

```
1. /**
2.  * Section3App0AppDelegate.m
3.  * Section3App0
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import "Section3App0AppDelegate.h"
9.
10. #import "Section3App0ViewController.h"
11.
12. @implementation Section3App0AppDelegate
13.
14.
15. @synthesize window=_window;
16.
17. @synthesize viewController=_viewController;
18.
19. - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
20. {
21.     // Override point for customization after application launch.
22.
23.     self.window.rootViewController = self.viewController;
24.     [self.window makeKeyAndVisible];
25.     return YES;
26. }
27.
28. - (void)applicationWillResignActive:(UIApplication *)application
29. {
30.     /*
31.      Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions
32.      (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.
33.      Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause
34.      the game.
35.      */
36. }
37.
38. - (void)applicationDidEnterBackground:(UIApplication *)application
39. {
40.     /*
41.      Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore
42.      your application to its current state in case it is terminated later.
43.      If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.
44.      */
45. }
```

```
46.     /*
47.     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.
48.     */
49. }
50.
51. - (void)applicationDidBecomeActive:(UIApplication *)application
52. {
53.     /*
54.     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.
55.     */
56. }
57.
58. - (void)applicationWillTerminate:(UIApplication *)application
59. {
60.     /*
61.     Called when the application is about to terminate.
62.     Save data if appropriate.
63.     See also applicationWillDidEnterBackground:.
64.     */
65. }
66.
67. - (void)dealloc
68. {
69.     [_window release];
70.     [_viewController release];
71.     [super dealloc];
72. }
73.
74. @end
```

```
1. /**
2.  * Section3App0ViewController.m
3.  * Section3App0
4.  * Tommy MacWilliam, 2011
5.  *
6.  * Create a fullscreen MKMapView
7.  * Don't forget to add MapKit.framework!
8.  *
9.  */
10.
11. #import "Section3App0ViewController.h"
12.
13. @implementation Section3App0ViewController
14.
15. @synthesize mapView=_mapView;
16.
17. - (void)dealloc
18. {
19.     // release ivars
20.     [_mapView release];
21.
22.     [super dealloc];
23. }
24.
25. - (void)didReceiveMemoryWarning
26. {
27.     [super didReceiveMemoryWarning];
28. }
29.
30. #pragma mark - View lifecycle
31.
32.
33. - (void)viewDidUnload
34. {
35.     // release properties
36.     self.mapView = nil;
37.
38.     [super viewDidUnload];
39. }
40.
41. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
42. {
43.     return (interfaceOrientation == UIInterfaceOrientationPortrait);
44. }
45.
46. @end
```

```
1. //
2. // main.m
3. // Section3App0
4. //
5. // Created by labuser on 4/11/11.
6. // Copyright 2011 Harvard University. All rights reserved.
7. //
8.
9. #import <UIKit/UIKit.h>
10.
11. int main(int argc, char *argv[])
12. {
13.     NSAutoreleasePool *pool = [[NSAutoreleasePool alloc] init];
14.     int retVal = UIApplicationMain(argc, argv, nil, nil);
15.     [pool release];
16.     return retVal;
17. }
```

```
1. /**
2.  * Section3ApplAppDelegate.m
3.  * Section3Appl
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import "Section3ApplAppDelegate.h"
9. #import "Section3ApplViewController.h"
10.
11. @implementation Section3ApplAppDelegate
12.
13. @synthesize window=_window;
14. @synthesize viewController=_viewController;
15.
16. - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
17. {
18.     // Override point for customization after application launch.
19.
20.     self.window.rootViewController = self.viewController;
21.     [self.window makeKeyAndVisible];
22.     return YES;
23. }
24.
25. - (void)applicationWillResignActive:(UIApplication *)application
26. {
27.     /*
28.      Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions
29.      (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.
30.      Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause
31.      the game.
32.      */
33. }
34.
35. - (void)applicationDidEnterBackground:(UIApplication *)application
36. {
37.     /*
38.      Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore
39.      your application to its current state in case it is terminated later.
40.      If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.
41.      */
42. }
43.
44. - (void)applicationWillEnterForeground:(UIApplication *)application
45. {
46.     /*
47.      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
48.      background.
49.      */
50. }
```

```
45.     */
46. }
47.
48. - (void)applicationDidBecomeActive:(UIApplication *)application
49. {
50.     /*
51.      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.
52.     */
53. }
54.
55. - (void)applicationWillTerminate:(UIApplication *)application
56. {
57.     /*
58.      Called when the application is about to terminate.
59.      Save data if appropriate.
60.      See also applicationDidEnterBackground:.
61.     */
62. }
63.
64. - (void)dealloc
65. {
66.     [_window release];
67.     [_viewController release];
68.     [super dealloc];
69. }
70.
71. @end
```

```
1. /**
2.  * Section3App1ViewController.m
3.  * Section3App1
4.  * Tommy MacWilliam, 2011
5.  *
6.  * Create a fullscreen MKMapView centered on a given point
7.  *
8.  */
9.
10. #import "Section3App1ViewController.h"
11. #import "MapKit/MapKit.h"
12.
13. @implementation Section3App1ViewController
14.
15. @synthesize mapView=_mapView;
16.
17. - (void)dealloc
18. {
19.     // release ivars
20.     [_mapView release];
21.
22.     [super dealloc];
23. }
24.
25. - (void)didReceiveMemoryWarning
26. {
27.     [super didReceiveMemoryWarning];
28. }
29.
30. #pragma mark - View lifecycle
31.
32. - (void)viewDidLoad
33. {
34.     [super viewDidLoad];
35.
36.     // define span for map: how much area will be shown
37.     MKCoordinateSpan span;
38.     span.latitudeDelta = 0.002;
39.     span.longitudeDelta = 0.002;
40.
41.     // define starting point for map
42.     CLLocationCoordinate2D start;
43.     start.latitude = 42.36873056998856;
44.     start.longitude = -71.11504912376404;
45.
46.     // create region, consisting of span and location
47.     MKCoordinateRegion region;
48.     region.span = span;
```

```
49.     region.center = start;
50.
51.     // move the map to our location
52.     [self.mapView setRegion:region animated:YES];
53. }
54.
55. - (void)viewDidUnload
56. {
57.     // release properties
58.     self.mapView = nil;
59.
60.     [super viewDidUnload];
61. }
62.
63. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
64. {
65.     return (interfaceOrientation == UIInterfaceOrientationPortrait);
66. }
67.
68. @end
```



```
1. //
2. // main.m
3. // Section3App1
4. //
5. // Created by labuser on 4/11/11.
6. // Copyright 2011 Harvard University. All rights reserved.
7. //
8.
9. #import <UIKit/UIKit.h>
10.
