

Assignment-3

Name: Manasa.R

Question:

In wokwi add LED and switch on and off from node-red

Wokwi:

Code:

```
#include "WiFi.h"
#include "PubSubClient.h"

int led_pin = 5;
char clientId[50];

WiFiClient espClient;
PubSubClient client(espClient);

void setup()
{
    Serial.begin(115200);
    pinMode(led_pin, OUTPUT);

    WiFi.mode(WIFI_STA);
    WiFi.begin("Wokwi-GUEST", "");

    client.setServer("mqtt-dashboard.com", 1883);
    client.setCallback(callback);
    mqttReconnect();
}

void mqttReconnect()
{
    while (!client.connected())
    {
        Serial.print("Attempting MQTT connection...");
        if (client.connect(clientId))
        {

```

```

        Serial.println("Connected");
        client.subscribe("Light");
    }
    else
    {
        Serial.println("Connection failed try again in 5 seconds");
        delay(5000);
    }
}
}

```

```

void callback(char* topic, byte* message, unsigned int length)

```

```

{
    String stMessage;
    for (int i = 0; i < length; i++)
    {
        stMessage += (char)message[i];
    }
    if (stMessage == "OFF")
    {
        digitalWrite(led_pin, LOW);
        Serial.print("LIGHT OFF!!\n");
    }
    else if (stMessage == "ON")
    {
        digitalWrite(led_pin, HIGH);
        Serial.print("LIGHT ON!!\n");
    }
    else
    {
        Serial.print("INVALID COMMAND!!");
    }
}

```

```

void loop()

