

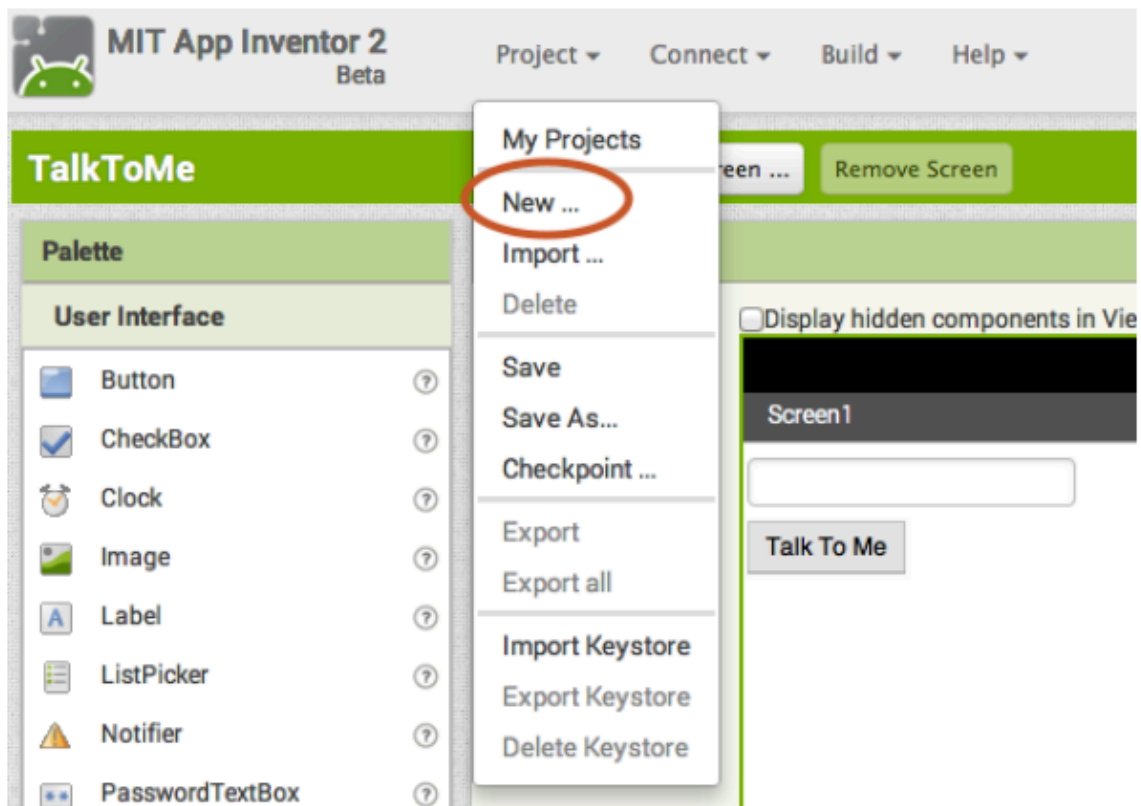


BallBounce: A simple game app

In this tutorial, you will learn about animation in App Inventor by making a Ball (a sprite) bounce around on the screen (on a Canvas).

Start a New Project

If you have another project open, go to My Projects menu and choose New Project.



BallBounce: A simple game app

In this tutorial, you will learn about animation in App Inventor by making a Ball (a sprite) bounce around on the screen (on a Canvas).

Start a New Project

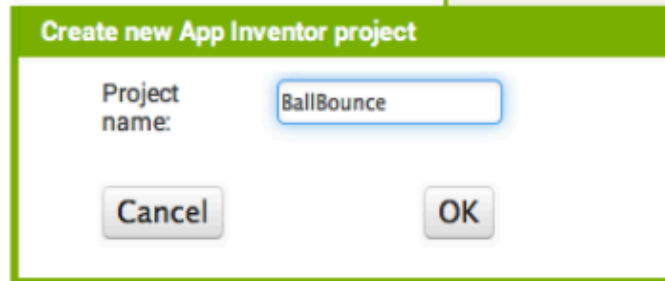
If you have another project open, go to My Projects menu and choose New Project.

