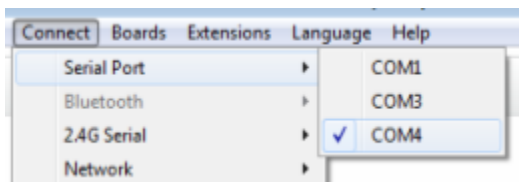


# Activities with mBot – PC and Mac

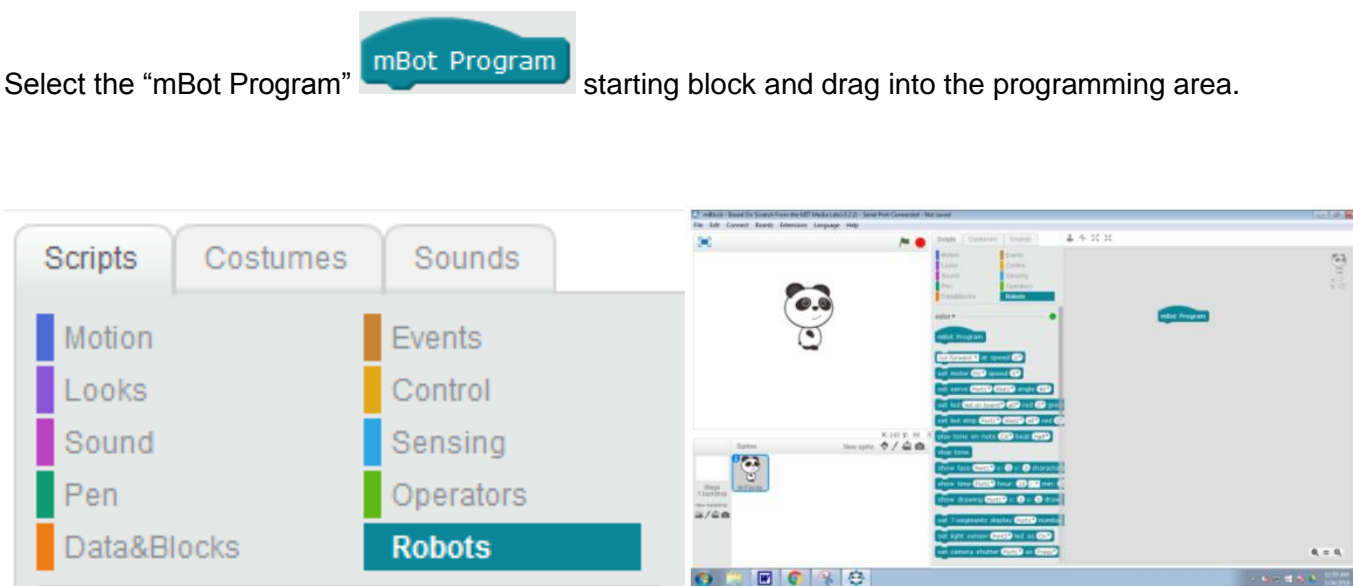
## 1. Making the mBot move!

To program the mBot, open the mBlock software on your computer and connect the mBot using the USB cable. In order to connect the mBot to the program, open the Connect tab, go down to the Serial Ports option and select either COM1, COM3, or COM4. The mBot will sound off to let you know which one it has connected to.



Next, click on the “Robots” tab under Scripts. This will pull up the different function options for the mBot.

Select the “mBot Program” starting block and drag into the programming area.



Now, select the “run forward” block and drag it under the starting block to connect them. You will need to adjust the speed to 100. 100 is the recommended speed for accuracy.

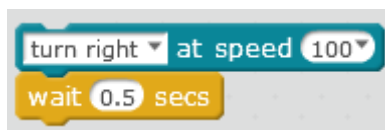


To determine how long the mBot is to travel forward, go to the “Control” options under Scripts, select the yellow “wait” block, and drag it into your code, connecting it to the previous blocks. Type in the number of seconds you want it to run forward.

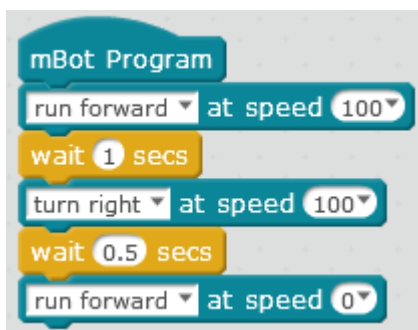


# Activities with mBot – PC and Mac

Go back to the “Robots” options and select the “run forward” block again but this time in the drop down menu select the “turn right” option. Setting the time to 0.5 seconds will allow the robot to turn approximately 90 degrees. Add another “wait” block.



To finish off the program, select a final “run forward” block and connect it to your code but this time keep the speed set to 0. This is necessary to turn off the mBot’s motors. Your final code should look like this:



To make the mBot run the program double click on the “mBot Program” and watch your mBot go!

To SAVE, click FILE and then SAVE PROJECT.

