**GESTURE CONTROL FOR LAPTOP**

**Gesture controlled Laptops or Computers** enables us to control certain functions on Laptop/Computer by simply waving our hands in front of it.

In this project, we will build our own **Gesture controlled Laptop/Computer** by combining the power of **Arduino Leonardo** and **Ultrasonic sensors**.

We will be using two Ultrasonic Sensors to determine the position of our hands and control a Media Player (Windows) based on the position.

Here, we will be controlling four actions:

* MUTE
* VOLUME UP
* VOLUME DOWN
* PLAY/PAUSE

**Pre-requisites:**

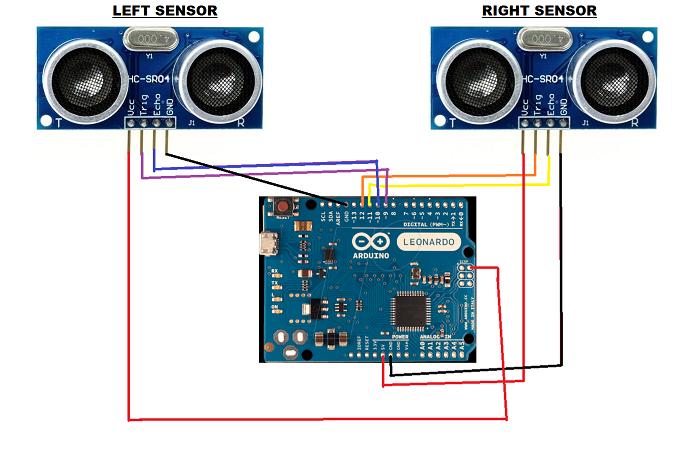
* Arduino Leonardo (1)
* Ultrasonic sensors – HC-SR04 (2)
* Jumper Wires (according to the connections 8 wires are required)
* Arduino IDE
* USB

**Concept Behind:**

The concept behind this is very simple. We will place two Ultrasonic sensors on the top of the monitor and it will read the distance between the monitor and our hands using Arduino.

Based on the distance, certain actions will be performed.

For the performance of the actions, we have used the Keyboard library of Arduino Leonardo as it has built-in USB communication.

**Circuit Diagram:**

**CONNECTIONS:**

* LEFT SENSOR:

Vcc: +5V

Trig: 9 (Pin)

Echo: 10 (Pin)

GND: Arduino GND

* RIGHT SENSOR:

Vcc: +5V

Trig: 12 (Pin)

Echo: 11 (Pin)

Gnd: Arduino GND

**Windows Media Player (shortcut keys):**

* MUTE:

F7 key

* PLAY/PAUSE:

Ctrl + p

* VOL UP:

F9 key

* VOL DOWN:

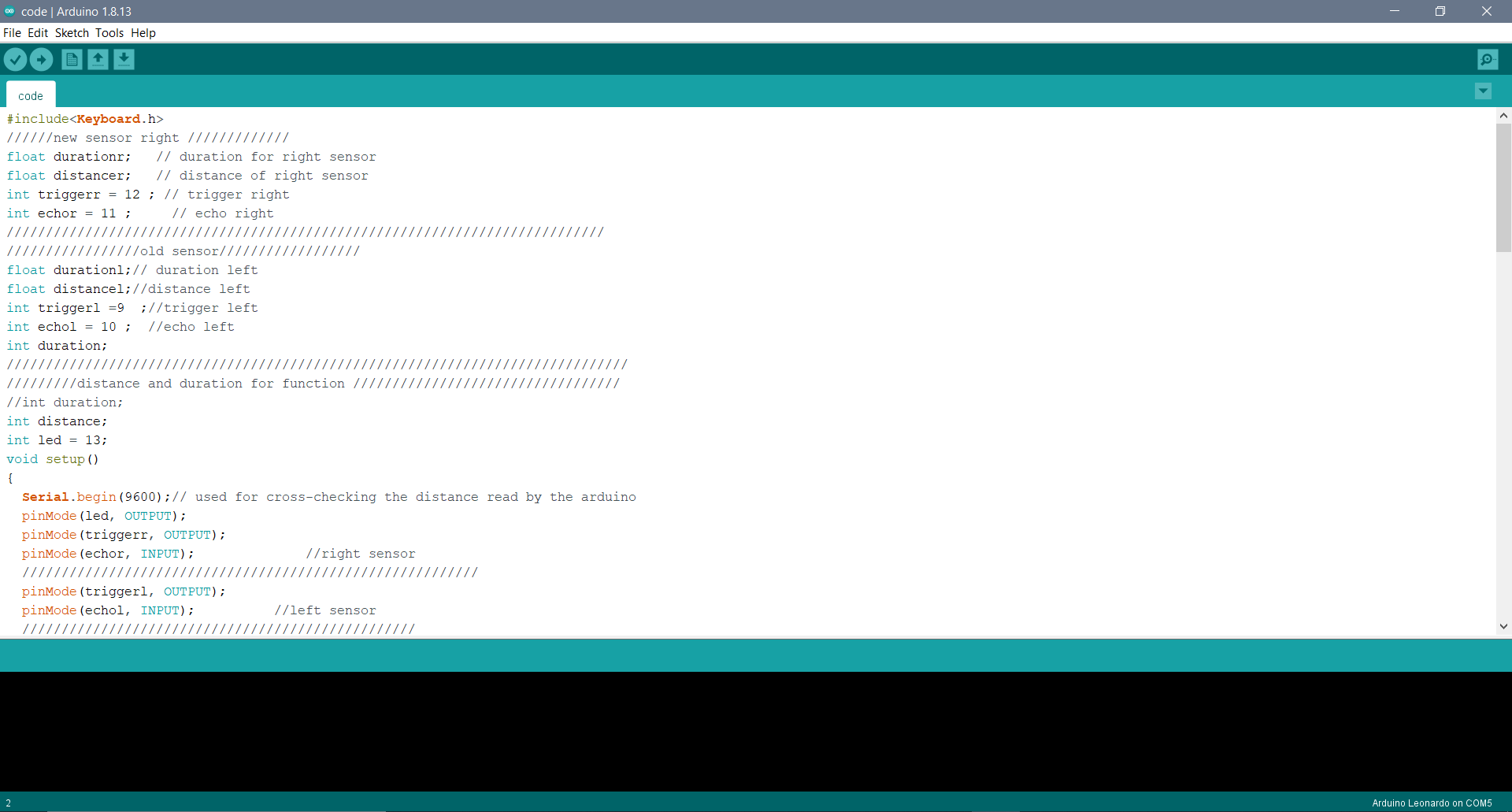
F8 key

**Actions:**

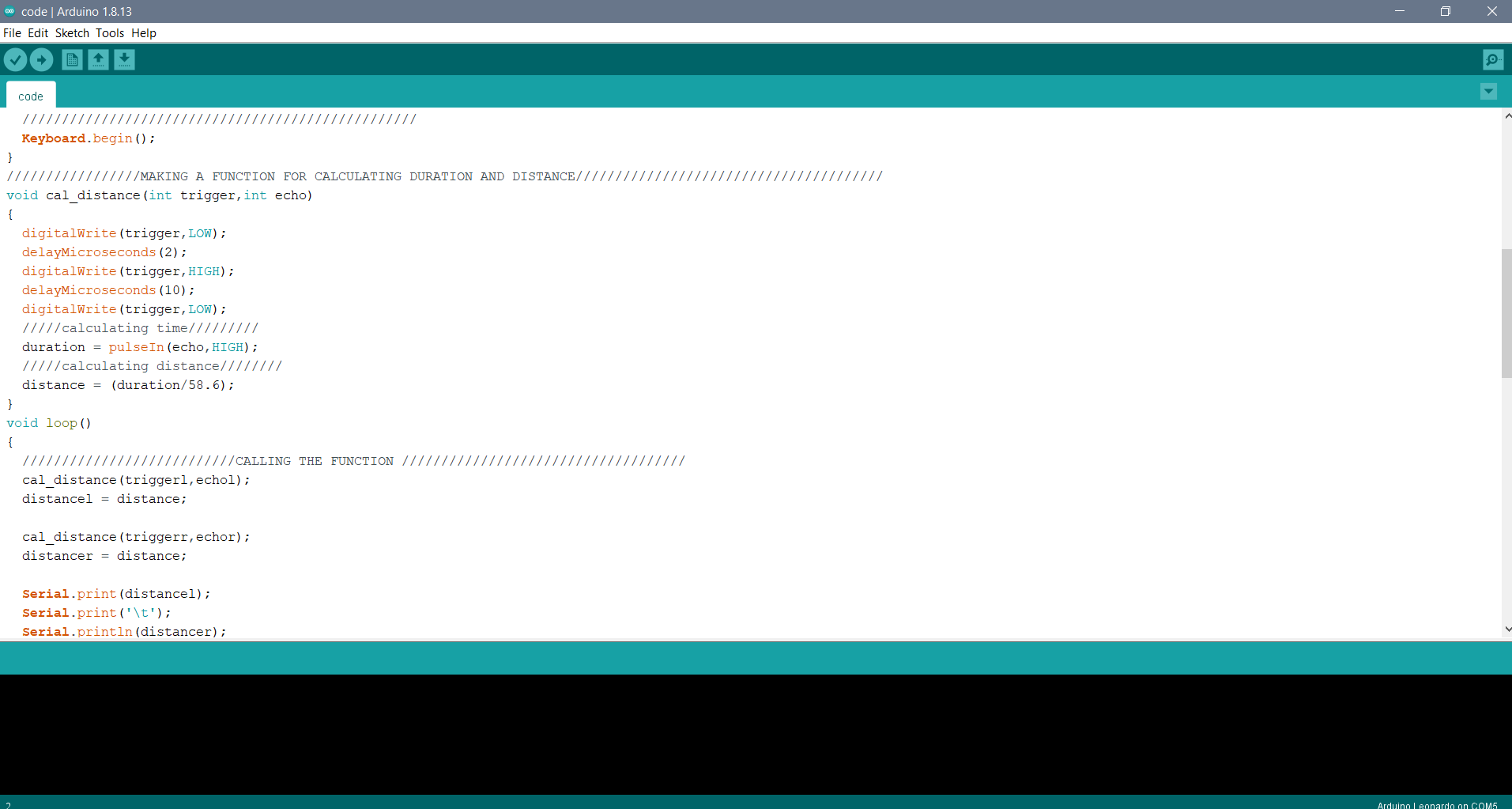
Left sensor for the distance of 10cm to 15cm will mute the video and for 20cm to 25cm will increase the volume.

Right sensor for the distance of 10cm to 15cm will play/pause the video and for 20cm to 25cm will decrease the volume.

