

```
1. //
2. // AppDelegate.h
3. // Gestures
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates gestures.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @class ViewController;
15.
16. @interface AppDelegate : UIResponder <UIApplicationDelegate>
17.
18. @property (strong, nonatomic) ViewController *viewController;
19. @property (strong, nonatomic) UIWindow *window;
20.
21. @end
```

```
1. //
2. // AppDelegate.m
3. // Gestures
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates gestures.
10. //
11.
12. #import "AppDelegate.h"
13.
14. #import "ViewController.h"
15.
16. @implementation AppDelegate
17.
18. @synthesize viewController=_viewController;
19. @synthesize window=_window;
20.
21. - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
22. {
23.     self.window = [[UIWindow alloc] initWithFrame:[[UIScreen mainScreen] bounds]];
24.     self.viewController = [[ViewController alloc] initWithNibName:@"ViewController" bundle:nil];
25.     self.window.rootViewController = self.viewController;
26.     [self.window makeKeyAndVisible];
27.     return YES;
28. }
29.
30. @end
```

```
1. //
2. // ViewController.h
3. // Gestures
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates gestures.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @interface ViewController : UIViewController <UIAlertViewDelegate>
15. @end
```

```
1. //
2. // ViewController.m
3. // Gestures
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates gestures.
10. //
11.
12. #import "ViewController.h"
13.
14.
15. @interface ViewController ()
16.
17. // private properties
18. @property (assign, nonatomic, readwrite) BOOL alertInProgress;
19. @property (nonatomic, readwrite, weak) IBOutlet UIImageView *imageView;
20. @property (assign, nonatomic, readwrite) int index;
21. @property (nonatomic, readwrite, strong) NSArray *tommies;
22.
23. // private methods
24. - (void)alertView:(UIAlertView *)alertView didDismissWithButtonIndex:(NSInteger)buttonIndex;
25. - (void)handleLongPress:(UILongPressGestureRecognizer *)sender;
26. - (void)handleSwipe:(UISwipeGestureRecognizer *)sender;
27.
28. @end
29.
30.
31. @implementation ViewController
32.
33. // synthesized properties
34. @synthesize alertInProgress=_alertInProgress;
35. @synthesize imageView=_imageView;
36. @synthesize index=_index;
37. @synthesize tommies=_tommies;
38.
39. - (void)alertView:(UIAlertView *)alertView didDismissWithButtonIndex:(NSInteger)buttonIndex
40. {
41.     self.alertInProgress = NO;
42. }
43.
44. - (void)handleLongPress:(UILongPressGestureRecognizer *)sender
45. {
46.     if (!self.alertInProgress) {
47.         self.alertInProgress = YES;
48.         UIAlertView *alertView = [[UIAlertView alloc] initWithTitle:@"Hey!"
```

```
49.             message:@"Stop that."
50.             delegate:self
51.             cancelButtonTitle:@"Fine"
52.             otherButtonTitles:nil];
53.     [alertView show];
54. }
55. }
56.
57. - (void)handleSwipe:(UISwipeGestureRecognizer *)sender
58. {
59.     // handle swipe
60.     UISwipeGestureRecognizerDirection direction = [(UISwipeGestureRecognizer *)sender direction];
61.     switch (direction)
62.     {
63.         // ignore up, down
64.         case UISwipeGestureRecognizerDirectionUp:
65.         case UISwipeGestureRecognizerDirectionDown:
66.             break;
67.
68.         // left
69.         case UISwipeGestureRecognizerDirectionLeft:
70.             self.index = (self.index + 1) % [self.tommies count];
71.             break;
72.
73.         // right
74.         case UISwipeGestureRecognizerDirectionRight:
75.             self.index = (self.index + [self.tommies count] - 1) % [self.tommies count];
76.             break;
77.     }
78.
79.     // update Tommy
80.     self.imageView.image = [UIImage imageNamed:[self.tommies objectAtIndex:self.index]];
81. }
82.
83. - (id)initWithNibName:(NSString *)nibNameOrNil bundle:(NSBundle *)nibBundleOrNil
84. {
85.     if (self = [super initWithNibName:nibNameOrNil bundle:nibBundleOrNil])
86.     {
87.         // keep track of alerts
88.         self.alertInProgress = NO;
89.
90.         // prepare Tommies
91.         self.tommies = [[NSArray alloc] initWithObjects:@"tommy1.jpg", @"tommy2.jpg", @"tommy3.jpg", nil];
92.         self.index = 0;
93.     }
94.     return self;
95. }
96.
```

```
97. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
98. {
99.     return (interfaceOrientation == UIInterfaceOrientationPortrait);
100. }
101.
102. - (void)viewDidLoad
103. {
104.     [super viewDidLoad];
105.
106.     // load image
107.     self.imageView.image = [UIImage imageNamed:[self.tommies objectAtIndex:self.index]];
108.
109.     // listen for long press
110.     UILongPressGestureRecognizer *longpressGesture = [[UILongPressGestureRecognizer alloc] initWithTarget:self action:@selector(
handleLongPress:)];
111.     [self.imageView addGestureRecognizer:longpressGesture];
112.
113.     // listen for right swipe
114.     UISwipeGestureRecognizer *swipeGesture = [[UISwipeGestureRecognizer alloc] initWithTarget:self action:@selector(handleSwipe:)];
115.     swipeGesture.direction = UISwipeGestureRecognizerDirectionRight;
116.     [self.imageView addGestureRecognizer:swipeGesture];
117.
