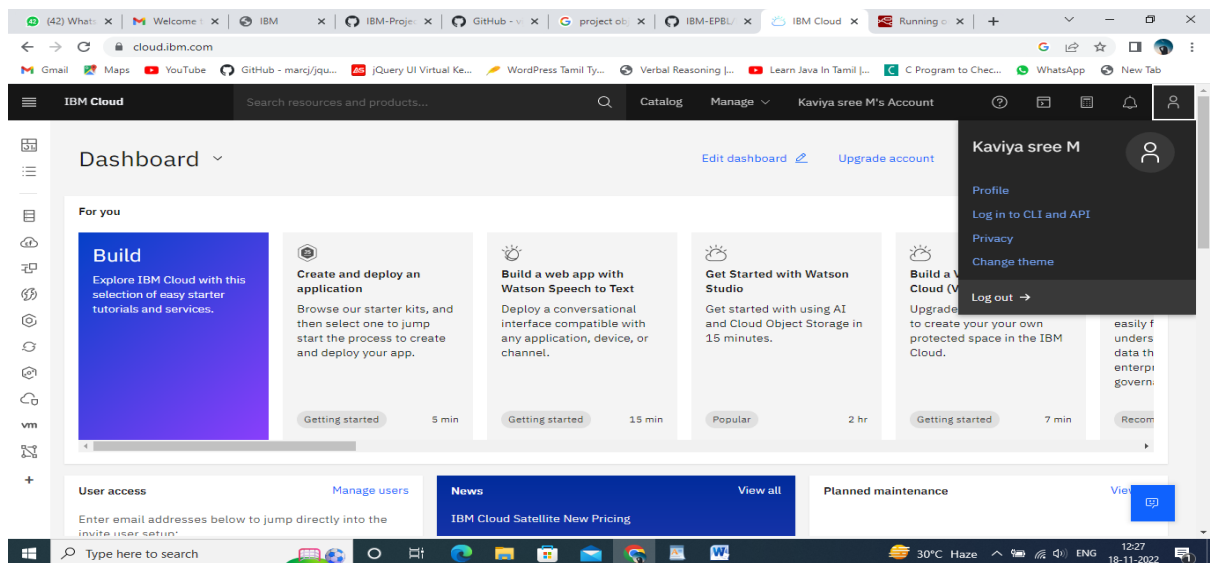


Create and Configure Node-Red service

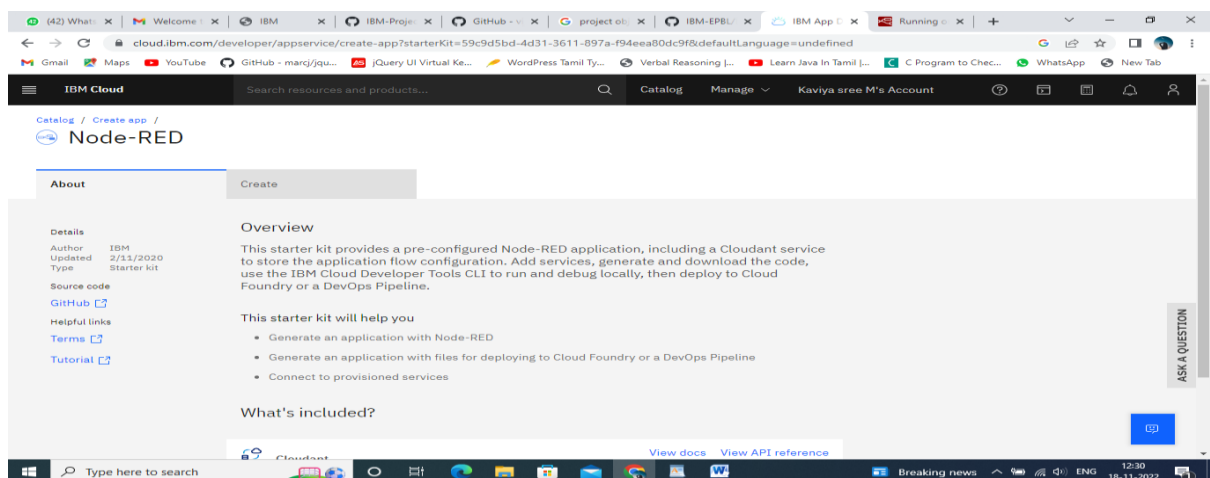
Team ID	PNT2022TMID23589
Project Name	SMART SOLUTIONS FOR RAILWAYS

CREATING NODE-RED IN IBM CLOUD

STEP 1: Open IBM cloud:



STEP 2: Go to catalog and search for node red app and open it:



STEP 3: Enter the app name, location and select the plan and click on create.:

The screenshot shows the IBM Cloud Developer console with the 'Create app' form for Node.js. The form includes fields for Platform (Node.js), Service details (Cloudant), Region (Chennai), Resource group (Default), and Pricing plan (node-red-kaviya-sree-cloudant-1668753643261). There are 'Cancel' and 'Create' buttons at the bottom.

Examples: env:dev, version:1

Platform

☒ Node.js

Service details

Cloudant

★ = You have existing instances of this service available to use in this kit. If you wish to use the existing service, select it from the pricing plan menu.

Region Chennai Resource group Default

Pricing plan

[Pricing details](#) [Terms](#)

STEP 4: click on deploy your app button:

The screenshot shows the IBM Cloud Developer console with the 'Node RED KAVIYA SREE' app details. The 'Details' section shows the App URL, Source (Download code), Resource group (Default), Deployment target, and Created date (11/18/2022). The 'Deployment Automation' section shows a 'Deploy your app' button.

Resource list / App details /

Node RED KAVIYA SREE

Actions...