```

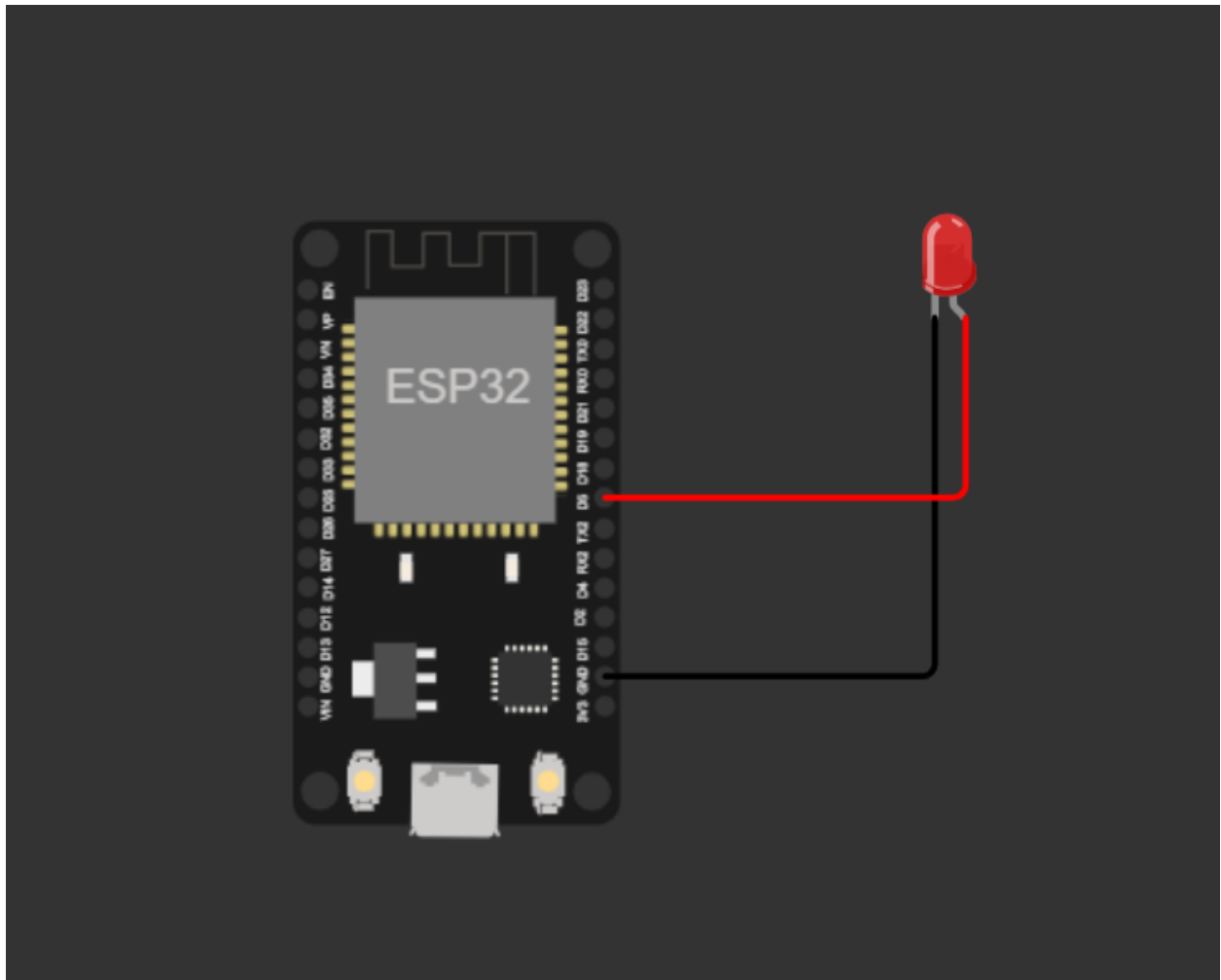
```

{
    if (!client.connected())
    {
        delay(10000);
        mqttReconnect();
    }
}

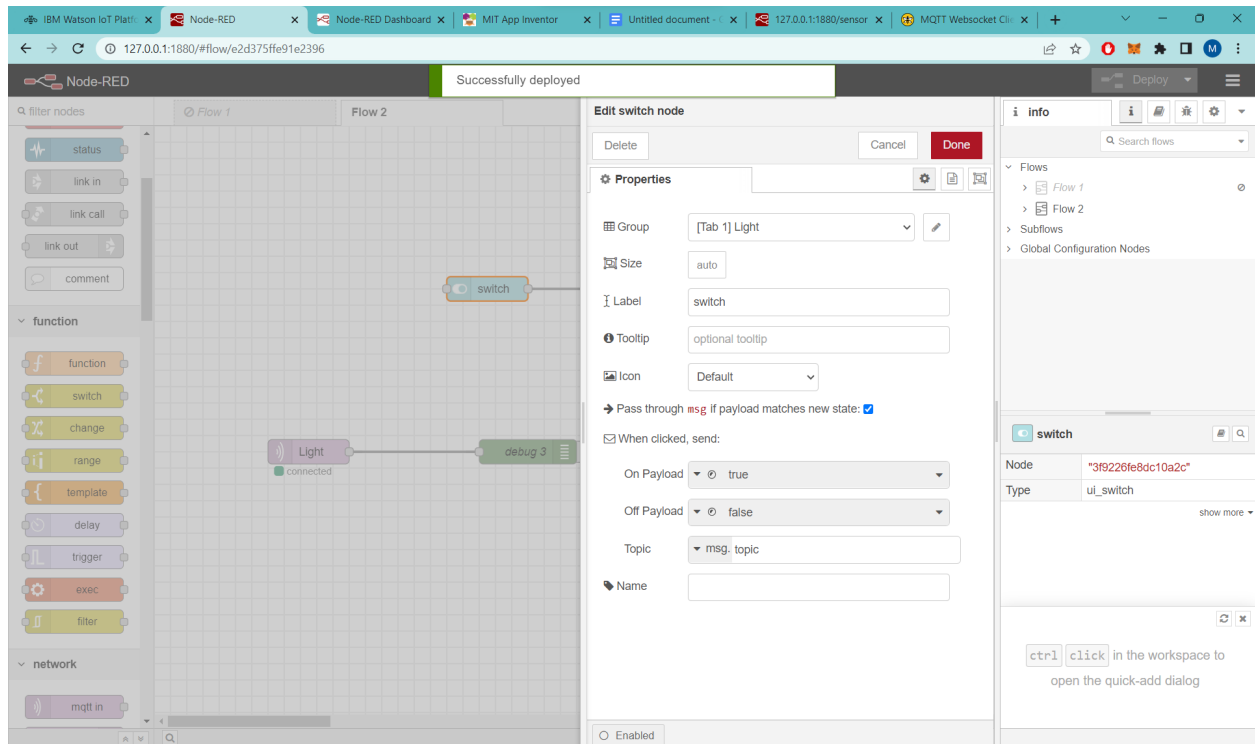
```

```
}  
client.loop();  
}
```