118.     // listen for left swipe
119.     swipeGesture = [[UISwipeGestureRecognizer alloc] initWithTarget:self action:@selector(handleSwipe:)];
120.     swipeGesture.direction = UISwipeGestureRecognizerDirectionLeft;
121.     [self.imageView addGestureRecognizer:swipeGesture];
122. }
123.
124. @end
```

```
1. //
2. // AppDelegate.h
3. // Hola1
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates localized nibs.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @class ViewController;
15.
16. @interface AppDelegate : UIResponder <UIApplicationDelegate>
17.
18. @property (strong, nonatomic) ViewController *viewController;
19. @property (strong, nonatomic) UIWindow *window;
20.
21. @end
```

```
1. //
2. // AppDelegate.m
3. // Hola1
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates localized nibs.
10. //
11.
12. #import "AppDelegate.h"
13. #import "ViewController.h"
14.
15. @implementation AppDelegate
16.
17. @synthesize viewController=_viewController;
18. @synthesize window=_window;
19.
20. - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
21. {
22.     self.window = [[UIWindow alloc] initWithFrame:[[UIScreen mainScreen] bounds]];
23.     self.viewController = [[ViewController alloc] initWithNibName:@"ViewController" bundle:nil];
24.     self.window.rootViewController = self.viewController;
25.     [self.window makeKeyAndVisible];
26.     return YES;
27. }
28.
29. @end
```



```
1. //
2. // ViewController.h
3. // Hola1
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates localized nibs.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @interface ViewController : UIViewController
15. @end
```

```
1. //
2. // ViewController.m
3. // Hola1
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates localized nibs.
10. //
11.
12. #import "ViewController.h"
13.
14. @implementation ViewController
15.
16. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
17. {
18.     return YES;
19. }
20.
21. @end
```

```
1. //
2. // AppDelegate.h
3. // Hola2
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates localized strings.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @class ViewController;
15.
16. @interface AppDelegate : UIResponder <UIApplicationDelegate>
17.
18. @property (strong, nonatomic) ViewController *viewController;
19. @property (strong, nonatomic) UIWindow *window;
20.
21. @end
```

```
1. //
2. // AppDelegate.m
3. // Hola2
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates localized strings.
10. //
11.
12. #import "AppDelegate.h"
13.
14. #import "ViewController.h"
15.
16. @implementation AppDelegate
17.
18. @synthesize viewController=_viewController;
19. @synthesize window=_window;
20.
21. - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
22. {
23.     self.window = [[UIWindow alloc] initWithFrame:[[UIScreen mainScreen] bounds]];
24.     self.viewController = [[ViewController alloc] initWithNibName:@"ViewController" bundle:nil];
25.     self.window.rootViewController = self.viewController;
26.     [self.window makeKeyAndVisible];
27.     return YES;
28. }
29.
30. @end
```

```
1. //
2. // ViewController.h
3. // Hola2
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates localized strings.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @interface ViewController : UIViewController
15. @end
```

```
1. //
2. // ViewController.m
3. // Hola2
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates localized strings.
10. //
11.
12. #import "ViewController.h"
13.
14.
15. @interface ViewController ()
16. @property (nonatomic, readwrite, weak) IBOutlet UILabel *label;
17. @end
18.
19.
20. @implementation ViewController
21.
22. @synthesize label=_label;
23.
24. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
25. {
26.     return YES;
27. }
28.
29. - (void)viewDidLoad
30. {
31.     [super viewDidLoad];
32.
33.     // localize greeting
34.     self.label.text = NSLocalizedString(@"GREETING", nil);
35. }
36.
37. @end
```

```
1. //
2. // AppDelegate.h
3. // MasterDetail
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Master-Detail Application.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @interface AppDelegate : UIResponder <UIApplicationDelegate>
15.
16. @property (strong, nonatomic) UINavigationController *navigationController;
17. @property (strong, nonatomic) UIWindow *window;
18.
19. @end
```

```
1. //
2. // AppDelegate.m
3. // MasterDetail
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Master-Detail Application.
10. //
11.
12. #import "AppDelegate.h"
13.
14. #import "MasterViewController.h"
15.
16. @implementation AppDelegate
17.
18. @synthesize navigationController=_navigationController;
19. @synthesize window=_window;
20.
21. - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
22. {
23.     self.window = [[UIWindow alloc] initWithFrame:[[UIScreen mainScreen] bounds]];
24.     MasterViewController *masterViewController = [[MasterViewController alloc] initWithNibName:@"MasterViewController" bundle:nil];
25.     self.navigationController = [[UINavigationController alloc] initWithRootViewController:masterViewController];
26.     self.window.rootViewController = self.navigationController;
27.     [self.window makeKeyAndVisible];
28.     return YES;
29. }
30.
31. @end
```



```
1. //
2. // DetailViewController.h
3. // MasterDetail
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Master-Detail Application.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @interface DetailViewController : UIViewController
15.
16. @property (strong, nonatomic) IBOutlet UILabel *detailDescriptionLabel;
17. @property (strong, nonatomic) id detailItem;
18.
19. @end
```

```
1. //
2. // DetailViewController.m
3. // MasterDetail
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Master-Detail Application.
10. //
11.
12. #import "DetailViewController.h"
13.
14. @interface DetailViewController ()
15. - (void)configureView;
16. @end
17.
18. @implementation DetailViewController
19.
