2017 Hour of Code with Tickle
Sphero Run

Overview

Team up with your Sphero for a good run! Move, spin and give it a few rounds of good turns. Just put some code blocks together on Tickle app to run your Sphero as you want it to. Try make it move in a square or aim for a figure 8. See if you and Sphero makes a good team completing all these fun challenges!

With Tickle, an user-friendly coding app, you can easily program LEGO WeDo 2.0 to move in a number of ways by simply dragging and dropping coding blocks to create a command for it to follow.

Lessons

Lesson 1: Intro to the Sphero

Overview

- Program a Sphero to move in a square pattern.

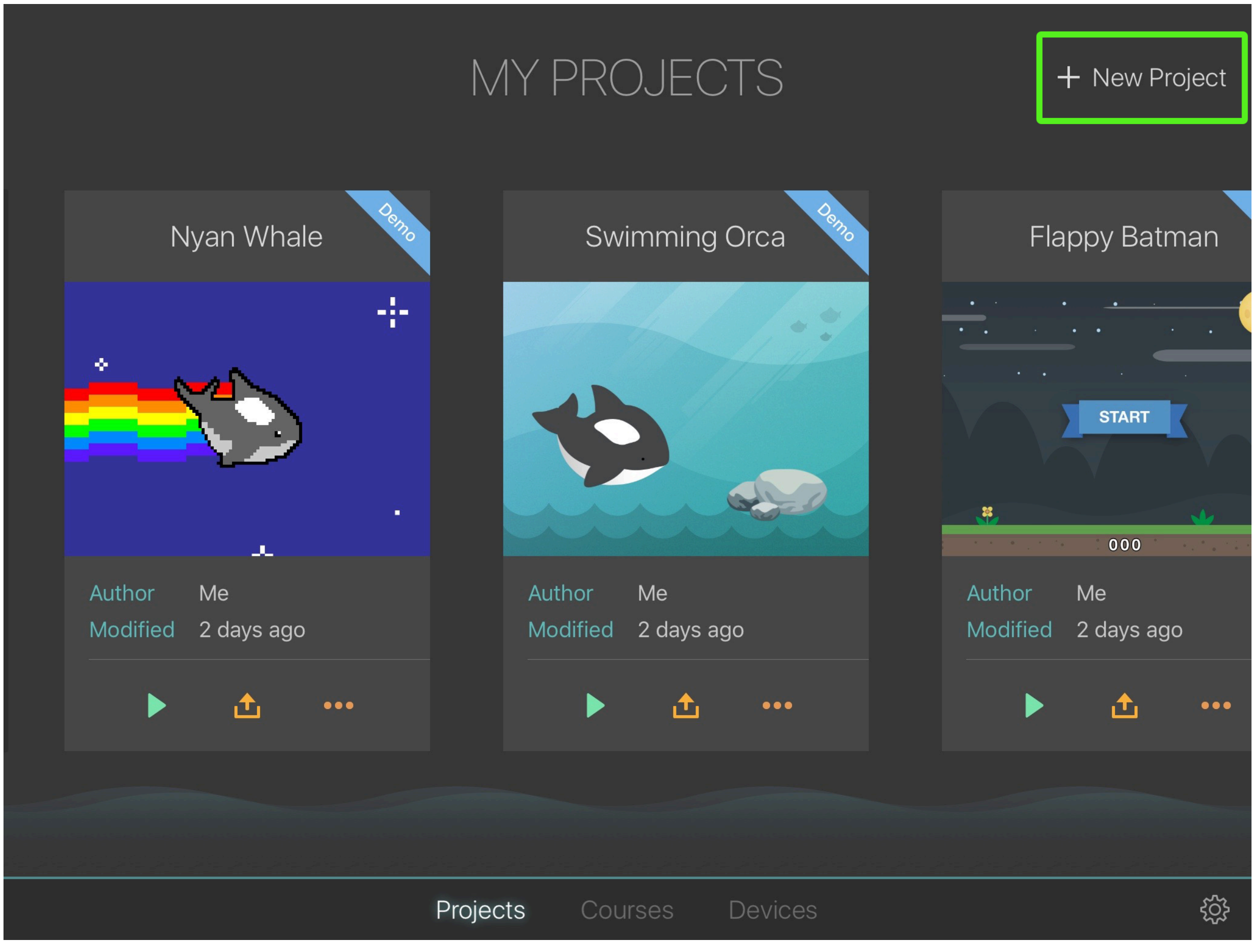
Objectives

- Become familiar with visual programming.
- Program a Sphero to move using basic motion commands.
- Learn how to use Loops, a control statement that allows code to be run multiple times consecutively.

Steps

1. Create a Sphero Project

- Create a new Sphero project by tapping `` `+ new Project` `` via “MY PROJECTS”. Choose the template for “Sphero”.



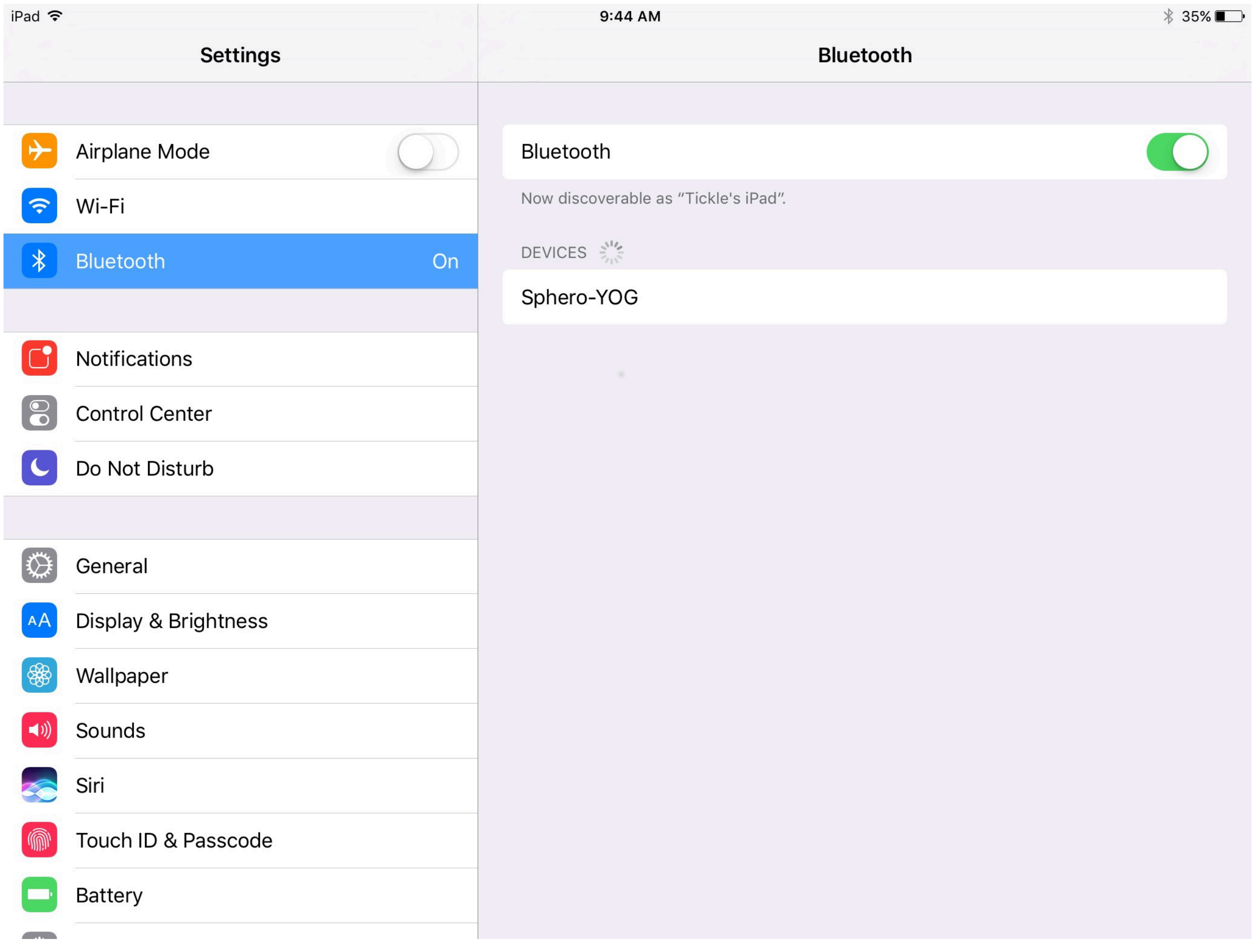
Nyan Whale
Author Me
Modified 2 days ago

Swimming Orca
Author Me
Modified 2 days ago

Flappy Batman
Author Me
Modified 2 days ago

2. Connect to the Sphero

- Activate Bluetooth on your iPhone or iPad, and select your Sphero.



Bluetooth

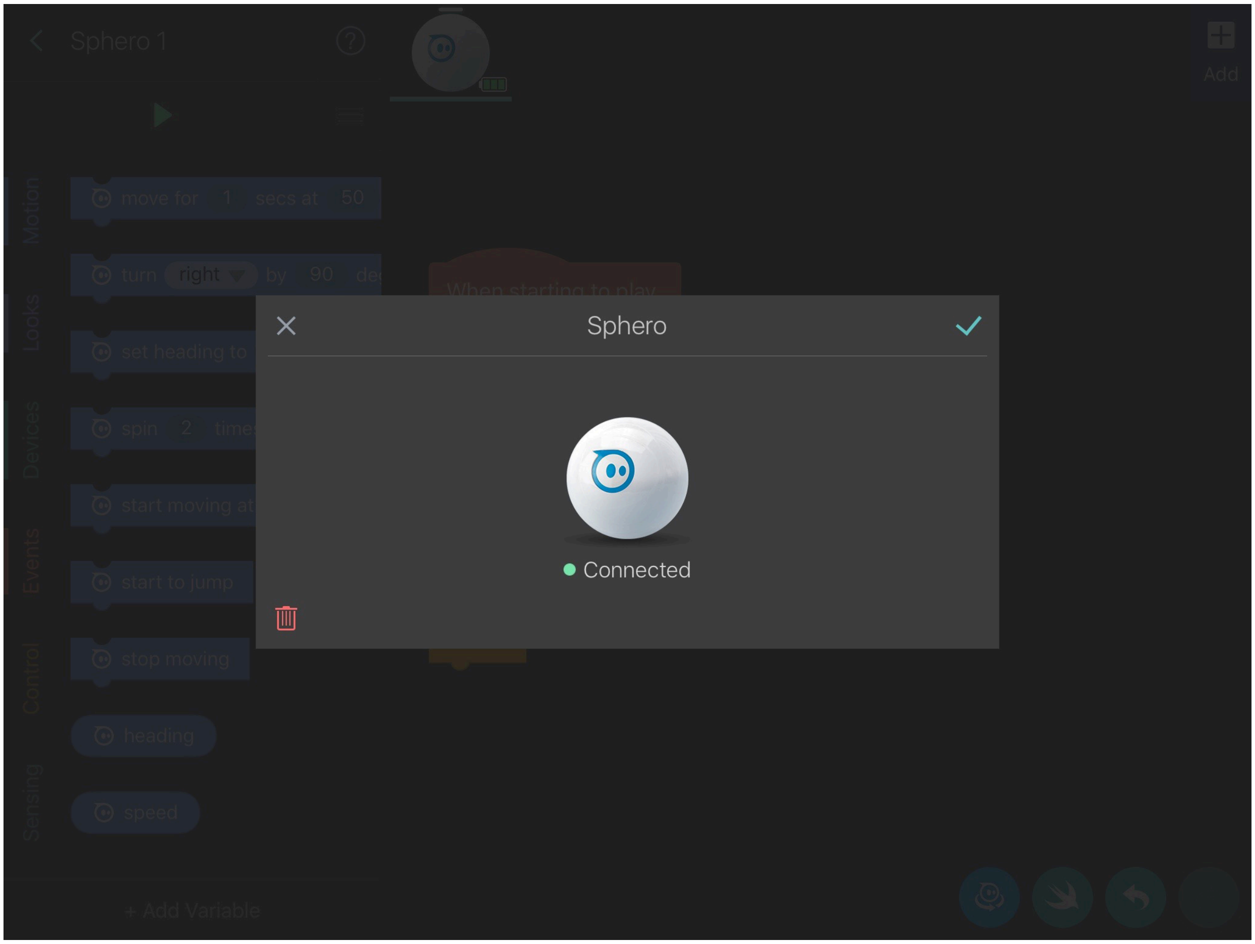
Bluetooth

Now discoverable as "Tickle's iPad".

