

# **Using Scratch to Teach Event-Driven Coding Skills**

## **Step-by-Step**

<https://tapggc.org/>



Pages 3-6 ..... General Set-up

Pages 7-25 ..... Level 1

Pages 26-31 ..... Level 2

Pages 32-38 ..... Level 3



Start by going to <https://github.com/TAP-GGC/makeysrace>. Then click on the code folder and select the incomplete code. This will contain all 3 levels of the game.

The screenshot shows the GitHub repository page for 'makeysrace' by 'TAP-GGC'. The repository is public and has 2 branches and 0 tags. The 'code' folder is highlighted with a red box. The repository is 43 commits ahead of the master branch. The 'code' folder was created 2 days ago. The repository also contains a 'Photos' folder, an 'archive' folder, and a 'README.md' file. The README file is titled 'Using Scratch to Teach Event-Driven Coding Skills' and has an 'About' section.

<https://tapggc.org/>

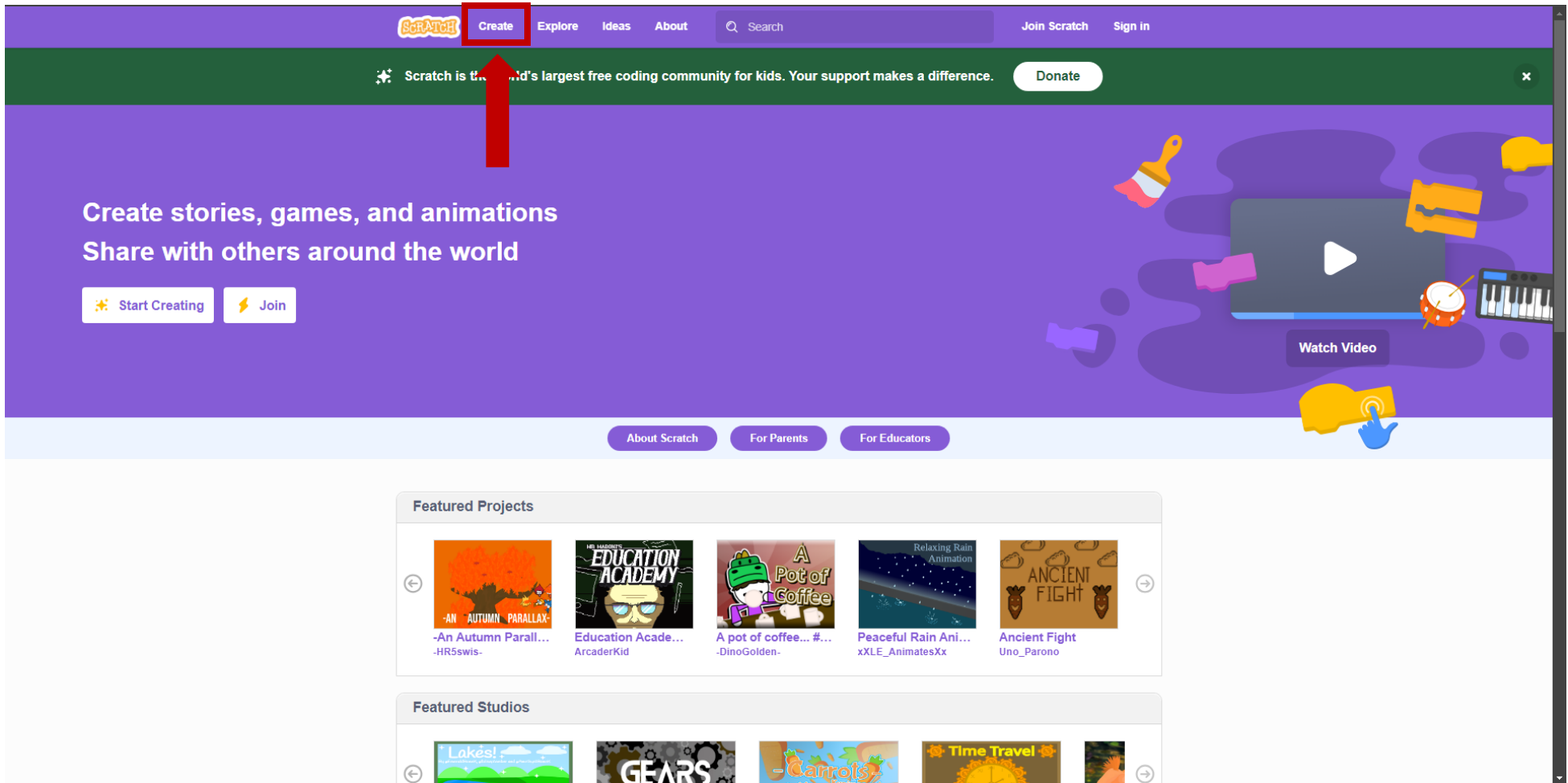


The screenshot shows the GitHub interface for the repository 'makeysrace' by user 'TAP-GGC'. The repository is on the 'code' tab, and the current branch is 'master'. A notification from user 'tlam8' indicates a deletion of 'code/Level1And2IncompleteCode.sb3'. Below this, a status bar shows 'This branch is 62 commits ahead of, 1 commit behind master'. The file list table contains the following entries:

Name	Last commit message
..	
CompletedCode.sb3	Add files via upload
IncompleteCode.sb3	Add files via upload

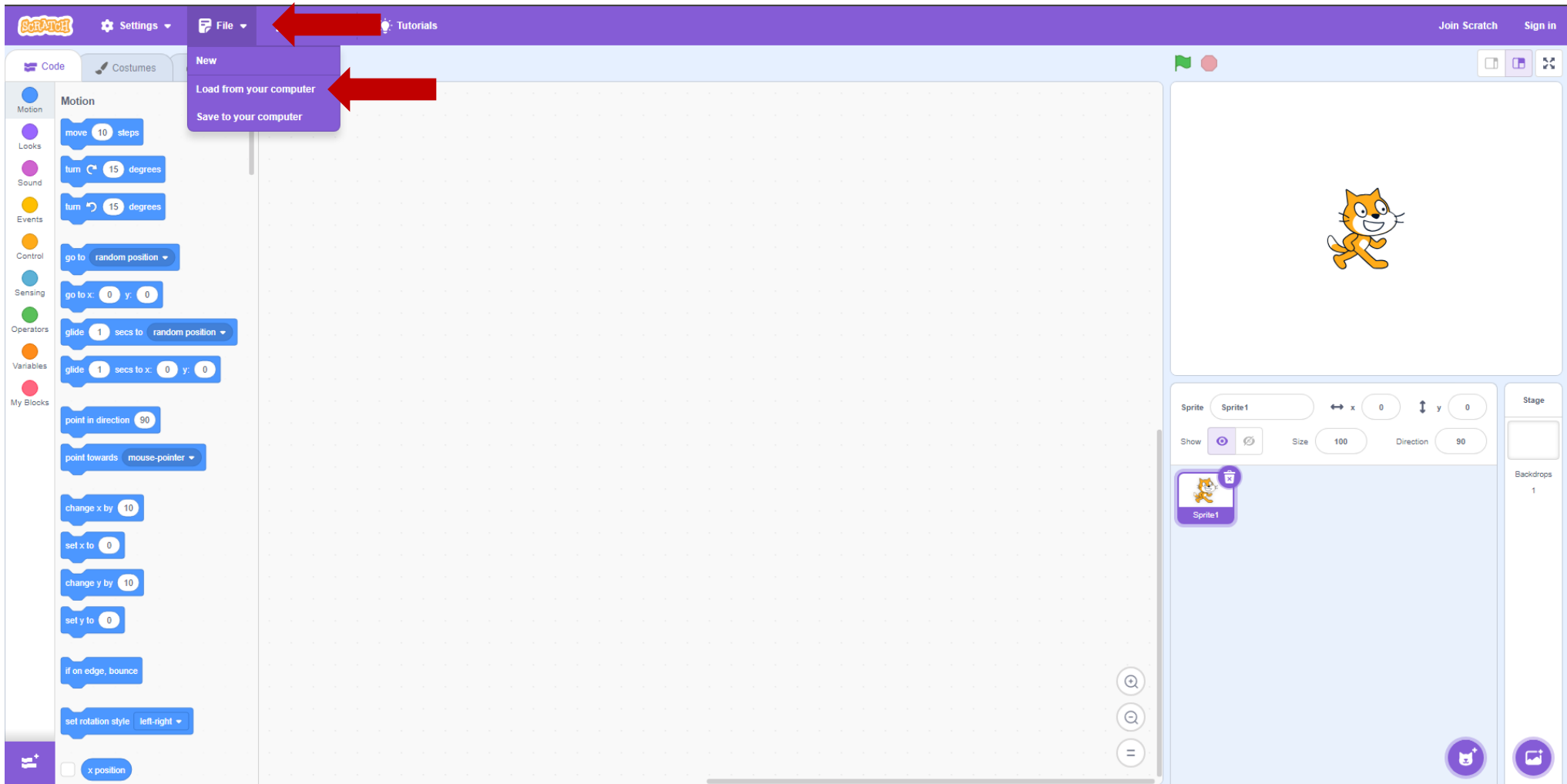
The row for 'IncompleteCode.sb3' is highlighted with a red border.

After you download the file, go to <https://scratch.mit.edu/> and then click on create.



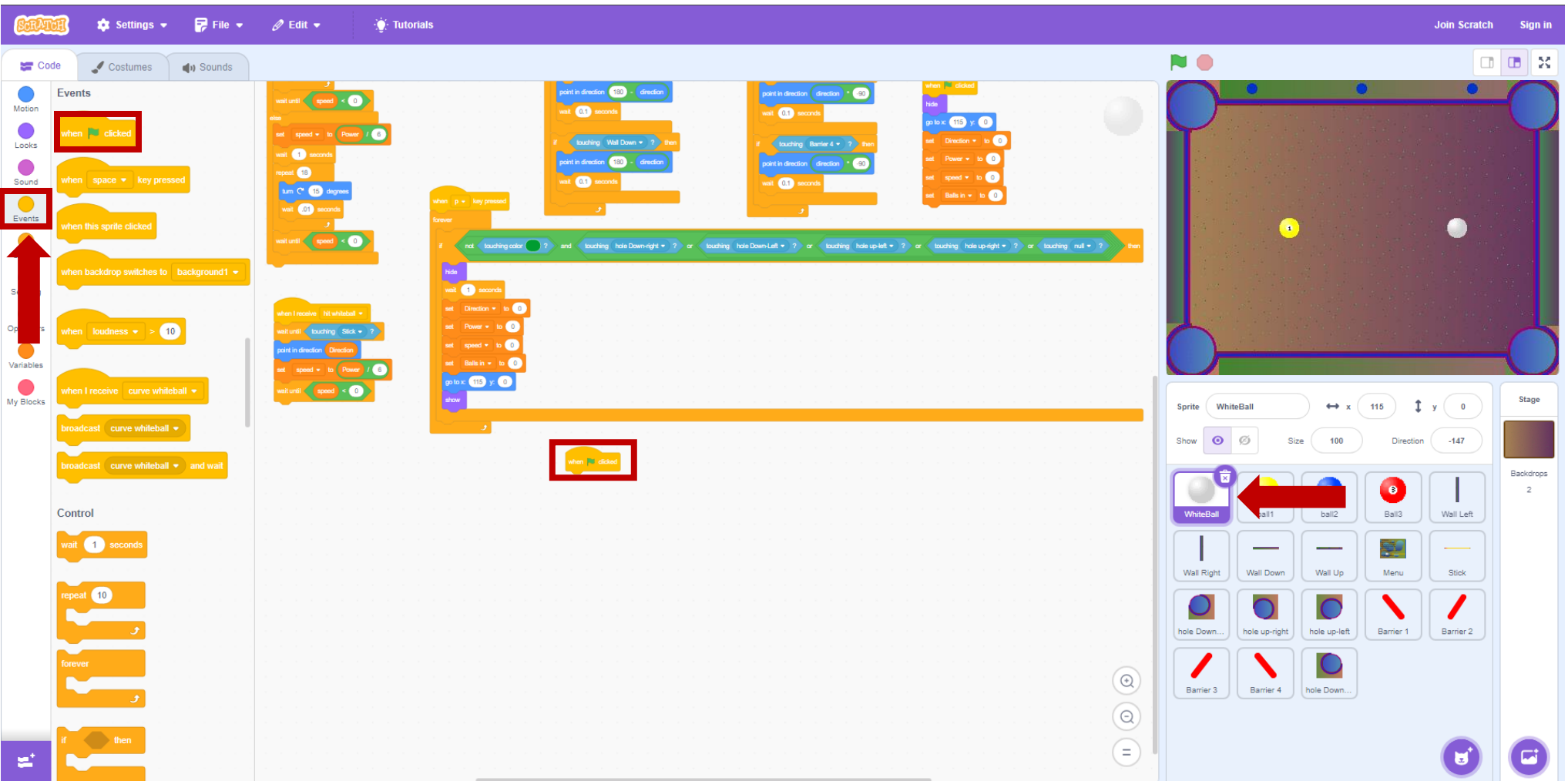
<https://tapggc.org/>

Now click on “file” and then “Load from your computer.” Then select the file that was just downloaded from GitHub. It should be called “IncompleteCode.sb3”.

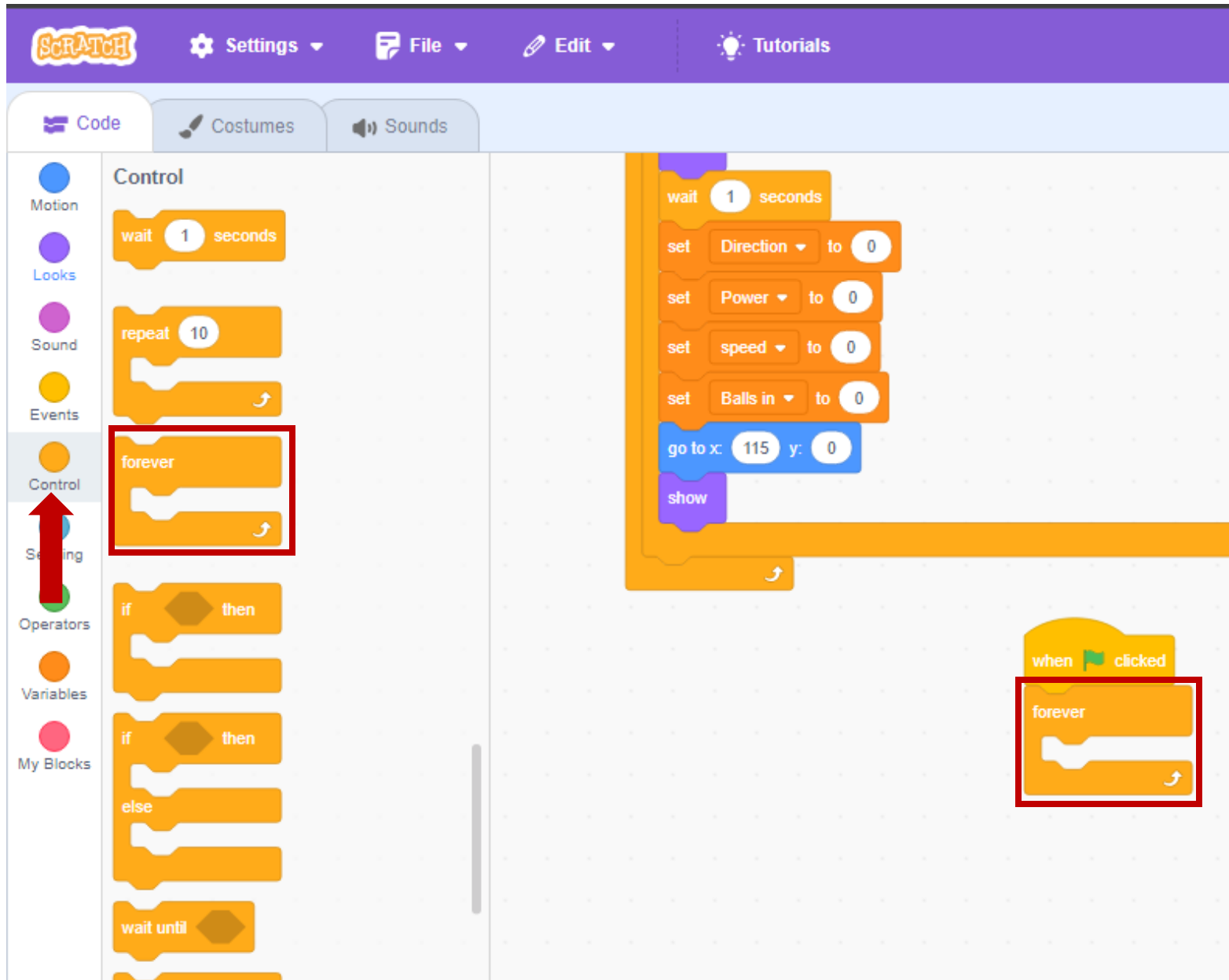


**This section will cover level 1 of the game.**

Make sure that the white ball is the selected sprite. Now click on “Events” on the left side of the screen. Then click and drag the “when flag clicked” and drop on an empty space on the canvas.

