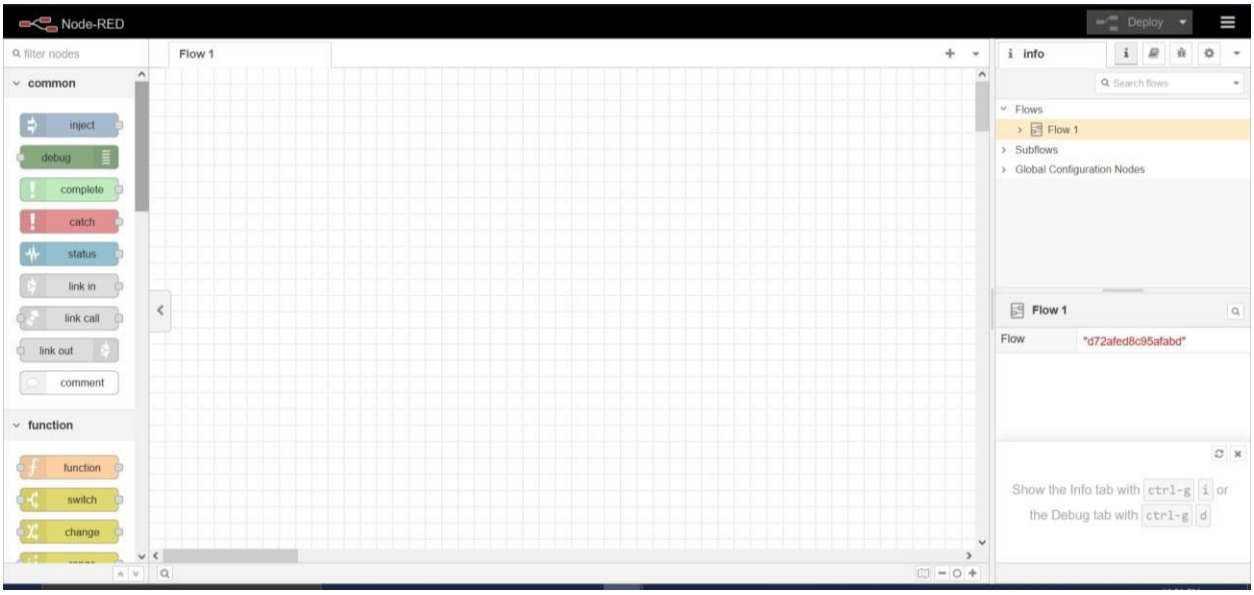
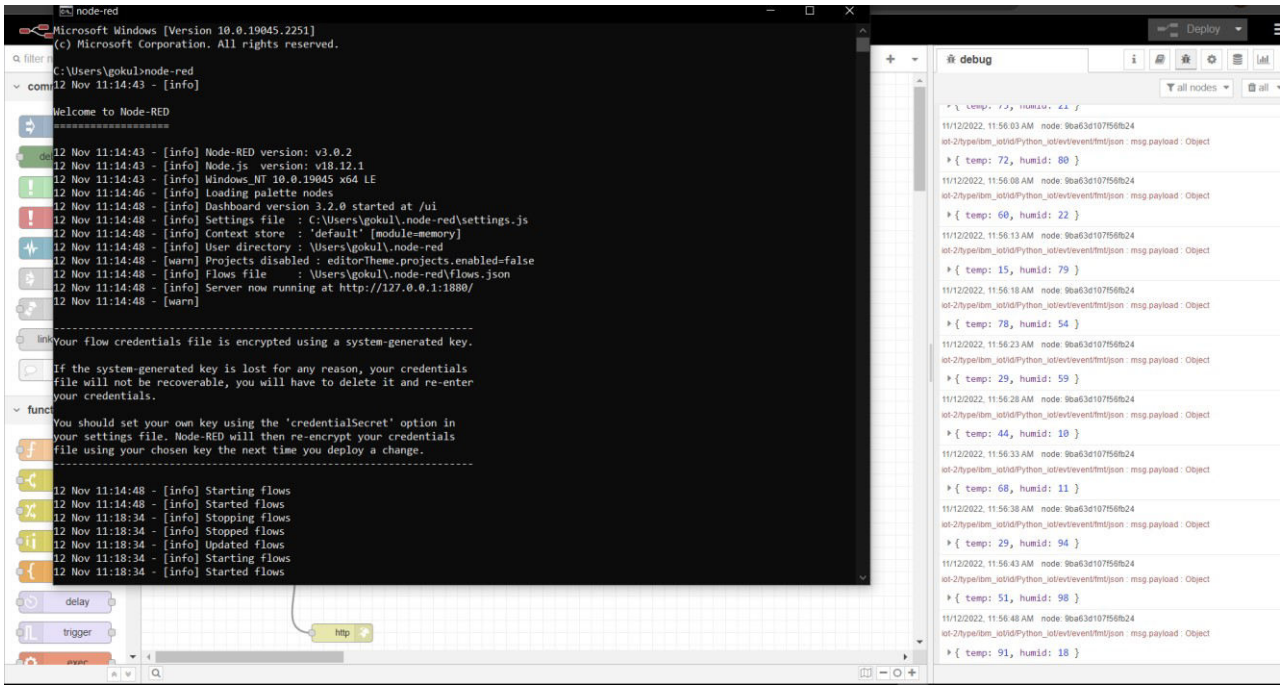
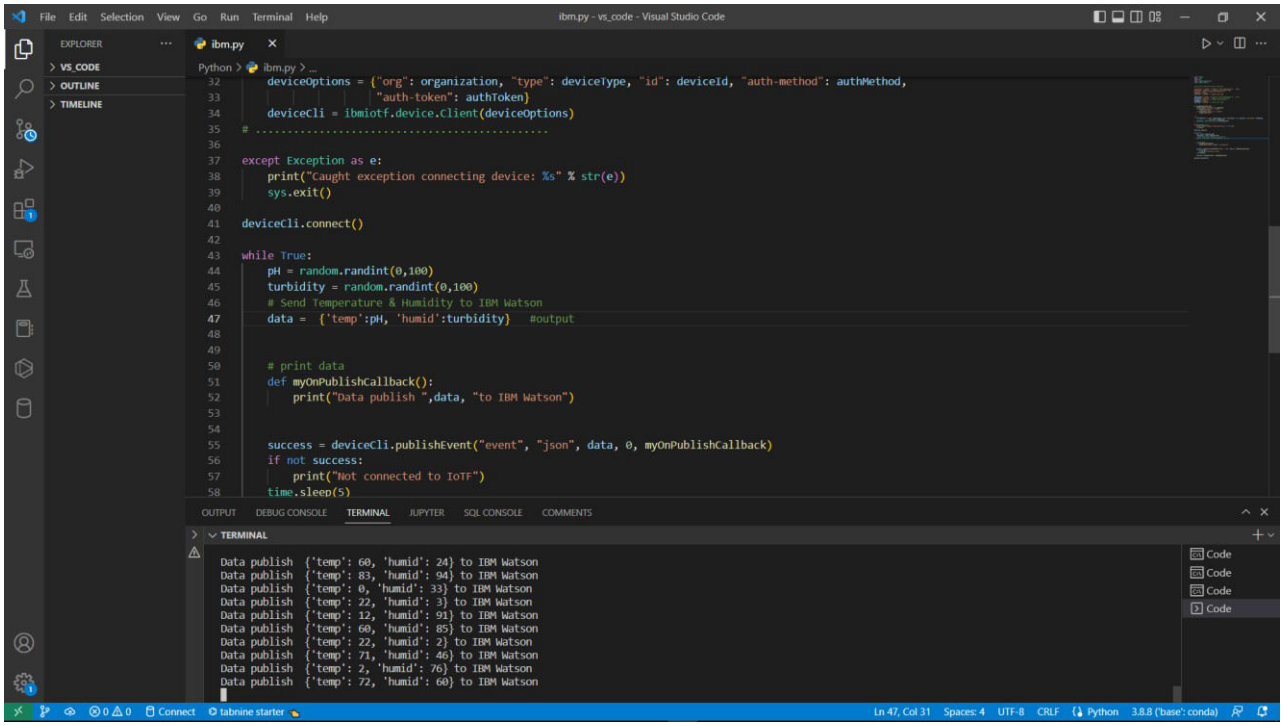
CREATION OF NODE-RED

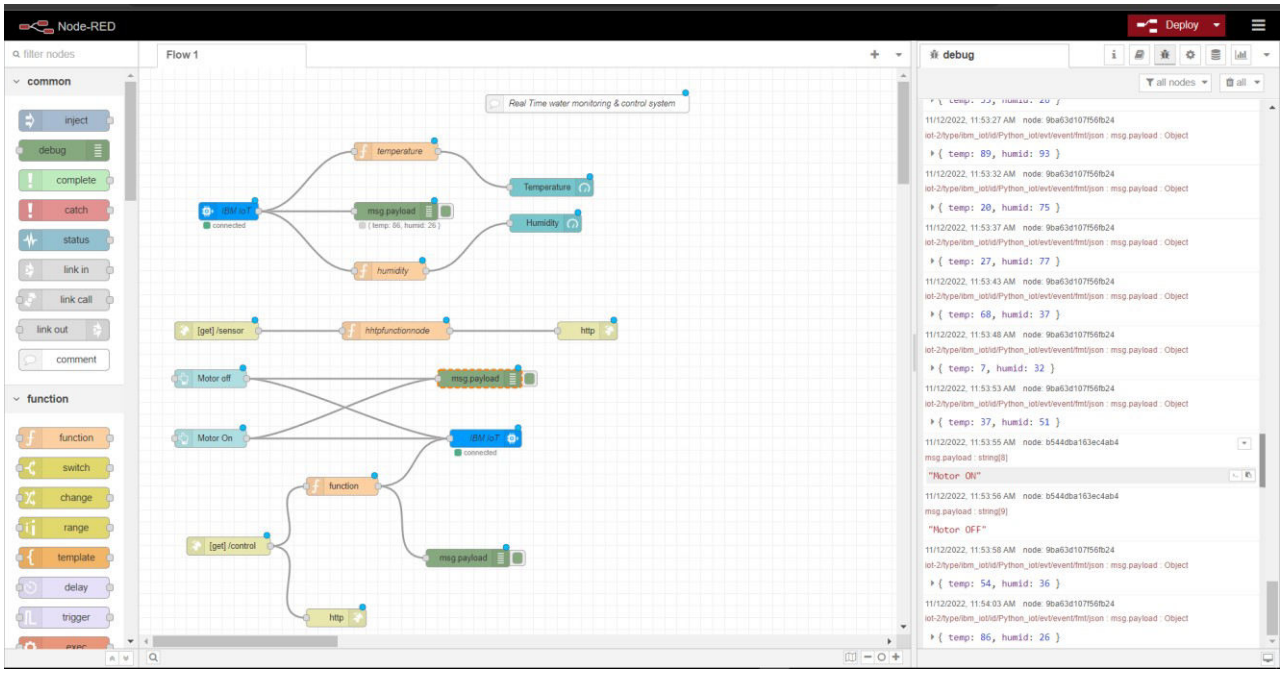
Date	16 November 2022
Team ID	PNT2022TMID36807
Project Name	Real – time River Water Quality Monitoring and Control System
Maximum Marks	4 Marks

Our task to create node-red is successfully done. A screenshot of this node-red screen is attached below:






File Edit Selection View Go Run Terminal Help
ibm.py - vs_code - Visual Studio Code
EXPLORER
VS_CODE
OUTLINE
TIMELINE
ibm.py
Python > ibm.py > ...
deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-method": authMethod,
"auth-token": authToken}
deviceCli = ibmiotf.device.Client(deviceOptions)
except Exception as e:
print("Caught exception connecting device: %s" % str(e))
sys.exit()
deviceCli.connect()
while True:
pH = random.randint(0,100)
turbidity = random.randint(0,100)
# Send Temperature & Humidity to IBM Watson
data = {'temp':pH, 'humid':turbidity} #output
# print data
def myOnPublishCallback():
print("Data publish ",data, "to IBM Watson")
success = deviceCli.publishEvent("event", "json", data, 0, myOnPublishCallback)
if not success:
print("Not connected to IoTF")
time.sleep(5)
OUTPUT DEBUG CONSOLE TERMINAL JUPYTER SQL CONSOLE COMMENTS
TERMINAL
Data publish {'temp': 60, 'humid': 24} to IBM Watson
Data publish {'temp': 83, 'humid': 94} to IBM Watson
Data publish {'temp': 0, 'humid': 33} to IBM Watson
Data publish {'temp': 22, 'humid': 3} to IBM Watson
Data publish {'temp': 12, 'humid': 91} to IBM Watson
Data publish {'temp': 60, 'humid': 85} to IBM Watson
Data publish {'temp': 22, 'humid': 2} to IBM Watson
Data publish {'temp': 71, 'humid': 46} to IBM Watson
Data publish {'temp': 2, 'humid': 76} to IBM Watson
Data publish {'temp': 72, 'humid': 60} to IBM Watson
Code
Code
Code
Code
Ln 47, Col 31 Spaces: 4 UTF-8 CRLF Python 3.8.8 ('base': conda)


Node-RED
Deploy
filter nodes
Flow 1
common
inject
debug
complete
catch
status
link in
link call
link out
comment
function
function
switch
change
range
template
delay
trigger
Real Time water monitoring & control system
temperature
Temperature
msg.payload
{ temp: 86, humid: 26 }
Humidity
humidity
[get] /sensor
hhtpfunctionnode
http
Motor off
msg.payload
Motor On
IBM IoT
connected
function
[get] /control
msg.payload
http
debug
all nodes
all
11/12/2022, 11:53:27 AM node: 9ba63d107f56fb24
iot-2/type/ibm_iot/id/Python_iot/evt/event/fmt/json : msg.payload : Object
{ temp: 89, humid: 93 }
11/12/2022, 11:53:32 AM node: 9ba63d107f56fb24
iot-2/type/ibm_iot/id/Python_iot/evt/event/fmt/json : msg.payload : Object
{ temp: 20, humid: 75 }
11/12/2022, 11:53:37 AM node: 9ba63d107f56fb24
iot-2/type/ibm_iot/id/Python_iot/evt/event/fmt/json : msg.payload : Object
{ temp: 27, humid: 77 }
11/12/2022, 11:53:43 AM node: 9ba63d107f56fb24
iot-2/type/ibm_iot/id/Python_iot/evt/event/fmt/json : msg.payload : Object
{ temp: 68, humid: 37 }
11/12/2022, 11:53:48 AM node: 9ba63d107f56fb24
iot-2/type/ibm_iot/id/Python_iot/evt/event/fmt/json : msg.payload : Object
{ temp: 7, humid: 32 }
11/12/2022, 11:53:53 AM node: 9ba63d107f56fb24
iot-2/type/ibm_iot/id/Python_iot/evt/event/fmt/json : msg.payload : Object
{ temp: 37, humid: 51 }
11/12/2022, 11:53:55 AM node: b544dba163ec4ab4
msg.payload : string[8]
"Motor ON"
11/12/2022, 11:53:56 AM node: b544dba163ec4ab4
msg.payload : string[9]
"Motor OFF"
11/12/2022, 11:53:58 AM node: 9ba63d107f56fb24
iot-2/type/ibm_iot/id/Python_iot/evt/event/fmt/json : msg.payload : Object
{ temp: 54, humid: 36 }
11/12/2022, 11:54:03 AM node: 9ba63d107f56fb24
iot-2/type/ibm_iot/id/Python_iot/evt/event/fmt/json : msg.payload : Object
{ temp: 86, humid: 26 }