IBM

ASSIGNMENT 3

TEAM ID: PNT2022TMID05356

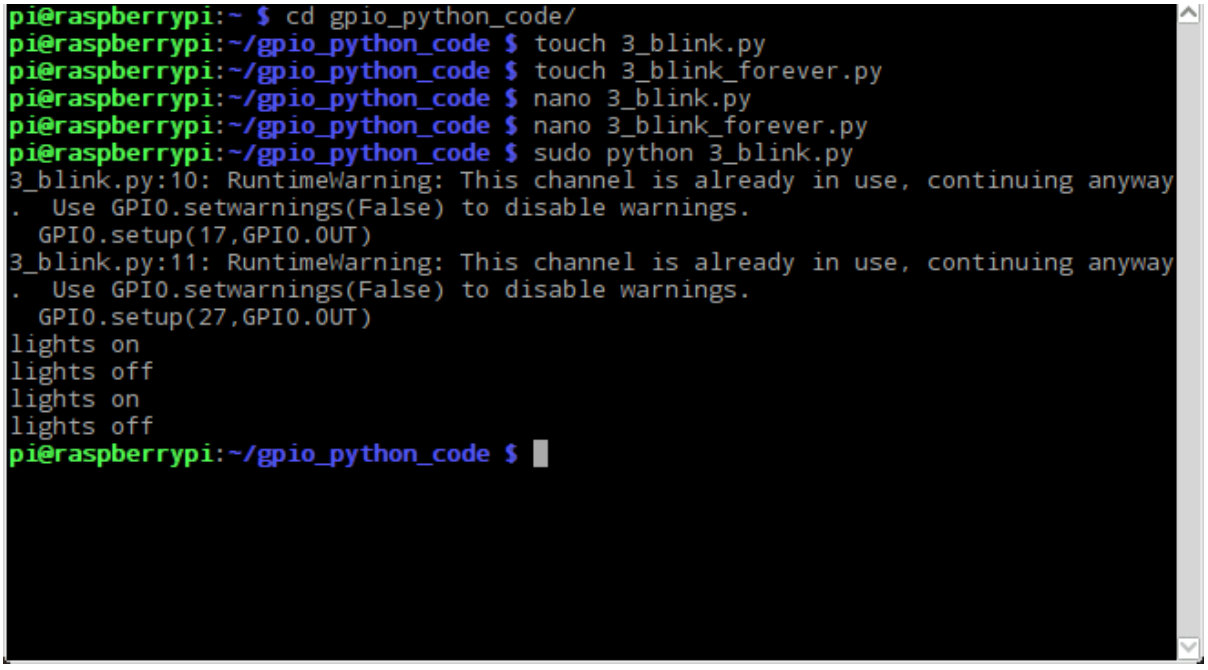
TASK 1: Write a python code for blinking LED using python

Code:

```
from time import sleep
import RPi.GPIO as
GPIO
GPIO.setmode(GPIO.B
CM)
GPIO.setup(17,GPIO.O
UT)
GPIO.setup(27,GPIO.O
UT)
print (“lights on”)
GPIO.output(17,GPIO.HI
GH)
GPIO.output(27,GPIO.HI
GH)
sleep(1)
print (“lights off”)
GPIO.output(17,GPIO.L
OW)
GPIO.output(27,GPIO.L
OW)
sleep(1)
print (“lights on”)
GPIO.output(17,GPIO.HI
GH)
GPIO.output(27,GPIO.HI
GH)
sleep(1)
print( “lights off”)
```

```
GPIO.output(17,GPIO.L
OW)
GPIO.output(27,GPIO.L
OW)
GPIO.cleanup()
```

Output:


pi@raspberrypi:~ $ cd gpio_python_code/
pi@raspberrypi:~/gpio_python_code $ touch 3_blink.py
pi@raspberrypi:~/gpio_python_code $ touch 3_blink_forever.py
pi@raspberrypi:~/gpio_python_code $ nano 3_blink.py
pi@raspberrypi:~/gpio_python_code $ nano 3_blink_forever.py
pi@raspberrypi:~/gpio_python_code $ sudo python 3_blink.py
3_blink.py:10: RuntimeWarning: This channel is already in use, continuing anyway
. Use GPIO.setwarnings(False) to disable warnings.
GPIO.setup(17,GPIO.OUT)
3_blink.py:11: RuntimeWarning: This channel is already in use, continuing anyway
. Use GPIO.setwarnings(False) to disable warnings.
GPIO.setup(27,GPIO.OUT)
lights on
lights off
lights on
lights off
pi@raspberrypi:~/gpio_python_code $