Details

App URL You must deploy your app first

Source

Resource group Default

Deployment target You must deploy your app first

Created 11/18/2022

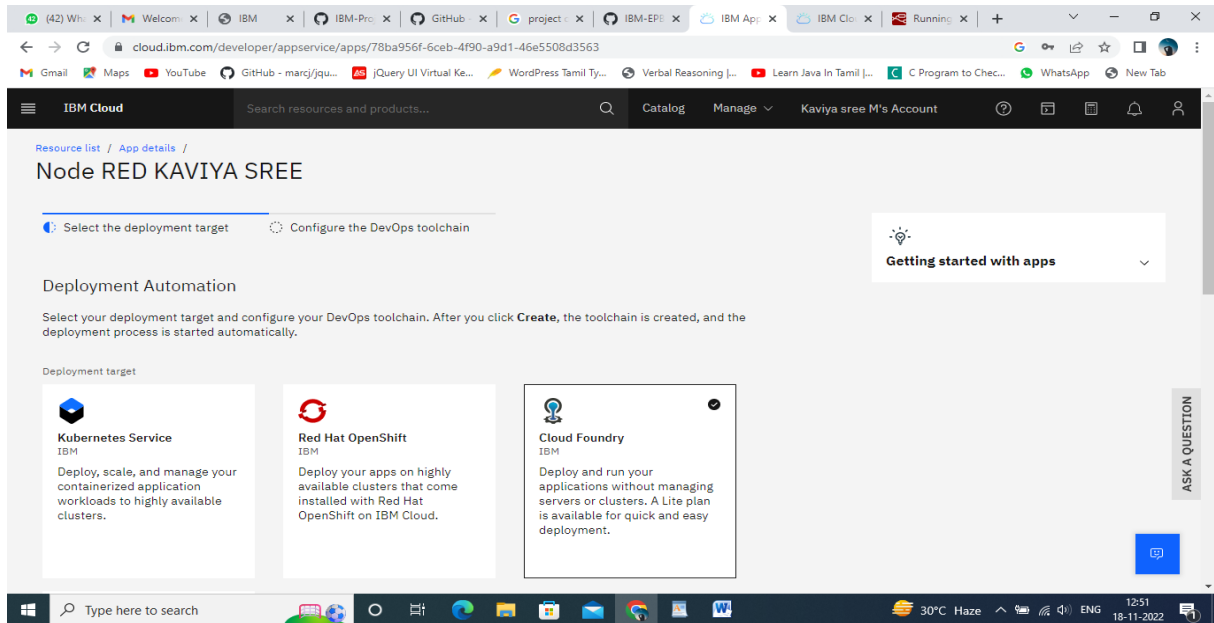
Services

Deployment Automation

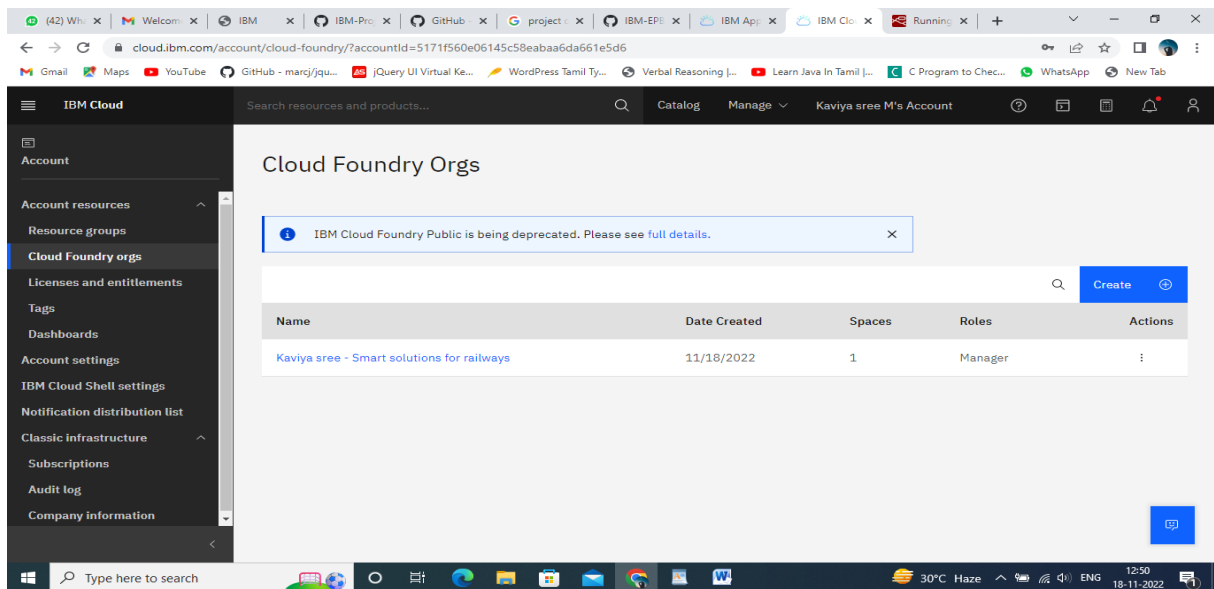
Configure Continuous Delivery

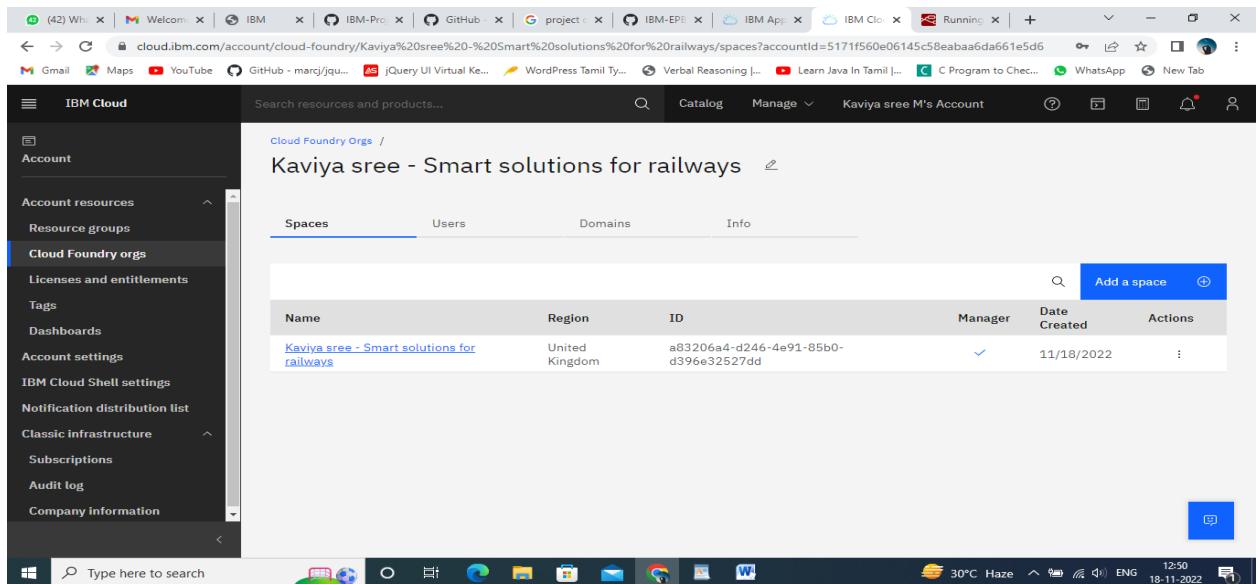
Continuous Delivery is not enabled for this app. Enable Continuous Delivery to automate builds, tests, and deployments through Delivery Pipeline, GitLab, and more.