BallBounce: A simple game app - 1



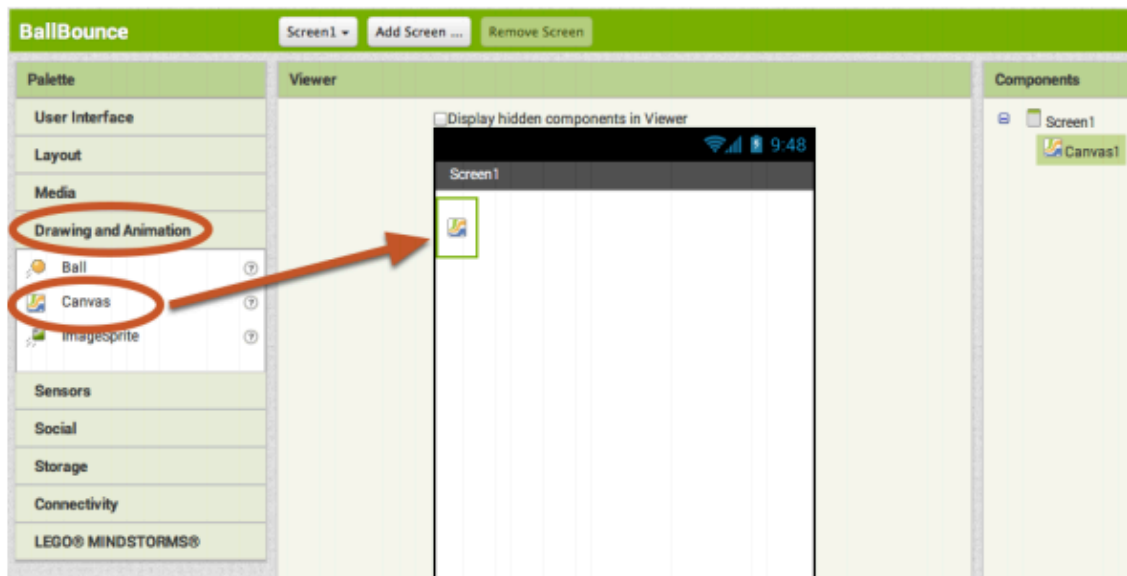
Name the Project

Call it something like "BallBounce". Remember, no spaces. But underscores are OK.



Add a Canvas

From the **Drawing and Animation drawer**, drag out a **Canvas component** and drop it onto the viewer.



BallBounce: A simple game app - 2

Name the Project

Call it something like "BallBounce". Remember, no spaces. But underscores are OK.