DEVICES

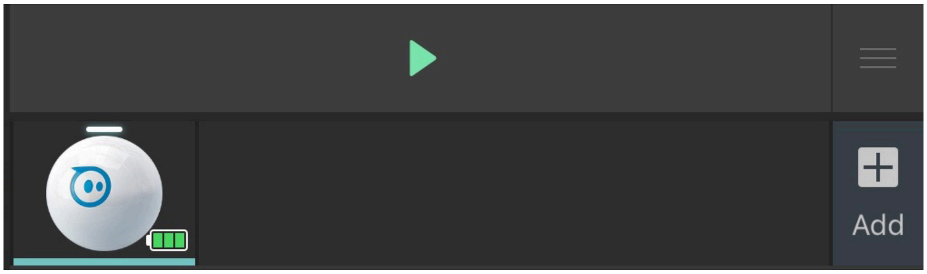
Sphero-YOG

- Your Sphero should automatically be detected and shown in the "Available Devices" menu in the Tickle App.



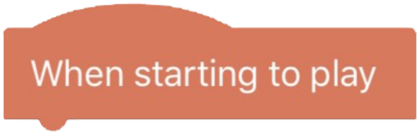
    - **Tip**: If your Sphero is powered on but not automatically connected, tap the Sphero icon to show troubleshooting instructions.

- The Sphero will show a battery status icon if it is successfully connected. Please charge the Sphero whenever the battery status becomes red, since the Sphero may stop responding to your commands.
    - Note: If the Sphero is not connected to Tickle, it may be due to the red battery status.

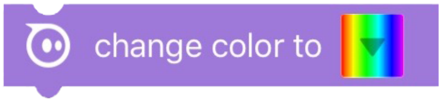


3. Block Review

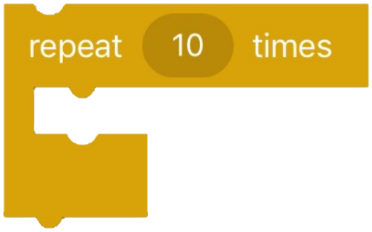
- Let’s start by reviewing the provided template that appears when the project is created:



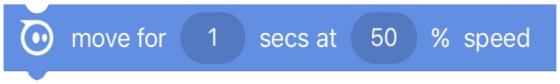
- <span class='blocks'> </span> : This event block will run the following code blocks when the green play button is pressed.



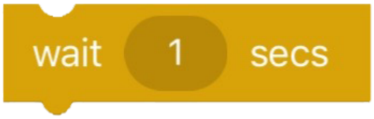
- <span class='blocks'> </span> : This looks block will change the sphero’s color to the selected color.

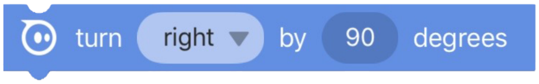


- <span class=’blocks-control’> </span> : Any code inside this control block loop will run a set number of times before moving on and performing other actions.
    - Note: for example, if you set the number to 4, the code inside will consecutively run 4 times, and then move on.



- <span class='blocks'> </span> : This motion block will move the sphero for a set duration at a certain speed.

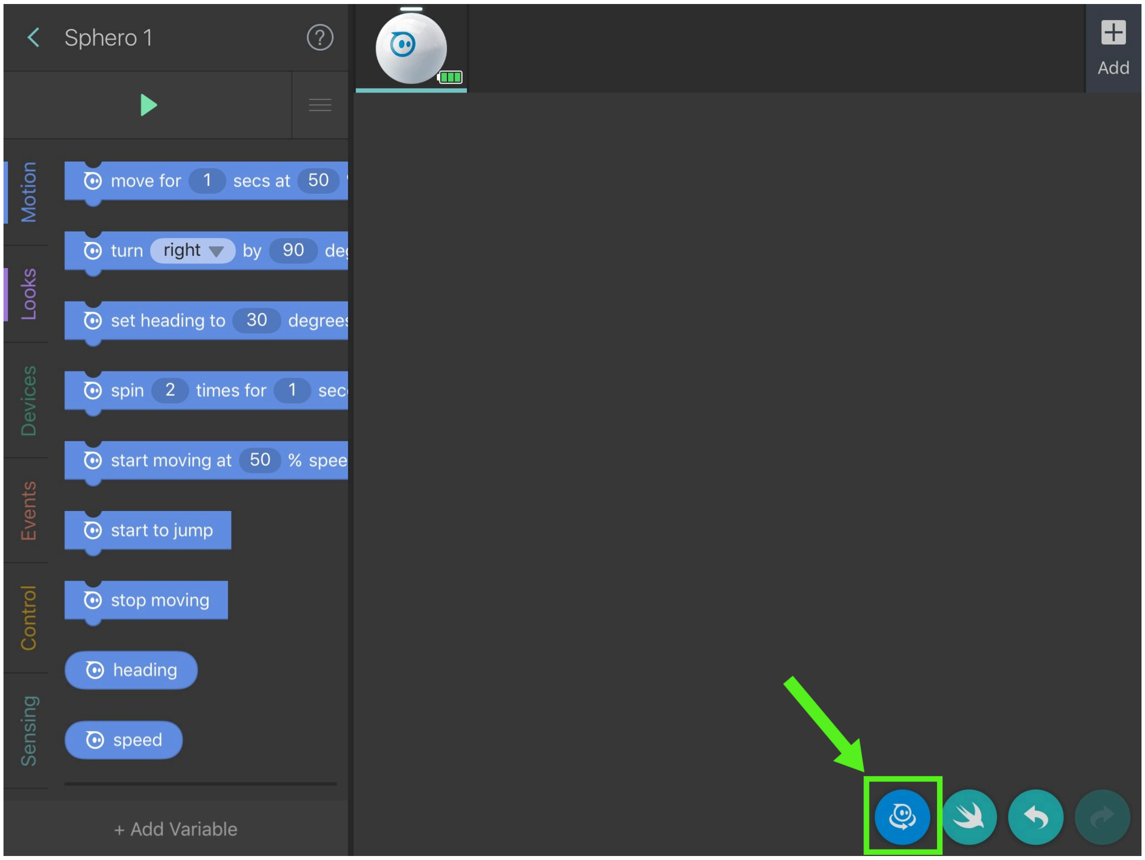
- <span class='blocks-control'> 

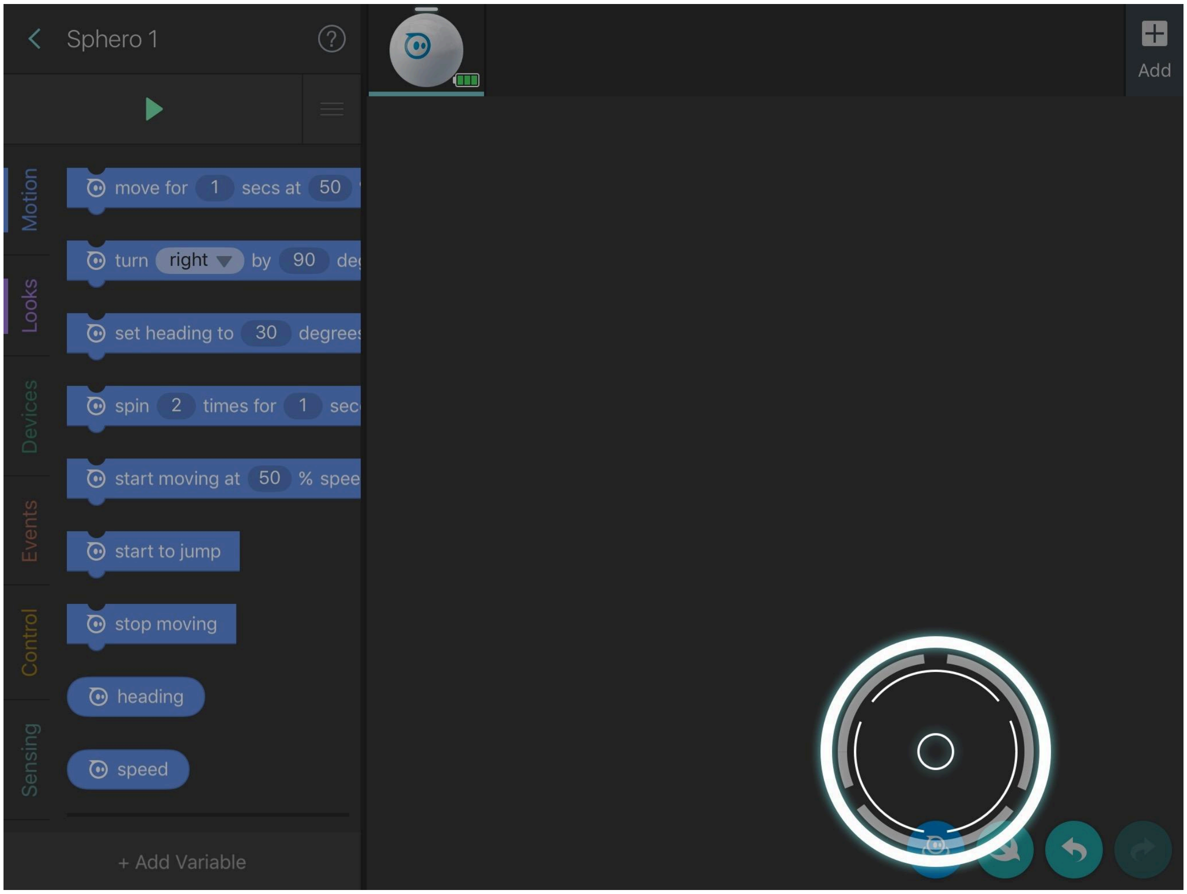
</span> : This control block will pause the current section of code for a set duration.
- <span class='blocks'> 

</span>: This motion block will make the sphero turn left or right by a set angle.

4. Program the Sphero to move in a square pattern

- Before we start on this lesson, remove the template program. Hold and drag the blocks beneath the `` `when starting to play` `` block, and drag it to the delete area that appears.
- First, put the Sphero on the ground and position it by using the calibrate button at the bottom of the screen. Hold the calibrate button and move around the circle to rotate the Sphero.
    - Note: The Sphero faces in the opposite direction of the blue light that appears when the calibrate mode is active (e.g. it faces forward when the blue light is in your direction).

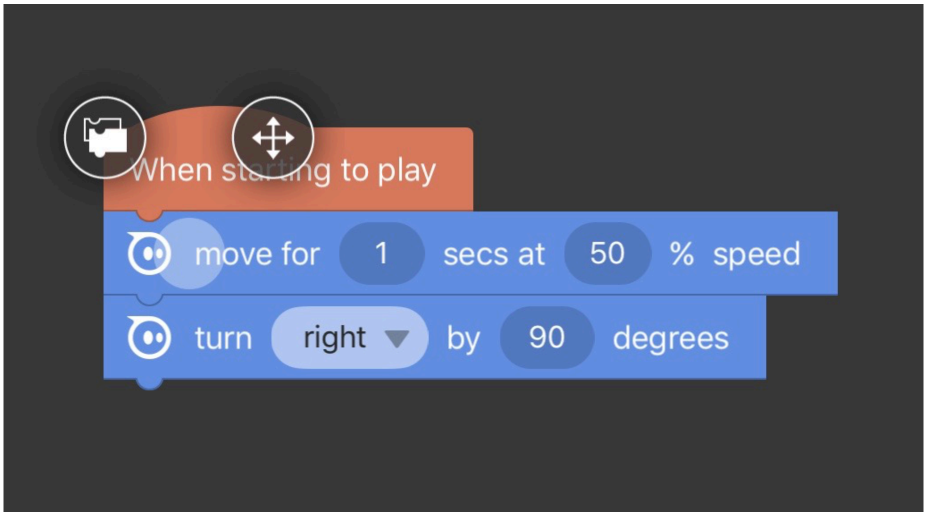

Sphero 1
Add
Motion
Looks
Devices
Events
Control
Sensing
move for 1 secs at 50
turn right by 90 deg
set heading to 30 degrees
spin 2 times for 1 sec
start moving at 50 % spee
start to jump
stop moving
heading
speed
+ Add Variable



- Let's begin by moving the Sphero forward. A `move for 1 secs at 50% speed` block will make the Sphero move for 1 second. You may increase the duration and/or speed for a larger square.
- A full rotation has 360 degrees, so each corner of a square will have 90 degrees. Add a `turn right by 90 degrees` block to turn.