11. int main(int argc, char *argv[])
12. {
13.     NSAutoreleasePool *pool = [[NSAutoreleasePool alloc] init];
14.     int retVal = UIApplicationMain(argc, argv, nil, nil);
15.     [pool release];
16.     return retVal;
17. }
```

```
1. /**
2.  * Section3App2AppDelegate.m
3.  * Section3App2
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import "Section3App2AppDelegate.h"
9. #import "Section3App2ViewController.h"
10.
11. @implementation Section3App2AppDelegate
12.
13. @synthesize window=_window;
14. @synthesize viewController=_viewController;
15.
16. - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
17. {
18.     // Override point for customization after application launch.
19.
20.     self.window.rootViewController = self.viewController;
21.     [self.window makeKeyAndVisible];
22.     return YES;
23. }
24.
25. - (void)applicationWillResignActive:(UIApplication *)application
26. {
27.     /*
28.      Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions
29.      (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.
30.      Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause
31.      the game.
32.      */
33. }
34.
35. - (void)applicationDidEnterBackground:(UIApplication *)application
36. {
37.     /*
38.      Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore
39.      your application to its current state in case it is terminated later.
40.      If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.
41.      */
42. }
43.
44. - (void)applicationWillEnterForeground:(UIApplication *)application
45. {
46.     /*
47.      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
48.      background.
49.      */
50. }
```

```
45.     */
46. }
47.
48. - (void)applicationDidBecomeActive:(UIApplication *)application
49. {
50.     /*
51.      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.
52.     */
53. }
54.
55. - (void)applicationWillTerminate:(UIApplication *)application
56. {
57.     /*
58.      Called when the application is about to terminate.
59.      Save data if appropriate.
60.      See also applicationWillEnterBackground:.
61.     */
62. }
63.
64. - (void)dealloc
65. {
66.     [_window release];
67.     [_viewController release];
68.     [super dealloc];
69. }
70.
71. @end
```

```
1. /**
2.  * Section3App2ViewController.m
3.  * Section3App2
4.  * Tommy MacWilliam, 2011
5.  *
6.  * Create an MKMapView with a single annotation
7.  *
8.  */
9.
10. #import "Section3App2ViewController.h"
11.
12. @implementation Section3App2ViewController
13.
14. @synthesize mapView=_mapView;
15.
16. - (void)dealloc
17. {
18.     // release ivars
19.     [_mapView release];
20.
21.     [super dealloc];
22. }
23.
24. - (void)didReceiveMemoryWarning
25. {
26.     [super didReceiveMemoryWarning];
27. }
28.
29. #pragma mark - View lifecycle
30.
31. - (void)viewDidLoad
32. {
33.     [super viewDidLoad];
34.
35.     // define span for map: how much area will be shown
36.     MKCoordinateSpan span;
37.     span.latitudeDelta = 0.002;
38.     span.longitudeDelta = 0.002;
39.
40.     // define starting point for map
41.     CLLocationCoordinate2D start;
42.     start.latitude = 42.36873056998856;
43.     start.longitude = -71.11504912376404;
44.
45.     // create region, consisting of span and location
46.     MKCoordinateRegion region;
47.     region.span = span;
48.     region.center = start;
```

```
49.
50.     // move the map to our location
51.     [self.mapView setRegion:region animated:YES];
52.
53.     // create new annotation
54.     MKPointAnnotation *annotation = [[MKPointAnnotation alloc] init];
55.     annotation.coordinate = start;
56.     annotation.title = @"Mather House";
57.     annotation.subtitle = @"The best house";
58.
59.     // add annotation to map
60.     [self.mapView addAnnotation:annotation];
61.     [annotation release];
62. }
63.
64. - (void)viewDidUnload
65. {
66.     // release properties
67.     self.mapView = nil;
68.
69.     [super viewDidUnload];
70. }
71.
72. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
73. {
74.     return (interfaceOrientation == UIInterfaceOrientationPortrait);
75. }
76.
77. @end
```

```
1. //
2. // main.m
3. // Section3App2
4. //
5. // Created by labuser on 4/11/11.
6. // Copyright 2011 Harvard University. All rights reserved.
7. //
8.
9. #import <UIKit/UIKit.h>
10.
11. int main(int argc, char *argv[])
12. {
13.     NSAutoreleasePool *pool = [[NSAutoreleasePool alloc] init];
14.     int retVal = UIApplicationMain(argc, argv, nil, nil);
15.     [pool release];
16.     return retVal;
17. }
```

```
1. /**
2.  * Marker.m
3.  * Section3App3
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import "CustomAnnotation.h"
9.
10. @implementation CustomAnnotation
11.
12. @synthesize coordinate=_coordinate, title=_title, subtitle=_subtitle;
13.
14. /**
15.  * Create a new annotation from a given coordinate
16.  *
17.  */
18. - (id)initWithCoordinate:(CLLocationCoordinate2D)coordinate
19. {
20.     self = [super init];
21.
22.     if (self != nil) {
23.         self.coordinate = coordinate;
24.     }
25.
26.     return self;
27. }
28.
29. - (void)dealloc
30. {
31.     // release properties
32.     self.title = nil;
33.     self.subtitle = nil;
34.
35.     [super dealloc];
36. }
37.
38. @end
```

```
1. /**
2.  * Section3App3AppDelegate.m
3.  * Section3App3
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import "Section3App3AppDelegate.h"
9. #import "Section3App3ViewController.h"
10.
11. @implementation Section3App3AppDelegate
12.
13. @synthesize window=_window;
14. @synthesize viewController=_viewController;
15.
16. - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
17. {
18.     // Override point for customization after application launch.
19.
20.     self.window.rootViewController = self.viewController;
21.     [self.window makeKeyAndVisible];
22.     return YES;
23. }
24.
25. - (void)applicationWillResignActive:(UIApplication *)application
26. {
27.     /*
28.      Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions
29.      (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.
30.      Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause
31.      the game.
32.      */
33. }
34.
35. - (void)applicationDidEnterBackground:(UIApplication *)application
36. {
37.     /*
38.      Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore
39.      your application to its current state in case it is terminated later.
40.      If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.
41.      */
42. }
43.
44. - (void)applicationWillEnterForeground:(UIApplication *)application
45. {
46.     /*
47.      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
48.      background.
49.      */
50. }
```



```
45.     */
46. }
47.
48. - (void)applicationDidBecomeActive:(UIApplication *)application
49. {
50.     /*
51.      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.
52.     */
53. }
54.
55. - (void)applicationWillTerminate:(UIApplication *)application
56. {
57.     /*
58.      Called when the application is about to terminate.
59.      Save data if appropriate.
60.      See also applicationWillEnterBackground:.
61.     */
62. }
63.
64. - (void)dealloc
65. {
66.     [_window release];
67.     [_viewController release];
68.     [super dealloc];
69. }
70.
71. @end
```

```
1. /**
2.  * Section3App3ViewController.m
3.  * Section3App3
4.  * Tommy MacWilliam, 2011
5.  *
6.  * Create an MKMapView with several customized annotations
7.  *
8.  */
9.