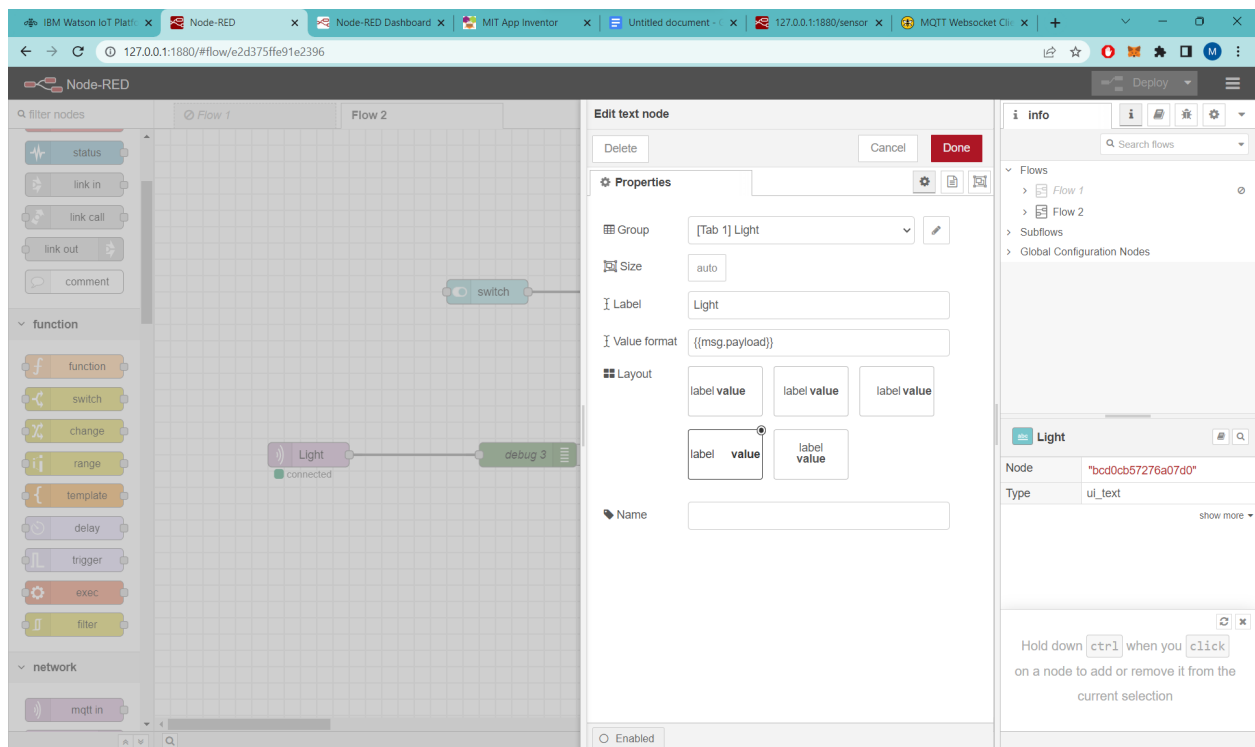
Circuit:



Node red:
Flow:



Text:



Mqtt out:

The screenshot shows the Node-RED web interface in a browser. The main workspace contains a flow with a 'switch' node connected to a 'Light' node, which is then connected to a 'debug 3' node. The 'Light' node is marked as 'connected'. The left sidebar shows various node categories: 'filter nodes', 'function', and 'network'. The 'mqtt in' node is visible in the 'network' category. The right sidebar shows the 'info' panel with a search bar and a list of flows. The 'Edit mqtt out node' panel is open, displaying the following properties:

- Server: Wokwi
- Topic: Light
- QoS: 0
- Retain: ☐
- Name: Name

A tip message is displayed: "Tip: Leave topic, qos or retain blank if you want to set them via msg properties." The 'Done' button is highlighted in red.

