```objc
 1: //
 2: //  RectangleViewController.h
 3: //  Rectangle
 4: //
 5: //  Created by Gabriel Parriaux on 10.12.12.
 6: //  Copyright (c) 2012 gymo. All rights reserved.
 7: //
 8:
 9: #import <UIKit/UIKit.h>
10: #import "RectangleView.h"
11:
12: @interface RectangleViewController : UIViewController
13:
14: @property (weak, nonatomic) IBOutlet RectangleView *rectangleView;
15:
16: @end
```

```objc
 1: //
 2: //  RectangleViewController.m
 3: //  Rectangle
 4: //
 5: //  Created by Gabriel Parriaux on 10.12.12.
 6: //  Copyright (c) 2012 gymo. All rights reserved.
 7: //
 8:
 9: #import "RectangleViewController.h"
10:
11: @interface RectangleViewController () <RectangleViewDataSource>
12:
13: @property (nonatomic) float hauteurRectangle;
14:
15: @end
16:
17: @implementation RectangleViewController
18: @synthesize rectangleView = _rectangleView;
19: @synthesize hauteurRectangle = _hauteurRectangle;
20:
21: - (void)setRectangleView:(RectangleView *)rectangleView
22: {
23:     _rectangleView = rectangleView;
24:     self.rectangleView.dataSource = self;
25:     [self.rectangleView addGestureRecognizer:[[UIPanGestureRecognizer alloc]
   initWithTarget:self action:@selector(etirerRectangle:)]];
26: }
27:
28: - (void)setHauteurRectangle:(float)hauteurRectangle
29: {
30:     _hauteurRectangle = hauteurRectangle;
31:     [self.rectangleView setNeedsDisplay];
32: }
33:
34: - (void)viewDidLoad
35: {
36:     self.hauteurRectangle = -100;
37: }
38:
39: - (void)etirerRectangle:(UIPanGestureRecognizer *)gesture
40: {
41:     if ((gesture.state == UIGestureRecognizerStateChanged) || (gesture.state ==
   UIGestureRecognizerStateEnded)) {
42:         CGPoint translation = [gesture translationInView:self.rectangleView];
43:         self.hauteurRectangle += translation.y / 2.0;
44:         [gesture setTranslation:CGPointZero inView:self.rectangleView];
45:     }
46: }
47:
48: - (float)fixerHauteur:(id)sender
49: {
50:     return self.hauteurRectangle;
51: }
52:
53: - (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
54: {
55:     return (interfaceOrientation != UIInterfaceOrientationPortraitUpsideDown);
56: }
57:
58: @end
```

```objc
 1: //
 2: //  RectangleView.h
 3: //  Rectangle
 4: //
 5: //  Created by Gabriel Parriaux on 10.12.12.
 6: //  Copyright (c) 2012 gymo. All rights reserved.
 7: //
 8:
 9: #import <UIKit/UIKit.h>
10:
11: @protocol RectangleViewDataSource;
12:
13: @interface RectangleView : UIView
14:
15: @property (nonatomic, weak) id <RectangleViewDataSource> dataSource;
16:
17: @end
18:
19: @protocol RectangleViewDataSource <NSObject>
20:
21: - (float)fixerHauteur:(id)sender;
22:
23: @end
```

```objc
1: //
2: //  RectangleView.m
3: //  Rectangle
4: //
5: //  Created by Gabriel Parriaux on 10.12.12.
6: //  Copyright (c) 2012 gymo. All rights reserved.
7: //
8:
9: #import "RectangleView.h"
10:
11: @implementation RectangleView
12:
13: - (id)initWithFrame:(CGRect)frame
14: {
15:     self = [super initWithFrame:frame];
16:     if (self) {
17:         // Initialization code
18:     }
19:     return self;
20: }
21:
22: - (void)drawRect:(CGRect)rect
23: {
24:     float hauteurRect = [self.dataSource fixerHauteur:self];
25:     CGContextRef context = UIGraphicsGetCurrentContext();
26:     CGContextSetRGBFillColor(context, 1.0, 0.0, 1.0, 1.0);
27:     CGContextBeginPath(context);
28:     CGRect rectangle = CGRectMake((self.bounds.origin.x + self.bounds.size.width /4),
   (self.bounds.origin.y + self.bounds.size.height * 3 / 4), (self.bounds.origin.x +
   self.bounds.size.width / 2), hauteurRect);
29:     CGContextAddRect(context, rectangle);
30:     CGContextFillPath(context);
31: }
32:
33:
34: @end
```