5. Duplicate our code!

- Use the built-in duplicate feature to easily copy a block of code. Since we need to do 4 sequences of moving forwards and then turning by 90 degrees, we can use this feature to quickly copy the code you've written!
- To duplicate, press your finger on the first block that you wish to copy (in this lesson, it will be the `move for 1 secs at 50% speed` block). Hold it until this display appears:

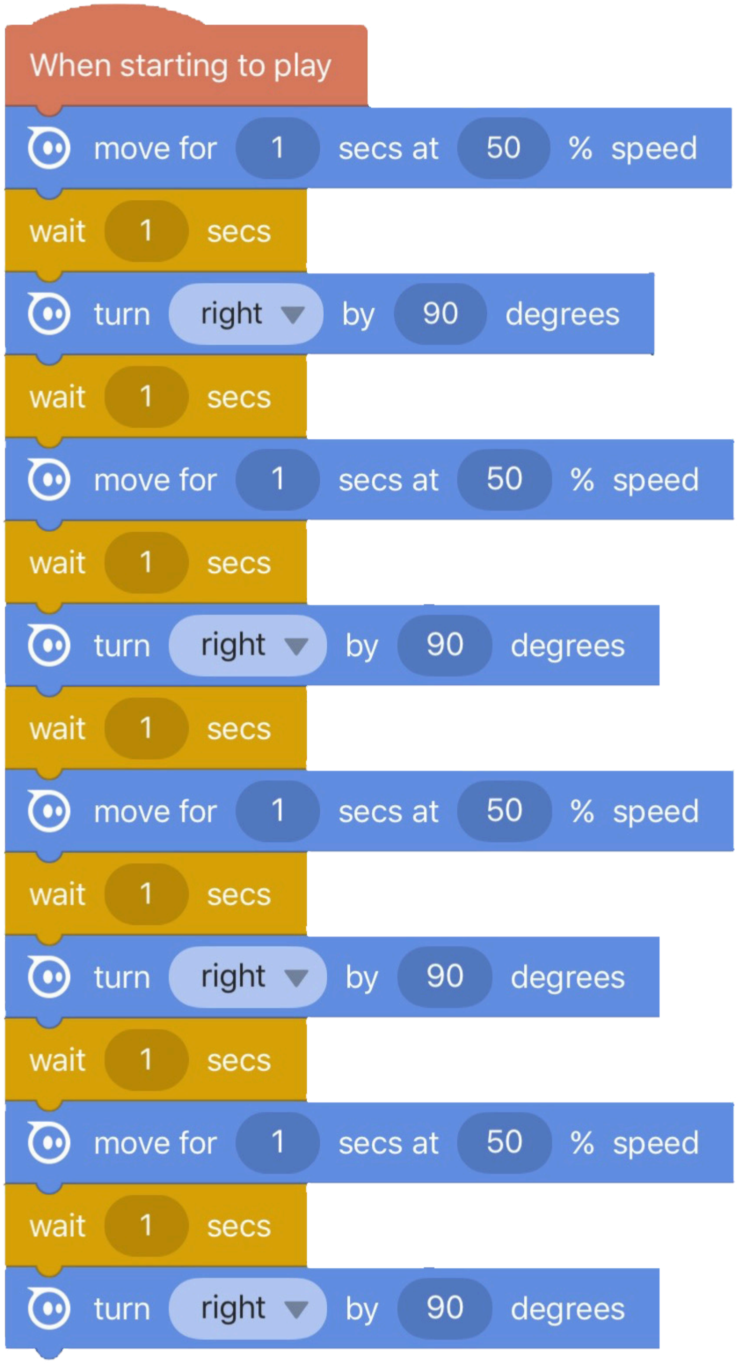


When starting to play

move for 1 secs at 50 % speed

turn right by 90 degrees

- While holding down, drag towards the Duplicate option on the left side. If successful, both the block you pressed and all blocks beneath it get duplicated. Drag it down and place it below the first copy.



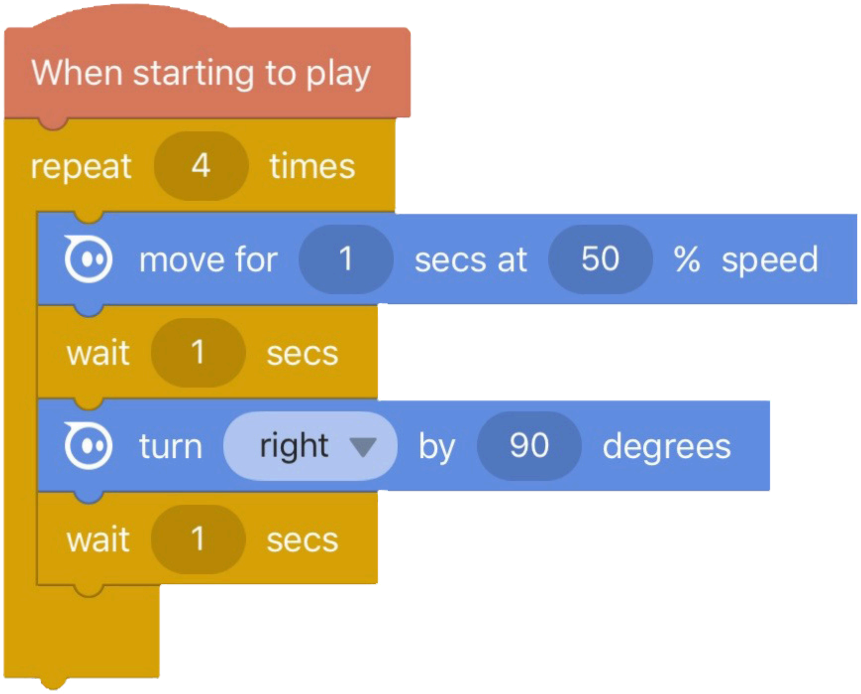
```
When starting to play
move for 1 secs at 50 % speed
wait 1 secs
turn right by 90 degrees
wait 1 secs
move for 1 secs at 50 % speed
wait 1 secs
turn right by 90 degrees
wait 1 secs
move for 1 secs at 50 % speed
wait 1 secs
turn right by 90 degrees
wait 1 secs
move for 1 secs at 50 % speed
wait 1 secs
turn right by 90 degrees
```

  - **Tip***:* If the actions are too fast, add a `` `wait 1 secs` `` block between each command to slow it down.

- The code is complete! Press the green *Play* button to watch the Sphero go!

6. Simplify the code!

- Instead of repeating the same code 4 times, try adding a `` `repeat 4 times` `` block with a single cycle of `` `move for 1 secs at 50% speed` `` and `` `turn right by 90 degrees` `` inside.
    - Note: In programming, a loop is a control statement that allows code to be executed repeatedly. In Tickle, a block whose label begins with “forever” or “repeat” is a looping construct.



```
When starting to play
repeat 4 times
    move for 1 secs at 50 % speed
    wait 1 secs
    turn right by 90 degrees
    wait 1 secs
```

7. Challenge 1

- Make the Sphero move in a pentagon (5 sides) and a hexagon (6 sides).
    - **Tip**: Calculate the degrees to turn each time by dividing 360° by the number of the sides of the geometric shape.

Lesson 2: Rolling, Spinning, and Color Changing

Overview

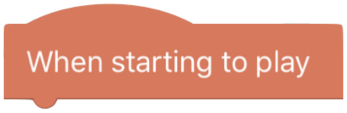
- Program a Sphero to spin, and then move in a triangle pattern.

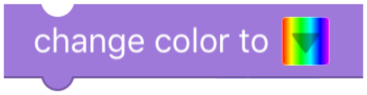
Objectives

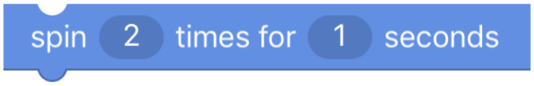
- Become familiar with visual programming.
- Program a Sphero by combining motion and turning blocks to properly move in a triangle pattern.
- Use repeat loops to move in the same pattern with fewer code blocks.

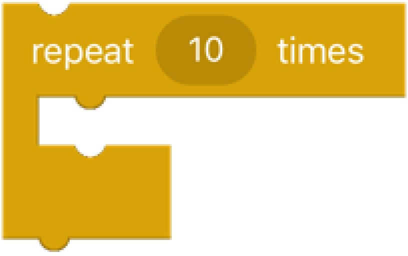
Steps

1. Let’s review the following coding blocks first.

- <span class='blocks'>  </span>: the blocks attached to this block start to execute after the `` `Play` `` button is tapped.

- <span class='blocks'>  </span>: Changes the color of the Sphero.

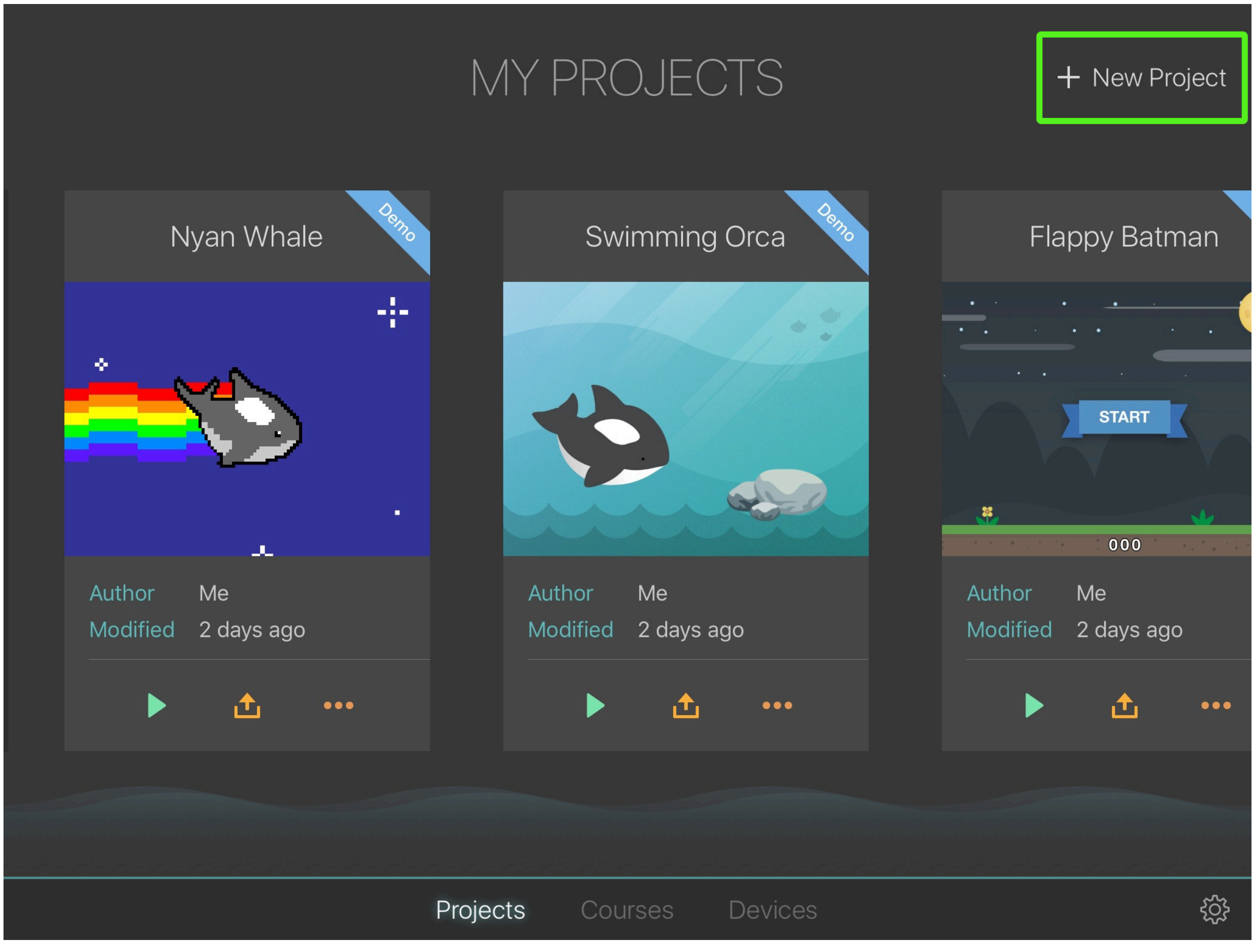
- <span class='blocks'>  </span>: The Sphero spins a certain number of times within a set duration.