Mqtt in :

The screenshot shows the Node-RED web interface in a browser. The main workspace contains a flow with a 'switch' node connected to a 'Light' node, which is then connected to a 'debug 3' node. The 'Light' node is marked as 'connected'. The left sidebar shows various node categories: 'filter nodes', 'function', and 'network'. The 'mqtt in' node is visible in the 'network' category. The right sidebar shows the 'info' panel with a search bar and a list of flows. The 'Edit mqtt in node' panel is open, displaying the following properties:

- Server: Wokwi
- Action: Subscribe to single topic
- Topic: Light
- QoS: 0
- Output: auto-detect (parsed JSON object, string or buf)
- Name: Name

A message is displayed: "You can confirm your changes in the node edit tray with **ctrl-enter** or cancel them with **ctrl-escape**". The 'Done' button is highlighted in red.

debug:

The screenshot shows the Node-RED web interface in a browser. The main workspace contains a flow with a 'Light' node connected to a 'debug 3' node. The left sidebar shows various node categories: function, network, and others. The right sidebar is open to the 'Edit debug node' configuration for 'debug 3'. The configuration includes:

- Output:** msg.payload
- To:** ☒ debug window, ☐ system console, ☐ node status (32 characters)
- Name:** debug 3

Below the configuration, there is a section for 'debug 3' showing the node ID '6de3d649e42a3259' and type 'debug'. A note at the bottom states: 'You can confirm your changes in the node edit tray with **ctrl+enter** or cancel them with **ctrl+escape**'.

Hivemq:

The screenshot shows the Hivemq MQTT client interface. The top section is the 'Connection' panel, which is currently 'connected'. It displays the following fields:

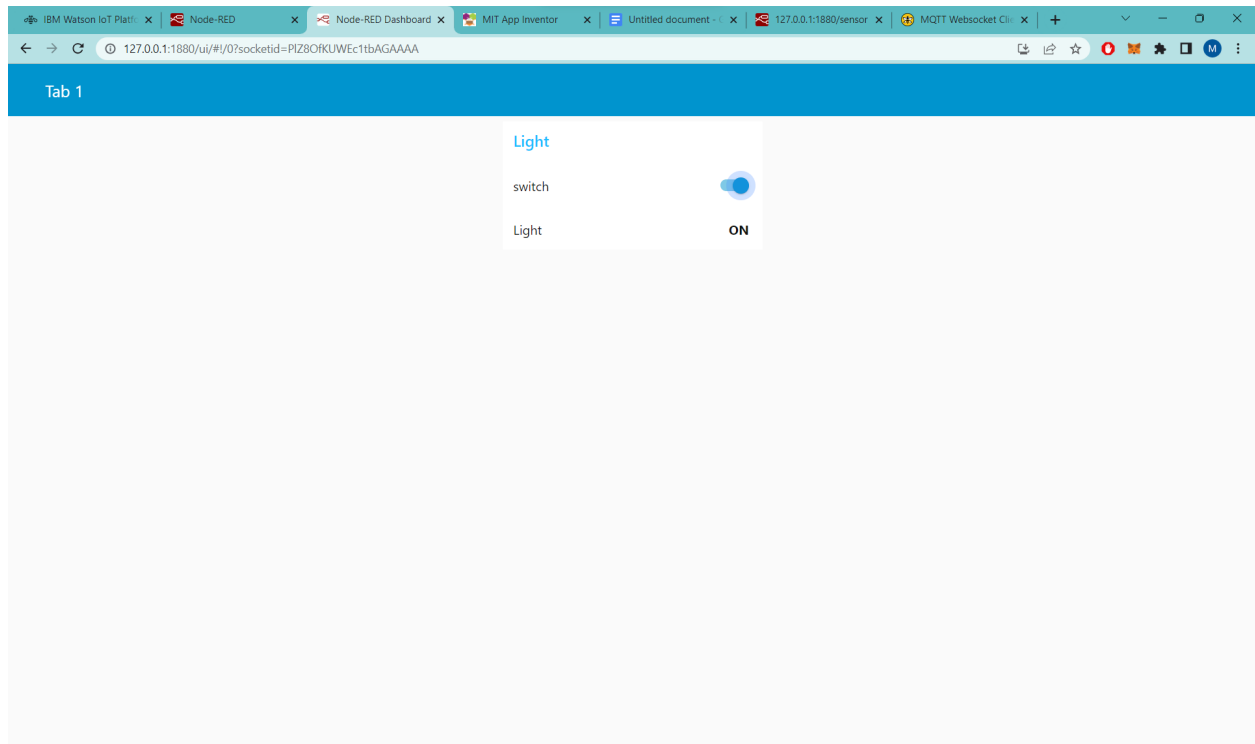
- Host: mqtt-dashboard.com
- Port: 8884
- ClientID: clientid-VE8MDIWMcr
- Username: (empty)
- Password: (empty)
- Keep Alive: 60
- SSL: ☒
- Clean Session: ☒
- Last-Will Topic: (empty)
- Last-Will QoS: 0
- Last-Will Retain: ☐
- Last-Will Message: (empty)

