



PYTHON & ARDUINO

Face Presence Detection

Ahmed Ashraf Montasser



Agenda

1

Idea of the project



2

Software



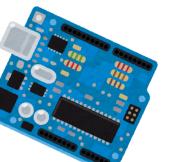
3

Hardware



4

Arduino Circuit



5

Arduino Code



6

Python Code



7

Material Used





IDEA

- ▶ The python program will start up the camera and the camera will start taking frames if a face is detected in these frames it will turn on a led and if there is no face detected it will turn on another led and a buzzer

Software

The Software used in this project is:

VS Code



Visual Studio Code

Arduino IDE

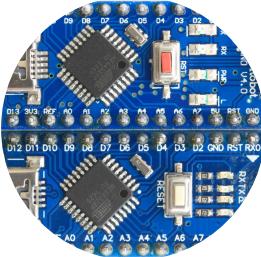


used to write the python code
and to connect it to the arduino

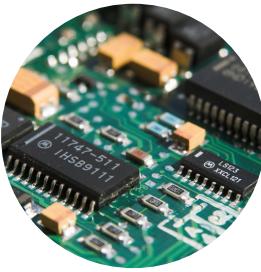
used to write the arduino code
and control the hardware

HardWare

The Hardware used in this project is:



Arduino Nano



BreadBoard



Jumper Wires



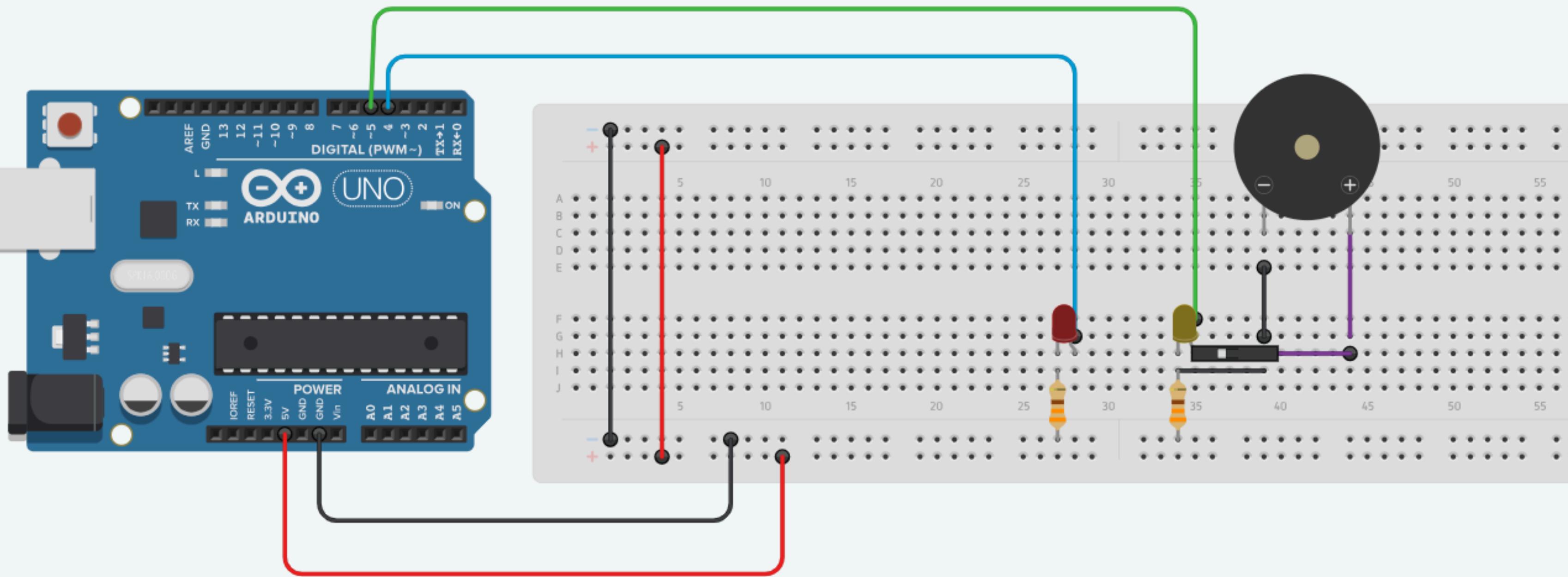
Led & Resistors



Buzzer

Arduino Circuit

The hardware and arduino
circuit connected together:



Arduino Code

```
#include <cvzone.h> // include libirary

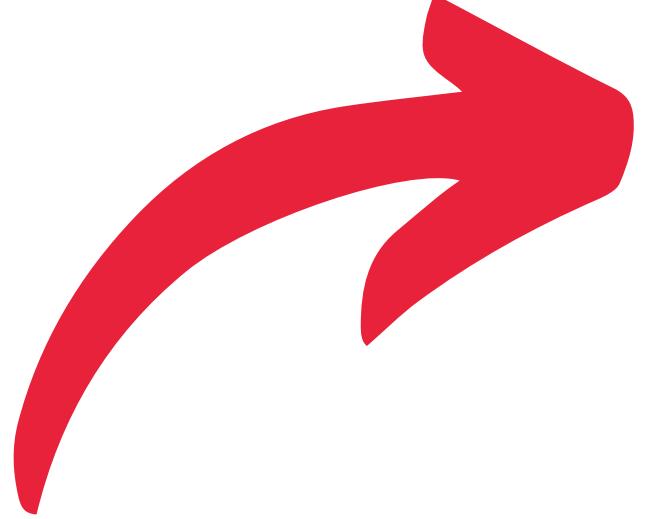
SerialData serialData(2, 1); // creat a serial of two classes
int valsRec[2]; // creat an array with two varaiables

void setup() {
    pinMode(4, OUTPUT); // output1
    pinMode(5, OUTPUT); // output 2
    serialData.begin(); // start serial communication

}

void loop() {
    serialData.Get(valsRec); // get the serial data and put it in the array
    digitalWrite(4, valsRec[0]); // turn on output according to the serial data
    digitalWrite(5, valsRec[1]); // turn on output according to the serial data
    delay(10);

}
```



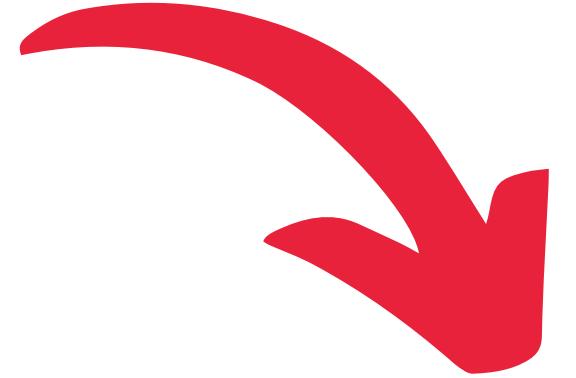
it includes the cvzone library which is responsible for computer vision projects

```
#include <cvzone.h> // include libirary
```

```
SerialData serialData(2, 1); // creat a serial of two classes  
int valsRec[2]; // creat an array with two varaibles
```



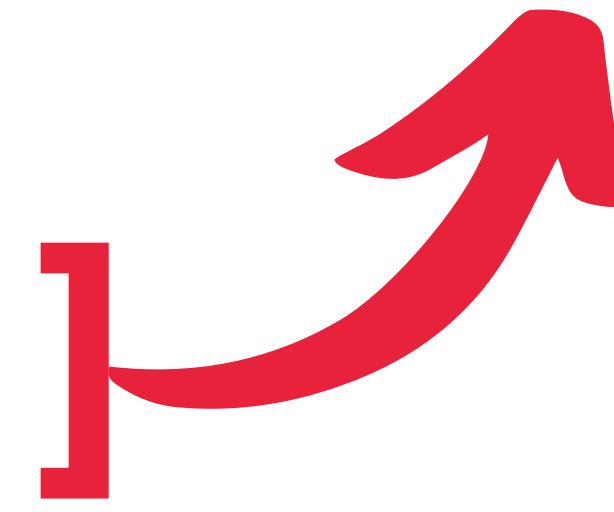
it creates an array that store the received data from the python program



it creates a serial object that starts serial communications and take two parameters

it defines the
hardware as outputs

```
void setup() {  
    pinMode(4, OUTPUT); // output1  
    pinMode(5, OUTPUT); // output 2  
    serialData.begin(); // start serial communication  
}
```



