

Black Girls Code “Bouncing Balls” Mobile Coding Challenge

Yammer – 7/28/15

Step 0:

Locate and open the project titled “Bouncing Balls Practice”

Challenge 1:

Create a new ball by tapping on the screen.

1. Go to the file: ViewController.m
2. Add a tap gesture recognizer (UITapGestureRecognizer) where the view is loaded (viewDidLoad):

```
UITapGestureRecognizer *tapGesture = [[UITapGestureRecognizer alloc]
initWithTarget:self action:@selector(createBall:));
[self.view addGestureRecognizer:tapGesture];
```

3. Create a ball and add it to the view where the tap was made (createBall). This method is called when a tap occurs.

```
CGPoint tapLocation = [gestureRecognizer locationInView:self.view];
UIView *ball = [[Ball alloc] initWithFrame:CGRectMake(tapLocation.x,
tapLocation.y, 50.0, 50.0)];
[self.view addSubview:ball];
```

4. Hit play and tap on the screen to see the balls appear!

Challenge 2:

Move the ball when panning the image.

1. Go to the file: Ball.m
2. Add a pan gesture recognizer (UIPanGestureRecognizer) where the view is initialized (initWithFrame):

```
UIPanGestureRecognizer *panRecognizer = [[UIPanGestureRecognizer alloc]
initWithTarget:self action:@selector(handlePan:)];
self.gestureRecognizers = @[panRecognizer];
```

3. When a pan occurs, move the ball by changing the ball’s center point to where the pan gesture ends (handlePan):

```
CGPoint translation = [panGestureRecognizer
translationInView:self.superview];
self.center = CGPointMake(self.lastLocation.x + translation.x,
```

```
self.lastLocation.y + translation.y);
```

4. Hit play and tap on the screen and move the ball by dragging it across the screen!

Challenge 3:

Add gravity so that the ball will actually fall and bounce on the ground like it would in real life!

1. Go to the file: Ball.m

2. Add gravity to the ball after it is first created (layoutSubviews):

```
[self addGravity];
```

3. Add gravity to the ball after the user stops panning the image (handlePan):

```
[self addGravity];
```

4. Hit play and tap on the screen to see the balls appear and drop to the ground. Also trying moving the ball up and see it drop to the ground again.

Challenge 4:

Change the image of the ball when it bounces by replacing the image name.

1. Go to the file: Ball.m

2. In the project toolbar, go to the Image.xcassets folder. Press the + symbol bollow to add a new image set. Pull the image into the 1x box.

3. In the delegate method, called “collisionBehavior”, replace the name of the image with the desired image:

```
self.ball.image = [UIImage imageNamed:@"newImageName"];
```

Challenge 5: Add sound when the ball bounces to the ground.

1. Download an audio file and copy it to the bundle by drag to the XCode project sidebar.

2. In the delegate method, called “collisionBehavior”, replace the name of the image with the desired image:

```
NSString *soundPath = [[NSBundle mainBundle]
                        pathForResource:@"newSound" ofType:@"mp3"];
NSURL *soundPathURL = [NSURL fileURLWithPath:soundPath];
AudioServicesCreateSystemSoundID((__bridge CFURLRef)soundPathURL,
&_sound);
AudioServicesPlaySystemSound(_sound);
```

What Next?

You can view code the here:

<https://github.com/aluong-yammer/BouncingBalls>

Think about what you could add to the app? Moving clouds? Birds flying across the sky? Take action!

Tools you'll need:

1. A Mac.

2. An Apple Developer's Account:

<https://developer.apple.com/register/>

3. Xcode:

<https://developer.apple.com/xcode/downloads/>

4. Check out some iOS tutorials:

<http://www.appcoda.com/ios-programming-course/>

<https://www.udemy.com/the-art-of-real-ios-programming/>

5. Questions?

Alda Luong: aluong@yammer-inc.com

Ann Robson: arobson@yammer-inc.com