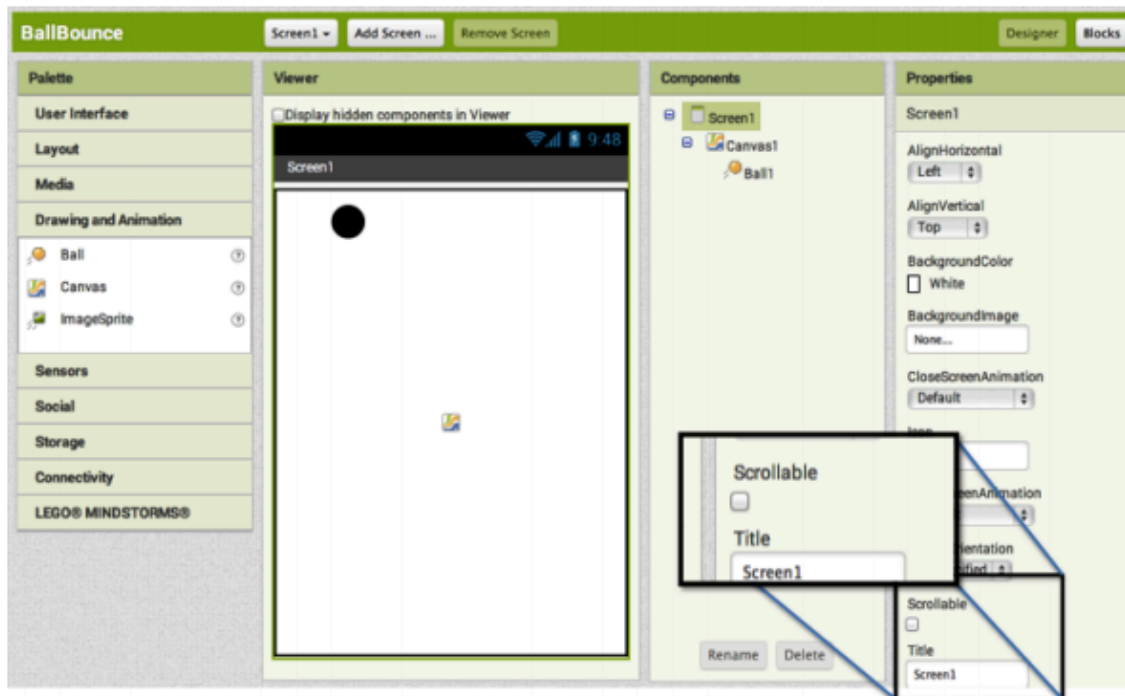
Add a Canvas

From the Drawing and Animation drawer, drag out a Canvas component and drop it onto the viewer.

BallBounce: A simple game app - 2

Set the Screen so that it does not scroll

The default setting for App Inventor is that the screen of your app will be "scrollable", which means that the user interface can go beyond the limit of the screen and the user can scroll down by swiping their finger (like scrolling on a web page). When you are using a Canvas, you have to **turn off the "Scrollable" setting (UNCHECK THE BOX)** so that the screen does not scroll. This will allow you to make the Canvas to fill up the whole screen.

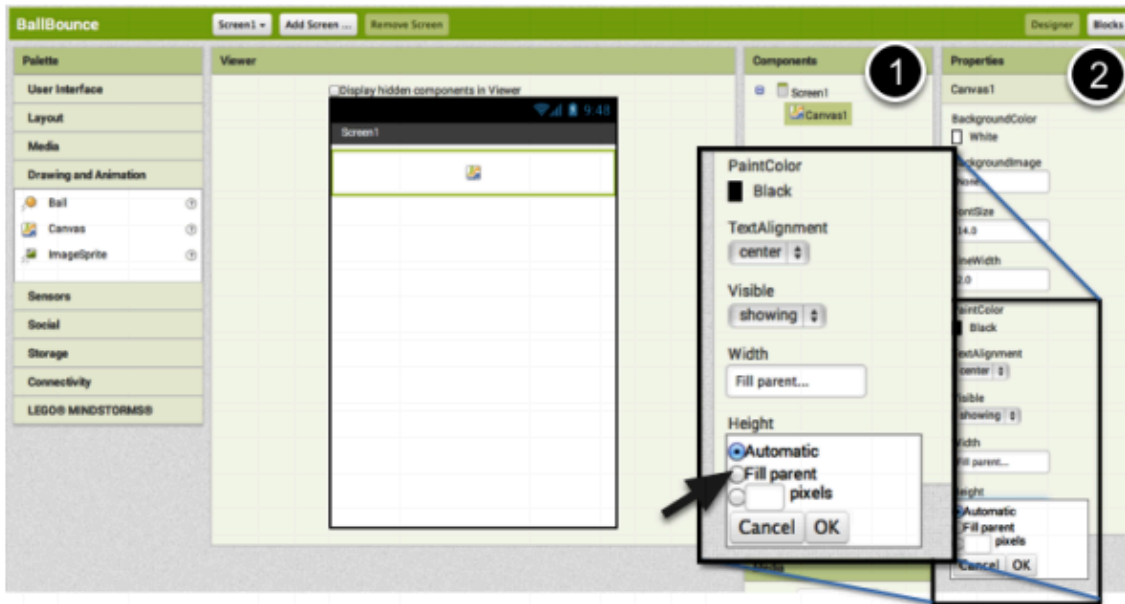


The default setting for App Inventor is that the screen of your app will be "scrollable", which means that the user interface can go beyond the limit of the screen and the user can scroll down by swiping their finger (like scrolling on a web page). When you are using a Canvas, you have to turn off the "Scrollable" setting (UNCHECK THE BOX) so that the screen does not scroll. This will allow you to make the Canvas to fill up the whole screen.

