

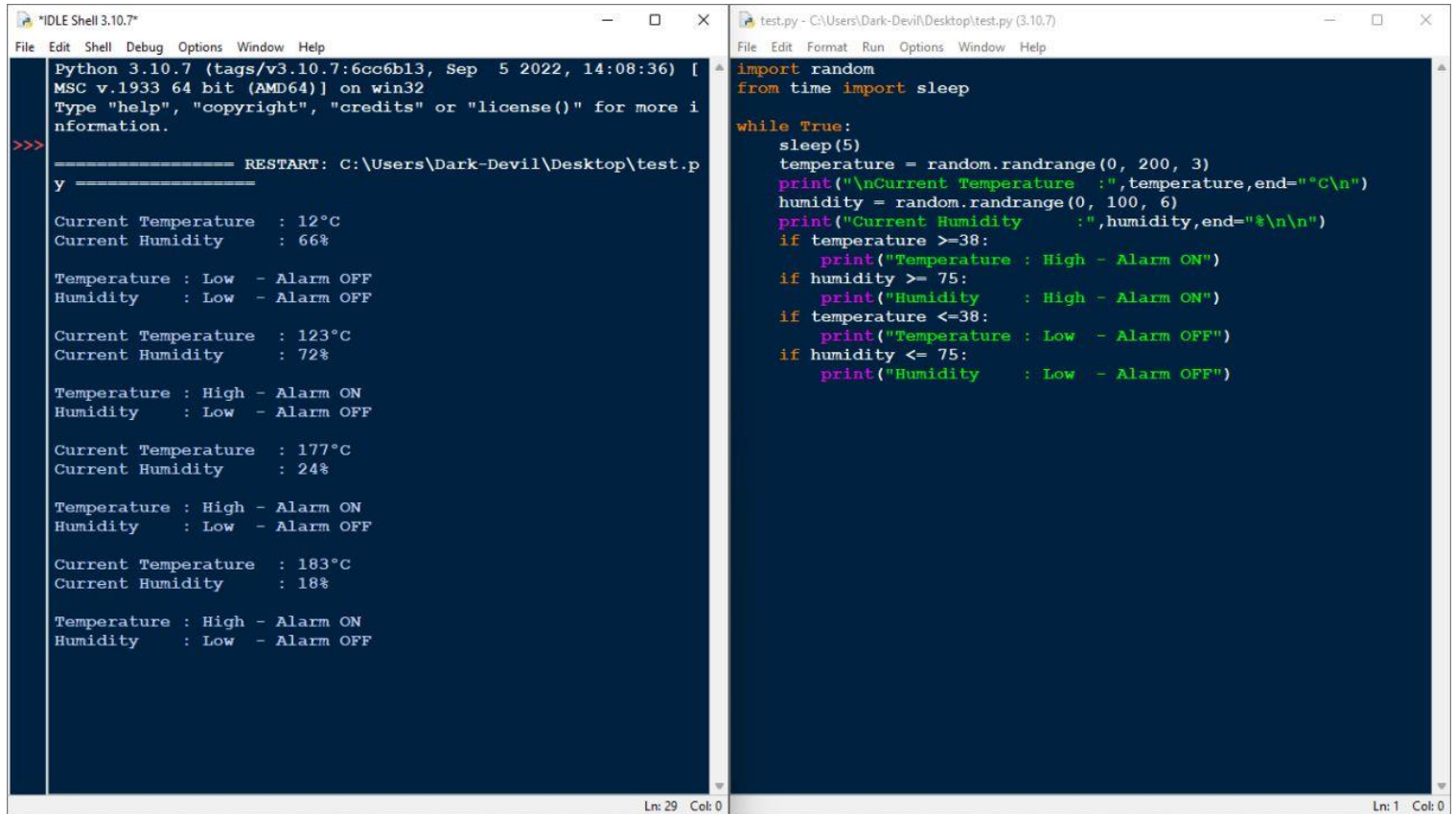
Configure The Mobile App For Controlling Motor Using Buttons

Date	17 November 2022
Team Id	PNT2022TMID28572
Title	Hazardous Area Monitoring for Industrial Plant using IoT

App Blocks to render the values and display it in app

The screenshot displays the MIT App Inventor web interface. The top navigation bar includes links for Projects, Connect, Build, Settings, and Help. The main workspace is divided into three panels: Blocks, Viewer, and Designer. The Blocks panel on the left lists various built-in components. The Viewer panel in the center shows two event-driven code blocks: 'when Button1.Click' followed by 'do' blocks containing 'set Web2.Url to' and 'call Web2.Get'. The URL is set to 'http://169.51.204.239:30739/control?command=alar...'. The Designer panel on the right shows a mobile app layout with a button and a text box. The bottom of the screen shows a Windows taskbar with various application icons and system status information.

Python block that changes the state of motor based on input from app



The image shows two side-by-side windows from a Python IDE. The left window, titled "IDLE Shell 3.10.7*", displays the output of a Python script. It shows a restart message, followed by three sets of sensor data (Current Temperature and Current Humidity) and their corresponding alarm states (Temperature: Low - Alarm OFF, Humidity: Low - Alarm OFF; Temperature: High - Alarm ON, Humidity: Low - Alarm OFF; Temperature: High - Alarm ON, Humidity: Low - Alarm OFF). The right window, titled "test.py - C:\Users\Dark-Devil\Desktop\test.py (3.10.7)", shows the source code of the script. The code imports random and time modules, and uses a while loop to generate random temperature and humidity values, printing them and checking for alarm conditions based on thresholds.

```
Python 3.10.7 (tags/v3.10.7:6cc6b13, Sep 5 2022, 14:08:36) [
MSC v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more i
nformation.
>>>
===== RESTART: C:\Users\Dark-Devil\Desktop\test.p
y =====
Current Temperature : 12°C
Current Humidity : 66%

Temperature : Low - Alarm OFF
Humidity : Low - Alarm OFF

Current Temperature : 123°C
Current Humidity : 72%

Temperature : High - Alarm ON
Humidity : Low - Alarm OFF

Current Temperature : 177°C
Current Humidity : 24%

Temperature : High - Alarm ON
Humidity : Low - Alarm OFF

Current Temperature : 183°C
Current Humidity : 18%

Temperature : High - Alarm ON
Humidity : Low - Alarm OFF

Ln: 29 Col: 0
```

```
test.py - C:\Users\Dark-Devil\Desktop\test.py (3.10.7)
File Edit Format Run Options Window Help
import random
from time import sleep

while True:
    sleep(5)
    temperature = random.randrange(0, 200, 3)
    print("\nCurrent Temperature :", temperature, end="°C\n")
    humidity = random.randrange(0, 100, 6)
    print("Current Humidity :", humidity, end="%\n\n")
    if temperature >= 38:
        print("Temperature : High - Alarm ON")
    if humidity >= 75:
        print("Humidity : High - Alarm ON")
    if temperature <= 38:
        print("Temperature : Low - Alarm OFF")
    if humidity <= 75:
        print("Humidity : Low - Alarm OFF")

Ln: 1 Col: 0
```