 * Just like we did for table cells, define each annotation
 *
 */
- (MKAnnotationView *)mapView:(MKMapView *)mapView viewForAnnotation:(id<MKAnnotation>)annotation
{
    // try to re-use pin annotation view
    MKPinAnnotationView *pin = (MKPinAnnotationView *)[mapView dequeueReusableAnnotationViewWithIdentifier:@"Marker"];

    // none available in the cache
    if (pin == nil) {
        // allocate new pin
        pin = [[[MKPinAnnotationView alloc] initWithAnnotation:annotation reuseIdentifier:@"Marker"] autorelease];

        // add detail disclosure button
        pin.rightCalloutAccessoryView = [UIButton buttonWithType:UIButtonTypeDetailDisclosure];
    }

    // change color to green (Red and Purple also available)
    pin.pinColor = MKPinAnnotationColorGreen;

    // animate pin dropping
    pin.animatesDrop = YES;

    // show callout when tapped
    pin.canShowCallout = YES;

    return pin;
}

/**
 * Fired when user taps detail disclosure button
 * Show pop-up of house tapped
 *
 */
- (void)mapView:(MKMapView *)mapView annotationView:(MKAnnotationView *)view calloutAccessoryControlTapped:(UIControl *)control
{
    // annotation is a property of MKAnnotationView, and we are using our Marker class as the nnotation
    UIAlertView *alert = [[UIAlertView alloc] initWithTitle:@"Detail Button Tapped" message:((CustomAnnotation *)view.annotation).title
                                                   delegate:nil cancelButtonTitle:@"Okay" otherButtonTitles: nil];
    [alert show];
    [alert release];
}

- (void)viewDidUnload
{
    // release properties
    self.mapView = nil;

    [super viewDidUnload];
}

- (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
{
    return (interfaceOrientation == UIInterfaceOrientationPortrait);
}

@end
```

```
//
//  main.m
//  Section3App3
//
//  Created by labuser on 4/11/11.
//  Copyright 2011 Harvard University. All rights reserved.
//

#import <UIKit/UIKit.h>

int main(int argc, char *argv[])
{
    NSAutoreleasePool *pool = [[NSAutoreleasePool alloc] init];
    int retVal = UIApplicationMain(argc, argv, nil, nil);
    [pool release];
    return retVal;
}
```

```
/**
 *  Section3App4AppDelegate.m
 *  Section3App4
 *  Tommy MacWilliam, 2011
 *
 */

#import "Section3App4AppDelegate.h"
#import "Section3App4ViewController.h"

@implementation Section3App4AppDelegate

@synthesize window=_window;
@synthesize viewController=_viewController;

- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    // Override point for customization after application launch.

    self.window.rootViewController = self.viewController;
    [self.window makeKeyAndVisible];
    return YES;
}

- (void)applicationWillResignActive:(UIApplication *)application
{
    /*
     Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions (such as an incoming phone call or SMS message) or when the user quits the application and it begins the transition to the background state.
     Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause the game.
     */
}

- (void)applicationDidEnterBackground:(UIApplication *)application
{
    /*
     Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore your application to its current state in case it is terminated later.
     If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.
     */
}

- (void)applicationWillEnterForeground:(UIApplication *)application
{
    /*
     Called as part of the transition from the background to the inactive state; here you can undo many of the changes made on entering the background.
     */
}

- (void)applicationDidBecomeActive:(UIApplication *)application
{
    /*
     Restart any tasks that were paused (or not yet started) while the application was inactive. If the application was previously in the background, optionally refresh the user interface.
     */
}

- (void)applicationWillTerminate:(UIApplication *)application
{
    /*
     Called when the application is about to terminate.
     Save data if appropriate.
     See also applicationDidEnterBackground:.
     */
}

- (void)dealloc
{
    [_window release];
    [_viewController release];
    [super dealloc];
}

@end
```

```
/**
 *  Section3App4ViewController.m
 *  Section3App4
 *  Tommy MacWilliam, 2011
 *
 *  Use Core Location to get current location
 *
 */

#import "Section3App4ViewController.h"
#import "CoreLocation/CoreLocation.h"

@implementation Section3App4ViewController

@synthesize latitudeLabel=_latitudeLabel, longitudeLabel=_longitudeLabel, locationManager=_locationManager;