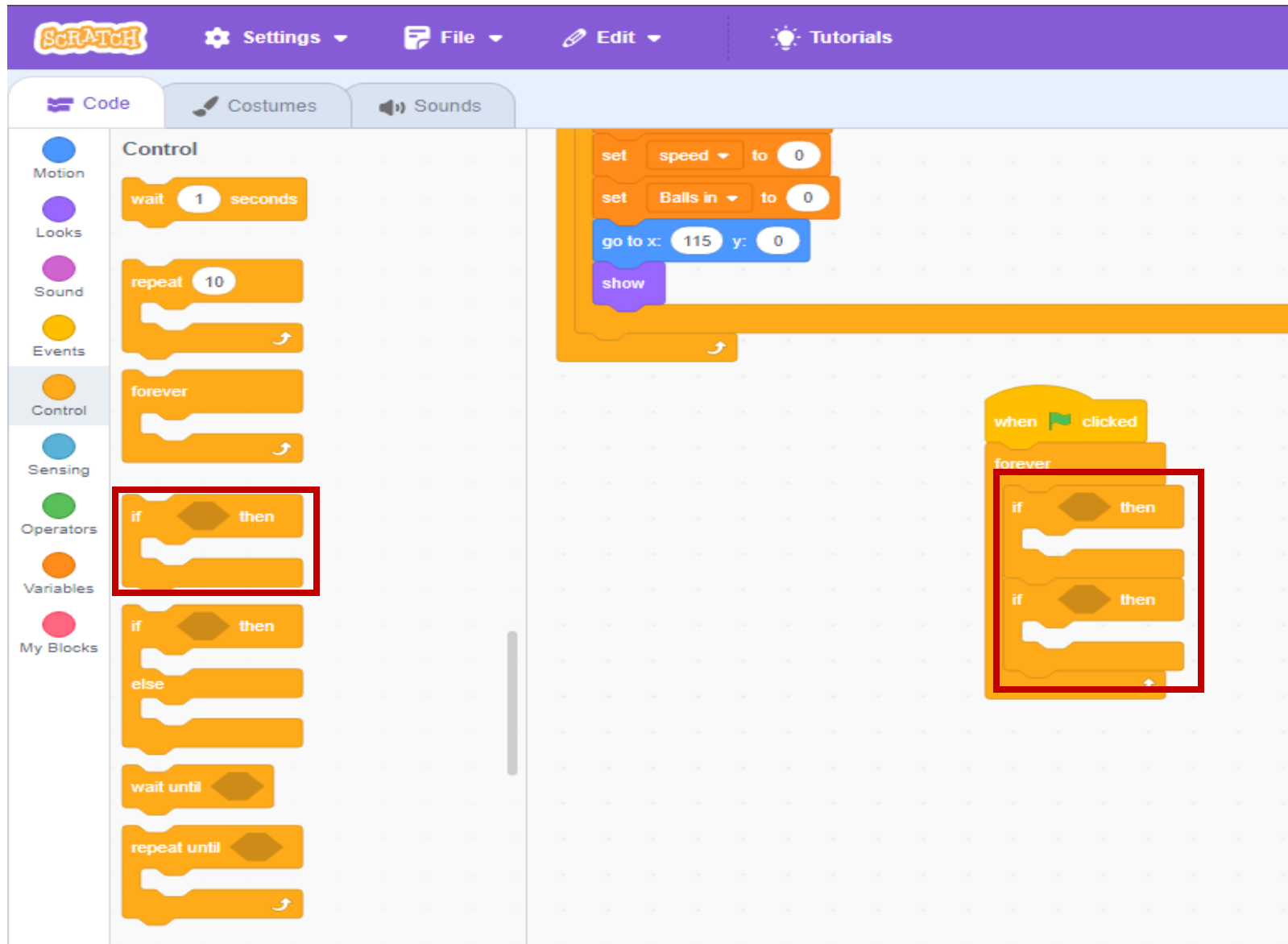


Now go to “Control” and grab the “forever” loop and attach it to code block.

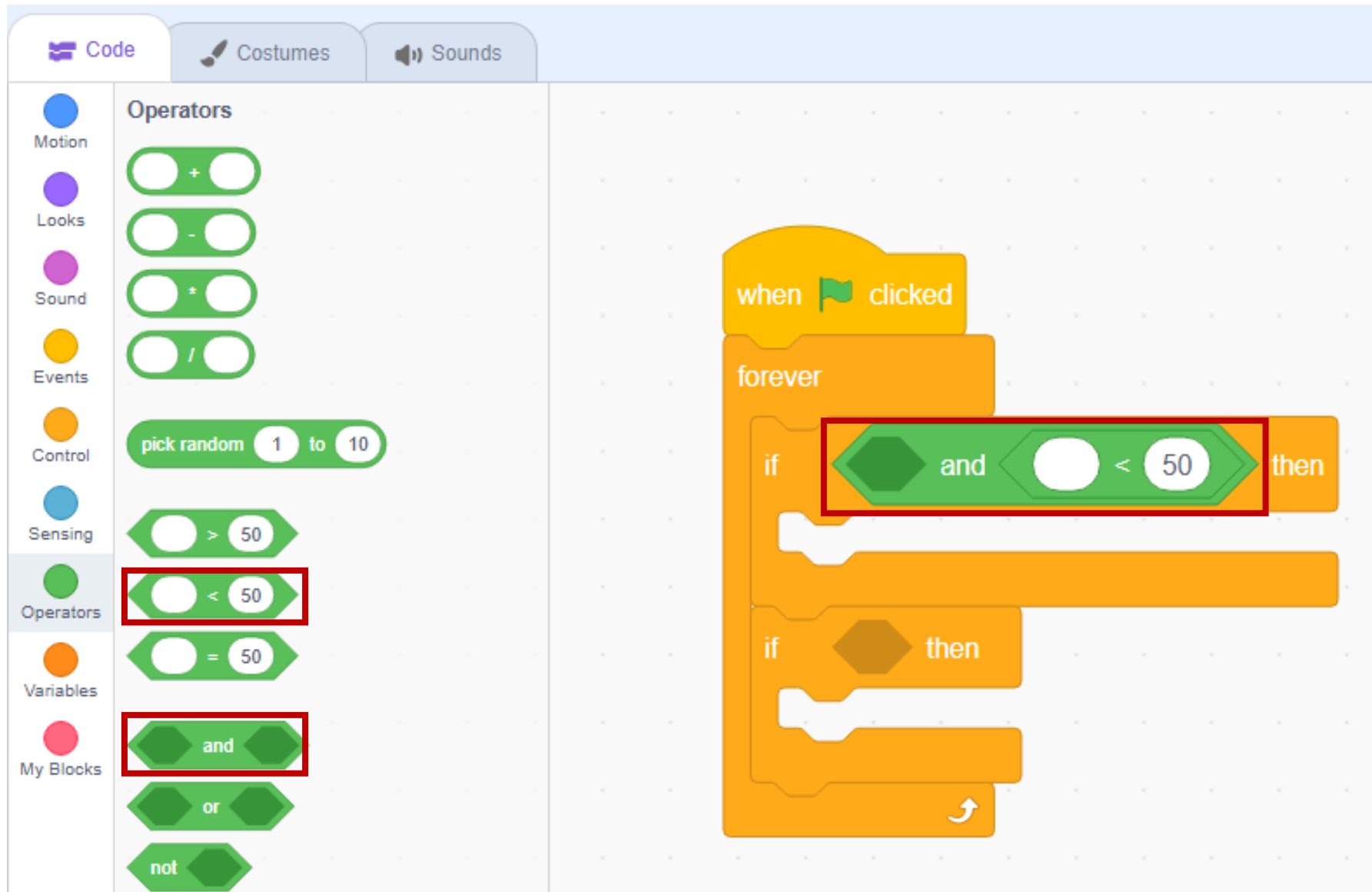




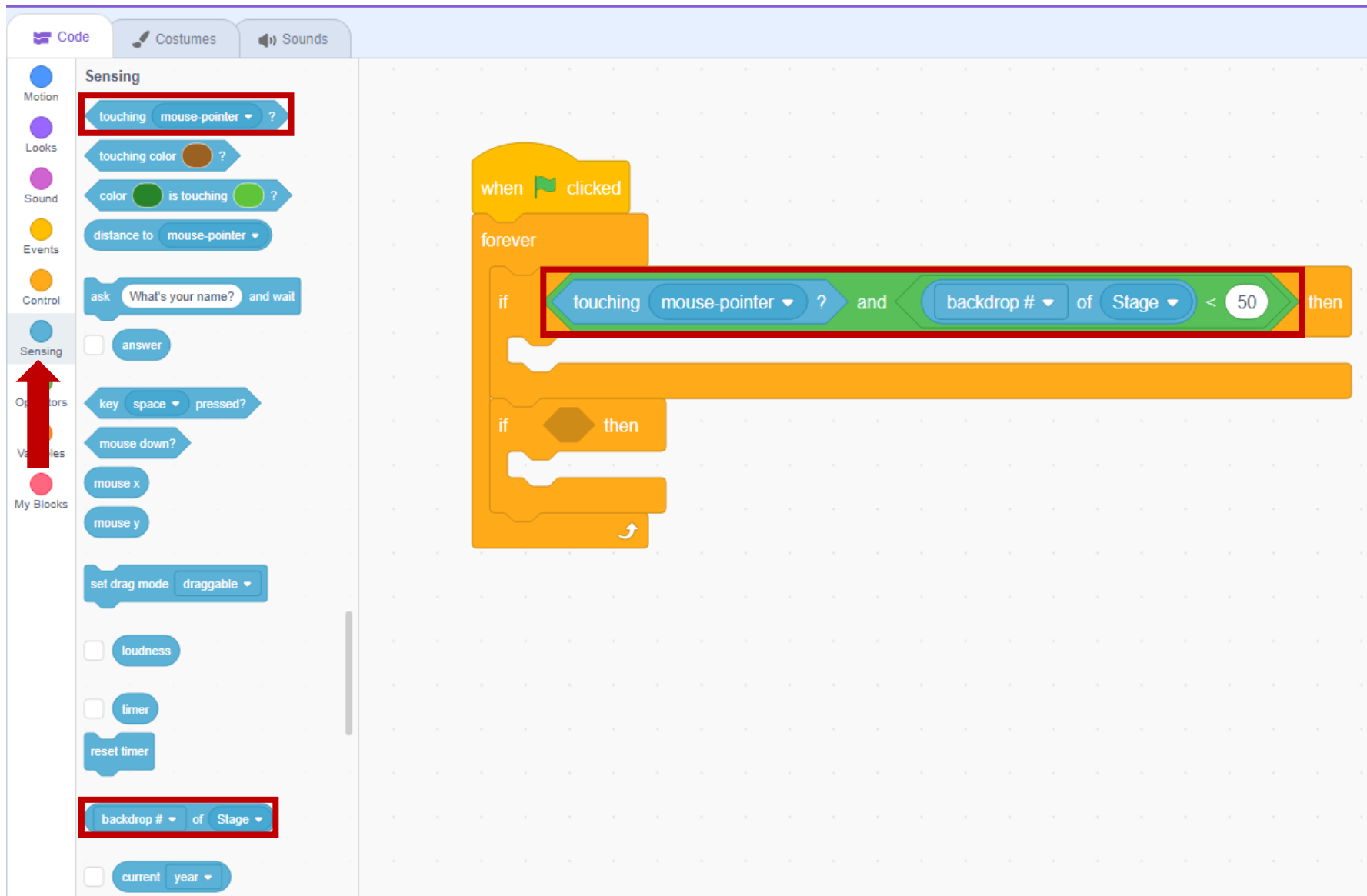
Now grab 2 “if-then” blocks and put them into the “forever loop” like shown below.

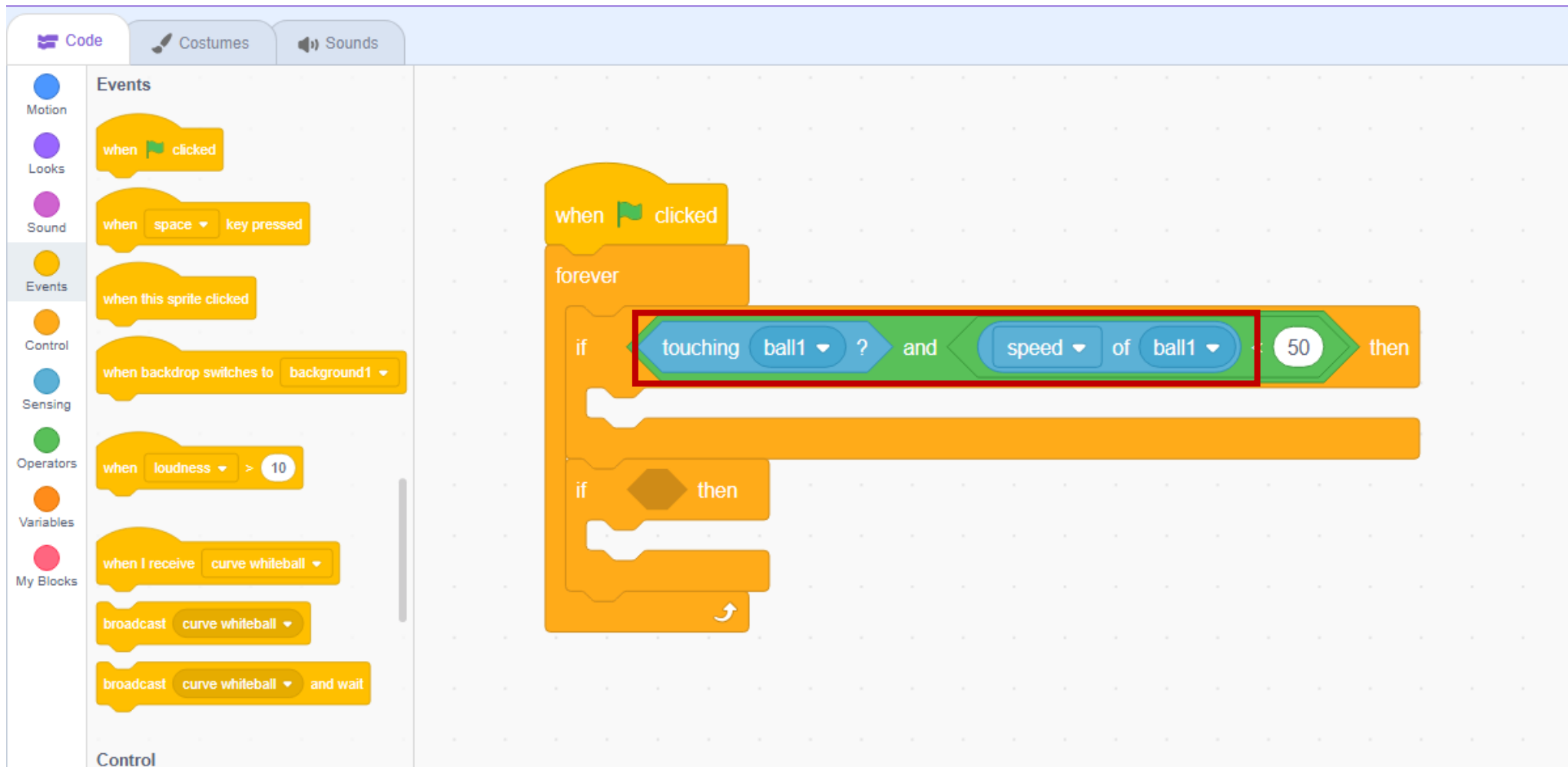


We will focus on the top “if-then” block for now. Go to “Operators”. Then grab the “\_\_ and \_\_” block and put it into the blank in the “if-then” block. Then grab the “\_\_ < 50” block and put it into the second blank of the “\_\_ and \_\_” block.