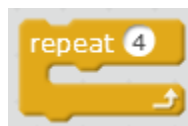
*To allow the mBot to run your program off of the USB cord, single click on the “mBot Program” block at the beginning of your code. Your code in Arduino should appear. Above the Arduino, click on the “Upload to Arduino” button. Once this has loaded, turn your mBot off and unplug the cord. When you trun your mBot back on it will automatically start running your code.*

# Activities with mBot – PC and Mac

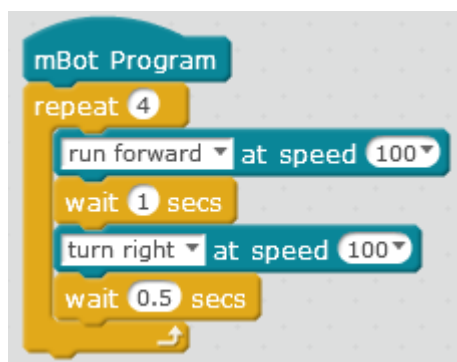
## 2. Making the mBot repeat! (Repeat Loop)

To move the mBot in a box, first you will need to select the “mBot Program” starting block and drag into the programming area to start your code.

Next, go to the “Control” options under Scripts, select the yellow “repeat” block, and drag it into your code. Since we want the robot to move in a square, set the number of times to 4.



Next, use some of the code from Activity 1 and insert it into the loop. Your code should look like this:



Finally, add a “run forward” block to the end of the code to stop the mBot after it has completed the loop.



To make the mBot run the program, double click on the “mBot Program” block.

To SAVE, click FILE and then SAVE PROJECT.

# Activities with mBot – PC and Mac

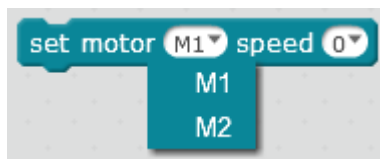
## 3. Controlling the motors individually!

The mBot has two motors, M1 and M2. M1 controls the left wheel and M2 controls the right wheel. You can also control the mBot in the same way as in the previous activities by controlling these motors individually.

In this activity we will program the mBot to move forward, turn right and then left by only manipulating the motors.

First you will need to select the “mBot Program” starting block and drag it into the programming area to start your code.

Next, select the “set motor” block. As seen here you can select which motor you want to control and how fast you want it to go.



To make the mBot go straight we will need to set both motors to speed 100. Drag a second “set motor” block under the first. Set the first as M1 and the second as M2, and set the speed at 100 for both. Add a “wait” block from the control settings and set it to 1 second.

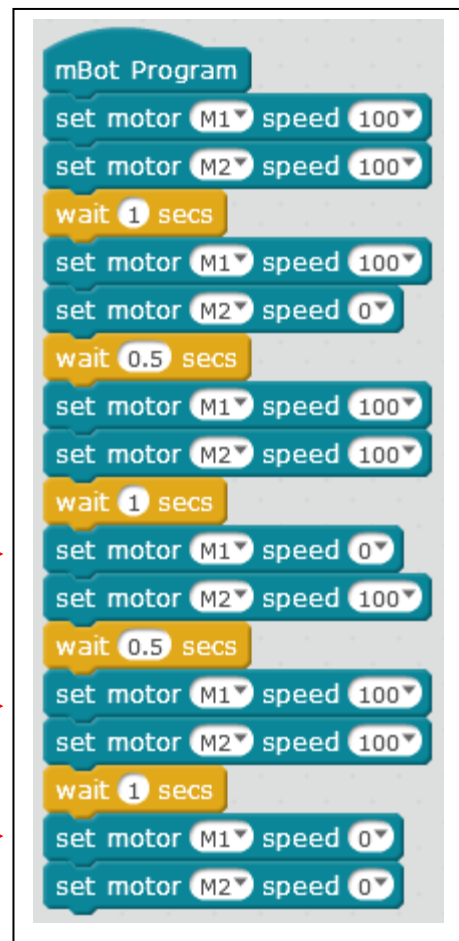
Next we want to make the mBot turn right. Add two more “set motor” blocks but this time set M1 speed to 100 and M2 speed to 0. Add a “wait” block from the control settings and set it to 0.5 seconds.

After turning the mBot have it go forward again. Add two more “set motor” blocks setting both M1 and M2 to 100. Add a “wait” block from the control settings and set it to 1 second.

Next we want to make the mBot turn left. Add two more “set motor” blocks but this time set M1 speed to 0 and M2 speed to 100. Add a “wait” block from the control settings and set it to 0.5 seconds.

Let’s have the mBot go straight once more to finish off its path. Add two more “set motor” blocks setting both M1 and M2 to 100. Add a “wait” block from the control settings and set it to 1 second.

Finally, to stop the mBot as two more final “set motor” blocks setting both of their speeds to 0. Your code should look like this:

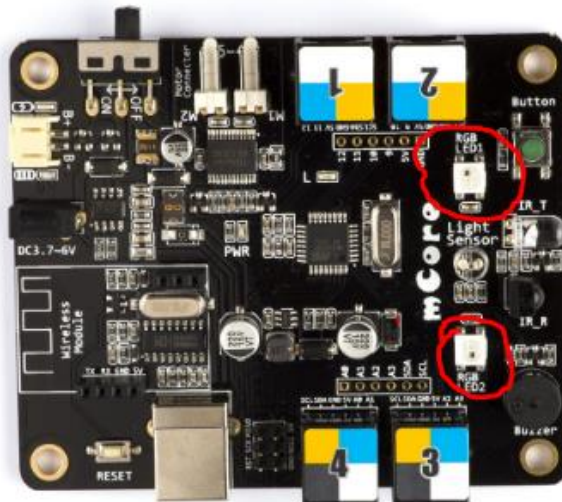


Double click the “mBot Program” block to start it.  
To SAVE, click FILE and then SAVE PROJECT.

