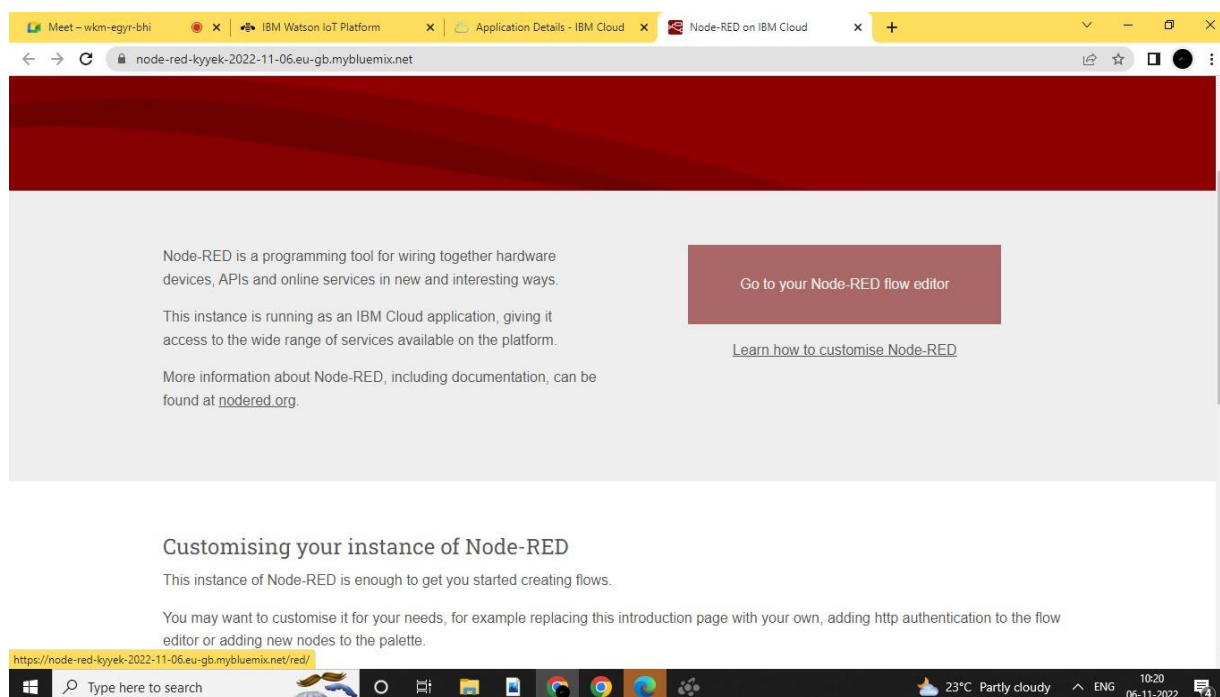


DATE	4 TH NOVEMBER 2022
TEAM ID	PNT2022TMID36645
PROJECT NAME	Smart Farmer -IoT Enabled Smart Farming Application

Step 1:



Meet - wkm-egyr-bhi x IBM Watson IoT Platform x Application Details - IBM Cloud x Node-RED on IBM Cloud x

node-red-kyyek-2022-11-06.eu-gb.mybluemix.net

Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways.

This instance is running as an IBM Cloud application, giving it access to the wide range of services available on the platform.

More information about Node-RED, including documentation, can be found at nodered.org.

[Go to your Node-RED flow editor](#)

[Learn how to customise Node-RED](#)

Customising your instance of Node-RED

This instance of Node-RED is enough to get you started creating flows.