- <span class='blocks-control'>  </span>: the blocks inside this block will be run repeatedly for a set number of times.

    - ○ **Tip**: This block works just like a for loop in other programming languages.

2. Create a Sphero Project

- Create a new Sphero project by tapping `+ new Project` via “MY PROJECTS”. Choose the template for “Sphero”.



MY PROJECTS

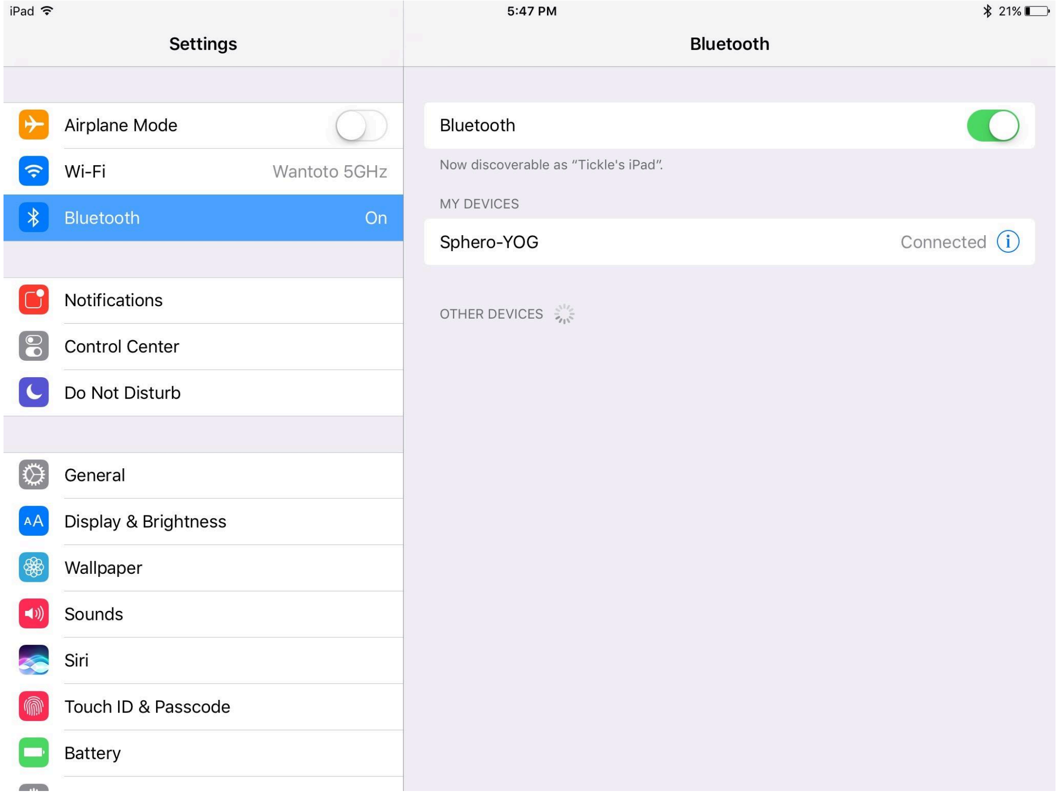
Nyan Whale
Author Me
Modified 2 days ago

Swimming Orca
Author Me
Modified 2 days ago

Flappy Batman
Author Me
Modified 2 days ago

3. Connect to the Sphero

- Activate Bluetooth on your iPhone or iPad, and select your Sphero.



Wi-Fi Wantoto 5GHz

Bluetooth On

Bluetooth

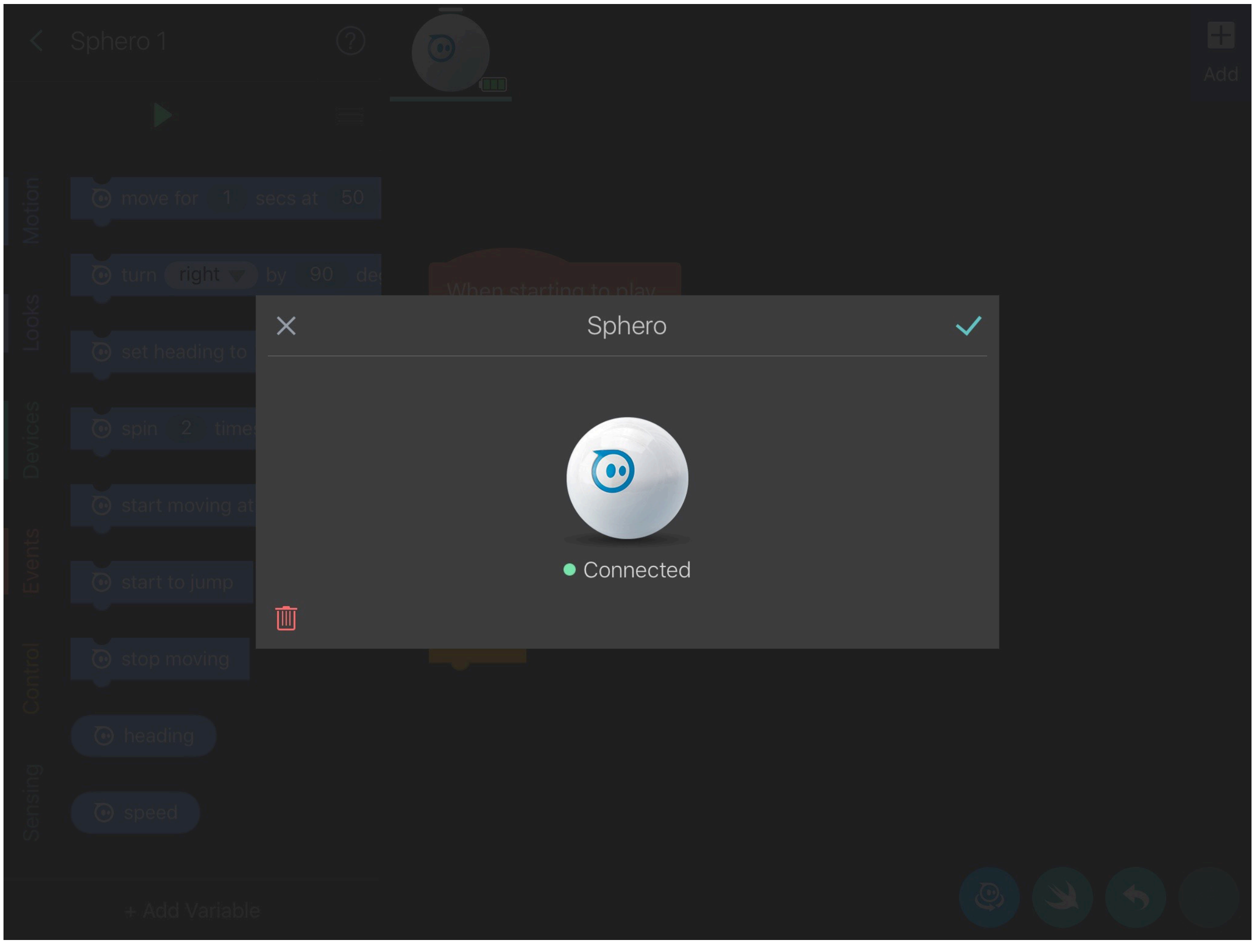
Now discoverable as “Tickle's iPad”.

MY DEVICES

Sphero-YOG Connected

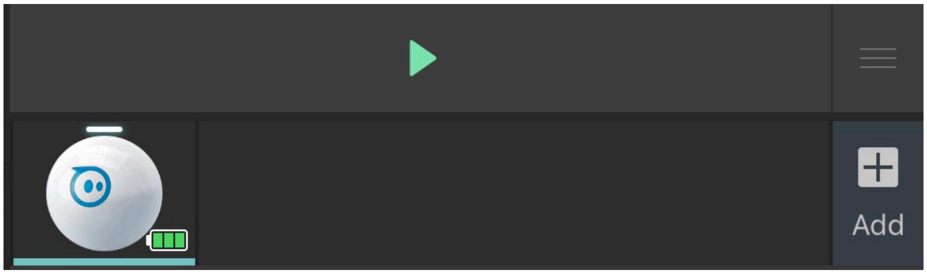
OTHER DEVICES

- Your Sphero should automatically be detected and shown in the “Available Devices” menu in the Tickle App.



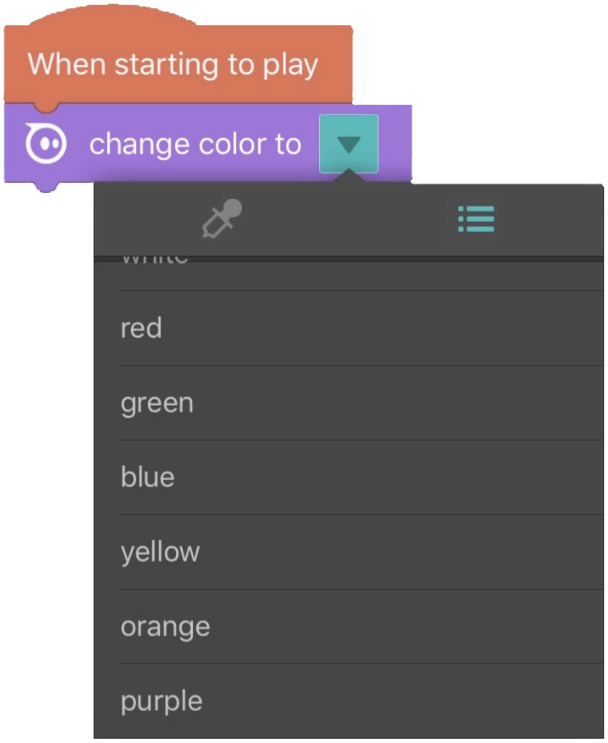
- **Tip*:*** If your Sphero is powered on but not automatically connected, tap the Sphero icon to show troubleshooting instructions.

- The Sphero will show a battery status icon if it's successfully connected. Please charge the Sphero whenever the battery status becomes red because the Sphero may stop responding to your commands.
  - Note: If the Sphero does not connect to Tickle, it may be due to the red battery status.

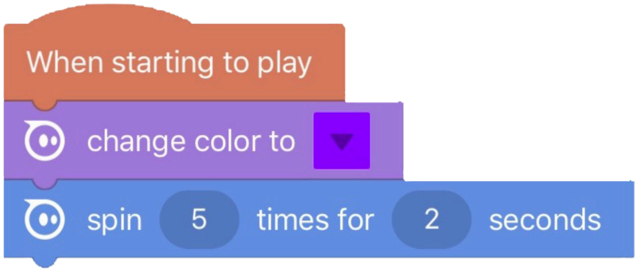


4. Program the Sphero to change color and spin

- To change the color of the Sphero, we add a `change color to ...` block. To change the color, press the color square. Let's set it to purple. To do so, press the preset colors button, and then press the purple button at the bottom of the list.