STEP 5: In deployment automation select cloud foundry and click on create.org:

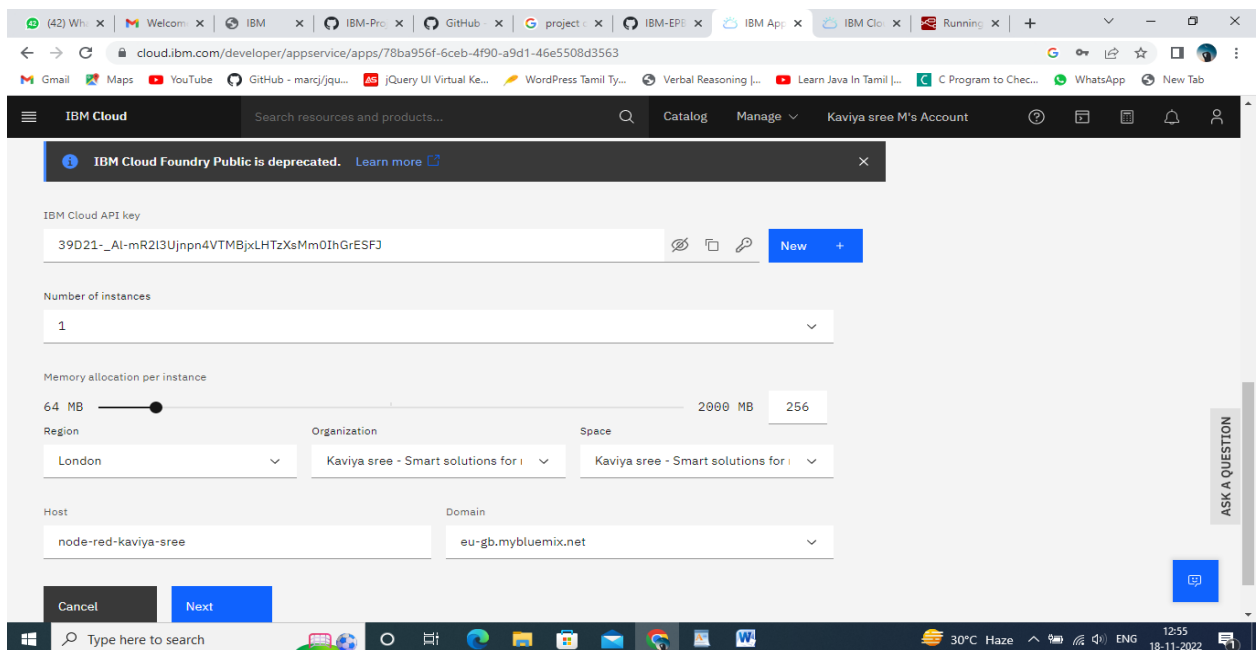


STEP 6: click on create button and enter the name and create a space:





STEP 7: In app development click new on API key and select region and click next:



STEP 8: select the region and click create:

The screenshot shows the IBM Cloud Developer console interface. The breadcrumb navigation is 'Resource list / App details / Node RED KAVIYA SREE'. The main heading is 'Node RED KAVIYA SREE'. Below it, there are two tabs: 'Select the deployment target' (active) and 'Configure the DevOps toolchain'. The 'Configure the DevOps toolchain' tab is selected, and it displays the following content:

Configure the DevOps toolchain

Give your toolchain a name and select the region to create your toolchain in.

DevOps toolchain name

NodeREDKAVIYASREE

Accept the default name, or enter a value up to 100 characters.

Region

London

Back Create

On the right side, there is a 'Getting started with apps' card. At the bottom right, there is an 'ASK A QUESTION' button. The bottom of the screen shows the Windows taskbar with various application icons and the system clock showing 12:56 on 18-11-2022.

STEP 9: Wait till you get the success in ci-pipeline and app URL is generated:

The screenshot shows the IBM Cloud Developer console interface for the application 'Node RED NGHJKJ 2022-11-04'. The breadcrumb navigation is 'Resource list / App details / Node RED NGHJKJ 2022-11-04'. The main heading is 'Node RED NGHJKJ 2022-11-04'. Below it, there are two tabs: 'Details' (active) and 'Deployment Automation'. The 'Details' tab is selected, and it displays the following information:

Details

App URL: You must deploy your app first

Source: <https://eu-gb.git.cloud.ibm.com/monishkumarts/NodeREDNGH...>

Resource group: Default

Deployment target: You must deploy your app first

Created: 11/4/2022

Services

Cloudant

Open dashboard Documentation API reference

Credentials

Connect existing services Create service

The 'Deployment Automation' tab is also visible, showing the following information:

Deployment Automation

Name: NodeREDNGHJKJ2022-11-04

Location: London

Tool integrations

Delivery Pipelines

Name: ci-pipeline

Status: No stages detected

Name: pr-pipeline

Status: No stages detected

On the right side, there is a 'Getting started quickly' card. At the bottom right, there is an 'ASK A QUESTION' button. The bottom of the screen shows the Windows taskbar with various application icons and the system clock showing 12:56 on 18-11-2022.