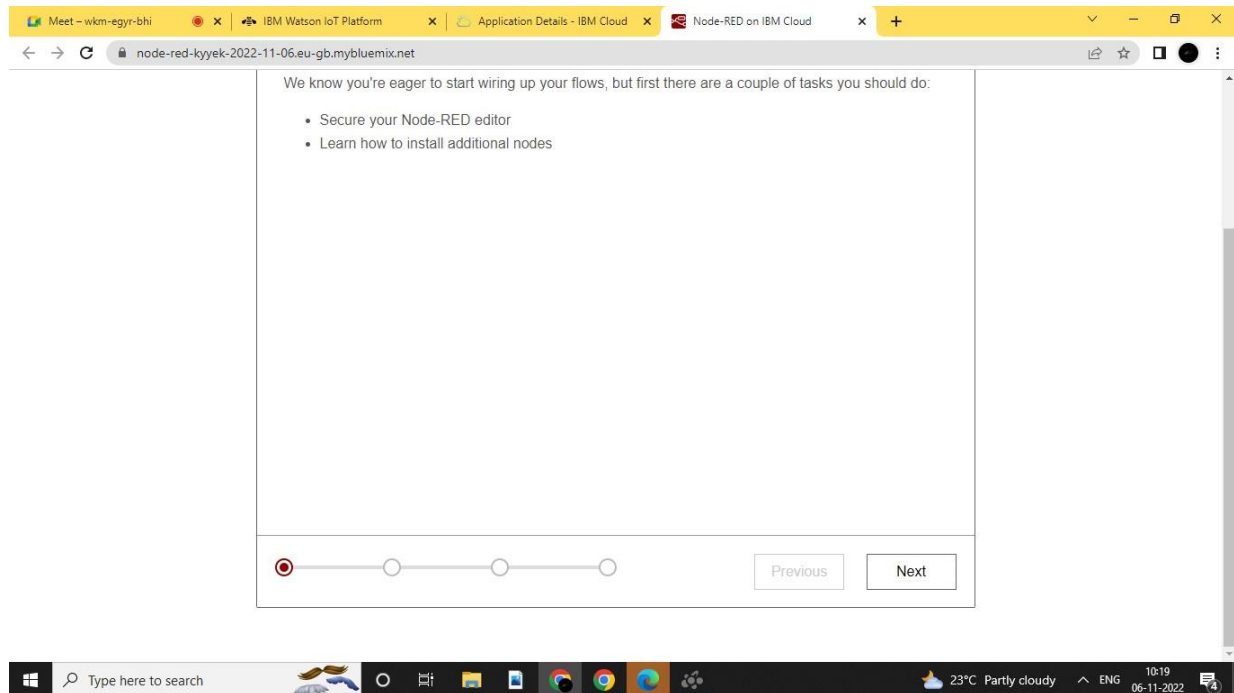
You may want to customise it for your needs, for example replacing this introduction page with your own, adding http authentication to the flow editor or adding new nodes to the palette.

