

## 02 DRAWING - WORKSHEET

Open the project **A02\_Drawing.aia**.

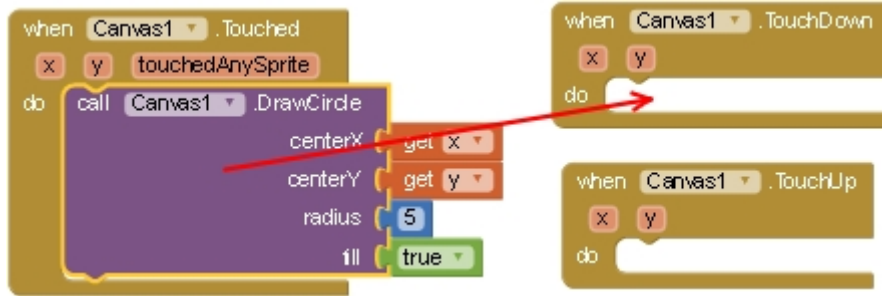
1. In the **Blocks** screen, open the block list of the **Canvas1** component and examine its events. Don't try anything at first, and just fill in the **What I'm Thinking** table column for each of the four events.

Hint: If you hover over a block and do not move the mouse, a yellow box (called a *hint*) will appear after a while, explaining the function of the block in English.

Try to understand the text, not just translate it mechanically into English.

Event	When will it occur? What does it mean parameters? What I think.	When did it occur? What does it mean parameters? What I found when I tried.
TouchDown		
TouchUp		
Touched		
Dragged		

2. Go back to **Task 1** and start testing to see if you were right by adding blocks to the program for all three of the other events, first moving the **DrawCircle** statement from the body of the **Touched** event to the body of the **TouchDown** event. Proceed so that the command is only ever in the body of one event, and try to see what happens, i.e. when the dots are drawn on the screen.



**Tip:** Try putting your finger on your phone screen, swiping and lifting it off the screen. What a difference

Do you observe when the **DrawCircle** command is in **Touched**, in **TouchDown**, and in **TouchUp**?

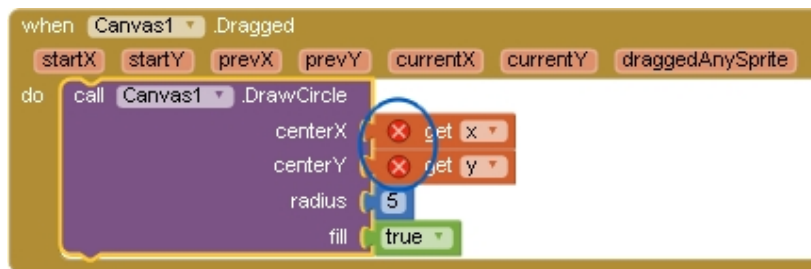
Then try to just put your finger down, don't move and pick it up.

**Tip 2:** You may not see when a dot is drawn under your finger. Make it bigger.

Based on the observations, fill in the third column of the table for **TouchUp**, **TouchDown** events

and **Touched**.

Then drag the **DrawCircle** command into the body of **Dragged**. Notice that the error markers for **get x** and **get y** have appeared:



This is because the **Dragged** event has no **x** and **y** parameters, **x** and **y** need to be changed to **currentX**

and **currentY**. Again, examine when the dots will be drawn and fill in the third column of the table for the **Dragged** event.

Then return the program to the state where the dots are drawn in the **Touched** event and **delete the other unnecessary events**.

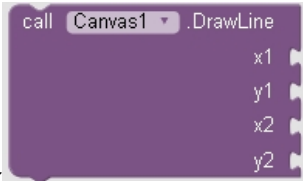
3. Explore the different parts of **App Inventor** and find the answers to your questions:

Which parameter needs to be changed and how to the <b>DrawCircle</b> command drew larger dots?	
Which parameter needs to be changed and how, so that the <b>drawCircle</b> command draws only circles and not circles, i.e., so that it does not color the inside of the circle?	
How can I delete an unnecessary block from the program? Try to find at least two different ways. Be careful, however, that when you try interference blocks did not spoil their program.	

<p>What blocks does the <b>Math</b> group contain in the <b>Blocks</b> screen <b>Blocks</b>? When are they likely to be useful?</p>	
<p>What blocks are contained in the <b>Colors</b> group in the <b>Blocks</b> screen <b>Blocks</b>? When are they likely to be useful?</p>	
<p>Explore the <b>Slider</b> component from the <b>User Interface</b> group. In order to use it properly you need to set its <b>Width</b> to some number of pixels (e.g. 80 pixels). What is set by such components in the programs you know? Which properties of the component determine the range of values it sets? What event occurs when you change position pointer (in English, what we drag is called <i>thumb</i>, I mean finger)?</p>	

4. Based on your findings, solve some of the problems:



- a) Program a line to be drawn when dragging a finger on **Canvas1**, i.e. to each time you move your finger, a line is drawn between the previous and current finger position:

 <p>Explore the block . What does the block do in the program? What do its parameters <b>x1</b>, <b>y1</b> and <b>x2</b>, <b>y2</b> mean?</p>	
<p>Using the knowledge gained from examining the <b>Canvas1.Dragged</b> event, place a <b>Canvas1.DrawLine</b> block in it and use the parameters of the <b>Dragged</b> event to draw a line from the previous finger position to the current position. Which parameter of the <b>Dragged</b> event did you use for which parameter of the <b>DrawLine</b> block?</p>	

- b) Add two new buttons that will set the drawing color: one button will set the blue color, the other will set the red color.  
First, add two new buttons to the project and name them **bBlue** and **bRed**.  
Place the buttons next to the **bDelete** button so they don't take up too much of the screen height. Tip: To place multiple components next to each other, use the **HorizontalArrangement** component from the **Layout** group and place all the buttons in it.

<p>Explore the properties of the buttons. Which property allows you to set the background color of the button? Use it by setting the background color of the buttons to the color they have in their name. Then they don't need any <b>Text</b>, so delete it.</p>	
<p>Examine the blocks of the <b>Canvas1</b> component and find the block that sets the drawing color. Which block is it? Use it in <b>bBlue</b> and <b>bRed</b> to set the desired color.</p>	

- c) Use the **Slider** component to adjust the line thickness. Add it next to the buttons for color, set its properties, and program its events to make its by pulling they set the line thickness from 1 to 5. Explore the blocks of the **Canvas1** component and find the one that sets the line thickness and use it correctly in the program.
- d) Change the command for drawing dots so that the radius of the dot is three times the set line thickness.
5. Explore the blocks provided by the **Camera1** component and use its features in the program to take a picture in the background of **Canvas1**.

<p>What the block will do</p> 	
<p>When the event is triggered</p>  <p>What does its <b>image</b> parameter mean?</p>	
<p>Add another button to the program, edit its text to <b>Take a picture</b>, call it <b>bOdfoot</b> and place it next to the <b>Select</b> button. Program it with a <b>Click</b> event to take a picture using the camera. What blocks will you use to do this? Program the camera so that the image becomes the background of <b>Canvas1</b> when the picture is taken. What blocks will you use for this?</p>	