20. @synthesize detailItem=_detailItem;
21. @synthesize detailDescriptionLabel=_detailDescriptionLabel;
22.
23. #pragma mark - Managing the detail item
24.
25. - (void)setDetailItem:(id)newDetailItem
26. {
27.     if (_detailItem != newDetailItem) {
28.         _detailItem = newDetailItem;
29.
30.         // Update the view.
31.         [self configureView];
32.     }
33. }
34.
35. - (void)configureView
36. {
37.     // Update the user interface for the detail item.
38.     if (self.detailItem) {
39.         self.detailDescriptionLabel.text = [self.detailItem description];
40.     }
41. }
42.
43. - (void)viewDidLoad
44. {
45.     [super viewDidLoad];
46.     [self configureView];
47. }
48.
```

```
49. - (void)viewDidUnload
50. {
51.     [super viewDidUnload];
52.     self.detailDescriptionLabel = nil;
53. }
54.
55. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
56. {
57.     return (interfaceOrientation != UIInterfaceOrientationPortraitUpsideDown);
58. }
59.
60. - (id)initWithNibName:(NSString *)nibNameOrNil bundle:(NSBundle *)nibBundleOrNil
61. {
62.     self = [super initWithNibName:nibNameOrNil bundle:nibBundleOrNil];
63.     if (self) {
64.         self.title = NSLocalizedString(@"Detail", @"Detail");
65.     }
66.     return self;
67. }
68.
69. @end
```

```
1. //
2. // MasterViewController.h
3. // MasterDetail
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Master-Detail Application.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @class DetailViewController;
15.
16. @interface MasterViewController : UITableViewController
17. @property (strong, nonatomic) DetailViewController *detailViewController;
18. @end
```

```
1. //
2. // MasterViewController.m
3. // MasterDetail
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Master-Detail Application.
10. //
11.
12. #import "MasterViewController.h"
13.
14. #import "DetailViewController.h"
15.
16. @interface MasterViewController () {
17.     NSMutableArray *_objects;
18. }
19. @end
20.
21. @implementation MasterViewController
22.
23. @synthesize detailViewController = _detailViewController;
24.
25. - (id)initWithNibName:(NSString *)nibNameOrNil bundle:(NSBundle *)nibBundleOrNil
26. {
27.     self = [super initWithNibName:nibNameOrNil bundle:nibBundleOrNil];
28.     if (self) {
29.         self.title = NSLocalizedString(@"Master", @"Master");
30.     }
31.     return self;
32. }
33.
34. - (void)viewDidLoad
35. {
36.     [super viewDidLoad];
37.     self.navigationItem.leftBarButtonItem = self.editButtonItem;
38.     UIBarButtonItem *addButton = [[UIBarButtonItem alloc] initWithBarButtonSystemItem:UIBarButtonSystemItemAdd target:self action:@selector(
insertNewObject:)];
39.     self.navigationItem.rightBarButtonItem = addButton;
40. }
41.
42. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
43. {
44.     return (interfaceOrientation != UIInterfaceOrientationPortraitUpsideDown);
45. }
46.
47. - (void)insertNewObject:(id)sender
```

```
48. {
49.     if (!_objects) {
50.         _objects = [[NSMutableArray alloc] init];
51.     }
52.     [_objects insertObject:[NSDate date] atIndex:0];
53.     NSIndexPath *indexPath = [NSIndexPath indexPathForRow:0 inSection:0];
54.     [self.tableView insertRowsAtIndexPaths:[NSArray arrayWithObject:indexPath] withRowAnimation:UITableViewRowAnimationAutomatic];
55. }
56.
57. #pragma mark - Table View
58.
59. - (NSInteger)numberOfSectionsInTableView:(UITableView *)tableView
60. {
61.     return 1;
62. }
63.
64. - (NSInteger)tableView:(UITableView *)tableView numberOfRowsInSection:(NSInteger)section
65. {
66.     return _objects.count;
67. }
68.
69. // Customize the appearance of table view cells.
70. - (UITableViewCell *)tableView:(UITableView *)tableView cellForRowAtIndexPath:(NSIndexPath *)indexPath
71. {
72.     static NSString *CellIdentifier = @"Cell";
73.
74.     UITableViewCell *cell = [tableView dequeueReusableCellWithIdentifier:CellIdentifier];
75.     if (cell == nil) {
76.         cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleDefault reuseIdentifier:CellIdentifier];
77.         cell.accessoryType = UITableViewCellAccessoryDisclosureIndicator;
78.     }
79.
80.     NSDate *object = [_objects objectAtIndex:indexPath.row];
81.     cell.textLabel.text = [object description];
82.     return cell;
83. }
84.
85. - (BOOL)tableView:(UITableView *)tableView canEditRowAtIndexPath:(NSIndexPath *)indexPath
86. {
87.     // Return NO if you do not want the specified item to be editable.
88.     return YES;
89. }
90.
91. - (void)tableView:(UITableView *)tableView commitEditingStyle:(UITableViewCellEditingStyle)editingStyle forRowAtIndexPath:(NSIndexPath *)indexPath
92. {
93.     if (editingStyle == UITableViewCellEditingStyleDelete) {
94.         [_objects removeObjectAtIndex:indexPath.row];
```

```
95.         [tableView deleteRowsAtIndexPaths:[NSArray arrayWithObject:indexPath] withRowAnimation:UITableViewRowAnimationFade];
96.     }
97. }
98.