10. #import "Section3App3ViewController.h"
11. #import "MapKit/MapKit.h"
12. #import "CustomAnnotation.h"
13.
14. @implementation Section3App3ViewController
15.
16. @synthesize mapView=_mapView;
17.
18. - (void)dealloc
19. {
20.     // release ivars
21.     [_mapView release];
22.
23.     [super dealloc];
24. }
25.
26. - (void)didReceiveMemoryWarning
27. {
28.     [super didReceiveMemoryWarning];
29. }
30.
31. #pragma mark - View lifecycle
32.
33. - (void)viewDidLoad
34. {
35.     [super viewDidLoad];
36.
37.     // define span for map: how much area will be shown
38.     MKCoordinateSpan span;
39.     span.latitudeDelta = 0.002;
40.     span.longitudeDelta = 0.002;
41.
42.     // define starting point for map
43.     CLLocationCoordinate2D start;
44.     start.latitude = 42.36873056998856;
45.     start.longitude = -71.11504912376404;
46.
47.     // create region, consisting of span and location
48.     MKCoordinateRegion region;
```

```
49.     region.span = span;
50.     region.center = start;
51.
52.     // move the map to our location
53.     [self.mapView setRegion:region animated:YES];
54.
55.     // create annotation for mather
56.     CustomAnnotation *mather = [[CustomAnnotation alloc] initWithCoordinate:start];
57.     mather.title = @"Mather House";
58.     mather.subtitle = @"The best house";
59.
60.     // create location for dunster
61.     CLLocationCoordinate2D dunsterLocation;
62.     dunsterLocation.latitude = 42.36846289215954;
63.     dunsterLocation.longitude = -71.11598941345215;
64.
65.     // create annotation for dunster
66.     CustomAnnotation *dunster = [[CustomAnnotation alloc] initWithCoordinate:dunsterLocation];
67.     dunster.title = @"Dunster House";
68.     dunster.subtitle = @"The worst house";
69.
70.     // add annotations to map
71.     [self.mapView addAnnotation:mather];
72.     [self.mapView addAnnotation:dunster];
73.
74.     // release annotations
75.     [mather release];
76.     [dunster release];
77.
78. }
79.
80. /**
81.  * Just like we did for table cells, define each annotation
82.  *
83.  */
84. - (MKAnnotationView *)mapView:(MKMapView *)mapView viewForAnnotation:(id<MKAnnotation>)annotation
85. {
86.     // try to re-use pin annotation view
87.     MKPinAnnotationView *pin = (MKPinAnnotationView *)[mapView dequeueReusableAnnotationViewWithIdentifier:@"Marker"];
88.
89.     // none available in the cache
90.     if (pin == nil) {
91.         // allocate new pin
92.         pin = [[[MKPinAnnotationView alloc] initWithAnnotation:annotation reuseIdentifier:@"Marker"] autorelease];
93.
94.         // add detail disclosure button
95.         pin.rightCalloutAccessoryView = [UIButton buttonWithType:UIButtonTypeDetailDisclosure];
96.     }
```

```
97.
98.     // change color to green (Red and Purple also available)
99.     pin.pinColor = MKPinAnnotationColorGreen;
100.
101.     // animate pin dropping
102.     pin.animatesDrop = YES;
103.
104.     // show callout when tapped
105.     pin.canShowCallout = YES;
106.
107.     return pin;
108. }
109.
110. /**
111.  * Fired when user taps detail disclosure button
112.  * Show pop-up of house tapped
113.  *
114.  */
115. - (void)mapView:(MKMapView *)mapView annotationView:(MKAnnotationView *)view calloutAccessoryControlTapped:(UIControl *)control
116. {
117.     // annotation is a property of MKAnnotationView, and we are using our Marker class as the nnotation
118.     UIAlertView *alert = [[UIAlertView alloc] initWithTitle:@"Detail Button Tapped" message:((CustomAnnotation *)view.annotation).title
119.                          delegate:nil cancelButtonTitle:@"Okay" otherButtonTitles: nil];
120.     [alert show];
121.     [alert release];
122. }
123.
124. - (void)viewDidUnload
125. {
126.     // release properties
127.     self.mapView = nil;
128.
129.     [super viewDidUnload];
130. }
131.
132. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
133. {
134.     return (interfaceOrientation == UIInterfaceOrientationPortrait);
135. }
136.
137. @end
```

```
1. //
2. // main.m
3. // Section3App3
4. //
5. // Created by labuser on 4/11/11.
6. // Copyright 2011 Harvard University. All rights reserved.
7. //
8.
9. #import <UIKit/UIKit.h>
10.
11. int main(int argc, char *argv[])
12. {
13.     NSAutoreleasePool *pool = [[NSAutoreleasePool alloc] init];
14.     int retVal = UIApplicationMain(argc, argv, nil, nil);
15.     [pool release];
16.     return retVal;
17. }
```

```
1. /**
2.  * Section3App4AppDelegate.m
3.  * Section3App4
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import "Section3App4AppDelegate.h"
9. #import "Section3App4ViewController.h"
10.
11. @implementation Section3App4AppDelegate
12.
13. @synthesize window=_window;
14. @synthesize viewController=_viewController;
15.
16. - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
17. {
18.     // Override point for customization after application launch.
19.
20.     self.window.rootViewController = self.viewController;
21.     [self.window makeKeyAndVisible];
22.     return YES;
23. }
24.
25. - (void)applicationWillResignActive:(UIApplication *)application
26. {
27.     /*
28.      Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions
29.      (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.
30.      Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause
31.      the game.
32.      */
33. }
34.
35. - (void)applicationDidEnterBackground:(UIApplication *)application
36. {
37.     /*
38.      Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore
39.      your application to its current state in case it is terminated later.
40.      If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.
41.      */
42. }
43.
44. - (void)applicationWillEnterForeground:(UIApplication *)application
45. {
46.     /*
47.      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
48.      background.
49.      */
50. }
```

```
45.     */
46. }
47.
48. - (void)applicationDidBecomeActive:(UIApplication *)application
49. {
50.     /*
51.      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.
52.     */
53. }
54.
55. - (void)applicationWillTerminate:(UIApplication *)application
56. {
57.     /*
58.      Called when the application is about to terminate.
59.      Save data if appropriate.
60.      See also applicationWillEnterBackground:.
61.     */
62. }
63.
64. - (void)dealloc
65. {
66.     [_window release];
67.     [_viewController release];
68.     [super dealloc];
69. }
70.
71. @end
```

```
1. /**
2.  * Section3App4ViewController.m
3.  * Section3App4
4.  * Tommy MacWilliam, 2011
5.  *
6.  * Use Core Location to get current location
7.  *
8.  */
9.
10. #import "Section3App4ViewController.h"
11. #import "CoreLocation/CoreLocation.h"
12.
13. @implementation Section3App4ViewController
14.