<https://node-red-kyyek-2022-11-06.eu-gb.mybluemix.net/red/>

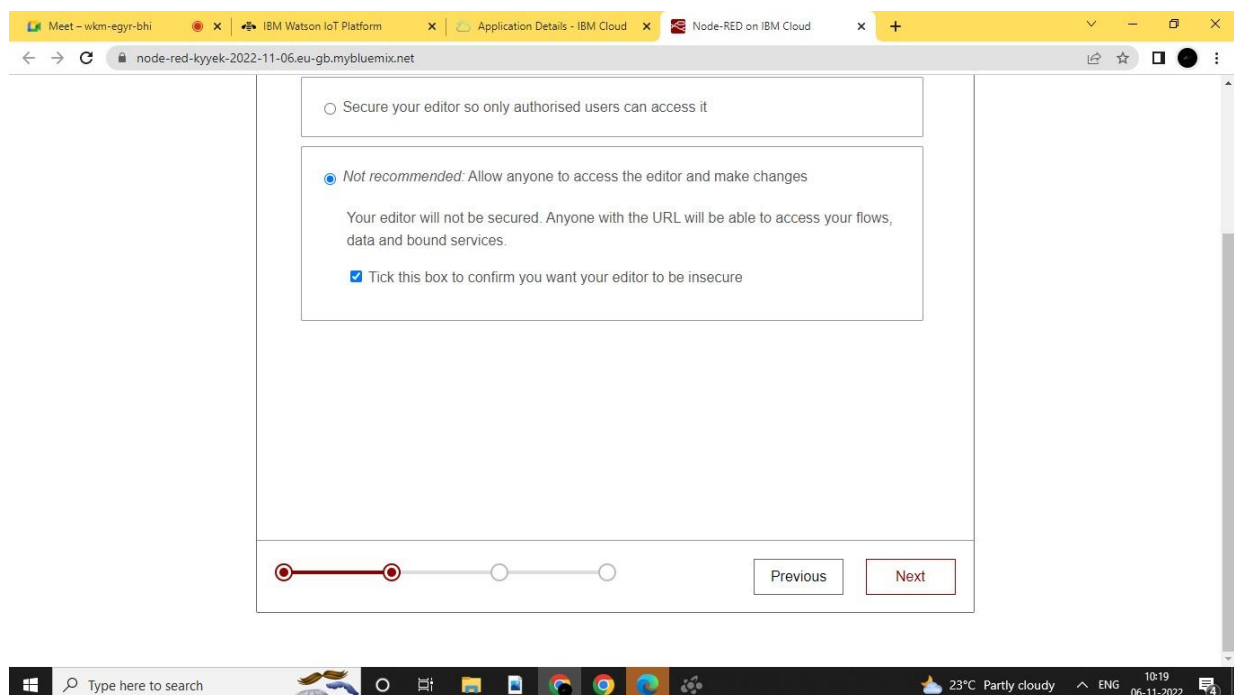
Type here to search

23°C Partly cloudy 10:20 06-11-2022

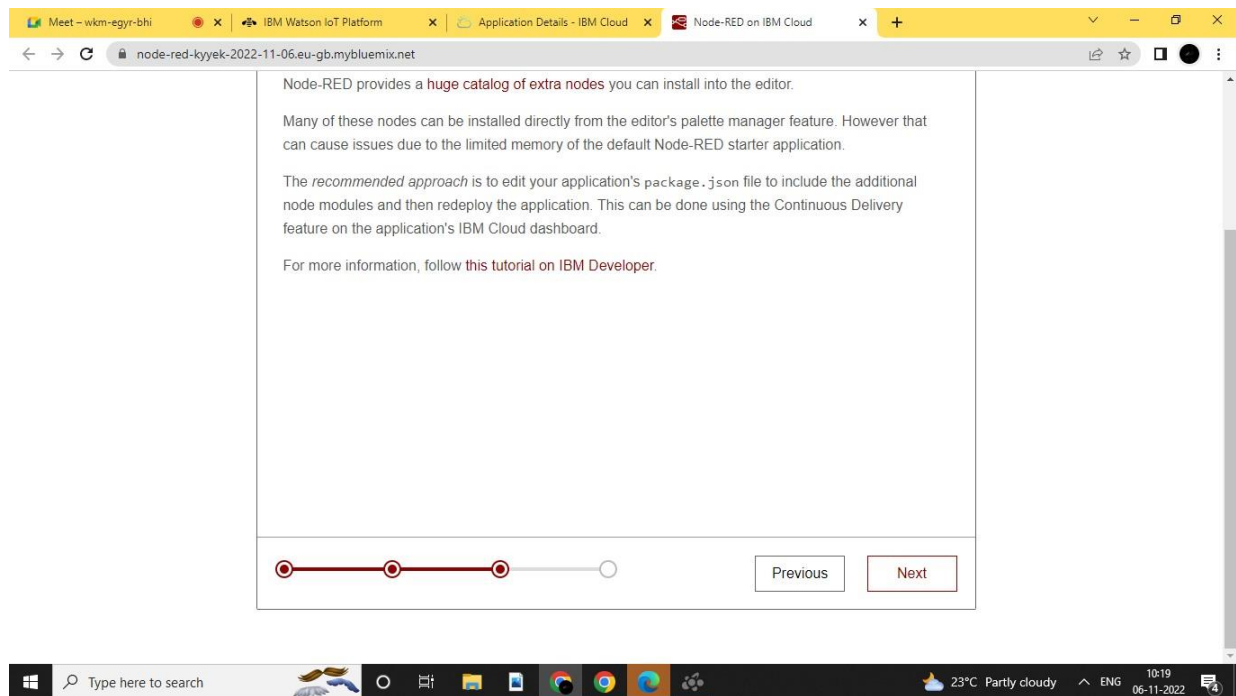
Step 2:



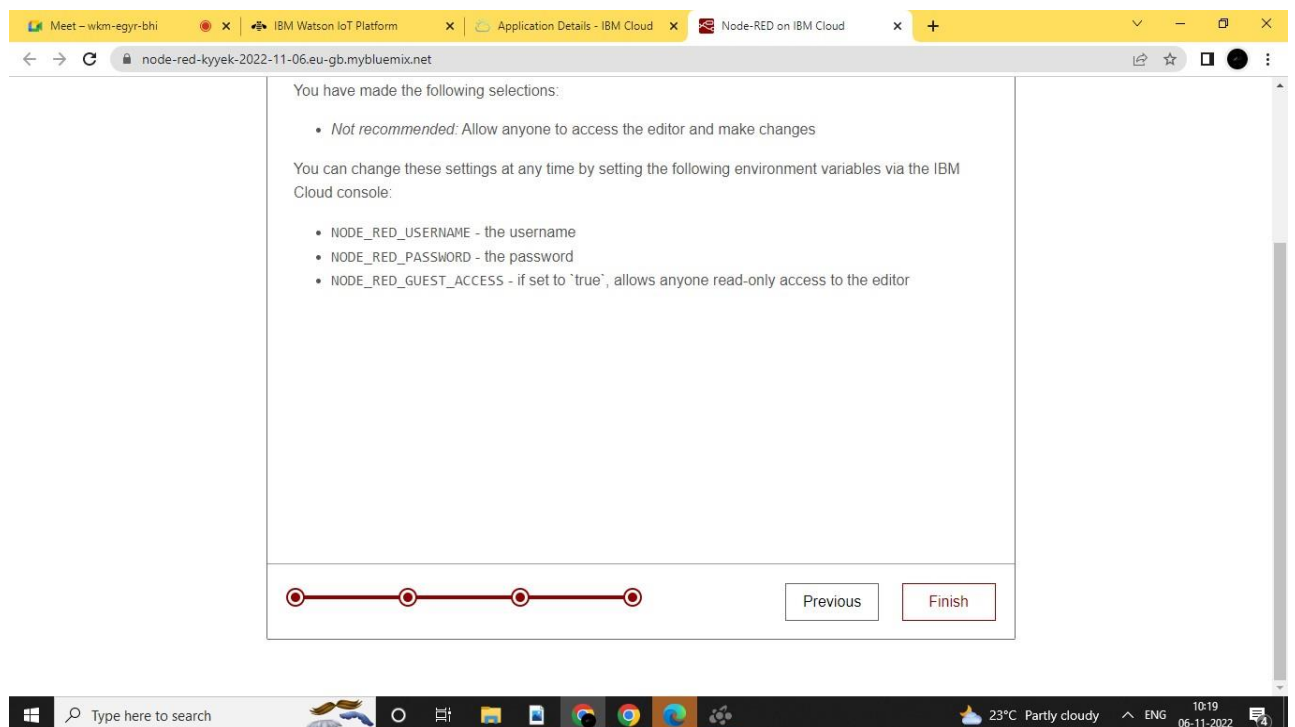
Step 3:



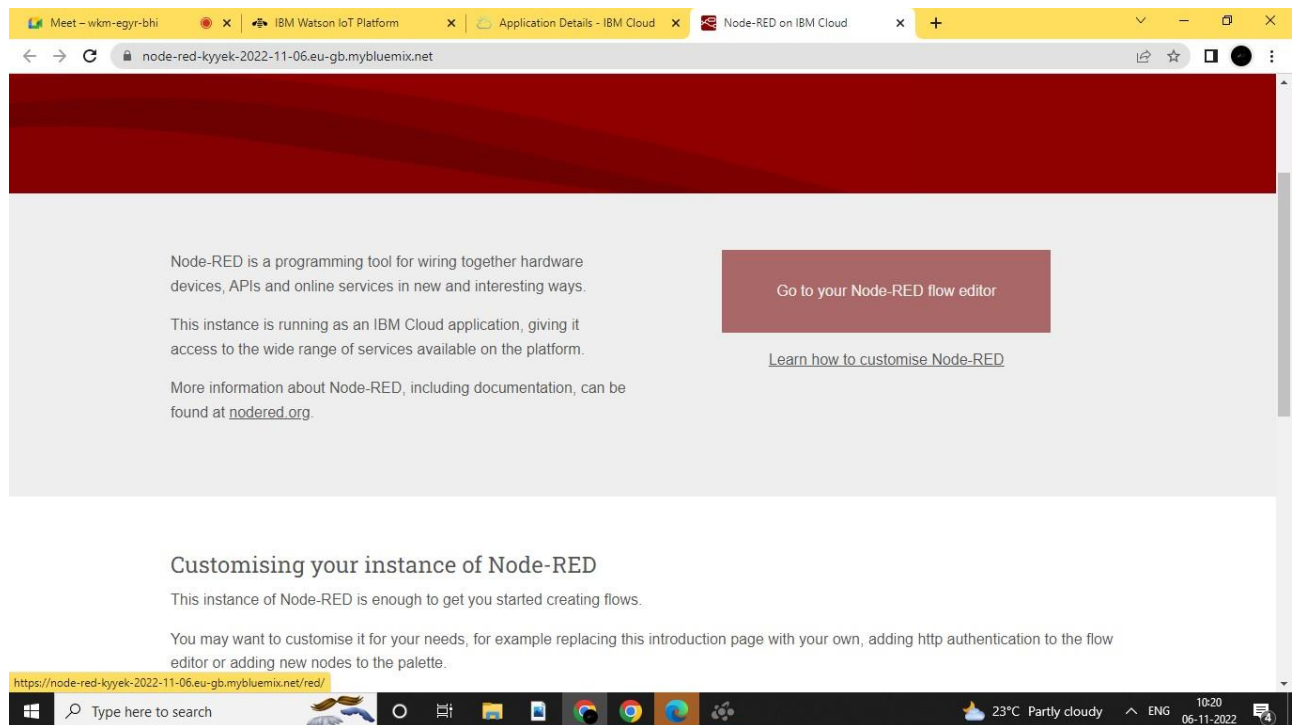
Step 4:



Step 5:



Step 6:



The screenshot shows a web browser window with the URL `node-red-kyyek-2022-11-06.eu-gb.mybluemix.net`. The page has a red header and a light gray body. It contains the following text:

Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways.

This instance is running as an IBM Cloud application, giving it access to the wide range of services available on the platform.

More information about Node-RED, including documentation, can be found at nodered.org.

There is a red button that says "Go to your Node-RED flow editor" and a link that says "Learn how to customise Node-RED".

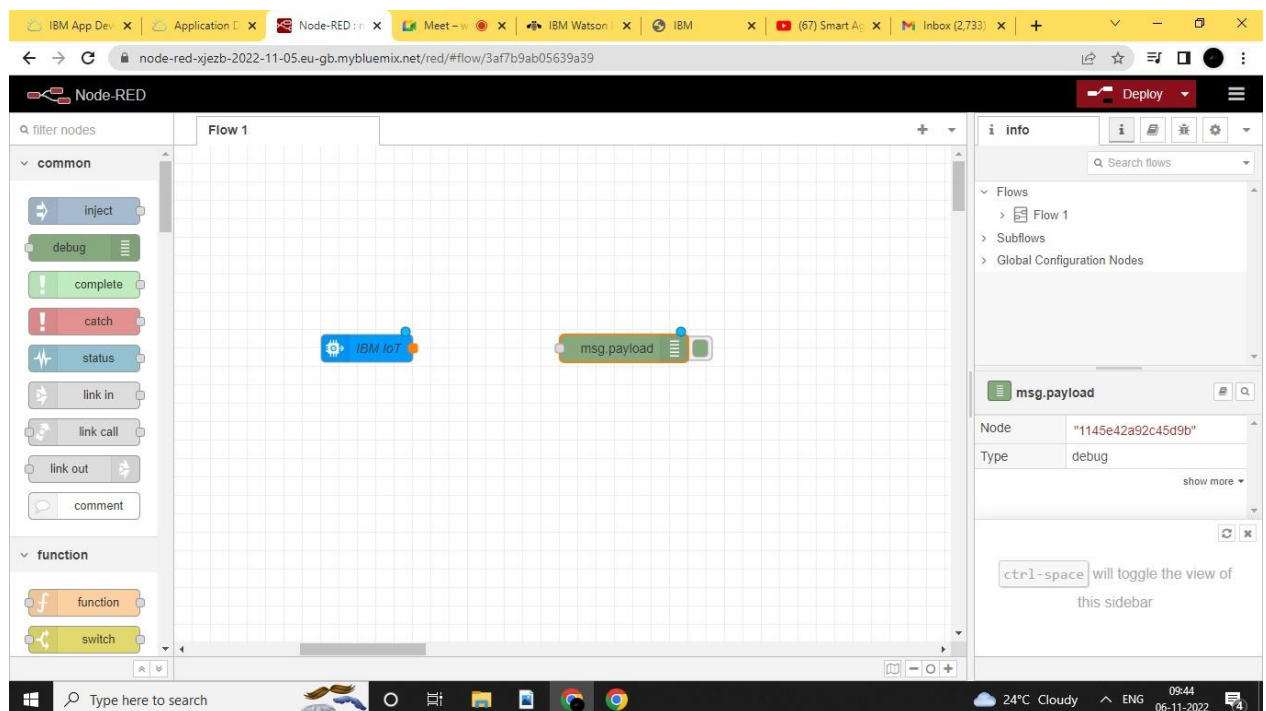
Below the introduction page, there is a section titled "Customising your instance of Node-RED" with the following text:

This instance of Node-RED is enough to get you started creating flows.

You may want to customise it for your needs, for example replacing this introduction page with your own, adding http authentication to the flow editor or adding new nodes to the palette.

The browser's address bar shows the URL `https://node-red-kyyek-2022-11-06.eu-gb.mybluemix.net/red/`. The Windows taskbar at the bottom shows the search bar and several application icons.

Step 7:



The screenshot shows the Node-RED flow editor interface in a web browser. The URL is `node-red-xjezb-2022-11-05.eu-gb.mybluemix.net/red/#flow/3af7b9ab05639a39`. The interface includes a left sidebar with a "filter nodes" search bar and two categories of nodes: "common" and "function". The "common" category includes nodes like "inject", "debug", "complete", "catch", "status", "link in", "link call", "link out", and "comment". The "function" category includes "function" and "switch".

The main workspace, titled "Flow 1", contains a flow with two nodes: "IBM IoT" and "msg.payload".

The right sidebar shows the "info" tab for the selected "msg.payload" node. It displays the node's ID ("1145e42a92c45d9b") and type ("debug"). Below this, there is a text box that says "ctrl-space will toggle the view of this sidebar".

The browser's address bar shows the URL `node-red-xjezb-2022-11-05.eu-gb.mybluemix.net/red/#flow/3af7b9ab05639a39`. The Windows taskbar at the bottom shows the search bar and several application icons.

Step 8:

The screenshot shows the Node-RED web interface in a browser. The flow consists of an **IBM IoT** node (labeled 'connected') connected to two **function** nodes. These function nodes are connected to a **msg payload** node. The debug console on the right displays a series of messages received from the IoT node, each containing a timestamp, node ID, and a JSON payload with temperature and humidity data.

```
11/6/2022, 9:43:41 AM node: f2f2649a.0d0d98  
iot-2/type/NodeMCU/id/24680/ev/1/fmt/json :  
msg.payload : Object  
  { Temperature: 45, Humidity: 28 }  
  
11/6/2022, 9:43:44 AM node: f2f2649a.0d0d98  
iot-2/type/NodeMCU/id/24680/ev/1/fmt/json :  
msg.payload : Object  
  { Temperature: 90, Humidity: 67 }  
  
11/6/2022, 9:43:47 AM node: f2f2649a.0d0d98  
iot-2/type/NodeMCU/id/24680/ev/1/fmt/json :  
msg.payload : Object  
  { Temperature: 78, Humidity: 82 }  
  
11/6/2022, 9:43:47 AM node: f2f2649a.0d0d98  
iot-2/type/NodeMCU/id/24680/ev/1/fmt/json :  
msg.payload : Object  
  { Temperature: 98, Humidity: 43 }  
  
11/6/2022, 9:43:50 AM node: f2f2649a.0d0d98  
iot-2/type/NodeMCU/id/24680/ev/1/fmt/json :  
msg.payload : Object  
  { Temperature: 1, Humidity: 55 }  
  
11/6/2022, 9:43:53 AM node: f2f2649a.0d0d98  
iot-2/type/NodeMCU/id/24680/ev/1/fmt/json :  
msg.payload : Object  
  { Temperature: 32, Humidity: 35 }
```