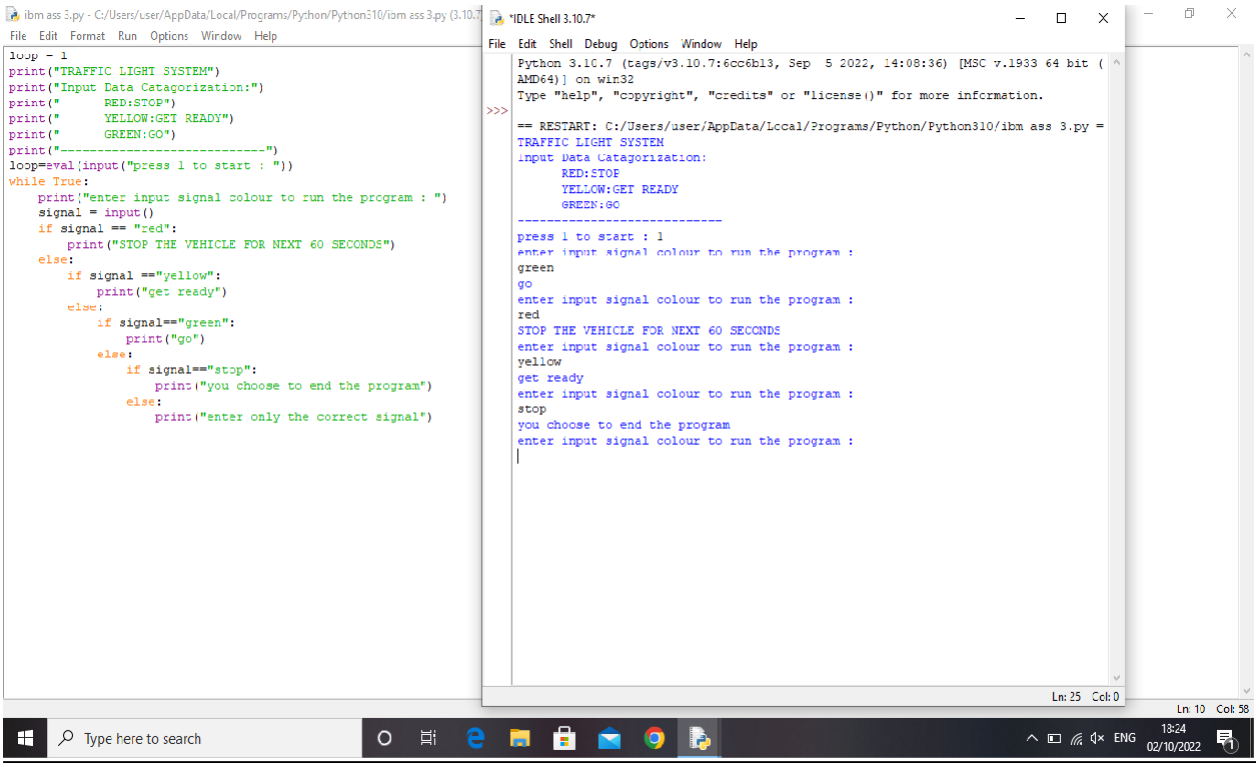
TASK 2:

Write a python code for traffic light system using python(should be communicatable with raspberrypi)

Code:

```
 loop =
    1

print("TRAFFIC LIGHT SYSTEM")

print("Input Data

Catagorization:") print("

             RED:STOP")

print("     YELLOW:GET

READY") print("

             GREEN:GO")

print("..............................-")

loop=eval(input("press 1 to start :

")) while True:

   print("enter input signal colour to run the program

   : ") signal = input()

   if signal == "red":

      print("STOP THE VEHICLE FOR NEXT 60 SECONDS")

   else:

      if signal =="yellow":

         print("get

      ready") else:

         if signal=="green":

            print("go")

         else:

            if signal=="stop":

               print("you choose to end the

            program") else:

               print("enter only the correct signal")
```

output:



Using RPi library:

```
import RPi.GPIO as GPIO
import  time
import signal
import sys
GPIO.setmode(GPIO.B
CM) GPIO.setup(9,
GPIO.OUT)
GPIO.setup(10,
GPIO.OUT)
GPIO.setup(11,
GPIO.OUT)
def allLightsOff(signal,
   frame): GPIO.output(9,
   False) GPIO.output(10,
   False) GPIO.output(11,
   False) GPIO.cleanup()
   sys.exit(0)
signal.signal(signal.SIGINT,
allLightsOff)
```

```
while True:
    GPIO.output(9,
    True) time.sleep(3)
    GPIO.output(10,
    True) time.sleep(1)
    GPIO.output(9,
    False)
    GPIO.output(10,
    False)
    GPIO.output(11,
    True) time.sleep(5)
    GPIO.output(11,
    False)
    GPIO.output(10,
    True) time.sleep(2)
    GPIO.output(10,
    False)
```