STEP 10: Now click on the generated APP URL:

The screenshot shows the IBM Cloud console interface for a resource named "Node RED NGHJKJ 2022-11-04". The top navigation bar includes the IBM Cloud logo, a search bar, and user account information. The main content area is divided into several panels:

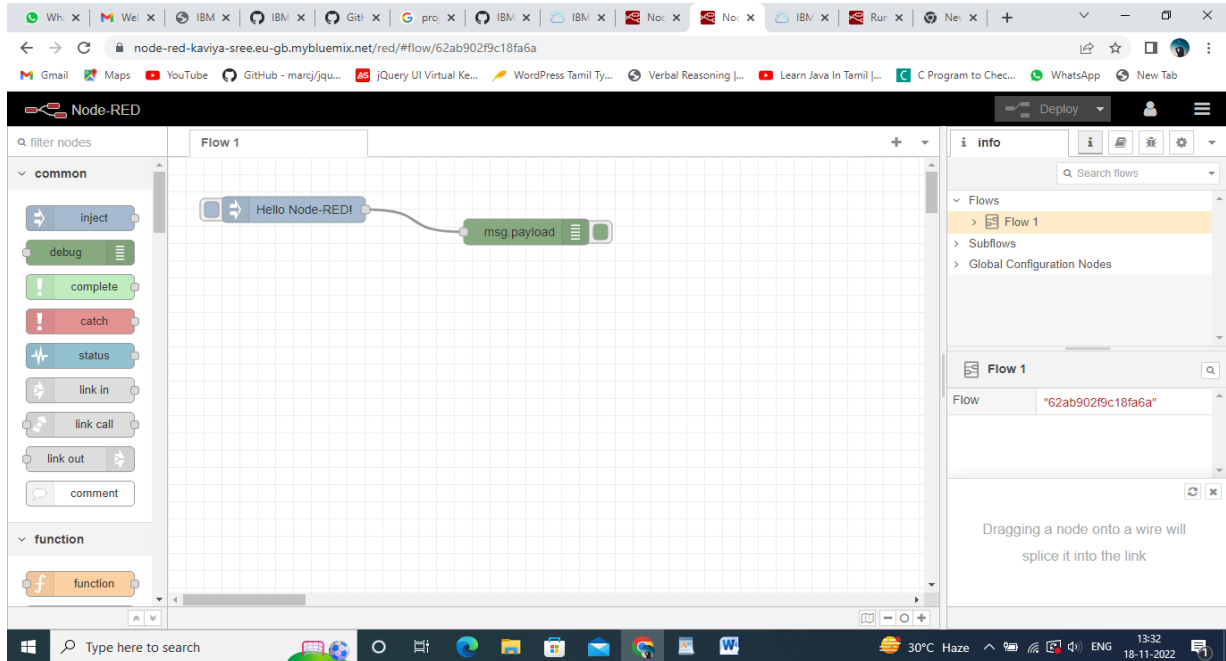
- Details:** A table showing key information about the application:

Property	Value
App URL	https://node-red-nghkj-2022-11-04.eu-gb.mybluemix.net
Source	https://eu-gb.git.cloud.ibm.com/monishkumarts/NodeREDNGH...
Resource group	Default
Deployment target	Node RED NGHJKJ 2022-11-04
Created	11/4/2022
- Services:** A section for managing services, including a "Cloudant" service with links to "Open dashboard", "Documentation", and "API reference". It also features buttons for "Connect existing services" and "Create service".
- Deployment Automation:** A section showing the deployment status. It lists two pipelines: "ci-pipeline" with a status of "Success" and "pr-pipeline" with a status of "No stages detected".
- Getting started quickly:** A sidebar on the right with a "Getting started quickly" section, providing a list of steps to configure the app, such as connecting services, downloading code, and deploying the app.

STEP 11 : You will redirected to your node-red on ibm cloud page:

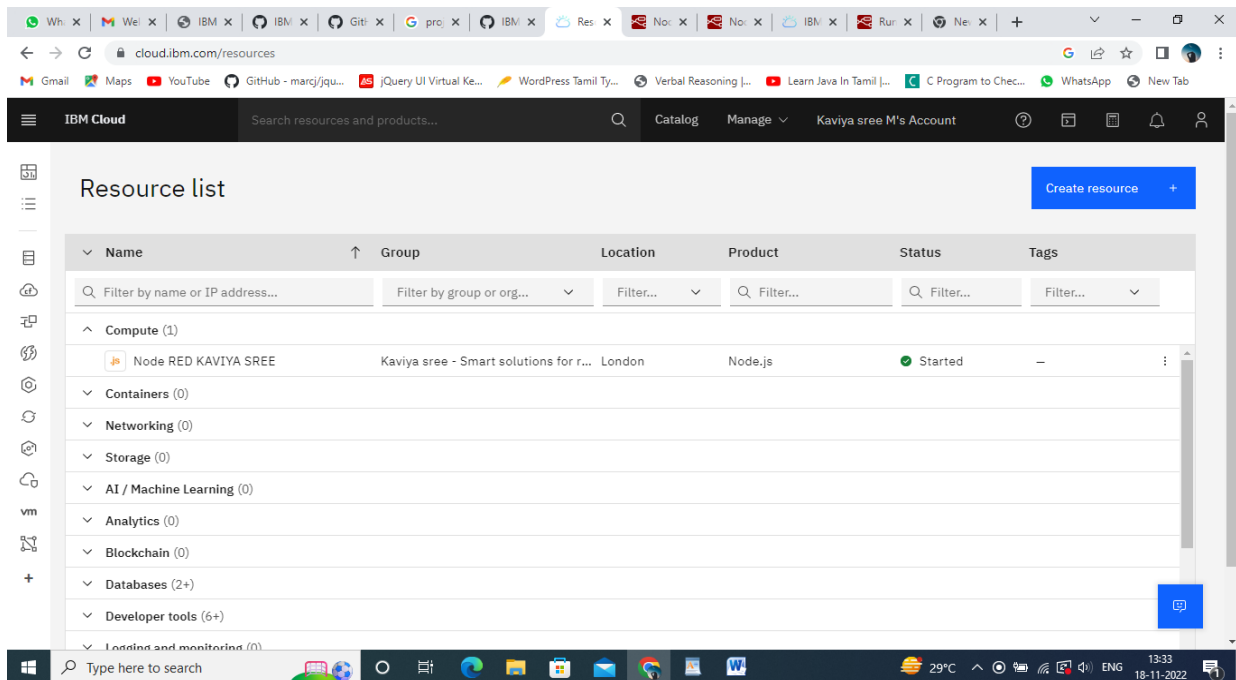
The screenshot shows the Node-RED web interface. The top header reads "Node-RED on IBM Cloud". The main content area has a dark red background with the "Node-RED" logo and the tagline "Flow-based programming for the Internet of Things". Below this, there is a light gray section with text explaining that Node-RED is a programming tool for wiring together hardware devices, APIs, and online services, and that this instance is running as an IBM Cloud application. It also provides a link to the Node-RED documentation at nodered.org.