Below the connection panel are three other panels:

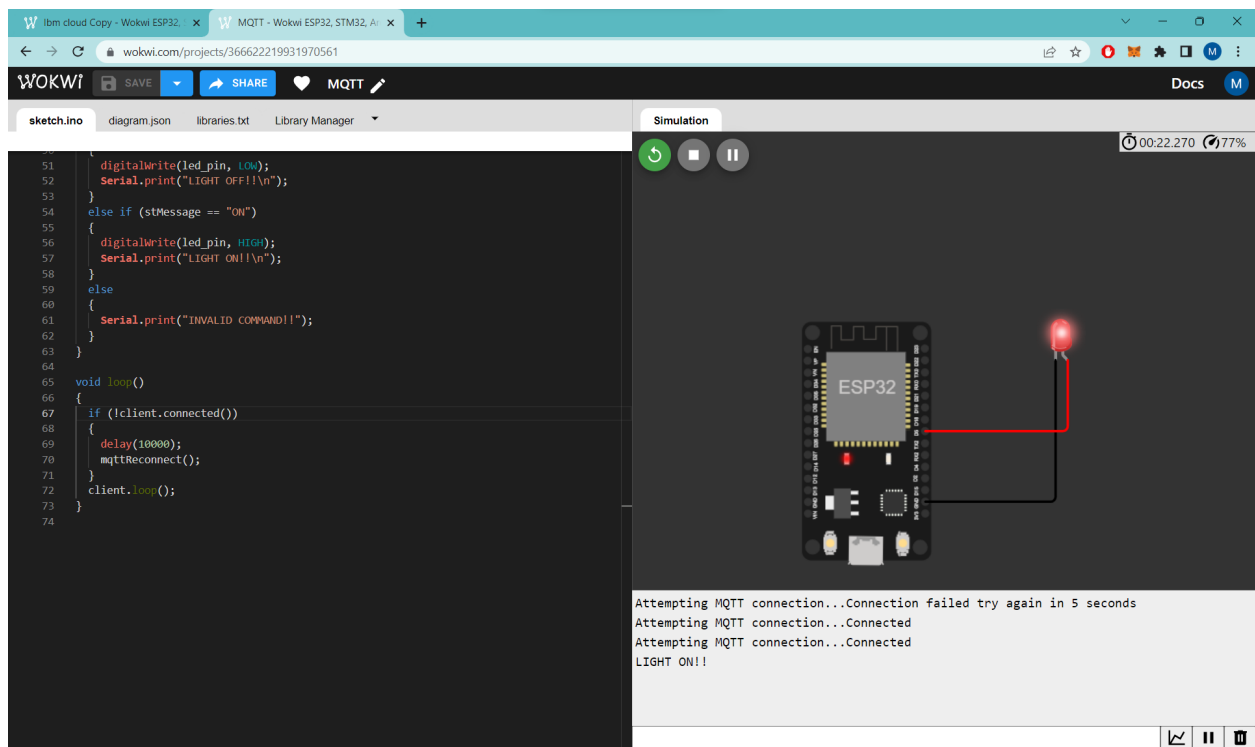
- Publish:** Topic: testtopic/1, QoS: 0, Retain: ☐. A 'Publish' button is visible.
- Subscriptions:** A list of subscriptions is shown, including 'Light' with QoS: 2.
- Messages:** A list of received messages is shown, including a message 'ON' received on '2023-06-08 19:04:57' on the topic 'Light' with QoS: 0.

