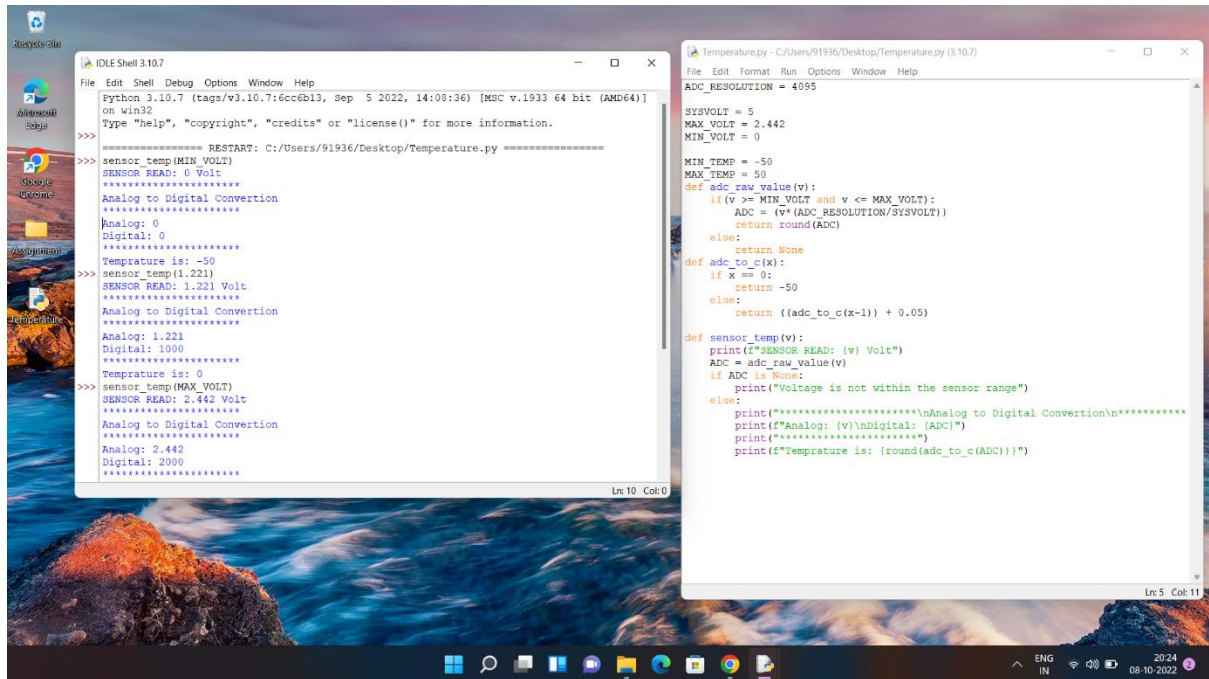


IBM ASSIGNMENT-2

BATCH-B104A6E IOT

Temperature.py



```
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 information.

>>> ===== RESTART: C:/Users/91936/Desktop/Temperature.py =====
>>> sensor_temp(MIN_VOLT)
SENSOR READ: 0 Volt
=====
Analog to Digital Conversion
=====
Analog: 0
Digital: 0
=====
Temperature is: -50
>>> sensor_temp(1.221)
SENSOR READ: 1.221 Volt
=====
Analog to Digital Conversion
=====
Analog: 1.221
Digital: 1000
=====
Temperature is: 0
>>> sensor_temp(MAX_VOLT)
SENSOR READ: 2.442 Volt
=====
Analog to Digital Conversion
=====
Analog: 2.442
Digital: 2000
=====

Ln:10 Col:0
```

```
Temperature.py - C:/Users/91936/Desktop/Temperature.py (3.10.7)
File Edit Format Run Options Window Help
ADC_RESOLUTION = 4095

SYSVOLT = 5
MAX_VOLT = 2.442
MIN_VOLT = 0

MIN_TEMP = -50
MAX_TEMP = 50
def adc_raw_value(v):
    if (v >= MIN_VOLT and v <= MAX_VOLT):
        ADC = (v*(ADC_RESOLUTION/SYSVOLT))
        return round(ADC)
    else:
        return None
def adc_to_c(x):
    if x == 0:
        return -50
    else:
        return ((adc_to_c(x-1)) + 0.05)

def sensor_temp(v):
    print(f"SENSOR READ: {v} Volt")
    ADC = adc_raw_value(v)
    if ADC is None:
        print("Voltage is not within the sensor range")
    else:
        print("=====Analog to Digital Conversion\n=====")
        print(f"Analog: {v}\nDigital: {ADC}")
        print("=====")
        print(f"Temperature is: {round(adc_to_c(ADC))}")

Ln:5 Col:11
```