# Activities with mBot – PC and Mac

## 4. Light it up!

Another feature of the mBot is the LED lights that it has located on its mCore board.



We can program when these shine and even what colors we want them to light up.

To start, you will need to select the “mBot Program” starting block and drag into the programming area to start your code.

Next select the “set led” block and attach it to the code. Here select the “led on board” option, make sure it is set to all and then indicate which color you want to shine and how brightly. Here I chose red and set it to 60. Keeping the other two colors at 0 will make sure that only red lights up.

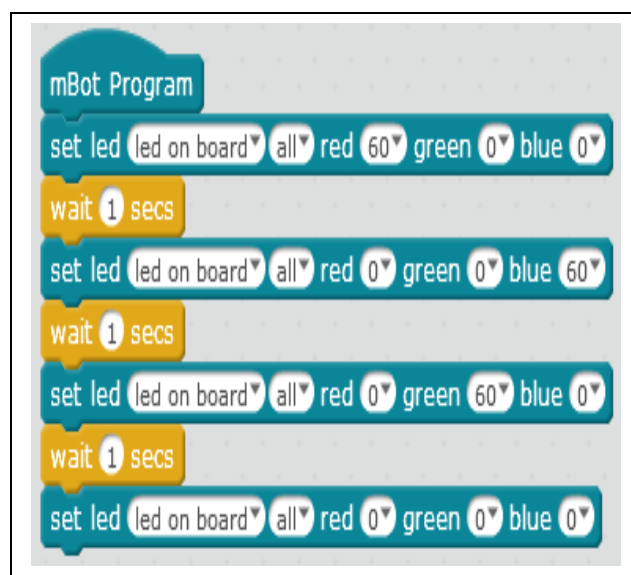


In order to indicate how long you want the lights to shine add a “wait” block and set it to 1 second.

Repeat this two more times, but the second time set blue to 60 and red and green to 0, and the third time set green to 60 and red and blue to 0. Be sure to add a wait block after each one.

Finally to shut the lights off add one more “set led” block and set all three colors to 0.

Double click the “mBot Program” block to start it.  
To SAVE, click FILE and then SAVE PROJECT.



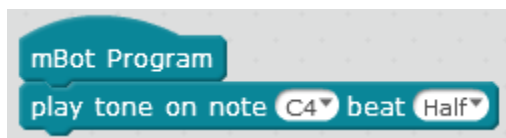
# Activities with mBot – PC and Mac

## 5. Sing, Sing, Sing!

The mBot can also sing notes and with programming you can tell it which notes and how long the notes should be, ultimately creating little tunes.

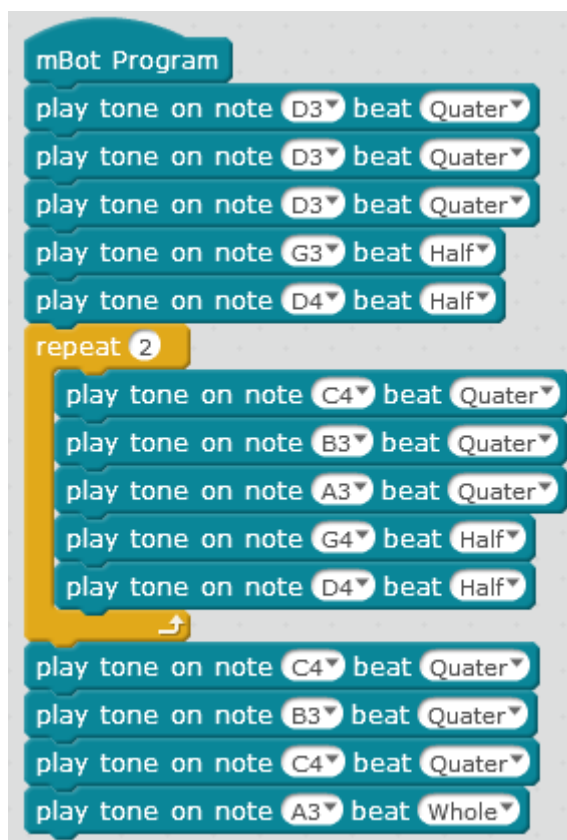
To do this you will first need to select the “mBot Program” starting block and drag into the programming area to start your code.

Next select the “play tone on note” block and add it to your code.



There are 23 different notes you can choose from. The notes range from C2 to D5, covering a little over 3 octaves. To create a tune simply drag over as many “play tone on note” blocks as you need and set their tones and how long you want them to be. The note lengths range from “Zero” to “Double.”

For example, here is the beginning of the Star Wars melody:



Double click the “mBot Program” block to start.  
To SAVE, click FILE and then SAVE PROJECT.