Output:

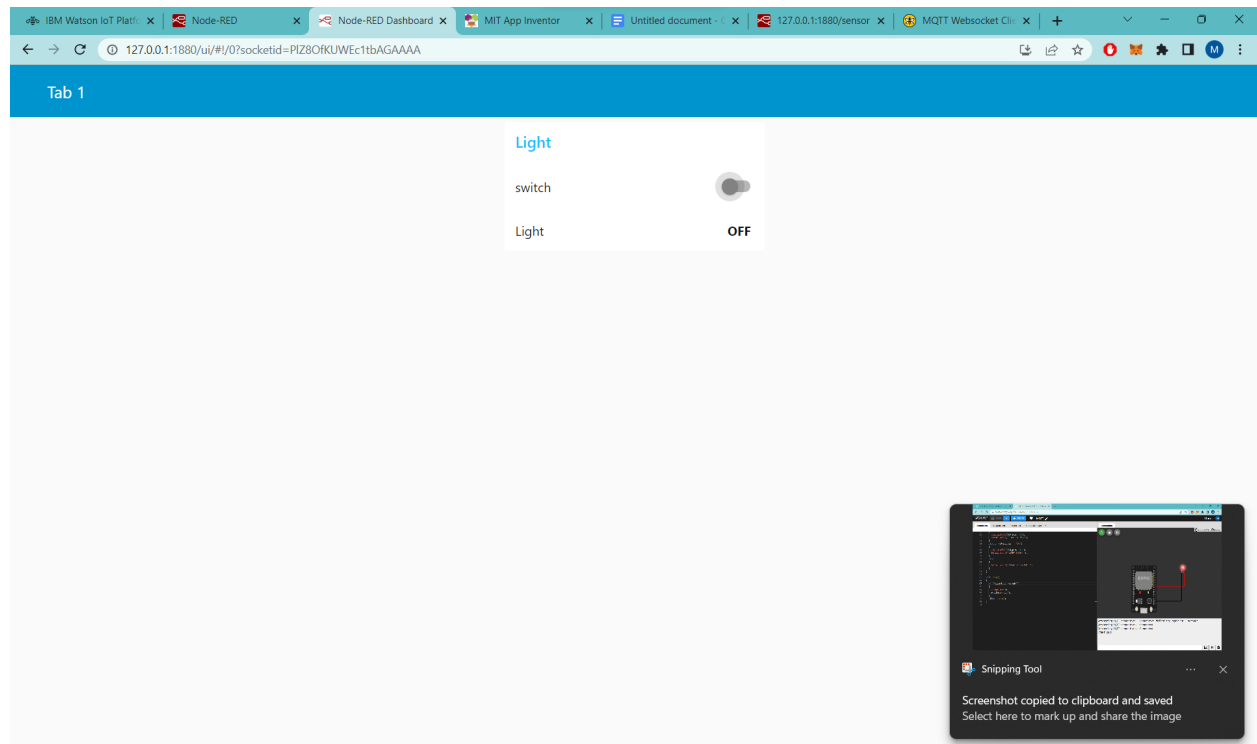
Node Red dashboard(switch on):



Wokwi output(light on):



Node-red dashboard(switch off):



Wokwi output(light off):