this line starts serial
communication

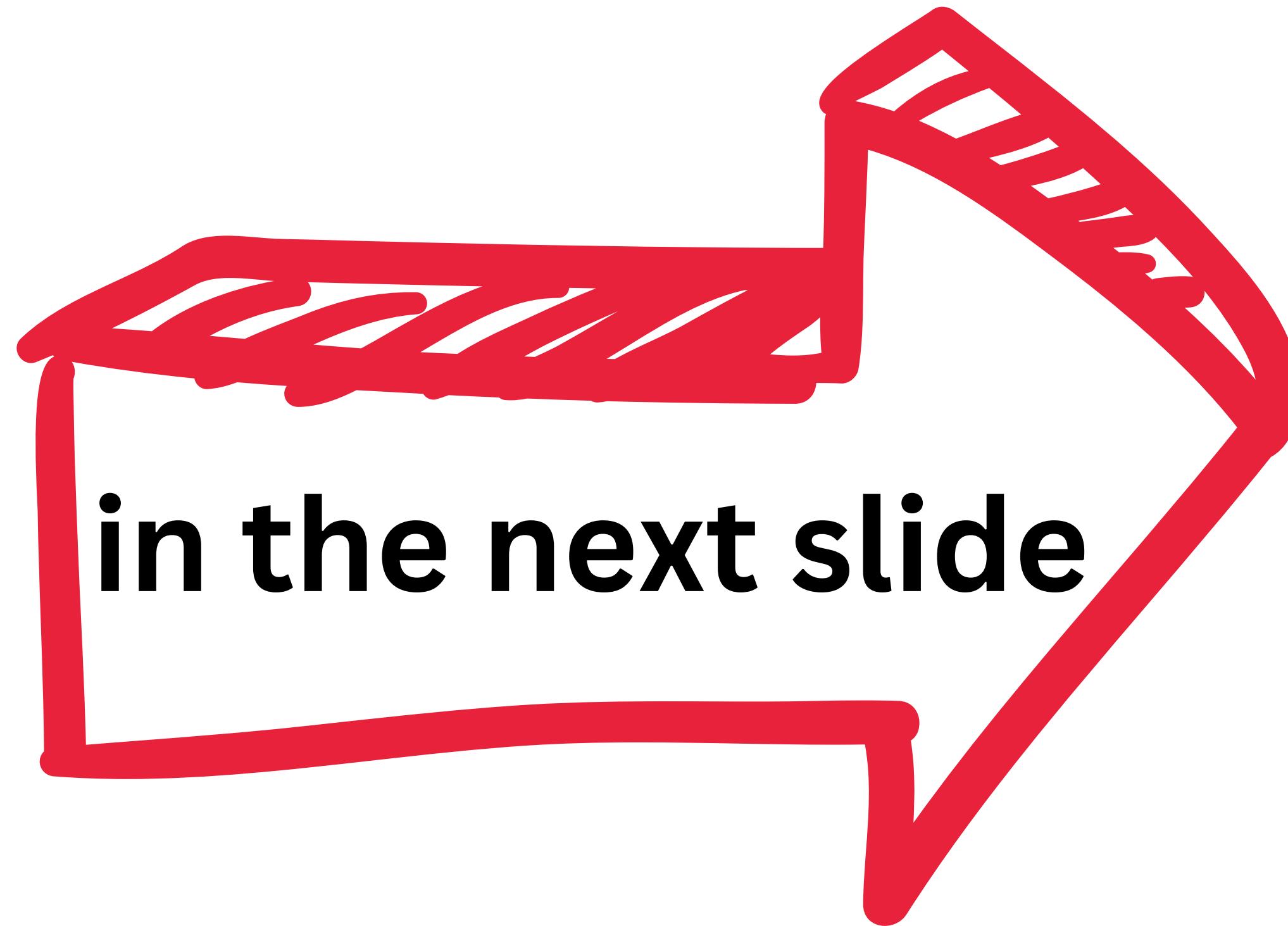
it reads the data from the serial port
and stores it in the array

```
void loop() {  
    serialData.Get(valsRec); // get the serial data  
    digitalWrite(4, valsRec[0]); // turn on output  
    digitalWrite(5, valsRec[1]); // turn on output  
    delay(10);  
}
```



it turns on the outputs according to the data
in the array the 1st element controls (pin4)
and the 2nd controls (pin5)

Python Code



[Back to Agenda](#)

```
1 import cv2 #imports opencv library used for computer vision and image & video processing
2
3 from cvzone.FaceDetectionModule import FaceDetector # from the cvzone library import the FaceDetection module
# and from the FaceDetection module import FaceDetector class which will help in face detection
4
5
6 from cvzone.SerialModule import SerialObject # from the cvzone library import SerialModule
# form SerialModule import SerialObject class to connect to the arduino using serial port
7
8 ##### to install libraries : in the terminal write "pip install (library name)" #####
9
0
1 cap=cv2.VideoCapture(0) # opens camera and start taking frames from it and store them in Cap
2 detector = FaceDetector() # it will start the face detections from the frames
3 arduino=SerialObject('COM14') # it will start sending serial data to arduino through the port ('COM14')
4
5 while True: #infinite loop
6     success,img = cap.read() # reads the frames that are stored in cap then stores them in (img)
# (success) indicates if the frames are read successfully or not
7
8     img, bBoxs = detector.findFaces(img) # detect face in the frames inside (img)
# and put the face in a box and return the face with the box to (bBox)
9
0
1     if bBoxs: # checks if a face is detected or not
2         arduino.sendData([1,0]) # sends data to the arduino to turn on the first output
3     else :
4         arduino.sendData([0,1]) # sends data to the arduino to turn on the second output
5
6     cv2.imshow("Ahmed Ashraf", img) # display the image in a window (face with box around it)
7
8     cv2.waitKey(1) # wait for a key press for 1 millisecond
9
```

Material

all the material
related to the
project will be
here





[Back to Agenda](#)