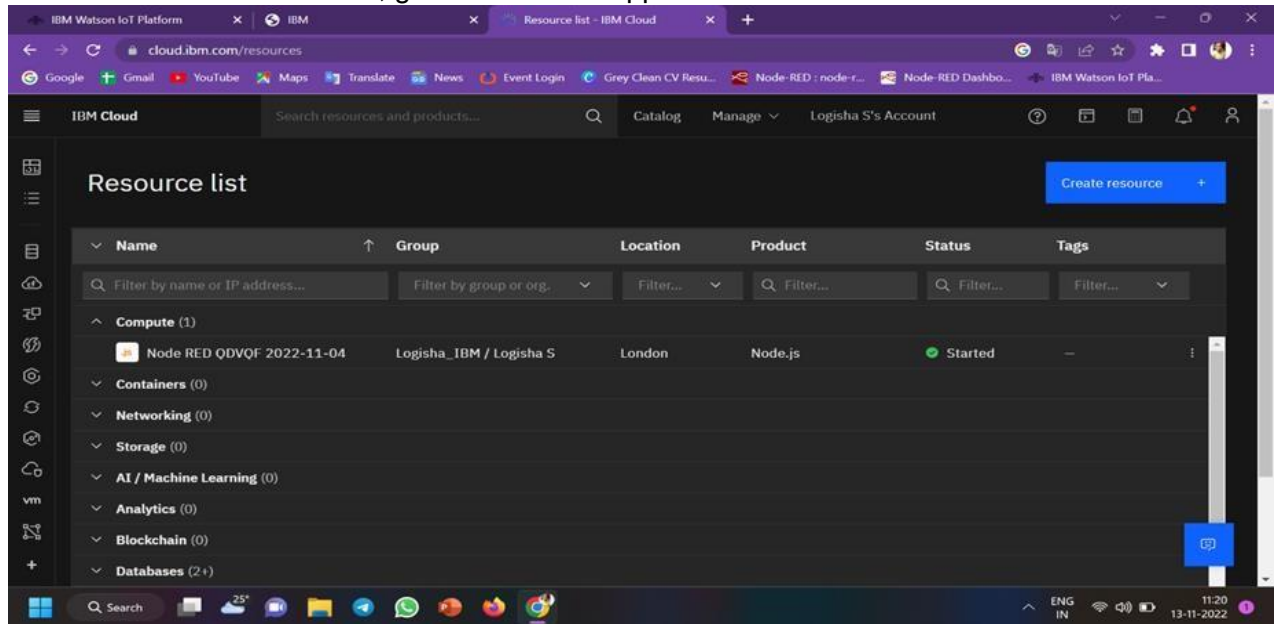


DEVELOP THE WEB APPLICATION USING NODE-RED

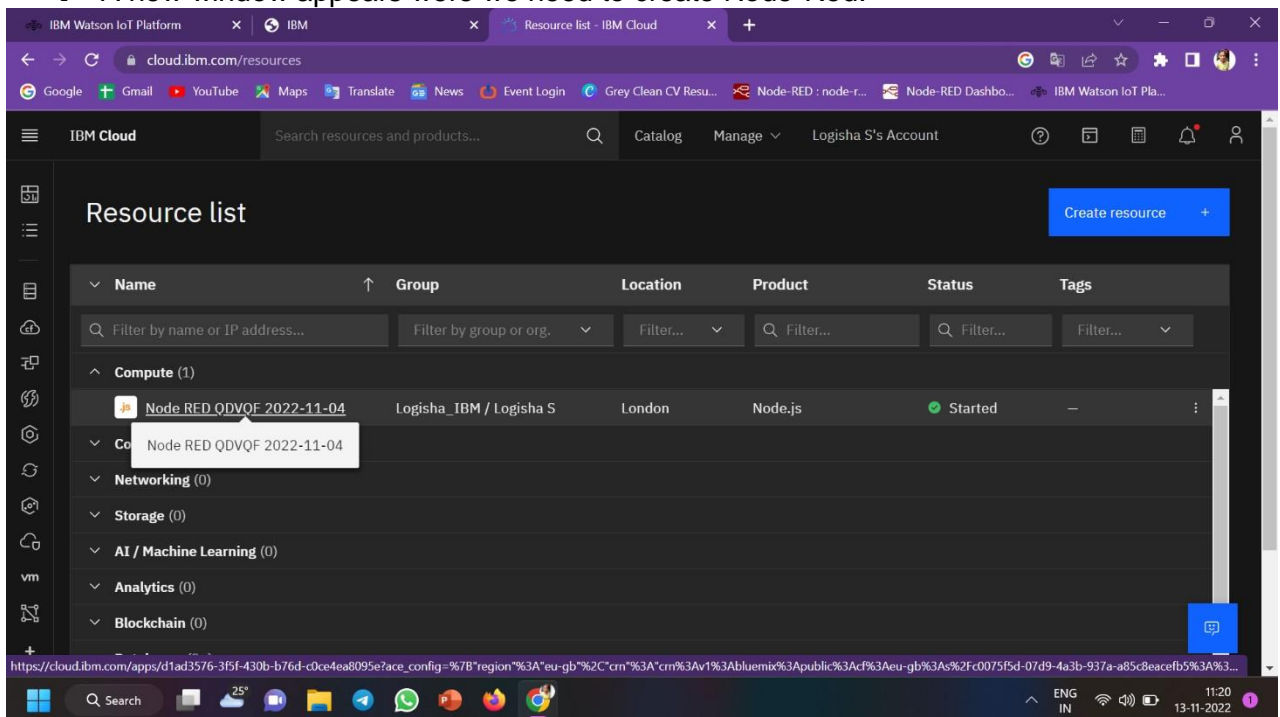
Date	18 NOVEMBER 2022
Team ID	PNT2022TMID16936