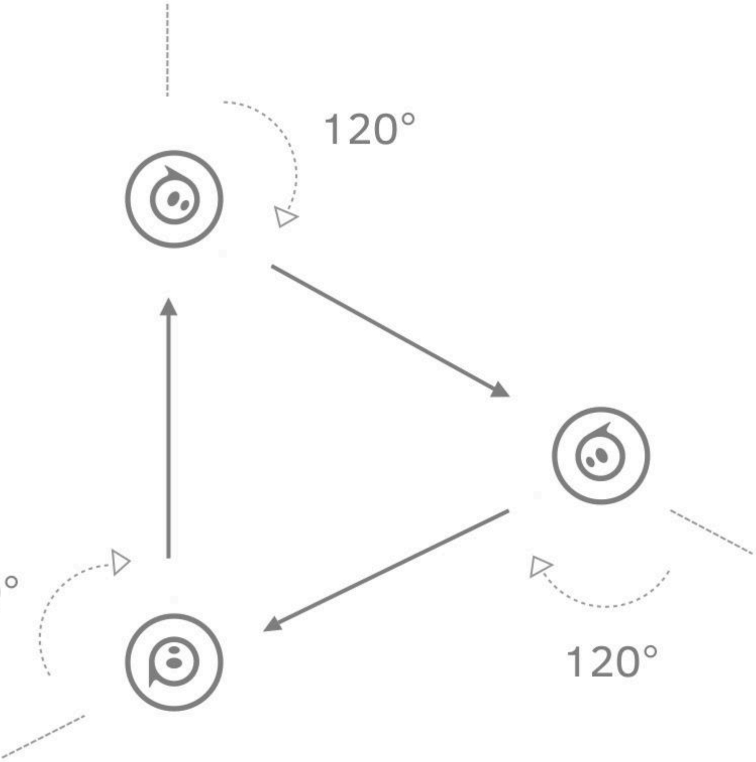
- To make the Sphero spin, we add a `spin 2 times for 1 seconds` block. Let's make the Sphero spin 5 times for 2 seconds. To do this, change the number of times to 5, and duration to 2.



When starting to play
change color to
spin 5 times for 2 seconds

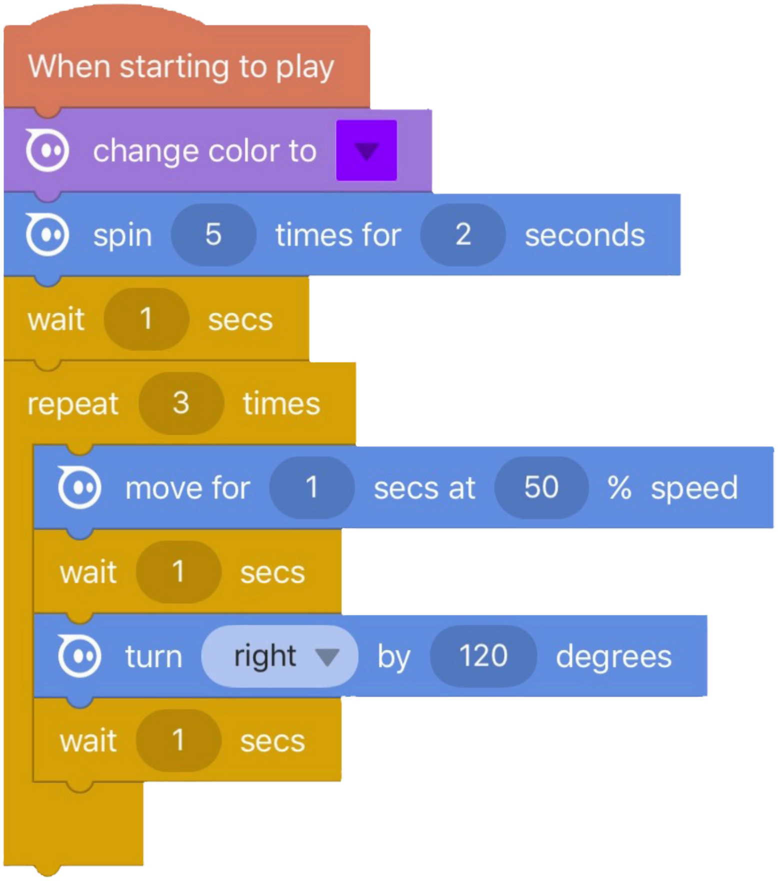
5. Program the Sphero to move in a triangle

- A triangle has 3 sides. A `repeat 3 times` block will be useful in simplifying the code. To roll in a triangle, the Sphero has to turn 3 times to complete a full 360° turn. The number of degrees to turn each time is then **360°** divided by **3** = **120°**.



- Now, let’s have the Sphero move in a triangle pattern. Just like in the first lesson, use a loop to simplify the code. We should loop 3 times, because a triangle has 3 sides.
    - Note: The Sphero should return to its original position after successfully executing all commands.

- The code should look like the following:



```
When starting to play
change color to ▼
spin 5 times for 2 seconds
wait 1 secs
repeat 3 times
    move for 1 secs at 50 % speed
    wait 1 secs
    turn right ▼ by 120 degrees
    wait 1 secs
```

- **Tip***:* If the actions are too fast to recognize, add a `` `wait 1 secs` `` block after each command. A `` `wait 1 secs` `` block will pause that block of code for 1 second. You may increase the pause by increasing the duration.

- Press the green *Play* button to watch the Sphero go!

Lesson 3: Rolling with Collisions and Shaking

Overview

- Program a Sphero to perform actions when it collides with objects and when the device is shaken.

Objectives

- Use new Event blocks in Tickle.
- Understand how each Event block creates unique interactions.
- Learn about event-driven programming concepts.

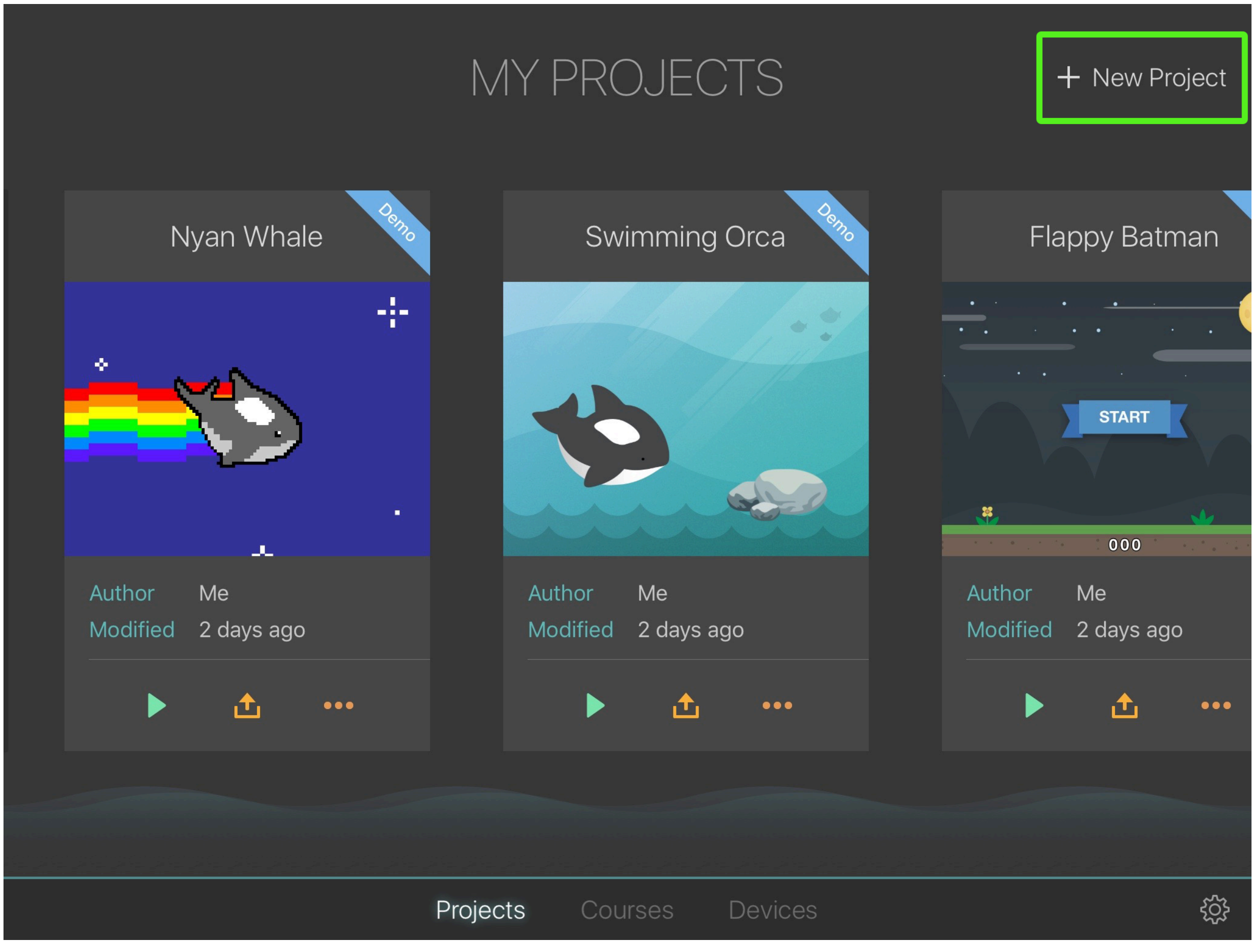
Event-driven Programming and Event Blocks

- Event-driven programming is a computer science concept in which the flow of a program is determined by events such as sensor input, user actions (like touching the screen or tilting the device), and messages from other programs.
- It is the dominant paradigm generally used in modern applications like graphical user interfaces (GUI) and web-pages that perform certain actions in response to user input.
- A simple way to think of it is "when something happens, do something." For example, “when it rains, open your umbrella.”

Steps

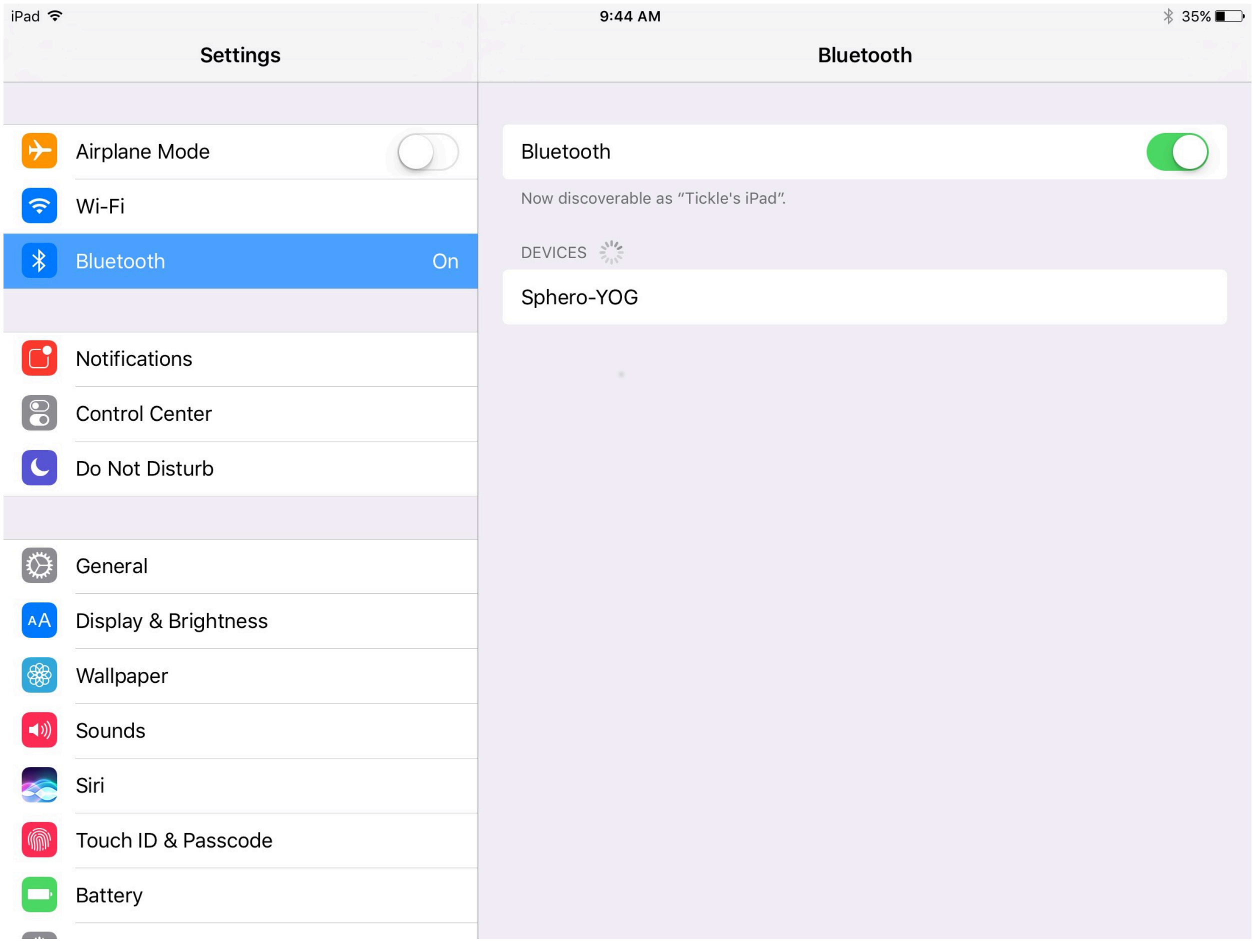
1. Create a Sphero Project

- Create a new Sphero project by tapping `` `+ new Project` `` via “MY PROJECTS.” Choose the template for “Sphero.”