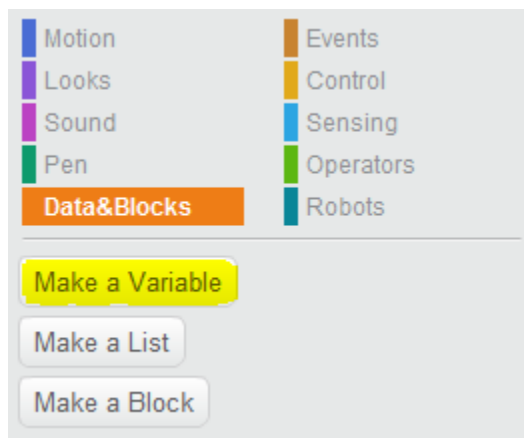
# Activities with mBot – PC and Mac

## 6. Don't Crash!

The mBot can be made to avoid obstacles to avoid crashes.

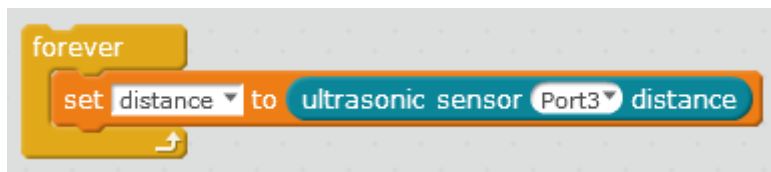
To do this, first you will need the “mBot Program” block.

Next a variable block is required. To assign a variable go to the orange “Data & Blocks” tab and click on the “Make a Variable” button:

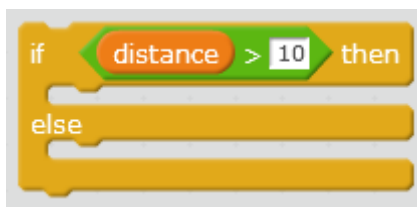


Name your variable “distance.”

Next go to the yellow “Control” tab. Drag a forever loop into your code. Within the loop place a variable “set” block. This time in the box next to “to” drag and drop a ultrasonic sensor block from the “Robots” tab. Be sure to set the correct port for the ultrasonic sensor.



Next, and still within the forever loop, add an “if else” block from the “Control” tab. Within the “if” condition slot place a green “>” block. For the left side, drag a “distance” variable, and for the right type “10.”



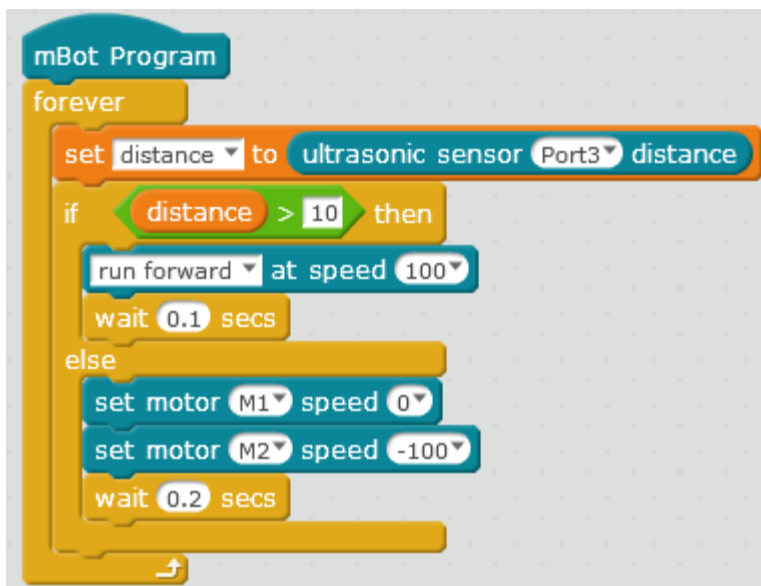
Within the “if” loop place a “run forward” block, setting the speed to 100, and a “wait” block setting the time to 0.1 seconds.



# Activities with mBot – PC and Mac

In the else loop drag two “set motor” blocks. Set the first as M1 with a speed of 0 and the second as M2 with a speed of -100. Add a final “wait” block and set as 0.2 seconds. This will move the mBot backwards and to the left every time it comes close to an object.

Your final code should look like this:



The forever loop will constantly check for obstacles and as long as they are more than 10 units away it will allow the mBot to move forward. When they become 10 units or closer, however, the mBot will back up, turn and continue moving forward.

Double click the “mBot Program” block to start.  
To SAVE, click FILE and then SAVE PROJECT.



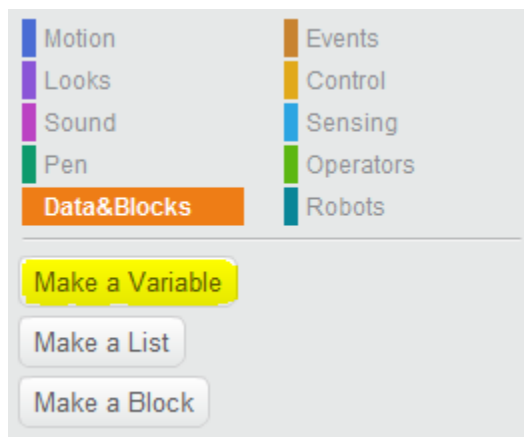
# Activities with mBot – PC and Mac

## 7. Follow the Line!

The mBot can be programmed to follow a line using sensors.