Project Name	Gas Leakage Monitoring and Alerting System

STEPS:

✚ IBM cloud dashboard, go to Node-Red application.

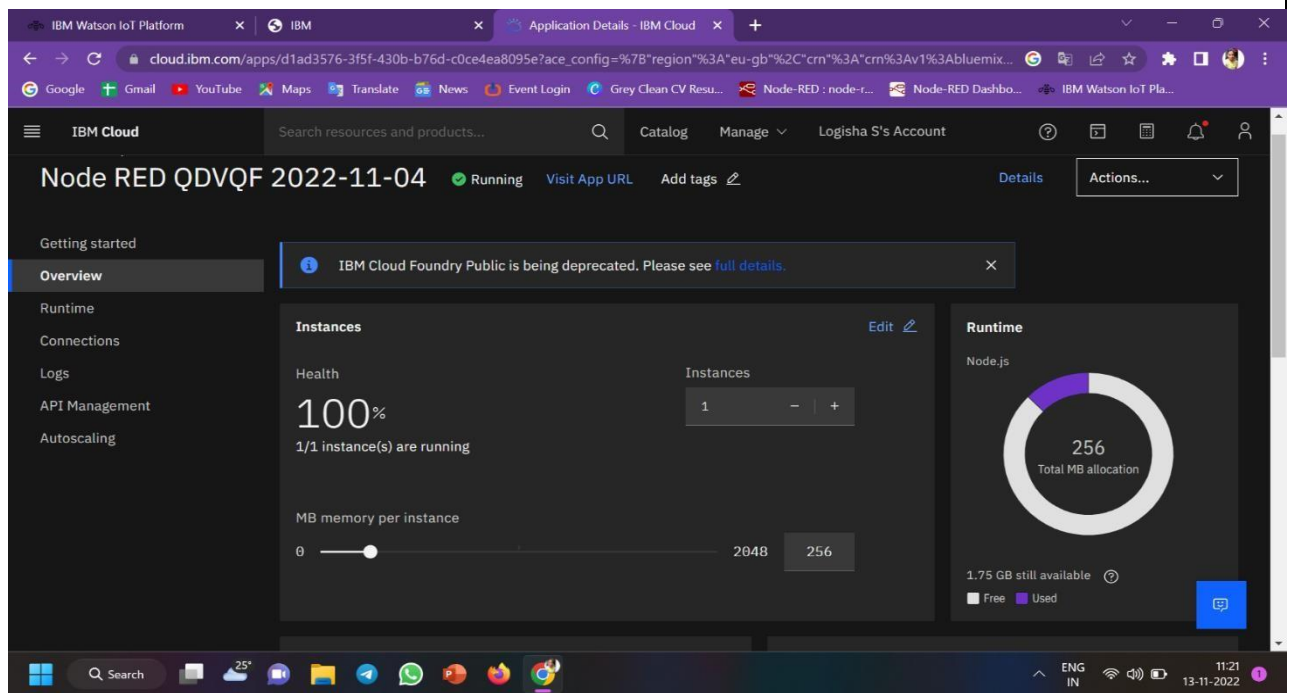


✚ A new window appears where we need to create Node-Red.