BallBounce: A simple game app - 3

Change the Height and Width of the Canvas to Fill Parent

Make sure the Canvas component is selected (#1) so that its properties show up in the Properties Pane (#2). Down at the bottom, **set the Height property to "Fill Parent"**. Do the **same with the Width property**.

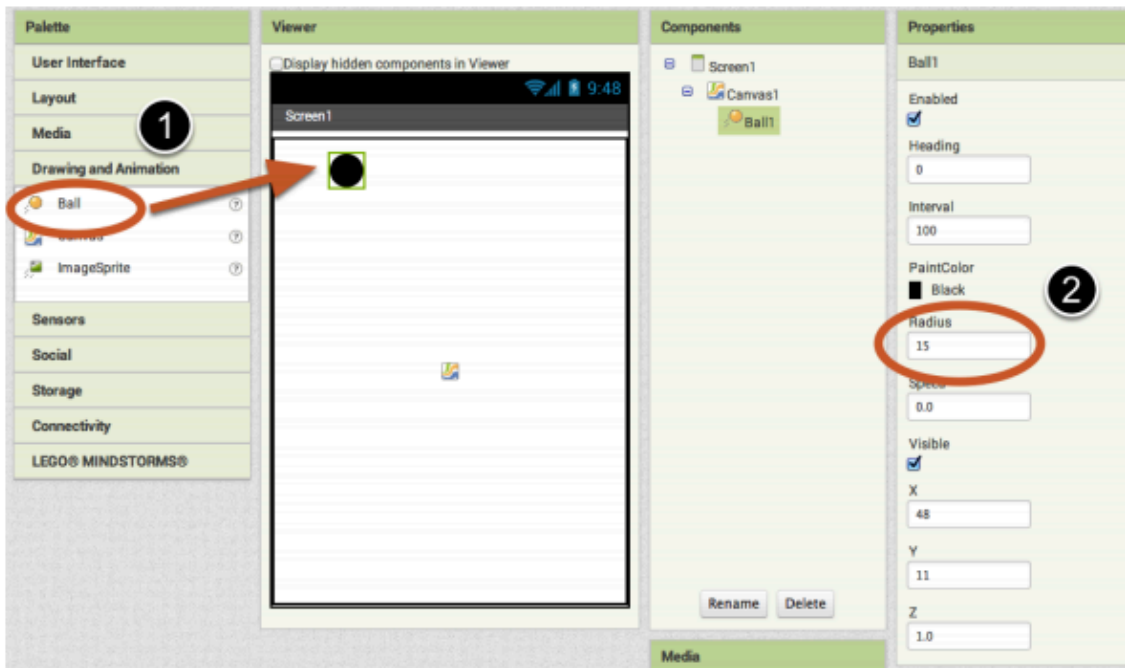


Make sure the Canvas component is selected (#1) so that its properties show up in the Properties Pane (#2). Down at the bottom, set the Height property to "Fill Parent". Do the same with the Width property.

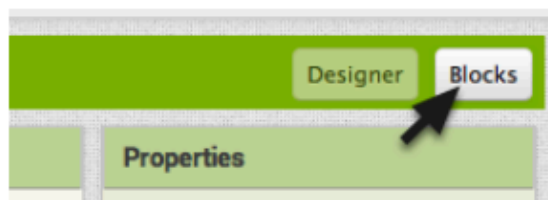
BallBounce: A simple game app - 4

Add a Ball

Now that we have a Canvas in place, we can add a Ball Sprite. This can also be found in the **Drawing and Animation drawer**. Drag out a **Ball component** and drop it onto the Canvas (#1). If you'd like the ball to show up better, you can change its **Radius property** in the Properties pane (#2).



Open the Blocks Editor.

