Challenging Task – 3

Name: Gangireddy Madhurima
Reg Number: 21MIS1155

Question: Predict linear Regression line for Light Intensity using a LDR Sensor: Use an LDR(Light Dependency Resistor) sensor to monitor light intensity and predict values using regression

Aim:

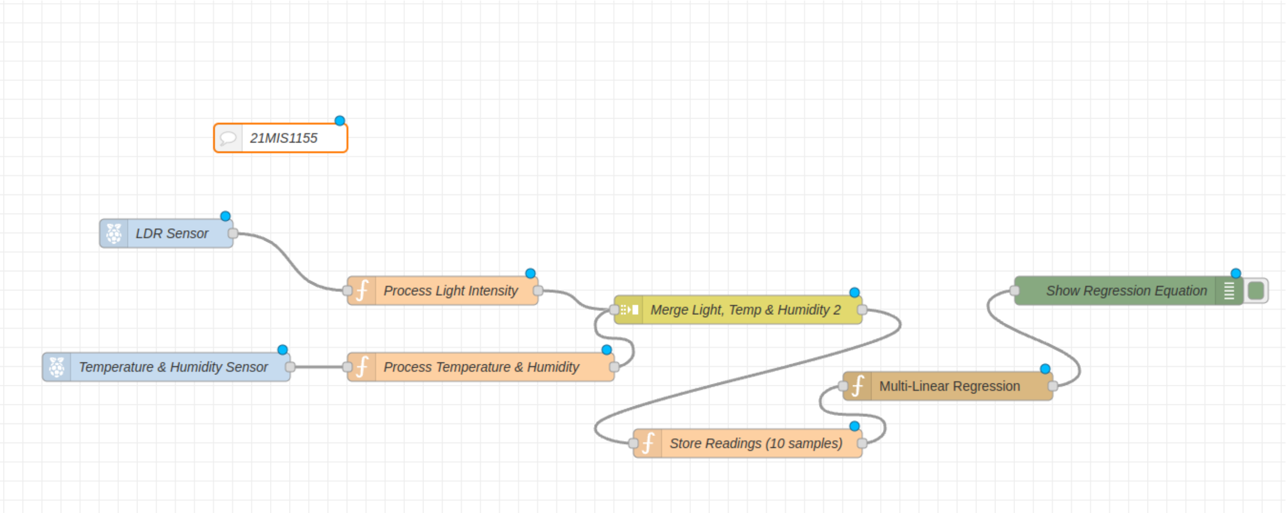
To monitor the light intensity using the **MQ-135** gas sensor along with **temperature** and **humidity** values from a **DHT22** sensor. The data will be processed in **Node-RED** to create a **linear regression model** that relates the Light Intensity to temperature and humidity, obtaining a regression equation.

Procedure:

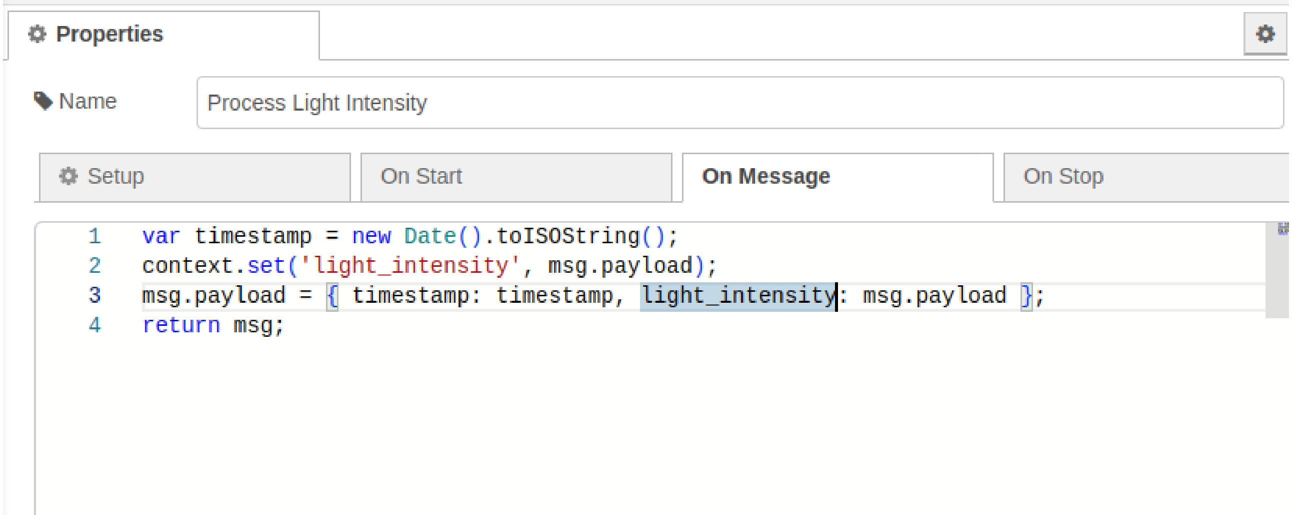
1. **Hardware Setup:**
   - **LDR Sensor**: Connect the **VCC** pin to 5V, **GND** pin to ground, and the **analog output** (A0) to the Raspberry Pi ADC (e.g., MCP3008) to get the Light Intensity data.
   - **DHT22 Temperature and Humidity Sensor**: Connect the **VCC** pin to 3.3V, **GND** to ground, and the **data pin** to a GPIO pin (e.g., GPIO 17) for reading temperature and humidity values.
2. **Node-RED Flow:**
   - Use the **GPIO Input Node** for reading data from the **LDR** sensor and **DHT22** sensor.
   - Process the data using **function nodes** to format the readings into a structured message.
   - Use a **Regression Node** to train a linear regression model with the light intensity level, temperature, and humidity values.

