

```
1. //
2. // Account.h
3. // ATM
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Simulates an ATM (that only accepts deposits).
10. //
11.
12. #import <Foundation/Foundation.h>
13.
14. @interface Account : NSObject {
15. }
16.
17. @property (assign, nonatomic) unsigned long long balance;
18.
19. @end
```

```
1. //
2. // Account.m
3. // ATM
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Simulates an ATM (that only accepts deposits).
10. //
11.
12. #import "Account.h"
13.
14. @implementation Account
15.
16. @synthesize balance=_balance;
17.
18. - (id)init
19. {
20.     if (self = [super init]) {
21.         self.balance = 0;
22.     }
23.     return self;
24. }
25.
26. @end
```

```
1. //
2. // AppDelegate.h
3. // ATM
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Simulates an ATM (that only accepts deposits).
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. // ATM
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Simulates an ATM (that only accepts deposits).
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. // ATM
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Simulates an ATM (that only accepts deposits).
10. //
11.
12. #import <UIKit/UIKit.h>
13. #import "Account.h"
14.
15. @interface ViewController : UIViewController {
16. }
17.
18. @property (nonatomic, strong) Account *account;
19. @property (assign, nonatomic) unsigned long long amount;
20. @property (nonatomic, strong) IBOutlet UILabel *balanceLabel;
21. @property (nonatomic, strong) IBOutlet UILabel *depositLabel;
22.
23. - (IBAction)clear:(id)sender;
24. - (IBAction)deposit:(id)sender;
25. - (IBAction)digit:(id)sender;
26. - (void)show;
27.
28. @end
```

```
1. //
2. // ViewController.m
3. // ATM
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Simulates an ATM (that only accepts deposits).
10. //
11.
12. #import "ViewController.h"
13.
14. @implementation ViewController
15.
16. @synthesize account=_account;
17. @synthesize amount=_amount;
18. @synthesize balanceLabel=_balanceLabel;
19. @synthesize depositLabel=_depositLabel;
20.
21. - (IBAction)clear:(id)sender
22. {
23.     // clear input
24.     self.amount = 0;
25.     [self show];
26. }
27.
28. - (IBAction)deposit:(id)sender
29. {
30.     // deposit amount
31.     self.account.balance += self.amount;
32.
33.     // clear input
34.     self.amount = 0;
35.     [self show];
36. }
37.
38. - (IBAction)digit:(id)sender
39. {
40.     // append digit to amount
41.     UIButton *b = (UIButton *)sender;
42.     self.amount = self.amount * 10 + b.tag;
43.     [self show];
44. }
45.
46. - (void)show
47. {
48.     // show balance
```

```
49.     self.balanceLabel.text = [NSString stringWithFormat:@"%$.11u", self.account.balance];
50.
51.     // show input
52.     self.depositLabel.text = [NSString stringWithFormat:@"%$.11u", self.amount];
53. }
54.
55. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
56. {
57.     return (interfaceOrientation == UIInterfaceOrientationPortrait);
58. }
59.
60. - (void)viewDidLoad
61. {
62.     // create account
63.     self.account = [[Account alloc] init];
64.     [self show];
65. }
66.
67. @end
```

```
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. // 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

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

```
1. //
2. // AppDelegate.m
3. // Utility
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Utility Application.
10. //
11.
12. #import "AppDelegate.h"
13. #import "MainViewController.h"
14.
15. @implementation AppDelegate
16.
17. @synthesize mainWindowController = _mainViewController;
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.mainViewController = [[MainViewController alloc] initWithNibName:@"MainViewController" bundle:nil];
24.     self.window.rootViewController = self.mainViewController;
25.     [self.window makeKeyAndVisible];
26.     return YES;
27. }
28.
29. @end
```

```
1. //
2. // FlipsideViewController.h
3. // Utility
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Utility Application.
10. //
11.
12. #import <UIKit/UIKit.h>
13.
14. @class FlipsideViewController;
15.
16. @protocol FlipsideViewControllerDelegate
17. - (void)flipsideViewControllerDidFinish:(FlipsideViewController *)controller;
18. @end
19.
20. @interface FlipsideViewController : UIViewController
21.
22. @property (weak, nonatomic) id <FlipsideViewControllerDelegate> delegate;
23.
24. - (IBAction)done:(id)sender;
25.
26. @end
```

```
1. //
2. // FlipsideViewController.m
3. // Utility
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Utility Application.
10. //
11.
12. #import "FlipsideViewController.h"
13.
14. @implementation FlipsideViewController
15.
16. @synthesize delegate = _delegate;
17.
18. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
19. {
20.     return (interfaceOrientation != UIInterfaceOrientationPortraitUpsideDown);
21. }
22.
23. #pragma mark - Actions
24.
25. - (IBAction)done:(id)sender
26. {
27.     [self.delegate flipsideViewControllerDidFinish:self];
28. }
29.
30. @end
```

```
1. //
2. // MainViewController.h
3. // Utility
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Utility Application.
10. //
11.
12. #import "FlipsideViewController.h"
13.
14. @interface MainViewController : UIViewController <FlipsideViewControllerDelegate>
15.
16. - (IBAction)showInfo:(id)sender;
17.
18. @end
```

```
1. //
2. // MainViewController.m
3. // Utility
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates a Utility Application.
10. //
11.
12. #import "MainViewController.h"
13.
14. @implementation MainViewController
15.
16. - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
17. {
18.     return (interfaceOrientation != UIInterfaceOrientationPortraitUpsideDown);
19. }
20.
21. #pragma mark - Flipside View
22.
23. - (void)flipsideViewControllerDidFinish:(FlipsideViewController *)controller
24. {
25.     [self dismissModalViewControllerAnimated:YES];
26. }
27.
28. - (IBAction)showInfo:(id)sender
29. {
30.     FlipsideViewController *controller = [[FlipsideViewController alloc] initWithNibName:@"FlipsideViewController" bundle:nil];
31.     controller.delegate = self;
32.     controller.modalTransitionStyle = UIModalTransitionStyleFlipHorizontal;
33.     [self presentModalViewController:controller animated:YES];
34. }
35.
36. @end
```