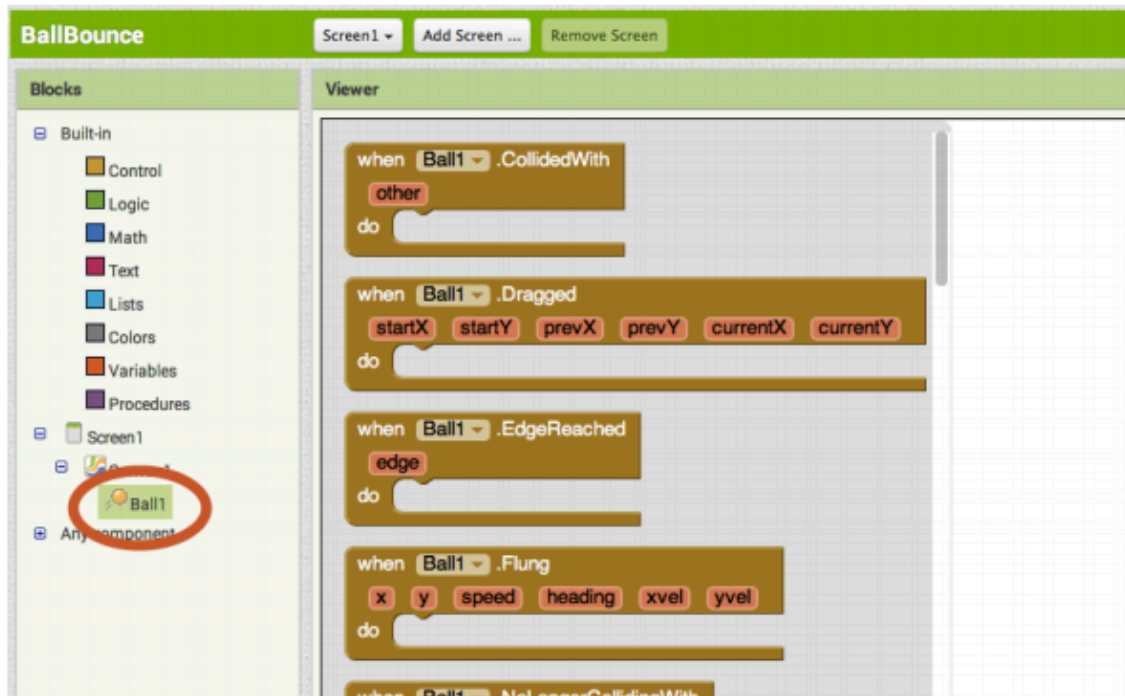


Now that we have a Canvas in place, we can add a Ball Sprite. This can also be found in the Drawing and Animation drawer. Drag out a Ball component and drop it onto the Canvas (#1). If you'd like the ball to show up better, you can change its Radius property in the Properties pane (#2).

BallBounce: A simple game app - 5

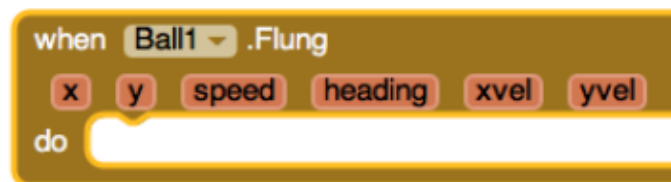
Open the Blocks Editor.

Open the Ball1 Drawer to view the Ball's blocks.



Drag out the Flung Event Handler

Choose the block **when Ball1.Flung** and drag-and-drop it onto the workspace. Flung refers to the user making a "Fling gesture" with his/her finger to "fling" the ball. Fling is a gesture like what a golf club does, not like how you launch Angry Birds! In App Inventor, the event handler for that type of gesture is called *when Flung*.