To do this, first you will need the “mBot Program” block.

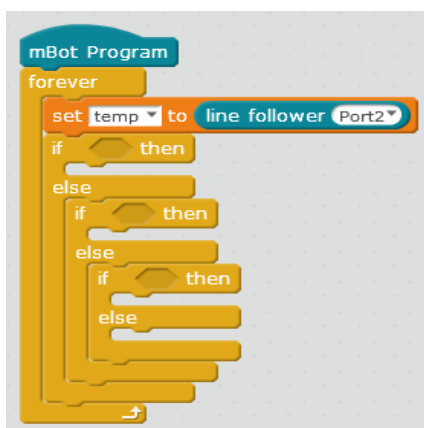
Step 1: A variable block is required. To assign a variable go to the orange “Data & Blocks” tab and click on the “Make a Variable” button:



Name your variable “temp.”

Step 2: Go to the yellow “Control” tab. Drag a forever loop into your code. Within the loop, place a variable “set” block. Inside the box next to “to,” drag a “line follower” block from the “Robots” tab. Make sure it is set to the correct port.

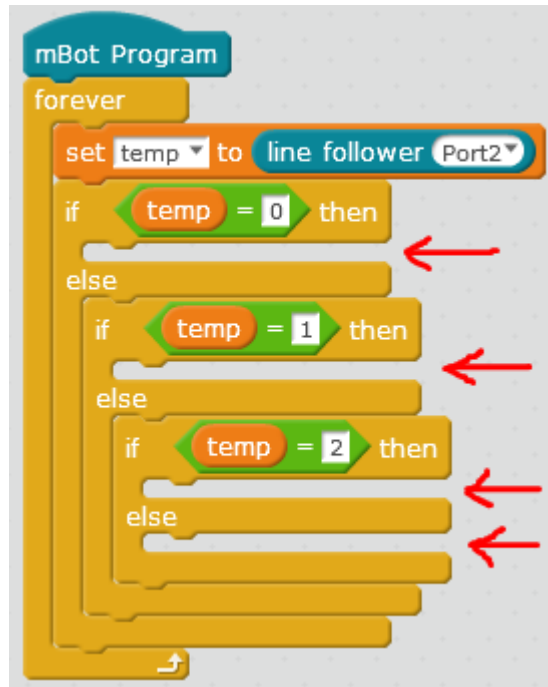
Step 3: Nest three “if else” blocks within the forever loop. This means that you place the first “if else” block under the “set” block. Place the second “if else” block within the “else” section of the first and then the third “if else” block within the “else” section of the second:



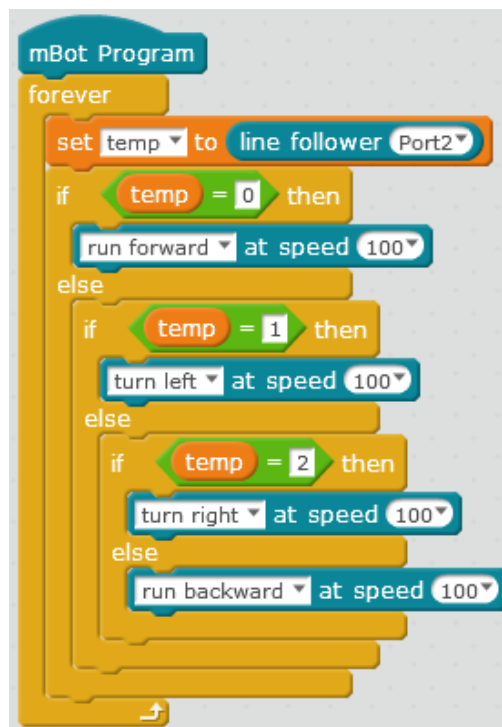
# Activities with mBot – PC and Mac

Step 4: In each of the “if” condition boxes place a green “=” block from the “Operators” tab. Set the left side to all three as the variable “temp” and the right side of the first as 0, the second as 1, and the third as 2.

Step 5: In the four slots shown below, place in each a “run forward” block from the “Robots” tab. Make the top one forward, the second one left, the third right, and the bottom one backward, setting all speeds to 100.

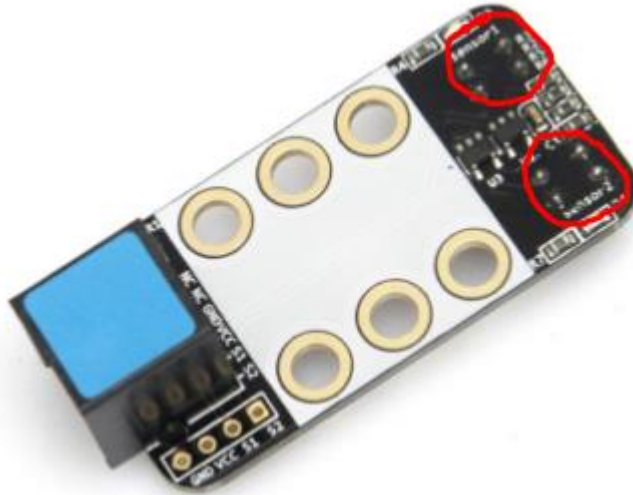


Your final code should look like this:



# Activities with mBot – PC and Mac

Note: The line follower on the mBot has two sensors a left and right one as seen here:



The mBot has already programmed to know that when the line follower = 0, that means that both sensors detect a line indicated by both blue lights shutting off. When this is the case the mBot can move forward.