99. - (void)tableView:(UITableView *)tableView didSelectRowAtIndexPath:(NSIndexPath *)indexPath
100. {
101.     if (!self.detailViewController) {
102.         self.detailViewController = [[DetailViewController alloc] initWithNibName:@"DetailViewController" bundle:nil];
103.     }
104.     NSDate *object = [_objects objectAtIndex:indexPath.row];
105.     self.detailViewController.detailItem = object;
106.     [self.navigationController pushViewController:self.detailViewController animated:YES];
107. }
108.
109. @end
```

```
1. //
2. // AppDelegate.h
3. // Nib2
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Single View Application with a UIAlertViewDelegate.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @class ViewController;
15.
16. @interface AppDelegate : UIResponder <UIApplicationDelegate>
17.
18. @property (strong, nonatomic) ViewController *viewController;
19. @property (strong, nonatomic) UIWindow *window;
20.
21. @end
```



```
1. //
2. // AppDelegate.m
3. // Nib2
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Single View Application with a UIAlertViewDelegate.
10. //
11.
12. #import "AppDelegate.h"
13. #import "ViewController.h"
14.
15. @implementation AppDelegate
16.
17. @synthesize viewController=_viewController;
18. @synthesize window=_window;
19.
20. - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
21. {
22.     self.window = [[UIWindow alloc] initWithFrame:[[UIScreen mainScreen] bounds]];
23.     self.viewController = [[ViewController alloc] initWithNibName:@"ViewController" bundle:nil];
24.     self.window.rootViewController = self.viewController;
25.     [self.window makeKeyAndVisible];
26.     return YES;
27. }
28.
29. @end
```

```
1. //
2. // ViewController.h
3. // Nib2
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Single View Application with a UIAlertViewDelegate.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @interface ViewController : UIViewController <UIAlertViewDelegate>
15.
16. @property (nonatomic, strong) IBOutlet UITextField *textField;
17.
18. - (IBAction)go:(id)sender;
19.
20. @end
```

```
1. //
2. // ViewController.m
3. // Nib2
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Single View Application with a UIAlertViewDelegate.
10. //
11.
12. #import "ViewController.h"
13.
14. @implementation ViewController
15.
16. @synthesize textField=_textField;
17.
18. - (void)alertView:(UIAlertView *)alertView didDismissWithButtonIndex:(NSInteger)buttonIndex
19. {
20.     // clear text field
21.     self.textField.text = nil;
22. }
23.
24. - (IBAction)go:(id)sender
25. {
26.     // hide keyboard
27.     [self.textField resignFirstResponder];
28.
29.     // show alert
30.     NSString *s = [NSString stringWithFormat:@"Hello, %@", self.textField.text];
31.     UIAlertView *alert = [[UIAlertView alloc] initWithTitle:@"Hello"
32.                                     message:s
33.                                     delegate:self
34.                                     cancelButtonTitle:@"Thanks!"
35.                                     otherButtonTitles:nil];
36.     [alert show];
37. }
38.
39. @end
```

```
1. //
2. // AppDelegate.h
3. // NoNib
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Single View Application with a view implemented without a nib (or storyboard).
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @class ViewController;
15.
16. @interface AppDelegate : UIResponder <UIApplicationDelegate>
17.
18. @property (strong, nonatomic) ViewController *viewController;
19. @property (strong, nonatomic) UIWindow *window;
20.
21. @end
```

```
1. //
2. // AppDelegate.m
3. // NoNib
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Single View Application with a view implemented without a nib (or storyboard).
10. //
11.
12. #import "AppDelegate.h"
13. #import "ViewController.h"
14.
15. @implementation AppDelegate
16.
17. @synthesize viewController=_viewController;
18. @synthesize window=_window;
19.
20. - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
21. {
22.     self.window = [[UIWindow alloc] initWithFrame:[[UIScreen mainScreen] bounds]];
23.     self.viewController = [[ViewController alloc] init];
24.     self.window.rootViewController = self.viewController;
25.     [self.window makeKeyAndVisible];
26.     return YES;
27. }
28.
29. @end
```

```
1. //
2. // ViewController.h
3. // NoNib
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Single View Application with a view implemented without a nib (or storyboard).
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @interface ViewController : UIViewController
15. @end
```

```
1. //
2. // ViewController.m
3. // NoNib
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Single View Application with a view implemented without a nib (or storyboard).
10. //
11.
12. #import "ViewController.h"
13.
14.
15. @interface ViewController ()
16.
17. // private property
18. @property (nonatomic, readwrite, strong) UITextField *textField;
19.
20. // private method
21. - (void)go:(id)sender;
22.
23. @end
24.
25.
26. @implementation ViewController
27.
28. @synthesize textField=_textField;
29.
30. #pragma mark - UIViewController
31.
32. - (void)loadView
33. {
34.     // create view
35.     self.view = [[UIView alloc] initWithFrame:[UIScreen mainScreen].applicationFrame];
36.     self.view.backgroundColor = [UIColor whiteColor];
37.
38.     // create text field
39.     CGRect frame = CGRectMake(20, 20, 280, 31);
40.     self.textField = [[UITextField alloc] initWithFrame:frame];
41.     self.textField.autocapitalizationType = UITextAutocapitalizationTypeWords;
42.     self.textField.autocorrectionType = UITextAutocorrectionTypeNo;
43.     self.textField.borderStyle = UITextBorderStyleRoundedRect;
44.     self.textField.contentVerticalAlignment = UIControlContentVerticalAlignmentCenter;
45.     self.textField.font = [UIFont systemFontOfSize:[UIFont systemFontOfSize]];
46.     self.textField.placeholder = @"Name";
47.     self.textField.returnKeyType = UIReturnKeyGo;
48.