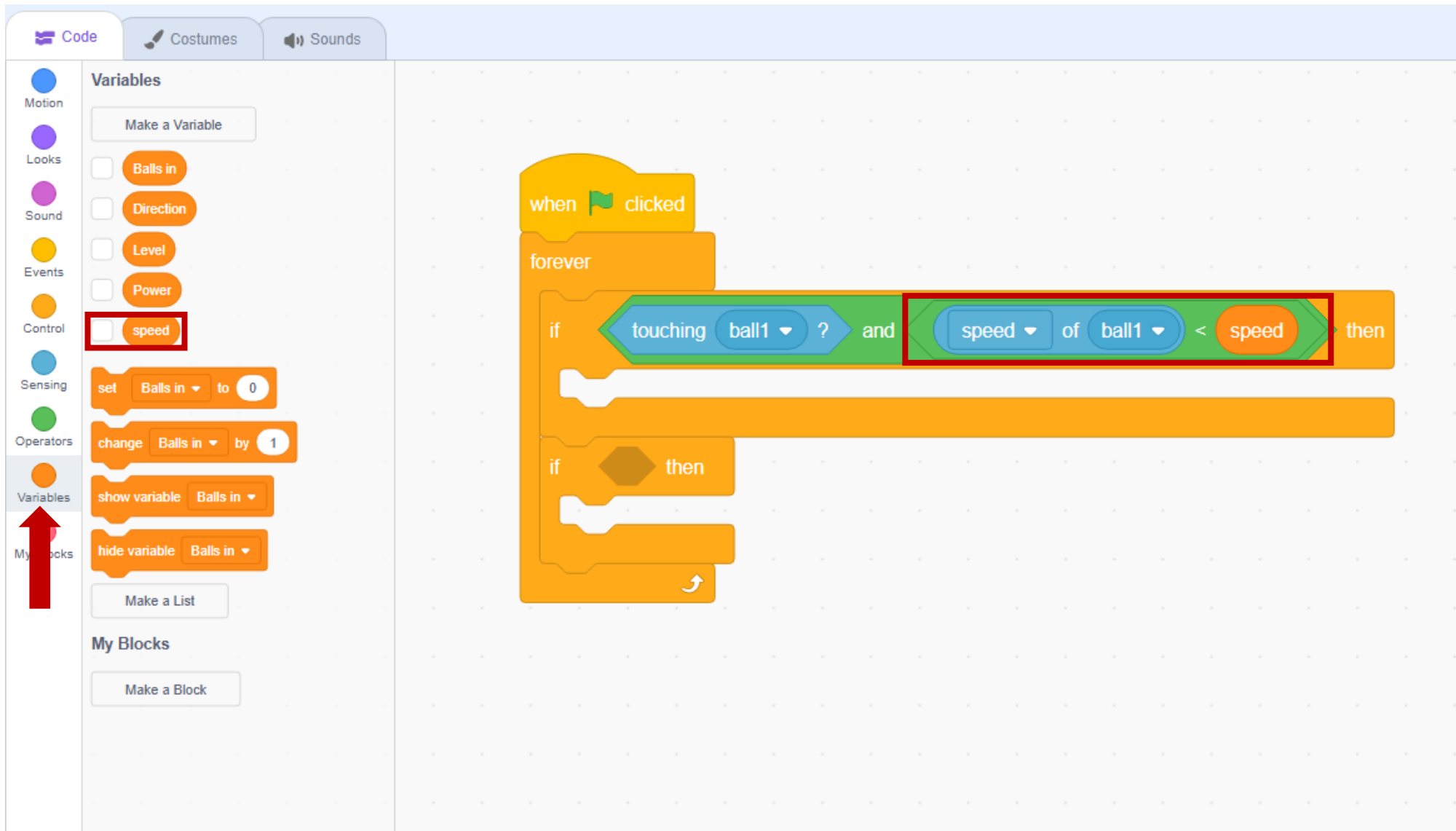


Go to “Sensing”. Grab the “touching mouse pointer?” block and insert it into the first blank of the “if-then” block. Then grab a “backdrop # of Stage” block and insert it into the “\_\_ < 50” block. Change the mouse pointer to “ball 1” and “backdrop # of Stage” to “speed of ball 1”.



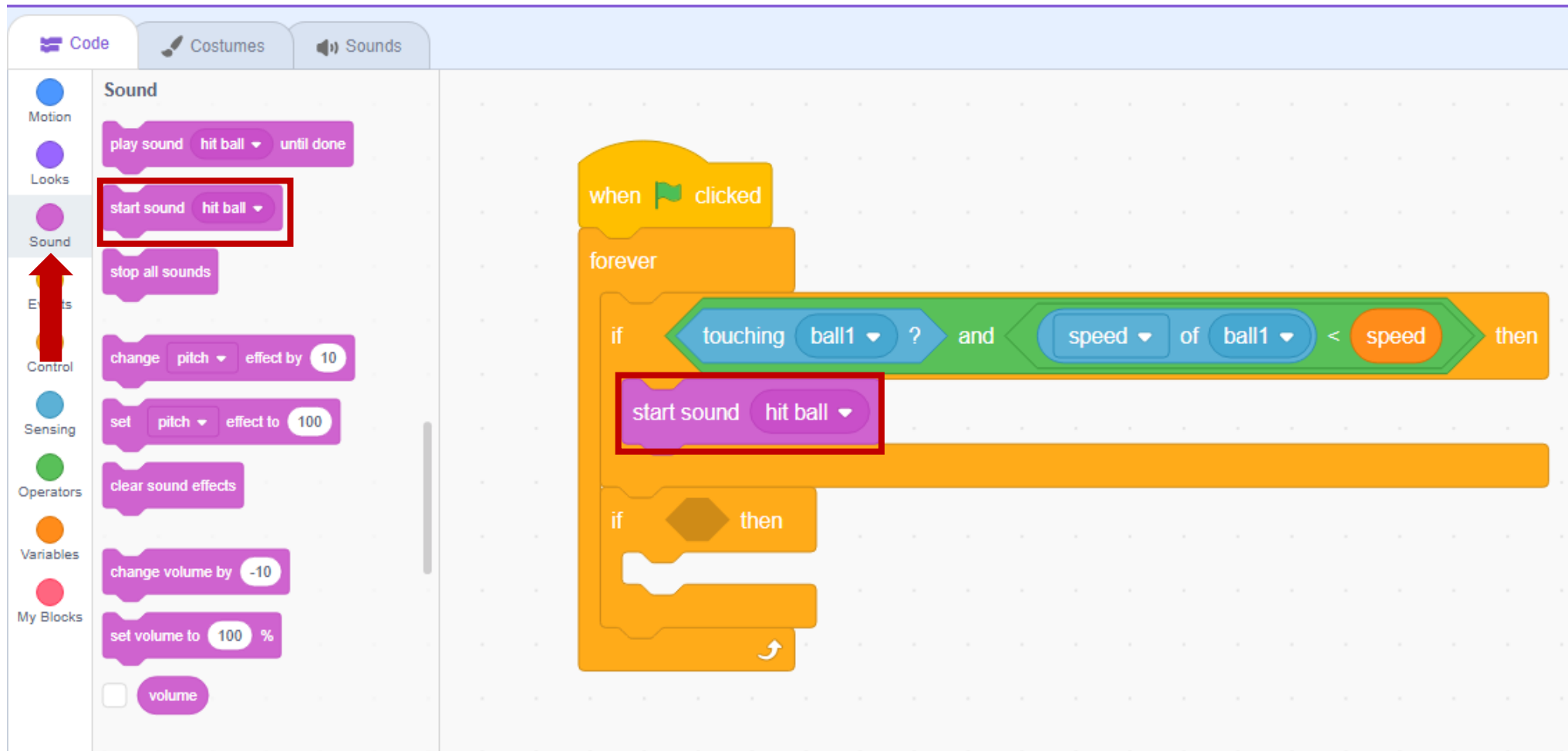


Now go to the “Variables” tab and grab the “speed” variable. Insert it into the “\_\_ < 50” block.

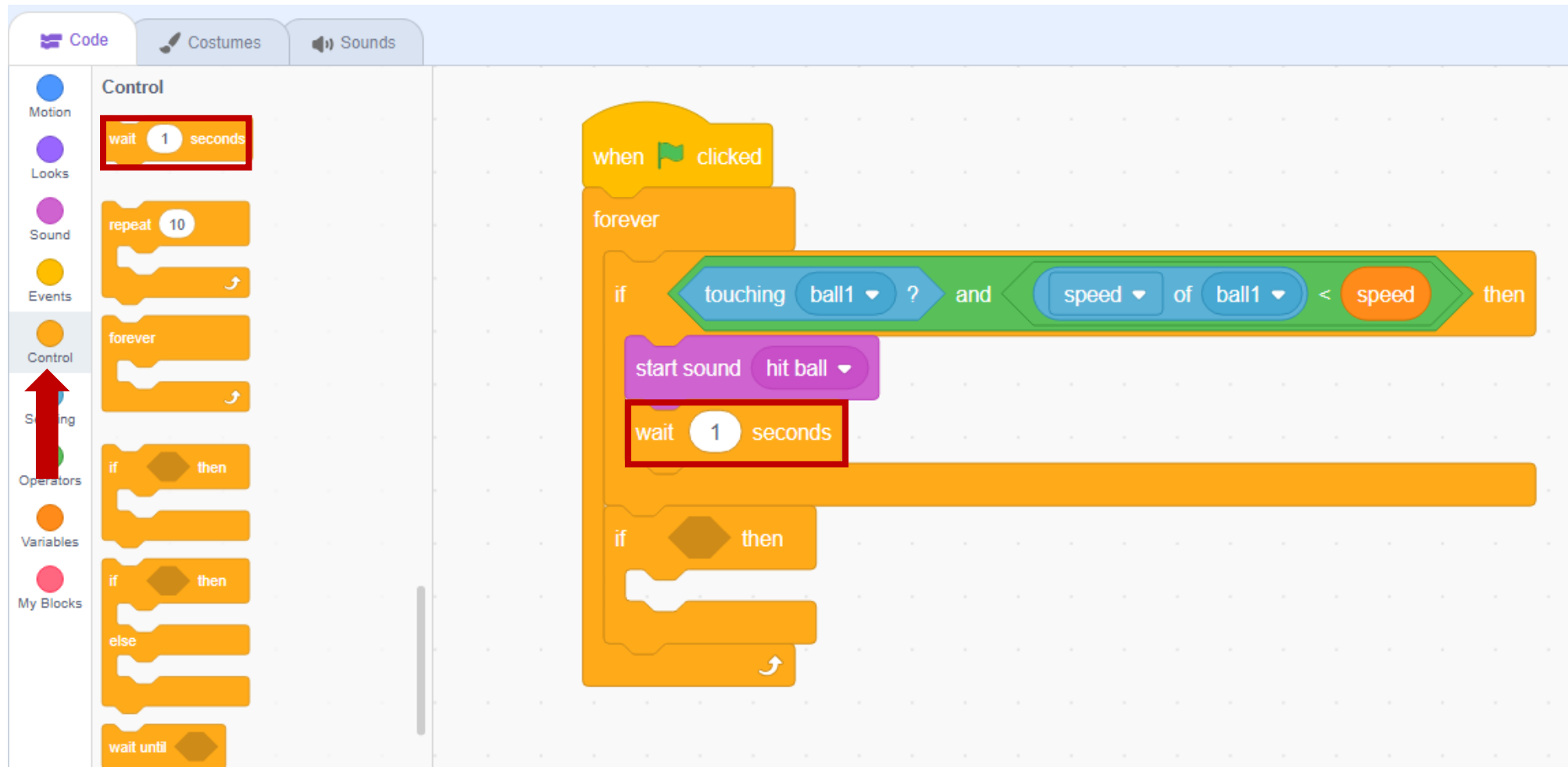


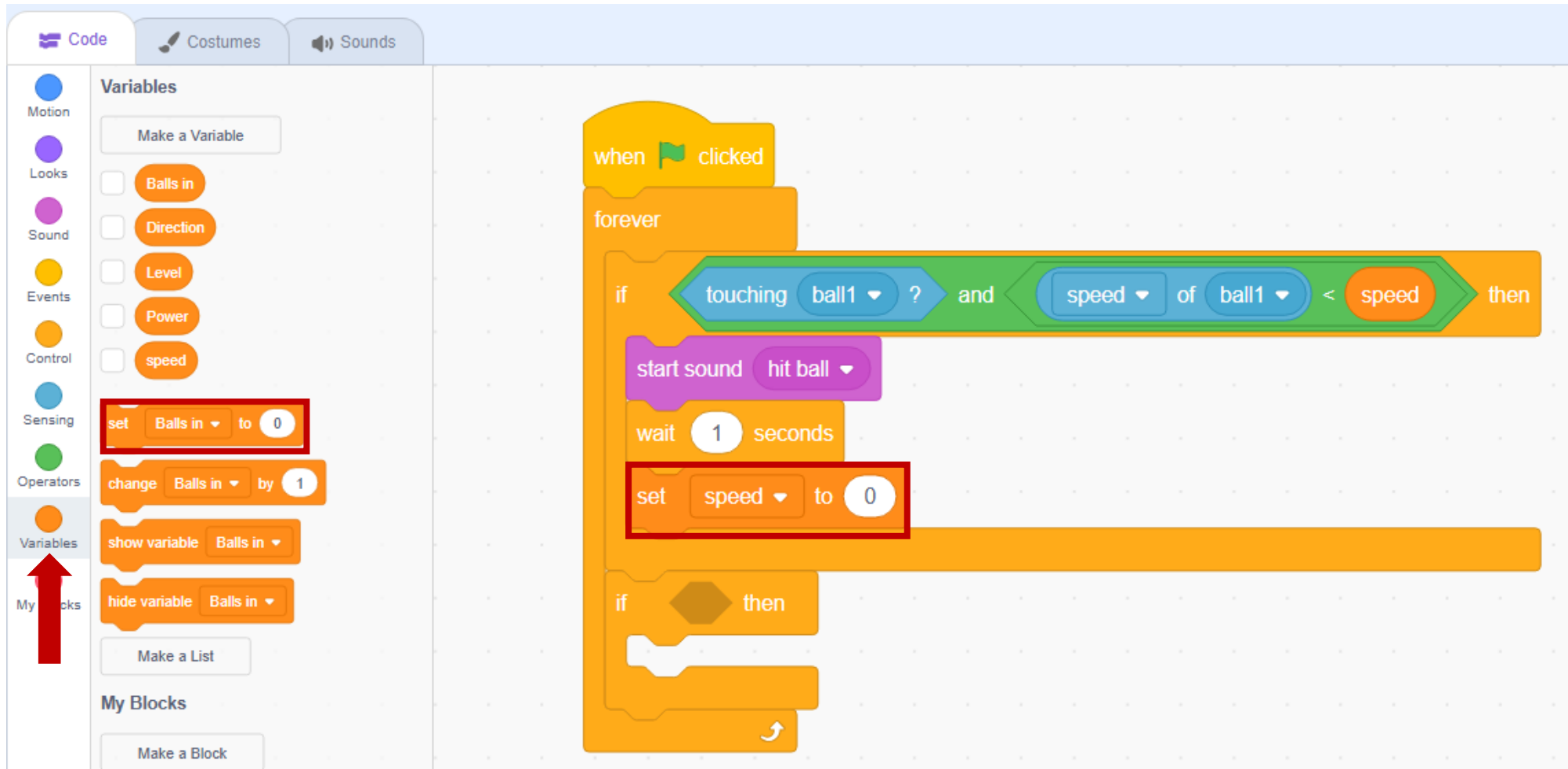
The image shows the Scratch code editor interface. On the left sidebar, the 'Variables' tab is selected, and the 'speed' variable is highlighted with a red box. A red arrow points to the 'Variables' tab. The main workspace shows a script starting with a 'when clicked' event block, followed by a 'forever' loop. Inside the loop, there is an 'if' statement with two conditions: 'touching ball1' and 'speed of ball1 < speed'. The 'speed' variable is highlighted in the 'if' statement. Below the 'if' statement, there is an empty 'if' statement block.

Go to “sounds” and grab the “start sound hit ball” and put it into the “if-then” statement.



Go to “Control” and then grab the “wait 1 seconds” block and insert it into the “if-then” statement. Then go to “Variables” and grab the “set balls in to 0” block and put it into the “if-then” statement. Change the variable to speed.