When the line follower = 1 the right sensor no longer detects a line indicated by the right blue light turning on. In order to get the mBot back on the line, therefore, we turn the mBot left until both sensors are activated and the mBot continues moving forward.

When the line follower = 2 the left sensor no longer detects a line indicated by the left blue light turning on. So we turn the mBot right until both sensors are activated and the mBot continues moving forward again.

The last condition occurs when both sensors no longer detect a line and in this case we move the mBot backwards.

Double click the “mBot Program” block to start.  
To SAVE, click FILE and then SAVE PROJECT.

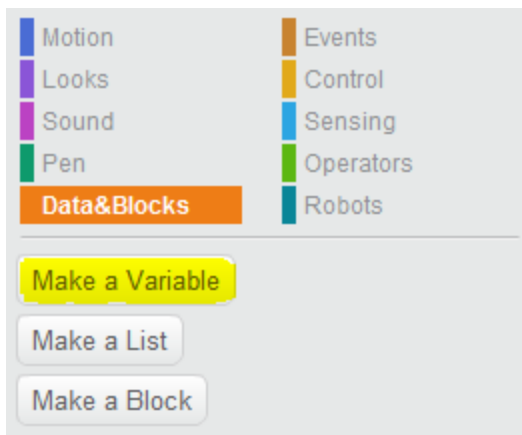
# Activities with mBot – PC and Mac

## 8. Inside the Box...or Circle

mBot can be programmed to stay inside lines drawn on the surface.

To do this, first you will need the “mBot Program” block.

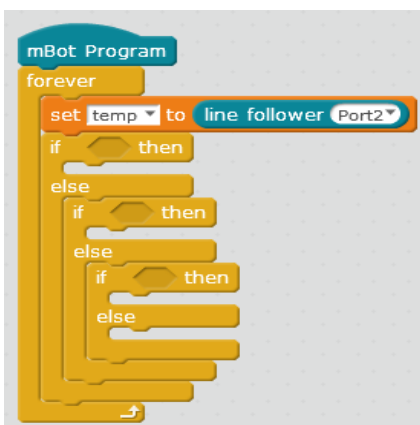
Step 1: A variable block is required. To assign a variable go to the orange “Data & Blocks tab and click on the “Make a Variable” button:



Name your variable “temp.”

Step 2: Go to the yellow “Control” tab. Drag into your code a forever loop. In there place a variable “set” block. Inside the box next to “to,” drag a “line follower” block from the “Robots” tab. Make sure it is set to the correct port.

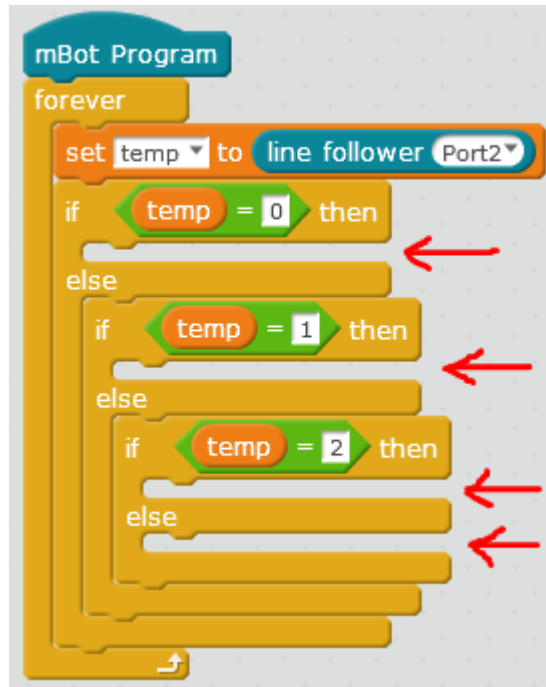
Step 3: Nest three “if else” blocks within the forever loop. This means that you place the first “if else” block under the “set” block. Place the second “if else” block within the “else” section of the first and then the third “if else” block within the “else” section of the second:



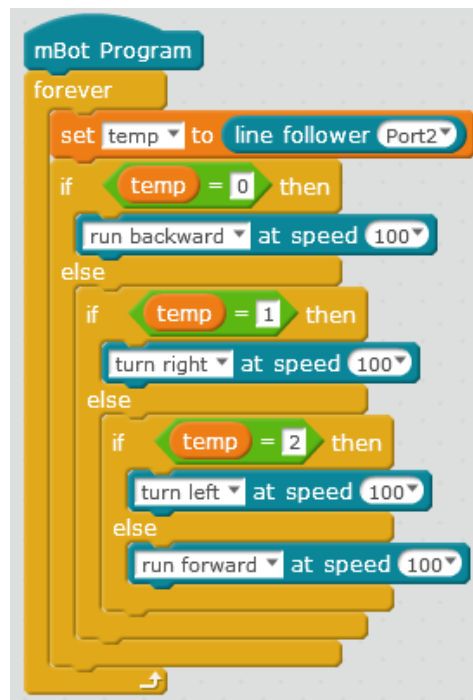
# Activities with mBot – PC and Mac

Step 4: In each of the “if” condition boxes place a green “=” block from the “Operators” tab. Set the left side to all three as the variable “temp” and the right side of the first as 0, the second as 1, and the third as 2.