Drag out the Flung Event Handler

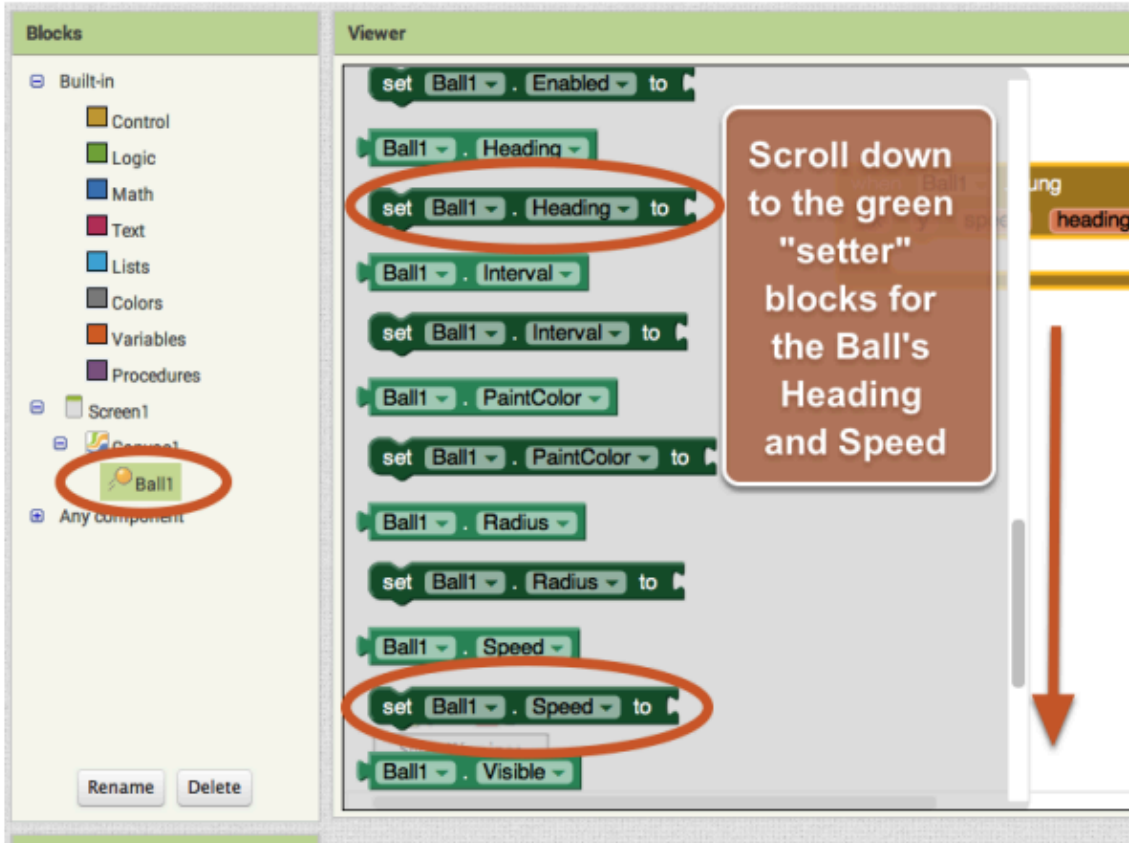
Choose the block when Ball1.Flung and drag-and-drop it onto the workspace. Flung refers to the user making a "Fling gesture" with his/her finger to "fling" the ball. Fling is a gesture like what a golf club does, not like how you launch Angry Birds! In App Inventor, the event handler for that type of gesture is called when Flung.

BallBounce: A simple game app - 6

Open the Ball1 Drawer to view the Ball's blocks.

Set the Ball's Heading and Speed. First get the setter blocks.

Open the Ball drawer and scroll down in the list of blocks to get the **set Ball1.Heading** and **set Ball1.Speed** blocks



Scroll down to the green "setter" blocks for the Ball's Heading and Speed

Set the Ball's Heading and Speed. First get the setter blocks.

Open the Ball drawer and scroll down in the list of blocks to get the set Ball1.Heading and set Ball1.Speed blocks

BallBounce: A simple game app - 7



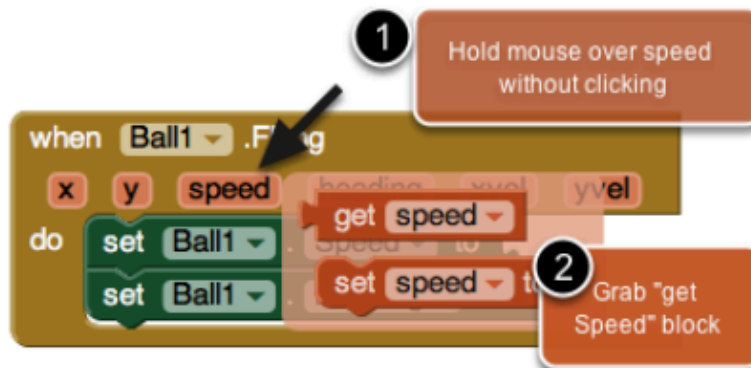
MIT App Inventor
appinventor.mit.edu

Plug the set Ball1.Speed and set Ball1.Heading into the Fling event handler



Set the Ball's speed to be the same as the Fling gesture's speed

Mouse over the "speed" parameter of the **when Ball1.Fling** event handler. The get and set blocks for the speed of the fling will pop up. Grab the **get speed** block and plug that into the **set Ball1.Speed** block.