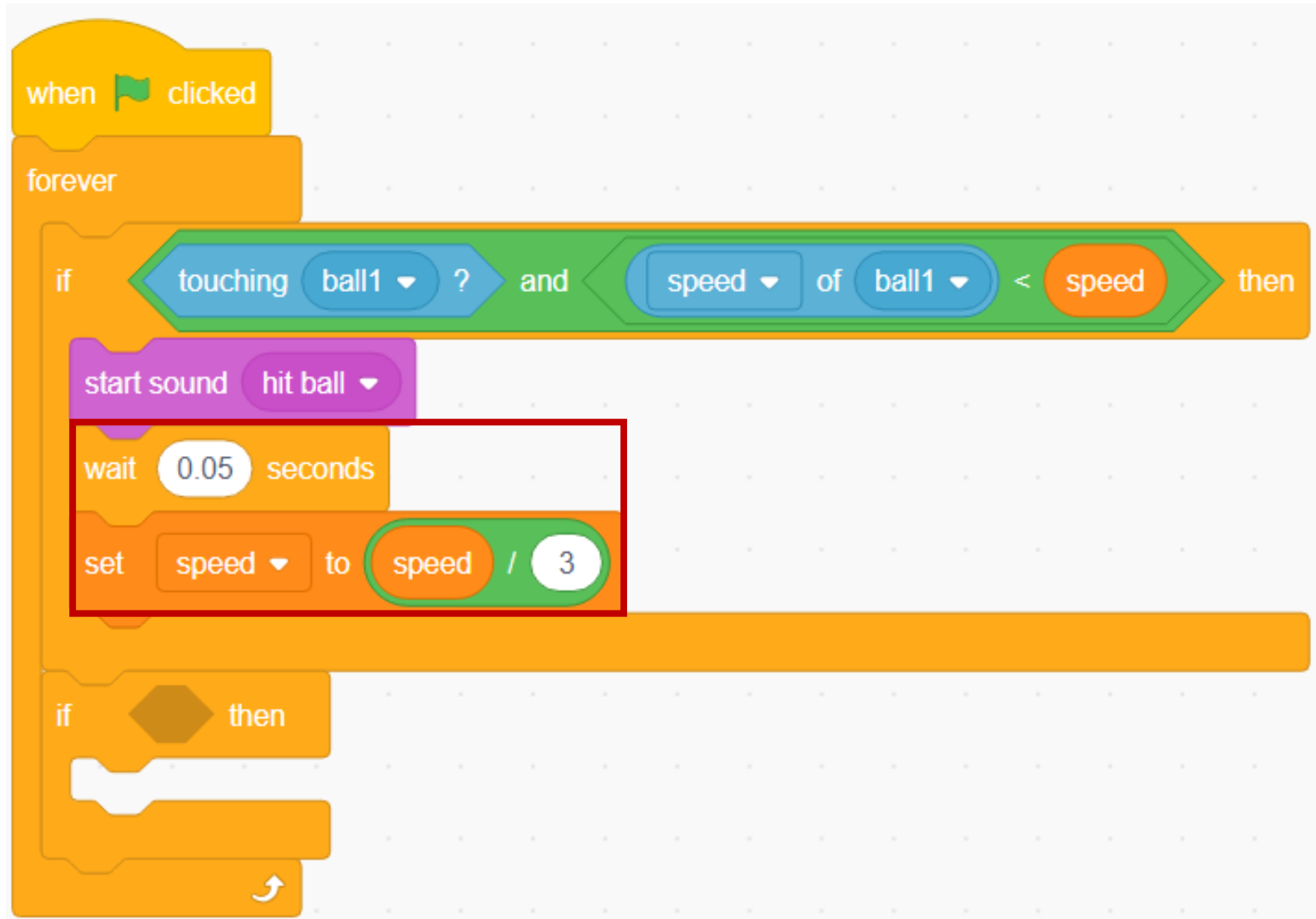




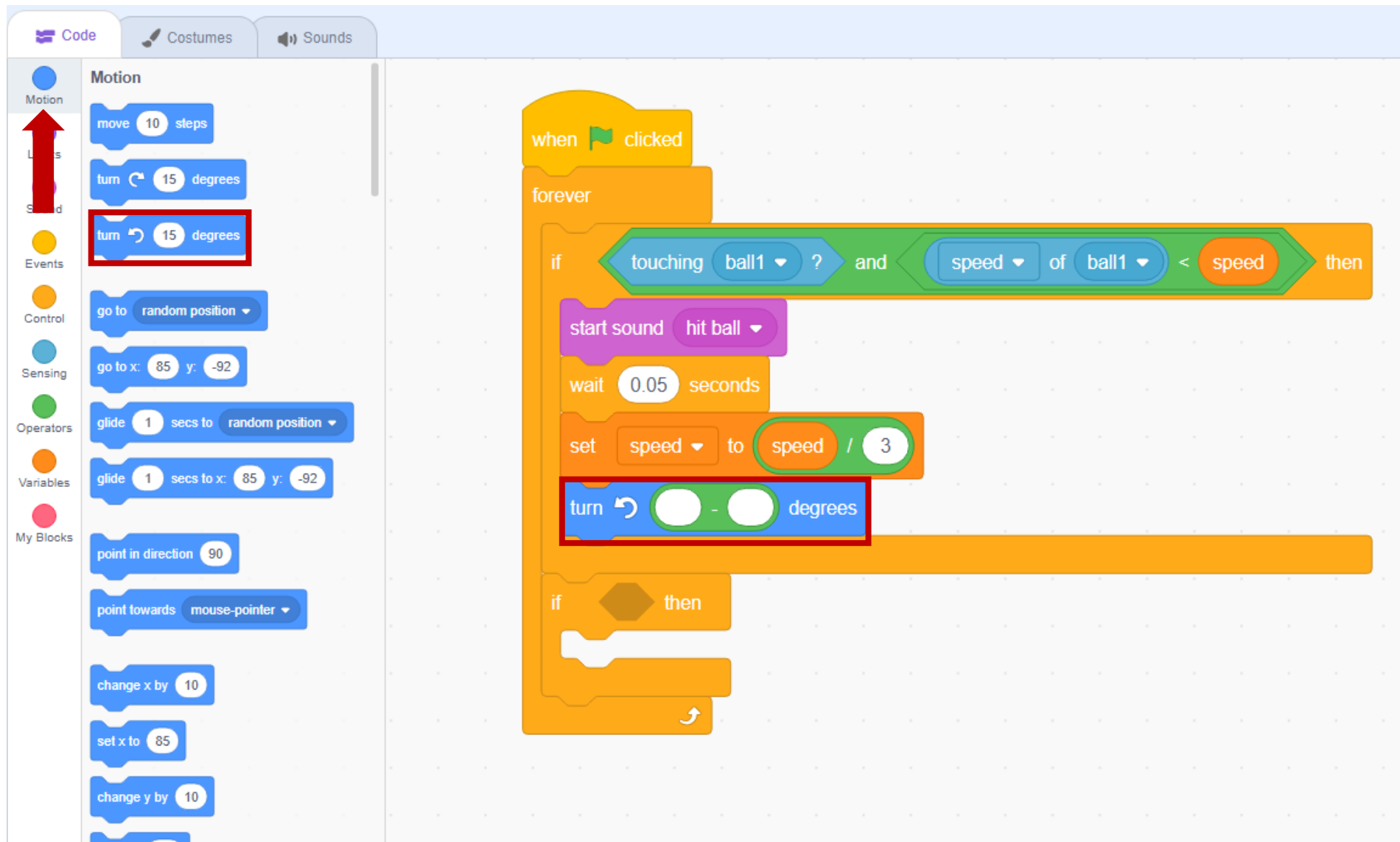
The image shows the Scratch code editor interface. On the left, the 'Variables' panel is visible, showing a variable named 'Balls in' set to 0. A red arrow points to the 'Variables' category in the left sidebar. The main workspace displays a script starting with 'when green flag clicked', followed by a 'forever' loop. Inside the loop, there is an 'if' condition: 'touching ball1?' and 'speed of ball1 < speed'. If true, it triggers 'start sound hit ball', 'wait 1 seconds', and 'set speed to 0'. There is also a 'show variable' block for 'Balls in' and a 'hide variable' block for 'Balls in'. The 'Variables' panel on the left shows 'Balls in' set to 0 and 'speed' set to 0.



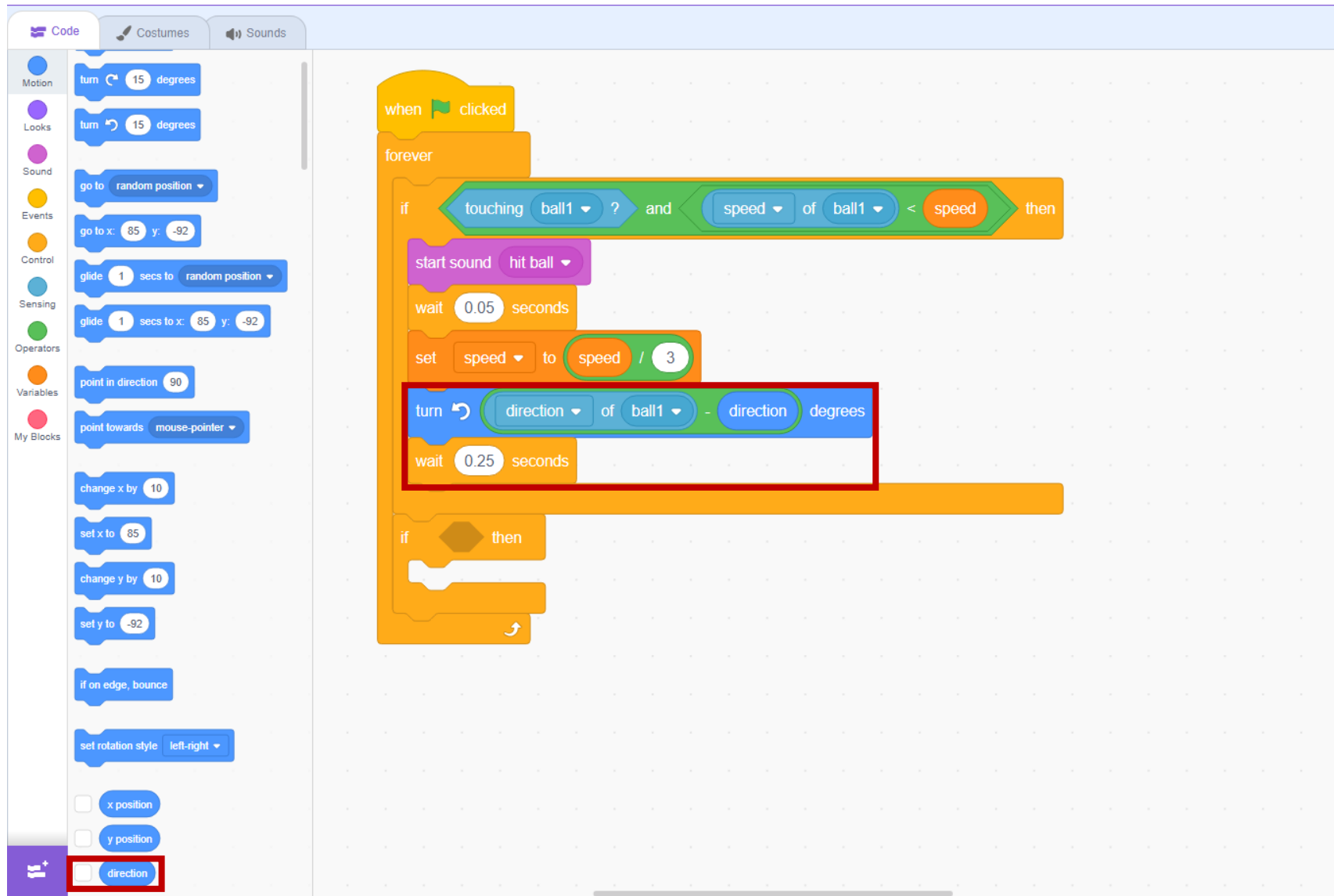
Change the wait time from 1 to 0.05 seconds. Then go to “Operators” and grab the “\_\_ / \_\_” block and put it into the “set speed to \_\_” block. Then go to “Variables” to grab the speed variable and put it into the first blank of the “\_\_ / \_\_” block. Then type 3 for the last blank. It should look like the section shown below.



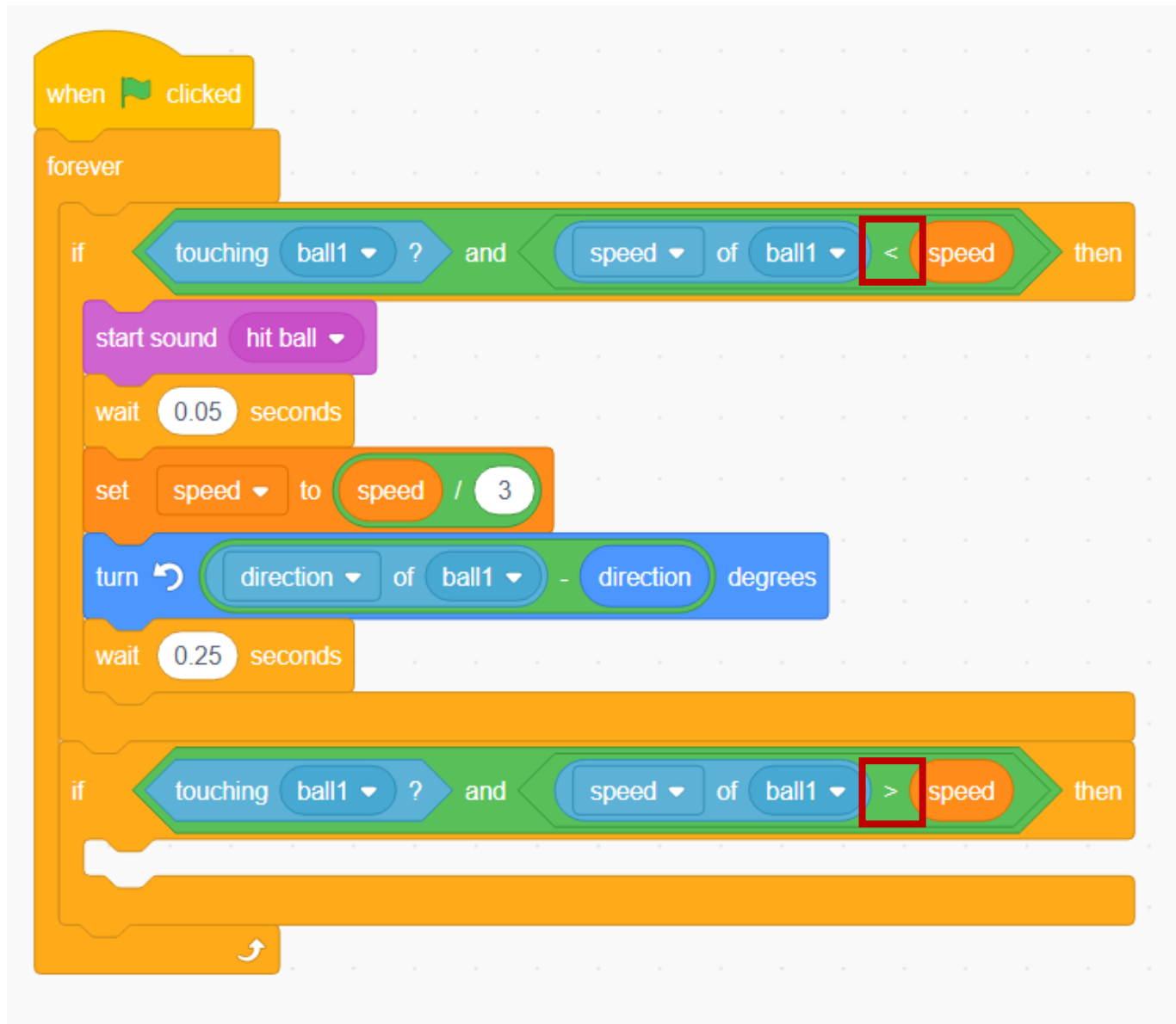
Go to “Motion” and grab the “turn counterclockwise 15 degrees” block and put it into the “if-then” statement. Then go to “Operators” and grab the “\_ - \_” block and put it into the “turn counterclockwise 15 degrees” block.



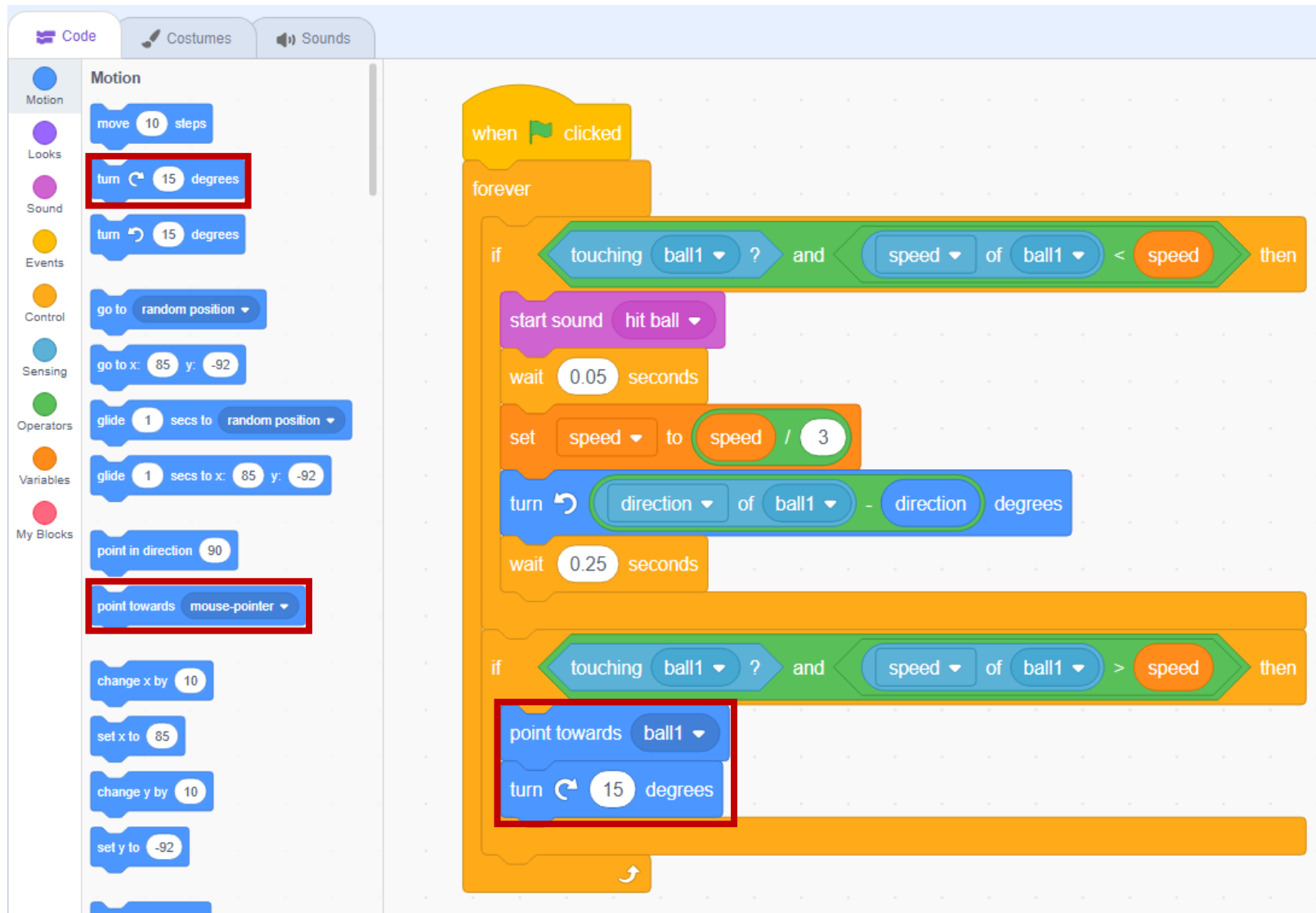
Go to “Sensing” and grab the “backdrop # of Stage” and add it to the first blank. Change variables so it says, “direction of ball 1”. Then go to “Motion” and grab the “direction” block and attach it to the second blank. Then go to control add grab the “wait 1 seconds” block and change it to 0.25.



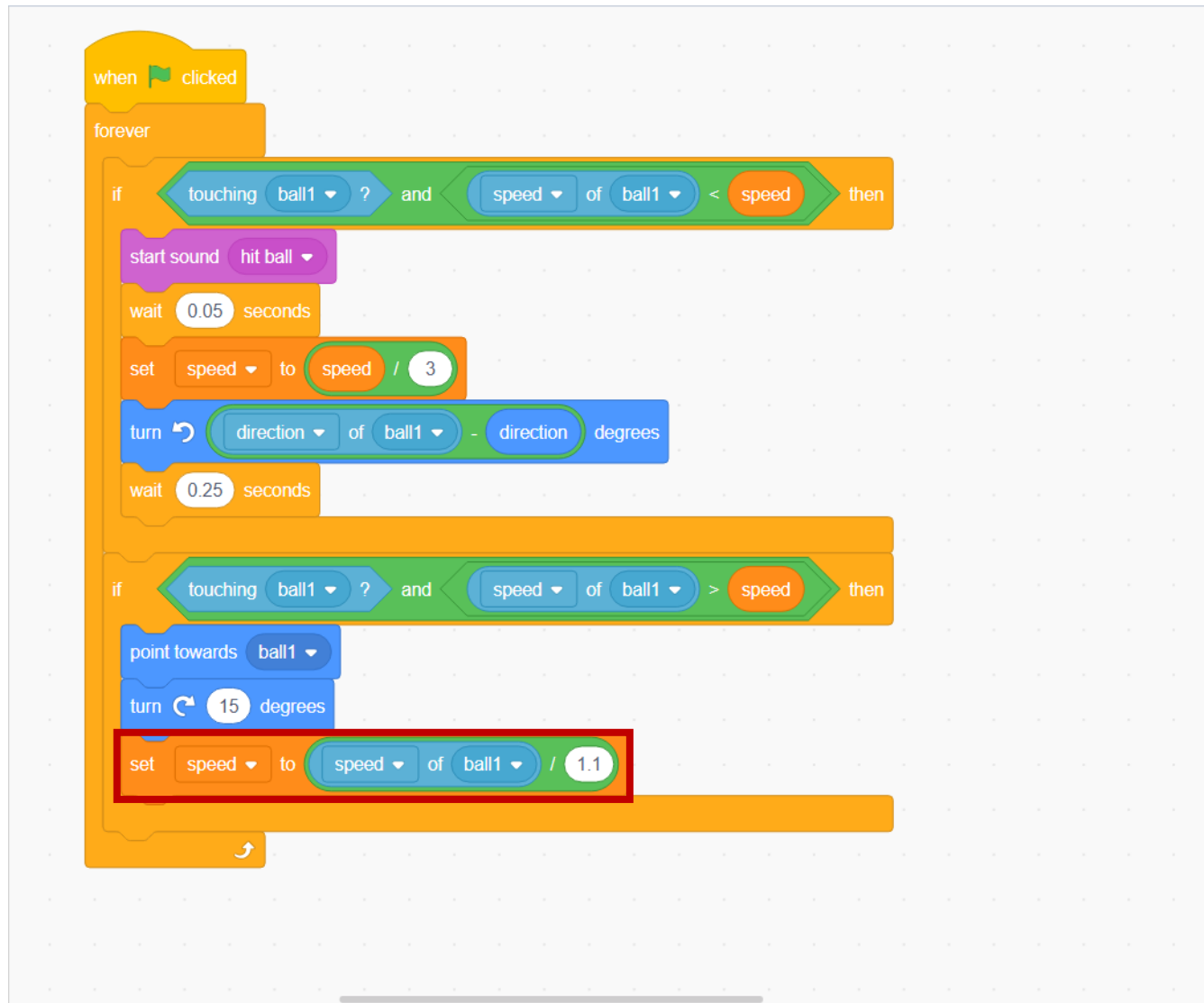
The second “if-then” block is very similar to the first. Insert the same blocks in the “if-then” blank, but grab “>” instead of the “<”.



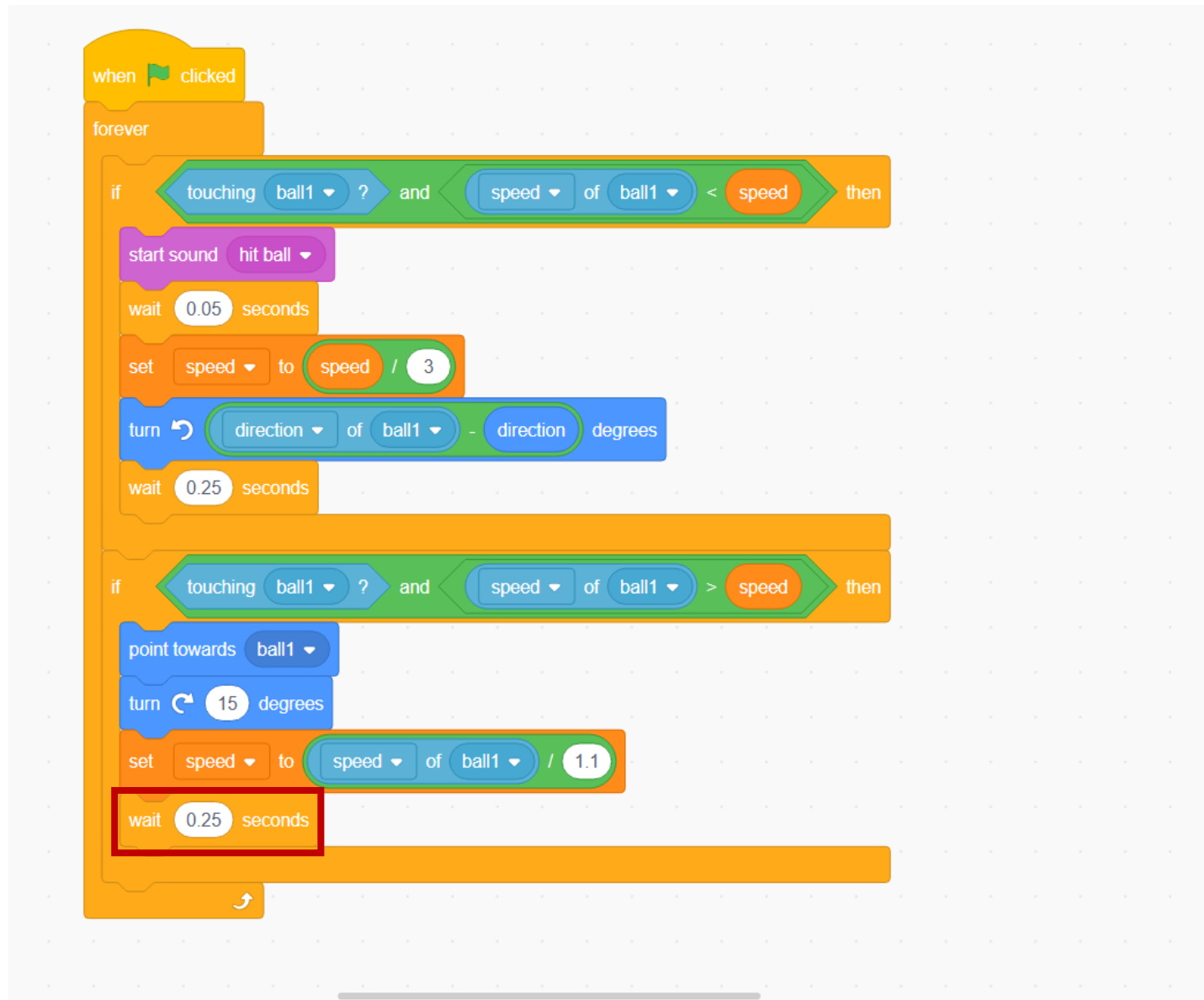
Go to the “Motion” tab and grab the “point towards mouse-pointer” block and change it “point towards ball 1”. Then add a “turn clockwise 15 degrees” block. Keep the same degree.



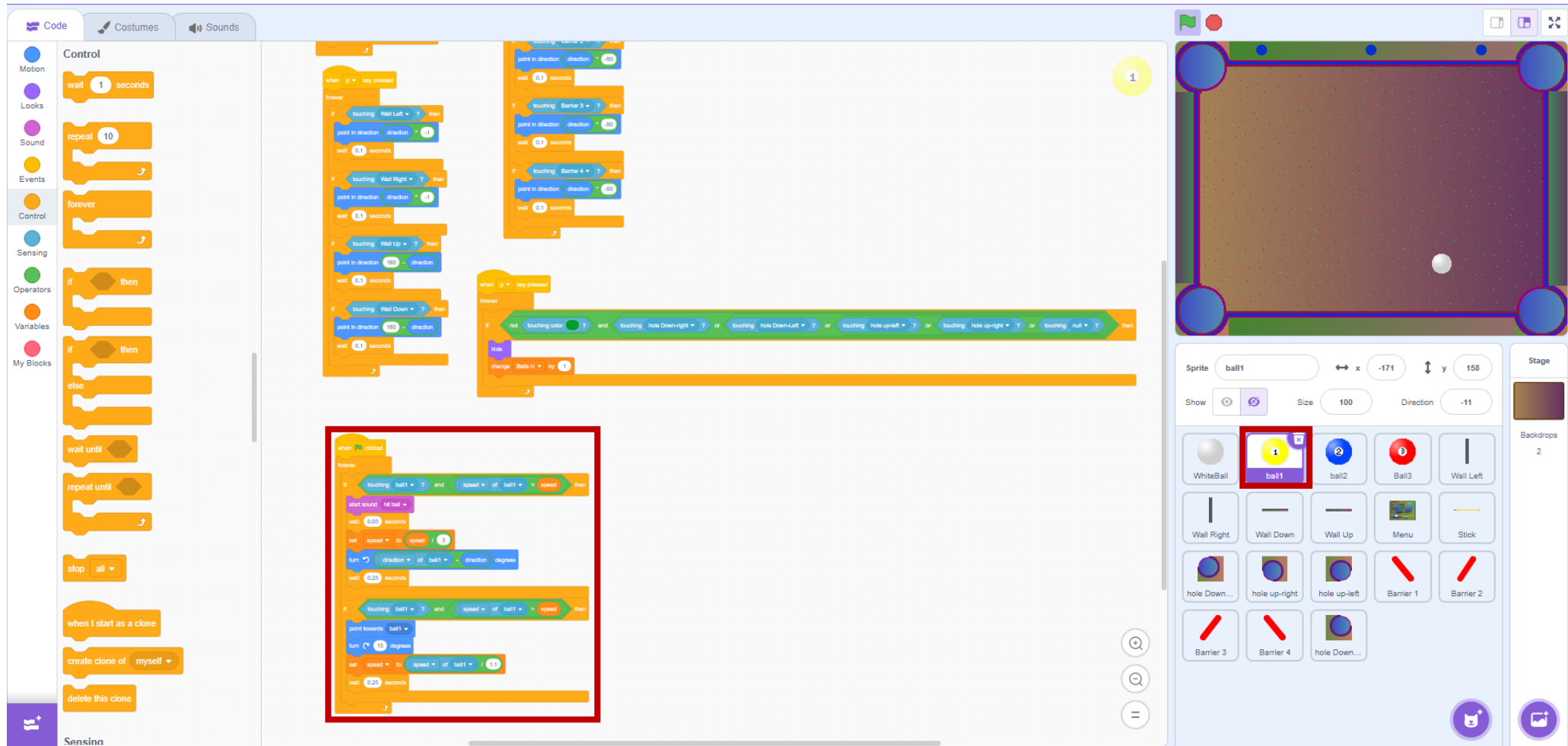
Go to the “Variables” tab and grab the “set balls in to 0” block and change the variable to “speed”. Then go to “Operators” and grab the “\_/\_”. Then go to “Sensing” and grab the “backdrop # of Stage” and insert it into the first blank. Change it to “speed of ball 1”. Then type 1.1 in the second blank.



Go to the “Control” tab and grab the “wait 1 second” block. Change the block so it will wait 0.25 seconds.



Click on the “when flag clicked” block in the code and then use ctrl+c to copy the code block. Now select the “ball 1” sprite. Use ctrl+v to paste the code onto the canvas. You may need to move around the canvas to find the code block.



The image shows the Scratch code editor interface. The code area contains several blocks, including a 'when flag clicked' block at the bottom, which is highlighted with a red rectangle. The 'ball 1' sprite is selected in the sprite area on the right. The code blocks include:

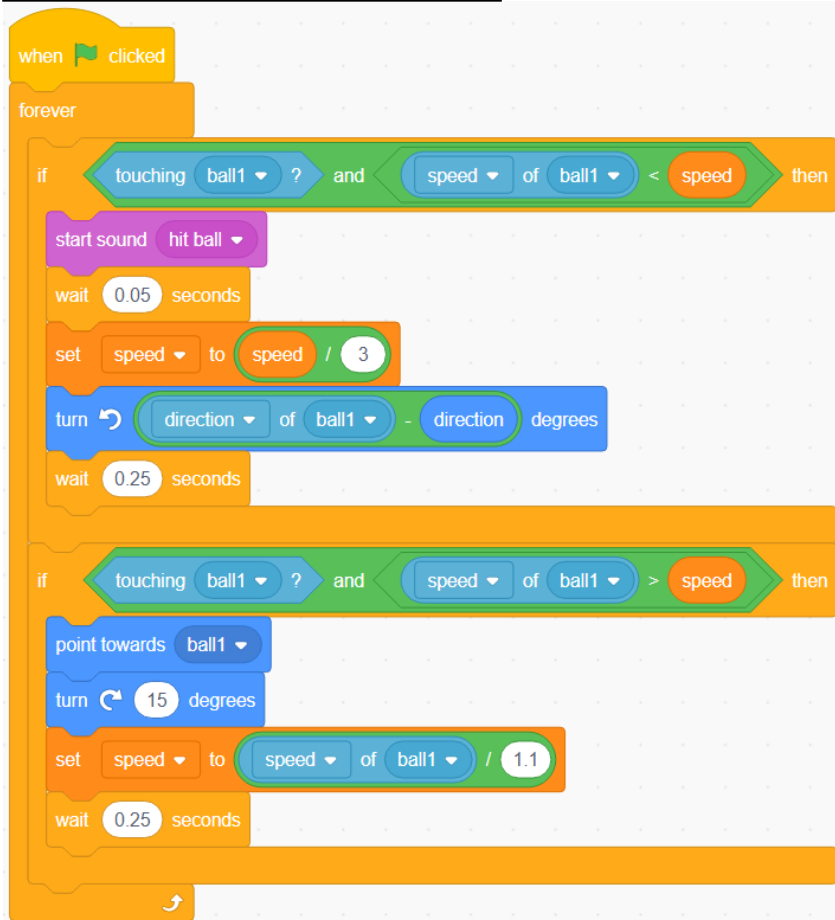
- when flag clicked
- repeat 10
- wait 1 seconds
- point in direction direction - 15
- wait 0.5 seconds
- point in direction direction + 15
- wait 0.5 seconds
- point in direction direction - 45
- wait 0.5 seconds
- point in direction direction + 45
- wait 0.5 seconds
- when I start as a clone
- create clone of myself
- delete this clone

The 'ball 1' sprite is a white ball with a yellow number 1. The 'ball 2' and 'ball 3' sprites are also visible in the sprite area.

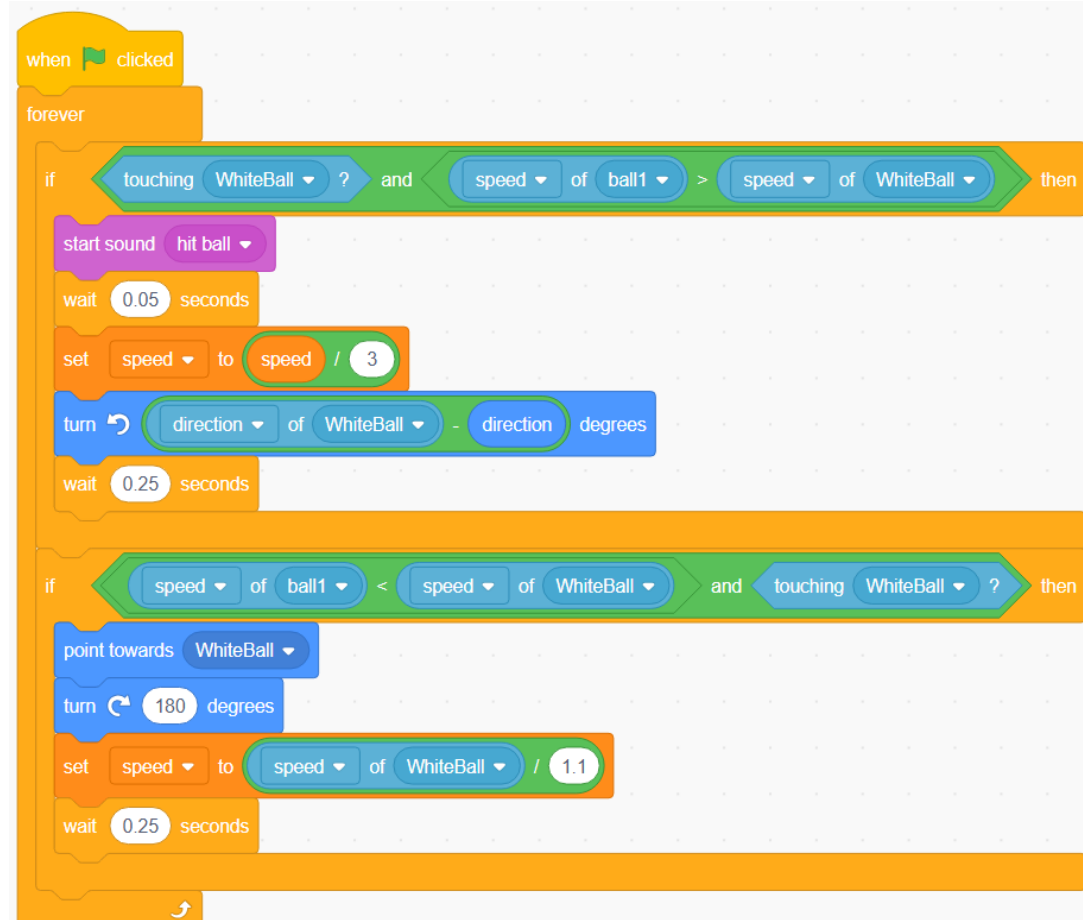


Now we have to change all instances of “ball 1” to “white ball”. Make sure that the block is still checking for speed not direction. We also have to change the “turn clockwise 15 degrees” block to “turn clockwise 180 degrees”. Use the two pictures below to compare the different code blocks.

Code block in “white ball” sprite

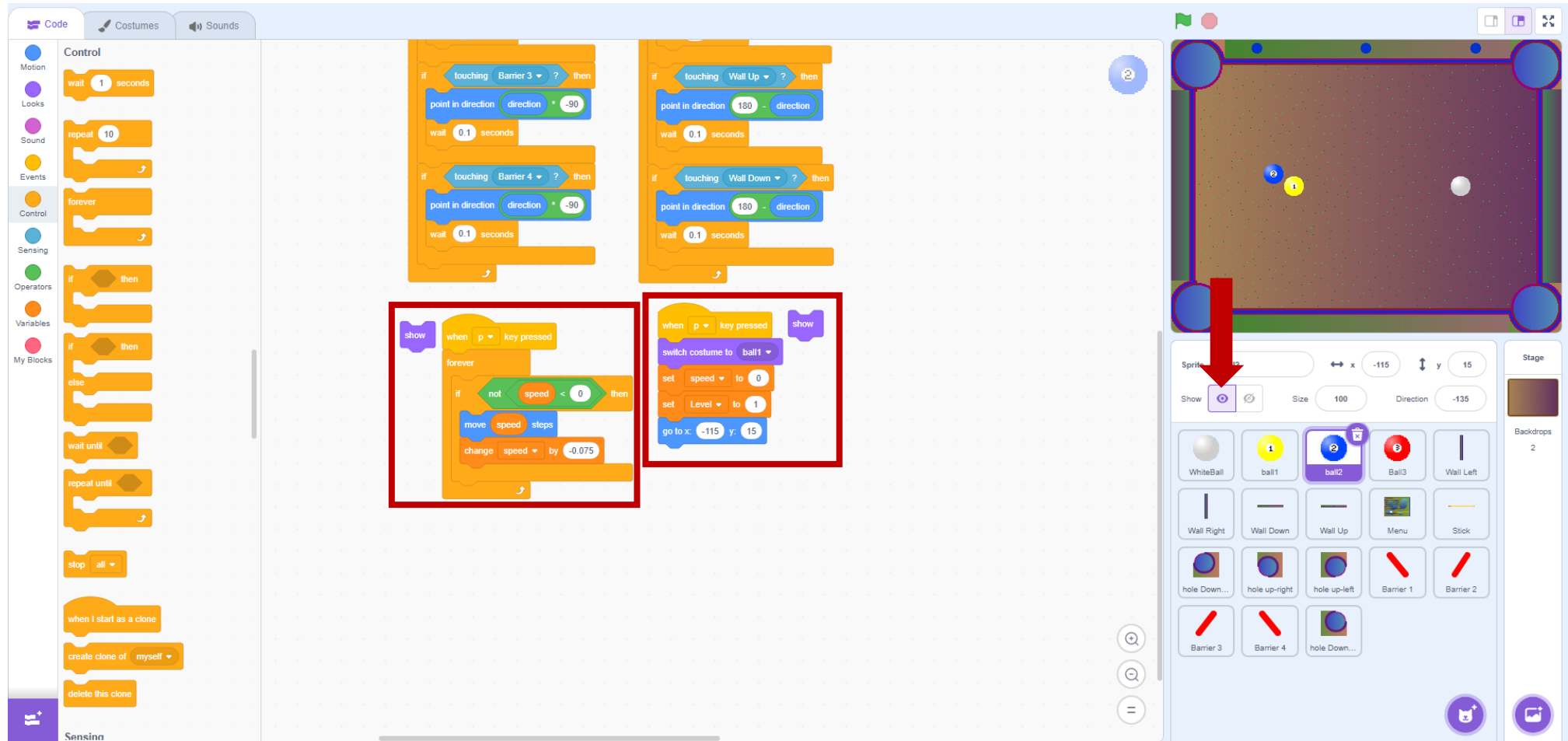


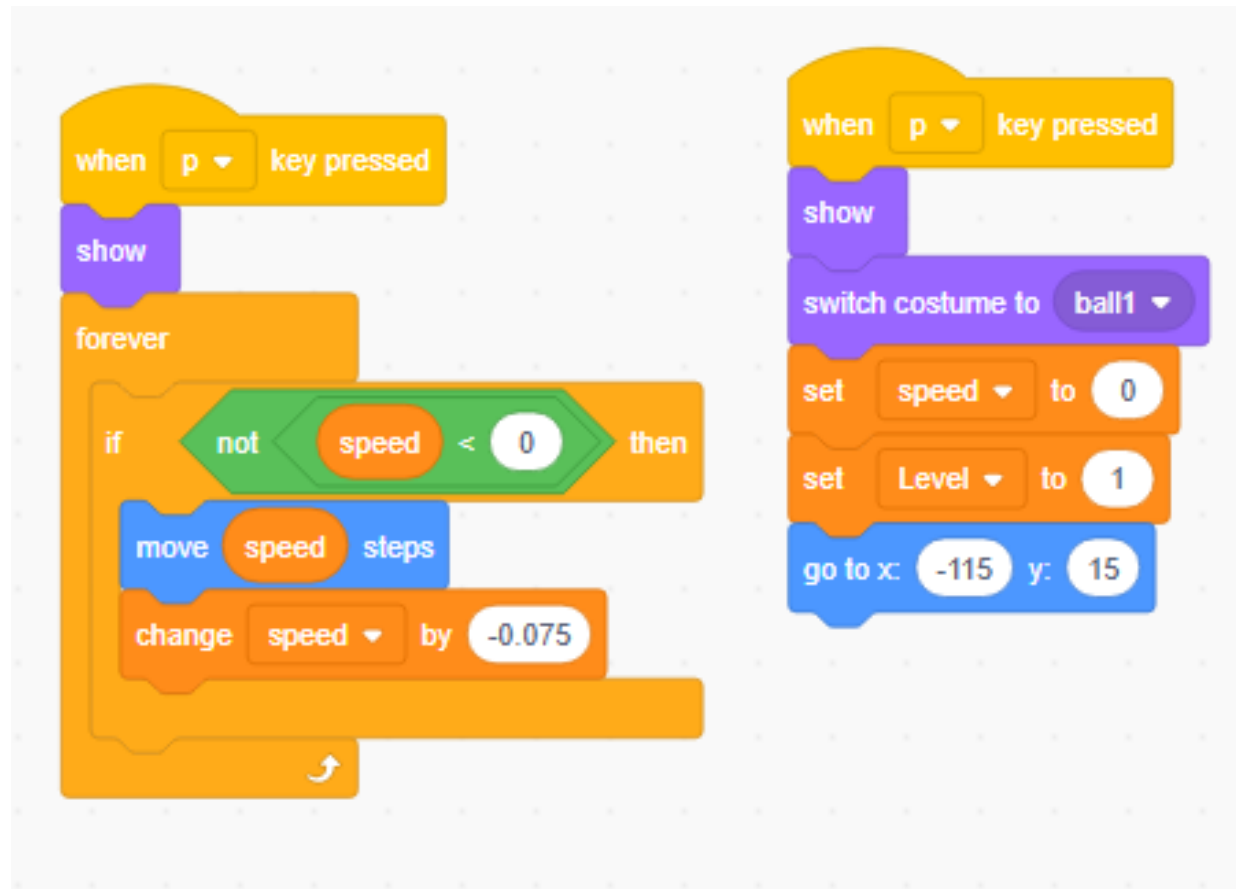
Code block in “ball 1” sprite



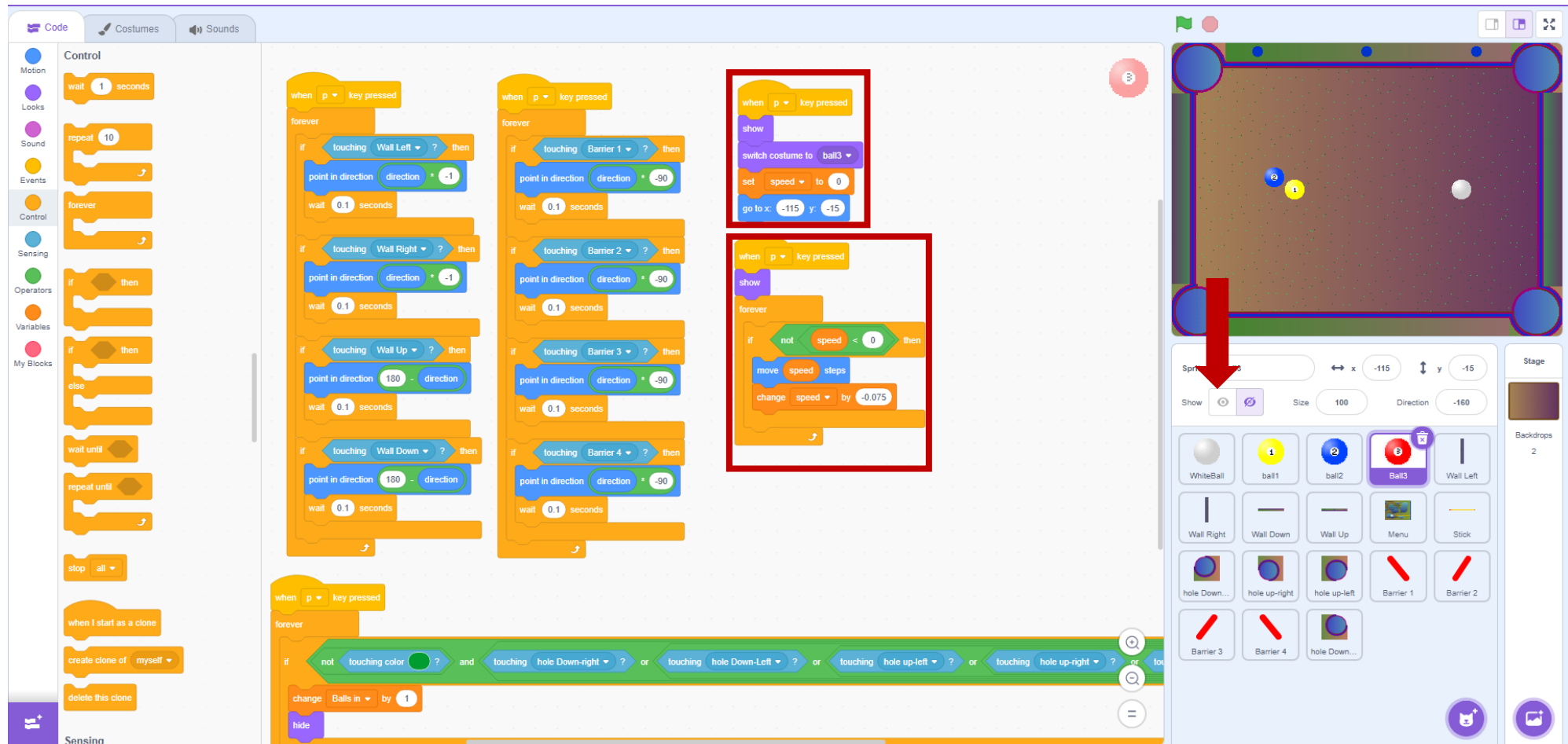
**This section will cover level 2 of the game**

First go to the “ball 2” sprite and click on the eyeball to make the ball visible. Then attach the two “show” block to the adjacent code blocks they are next to.





Repeat this process for ball 3.



The image displays the Scratch code editor with three event-driven scripts for ball movement, each triggered by a key press. The scripts are designed to move a ball towards a wall or barrier and then change its direction and speed.

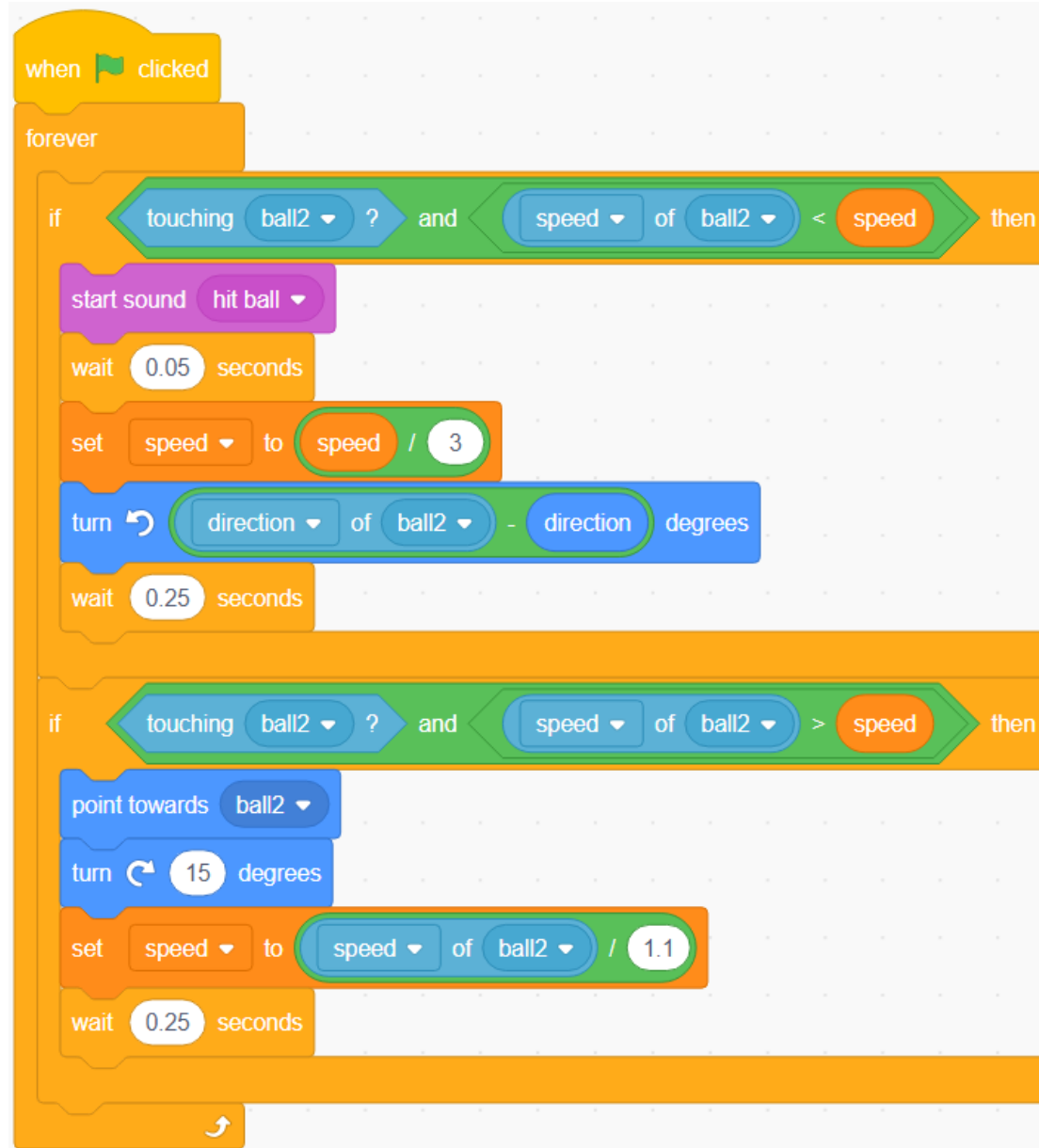
**Script 1 (Left):** When key pressed, enter a forever loop. If touching Wall Left, point in direction  $\text{direction} - 1$ , wait 0.1 seconds, and repeat. If touching Wall Right, point in direction  $\text{direction} + 1$ , wait 0.1 seconds, and repeat. If touching Wall Up, point in direction  $180 - \text{direction}$ , wait 0.1 seconds, and repeat. If touching Wall Down, point in direction  $180 + \text{direction}$ , wait 0.1 seconds, and repeat.

**Script 2 (Middle):** When key pressed, enter a forever loop. If touching Barrier 1, point in direction  $\text{direction} - 90$ , wait 0.1 seconds, and repeat. If touching Barrier 2, point in direction  $\text{direction} + 90$ , wait 0.1 seconds, and repeat. If touching Barrier 3, point in direction  $\text{direction} - 90$ , wait 0.1 seconds, and repeat. If touching Barrier 4, point in direction  $\text{direction} + 90$ , wait 0.1 seconds, and repeat.

**Script 3 (Right):** When key pressed, show the ball, switch costume to ball3, set speed to 0, and go to x: -115 y: -15. Then enter a forever loop. If not speed < 0, then move speed steps, change speed by -0.075, and repeat.

The stage preview on the right shows a ball moving towards a wall. A red arrow points to the ball's position on the stage.

Select the “white ball” sprite and create the same code block as the one in level 1. Change all instance of “ball 1” to “ball 2”.



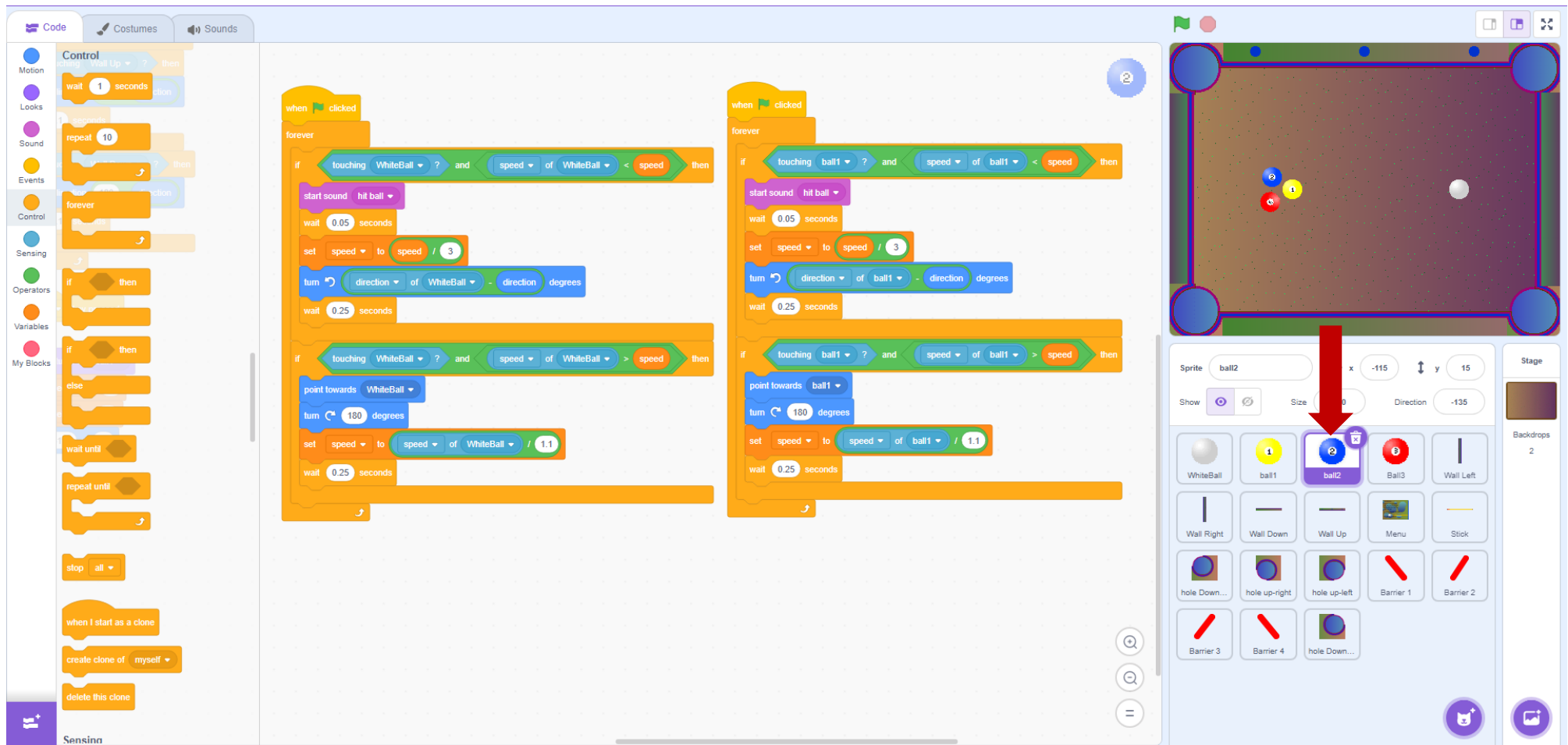
Now repeat the same process in the ball 1 sprite and the ball 2 sprite. The correct code for the interactions for the white ball, ball 1, and ball 2 are provided below. Now try to create the code for ball 3.

The image shows the Scratch code editor with a script for a ball collision. The script is as follows:

```

when clicked
  forever loop
    if touching ball2 and speed of ball2 < speed then
      start sound hit ball
      wait 0.05 seconds
      set speed to speed / 3
      turn direction of ball2 - direction degrees
      wait 0.25 seconds
    if touching ball2 and speed of ball2 > speed then
      point towards ball2
      turn 180 degrees
      set speed to speed of ball2 / 1.1
      wait 0.25 seconds
  
```

The stage view on the right shows a pool table with three balls: ball1 (white), ball2 (blue), and ball3 (red). A red arrow points to the 'ball1' sprite in the sprite list.



The image displays the Scratch code editor with two scripts for ball movement and collision detection. The left script is for a 'WhiteBall' and the right script is for a 'ball1'.

**Left Script (WhiteBall):**

- when clicked
- forever loop:
  - if touching WhiteBall? and speed of WhiteBall < speed then:
    - start sound hit ball
    - wait 0.05 seconds
    - set speed to speed / 3
    - turn direction of WhiteBall - direction degrees
    - wait 0.25 seconds
  - if touching WhiteBall? and speed of WhiteBall > speed then:
    - point towards WhiteBall
    - turn 180 degrees
    - set speed to speed of WhiteBall / 1.1
    - wait 0.25 seconds

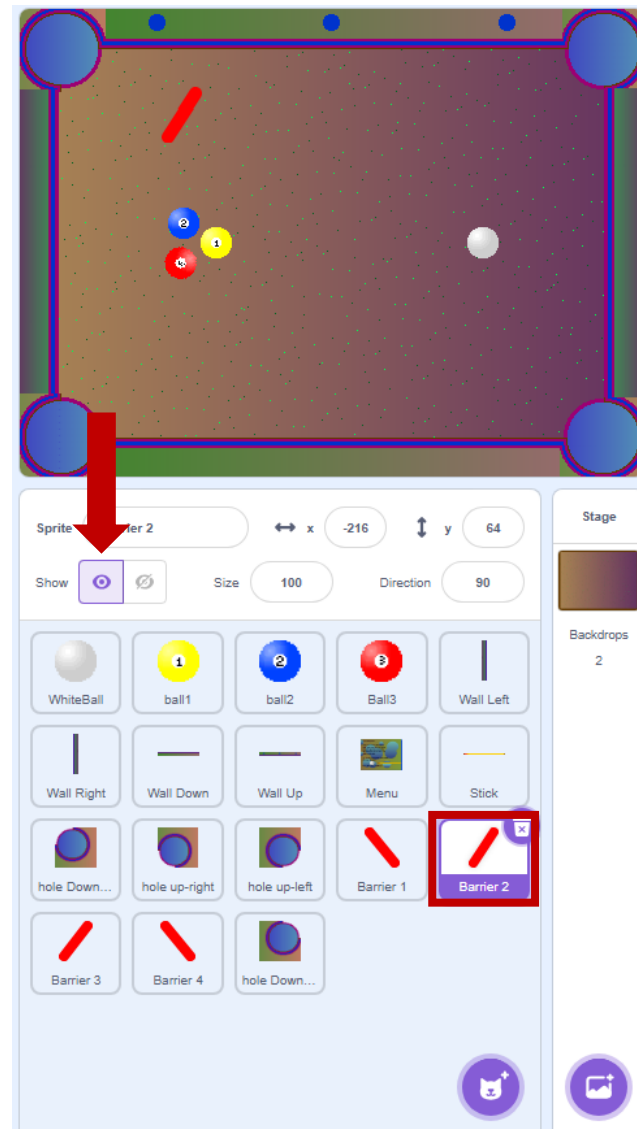
**Right Script (ball1):**

- when clicked
- forever loop:
  - if touching ball1? and speed of ball1 < speed then:
    - start sound hit ball
    - wait 0.05 seconds
    - set speed to speed / 3
    - turn direction of ball1 - direction degrees
    - wait 0.25 seconds
  - if touching ball1? and speed of ball1 > speed then:
    - point towards ball1
    - turn 180 degrees
    - set speed to speed of ball1 / 1.1
    - wait 0.25 seconds

The right panel shows the stage with a ball2 sprite selected. A red arrow points to the 'ball2' sprite in the sprite list.

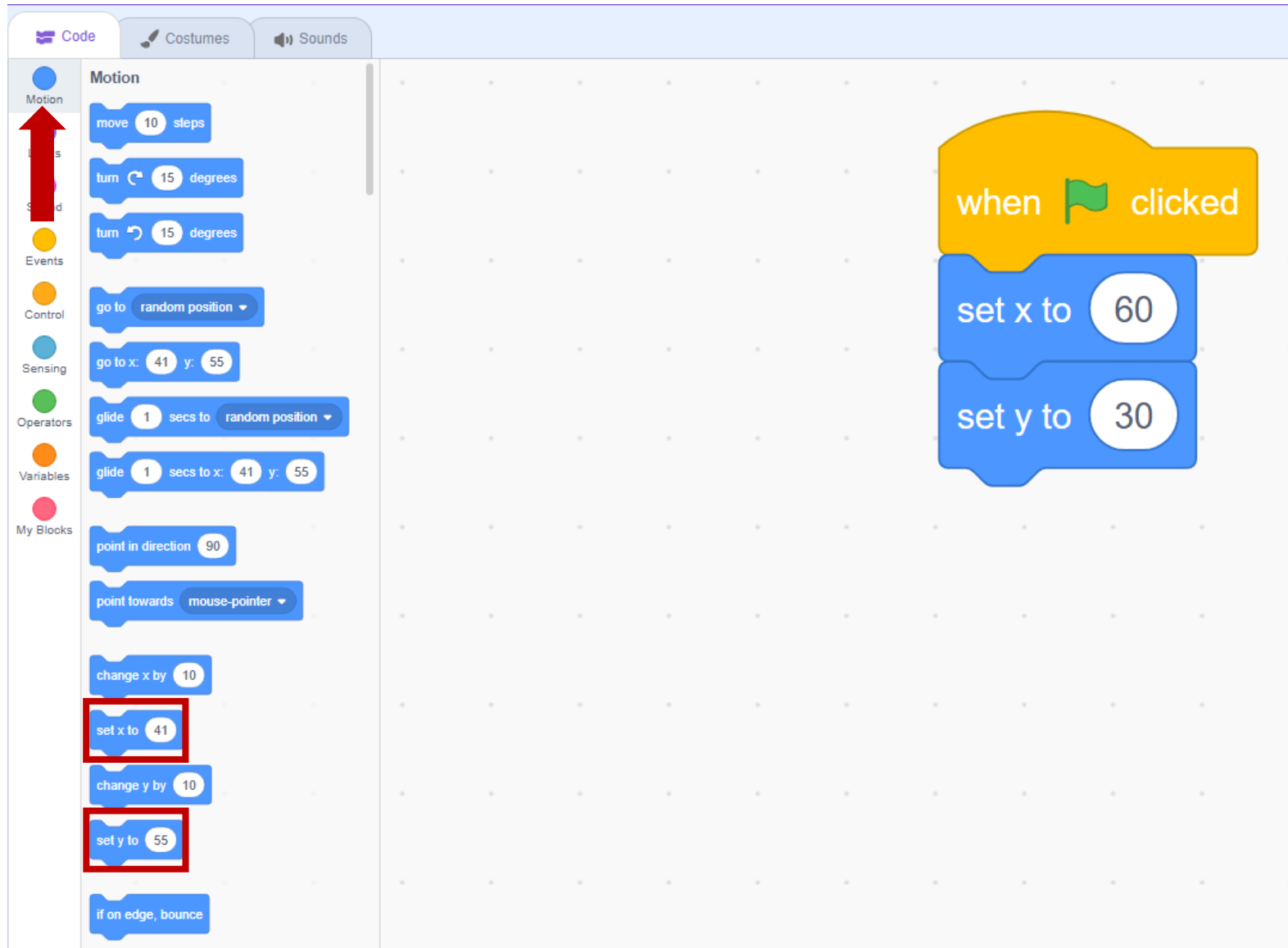
**This section will cover level 3 of the game**

Go to each of the barriers and click on the eyeball to make them visible.