STEP 12: Click on node-red flow editor and you will be redirected to your node-red workspace:

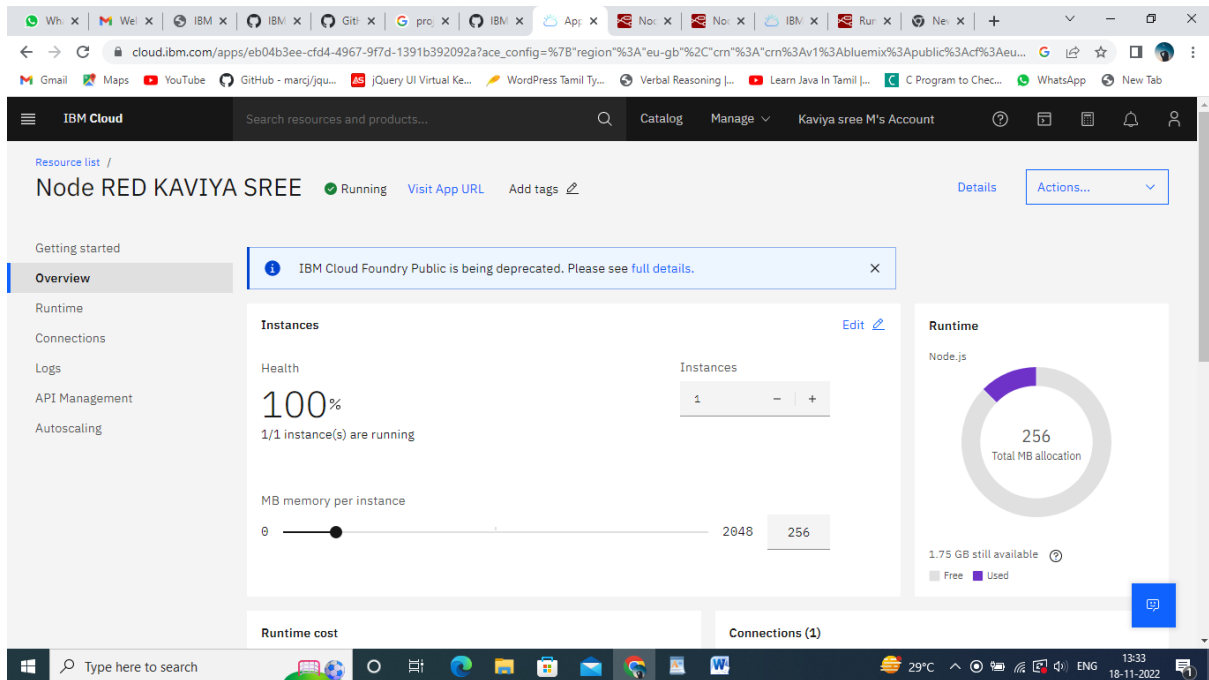


DIRECTING TO CREATED NODE-RED WORKSPACE

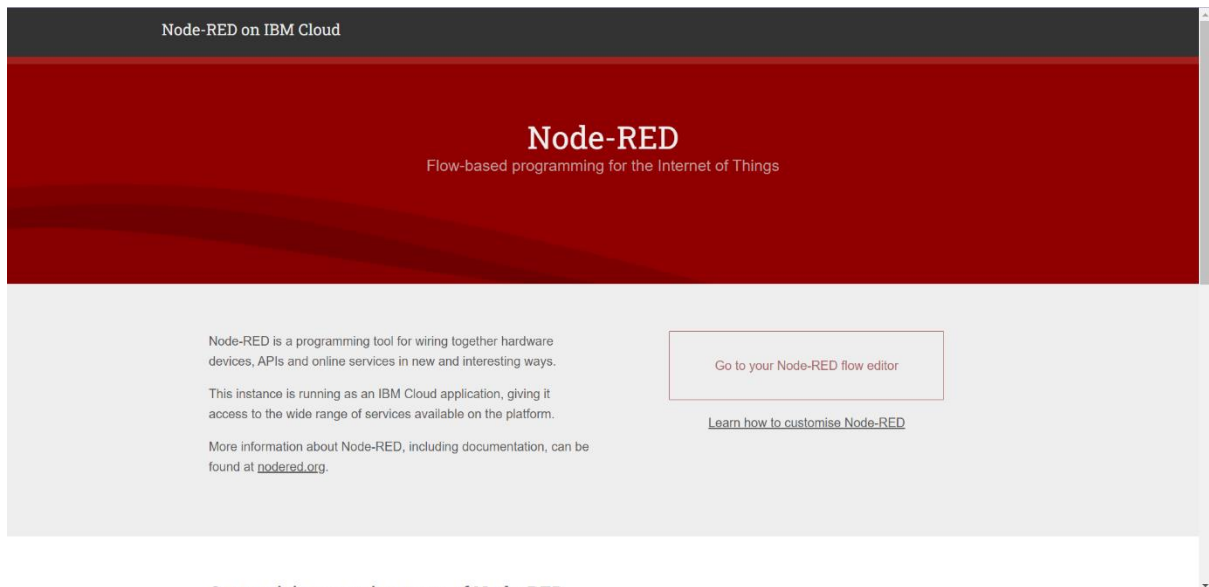
STEP 13: In resource select compute and click on node-red :