- The regression output will generate a prediction of the light intensity level based on temperature and humidity.
- Output the regression results using a **Debug Node**.

Node-RED Set Up:

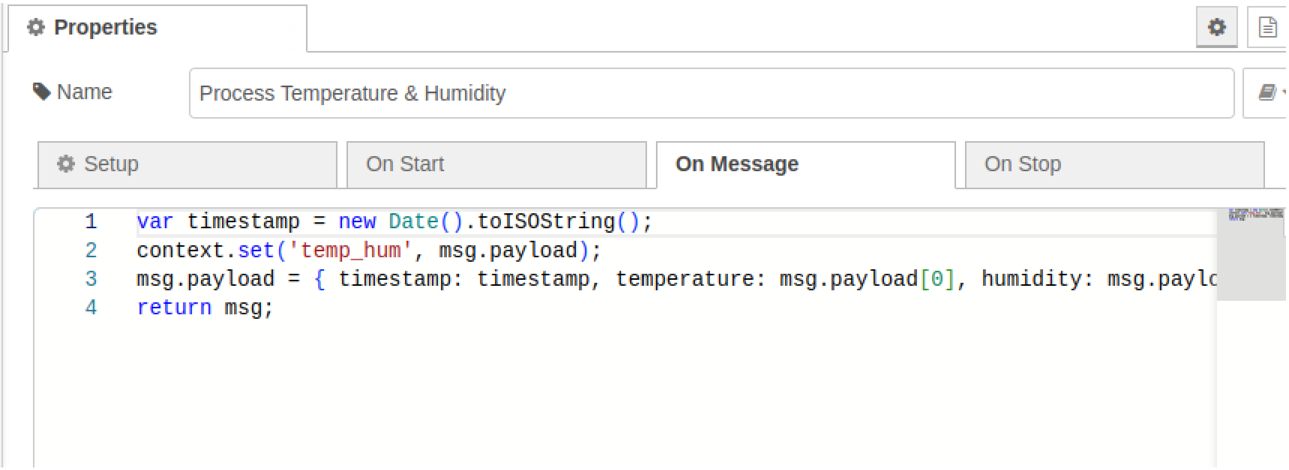

21MIS1155
LDR Sensor
Process Light Intensity
Merge Light, Temp & Humidity 2
Show Regression Equation
Temperature & Humidity Sensor
Process Temperature & Humidity
Multi-Linear Regression
Store Readings (10 samples)

Functions:



Name: Process Light Intensity

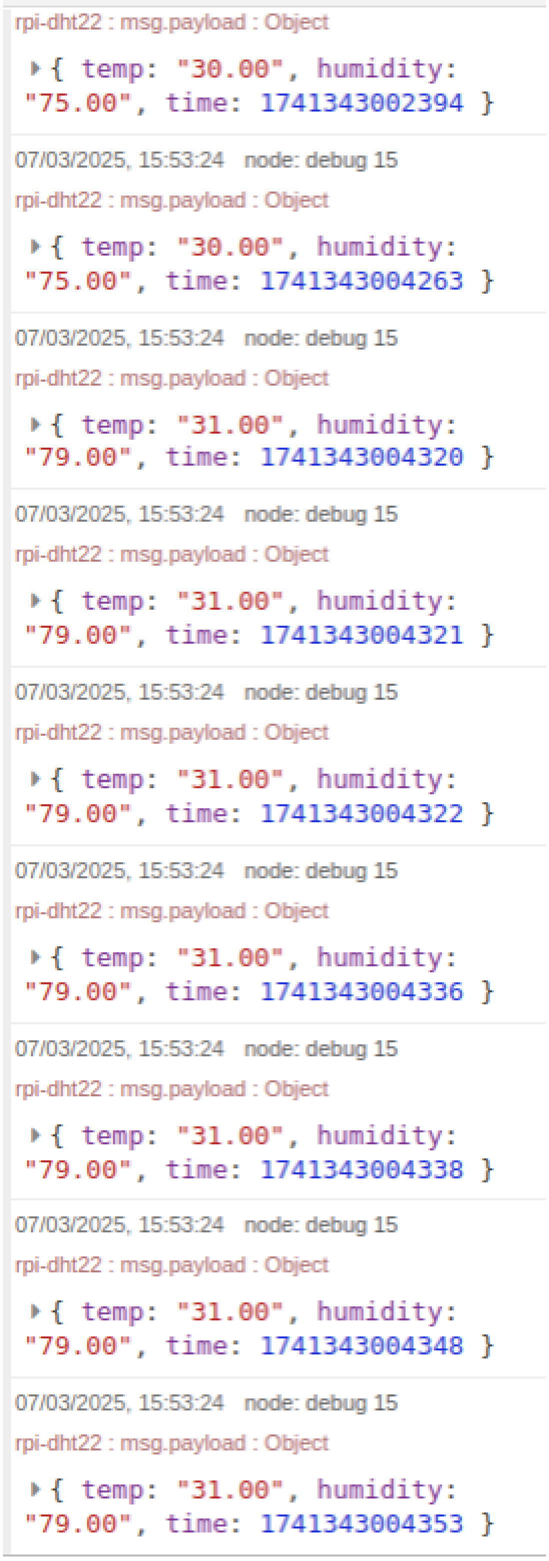
```
var timestamp = new Date().toISOString();
context.set('light_intensity', msg.payload);
msg.payload = { timestamp: timestamp, light_intensity: msg.payload };
return msg;
```



Name: Process Temperature & Humidity

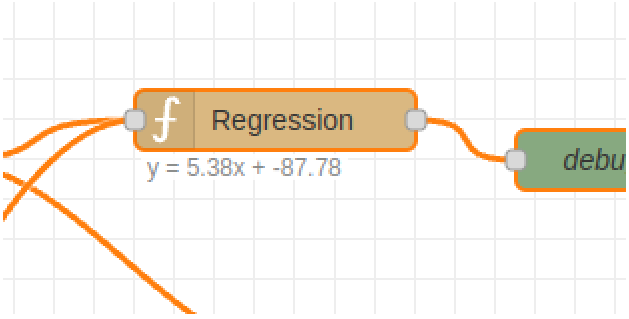
```
var timestamp = new Date().toISOString();
context.set('temp_hum', msg.payload);
msg.payload = { timestamp: timestamp, temperature: msg.payload[0], humidity: msg.paylo
return msg;
```

Output:

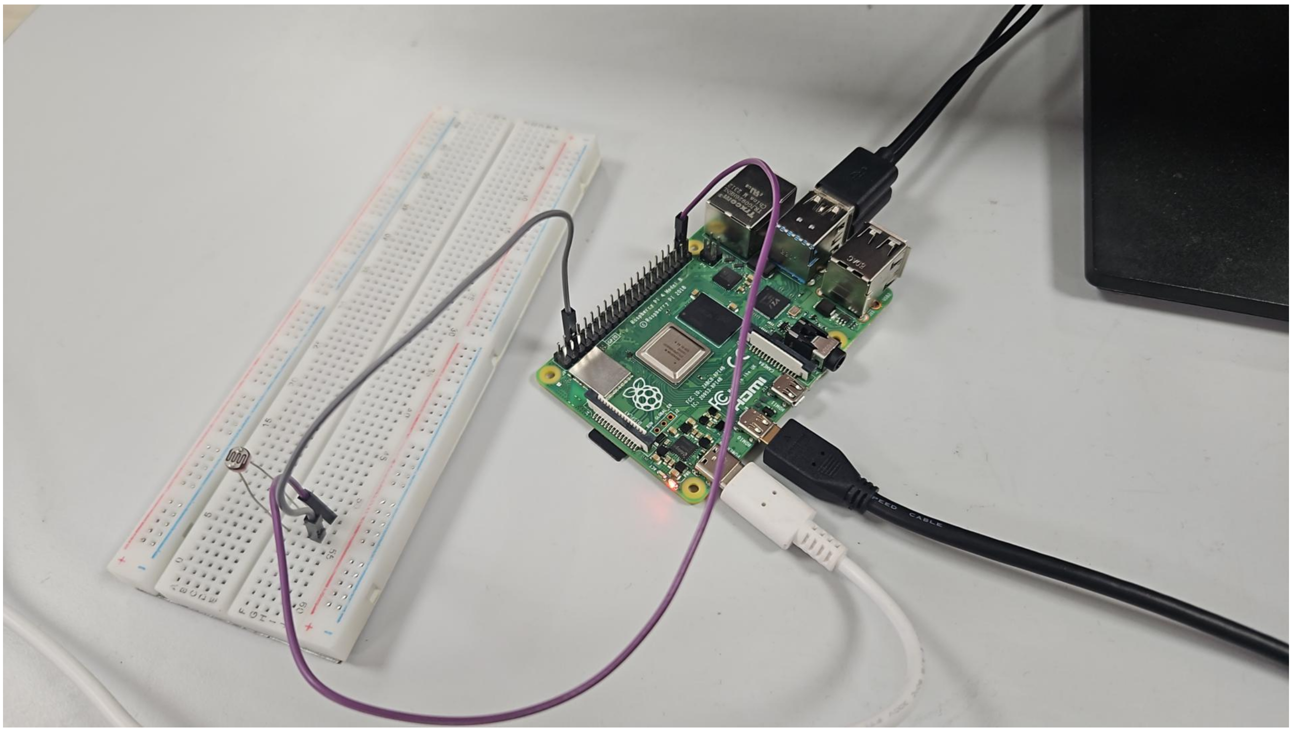


```
rpi-dht22 : msg.payload : Object
 ▸{ temp: "30.00", humidity:
"75.00", time: 1741343002394 }

07/03/2025, 15:53:24   node: debug 15
rpi-dht22 : msg.payload : Object
 ▸{ temp: "30.00", humidity:
"75.00", time: 1741343004263 }

07/03/2025, 15:53:24   node: debug 15
rpi-dht22 : msg.payload : Object
 ▸{ temp: "31.00", humidity:
"79.00", time: 1741343004320 }

07/03/2025, 15:53:24   node: debug 15
rpi-dht22 : msg.payload : Object
 ▸{ temp: "31.00", humidity:
"79.00", time: 1741343004321 }

07/03/2025, 15:53:24   node: debug 15
rpi-dht22 : msg.payload : Object
 ▸{ temp: "31.00", humidity:
"79.00", time: 1741343004322 }

07/03/2025, 15:53:24   node: debug 15
rpi-dht22 : msg.payload : Object
 ▸{ temp: "31.00", humidity:
"79.00", time: 1741343004336 }

07/03/2025, 15:53:24   node: debug 15
rpi-dht22 : msg.payload : Object
 ▸{ temp: "31.00", humidity:
"79.00", time: 1741343004338 }

07/03/2025, 15:53:24   node: debug 15
rpi-dht22 : msg.payload : Object
 ▸{ temp: "31.00", humidity:
"79.00", time: 1741343004348 }

07/03/2025, 15:53:24   node: debug 15
rpi-dht22 : msg.payload : Object
 ▸{ temp: "31.00", humidity:
"79.00", time: 1741343004353 }
```

Regression Equation:



Hardware Set Up:



JSON Code:

```
[
  {
    "id": "e0b26ac922a1509b",
    "type": "tab",
    "label": "Flow 5",
    "disabled": false,
    "info": ""
  },
  {
    "id": "e98855bf6b67711e",
    "type": "rpi-gpio in",
    "z": "e0b26ac922a1509b",
    "name": "LDR Sensor",
    "pin": "21",
    "intype": "tri",
    "debounce": "25",
        "read": true,
        "bcm": true,
        "x": 190,
        "y": 280,
        "wires": [
            [
                "84887e5fd3a4d274"
            ]
        ]
    },
    {
        "id": "afc7f69afc894232",
        "type": "rpi-gpio in",
        "z": "e0b26ac922a1509b",
        "name": "Temperature & Humidity Sensor",
        "pin": "26",
        "intype": "tri",
        "debounce": "25",
        "read": true,
        "bcm": true,
        "x": 190,
        "y": 420,
        "wires": [
            [
                "98b6ad9a1b16c1a1"
            ]
        ]
    },
    {
        "id": "84887e5fd3a4d274",
        "type": "function",
        "z": "e0b26ac922a1509b",
        "name": "Process Light Intensity",
        "func": "var timestamp = new Date().toISOString();\ncontext.set('light_intensity', msg.payload);\nmsg.payload = { timestamp: timestamp, light_intensity: msg.payload };\nreturn msg;",
        "outputs": 1,
        "timeout": "",
        "noerr": 0,
        "initialize": "",
        "finalize": "",
        "libs": [],
        "x": 480,
        "y": 340,
        "wires": [
            [
                "5fb47a9f9d05fb60"
            ]
        ]
    },
    {
        "id": "98b6ad9a1b16c1a1",
        "type": "function",
        "z": "e0b26ac922a1509b",
        "name": "Process Temperature & Humidity",
        "func": "var timestamp = new Date().toISOString();\ncontext.set('temp_hum', msg.payload);\nmsg.payload = { timestamp: timestamp, temperature: msg.payload[0], humidity: msg.payload[1] };\nreturn msg;",
        "outputs": 1,
        "timeout": "",
        "noerr": 0,
        "initialize": "",
        "finalize": "",
        "libs": [],
        "x": 520,
        "y": 420,
        "wires": [
            [
                "5fb47a9f9d05fb60"
            ]
        ]
    },
    {
        "id": "5fb47a9f9d05fb60",
        "type": "join",
        "z": "e0b26ac922a1509b",
        "name": "Merge Light, Temp & Humidity",
        "mode": "custom",
        "build": "array",
        "property": "payload",
        "propertyType": "msg",
        "key": "topic",
        "joiner": ",",
        "joinerType": "str",
        "useparts": true,
        "accumulate": false,
        "timeout": "2",
        "count": "2",
        "reduceRight": false,
        "reduceExp": "",
        "reduceInit": "",
        "reduceInitType": "",
        "reduceFixup": "",
        "x": 790,
        "y": 360,
        "wires": [
            [
                "122aaeffd6a15e4e"
            ]
        ]
    },
    {
        "id": "122aaeffd6a15e4e",
        "type": "function",
        "z": "e0b26ac922a1509b",
        "name": "Store Readings (10 samples)",
        "func": "var readings = context.get('readings') || [];\nvar data = {\n    gas_level: msg.payload[0].gas_level,\n    temperature: msg.payload[1].temperature,\n    humidity: msg.payload[1].humidity\n};\nreadings.push(data);\nif (readings.length > 10) readings.shift();\ncontext.set('readings', readings);\nmsg.payload = readings;\nreturn msg;",
        "outputs": 1,
        "timeout": "",
        "noerr": 0,
        "initialize": "",
        "finalize": "",
        "libs": [],
        "x": 800,
        "y": 500,
        "wires": [
            [
                "2ed4114833c07b5e"
            ]
        ]
    },
    {
        "id": "2ed4114833c07b5e",
        "type": "regression",
        "z": "e0b26ac922a1509b",
        "name": "Multi-Linear Regression",
        "dataSetSize": 10,
        "regressionType": "polynomial",
        "polynomialOrder": "2",
        "precision": "2",
        "xInputField": "payload[*].gas_level",
        "xInputFieldType": "msg",
        "yInputField": "payload[*].temperature",
        "yInputFieldType": "flow",
        "yOutputField": "payload.result",
        "yOutputFieldType": "msg",
        "functionOutputField": "payload.equation",
        "functionOutputFieldType": "msg",
        "resultOnly": true,
        "x": 1010,
        "y": 440,
        "wires": [
            [
                "ba758aa3997232b7"
            ]
        ]
    },
    {
        "id": "ba758aa3997232b7",
        "type": "debug",
        "z": "e0b26ac922a1509b",
        "name": "Show Regression Equation",
        "active": true,
        "tosidebar": true,
        "console": false,
        "tostatus": false,
        "complete": "payload",
        "statusVal": "",
        "statusType": "auto",
        "x": 1200,
        "y": 340,
        "wires": []
    },
    {
        "id": "674d71d846d2ee29",
        "type": "comment",
        "z": "e0b26ac922a1509b",
        "name": "21MIS1155",
        "info": "",
        "x": 310,
        "y": 180,
        "wires": []
    }
]
```