**Arduino Program:**

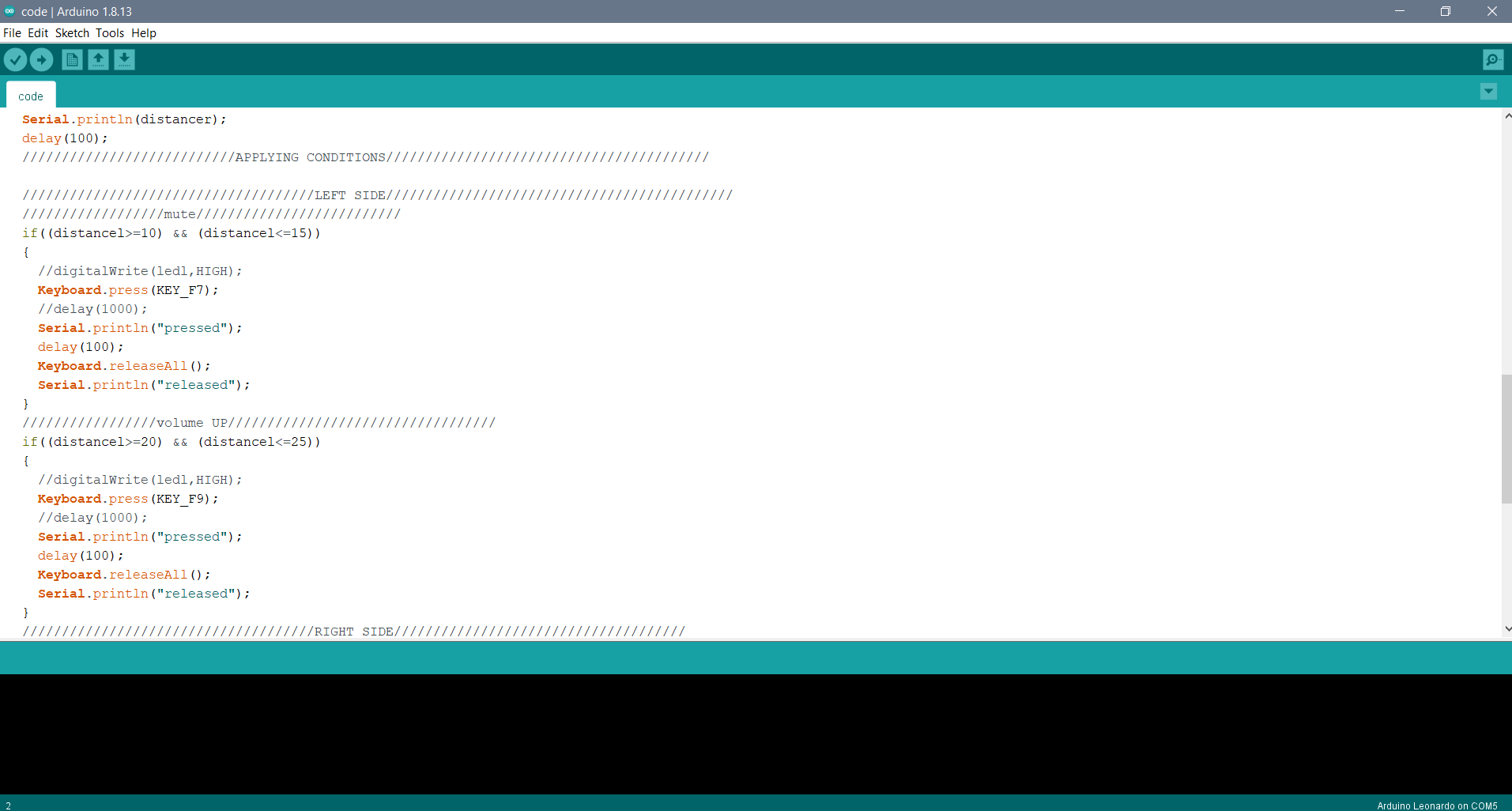
Firstly, include the **Keyboard library** and then declaration of the variables used.

Second step is to start with the technical loop i.e., the void setup () and then initialize the pins and begin the Serial and Keyboard library.