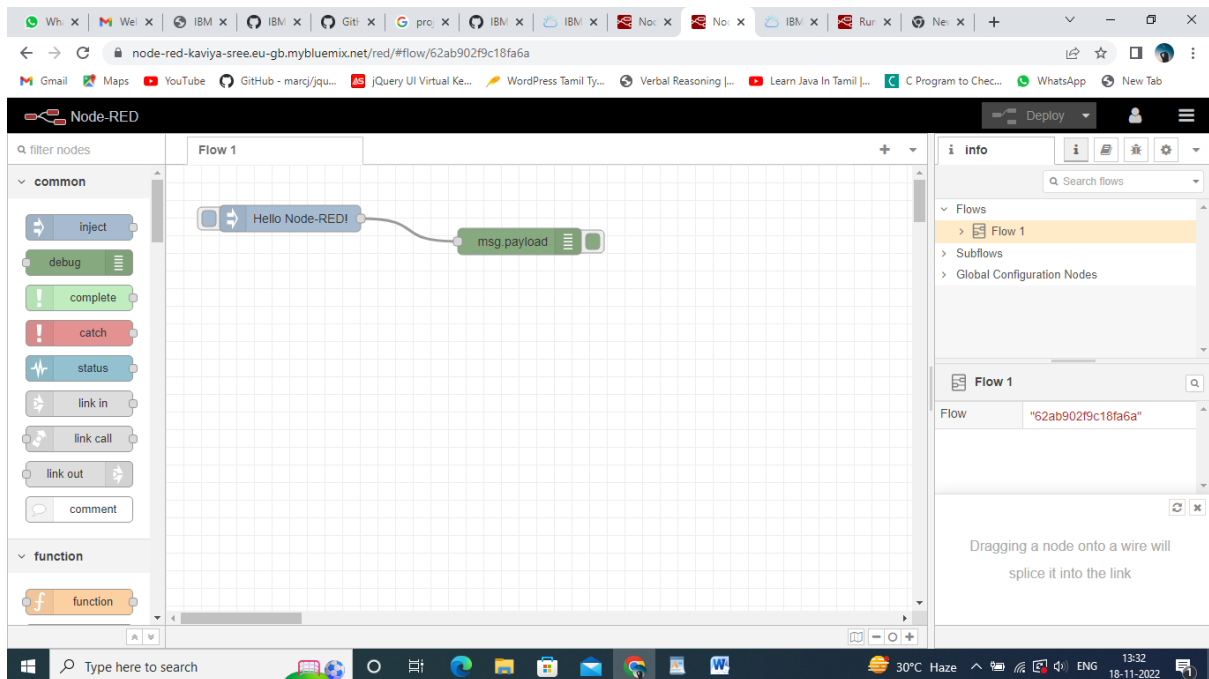
STEP 14: click on visit app url to be redirected to node red:



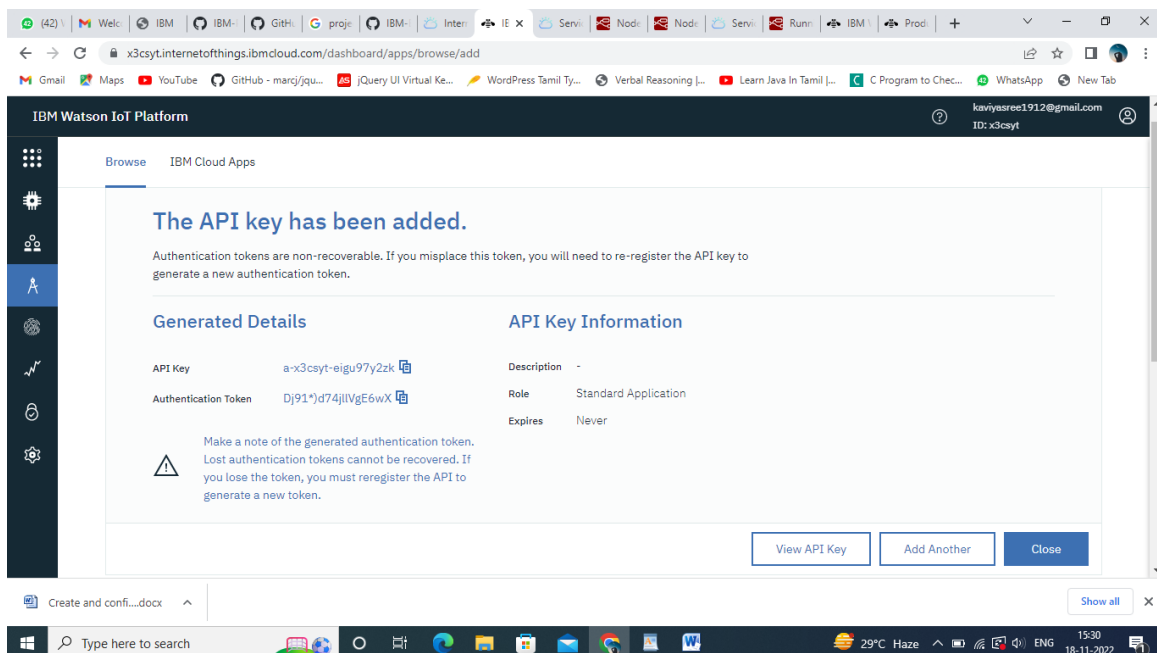
STEP 15: Click on go to your NODE-RED flow editor button:



STEP 16: You will be redirected to the node red flow editor:



STEP 17: Generating API key and Authentication token:



STEP 18: Edit Ibm Iot in node:

The screenshot shows the Node-RED web interface. On the left, the 'common' node palette is visible. In the center workspace, a flow named 'Flow 1' contains an 'IBM IoT' node, which is marked as 'connected'. On the right, the 'Edit ibmiot in node' configuration panel is open. It includes a 'Delete' button, 'Cancel', and 'Done' buttons. The 'Properties' section contains the following settings:

- Authentication: API Key
- API Key: e19c2b2383d75b20
- Input Type: Device Event
- Device Type: ☐ All or b11m3edevicetype
- Device Id: ☐ All or MKSJ16
- Event: ☒ All or +
- Format: ☐ All or json
- QoS: 0
- Name: IBM IoT
- Service: registered

Below the properties, a yellow note states: 'Use the Input Type property to configure this node to receive Events sent by IoT Devices, Commands sent to IoT Devices, Status Messages referring to IoT Devices, or Status Messages referring to'. At the bottom of the panel, there is an 'Enabled' checkbox.

STEP 19: Connect IbmIot in and debug 1 and deploy:

The screenshot shows the Node-RED web interface after the connection. The 'IBM IoT' node is now connected to a 'msg.payload' node. The 'debug' node palette is visible on the left. On the right, the 'info' panel is open, showing the 'IBM IoT' node details:

- Node: *5c22a9fb4112a5ec*
- Type: ibmiot in

Below the node details, a message states: 'Move the selected nodes using the ↑ ↓ and → keys. Hold ⇧ to nudge them further'.

STEP 20: Edit gauge node (the gauge nodes named latitude,longitude and available_seats as fig1,fig2,fig3):