15. @synthesize latitudeLabel=_latitudeLabel, longitudeLabel=_longitudeLabel, locationManager=_locationManager;
16.
17. - (void)dealloc
18. {
19.     // release ivars
20.     [_locationManager release];
21.     [_latitudeLabel release];
22.     [_longitudeLabel release];
23.
24.     [super dealloc];
25. }
26.
27. - (void)didReceiveMemoryWarning
28. {
29.     [super didReceiveMemoryWarning];
30. }
31.
32. #pragma mark - View lifecycle
33.
34. - (void)viewDidLoad
35. {
36.     [super viewDidLoad];
37.
38.     // create new location manager
39.     self.locationManager = [[CLLocationManager alloc] init];
40.     self.locationManager.delegate = self;
41.     self.locationManager.desiredAccuracy = kCLLocationAccuracyBest;
42.     self.locationManager.distanceFilter = kCLDistanceFilterNone;
43.
44.     // start location manager
45.     [self.locationManager startUpdatingLocation];
46. }
47.
48. /**
```



```
49.  * Fired when the user has changed location
50.  *
51.  */
52. - (void)locationManager:(CLLocationManager *)manager didUpdateToLocation:(CLLocation *)newLocation fromLocation:(CLLocation *)oldLocation
53. {
54.     // update UI with location information
55.     self.latitudeLabel.text = [NSString stringWithFormat:@"%g", newLocation.coordinate.latitude];
56.     self.longitudeLabel.text = [NSString stringWithFormat:@"%g", newLocation.coordinate.longitude];
57. }
58.
59. - (void)viewDidUnload
60. {
61.     // release properties
62.     self.locationManager = nil;
63.     self.latitudeLabel = nil;
64.     self.longitudeLabel = nil;
65.
66.     [super viewDidUnload];
67. }
68.
69. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
70. {
71.     return (interfaceOrientation == UIInterfaceOrientationPortrait);
72. }
73.
74. @end
```

```
1. //
2. // main.m
3. // Section3App4
4. //
5. // Created by labuser on 4/12/11.
6. // Copyright 2011 Harvard University. All rights reserved.
7. //
8.
9. #import <UIKit/UIKit.h>
10.
11. int main(int argc, char *argv[])
12. {
13.     NSAutoreleasePool *pool = [[NSAutoreleasePool alloc] init];
14.     int retVal = UIApplicationMain(argc, argv, nil, nil);
15.     [pool release];
16.     return retVal;
17. }
```

```
1. /**
2.  * Section3App5AppDelegate.m
3.  * Section3App5
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import "Section3App5AppDelegate.h"
9. #import "Section3App5ViewController.h"
10.
11. @implementation Section3App5AppDelegate
12.
13. @synthesize window=_window;
14. @synthesize viewController=_viewController;
15.
16. - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
17. {
18.     // Override point for customization after application launch.
19.
20.     self.window.rootViewController = self.viewController;
21.     [self.window makeKeyAndVisible];
22.     return YES;
23. }
24.
25. - (void)applicationWillResignActive:(UIApplication *)application
26. {
27.     /*
28.      Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions
29.      (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.
30.      Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause
31.      the game.
32.      */
33. }
34.
35. - (void)applicationDidEnterBackground:(UIApplication *)application
36. {
37.     /*
38.      Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore
39.      your application to its current state in case it is terminated later.
40.      If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.
41.      */
42. }
43.
44. - (void)applicationWillEnterForeground:(UIApplication *)application
45. {
46.     /*
47.      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
48.      background.
49.      */
50. }
```

```
45.     */
46. }
47.
48. - (void)applicationDidBecomeActive:(UIApplication *)application
49. {
50.     /*
51.      Restart any tasks that were paused (or not yet started) while the application was inactive. If the application was previously in the
52.      background, optionally refresh the user interface.
53.     */
54. }
55. - (void)applicationWillTerminate:(UIApplication *)application
56. {
57.     /*
58.      Called when the application is about to terminate.
59.      Save data if appropriate.
60.      See also applicationDidEnterBackground:.
61.     */
62. }
63.
64. - (void)dealloc
65. {
66.     [_window release];
67.     [_viewController release];
68.     [super dealloc];
69. }
70.
71. @end
```

```
1. /**
2.  * Section3App5ViewController.m
3.  * Section3App5
4.  * Tommy MacWilliam, 2011
5.  *
6.  * Play a movie from the web
7.  *
8.  */
9.
10. #import "Section3App5ViewController.h"
11. #import "MediaPlayer/MediaPlayer.h"
12.
13. @implementation Section3App5ViewController
14.
15. @synthesize moviePlayer=_moviePlayer;
16.
17. - (void)dealloc
18. {
19.     // release ivars
20.     [_moviePlayer release];
21.
22.     [super dealloc];
23. }
24.
25. - (void)didReceiveMemoryWarning
26. {
27.     [super didReceiveMemoryWarning];
28. }
29.
30. #pragma mark - View lifecycle
31.
32. - (void)viewDidLoad
33. {
34.     [super viewDidLoad];
35.
36.     // create URL for movie
37.     NSURL *url = [NSURL URLWithString:@"http://cs76.tv/2011/spring/lectures/0/lecture0.mp4"];
38.
39.     // create new movie player with set dimensions
40.     self.moviePlayer = [[MPMoviePlayerController alloc] initWithContentURL:url];
41.     self.moviePlayer.view.frame = CGRectMake(0, 0, 300, 300);
42.
43.     // register notification
44.     [[NSNotificationCenter defaultCenter] addObserver:self selector:@selector(playbackDidFinish)
45.                                             name:MPMoviePlayerPlaybackDidFinishNotification object:nil];
46.     [[NSNotificationCenter defaultCenter] addObserver:self selector:@selector(playbackStateDidChange)
47.                                             name:MPMoviePlayerPlaybackStateDidChangeNotification object:nil];
48.
```

```
49.     // add movie player to view and play movie
50.     [self.view addSubview:self.moviePlayer.view];
51.     [self.moviePlayer play];
52. }
53.
54. /**
55.  * Callback for playback finishing
56.  *
57.  */
58. - (void)playbackDidFinish
59. {
60.     UIAlertView *alert = [[UIAlertView alloc] initWithTitle:@"All done!" message:@"Hope you liked it!" delegate:nil
61.                        cancelButtonTitle:@"I did!" otherButtonTitles:@"It was awful", nil];
62.     [alert show];
63.     [alert release];
64. }
65.
66. /**
67.  * Callback for playback state change
68.  *
69.  */
70. - (void)playbackStateDidChange
71. {
72.     UIAlertView *alert = [[UIAlertView alloc] initWithTitle:@"State Changed!" message:@"Looks like you pressed a button!" delegate:nil
73.                        cancelButtonTitle:@"I did!" otherButtonTitles:@"Liar!", nil];
74.     [alert show];
75.     [alert release];
76. }
77.
78. - (void)viewDidUnload
79. {
80.     // release properties
81.     self.moviePlayer = nil;
82.