2. Connect to the Sphero

- Activate Bluetooth on your iPhone or iPad, and select your Sphero.



iPad 9:44 AM 35%

Settings

- Airplane Mode ○
- Wi-Fi
- Bluetooth On
- Notifications
- Control Center
- Do Not Disturb
- General
- Display & Brightness
- Wallpaper
- Sounds
- Siri
- Touch ID & Passcode
- Battery

Bluetooth

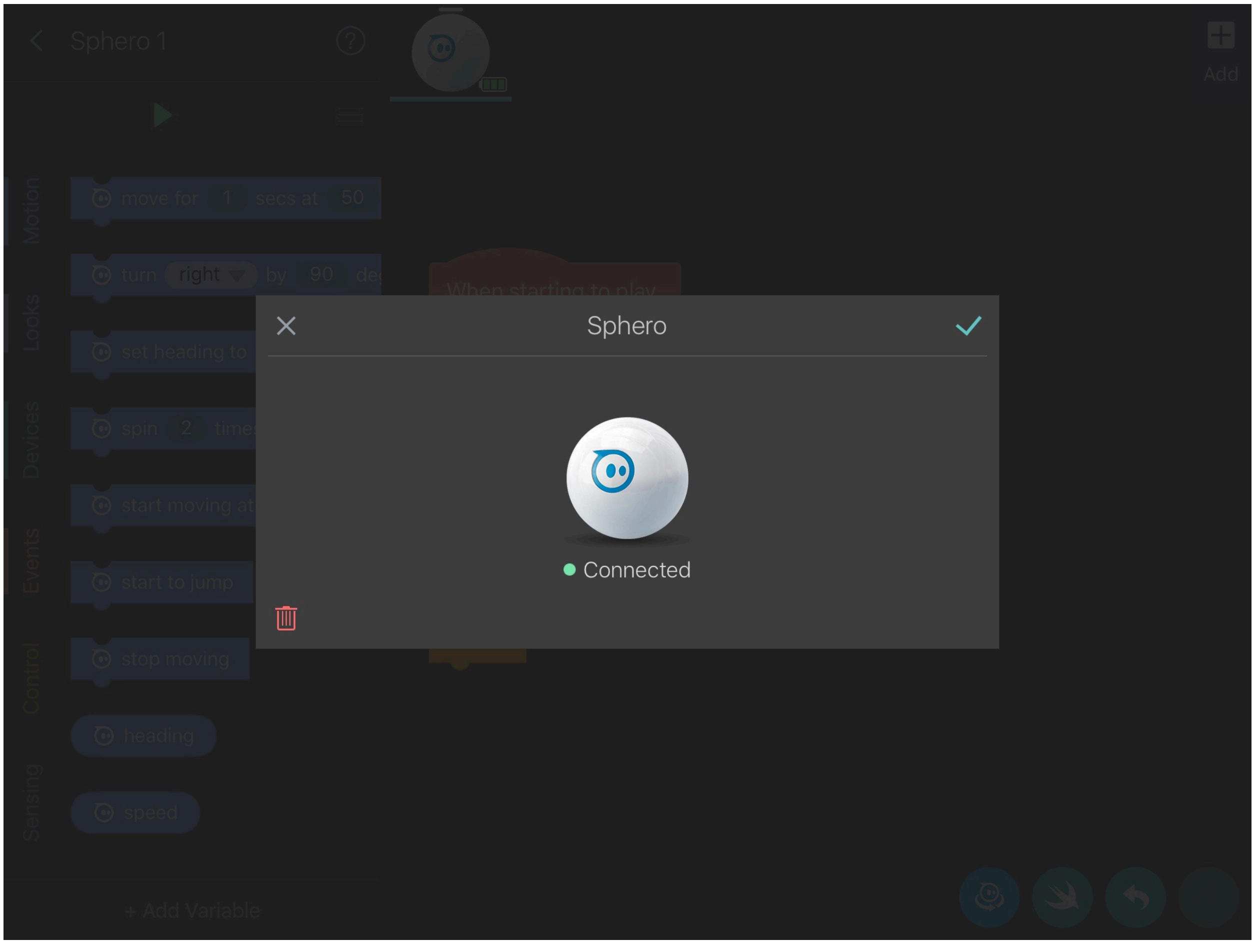
Bluetooth ●

Now discoverable as "Tickle's iPad".

DEVICES

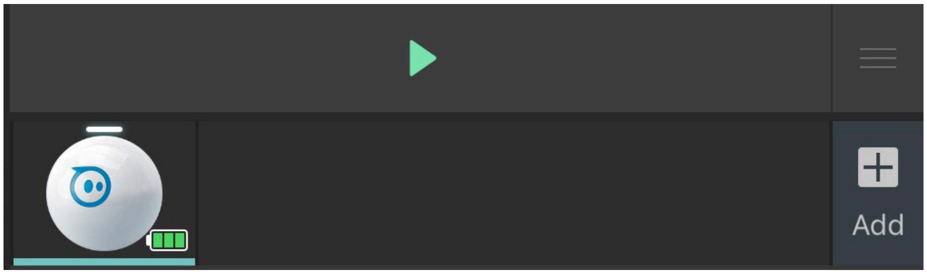
Sphero-YOG

- Your Sphero should automatically be detected and shown in the "Available Devices" menu in the Tickle App.



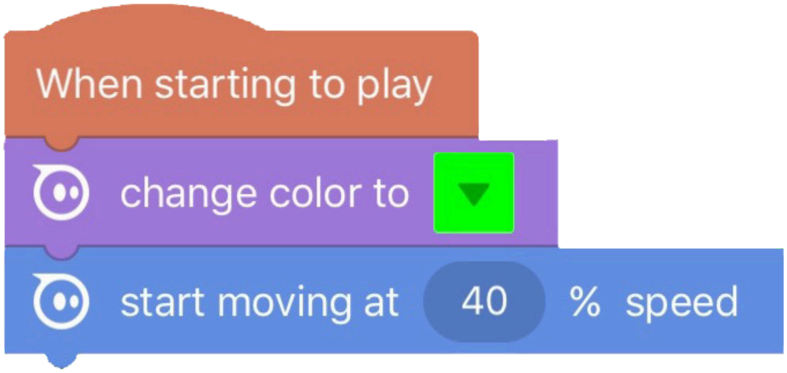
  - **Tip*:*** If your Sphero is powered on but not automatically connected, tap the Sphero icon to show troubleshooting instructions.

- The Sphero will show a battery status icon if it's successfully connected. Please charge the Sphero whenever the battery status becomes red, because the Sphero may stop responding to your commands.
  - Note: If the Sphero is not connected to Tickle, it may be due to the red battery status.



3. Moving forward

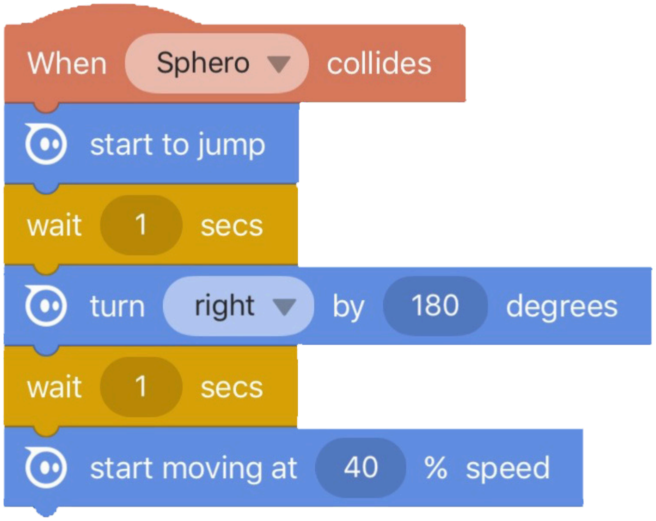
- Let’s begin by changing to a new color. For this project we will choose green.
- Let’s give the Sphero a starting motion as well. At this point the code should look like the following:



```
When starting to play
change color to 🟩
start moving at 40 % speed
```

4. Colliding with other objects

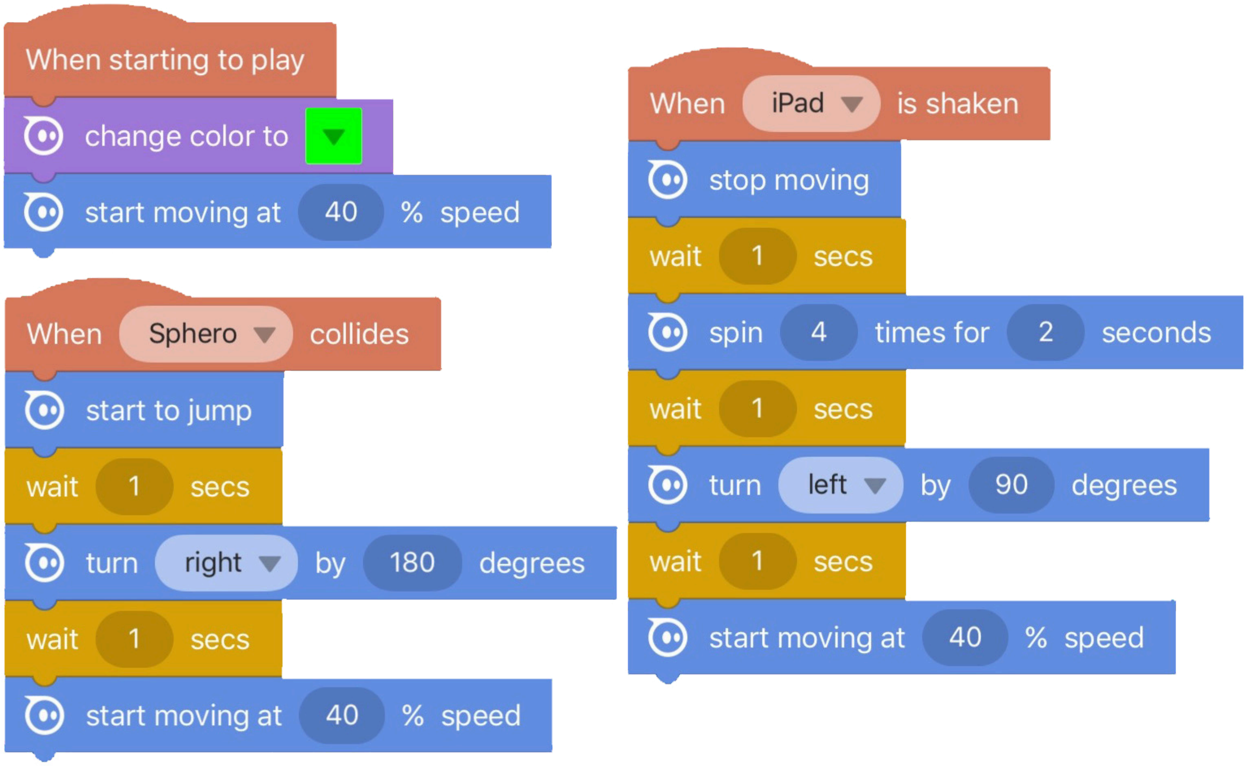
- Try putting an object in front of the Sphero. This next block of code will generate interactions when the Sphero collides with the object. Start by selecting the `events` category and dragging out a `When Sphero collides` block.
- The `start to jump` block will perform a jumping action until the Sphero is given another command. Let’s program the Sphero to jump when it first collides, and then turn around and continue rolling.



```
When Sphero ▾ collides
start to jump
wait 1 secs
turn right ▾ by 180 degrees
wait 1 secs
start moving at 40 % speed
```

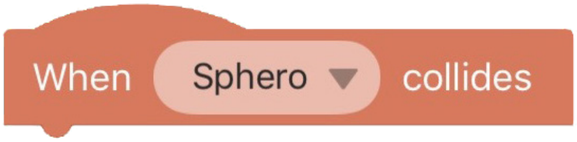
5. Let’s shake things up!

