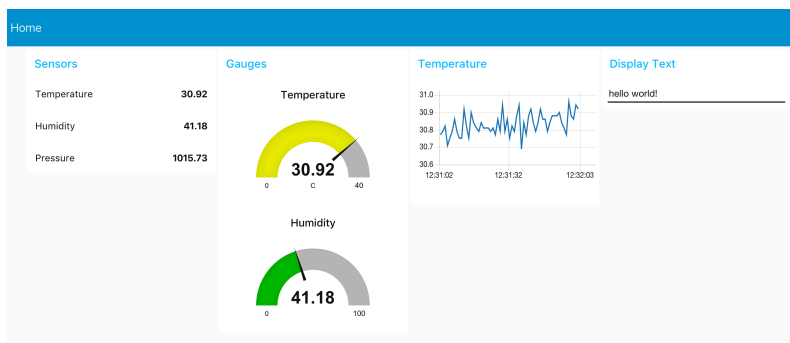
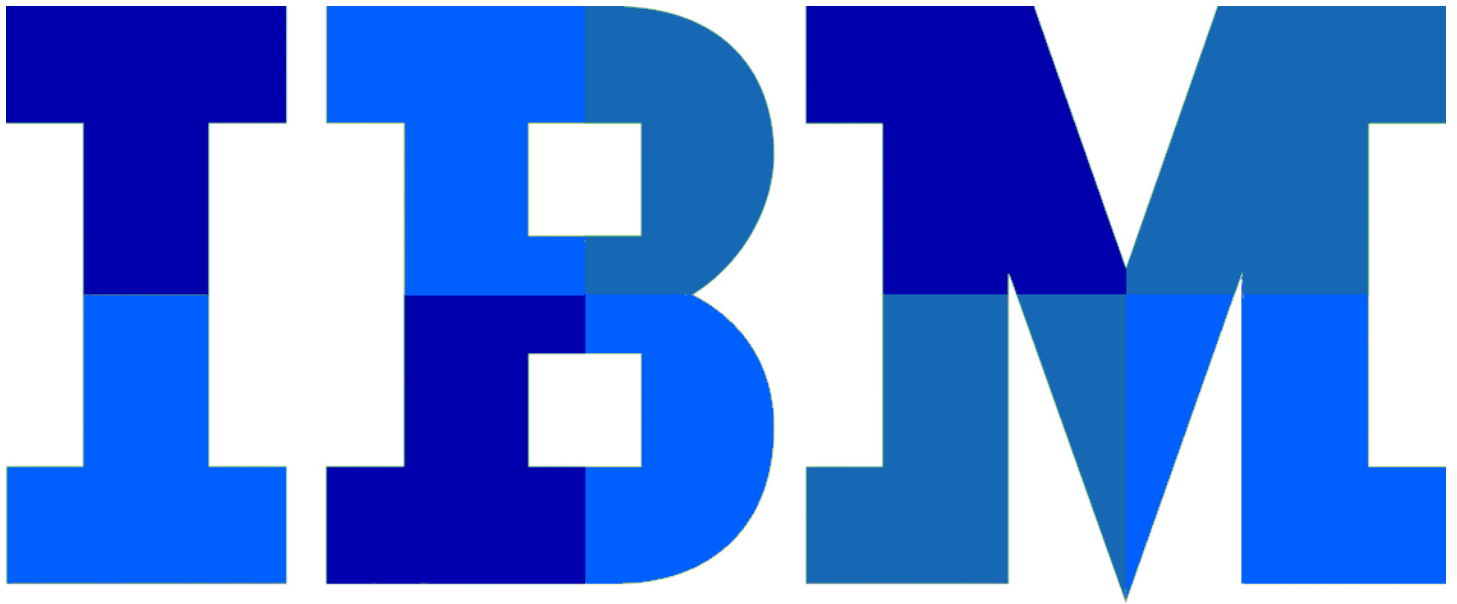