83.     [super viewDidUnload];
84. }
85.
86. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
87. {
88.     return YES;
89. }
90.
91. @end
```

```
1. //
2. // main.m
3. // Section3App5
4. //
5. // Created by labuser on 4/12/11.
6. // Copyright 2011 Harvard University. All rights reserved.
7. //
8.
9. #import <UIKit/UIKit.h>
10.
11. int main(int argc, char *argv[])
12. {
13.     NSAutoreleasePool *pool = [[NSAutoreleasePool alloc] init];
14.     int retVal = UIApplicationMain(argc, argv, nil, nil);
15.     [pool release];
16.     return retVal;
17. }
```

```
1. /**
2.  * FlipsideViewController.m
3.  * Section3App6
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import "FlipsideViewController.h"
9.
10. @implementation FlipsideViewController
11.
12. @synthesize delegate=_delegate, webView=_webView, url=_url;
13.
14. - (void)dealloc
15. {
16.     // release properties
17.     self.webView = nil;
18.
19.     [super dealloc];
20. }
21.
22. - (void)didReceiveMemoryWarning
23. {
24.     [super didReceiveMemoryWarning];
25. }
26.
27. #pragma mark - View lifecycle
28.
29. - (void)viewDidLoad
30. {
31.     [super viewDidLoad];
32.     self.view.backgroundColor = [UIColor viewFlipsideBackgroundColor];
33. }
34.
35. /**
36.  * Remember, viewWillAppear is NOT called every time the view will appear!
37.  *
38.  */
39. - (void)viewWillAppear:(BOOL)animated
40. {
41.     [super viewWillAppear:animated];
42.
43.     // load URL specified by other controller
44.     [self.webView loadRequest:[NSURLRequest requestWithURL:[NSURL URLWithString:self.url]]];
45. }
46.
47. - (void)viewDidUnload
48. {
```



```
49.     // release properties
50.     self.webView = nil;
51.
52.     [super viewDidLoad];
53. }
54.
55. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
56. {
57.     return (interfaceOrientation == UIInterfaceOrientationPortrait);
58. }
59.
60. #pragma mark - Actions
61.
62. - (IBAction)done:(id)sender
63. {
64.     [self.delegate flipsideViewControllerDidFinish:self];
65. }
66.
67. @end
```

```
1. /**
2.  * MainViewController.m
3.  * Section3App6
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import "MainViewController.h"
9.
10. @implementation MainViewController
11.
12. @synthesize pdfButton=_pdfButton, movieButton=_movieButton, youtubeButton=_youtubeButton, wordButton=_wordButton,
13.     spreadsheetButton=_spreadsheetButton;
14.
15. - (void)flipsideViewControllerDidFinish:(FlipsideViewController *)controller
16. {
17.     [self dismissModalViewControllerAnimated:YES];
18. }
19.
20. /**
21.  * Fired when user taps "Show PDF" button
22.  *
23.  */
24. - (IBAction)showPdf
25. {
26.     [self showUrl:@"http://cdn.cs76.net/2011/spring/projects/html5-staff/html5-staff.pdf"];
27. }
28.
29. /**
30.  * Fired when user taps "Show Movie" button
31.  *
32.  */
33. - (IBAction)showMovie
34. {
35.     [self showUrl:@"http://cs76.tv/2011/spring/lectures/0/lecture0.mp4"];
36. }
37.
38. /**
39.  * Fired when user taps "Show YouTube" button
40.  *
41.  */
42. - (IBAction)showYoutube
43. {
44.     [self showUrl:@"http://www.youtube.com/watch?v=XZ5TajZYW6Y"];
45. }
46.
47. /**
48.  * Fired when user taps "Show Word document" button
```

```
49.  *
50.  */
51.  - (IBAction)showWord
52.  {
53.      [self showUrl:@"http://accelerateu.org/assessments/ELA6/Penguins%20Are%20Funny%20Birds.doc"];
54.  }
55.
56.  /**
57.   * Fired when user taps "Show Excel spreadsheet" button
58.   *
59.   */
60.  - (IBAction)showSpreadsheet
61.  {
62.      [self showUrl:@"http://www.pitt.edu/~kiesling/dude/DudeSurveyData.xls"];
63.  }
64.
65.  /**
66.   * Send a URL to the FlipsideViewController
67.   *
68.   */
69.  - (void)showUrl:(NSString *)url
70.  {
71.      // create new instance of FlipsideViewController nothing new here
72.      FlipsideViewController *controller = [[FlipsideViewController alloc] initWithNibName:@"FlipsideView" bundle:nil];
73.      controller.delegate = self;
74.      controller.url = url;
75.
76.      // show new view controller, nothing new here either
77.      controller.modalTransitionStyle = UIModalTransitionStyleFlipHorizontal;
78.      [self presentModalViewController:controller animated:YES];
79.
80.      [controller release];
81.  }
82.
83.  - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
84.  {
85.      return (interfaceOrientation == UIInterfaceOrientationPortrait);
86.  }
87.
88.  - (void)didReceiveMemoryWarning
89.  {
90.      [super didReceiveMemoryWarning];
91.  }
92.
93.  - (void)viewDidUnload
94.  {
95.      // release properties
96.      self.pdfButton = nil;
```

```
97.     self.movieButton = nil;
98.     self.youtubeButton = nil;
99.     self.wordButton = nil;
100.    self.spreadsheetButton = nil;
101.
102.    [super viewDidLoad];
103. }
104.
105. - (void)dealloc
106. {
107.     // release ivars
108.     [_pdfButton release];
109.     [_movieButton release];
110.     [_youtubeButton release];
111.     [_wordButton release];
112.     [_spreadsheetButton release];
113.
114.     [super dealloc];
115. }
116.
117. @end
```

```
1. /**
2.  * Section3App6AppDelegate.m
3.  * Section3App6
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import "Section3App6AppDelegate.h"
9. #import "MainViewController.h"
10.
11. @implementation Section3App6AppDelegate
12.
13. @synthesize window=_window;
14.
15. @synthesize mainViewController=_mainViewController;
16.
17. - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
18. {
19.     // Override point for customization after application launch.
20.     // Add the main view controller's view to the window and display.
21.     self.window.rootViewController = self.mainViewController;
22.     [self.window makeKeyAndVisible];
23.     return YES;
24. }
25.
26. - (void)applicationWillResignActive:(UIApplication *)application
27. {
28.     /*
29.      Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions
30.      (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.
31.      Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause
32.      the game.
33.      */
34. }
35.
36. - (void)applicationDidEnterBackground:(UIApplication *)application
37. {
38.     /*
39.      Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore
40.      your application to its current state in case it is terminated later.
41.      If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.
42.      */
43. }
44.
45. - (void)applicationWillEnterForeground:(UIApplication *)application
46. {
47.     /*
48.      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.  
46.     */  
47. }  
48.  