Step 9:

The screenshot shows the Node-RED web interface with the **Edit function node** dialog open for the 'Humidity' function. The dialog displays the function code: `1 msg.payload=msg.payload.Humidity` and `2 return msg;`. The debug console on the right displays a series of messages received from the IoT node, each containing a timestamp, node ID, and a JSON payload with temperature and humidity data.

```
11/6/2022, 9:44:16 AM node: f2f2649a.0d0d98  
iot-2/type/NodeMCU/id/24680/ev/1/fmt/json :  
msg.payload : Object  
  { Temperature: 23, Humidity: 47 }  
  
11/6/2022, 9:44:19 AM node: f2f2649a.0d0d98  
iot-2/type/NodeMCU/id/24680/ev/1/fmt/json :  
msg.payload : Object  
  { Temperature: 73, Humidity: 1 }  
  
11/6/2022, 9:44:22 AM node: f2f2649a.0d0d98  
iot-2/type/NodeMCU/id/24680/ev/1/fmt/json :  
msg.payload : Object  
  { Temperature: 47, Humidity: 84 }  
  
11/6/2022, 9:44:25 AM node: f2f2649a.0d0d98  
iot-2/type/NodeMCU/id/24680/ev/1/fmt/json :  
msg.payload : Object  
  { Temperature: 15, Humidity: 90 }  
  
11/6/2022, 9:44:28 AM node: f2f2649a.0d0d98  
iot-2/type/NodeMCU/id/24680/ev/1/fmt/json :  
msg.payload : Object  
  { Temperature: 85, Humidity: 18 }
```

Step 10:

The screenshot shows the Node-RED web interface. On the left, the 'common' node palette is visible. The central workspace contains a flow with an 'IBM IoT' node connected to a function node. The 'Edit function node' panel is open, showing the following code:

```
1 msg.payload=msg.payload.Temperature
2 return msg;
```

The 'debug' console on the right displays the following log messages:

```
msg payload : Object
{ Temperature: 45, Humidity: 28 }
11/6/2022, 9:43:41 AM node: f2f2649a.0d0d98
iot-2/type/NodeMCUID/24680/evl/event_1/fmt/json :
msg payload : Object
{ Temperature: 90, Humidity: 67 }
11/6/2022, 9:43:44 AM node: f2f2649a.0d0d98
iot-2/type/NodeMCUID/24680/evl/event_1/fmt/json :
msg payload : Object
{ Temperature: 78, Humidity: 82 }
11/6/2022, 9:43:47 AM node: f2f2649a.0d0d98
iot-2/type/NodeMCUID/24680/evl/event_1/fmt/json :
msg payload : Object
{ Temperature: 98, Humidity: 43 }
11/6/2022, 9:43:50 AM node: f2f2649a.0d0d98
iot-2/type/NodeMCUID/24680/evl/event_1/fmt/json :
msg payload : Object
{ Temperature: 1, Humidity: 55 }
11/6/2022, 9:43:53 AM node: f2f2649a.0d0d98
iot-2/type/NodeMCUID/24680/evl/event_1/fmt/json :
msg payload : Object
{ Temperature: 32, Humidity: 35 }
```

Step 11:

The screenshot shows the Node-RED web interface. The 'User Settings' panel is open, displaying a list of nodes available for installation. The 'debug' console on the right displays the following log messages:

```
msg payload : Object
{ Temperature: 34, Humidity: 3 }
11/6/2022, 9:54:49 AM node: f2f2649a.0d0d98
iot-2/type/NodeMCUID/24680/evl/event_1/fmt/json :
msg payload : number
34
11/6/2022, 9:54:49 AM node: f2f2649a.0d0d98
iot-2/type/NodeMCUID/24680/evl/event_1/fmt/json :
msg payload : number
3
11/6/2022, 9:54:52 AM node: f2f2649a.0d0d98
iot-2/type/NodeMCUID/24680/evl/event_1/fmt/json :
msg payload : Object
{ Temperature: 96, Humidity: 44 }
11/6/2022, 9:54:52 AM node: f2f2649a.0d0d98
iot-2/type/NodeMCUID/24680/evl/event_1/fmt/json :
msg payload : number
96
11/6/2022, 9:54:52 AM node: f2f2649a.0d0d98
iot-2/type/NodeMCUID/24680/evl/event_1/fmt/json :
msg payload : number
44
```