```

```
49. // listen for Did End on Exit
50. [self.textField addTarget:self action:@selector(go:) forControlEvents:UIControlEventEditingDidEndOnExit];
51.
52. // add text field to view
53. [self.view addSubview:self.textField];
54.
55. // create button
56. frame = CGRectMake(124, 59, 72, 37);
57. UIButton *button = [UIButton buttonWithType:UIButtonTypeRoundedRect];
58. button.frame = frame;
59. [button setTitle:@"Go" forState:UIControlStateNormal];
60.
61. // listen for Touch Up Inside
62. [button addTarget:self action:@selector(go:) forControlEvents:UIControlEventTouchUpInside];
63.
64. // add button to view
65. [self.view addSubview:button];
66. }
67.
68. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
69. {
70.     return (interfaceOrientation == UIInterfaceOrientationPortrait);
71. }
72.
73. #pragma mark - UIAlertViewDelegate
74.
75. - (void)alertView:(UIAlertView *)alertView didDismissWithButtonIndex:(NSInteger)buttonIndex
76. {
77.     // clear text field
78.     self.textField.text = nil;
79. }
80.
81. #pragma mark - Actions
82.
83. - (void)go:(id)sender
84. {
85.     // hide keyboard
86.     [self.textField resignFirstResponder];
87.
88.     // show alert
89.     NSString *s = [NSString stringWithFormat:@"Hello, %@", self.textField.text];
90.     UIAlertView *alert = [[UIAlertView alloc] initWithTitle:@"Hello"
91.                                     message:s
92.                                     delegate:self
93.                                     cancelButtonTitle:@"Thanks!"
94.                                     otherButtonTitles:nil];
95.     [alert show];
96. }
```



97.

98. @end

```
1. //
2. // AppDelegate.h
3. // Paddle
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates Core Graphics with a rectangle.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @class ViewController;
15.
16. @interface AppDelegate : UIResponder <UIApplicationDelegate>
17.
18. @property (strong, nonatomic) ViewController *viewController;
19. @property (strong, nonatomic) UIWindow *window;
20.
21. @end
```

```
1. //
2. // AppDelegate.m
3. // Paddle
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates Core Graphics with a rectangle.
10. //
11.
12. #import "AppDelegate.h"
13. #import "ViewController.h"
14.
15. @implementation AppDelegate
16.
17. @synthesize viewController=_viewController;
18. @synthesize window=_window;
19.
20. - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
21. {
22.     self.window = [[UIWindow alloc] initWithFrame:[[UIScreen mainScreen] bounds]];
23.     self.viewController = [[ViewController alloc] initWithNibName:@"ViewController" bundle:nil];
24.     self.window.rootViewController = self.viewController;
25.     [self.window makeKeyAndVisible];
26.     return YES;
27. }
28.
29. @end
```

```
1. //
2. // PaddleView.h
3. // Paddle
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates Core Graphics with a rectangle.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @interface PaddleView : UIView
15. @end
```

```
1. //
2. // PaddleView.m
3. // Paddle
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates Core Graphics with a rectangle.
10. //
11.
12. #import "PaddleView.h"
13.
14. @implementation PaddleView
15.
16. - (void)drawRect:(CGRect)rect
17. {
18.     CGRect square = CGRectMake(0.0f, 0.0f, 10.0f, 60.0f);
19.     [[UIColor whiteColor] set];
20.     UIRectFill(square);
21. }
22.
23. @end
```

```
1. //
2. // ViewController.h
3. // Paddle
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates Core Graphics with a rectangle.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @interface ViewController : UIViewController
15. @end
```

```
1. //
2. // ViewController.m
3. // Paddle
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates Core Graphics with a rectangle.
10. //
11.
12. #import "ViewController.h"
13. #import "PaddleView.h"
14.
15.
16. @interface ViewController ()
17.
18. // private property
19. @property (nonatomic, retain) PaddleView *paddleView;
20.
21. @end
22.
23.
24. @implementation ViewController
25.
26. @synthesize paddleView=_paddleView;
27.
28. - (id)initWithNibName:(NSString *)nibNameOrNil bundle:(NSBundle *)nibBundleOrNil
29. {
30.     if (self = [super initWithNibName:nibNameOrNil bundle:nibBundleOrNil])
31.     {
32.         // add paddle
33.         self.paddleView = [[PaddleView alloc] initWithFrame:CGRectMake(10.0f, 10.0f, 10.0f, 60.0f)];
34.         [self.view addSubview:self.paddleView];
35.     }
36.     return self;
37. }
38.
39. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
40. {
41.     return (interfaceOrientation == UIInterfaceOrientationLandscapeRight);
42. }
43.
44. - (void)touchesMoved:(NSSet *)touches withEvent:(UIEvent *)event
45. {
46.     // follow user's finger vertically with paddle
47.     UITouch *touch = [[event allTouches] anyObject];
48.     CGPoint location = [touch locationInView:self.view];
```

```
49.     self.paddleView.center = CGPointMake(self.paddleView.center.x, location.y);
50. }
51.
52. @end
```



```
1. //
2. // AppDelegate.h
3. // Plist
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates property lists.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @class ViewController;
15.