49. - (void)applicationDidBecomeActive:(UIApplication *)application  
50. {  
51.     /*  
52.     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.  
53.     */  
54. }  
55.  
56. - (void)applicationWillTerminate:(UIApplication *)application  
57. {  
58.     /*  
59.     Called when the application is about to terminate.  
60.     Save data if appropriate.  
61.     See also applicationDidEnterBackground:.  
62.     */  
63. }  
64.  
65. - (void)dealloc  
66. {  
67.     [_window release];  
68.     [_mainViewController release];  
69.     [super dealloc];  
70. }  
71.  
72. @end
```

```
1. //
2. // main.m
3. // Section3App6
4. //
5. // Created by labuser on 4/12/11.
6. // Copyright 2011 Harvard University. All rights reserved.
7. //
8.
9. #import <UIKit/UIKit.h>
10.
11. int main(int argc, char *argv[])
12. {
13.     NSAutoreleasePool *pool = [[NSAutoreleasePool alloc] init];
14.     int retVal = UIApplicationMain(argc, argv, nil, nil);
15.     [pool release];
16.     return retVal;
17. }
```

```
1. /**
2.  * CustomAnnotation.m
3.  * Section3App7
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import "CustomAnnotation.h"
9.
10. @implementation CustomAnnotation
11.
12. @synthesize coordinate=_coordinate, title=_title, subtitle=_subtitle;
13.
14. /**
15.  * Create a new annotation from a given coordinate
16.  *
17.  */
18. - (id)initWithCoordinate:(CLLocationCoordinate2D)coordinate
19. {
20.     self = [super init];
21.
22.     if (self != nil) {
23.         self.coordinate = coordinate;
24.     }
25.
26.     return self;
27. }
28.
29. - (void)dealloc
30. {
31.     // release properties
32.     self.title = nil;
33.     self.subtitle = nil;
34.
35.     [super dealloc];
36. }
37.
38. @end
```



```
1. /**
2.  * Section3App7AppDelegate.m
3.  * Section3App7
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import "Section3App7AppDelegate.h"
9. #import "Section3App7ViewController.h"
10.
11. @implementation Section3App7AppDelegate
12.
13. @synthesize window=_window;
14. @synthesize viewController=_viewController;
15.
16. - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
17. {
18.     // Override point for customization after application launch.
19.
20.     self.window.rootViewController = self.viewController;
21.     [self.window makeKeyAndVisible];
22.     return YES;
23. }
24.
25. - (void)applicationWillResignActive:(UIApplication *)application
26. {
27.     /*
28.      Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions
29.      (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.
30.      Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause
31.      the game.
32.      */
33. }
34.
35. - (void)applicationDidEnterBackground:(UIApplication *)application
36. {
37.     /*
38.      Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore
39.      your application to its current state in case it is terminated later.
40.      If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.
41.      */
42. }
43.
44. - (void)applicationWillEnterForeground:(UIApplication *)application
45. {
46.     /*
47.      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
48.      background.
49.      */
50. }
```

```
45.     */
46. }
47.
48. - (void)applicationDidBecomeActive:(UIApplication *)application
49. {
50.     /*
51.      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.
52.     */
53. }
54.
55. - (void)applicationWillTerminate:(UIApplication *)application
56. {
57.     /*
58.      Called when the application is about to terminate.
59.      Save data if appropriate.
60.      See also applicationWillEnterBackground:.
61.     */
62. }
63.
64. - (void)dealloc
65. {
66.     [_window release];
67.     [_viewController release];
68.     [super dealloc];
69. }
70.
71. @end
```

```
1. /**
2.  * Section3App7ViewController.m
3.  * Section3App7
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import "Section3App7ViewController.h"
9. #import "CustomAnnotation.h"
10.
11. @implementation Section3App7ViewController
12.
13. @synthesize mapView=_mapView;
14.
15. - (void)dealloc
16. {
17.     // release ivars
18.     [_mapView release];
19.
20.     [super dealloc];
21. }
22.
23. - (void)didReceiveMemoryWarning
24. {
25.     [super didReceiveMemoryWarning];
26. }
27.
28. #pragma mark - View lifecycle
29.
30. - (void)viewDidLoad
31. {
32.     [super viewDidLoad];
33.
34.     // define span for map: how much area will be shown
35.     MKCoordinateSpan span;
36.     span.latitudeDelta = 0.002;
37.     span.longitudeDelta = 0.002;
38.
39.     // define starting point for map
40.     CLLocationCoordinate2D start;
41.     start.latitude = 42.36873056998856;
42.     start.longitude = -71.11504912376404;
43.
44.     // create region, consisting of span and location
45.     MKCoordinateRegion region;
46.     region.span = span;
47.     region.center = start;
48.
```

```
49.     // move the map to our location
50.     [self.mapView setRegion:region animated:YES];
51.
52.     // create annotation for mather
53.     CustomAnnotation *mather = [[CustomAnnotation alloc] initWithCoordinate:start];
54.     mather.title = @"Mather House";
55.     mather.subtitle = @"The best house";
56.
57.     // create location for dunster
58.     CLLocationCoordinate2D dunsterLocation;
59.     dunsterLocation.latitude = 42.36846289215954;
60.     dunsterLocation.longitude = -71.11598941345215;
61.
62.     // create annotation for dunster
63.     CustomAnnotation *dunster = [[CustomAnnotation alloc] initWithCoordinate:dunsterLocation];
64.     dunster.title = @"Dunster House";
65.     dunster.subtitle = @"The worst house";
66.
67.     // add annotations to map
68.     [self.mapView addAnnotation:mather];
69.     [self.mapView addAnnotation:dunster];
70.
71.     // create a C array of coordinates to pass to our polyline
72.     CLLocationCoordinate2D polylineCoordinates[] = { start, dunsterLocation };
73.
74.     // create polyline connecting both houses and add to map
75.     MKPolyline *line = [MKPolyline polylineWithCoordinates:polylineCoordinates count:2];
76.     [self.mapView addOverlay:line];
77.
78.     // create a circle around the center of the map (distance in meters)
79.     MKCircle *circle = [MKCircle circleWithCenterCoordinate:start radius:100.0];
80.     [self.mapView addOverlay:circle];
81.
82.     // release annotations
83.     [mather release];
84.     [dunster release];
85. }
86.
87. /**
88.  * Just like we did for table cells, define each annotation
89.  *
90.  */
91. - (MKAnnotationView *)mapView:(MKMapView *)mapView viewForAnnotation:(id<MKAnnotation>)annotation
92. {
93.     // try to re-use pin annotation view
94.     MKAnnotationView *annotationView = (MKAnnotationView *)[mapView dequeueReusableAnnotationViewWithIdentifier:@"Marker"];
95.
96.     // none available in the cache
```

```
97.     if (annotationView == nil) {
98.         // allocate new annotation view
99.         annotationView = [[[MKAnnotationView alloc] initWithAnnotation:annotation reuseIdentifier:@"Marker"] autorelease];
100.