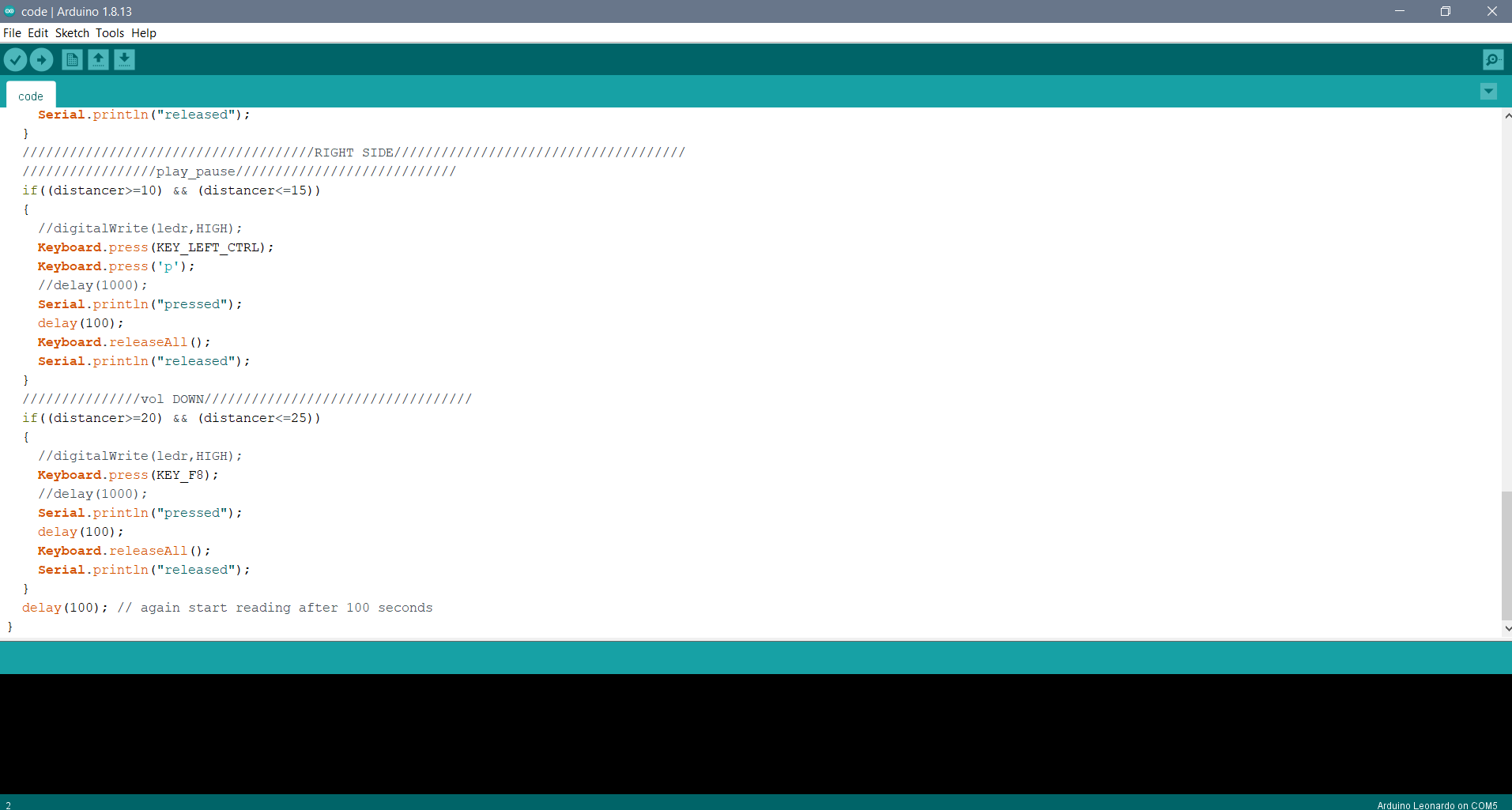
Third step will be the making of a function (cal\_distance) for the calculation of the distance.

In the fourth step, we will call the function and pass on the values and printing them on the serial port.



Fifth step will be giving the conditions based on the distance measured by both the sensors.

And at last, we will give a delay of 100 seconds for the next scan to begin.



Now, verify and then upload the code.

**Remember,** to select the port and board accordingly from the Tools options.

This code can be used for controlling the volume and play/pause functions.

Functions can be replaced with their shortcut keys as per the requirement.