16. @interface AppDelegate : UIResponder <UIApplicationDelegate>
17.
18. @property (strong, nonatomic) ViewController *viewController;
19. @property (strong, nonatomic) UIWindow *window;
20.
21. @end
```

```
1. //
2. // AppDelegate.m
3. // Plist
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates property lists.
10. //
11.
12. #import "AppDelegate.h"
13. #import "ViewController.h"
14.
15. @implementation AppDelegate
16.
17. @synthesize viewController=_viewController;
18. @synthesize window=_window;
19.
20. - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
21. {
22.     self.window = [[UIWindow alloc] initWithFrame:[[UIScreen mainScreen] bounds]];
23.     self.viewController = [[ViewController alloc] initWithNibName:@"ViewController" bundle:nil];
24.     self.window.rootViewController = self.viewController;
25.     [self.window makeKeyAndVisible];
26.     return YES;
27. }
28.
29. @end
```

```
1. //
2. // ViewController.h
3. // Plist
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates property lists.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @interface ViewController : UIViewController
15. @end
```

```
1. //
2. // ViewController.m
3. // Plist
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates property lists.
10. //
11.
12. #import "ViewController.h"
13.
14.
15. @interface ViewController ()
16.
17. // private property
18. @property (nonatomic, readwrite, strong) NSArray *words;
19.
20. @end
21.
22.
23. @implementation ViewController
24.
25. @synthesize words=_words;
26.
27. - (id)initWithNibName:(NSString *)nibNameOrNil bundle:(NSBundle *)nibBundleOrNil
28. {
29.     if (self = [super initWithNibName:nibNameOrNil bundle:nibBundleOrNil])
30.     {
31.         // load words
32.         NSString *path = [[NSBundle mainBundle] pathForResource:@"small" ofType:@"plist"];
33.         NSArray *words = [[NSArray alloc] initWithContentsOfFile:path];
34.         self.words = words;
35.     }
36.     return self;
37. }
38.
39. - (NSInteger)numberOfSectionsInTableView:(UITableView *)tableView
40. {
41.     return 1;
42. }
43.
44. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
45. {
46.     return (interfaceOrientation == UIInterfaceOrientationPortrait);
47. }
48.
```

```
49. - (UITableViewCell *)tableView:(UITableView *)tableView cellForRowAtIndexPath:(NSIndexPath *)indexPath
50. {
51.     // allocate cell, reusing if possible
52.     static NSString *CellIdentifier = @"Cell";
53.     UITableViewCell *cell = [tableView dequeueReusableCellWithIdentifier:CellIdentifier];
54.     if (cell == nil)
55.         cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleDefault reuseIdentifier:CellIdentifier];
56.
57.     // configure cell
58.     cell.selectionStyle = UITableViewCellSelectionStyleNone;
59.     cell.textLabel.text = [self.words objectAtIndex:indexPath.row];
60.
61.     return cell;
62. }
63.
64. - (NSInteger)tableView:(UITableView *)tableView numberOfRowsInSection:(NSInteger)section
65. {
66.     return [self.words count];
67. }
68.
69. @end
```

```
1. //
2. // AppDelegate.h
3. // Pong
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates Core Graphics.
10. // Inspired by iTennis by Brandon Trebitowski.
11. //
12.
13. #import <UIKit/UIKit.h>
14.
15. @class ViewController;
16.
17. @interface AppDelegate : UIResponder <UIApplicationDelegate>
18.
19. @property (strong, nonatomic) ViewController *viewController;
20. @property (strong, nonatomic) UIWindow *window;
21.
22. @end
```

```
1. //
2. // AppDelegate.m
3. // Pong
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates Core Graphics.
10. // Inspired by iTennis by Brandon Trebitowski.
11. //
12.
13. #import "AppDelegate.h"
14.
15. #import "ViewController.h"
16.
17. @implementation AppDelegate
18.
19. @synthesize viewController=_viewController;
20. @synthesize window=_window;
21.
22. - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
23. {
24.     self.window = [[UIWindow alloc] initWithFrame:[[UIScreen mainScreen] bounds]];
25.     self.viewController = [[ViewController alloc] initWithNibName:@"ViewController" bundle:nil];
26.     self.window.rootViewController = self.viewController;
27.     [self.window makeKeyAndVisible];
28.     return YES;
29. }
30.
31. - (void)applicationWillResignActive:(UIApplication *)application
32. {
33.     [self.viewController kickoff];
34. }
35.
36. @end
```

```
1. //
2. // PongView.h
3. // Pong
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates Core Graphics.
10. // Inspired by iTennis by Brandon Trebitowski.
11. //
12.
13. #import <UIKit/UIKit.h>
14.
15. @interface PongView : UIView
16. @end
```



```
1. //
2. // PongView.m
3. // Pong
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates Core Graphics.
10. // Inspired by iTennis by Brandon Trebitowski.
11. //
12.
13. #import "PongView.h"
14.
15. @implementation PongView
16.
17. - (void)drawRect:(CGRect)rect
18. {
19.     // get context
20.     CGContextRef context = UIGraphicsGetCurrentContext();
21.
22.     // draw dashed midfield line
23.     CGFloat dashes[] = {1,1};
24.     CGContextSetLineDash(context, 0.0, dashes, 2);
25.     CGContextSetStrokeColorWithColor(context, [UIColor whiteColor].CGColor);
26.     CGContextSetLineWidth(context, 5.0f);
27.     CGContextMoveToPoint(context, 240.0f, 0.0f);
28.     CGContextAddLineToPoint(context, 240.0f, 320.0f);
29.     CGContextStrokePath(context);
30. }
31.