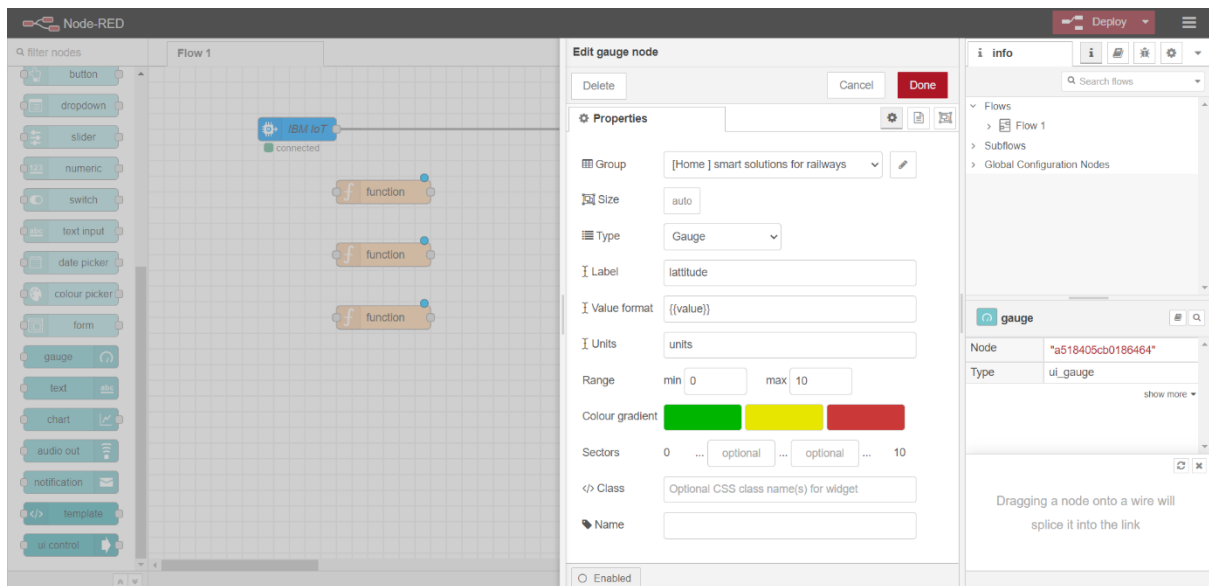


FIG 1

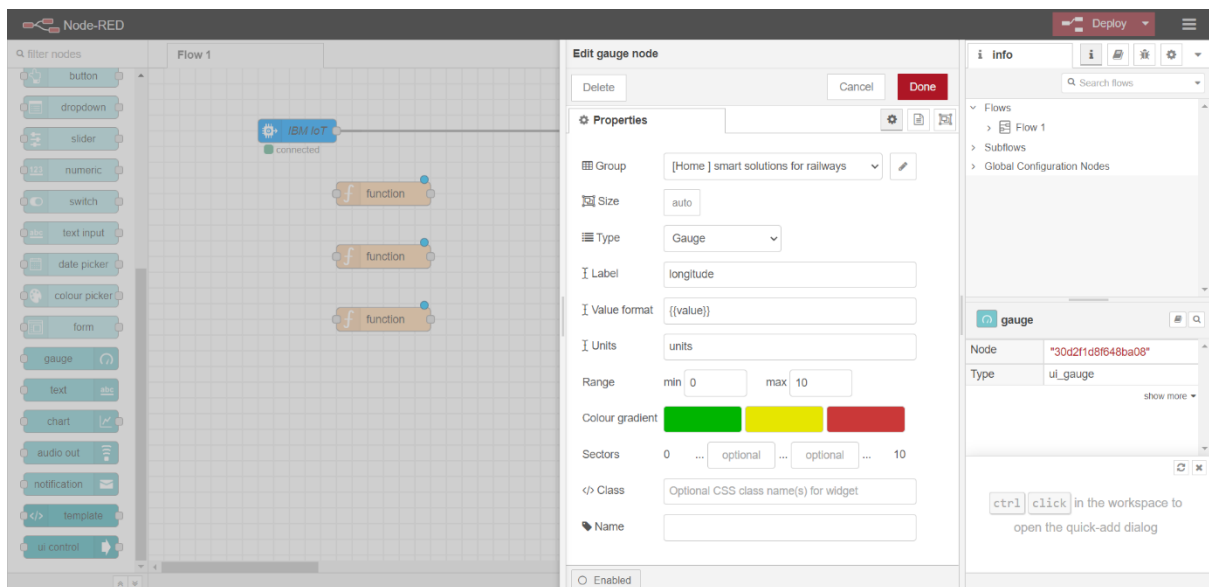


FIG 2

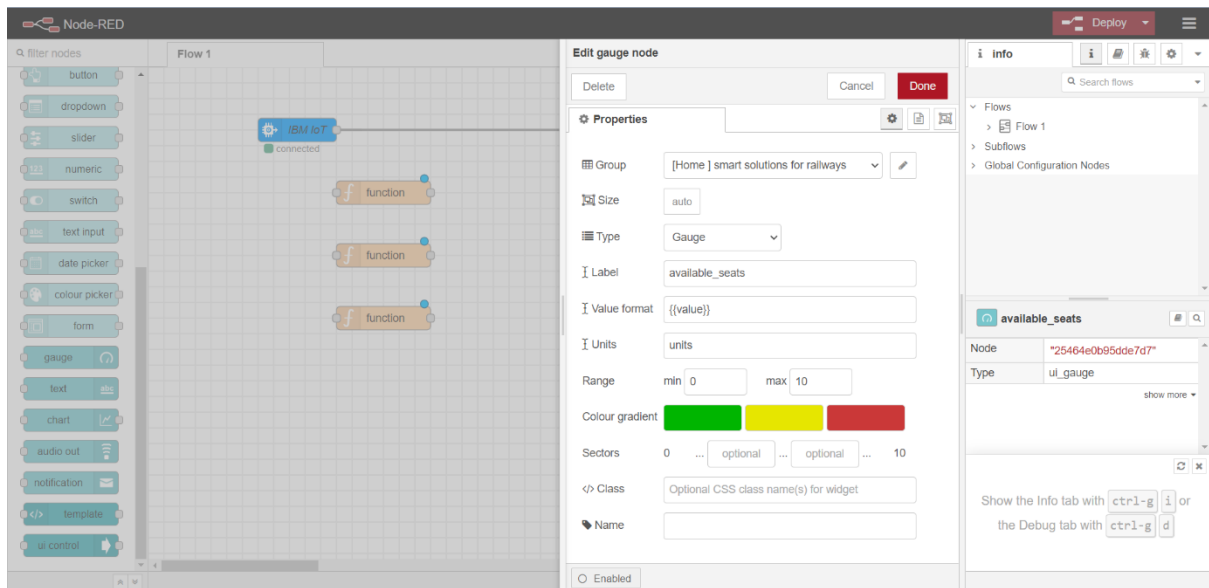
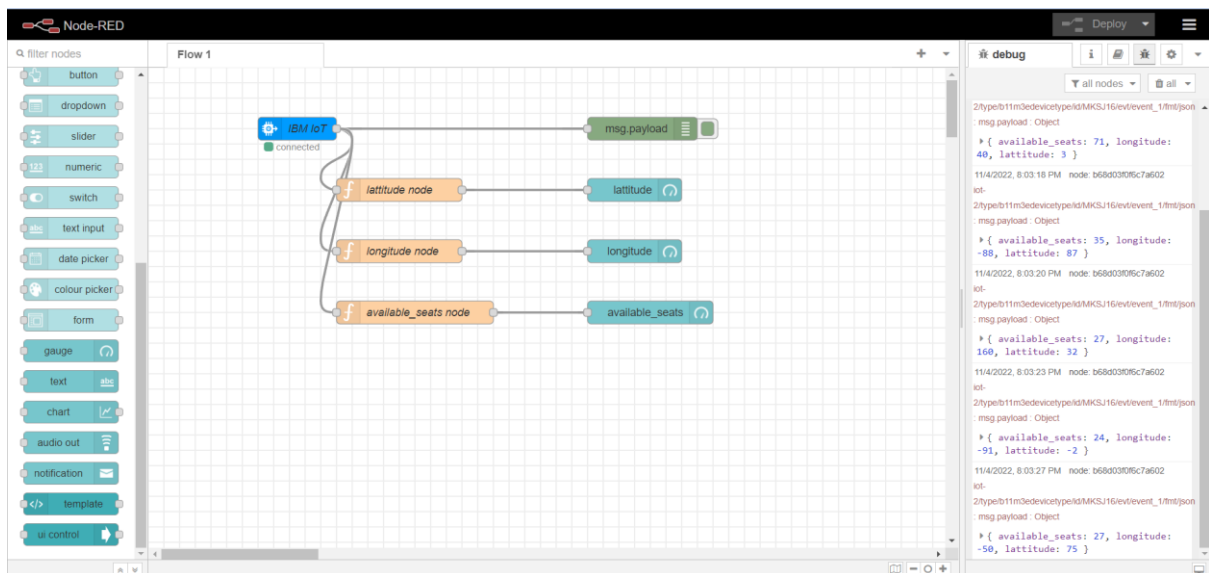


FIG 3

STEP 21: Generate debug message from IBM Watson IoT Platform and connect the nodes:



[← Back](#)

Device Drilldown - MKSJ16

- Connection Information
- Recent Events**
- State
- Device Information
- Metadata
- Diagnostics
- Connection Logs
- Device Actions

Recent Events

The recent events listed show the live stream of data that is coming and going from your device.

Event	Value	For
event_1	{"available seats":60,"longitude":-14,"latitude"...}	jsc
event_1	{"available seats":8,"longitude":-178,"latitude"...}	jsc
event_1	{"available seats":45,"longitude":-34,"latitude"...}	jsc
event_1	{"available seats":47,"longitude":-111,"latitude"...}	jsc
event_1	{"available seats":17,"longitude":21,"latitude":...}	jsc

State

This table shows a list of data points that are reported by this device.

Device Type: b11m3edevicetype

Events 1 New event type +

Event type name Send

Schedule
 Every Minute ▾

Payload
Specify the event payload in the editor window or by uploading a CSV file.

```
0 {  
1   "available seats": random(0, 100),  
2   "longitude": random(-180, 180),  
3   "latitude": random(-90, 90)  
4 }  
5
```

Upload a CSV file

Cancel Save