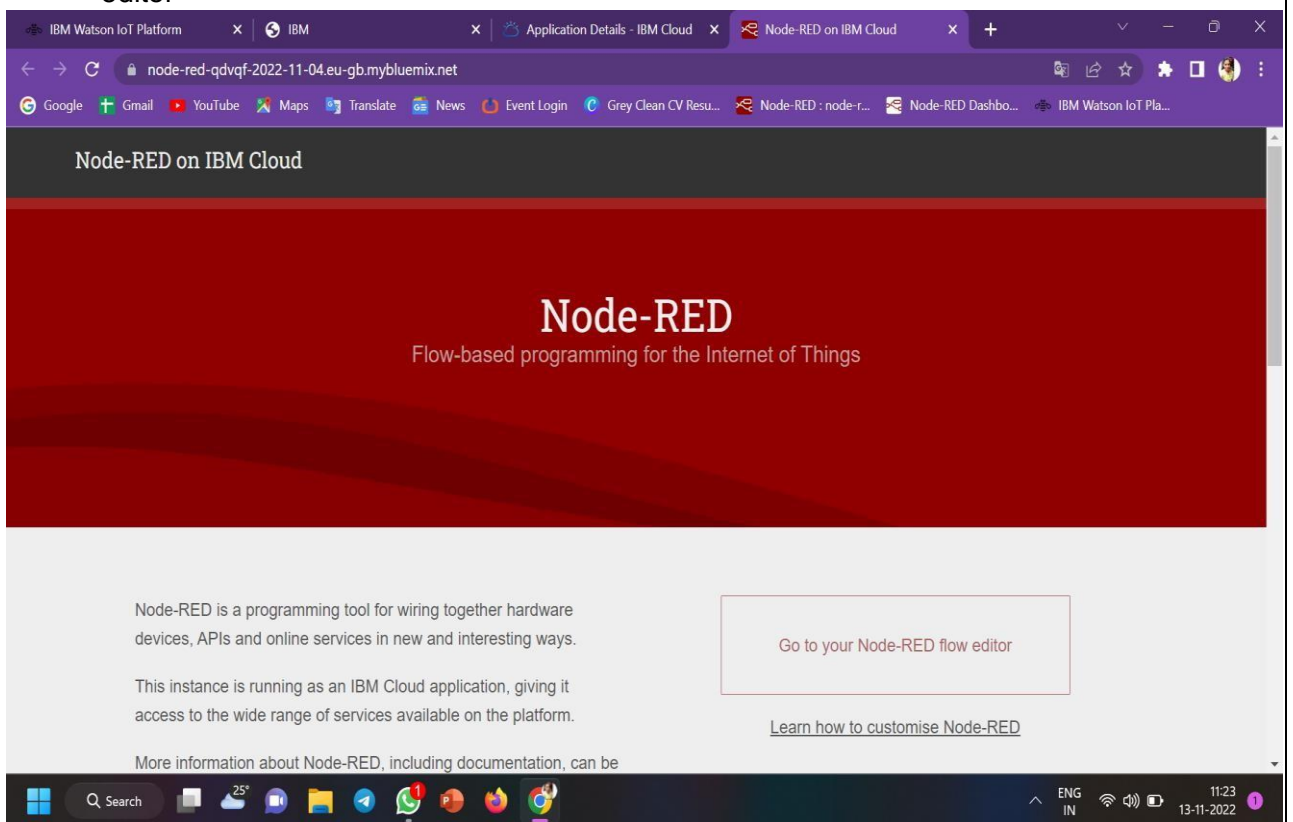


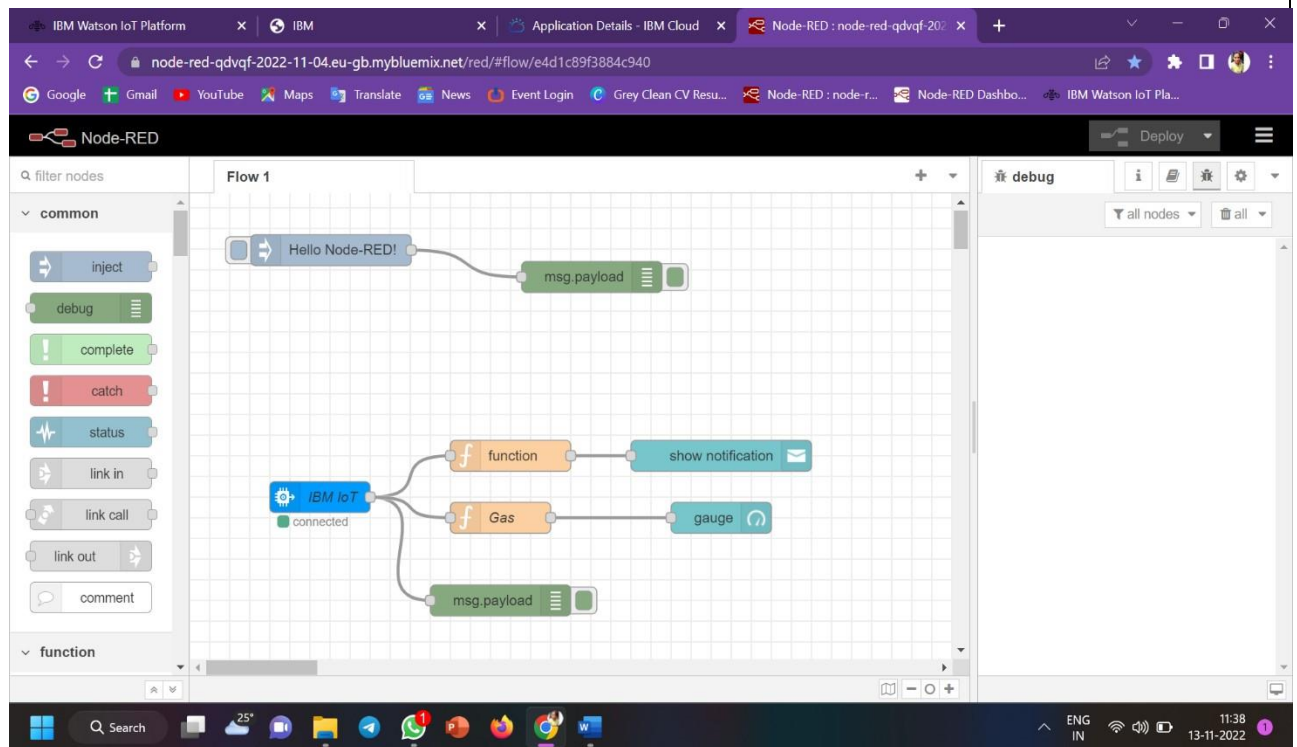
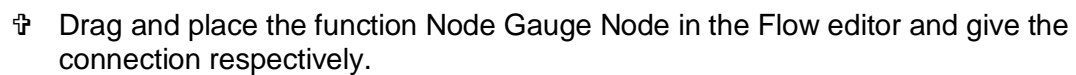
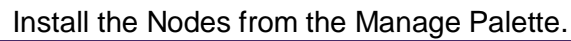


✚ Click on visit App URL in Node-Red service dashboard.