32. @end
```

```
1. //
2. // ViewController.h
3. // Pong
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates Core Graphics.
10. // Inspired by iTennis by Brandon Trebitowski.
11. //
12.
13. #import <UIKit/UIKit.h>
14.
15. @interface ViewController : UIViewController
16. - (void)kickoff;
17. @end
```

```
1. //
2. // ViewController.m
3. // Pong
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates Core Graphics.
10. // Inspired by iTennis by Brandon Trebitowski.
11. //
12.
13. #import "ViewController.h"
14.
15.
16. // default velocity
17. static const float VELOCITY = 10.0f;
18.
19.
20. @interface ViewController () {
21. @private
22. CGPoint _velocity;
23. }
24.
25. // private properties
26. @property (nonatomic, readwrite, weak) IBOutlet UIImageView *ball;
27. @property (nonatomic, readwrite, weak) IBOutlet UILabel *labelLeft;
28. @property (nonatomic, readwrite, weak) IBOutlet UILabel *labelRight;
29. @property (nonatomic, readwrite, weak) IBOutlet UIImageView *paddleLeft;
30. @property (nonatomic, readwrite, weak) IBOutlet UIImageView *paddleRight;
31. @property (assign, nonatomic, readwrite) BOOL paused;
32. @property (assign, nonatomic, readwrite) NSInteger scoreLeft;
33. @property (assign, nonatomic, readwrite) NSInteger scoreRight;
34.
35. @end
36.
37.
38. @implementation ViewController
39.
40. @synthesize ball=_ball;
41. @synthesize labelLeft=_labelLeft;
42. @synthesize labelRight=_labelRight;
43. @synthesize paddleLeft=_paddleLeft;
44. @synthesize paddleRight=_paddleRight;
45. @synthesize paused=_paused;
46. @synthesize scoreLeft=_scoreLeft;
47. @synthesize scoreRight=_scoreRight;
48.
```

```
49.
50. - (id)initWithNibName:(NSString *)nibNameOrNil bundle:(NSBundle *)nibBundleOrNil
51. {
52.     if (self = [super initWithNibName:nibNameOrNil bundle:nibBundleOrNil])
53.     {
54.         // initialize scores
55.         self.scoreLeft = 0;
56.         self.scoreRight = 0;
57.
58.         // initialize ball's velocity
59.         _velocity = CGPointMake(VELOCITY, VELOCITY);
60.
61.         // schedule movement
62.         [NSTimer scheduledTimerWithTimeInterval:0.05 target:self selector:@selector(play) userInfo:nil repeats:YES];
63.     }
64.     return self;
65. }
66.
67. - (void)kickoff
68. {
69.     // pause
70.     self.paused = YES;
71.
72.     // update scores
73.     self.labelLeft.text = [NSString stringWithFormat:@"%u", self.scoreLeft];
74.     self.labelRight.text = [NSString stringWithFormat:@"%u", self.scoreRight];
75.
76.     // center ball
77.     self.ball.center = CGPointMake(240.0f, 160.0f);
78.
79.     // align paddles
80.     self.paddleLeft.center = CGPointMake(25.0f, 160.0f);
81.     self.paddleRight.center = CGPointMake(455.0f, 160.0f);
82. }
83.
84. - (void)play
85. {
86.     // check whether paused
87.     if (self.paused)
88.         return;
89.
90.     // move ball
91.     self.ball.center = CGPointMake(self.ball.center.x + _velocity.x, self.ball.center.y + _velocity.y);
92.
93.     // detect goals
94.     if (self.ball.center.x < 5) {
95.         self.scoreRight++;
96.         [self kickoff];
```

```
97.     }
98.     else if (self.view.bounds.size.width - 5 < self.ball.center.x) {
99.         self.scoreLeft++;
100.        [self kickoff];
101.    }
102.
103.    // bounce off of top and bottom walls
104.    if (self.ball.center.y < 5 || self.view.bounds.size.height - 5 < self.ball.center.y) {
105.        _velocity.y = -_velocity.y;
106.    }
107.
108.    // bounce off of left paddle
109.    if (CGRectIntersectsRect(self.ball.frame, self.paddleLeft.frame)) {
110.        if (self.paddleLeft.center.x < self.ball.center.x) {
111.            _velocity.x = -_velocity.x;
112.        }
113.    }
114.
115.    // bounce off of right paddle
116.    if (CGRectIntersectsRect(self.ball.frame, self.paddleRight.frame)) {
117.        if (self.ball.center.x < self.paddleRight.center.x) {
118.            _velocity.x = -_velocity.x;
119.        }
120.    }
121.
122.    // move opponent as ball approaches
123.    if (_velocity.x < 0) {
124.
125.        // move down
126.        if (self.ball.center.y < self.paddleLeft.center.y) {
127.            self.paddleLeft.center = CGPointMake(self.paddleLeft.center.x, self.paddleLeft.center.y - 10.0f);
128.        }
129.
130.        // move up
131.        else if (self.ball.center.y > self.paddleLeft.center.y) {
132.            self.paddleLeft.center = CGPointMake(self.paddleLeft.center.x, self.paddleLeft.center.y + 10.0f);
133.        }
134.    }
135. }
136.
137. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
138. {
139.     return (interfaceOrientation == UIInterfaceOrientationLandscapeRight);
140. }
141.