101.         // add detail disclosure button
102.         annotationView.rightCalloutAccessoryView = [UIButton buttonWithType:UIButtonTypeDetailDisclosure];
103.     }
104.
105.     // show callout when tapped
106.     annotationView.canShowCallout = YES;
107.
108.     // set image for annotation view (no more pins!)
109.     annotationView.image = [UIImage imageNamed:[NSBundle mainBundle] pathForResource:@"shield" ofType:@"png"];
110.
111.     return annotationView;
112. }
113.
114. /**
115.  * Fired when user taps detail disclosure button
116.  * Show pop-up of house tapped
117.  *
118.  */
119. - (void)mapView:(MKMapView *)mapView annotationView:(MKAnnotationView *)view calloutAccessoryControlTapped:(UIControl *)control
120. {
121.     // annotation is a property of MKAnnotationView, and we are using our Marker class as the nnotation
122.     UIAlertView *alert = [[UIAlertView alloc] initWithTitle:@"Detail Button Tapped" message:((CustomAnnotation *)view.annotation).title
123.                         delegate:nil cancelButtonTitle:@"Okay" otherButtonTitles: nil];
124.     [alert show];
125.     [alert release];
126. }
127.
128. /**
129.  * Just as we did for annotations, define each overlay
130.  *
131.  */
132. -(MKOverlayView *)mapView:(MKMapView *)mapView viewForOverlay:(id)overlay
133. {
134.     // overlay is a polyline
135.     if ([overlay isKindOfClass:[MKPolyline class]]) {
136.         // create view for polyline
137.         MKPolylineView *polylineView = [[[MKPolylineView alloc] initWithOverlay:overlay] autorelease];
138.
139.         // set color and width
140.         polylineView.strokeColor = [UIColor blueColor];
141.         polylineView.lineWidth = 2.0;
142.
143.         return polylineView;
144.     }
```

```
145.
146.     // overlay is a circle
147.     else if ([overlay isKindOfClass:[MKCircle class]]) {
148.         // create view for circle
149.         MKCircleView *circleView = [[MKCircleView alloc] initWithOverlay:overlay] autorelease];
150.
151.         // set color and width
152.         circleView.strokeColor = [UIColor redColor];
153.         circleView.lineWidth = 5.0;
154.
155.         return circleView;
156.     }
157.
158.     return nil;
159. }
160.
161. - (void)viewDidUnload
162. {
163.     // release properties
164.     self.mapView = nil;
165.
166.     [super viewDidUnload];
167. }
168.
169. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
170. {
171.     return (interfaceOrientation == UIInterfaceOrientationPortrait);
172. }
173.
174. @end
```

```
1. //
2. // main.m
3. // Section3App7
4. //
5. // Created by labuser on 4/12/11.
6. // Copyright 2011 Harvard University. All rights reserved.
7. //
8.
9. #import <UIKit/UIKit.h>
10.
11. int main(int argc, char *argv[])
12. {
13.     NSAutoreleasePool *pool = [[NSAutoreleasePool alloc] init];
14.     int retVal = UIApplicationMain(argc, argv, nil, nil);
15.     [pool release];
16.     return retVal;
17. }
```

```
1. /**
2.  * Section3App0AppDelegate.h
3.  * Section3App0
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import <UIKit/UIKit.h>
9.
10. @class Section3App0ViewController;
11.
12. @interface Section3App0AppDelegate : NSObject <UIApplicationDelegate> {
13.
14. }
15.
16. @property (nonatomic, retain) IBOutlet UIWindow *window;
17.
18. @property (nonatomic, retain) IBOutlet Section3App0ViewController *viewController;
19.
20. @end
```



```
1. /**
2.  * Section3App0ViewController.h
3.  * Section3App0
4.  * Tommy MacWilliam, 2011
5.  * Don't forget to #import "MapKit/MapKit.h"!
6.  *
7.  */
8.
9. #import <UIKit/UIKit.h>
10. #import "MapKit/MapKit.h"
11.
12. @interface Section3App0ViewController : UIViewController {
13.     MKMapView *_mapView;
14. }
15.
16. @property (nonatomic, retain) IBOutlet MKMapView *mapView;
17.
18. @end
```

```
1. /**
2.  * Section3ApplAppDelegate.h
3.  * Section3Appl
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import <UIKit/UIKit.h>
9.
10. @class Section3ApplViewController;
11.
12. @interface Section3ApplAppDelegate : NSObject <UIApplicationDelegate> {
13.
14. }
15.
16. @property (nonatomic, retain) IBOutlet UIWindow *window;
17. @property (nonatomic, retain) IBOutlet Section3ApplViewController *viewController;
18.
19. @end
```

```
1. /**
2.  * Section3App1ViewController.h
3.  * Section3App1
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import <UIKit/UIKit.h>
9. #import "MapKit/MapKit.h"
10.
11. @interface Section3App1ViewController : UIViewController {
12.     MKMapView *_mapView;
13. }
14.
15. @property (nonatomic, retain) IBOutlet MKMapView *mapView;
16.
17. @end
```

```
1. /**
2.  * Section3App2AppDelegate.h
3.  * Section3App2
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import <UIKit/UIKit.h>
9.
10. @class Section3App2ViewController;
11.
12. @interface Section3App2AppDelegate : NSObject <UIApplicationDelegate> {
13.
14. }
15.
16. @property (nonatomic, retain) IBOutlet UIWindow *window;
17. @property (nonatomic, retain) IBOutlet Section3App2ViewController *viewController;
18.
19. @end
```

```
1. /**
2.  * Section3App2ViewController.h
3.  * Section3App2
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import <UIKit/UIKit.h>
9. #import "MapKit/MapKit.h"
10.
11. // don't forget to connect both the outlet and delegate in IB!
12. @interface Section3App2ViewController : UIViewController {
13.     MKMapView *_mapView;
14. }
15.
16. @property (nonatomic, retain) IBOutlet MKMapView *mapView;
17.
18. @end
```

```
1. /**
2.  * Marker.h
3.  * Section3App3
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import <Foundation/Foundation.h>
9. #import "MapKit/Mapkit.h"
10.
11. @interface CustomAnnotation : NSObject <MKAnnotation> {
12.     CLLocationCoordinate2D _coordinate;
13.     NSString *_title;
14.     NSString *_subtitle;
15. }
16.
17. @property (nonatomic, assign) CLLocationCoordinate2D coordinate;
18. @property (nonatomic, retain) NSString *title;
19. @property (nonatomic, retain) NSString *subtitle;
20.
21. - (id)initWithCoordinate:(CLLocationCoordinate2D)coordinate;
22.
23. @end
```

```
1. /**
2.  * Section3App3AppDelegate.h
3.  * Section3App3
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import <UIKit/UIKit.h>
9.
10. @class Section3App3ViewController;
11.