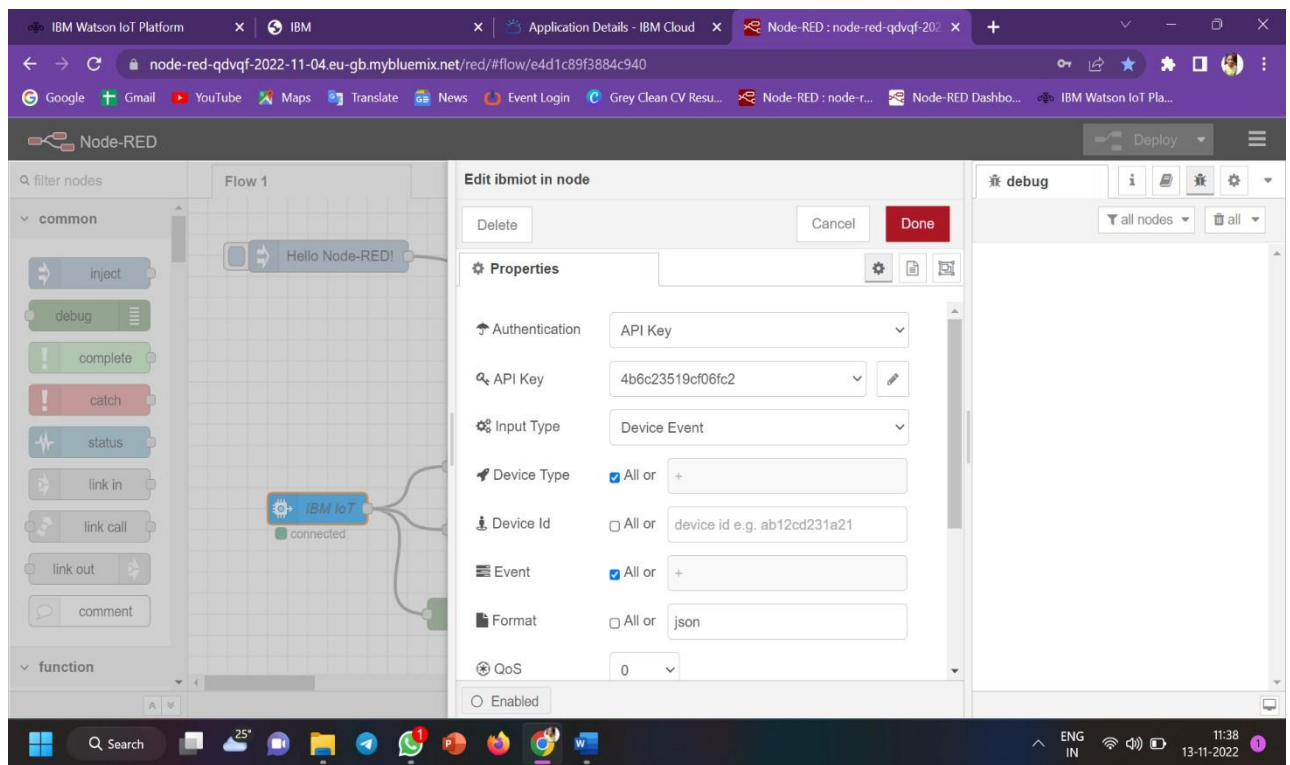
✚ Click on your Node-Red flow editor where you will be redirected to the Node-Red flow editor