- (void)dealloc
{
    // release ivars
    [_locationManager release];
    [_latitudeLabel release];
    [_longitudeLabel release];

    [super dealloc];
}

- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
}

#pragma mark - View lifecycle

- (void)viewDidLoad
{
    [super viewDidLoad];

    // create new location manager
    self.locationManager = [[CLLocationManager alloc] init];
    self.locationManager.delegate = self;
    self.locationManager.desiredAccuracy = kCLLocationAccuracyBest;
    self.locationManager.distanceFilter = kCLDistanceFilterNone;

    // start location manager
    [self.locationManager startUpdatingLocation];
}

/**
```

```
 * Fired when the user has changed location
 *
 */
- (void)locationManager:(CLLocationManager *)manager didUpdateToLocation:(CLLocation *)newLocation fromLocation:(CLLocation *)oldLocation
{
    // update UI with location information
    self.latitudeLabel.text = [NSString stringWithFormat:@"%g", newLocation.coordinate.latitude];
    self.longitudeLabel.text = [NSString stringWithFormat:@"%g", newLocation.coordinate.longitude];
}

- (void)viewDidUnload
{
    // release properties
    self.locationManager = nil;
    self.latitudeLabel = nil;
    self.longitudeLabel = nil;

    [super viewDidUnload];
}

- (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
{
    return (interfaceOrientation == UIInterfaceOrientationPortrait);
}

@end
```

```
//
//  main.m
//  Section3App4
//
//  Created by labuser on 4/12/11.
//  Copyright 2011 Harvard University. All rights reserved.
//

#import <UIKit/UIKit.h>

int main(int argc, char *argv[])
{
    NSAutoreleasePool *pool = [[NSAutoreleasePool alloc] init];
    int retVal = UIApplicationMain(argc, argv, nil, nil);
    [pool release];
    return retVal;
}
```

```
/**
 *  Section3App5AppDelegate.m
 *  Section3App5
 *  Tommy MacWilliam, 2011
 *
 */

#import "Section3App5AppDelegate.h"
#import "Section3App5ViewController.h"

@implementation Section3App5AppDelegate

@synthesize window=_window;
@synthesize viewController=_viewController;

- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    // Override point for customization after application launch.

    self.window.rootViewController = self.viewController;
    [self.window makeKeyAndVisible];
    return YES;
}

- (void)applicationWillResignActive:(UIApplication *)application
{
    /*
     Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions (such as an incoming phone call or SMS message) or when the user quits the application and it begins the transition to the background state.
     Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause the game.
     */
}

- (void)applicationDidEnterBackground:(UIApplication *)application
{
    /*
     Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore your application to its current state in case it is terminated later.
     If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.
     */
}

- (void)applicationWillEnterForeground:(UIApplication *)application
{
    /*
     Called as part of the transition from the background to the inactive state; here you can undo many of the changes made on entering the background.
     */
}

- (void)applicationDidBecomeActive:(UIApplication *)application
{
    /*
     Restart any tasks that were paused (or not yet started) while the application was inactive. If the application was previously in the background, optionally refresh the user interface.
     */
}

- (void)applicationWillTerminate:(UIApplication *)application
{
    /*
     Called when the application is about to terminate.
     Save data if appropriate.
     See also applicationDidEnterBackground:.
     */
}

- (void)dealloc
{
    [_window release];
    [_viewController release];
    [super dealloc];
}

@end
```

```
/**
 *  Section3App5ViewController.m
 *  Section3App5
 *  Tommy MacWilliam, 2011
 *
 *  Play a movie from the web
 *
 */

#import "Section3App5ViewController.h"
#import "MediaPlayer/MediaPlayer.h"

@implementation Section3App5ViewController

@synthesize moviePlayer=_moviePlayer;

- (void)dealloc
{
    // release ivars
    [_moviePlayer release];

    [super dealloc];
}

- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
}

#pragma mark - View lifecycle

- (void)viewDidLoad
{
    [super viewDidLoad];

    // create URL for movie
    NSURL *url = [NSURL URLWithString:@"http://cs76.tv/2011/spring/lectures/0/lecture0.mp4"];

    // create new movie player with set dimensions
    self.moviePlayer = [[MPMoviePlayerController alloc] initWithContentURL:url];
    self.moviePlayer.view.frame = CGRectMake(0, 0, 300, 300);

    // register notification
    [[NSNotificationCenter defaultCenter] addObserver:self selector:@selector(playbackDidFinish)
                                                 name:MPMoviePlayerPlaybackDidFinishNotification object:nil];
    [[NSNotificationCenter defaultCenter] addObserver:self selector:@selector(playbackStateDidChange)
                                                 name:MPMoviePlayerPlaybackStateDidChangeNotification object:nil];

    // add movie player to view and play movie
    [self.view addSubview:self.moviePlayer.view];
    [self.moviePlayer play];
}

/**
 * Callback for playback finishing
 *
 */
- (void)playbackDidFinish
{
    UIAlertView *alert = [[UIAlertView alloc] initWithTitle:@"All done!" message:@"Hope you liked it!" delegate:nil
                                          cancelButtonTitle:@"I did!" otherButtonTitles:@"It was awful", nil];
    [alert show];
    [alert release];
}

/**
 * Callback for playback state change
 *
 */
- (void)playbackStateDidChange
{
    UIAlertView *alert = [[UIAlertView alloc] initWithTitle:@"State Changed!" message:@"Looks like you pressed a button!" delegate:nil
                                          cancelButtonTitle:@"I did!" otherButtonTitles:@"Liar!", nil];
    [alert show];
    [alert release];
}

- (void)viewDidUnload
{
    // release properties
    self.moviePlayer = nil;

    [super viewDidUnload];
}

- (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
{
    return YES;
}

@end
```

```
//
//  main.m
//  Section3App5
//
//  Created by labuser on 4/12/11.
//  Copyright 2011 Harvard University. All rights reserved.
//

#import <UIKit/UIKit.h>

int main(int argc, char *argv[])
{
    NSAutoreleasePool *pool = [[NSAutoreleasePool alloc] init];
    int retVal = UIApplicationMain(argc, argv, nil, nil);
    [pool release];
    return retVal;
}
```

```
/**
 *  FlipsideViewController.m
 *  Section3App6
 *  Tommy MacWilliam, 2011
 *
 */

#import "FlipsideViewController.h"

@implementation FlipsideViewController

@synthesize delegate=_delegate, webView=_webView, url=_url;

- (void)dealloc
{
    // release properties
    self.webView = nil;

    [super dealloc];
}

- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
}

#pragma mark - View lifecycle

- (void)viewDidLoad
{
    [super viewDidLoad];
    self.view.backgroundColor = [UIColor viewFlipsideBackgroundColor];
}

/**
 * Remember, viewWillLoad is NOT called every time the view will appear!
 *
 */
- (void)viewWillAppear:(BOOL)animated
{
    [super viewWillAppear:animated];

    // load URL specified by other controller
    [self.webView loadRequest:[NSURLRequest requestWithURL:[NSURL URLWithString:self.url]]];
}

- (void)viewDidUnload
{
    // release properties
    self.webView = nil;

    [super viewDidUnload];
}

- (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
{
    return (interfaceOrientation == UIInterfaceOrientationPortrait);
}

#pragma mark - Actions

- (IBAction)done:(id)sender
{
    [self.delegate flipsideViewControllerDidFinish:self];
}

@end
```

```
/**
 *  MainViewController.m
 *  Section3App6
 *  Tommy MacWilliam, 2011
 *
 */

#import "MainViewController.h"

@implementation MainViewController

@synthesize pdfButton=_pdfButton, movieButton=_movieButton, youtubeButton=_youtubeButton, wordButton=_wordButton,
    spreadsheetButton=_spreadsheetButton;

- (void)flipsideViewControllerDidFinish:(FlipsideViewController *)controller
{
    [self dismissModalViewControllerAnimated:YES];
}

/**
 * Fired when user taps "Show PDF" button
 *
 */
- (IBAction)showPdf
{
    [self showUrl:@"http://cdn.cs76.net/2011/spring/projects/html5-staff/html5-staff.pdf"];
}

/**
 * Fired when user taps "Show Movie" button
 *
 */
- (IBAction)showMovie
{
    [self showUrl:@"http://cs76.tv/2011/spring/lectures/0/lecture0.mp4"];
}

/**
 * Fired when user taps "Show YouTube" button
 *
 */
- (IBAction)showYoutube
{
    [self showUrl:@"http://www.youtube.com/watch?v=XZ5TajZYW6Y"];
}

/**
 * Fired when user taps "Show Word document" button
 *
 */
- (IBAction)showWord
{
    [self showUrl:@"http://accelerateu.org/assessments/ELA6/Penguins%20Are%20Funny%20Birds.doc"];
}

/**
 * Fired when user taps "Show Excel spreadsheet" button
 *
 */
- (IBAction)showSpreadsheet
{
    [self showUrl:@"http://www.pitt.edu/~kiesling/dude/DudeSurveyData.xls"];
}

/**
 * Send a URL to the FlipsideViewController
 *
 */
- (void)showUrl:(NSString *)url
{
    // create new instance of FlipsideViewController nothing new here
    FlipsideViewController *controller = [[FlipsideViewController alloc] initWithNibName:@"FlipsideView" bundle:nil];
    controller.delegate = self;
    controller.url = url;

    // show new view controller, nothing new here either
    controller.modalTransitionStyle = UIModalTransitionStyleFlipHorizontal;
    [self presentModalViewController:controller animated:YES];

    [controller release];
}

- (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
{
    return (interfaceOrientation == UIInterfaceOrientationPortrait);
}

- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
}

- (void)viewDidUnload
{
    // release properties
    self.pdfButton = nil;
```

```objc
    self.movieButton = nil;
    self.youtubeButton = nil;
    self.wordButton = nil;
    self.spreadsheetButton = nil;

    [super viewDidUnload];
}

- (void)dealloc
{
    // release ivars
    [_pdfButton release];
    [_movieButton release];
    [_youtubeButton release];
    [_wordButton release];
    [_spreadsheetButton release];

    [super dealloc];
}

@end
```

```
/**
 *  Section3App6AppDelegate.m
 *  Section3App6
 *  Tommy MacWilliam, 2011
 *
 */

#import "Section3App6AppDelegate.h"
#import "MainViewController.h"

@implementation Section3App6AppDelegate

@synthesize window=_window;

@synthesize mainViewController=_mainViewController;

- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    // Override point for customization after application launch.
    // Add the main view controller's view to the window and display.
    self.window.rootViewController = self.mainViewController;
    [self.window makeKeyAndVisible];
    return YES;
}

- (void)applicationWillResignActive:(UIApplication *)application
{
    /*
     Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions (such as an incoming phone call or SMS message) or when the user quits the application and it begins the transition to the background state.
     Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause the game.
     */
}

- (void)applicationDidEnterBackground:(UIApplication *)application
{
    /*
     Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore your application to its current state in case it is terminated later.
     If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.
     */
}

- (void)applicationWillEnterForeground:(UIApplication *)application
{
    /*
     Called as part of the transition from the background to the inactive state; here you can undo many of the changes made on entering the
background.
     */
}

- (void)applicationDidBecomeActive:(UIApplication *)application
{
    /*
     Restart any tasks that were paused (or not yet started) while the application was inactive. If the application was previously in the background, optionally refresh the user interface.
     */
}

- (void)applicationWillTerminate:(UIApplication *)application
{
    /*
     Called when the application is about to terminate.
     Save data if appropriate.
     See also applicationDidEnterBackground:.
     */
}

- (void)dealloc
{
    [_window release];
    [_mainViewController release];
    [super dealloc];
}

@end
```

```
//
//  main.m
//  Section3App6
//
//  Created by labuser on 4/12/11.
//  Copyright 2011 Harvard University. All rights reserved.
//

#import <UIKit/UIKit.h>

int main(int argc, char *argv[])
{
    NSAutoreleasePool *pool = [[NSAutoreleasePool alloc] init];
    int retVal = UIApplicationMain(argc, argv, nil, nil);
    [pool release];
    return retVal;
}
```

```
/**
 *  CustomAnnotation.m
 *  Section3App7
 *  Tommy MacWilliam, 2011
 *
 */

#import "CustomAnnotation.h"

@implementation CustomAnnotation

@synthesize coordinate=_coordinate, title=_title, subtitle=_subtitle;

/**
 * Create a new annotation from a given coordinate
 *
 */
- (id)initWithCoordinate:(CLLocationCoordinate2D)coordinate
{
    self = [super init];

    if (self != nil) {
        self.coordinate = coordinate;
    }

    return self;
}

- (void)dealloc
{
    // release properties
    self.title = nil;
    self.subtitle = nil;

    [super dealloc];
}

@end
```

```
/**
 *  Section3App7AppDelegate.m
 *  Section3App7
 *  Tommy MacWilliam, 2011
 *
 */

#import "Section3App7AppDelegate.h"
#import "Section3App7ViewController.h"

@implementation Section3App7AppDelegate

@synthesize window=_window;
@synthesize viewController=_viewController;

- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    // Override point for customization after application launch.

    self.window.rootViewController = self.viewController;
    [self.window makeKeyAndVisible];
    return YES;
}

- (void)applicationWillResignActive:(UIApplication *)application
{
    /*
     Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions (such as an incoming phone call or SMS message) or when the user quits the application and it begins the transition to the background state.
     Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause the game.
     */
}

- (void)applicationDidEnterBackground:(UIApplication *)application
{
    /*
     Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore your application to its current state in case it is terminated later.
     If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.
     */
}

- (void)applicationWillEnterForeground:(UIApplication *)application
{
    /*
     Called as part of the transition from the background to the inactive state; here you can undo many of the changes made on entering the background.
     */
}

- (void)applicationDidBecomeActive:(UIApplication *)application
{
    /*
     Restart any tasks that were paused (or not yet started) while the application was inactive. If the application was previously in the background, optionally refresh the user interface.
     */
}

- (void)applicationWillTerminate:(UIApplication *)application
{
    /*
     Called when the application is about to terminate.
     Save data if appropriate.
     See also applicationDidEnterBackground:.
     */
}

- (void)dealloc
{
    [_window release];
    [_viewController release];
    [super dealloc];
}

@end
```

```
/**
 *  Section3App7ViewController.m
 *  Section3App7
 *  Tommy MacWilliam, 2011
 *
 */

#import "Section3App7ViewController.h"
#import "CustomAnnotation.h"

@implementation Section3App7ViewController

@synthesize mapView=_mapView;

- (void)dealloc
{
    // release ivars
    [_mapView release];

    [super dealloc];
}

- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
}

#pragma mark - View lifecycle

- (void)viewDidLoad
{
    [super viewDidLoad];

    // define span for map: how much area will be shown
    MKCoordinateSpan span;
    span.latitudeDelta = 0.002;
    span.longitudeDelta = 0.002;

    // define starting point for map
    CLLocationCoordinate2D start;
    start.latitude = 42.36873056998856;
    start.longitude = -71.11504912376404;

    // create region, consisting of span and location
    MKCoordinateRegion region;
    region.span = span;
    region.center = start;

    // move the map to our location
    [self.mapView setRegion:region animated:YES];

    // create annotation for mather
    CustomAnnotation *mather = [[CustomAnnotation alloc] initWithCoordinate:start];
    mather.title = @"Mather House";
    mather.subtitle = @"The best house";

    // create location for dunster
    CLLocationCoordinate2D dunsterLocation;
    dunsterLocation.latitude = 42.36846289215954;
    dunsterLocation.longitude = -71.11598941345215;

    // create annotation for dunster
    CustomAnnotation *dunster = [[CustomAnnotation alloc] initWithCoordinate:dunsterLocation];
    dunster.title = @"Dunster House";
    dunster.subtitle = @"The worst house";

    // add annotations to map
    [self.mapView addAnnotation:mather];
    [self.mapView addAnnotation:dunster];

    // create a C array of coordinates to pass to our polyline
    CLLocationCoordinate2D polylineCoordinates[] = { start, dunsterLocation };

    // create polyline connecting both houses and add to map
    MKPolyline *line = [MKPolyline polylineWithCoordinates:polylineCoordinates count:2];
    [self.mapView addOverlay:line];

    // create a circle around the center of the map (distance in meters)
    MKCircle *circle = [MKCircle circleWithCenterCoordinate:start radius:100.0];
    [self.mapView addOverlay:circle];

    // release annotations
    [mather release];
    [dunster release];
}

/**
 * Just like we did for table cells, define each annotation
 *
 */
- (MKAnnotationView *)mapView:(MKMapView *)mapView viewForAnnotation:(id<MKAnnotation>)annotation
{
    // try to re-use pin annotation view
    MKAnnotationView *annotationView = (MKAnnotationView *)[mapView dequeueReusableAnnotationViewWithIdentifier:@"Marker"];

    // none available in the cache
    if (annotationView == nil) {
        // allocate new annotation view
        annotationView = [[[MKAnnotationView alloc] initWithAnnotation:annotation reuseIdentifier:@"Marker"] autorelease];

        // add detail disclosure button
        annotationView.rightCalloutAccessoryView = [UIButton buttonWithType:UIButtonTypeDetailDisclosure];
    }

    // show callout when tapped
    annotationView.canShowCallout = YES;

    // set image for annotation view (no more pins!)
    annotationView.image = [UIImage imageWithContentsOfFile:[[NSBundle mainBundle] pathForResource:@"shield" ofType:@"png"]];

    return annotationView;
}

/**
 * Fired when user taps detail disclosure button
 * Show pop-up of house tapped
 *
 */
- (void)mapView:(MKMapView *)mapView annotationView:(MKAnnotationView *)view calloutAccessoryControlTapped:(UIControl *)control
{
    // annotation is a property of MKAnnotationView, and we are using our Marker class as the nnotation
    UIAlertView *alert = [[UIAlertView alloc] initWithTitle:@"Detail Button Tapped" message:((CustomAnnotation *)view.annotation).title
                                                   delegate:nil cancelButtonTitle:@"Okay" otherButtonTitles: nil];
    [alert show];
    [alert release];
}

/**
 * Just as we did for annotations, define each overlay
 *
 */
-(MKOverlayView *)mapView:(MKMapView *)mapView viewForOverlay:(id)overlay
{
    // overlay is a polyline
    if ([overlay isKindOfClass:[MKPolyline class]]) {
        // create view for polyline
        MKPolylineView *polylineView = [[[MKPolylineView alloc] initWithOverlay:overlay] autorelease];

        // set color and width
        polylineView.strokeColor = [UIColor blueColor];
        polylineView.lineWidth = 2.0;

        return polylineView;
    }

    // overlay is a circle
    else if ([overlay isKindOfClass:[MKCircle class]]) {
        // create view for circle
        MKCircleView *circleView = [[[MKCircleView alloc] initWithOverlay:overlay] autorelease];

        // set color and width
        circleView.strokeColor = [UIColor redColor];
        circleView.lineWidth = 5.0;

        return circleView;
    }

    return nil;
}

- (void)viewDidUnload
{
    // release properties
    self.mapView = nil;

    [super viewDidUnload];
}

- (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
{
    return (interfaceOrientation == UIInterfaceOrientationPortrait);
}

@end
```

```objc
//
//  main.m
//  Section3App7
//
//  Created by labuser on 4/12/11.
//  Copyright 2011 Harvard University. All rights reserved.
//

#import <UIKit/UIKit.h>

int main(int argc, char *argv[])
{
    NSAutoreleasePool *pool = [[NSAutoreleasePool alloc] init];
    int retVal = UIApplicationMain(argc, argv, nil, nil);
    [pool release];
    return retVal;
}
```

```
/**
 *  Section3App0AppDelegate.h
 *  Section3App0
 *  Tommy MacWilliam, 2011
 *
 */

#import <UIKit/UIKit.h>

@class Section3App0ViewController;

@interface Section3App0AppDelegate : NSObject <UIApplicationDelegate> {

}

@property (nonatomic, retain) IBOutlet UIWindow *window;

@property (nonatomic, retain) IBOutlet Section3App0ViewController *viewController;

@end
```

```
/**
 *  Section3App0ViewController.h
 *  Section3App0
 *  Tommy MacWilliam, 2011
 *  Don't forget to #import "MapKit/MapKit.h"!
 *
 */

#import <UIKit/UIKit.h>
#import "MapKit/MapKit.h"

@interface Section3App0ViewController : UIViewController {
    MKMapView *_mapView;
}

@property (nonatomic, retain) IBOutlet MKMapView *mapView;

@end
```

Section3App1/Section3App1/Section3App1AppDelegate.h

```
/**
 *  Section3App1AppDelegate.h
 *  Section3App1
 *  Tommy MacWilliam, 2011
 *
 */

#import <UIKit/UIKit.h>

@class Section3App1ViewController;

@interface Section3App1AppDelegate : NSObject <UIApplicationDelegate> {

}

@property (nonatomic, retain) IBOutlet UIWindow *window;
@property (nonatomic, retain) IBOutlet Section3App1ViewController *viewController;

@end
```

```
/**
 *  Section3App1ViewController.h
 *  Section3App1
 *  Tommy MacWilliam, 2011
 *
 */

#import <UIKit/UIKit.h>
#import "MapKit/MapKit.h"

@interface Section3App1ViewController : UIViewController {
    MKMapView *_mapView;
}

@property (nonatomic, retain) IBOutlet MKMapView *mapView;

@end
```

```
/**
 *  Section3App2AppDelegate.h
 *  Section3App2
 *  Tommy MacWilliam, 2011
 *
 */

#import <UIKit/UIKit.h>

@class Section3App2ViewController;

@interface Section3App2AppDelegate : NSObject <UIApplicationDelegate> {

}

@property (nonatomic, retain) IBOutlet UIWindow *window;
@property (nonatomic, retain) IBOutlet Section3App2ViewController *viewController;

@end
```

```
/**
 *  Section3App2ViewController.h
 *  Section3App2
 *  Tommy MacWilliam, 2011
 *
 */

#import <UIKit/UIKit.h>
#import "MapKit/MapKit.h"

// don't forget to connect both the outlet and delegate in IB!
@interface Section3App2ViewController : UIViewController {
    MKMapView *_mapView;
}

@property (nonatomic, retain) IBOutlet MKMapView *mapView;

@end
```

```
/**
 *  Marker.h
 *  Section3App3
 *  Tommy MacWilliam, 2011
 *
 */

#import <Foundation/Foundation.h>
#import "MapKit/Mapkit.h"

@interface CustomAnnotation : NSObject <MKAnnotation> {
    CLLocationCoordinate2D _coordinate;
    NSString *_title;
    NSString *_subtitle;
}

@property (nonatomic, assign) CLLocationCoordinate2D coordinate;
@property (nonatomic, retain) NSString *title;
@property (nonatomic, retain) NSString *subtitle;

- (id)initWithCoordinate:(CLLocationCoordinate2D)coordinate;

@end
```

Section3App3/Section3App3/Section3App3AppDelegate.h

```
/**
 *  Section3App3AppDelegate.h
 *  Section3App3
 *  Tommy MacWilliam, 2011
 *
 */

#import <UIKit/UIKit.h>

@class Section3App3ViewController;

@interface Section3App3AppDelegate : NSObject <UIApplicationDelegate> {

}

@property (nonatomic, retain) IBOutlet UIWindow *window;
@property (nonatomic, retain) IBOutlet Section3App3ViewController *viewController;

@end
```

```
/**
 *  Section3App3ViewController.h
 *  Section3App3
 *  Tommy MacWilliam, 2011
 *
 */

#import <UIKit/UIKit.h>
#import "MapKit/MapKit.h"

@interface Section3App3ViewController : UIViewController <MKMapViewDelegate> {
    MKMapView *_mapView;
}

@property (nonatomic, retain) IBOutlet MKMapView *mapView;

@end
```

```
/**
 *  Section3App4AppDelegate.h
 *  Section3App4
 *  Tommy MacWilliam, 2011
 *
 */

#import <UIKit/UIKit.h>

@class Section3App4ViewController;

@interface Section3App4AppDelegate : NSObject <UIApplicationDelegate> {

}

@property (nonatomic, retain) IBOutlet UIWindow *window;
@property (nonatomic, retain) IBOutlet Section3App4ViewController *viewController;

@end
```

```
/**
 *  Section3App4ViewController.h
 *  Section3App4
 *  Tommy MacWilliam, 2011
 *
 */

#import <UIKit/UIKit.h>
#import "CoreLocation/CoreLocation.h"

@interface Section3App4ViewController : UIViewController <CLLocationManagerDelegate> {
    UILabel *_latitudeLabel;
    UILabel *_longitudeLabel;
    CLLocationManager *_locationManager;
}

@property (nonatomic, retain) IBOutlet UILabel *latitudeLabel;
@property (nonatomic, retain) IBOutlet UILabel *longitudeLabel;
@property (nonatomic, retain) CLLocationManager *locationManager;

@end
```

```
/**
 *  Section3App5AppDelegate.h
 *  Section3App5
 *  Tommy MacWilliam, 2011
 *
 */

#import <UIKit/UIKit.h>

@class Section3App5ViewController;

@interface Section3App5AppDelegate : NSObject <UIApplicationDelegate> {

}

@property (nonatomic, retain) IBOutlet UIWindow *window;
@property (nonatomic, retain) IBOutlet Section3App5ViewController *viewController;