Step 12:

The screenshot shows the Node-RED web interface in a browser. The main workspace displays a flow named 'Flow 1'. It starts with an 'IBM IoT' node (labeled 'connected') that branches into two function nodes: 'Temperature' and 'Humidity'. The 'Temperature' node is connected to a 'msg payload' node, which is then connected to a 'gauge' node. Similarly, the 'Humidity' node is connected to the same 'msg payload' node, which is also connected to another 'gauge' node. The left sidebar shows a list of available nodes, including numeric, dropdown, switch, text input, date picker, colour picker, form, text, gauge, chart, audio out, notification, ui control, and template. The right sidebar shows a 'debug' console with a log of messages, including a JSON object: { Temperature: 96, Humidity: 44 }.

Step 13:

The screenshot shows the Node-RED web interface with the 'Edit gauge node' dialog box open for the 'Temperature' gauge. The dialog box has a 'Properties' section with the following settings: 'Group' is '[Temperature] Smart Farming', 'Size' is 'auto', 'Type' is 'Gauge', 'Label' is 'Temperature', 'Value format' is '{value}', 'Units' is '°C', 'Range' is 'min 0 max 100', 'Colour gradient' is a gradient from green to yellow to red, 'Sectors' are '0', 'optional', 'optional', '100', and 'Class' is 'Optional CSS class name(s) for widget'. The 'Enabled' checkbox is checked. The background shows the same flow as in Step 12, but the 'msg payload' node is now connected to the 'Temperature' gauge. The right sidebar shows the 'debug' console with the same log of messages.

Step 14:

The screenshot shows the Node-RED web interface in a browser. The main workspace displays a flow named 'Flow 1' with an 'IBM IoT' node (status: connected) connected to two nodes: 'Temperature' and 'Humidity'. The 'Edit gauge node' panel is open on the right, showing the configuration for the 'Humidity' gauge. The properties are as follows:

- Group: [Temperature] Smart Farming
- Size: auto
- Type: Gauge
- Label: Humidity
- Value format: {{value}}
- Units: gal
- Range: min 0, max 100
- Colour gradient: Green, Yellow, Red
- Sectors: 0, optional, optional, 100
- Class: Optional CSS class name(s) for widget
- Enabled: ☐

The debug console on the right shows the following messages:

```
msg.payload : Object
{ Temperature: 96, Humidity: 44 }
11/6/2022, 9:54:52 AM node: f2f2649a.0d0d98
iot-2/type/NodeMCUID/24680/evt/event_1/fmt/json :
msg.payload : number
96
11/6/2022, 9:54:52 AM node: f2f2649a.0d0d98
iot-2/type/NodeMCUID/24680/evt/event_1/fmt/json :
msg.payload : number
44
11/6/2022, 9:56:14 AM node: f2f2649a.0d0d98
iot-2/type/NodeMCUID/24680/evt/event_1/fmt/json :
msg.payload : Object
{ Temperature: 15, Humidity: 95 }
11/6/2022, 9:56:14 AM node: f2f2649a.0d0d98
iot-2/type/NodeMCUID/24680/evt/event_1/fmt/json :
msg.payload : number
15
11/6/2022, 9:56:14 AM node: f2f2649a.0d0d98
iot-2/type/NodeMCUID/24680/evt/event_1/fmt/json :
msg.payload : number
95
```

Step 15:

The screenshot shows a Google search page in a browser. The address bar displays the URL <https://node-red-xjezb-2022-11-05.eu-gb.mybluemix.net/ui>. The search bar is empty, and the Google logo is visible. The page shows the standard Google search interface with a search bar, a search button, and a 'Search Google or type a URL' prompt. Below the search bar are links for 'Google', 'Web Store', and 'Add shortcut'. The bottom of the page shows the Windows taskbar with the search bar and system tray.

Step 16:

