

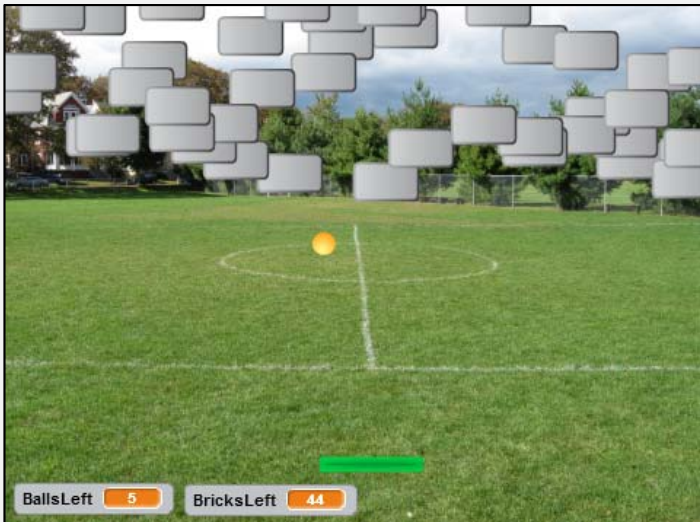
# A Breakout Game Created in Scratch

<http://cs.oregonstate.edu/~mjb/scratch>

**Mike Bailey**

[mjb@cs.oregonstate.edu](mailto:mjb@cs.oregonstate.edu)

**Oregon State University**



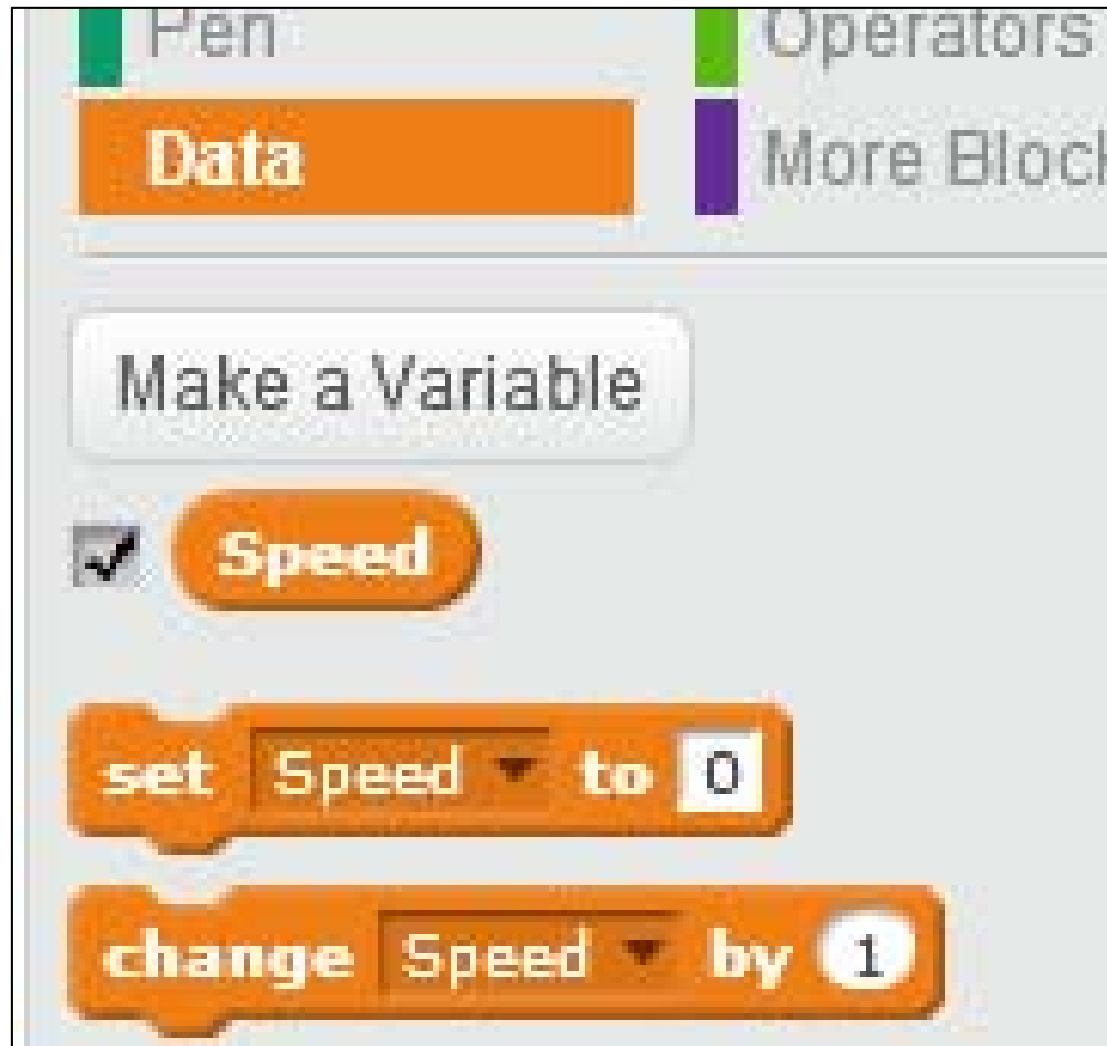
Oregon State University  
Computer Graphics

## Events

2<sub>2</sub>



## Data





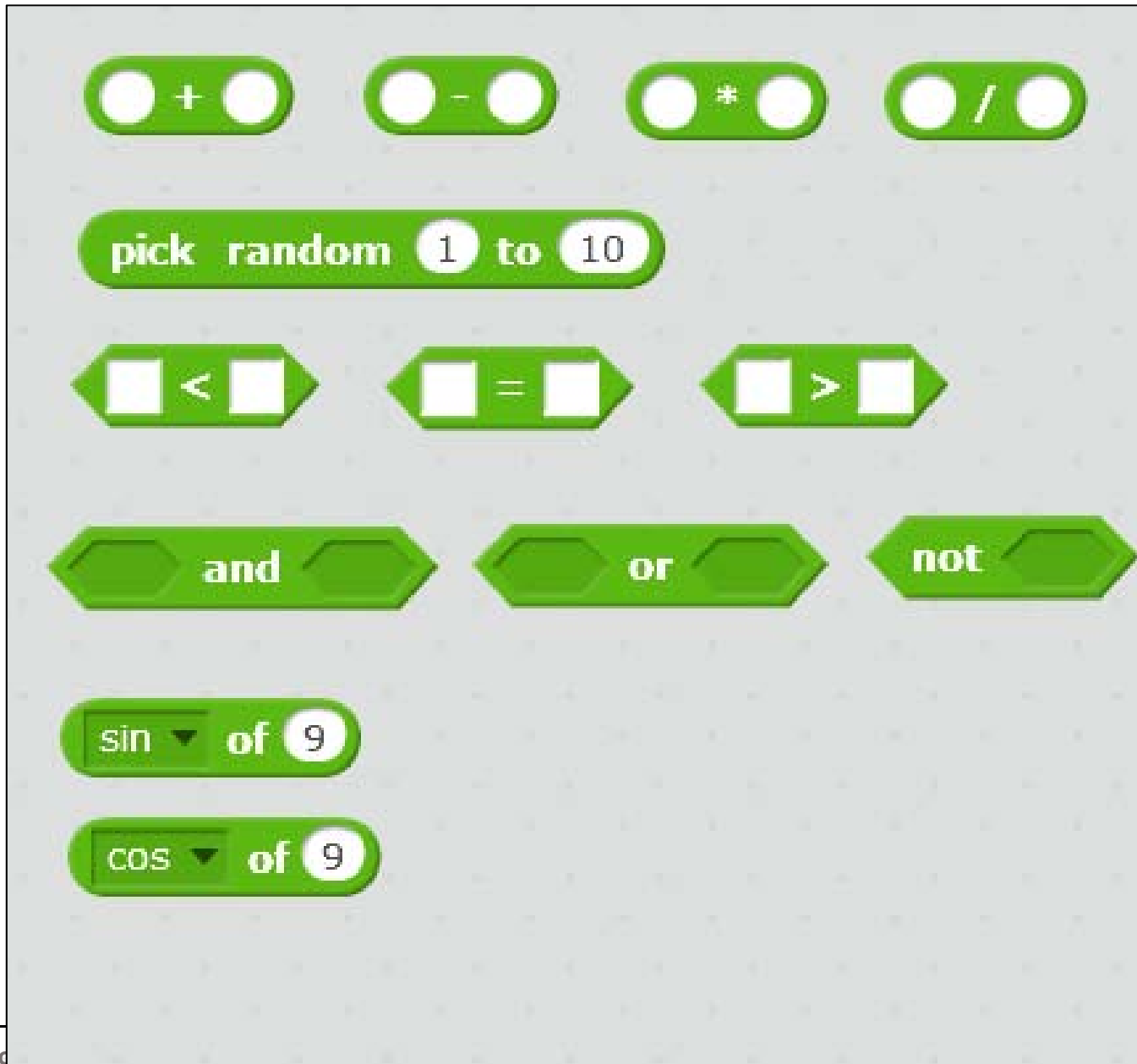
## Sensing

5



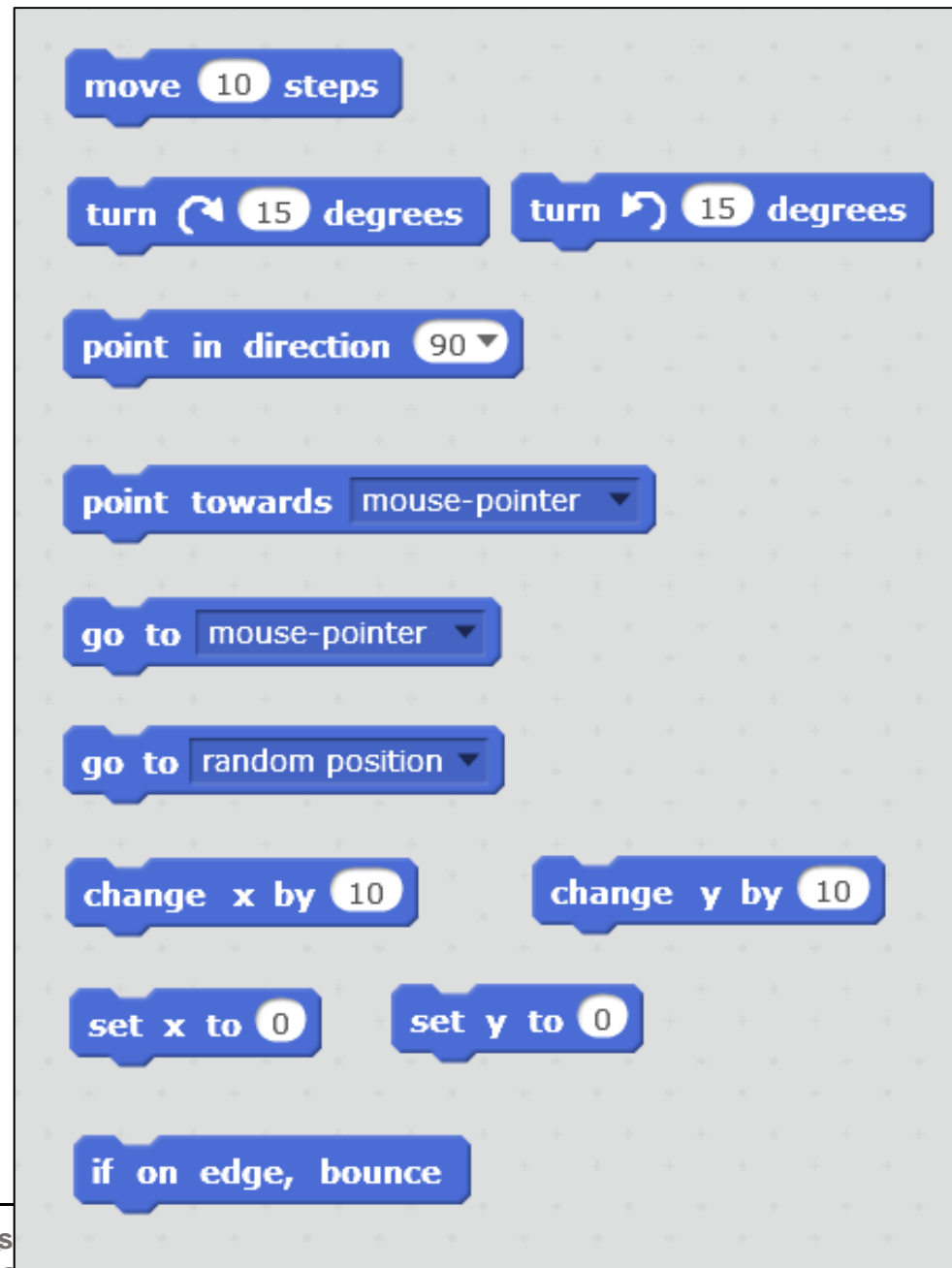
## Operators

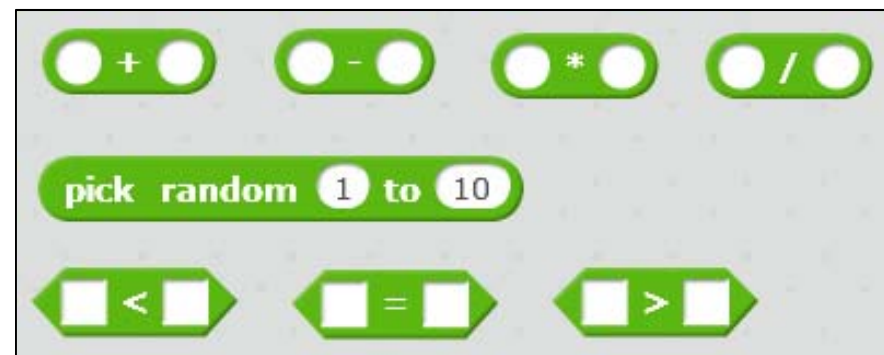
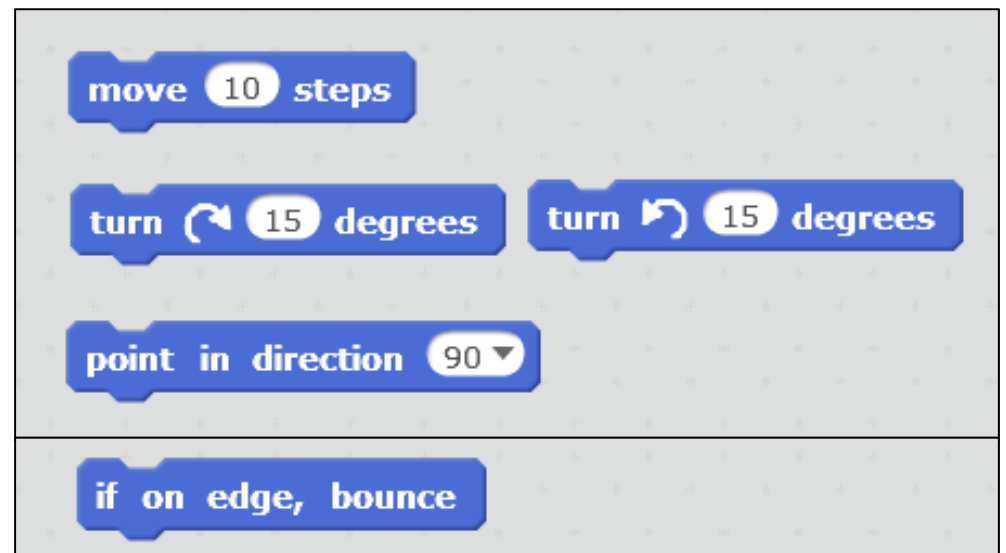
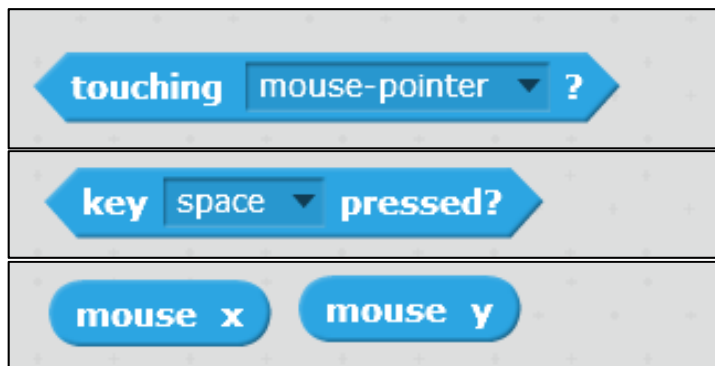
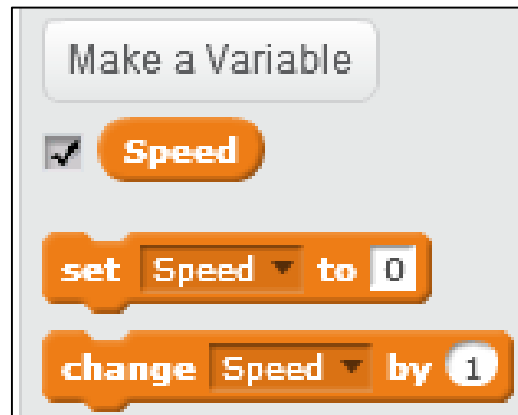
6



## Motion

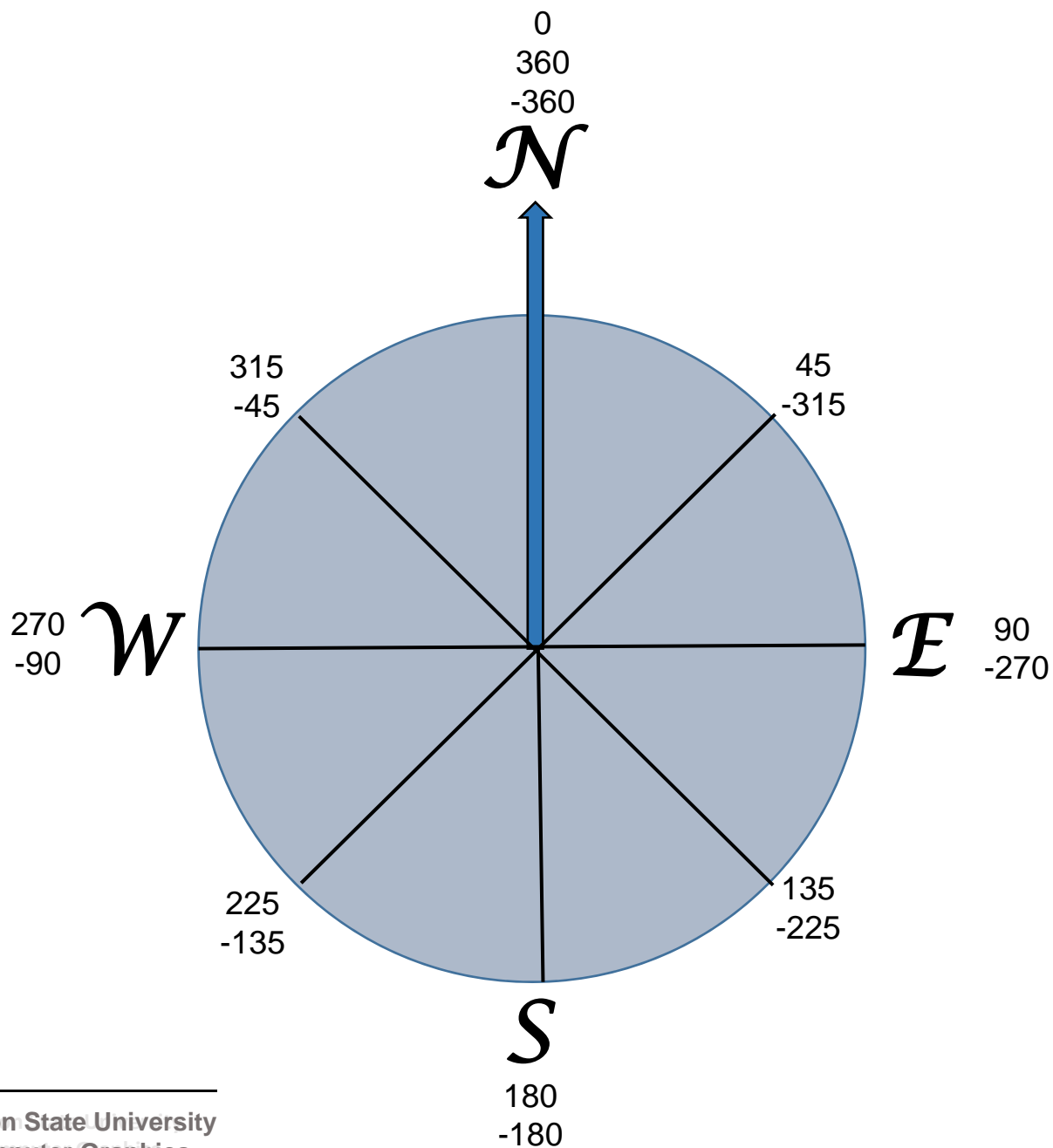
7







## Scratch Handles the Directions you are Going Like a Compass Does



## Three Actors

Displaying Data Variables

Three Actors

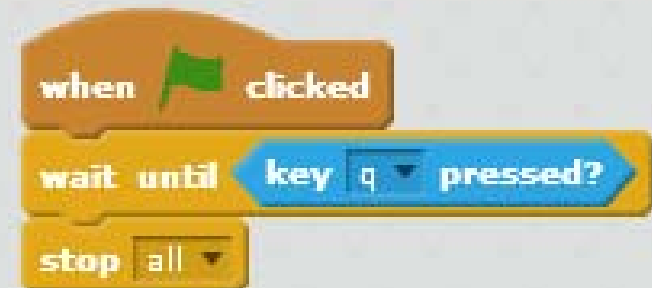


## Paddle

When the program starts, set some good values into the variables, get everyone else going, then have the Paddle continuously follow the mouse pointer



The 'q' key can end the game



## Bricks

Create the bricks to start with

Test to see if you've won

x: -57  
y: 31

when I receive CreateBricks

show

repeat BricksLeft

create clone of Button3

hide

when green flag clicked

wait until BricksLeft = 0

broadcast YouWin

Position each brick in a random location.  
Then, wait for the Ball to hit the brick. When it does, decrease the brick count, bounce the Ball (which will result in a pop sound), and delete this brick.

when I start as a clone

go to x: pick random -240 to 240 y: pick random 50 to 180

forever

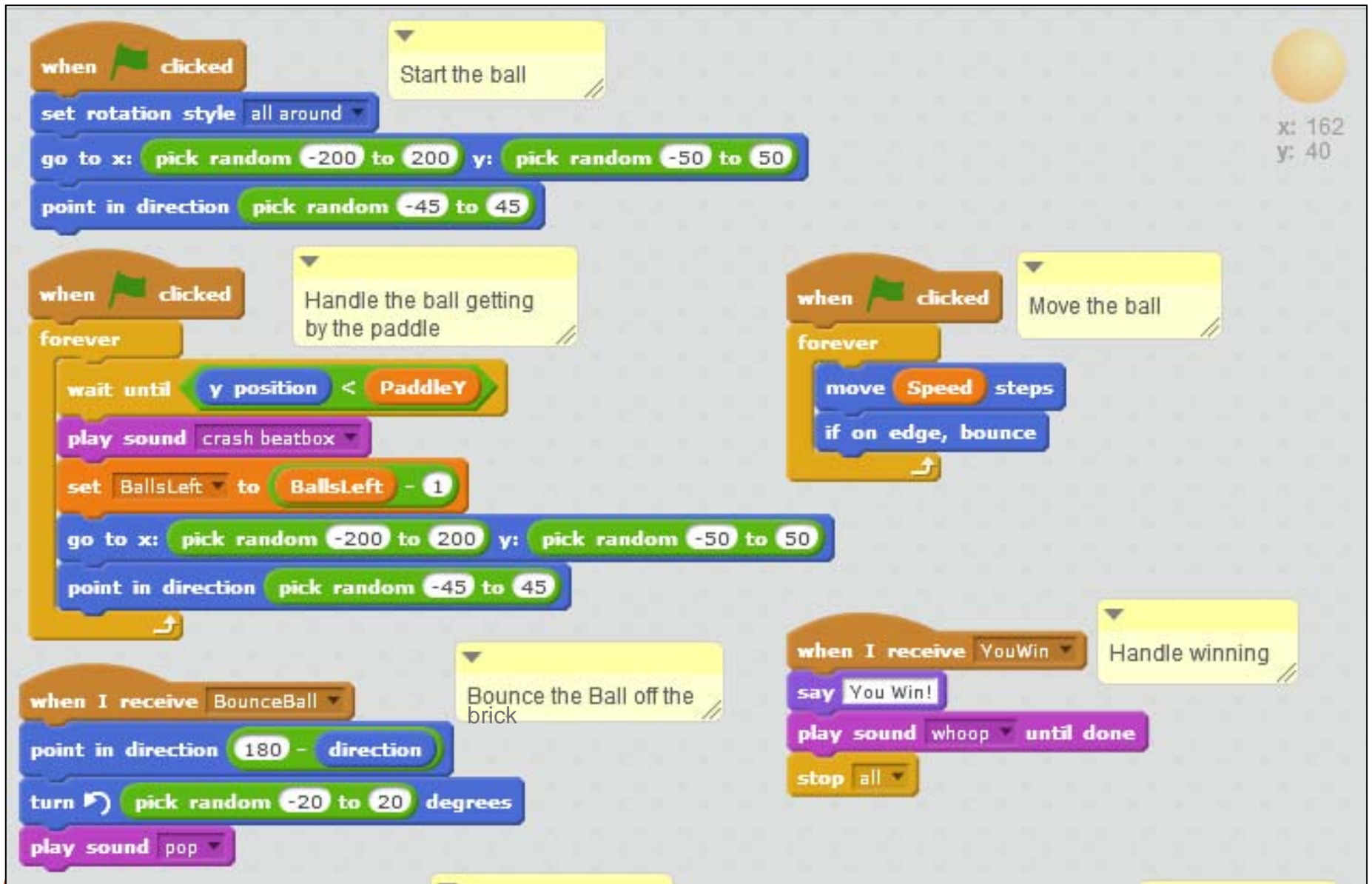
wait until touching Ball ?

set BricksLeft to BricksLeft - 1

broadcast BounceBall

delete this clone

## Ball -- top



The image displays a Scratch script for a ball game, organized into four distinct event-driven blocks. Each block is preceded by a yellow comment box. The first block, 'when clicked', initializes the ball's position and direction. The second block, 'when clicked', handles the ball hitting the paddle, triggering a sound, decrementing a counter, and respawning the ball. The third block, 'when I receive BounceBall', handles the ball hitting a brick, reversing its direction and adding a random turn. The fourth block, 'when I receive YouWin', handles the game's conclusion with a win message and sound. A ball sprite is shown in the top right corner with its coordinates.

```
when clicked
  Start the ball
  set rotation style to all around
  go to x: pick random -200 to 200 y: pick random -50 to 50
  point in direction pick random -45 to 45

when clicked
  Handle the ball getting by the paddle
  forever loop
    wait until y position < PaddleY
    play sound crash beatbox
    set BallsLeft to BallsLeft - 1
    go to x: pick random -200 to 200 y: pick random -50 to 50
    point in direction pick random -45 to 45

when I receive BounceBall
  Bounce the Ball off the brick
  point in direction 180 - direction
  turn pick random -20 to 20 degrees
  play sound pop

when I receive YouWin
  Handle winning
  say You Win!
  play sound whoop until done
  stop all
```

Ball sprite coordinates: x: 162, y: 40



## Ball -- bottom