12. @interface Section3App3AppDelegate : NSObject <UIApplicationDelegate> {
13.
14. }
15.
16. @property (nonatomic, retain) IBOutlet UIWindow *window;
17. @property (nonatomic, retain) IBOutlet Section3App3ViewController *viewController;
18.
19. @end
```

```
1. /**
2.  * Section3App3ViewController.h
3.  * Section3App3
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import <UIKit/UIKit.h>
9. #import "MapKit/MapKit.h"
10.
11. @interface Section3App3ViewController : UIViewController <MKMapViewDelegate> {
12.     MKMapView *_mapView;
13. }
14.
15. @property (nonatomic, retain) IBOutlet MKMapView *mapView;
16.
17. @end
```



```
1. /**
2.  * Section3App4AppDelegate.h
3.  * Section3App4
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import <UIKit/UIKit.h>
9.
10. @class Section3App4ViewController;
11.
12. @interface Section3App4AppDelegate : NSObject <UIApplicationDelegate> {
13.
14. }
15.
16. @property (nonatomic, retain) IBOutlet UIWindow *window;
17. @property (nonatomic, retain) IBOutlet Section3App4ViewController *viewController;
18.
19. @end
```

```
1. /**
2.  * Section3App4ViewController.h
3.  * Section3App4
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import <UIKit/UIKit.h>
9. #import "CoreLocation/CoreLocation.h"
10.
11. @interface Section3App4ViewController : UIViewController <CLLocationManagerDelegate> {
12.     UILabel *_latitudeLabel;
13.     UILabel *_longitudeLabel;
14.     CLLocationManager *_locationManager;
15. }
16.
17. @property (nonatomic, retain) IBOutlet UILabel *latitudeLabel;
18. @property (nonatomic, retain) IBOutlet UILabel *longitudeLabel;
19. @property (nonatomic, retain) CLLocationManager *locationManager;
20.
21. @end
```

```
1. /**
2.  * Section3App5AppDelegate.h
3.  * Section3App5
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import <UIKit/UIKit.h>
9.
10. @class Section3App5ViewController;
11.
12. @interface Section3App5AppDelegate : NSObject <UIApplicationDelegate> {
13.
14. }
15.
16. @property (nonatomic, retain) IBOutlet UIWindow *window;
17. @property (nonatomic, retain) IBOutlet Section3App5ViewController *viewController;
18.
19. @end
```

```
1. /**
2.  * Section3App5ViewController.h
3.  * Section3App5
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import <UIKit/UIKit.h>
9. #import "MediaPlayer/MediaPlayer.h"
10.
11. @interface Section3App5ViewController : UIViewController {
12.     MPMoviePlayerController *_moviePlayer;
13. }
14.
15. @property (nonatomic, retain) MPMoviePlayerController *moviePlayer;
16.
17. - (void)playbackDidFinish;
18. - (void)playbackStateDidChange;
19.
20. @end
```

```
1. /**
2.  * FlipsideViewController.h
3.  * Section3App6
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import <UIKit/UIKit.h>
9.
10. @protocol FlipsideViewControllerDelegate;
11.
12. @interface FlipsideViewController : UIViewController {
13.     UIWebView *_webView;
14.     NSString *_url;
15. }
16.
17. @property (nonatomic, assign) id <FlipsideViewControllerDelegate> delegate;
18. @property (nonatomic, retain) IBOutlet UIWebView *webView;
19. @property (nonatomic, retain) NSString *url;
20.
21. - (IBAction)done:(id)sender;
22.
23. @end
24.
25.
26. @protocol FlipsideViewControllerDelegate
27. - (void)flipsideViewControllerDidFinish:(FlipsideViewController *)controller;
28. @end
```

```
1. /**
2.  * MainViewController.h
3.  * Section3App6
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import "FlipsideViewController.h"
9.
10. @interface MainViewController : UIViewController <FlipsideViewControllerDelegate> {
11.     UIButton *_pdfButton;
12.     UIButton *_movieButton;
13.     UIButton *_youtubeButton;
14.     UIButton *_wordButton;
15.     UIButton *_spreadsheetButton;
16. }
17.
18. @property (nonatomic, retain) IBOutlet UIButton *pdfButton;
19. @property (nonatomic, retain) IBOutlet UIButton *movieButton;
20. @property (nonatomic, retain) IBOutlet UIButton *youtubeButton;
21. @property (nonatomic, retain) IBOutlet UIButton *wordButton;
22. @property (nonatomic, retain) IBOutlet UIButton *spreadsheetButton;
23.
24. - (IBAction)showPdf;
25. - (IBAction)showMovie;
26. - (IBAction)showYoutube;
27. - (IBAction)showWord;
28. - (IBAction)showSpreadsheet;
29. - (void)showUrl:(NSString *)url;
30.
31. @end
```

```
1. /**
2.  * Section3App6AppDelegate.h
3.  * Section3App6
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import <UIKit/UIKit.h>
9.
10. @class MainViewController;
11.
12. @interface Section3App6AppDelegate : NSObject <UIApplicationDelegate> {
13.
14. }
15.
16. @property (nonatomic, retain) IBOutlet UIWindow *window;
17. @property (nonatomic, retain) IBOutlet MainViewController *mainViewController;
18.
19. @end
```

```
1. /**
2.  * CustomAnnotation.h
3.  * Section3App7
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import <Foundation/Foundation.h>
9. #import "MapKit/MapKit.h"
10.
11. @interface CustomAnnotation : NSObject <MKAnnotation> {
12.     CLLocationCoordinate2D _coordinate;
13.     NSString *_title;
14.     NSString *_subtitle;
15. }
16.
17. @property (nonatomic, assign) CLLocationCoordinate2D coordinate;
18. @property (nonatomic, retain) NSString *title;
19. @property (nonatomic, retain) NSString *subtitle;
20.
21. - (id)initWithCoordinate:(CLLocationCoordinate2D)coordinate;
22.
23. @end;
```



```
1. /**
2.  * Section3App7AppDelegate.h
3.  * Section3App7
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import <UIKit/UIKit.h>
9.
10. @class Section3App7ViewController;
11.
12. @interface Section3App7AppDelegate : NSObject <UIApplicationDelegate> {
13.
14. }
15.
16. @property (nonatomic, retain) IBOutlet UIWindow *window;
17. @property (nonatomic, retain) IBOutlet Section3App7ViewController *viewController;
18.
19. @end
```

```
1. /**
2.  * Section3App7ViewController.h
3.  * Section3App7
4.  * Tommy MacWilliam, 2011
5.  *
6.  */
7.
8. #import <UIKit/UIKit.h>
9. #import "MapKit/MapKit.h"
10.
11. @interface Section3App7ViewController : UIViewController <MKMapViewDelegate> {
12.     MKMapView *_mapView;
13. }
14.
15. @property (nonatomic, retain) IBOutlet MKMapView *mapView;
16.
17. @end
```