In the four slots shown below, place in each a “run forward” block from the “Robots” tab. Make the top one backward, the second one right, the third left, and the bottom one forward, setting all speeds to 100.



Your final code should look like this:



# Activities with mBot – PC and Mac

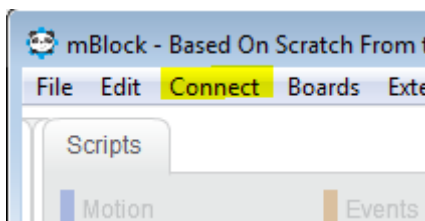
## 9. Writing and Drawing – Using LED Matrix Accessory

The mBot also has many different extensions and add-ons that can be attached to it and used for even more cool projects.

For example, the Me LED Matrix 8x16 is a LED board covered by a translucent acrylic screen which can project words, the time of day, and even drawings.

Here we are going to program our LED Matrix to show the word “Hi”, the time of day, and then a drawing of a heart.

First, in order to connect the Matrix to mBlock you will need to go to the Connect tab at the top of the page and under in select “Upgrade Firmware.” This will take a couple seconds to load.



Now you are ready to program to Matrix.

As always to start the code, place the “mBot Program” starting block in the programming area.

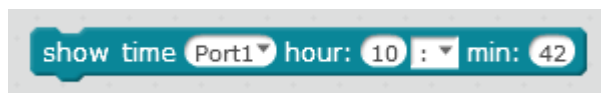
Next drag the “show face” block and attach it to your code. Be sure to take note of what Port the LED Matrix is connected to on your mBot and select accordingly in the drop down menu on the block. The x and y options should both be set to 0. Then under characters type in what you want the mBot to say. In this case, we are going to type the word “Hi.”



To specify how long we want this to shine on our screen add a control “wait” block and set it to 3 seconds.



Next we want to show the time of day. Drag the “show time” block and add it to your code. Set the Port to the same one as before and then set the hour and minutes.



Add another “wait” block.

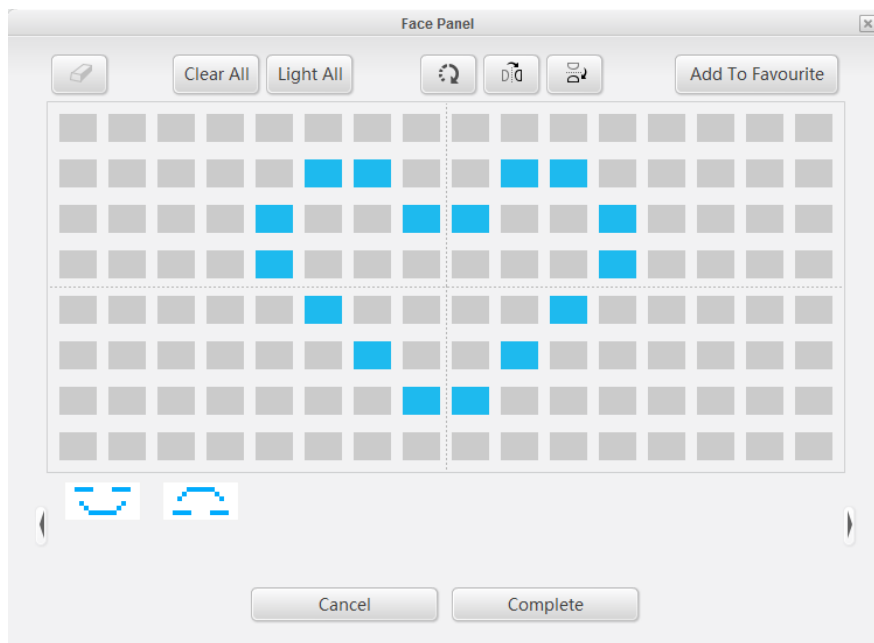


# Activities with mBot – PC and Mac

Finally, let's add a drawing. Drag a “show drawing” block into your code. After setting the Port and making sure the x and y options are set to 0, double click on the highlighted blank box next to “draw:”

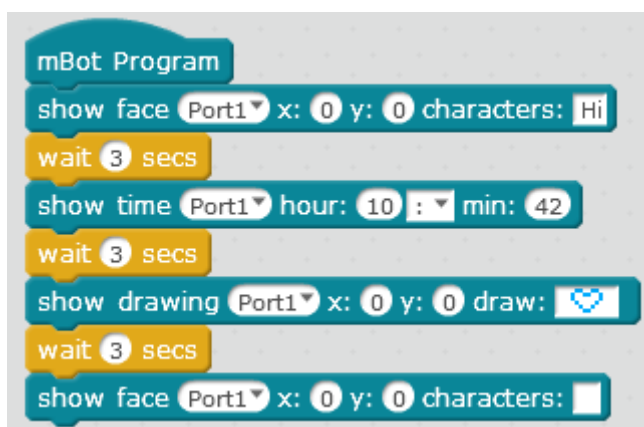


A grid of 8x16 should come up when you do so. By clicking on the individual boxes you can select which lights you want to light up. To draw a heart select these:



Finally, add a last “wait” block and one more “show face” block this time leaving the box next to “characters:” blank. This will turn the Matrix off.