- Let’s add some more interactions between the user and Sphero. The `When ... is shaken` block will perform actions when something is shaken. Select the Ipad or Iphone to make the Sphero perform actions when you shake your device.
- To make things interesting, let’s program the Sphero to spin 4 times for 2 seconds, and then turn left by 90 degrees and continue rolling.
    - Note: A `stop moving` block can be used to stop the Sphero. Here, the Sphero will stop rolling first, and then begin its jumping. It is useful because if it is **not** placed, the Sphero will keep rolling while spinning.

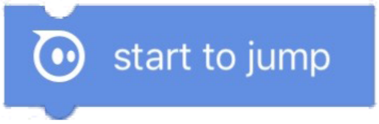


```
When starting to play
  change color to 🟩
  start moving at 40 % speed

When Sphero collides
  start to jump
  wait 1 secs
  turn right by 180 degrees
  wait 1 secs
  start moving at 40 % speed

When iPad is shaken
  stop moving
  wait 1 secs
  spin 4 times for 2 seconds
  wait 1 secs
  turn left by 90 degrees
  wait 1 secs
  start moving at 40 % speed
```

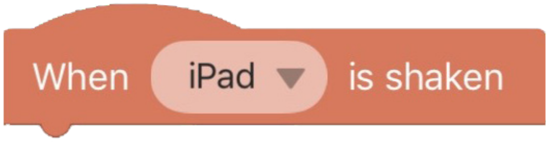
6. Let's review the new blocks used today.


When Sphero ▼ collides

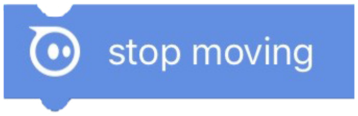
- <span class='blocks'> </span> : When the Sphero collides with an object, it will perform the following blocks of code.


start to jump

- <span class='blocks'> </span> : The Sphero starts to jump until another command is given.


When iPad ▼ is shaken

- <span class='blocks'> </span>: When the iPad (or your device) is shaken, the Sphero performs the following blocks of code.

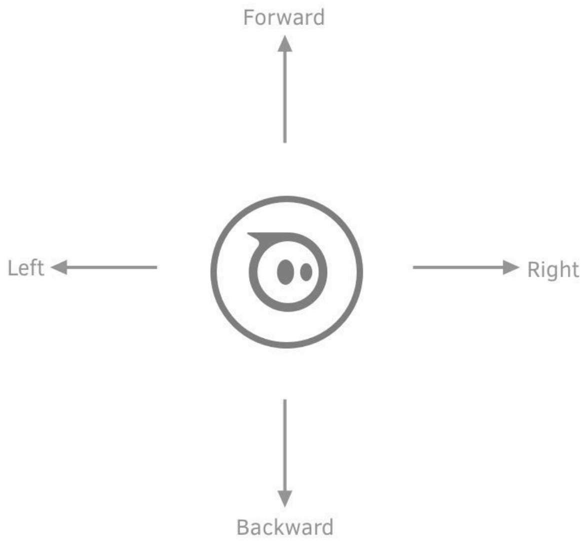

stop moving

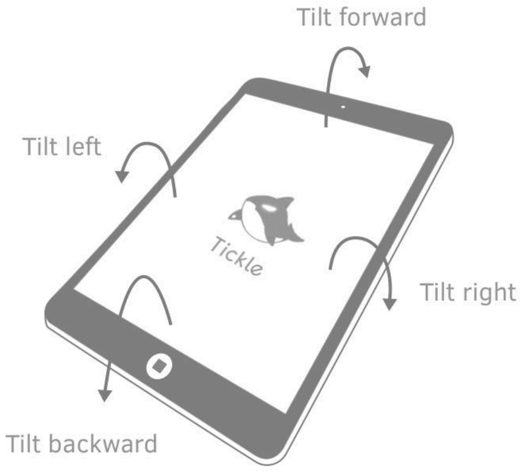
- <span class='blocks'> </span>: The Sphero will stop all its movements until another command is given.

Lesson 4: Steer the Sphero using Motion Sensors

Overview

- Utilize the tilting sensors of your phones and tablets to pilot the Sphero. For example, tilting your iPad backwards will make the Sphero move towards you, and shaking the iPad will make the Sphero jump!





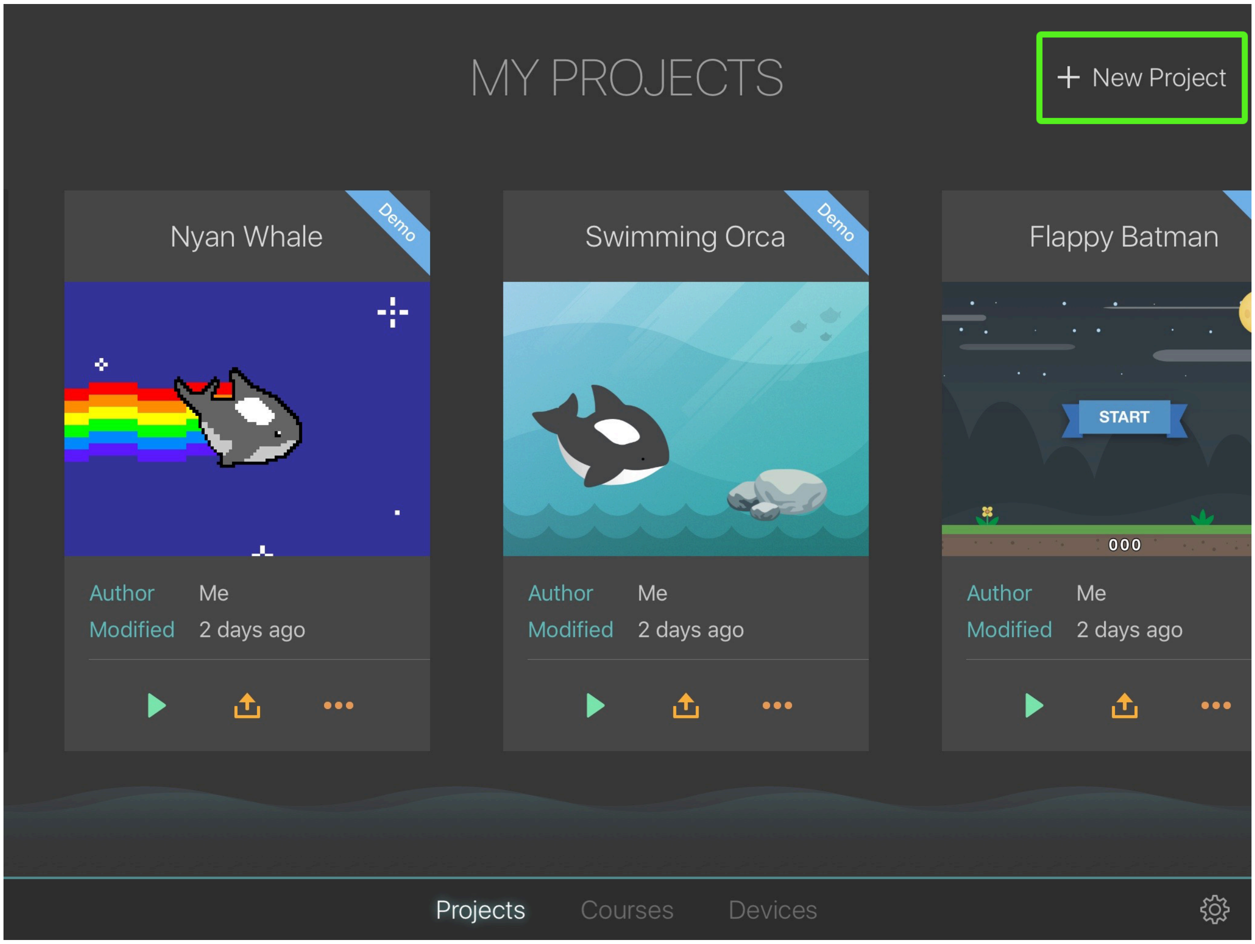
Objectives

- Use motion sensors in smartphones and tablets to sense motion events (tilt and shake) to pilot the Sphero.
- Further understand the programming concept of “event-driven programming.”

Steps

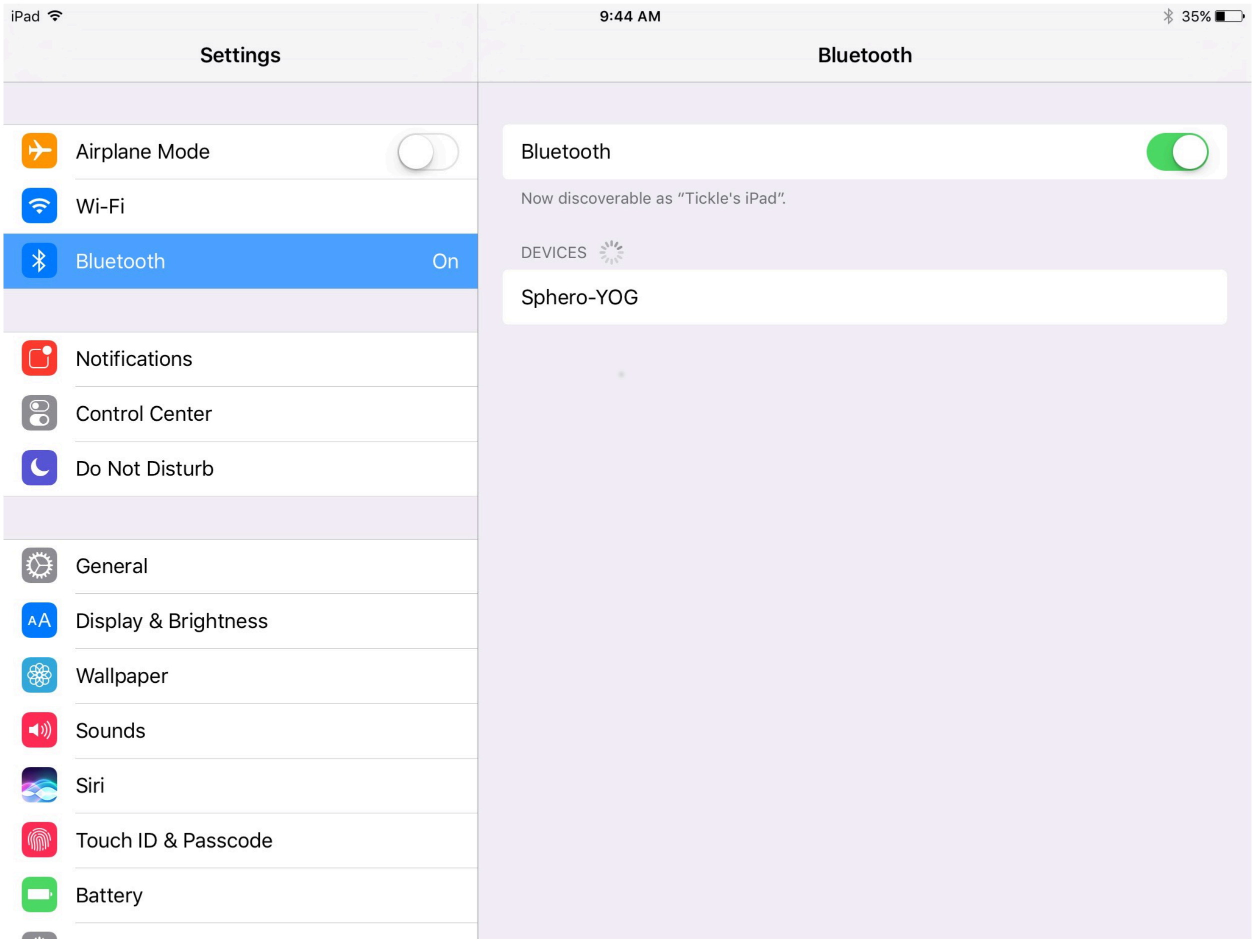
1. Create a Sphero Project

- Create a new Sphero project by tapping `` `+ new Project` `` via “MY PROJECTS.” Choose the template for “Sphero.”