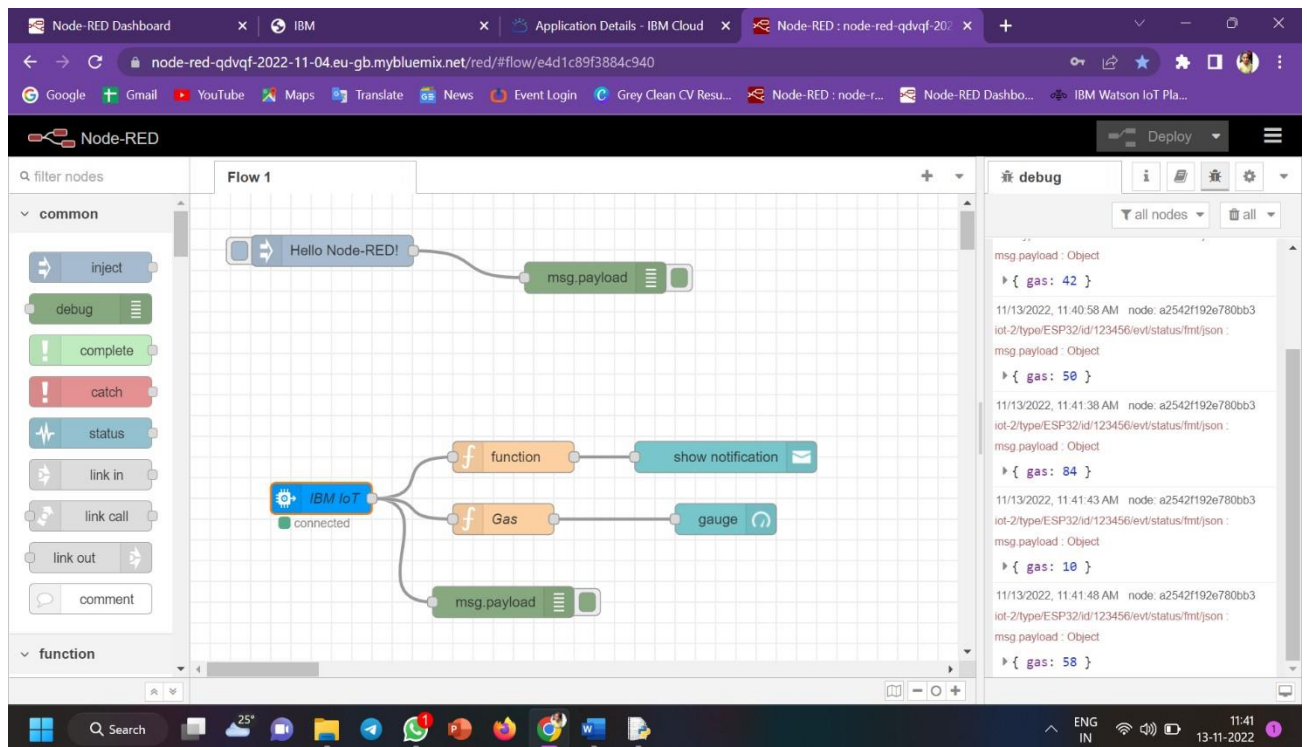




To generate API Key go to IBM IoT Platform , In Apps section ->Click on generate API key.



✚ Click on the Deploy option to check the connection status please the debug Node in the flow editor and click on the deploy to see the respective value in the debug tab.





Edit the Gauge Node by giving the range values from minimum to maximum values.

Node-RED Dashboard

node-red-qdvqf-2022-11-04.eu-gb.mybluemix.net/red/#flow/e4d1c89f3884c940

Node-RED

Flow 1

common

function

inject

debug

complete

catch

status

link in

link call

link out

comment

IBM IoT

Hello Node-RED!

gauge

min -20 max 100

Done

debug

msg.payload: Object

{ gas: 51 }

11/13/2022, 11:48:41 AM node: a2542f192e780bb3

iot-2/type/ESP32/id/123456/evt/status/fmt/json :

msg.payload: Object

{ gas: 44 }

11/13/2022, 11:48:46 AM node: a2542f192e780bb3

iot-2/type/ESP32/id/123456/evt/status/fmt/json :

msg.payload: Object

{ gas: 55 }

11/13/2022, 11:48:51 AM node: a2542f192e780bb3

iot-2/type/ESP32/id/123456/evt/status/fmt/json :

msg.payload: Object

{ gas: -4 }

11/13/2022, 11:48:56 AM node: a2542f192e780bb3

iot-2/type/ESP32/id/123456/evt/status/fmt/json :

msg.payload: Object

{ gas: 14 }

After editing the Nodes deploy it.

Node-RED Dashboard

node-red-qdvqf-2022-11-04.eu-gb.mybluemix.net/red/#flow/e4d1c89f3884c940

Node-RED

Flow 1

common

function

inject

debug

complete

catch

status

link in

link call

link out

comment

IBM IoT

Hello Node-RED!

function

Gas

gauge

msg.payload

msg.payload

show notification

debug

msg.payload: Object

{ gas: 6 }

11/13/2022, 11:50:37 AM node: a2542f192e780bb3

iot-2/type/ESP32/id/123456/evt/status/fmt/json :

msg.payload: Object

{ gas: 98 }

11/13/2022, 11:50:42 AM node: a2542f192e780bb3

iot-2/type/ESP32/id/123456/evt/status/fmt/json :

msg.payload: Object

{ gas: 44 }

11/13/2022, 11:50:47 AM node: a2542f192e780bb3

iot-2/type/ESP32/id/123456/evt/status/fmt/json :

msg.payload: Object

{ gas: 48 }

11/13/2022, 11:50:52 AM node: a2542f192e780bb3

iot-2/type/ESP32/id/123456/evt/status/fmt/json :

msg.payload: Object

{ gas: 21 }

RESULT:

Thus, the Node-Red Web Application is created successfully.