Set the Ball's speed to be the same as the Fling gesture's speed

Mouse over the "speed" parameter of the when Ball1.Flung event handler. The get and set blocks for the speed of the fling will pop up. Grab the get speed block and plug that into the set Ball1.Speed block.

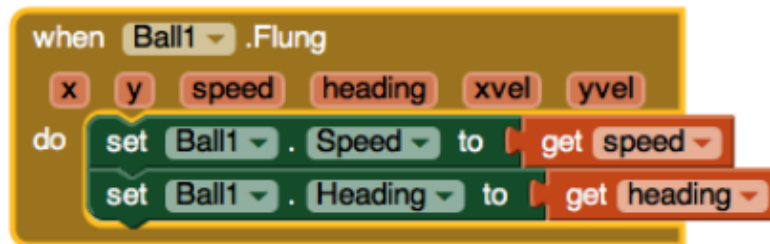
BallBounce: A simple game app - 8

Plug the set Ball1.Speed and set Ball1.Heading into the Fling event handler



Set the Ball's heading to be the same as the Fling gesture's heading

Do the same for the Ball's heading. Mouse over the **heading** parameter and you'll see the **get heading** block appear. Grab that block, and click it into the **set Ball1.Heading** block.



Set the Ball's heading to be the same as the Fling gesture's heading

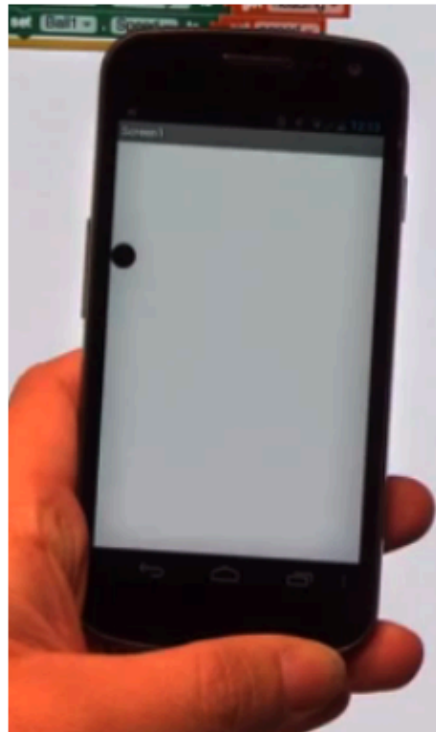
Do the same for the Ball's heading. Mouse over the heading parameter and you'll see the get heading block appear. Grab that block, and click it into the set Ball1 Heading block.

BallBounce: A simple game app - 9



Test it out

A good habit while building apps is to test while you build. App Inventor lets you do this easily because you can have a live connection between your phone (or emulator) and the App Inventor development environment. If you don't have a phone (or emulator) connected, go to the connection instructions and then come back to this tutorial. (Connection instructions are in Tutorial #1 or on the website under "Getting Started".)



Why does the Ball get stuck on the side of the screen?!

After flinging your ball across the screen, you probably noticed that it got stuck on the side. This is because the ball's heading has not changed even though it hit the side of the canvas. To make the ball "bounce" off the edge of the screen, we can program in a new event handler called "When Edge Reached".

BallBounce: A simple game app - 10

Test it out

A good habit while building apps is to test while you build. App Inventor lets you do this easily because you can have a live connection between your phone (or emulator) and the App Inventor development environment. If you don't have a phone (or emulator) connected, go to the connection instructions and then come back to this tutorial. (Connection instructions are in Tutorial #1 or on the website under "Getting Started".)

Why does the Ball get stuck on the side of the screen?!

After flinging your ball across the screen, you probably noticed that it got stuck on the side. This is because the ball's heading has not changed even though it hit the side of the canvas. To make the ball "bounce" off the edge of the screen, we can program in a new event handler called "When Edge Reached".

BallBounce: A simple game app - 10