@end
```

```
/**
 *  Section3App5ViewController.h
 *  Section3App5
 *  Tommy MacWilliam, 2011
 *
 */

#import <UIKit/UIKit.h>
#import "MediaPlayer/MediaPlayer.h"

@interface Section3App5ViewController : UIViewController {
    MPMoviePlayerController *_moviePlayer;
}

@property (nonatomic, retain) MPMoviePlayerController *moviePlayer;

- (void)playbackDidFinish;
- (void)playbackStateDidChange;

@end
```

```
/**
 *  FlipsideViewController.h
 *  Section3App6
 *  Tommy MacWilliam, 2011
 *
 */

#import <UIKit/UIKit.h>

@protocol FlipsideViewControllerDelegate;

@interface FlipsideViewController : UIViewController {
    UIWebView *_webView;
    NSString *_url;
}

@property (nonatomic, assign) id <FlipsideViewControllerDelegate> delegate;
@property (nonatomic, retain) IBOutlet UIWebView *webView;
@property (nonatomic, retain) NSString *url;

- (IBAction)done:(id)sender;

@end


@protocol FlipsideViewControllerDelegate
- (void)flipsideViewControllerDidFinish:(FlipsideViewController *)controller;
@end
```

```
/**
 *  MainViewController.h
 *  Section3App6
 *  Tommy MacWilliam, 2011
 *
 */

#import "FlipsideViewController.h"

@interface MainViewController : UIViewController <FlipsideViewControllerDelegate> {
    UIButton *_pdfButton;
    UIButton *_movieButton;
    UIButton *_youtubeButton;
    UIButton *_wordButton;
    UIButton *_spreadsheetButton;
}

@property (nonatomic, retain) IBOutlet UIButton *pdfButton;
@property (nonatomic, retain) IBOutlet UIButton *movieButton;
@property (nonatomic, retain) IBOutlet UIButton *youtubeButton;
@property (nonatomic, retain) IBOutlet UIButton *wordButton;
@property (nonatomic, retain) IBOutlet UIButton *spreadsheetButton;

- (IBAction)showPdf;
- (IBAction)showMovie;
- (IBAction)showYoutube;
- (IBAction)showWord;
- (IBAction)showSpreadsheet;
- (void)showUrl:(NSString *)url;

@end
```

```
/**
 *  Section3App6AppDelegate.h
 *  Section3App6
 *  Tommy MacWIlliam, 2011
 *
 */

#import <UIKit/UIKit.h>

@class MainViewController;

@interface Section3App6AppDelegate : NSObject <UIApplicationDelegate> {

}

@property (nonatomic, retain) IBOutlet UIWindow *window;
@property (nonatomic, retain) IBOutlet MainViewController *mainViewController;

@end
```

```objc
/**
 *  CustomAnnotation.h
 *  Section3App7
 *  Tommy MacWilliam, 2011
 *
 */

#import <Foundation/Foundation.h>
#import "MapKit/MapKit.h"

@interface CustomAnnotation : NSObject <MKAnnotation> {
    CLLocationCoordinate2D _coordinate;
    NSString *_title;
    NSString *_subtitle;
}

@property (nonatomic, assign) CLLocationCoordinate2D coordinate;
@property (nonatomic, retain) NSString *title;
@property (nonatomic, retain) NSString *subtitle;

- (id)initWithCoordinate:(CLLocationCoordinate2D)coordinate;

@end;
```

```
/**
 *  Section3App7AppDelegate.h
 *  Section3App7
 *  Tommy MacWilliam, 2011
 *
 */

#import <UIKit/UIKit.h>

@class Section3App7ViewController;

@interface Section3App7AppDelegate : NSObject <UIApplicationDelegate> {

}

@property (nonatomic, retain) IBOutlet UIWindow *window;
@property (nonatomic, retain) IBOutlet Section3App7ViewController *viewController;

@end
```

```
/**
 *  Section3App7ViewController.h
 *  Section3App7
 *  Tommy MacWilliam, 2011
 *
 */

#import <UIKit/UIKit.h>
#import "MapKit/MapKit.h"

@interface Section3App7ViewController : UIViewController <MKMapViewDelegate> {
    MKMapView *_mapView;
}

@property (nonatomic, retain) IBOutlet MKMapView *mapView;

@end
```