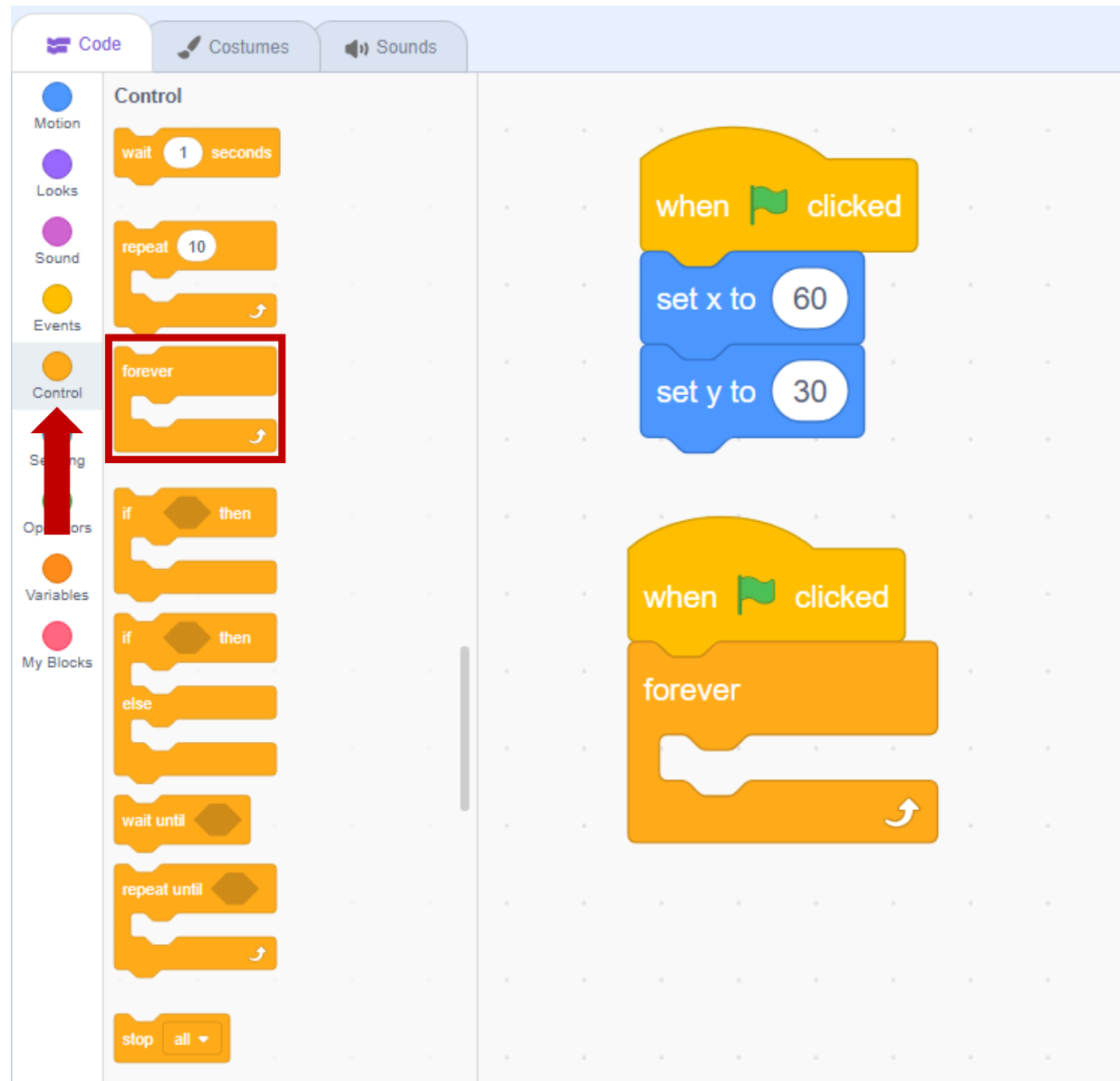




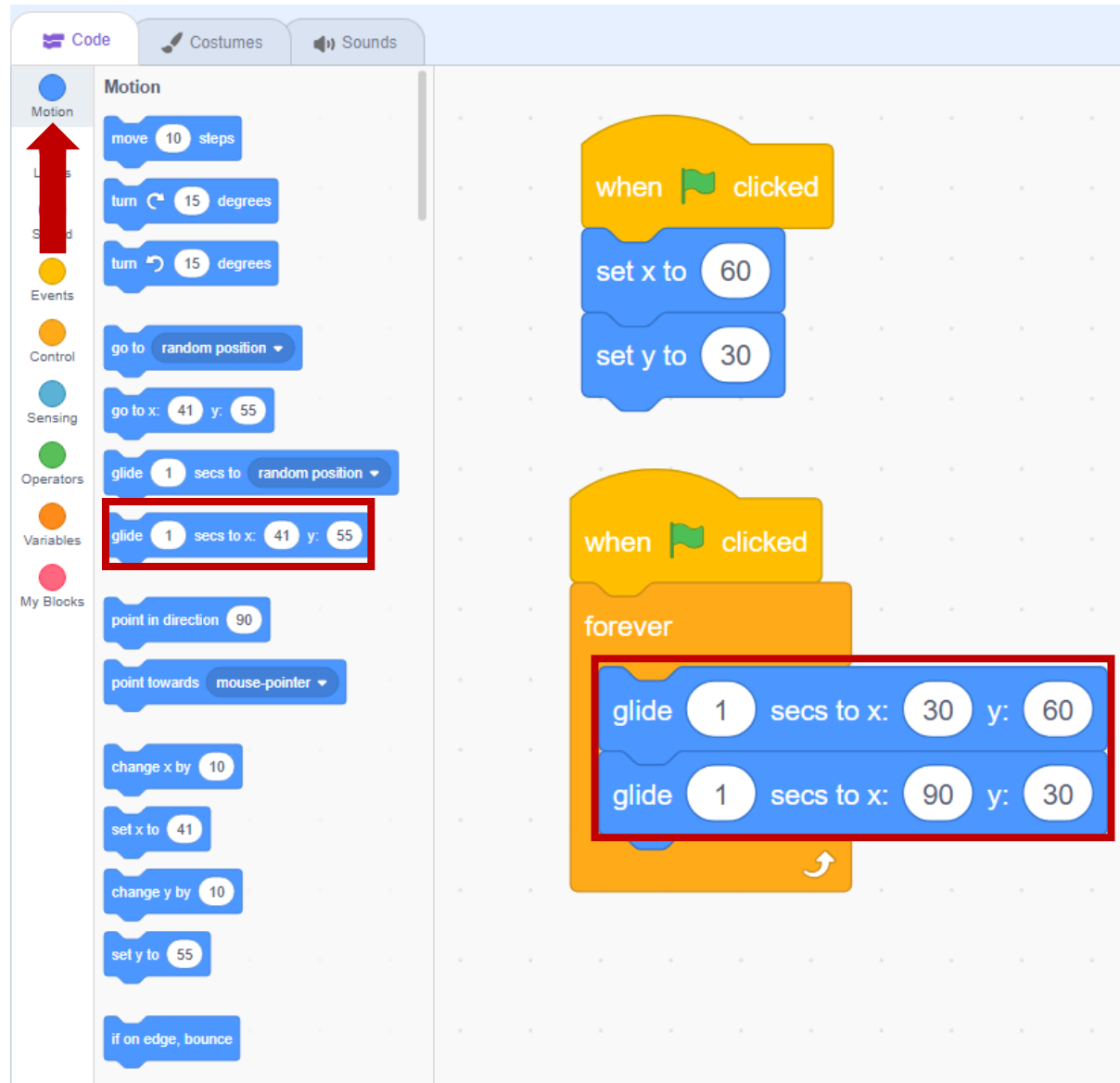
Go to the “Events” tab and select the “when flag pressed” block. Then go to the “Motion” tab and grab a “set x to \_\_\_” and a “set y to \_\_\_” block. Change the x to 60 and the y to 30.



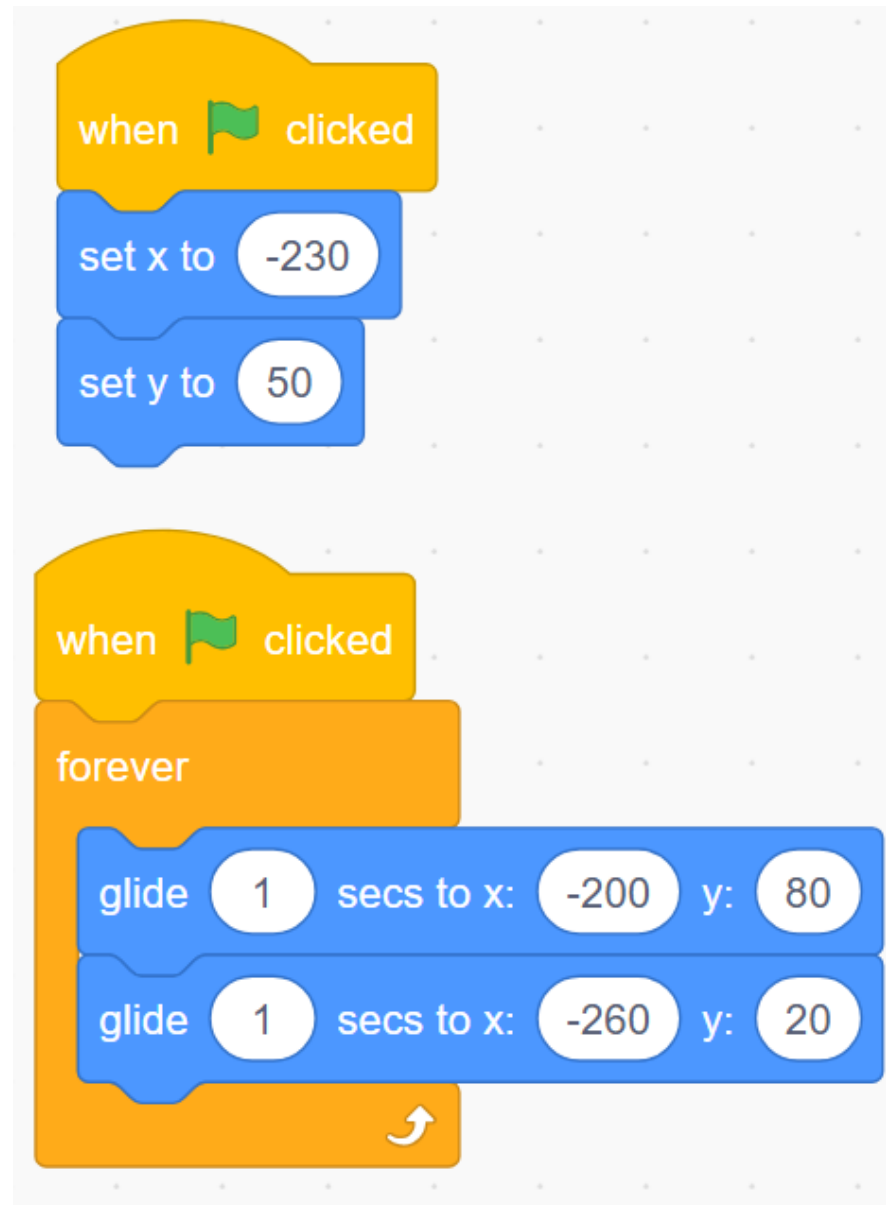
Go back to the “Events” tab and grab the “when flag pressed” block. Then go to the “Control” tab and grab a forever loop.



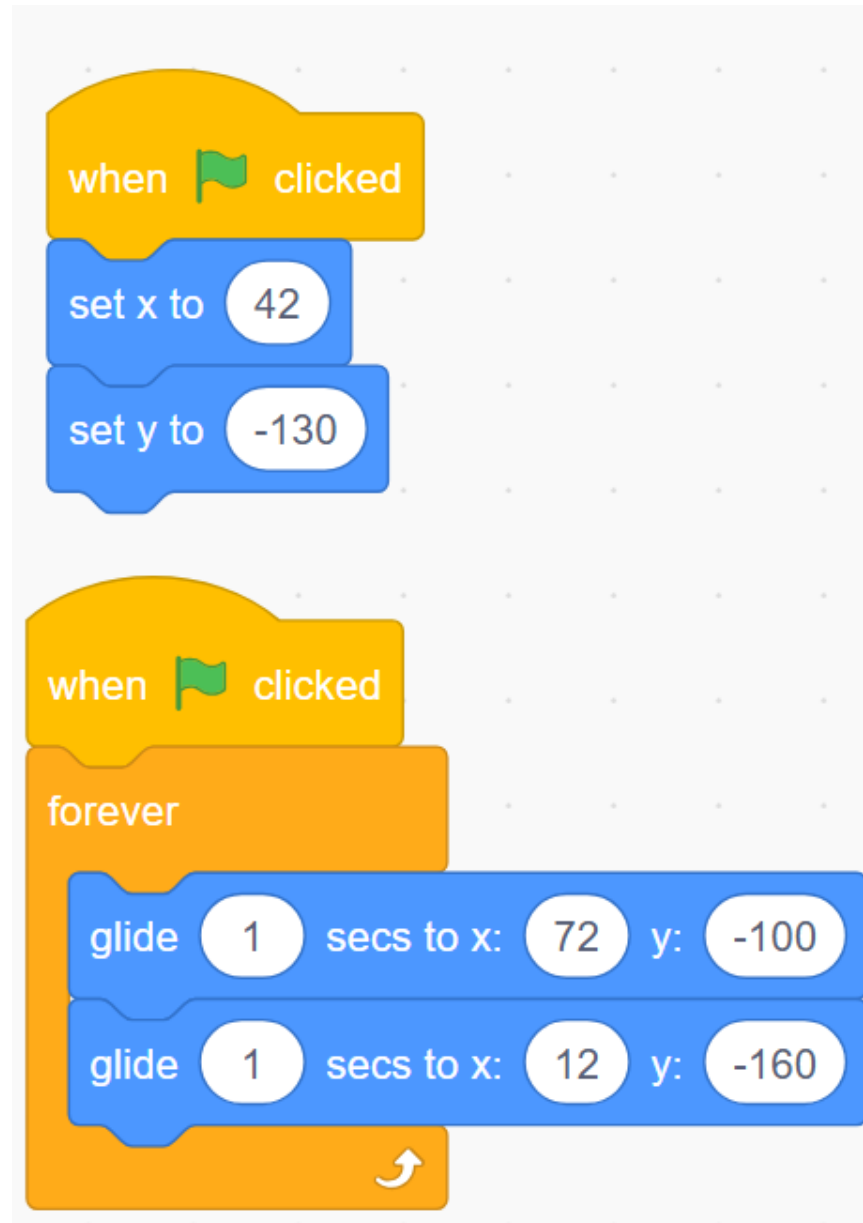
Go to the “Motion” tab and grab two “glide \_\_ secs to x: \_\_ y: \_\_” and insert them both inside the forever loop. Set the glide to 1 sec for both. Set the x to 30 and y to 60 for one of the glide blocks. Then set the x to 90 and y to 30 for the other.



This is a similar process for the remaining barriers. Below is the code for the barrier 2.

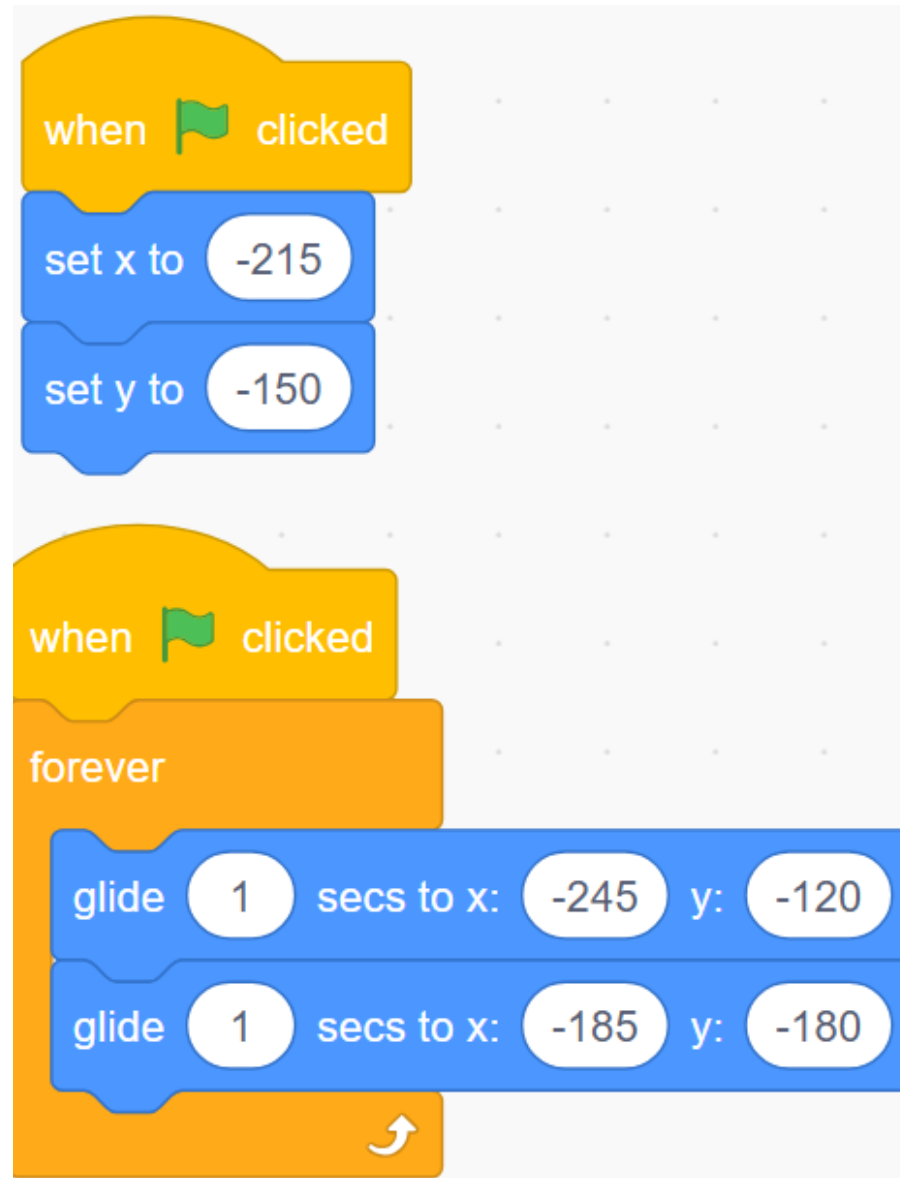


Below is the code for barrier 3.



<https://tapggc.org/>

Below is the code for barrier 4.



<https://tapggc.org/>