142. - (void)touchesBegan:(NSSet *)touches withEvent:(UIEvent *)event
143. {
144.     // unpause if paused
```

```
145.     if (self.paused)
146.         self.paused = NO;
147.     }
148.
149. - (void)touchesMoved:(NSSet *)touches withEvent:(UIEvent *)event
150. {
151.     // follow user's finger vertically with left paddle
152.     UITouch *touch = [[event allTouches] anyObject];
153.     CGPoint location = [touch locationInView:self.view];
154.     self.paddleRight.center = CGPointMake(self.paddleRight.center.x, location.y);
155. }
156.
157. - (void)viewDidAppear:(BOOL)animated
158. {
159.     // play ball!
160.     [self kickoff];
161. }
162.
163. @end
```

```
1. //
2. // AppDelegate.h
3. // Transformations
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates affine transformations of Tommy.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @class ViewController;
15.
16. @interface AppDelegate : UIResponder <UIApplicationDelegate>
17.
18. @property (strong, nonatomic) ViewController *viewController;
19. @property (strong, nonatomic) UIWindow *window;
20.
21. @end
```

```
1. //
2. // AppDelegate.m
3. // Transformations
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates affine transformations of Tommy.
10. //
11.
12. #import "AppDelegate.h"
13.
14. #import "ViewController.h"
15.
16. @implementation AppDelegate
17.
18. @synthesize viewController=_viewController;
19. @synthesize window=_window;
20.
21. - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
22. {
23.     self.window = [[UIWindow alloc] initWithFrame:[[UIScreen mainScreen] bounds]];
24.     self.viewController = [[ViewController alloc] initWithNibName:@"ViewController" bundle:nil];
25.     self.window.rootViewController = self.viewController;
26.     [self.window makeKeyAndVisible];
27.     return YES;
28. }
29.
30. @end
```



```
1. //
2. // ViewController.h
3. // Transformations
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates affine transformations of Tommy.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @interface ViewController : UIViewController
15. @end
```

```
1. //
2. // ViewController.m
3. // Transformations
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates affine transformations of Tommy.
10. //
11.
12. #import "ViewController.h"
13.
14.
15. @interface ViewController () {
16. @private
17.     CGPoint _translation;
18. }
19.
20. // private properties
21. @property (nonatomic, readwrite, weak) IBOutlet UIImageView *imageView;
22. @property (assign, nonatomic, readwrite) float scale;
23. @property (nonatomic, readwrite, strong) NSArray *tommies;
24.
25. // private methods
26. - (void)handlePan:(UIPanGestureRecognizer *)sender;
27. - (void)handlePinch:(UIPinchGestureRecognizer *)sender;
28.
29. @end
30.
31.
32. @implementation ViewController
33.
34. @synthesize imageView=_imageView;
35. @synthesize scale=_scale;
36. @synthesize tommies=_tommies;
37.
38. - (id)initWithNibName:(NSString *)nibNameOrNil bundle:(NSBundle *)nibBundleOrNil
39. {
40.     if (self = [super initWithNibName:nibNameOrNil bundle:nibBundleOrNil])
41.     {
42.         // prepare Tommy
43.         self.scale = 1.0;
44.         _translation.x = 0.0;
45.         _translation.y = 0.0;
46.     }
47.     return self;
48. }
```

```
49.
50. - (void)handlePan:(UIPanGestureRecognizer *)sender
51. {
52.     // translate Tommy
53.     CGPoint translation = [sender translationInView:self.imageView];
54.     CGAffineTransform scale = CGAffineTransformMakeScale(self.scale, self.scale);
55.     CGAffineTransform translate = CGAffineTransformMakeTranslation(_translation.x + translation.x * self.scale, _translation.y + translation.y
    * self.scale);
56.     sender.view.transform = CGAffineTransformConcat(scale, translate);
57.
58.     // remember translation once done panning
59.     if (sender.state == UIGestureRecognizerStateEnded)
60.     {
61.         _translation.x += translation.x * self.scale;
62.         _translation.y += translation.y * self.scale;
63.     }
64. }
65.
66. - (void)handlePinch:(UIPinchGestureRecognizer *)sender
67. {
68.     // scale Tommy
69.     CGFloat factor = [sender scale];
70.     CGAffineTransform scale = CGAffineTransformMakeScale(self.scale * factor, self.scale * factor);
71.     CGAffineTransform translate = CGAffineTransformMakeTranslation(_translation.x, _translation.y);
72.     sender.view.transform = CGAffineTransformConcat(scale, translate);
73.
74.     // remember scale once done pinching
75.     if (sender.state == UIGestureRecognizerStateEnded)
76.         self.scale *= factor;
77. }
78.
79. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
80. {
81.     return (interfaceOrientation == UIInterfaceOrientationPortrait);
82. }
83.
84. - (void)viewDidLoad
85. {
86.     [super viewDidLoad];
87.
88.     // listen for pan
89.     UIPanGestureRecognizer *panGesture = [[UIPanGestureRecognizer alloc] initWithTarget:self action:@selector(handlePan:)];
90.     [self.imageView addGestureRecognizer:panGesture];
91.
92.     // listen for pinch
93.     UIPinchGestureRecognizer *pinchGesture = [[UIPinchGestureRecognizer alloc] initWithTarget:self action:@selector(handlePinch:)];
94.     [self.imageView addGestureRecognizer:pinchGesture];
95. }
```

96.

97. @end