2. Connect to the Sphero

- Activate Bluetooth on your iPhone or iPad, and select your Sphero.



Bluetooth

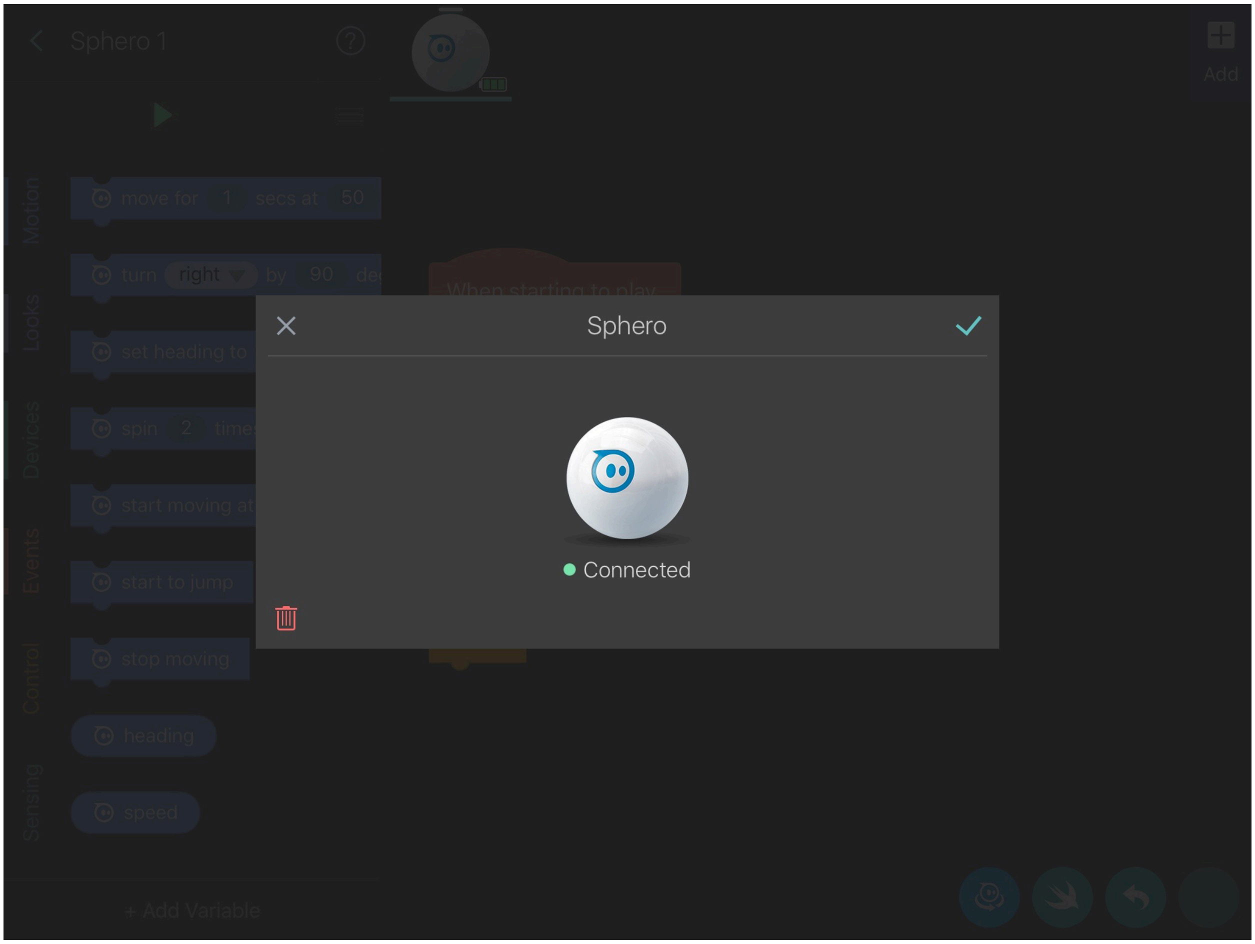
Bluetooth ●

Now discoverable as "Tickle's iPad".

DEVICES

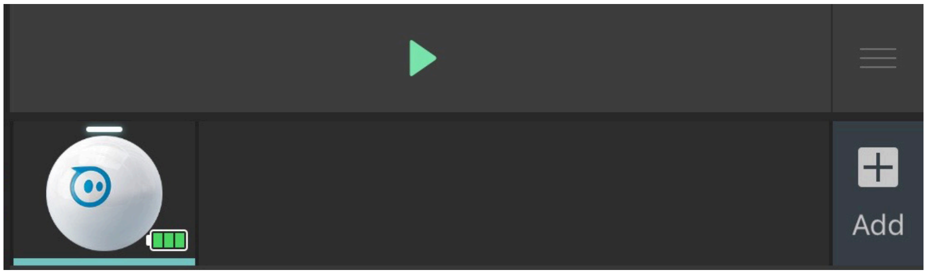
Sphero-YOG

- Your Sphero should automatically be detected and shown in the "Available Devices" menu in the Tickle App.



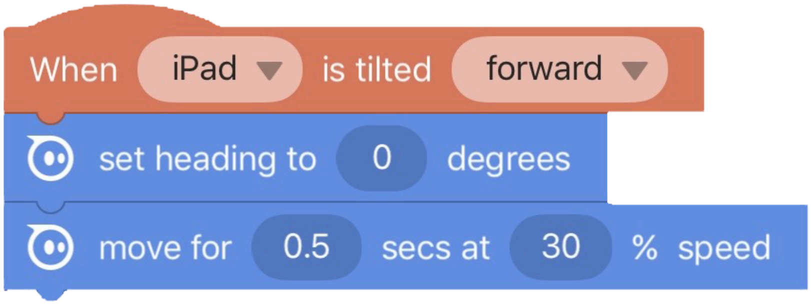
   - **Tip*:*** If your Sphero is powered on but not automatically connected, tap the Sphero icon to show troubleshooting instructions.

- The Sphero will show a battery status icon if it's successfully connected. Please charge the Sphero whenever the battery status becomes red, because the Sphero may stop responding to your commands.
   - Note: If the Sphero is not connected to Tickle, it may be due to the red battery status.



3. Move forward using motion controls

- Under the events category, add the motion sensing event block `` `when iPad is tilted forward` ``.
- Since the forward direction is at 0 degree, add a `` `set heading to 0 degrees` `` block to set the direction to forward. Then add a movement block like the following:

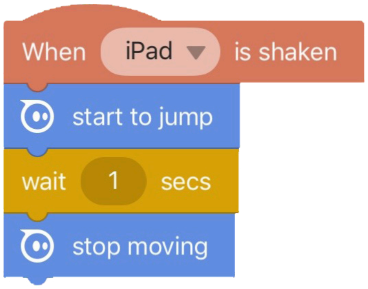


```
When iPad ▼ is tilted forward ▼
set heading to 0 degrees
move for 0.5 secs at 30 % speed
```

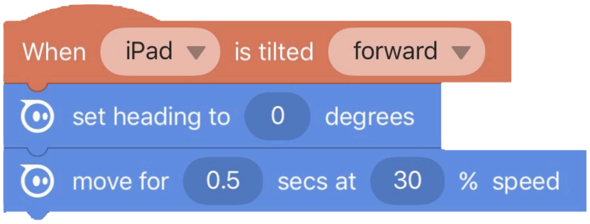
  - **Tip*:*** Setting the duration to 0.5 seconds means that the Sphero will move forwards for half a second, and then check which direction the device is tilted. To increase responsiveness, reduce the duration of each movement.
  - Note: Try setting the duration to be longer and see what happens! Also, try setting it to be very, very short!

4. Use Motion Sensors to move the Sphero in different directions

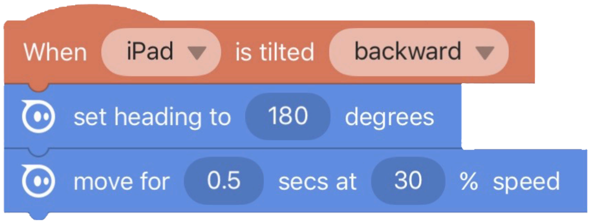
- To pilot the Sphero in all 4 directions, add a `` `when ... is tilted ...` `` event block for each direction. Set each heading to their respective angles.
- Also, add a `` `when ... is shaken` `` event block to make the Sphero jump!



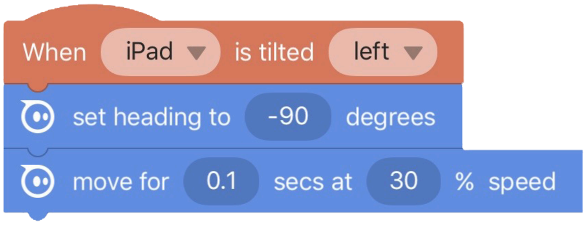
```
When iPad is shaken
  start to jump
  wait 1 secs
  stop moving
```



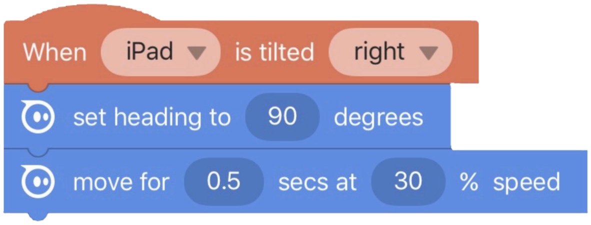
```
When iPad is tilted forward
  set heading to 0 degrees
  move for 0.5 secs at 30 % speed
```



```
When iPad is tilted backward
  set heading to 180 degrees
  move for 0.5 secs at 30 % speed
```

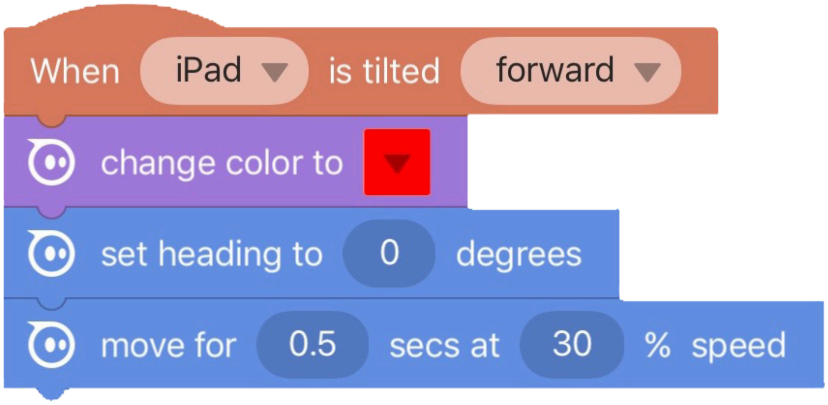


```
When iPad is tilted left
  set heading to -90 degrees
  move for 0.1 secs at 30 % speed
```



```
When iPad is tilted right
  set heading to 90 degrees
  move for 0.5 secs at 30 % speed
```

- To make things more fun, try adding a different color for each direction!



```
When iPad is tilted forward
  change color to 🟥
  set heading to 0 degrees
  move for 0.5 secs at 30 % speed
```