Your final code should look like this:



Double click the “mBot Program” block and watch your mBot light up!

To SAVE, click FILE and then SAVE PROJECT.

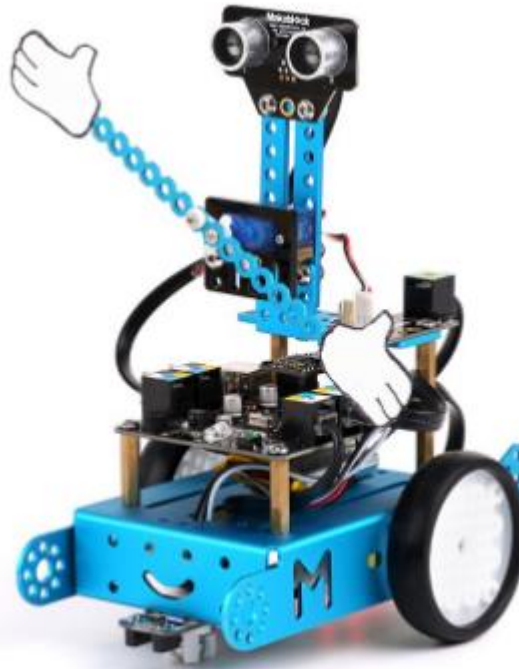


# Activities with mBot – PC and Mac

## 10. Friendly mBot – Using Servo and Extension Kit

There are other attachments as well that allow the mBot to move and work more efficiently.

For example, there is an addition called the mBot Servo Pack. This comes with a rotational motor. Here we will demonstrate how to give the mBot a rotating arm that moves from side to side when assembled with the Servo pack. An additional pack called the mBot Bracket pack is also required. The end result is seen here (the hands are not included in either pack):



Once the mBot is assembled with the attachment, here is how to program it.

Start the program off with the “mBot Program” block.

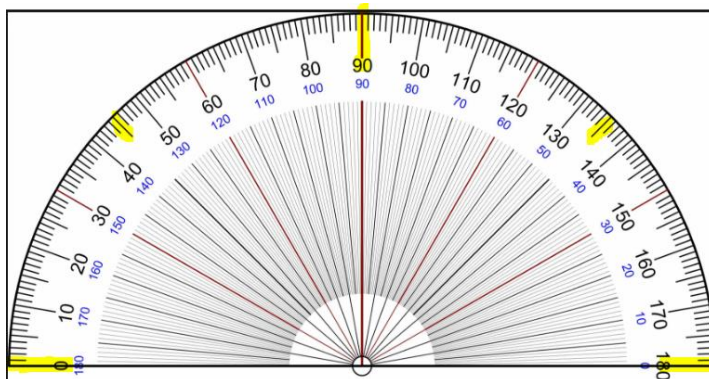
Next find the “set servo” block. Add this to your code as well. Make sure to set the corresponding Port and Slot that the servo is plugged into on your mBot.



# Activities with mBot – PC and Mac

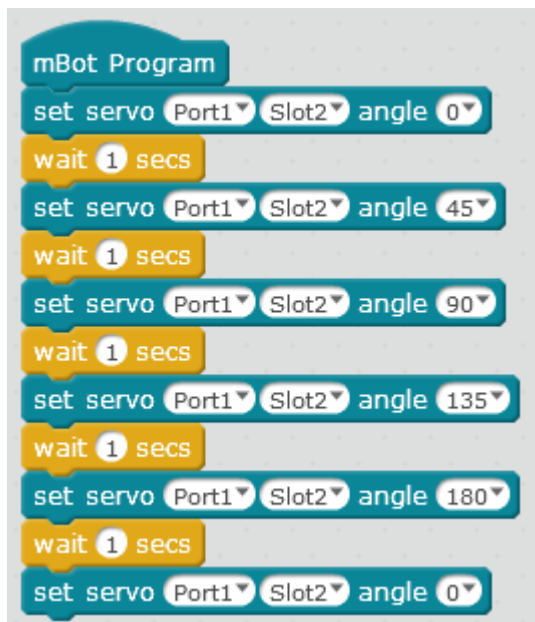
The options next to the word “angle” allow you to control the position of the motor such as on a protractor.

For example:



In this code let's set the first angle to 0. Add a “wait” block and another “set servo” block this time setting it to 45 degrees. Continue this pattern with a 90, 135, 180, and a final 0 block.

Double click the “mBot Program” to watch the mBot. This is what your final code should look like.



If you want the arm to be parallel to the ground when the servo has been installed vertically, set the angle to 23.5 degrees (24 was still slightly off) by typing your number in the drop down menu.

This servo motor is vertical when the arrow points upwards.



To SAVE, click FILE and then SAVE PROJECT.

\*\*\*\*\*Technical support, please call 281-312-1297, 1-866-722-2051 or email [support@smartschoolsystems.com](mailto:support@smartschoolsystems.com)

Page | 17