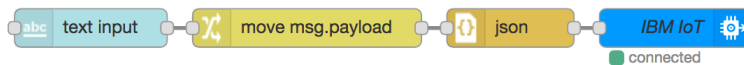
Raspberry Pi Sense HAT

Part 4: Commanding the Sense HAT via Watson IoT

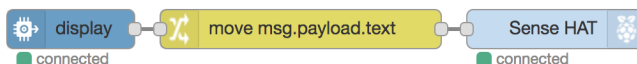
Author: JeanCarl Bisson | jbisson@us.ibm.com | [@dothewww](https://twitter.com/dothewww)



Add a text input element to the Node-RED dashboard to capture messages



Send message as an IoT Command



Receive an IoT Command and control Sense HAT to display message



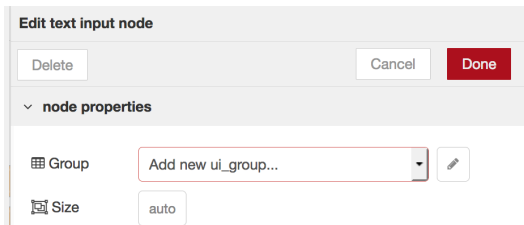
A digital copy of this lab and completed flows can be found at:
<http://ibm.biz/sense-hat-commands>



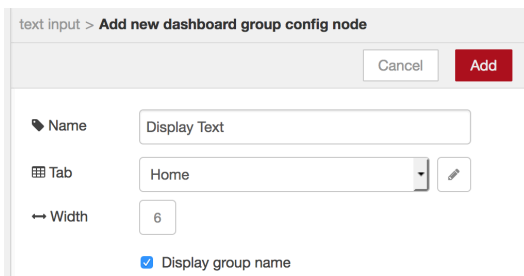
Add a Text Input in Node-RED

In this lab, we'll use Node-RED to create a text input to capture text from the user. We'll send that text as an IoT command via the Watson IoT Platform to the Raspberry Pi. The Raspberry Pi, also running Node-RED, will receive the command and display the text via the 8x8 LED display on the Sense HAT. In this section, we'll start by creating a new UI group in the Node-RED dashboard running in IBM Bluemix. If you haven't completed the previous part of the lab to install the Node-RED dashboard nodes, please complete the section titled **Install Node-RED Dashboard**.

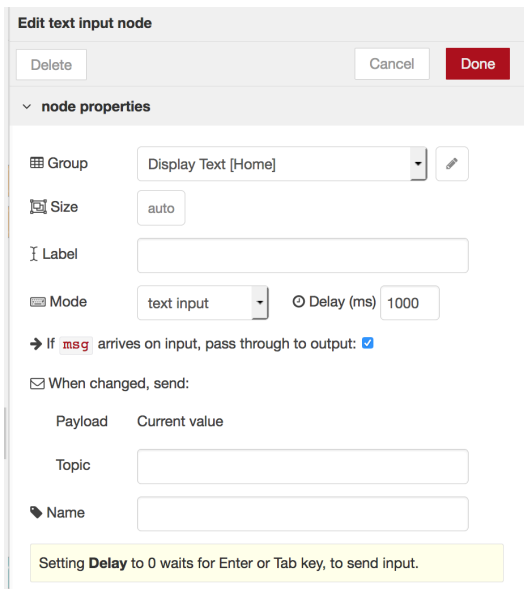
1. Add a  node. Select **Add new ui_group...** from the **Group** drop-down menu. Click on the pencil icon to configure the new group.



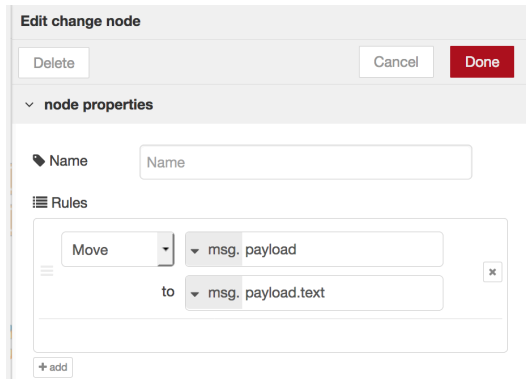
2. Name the new dashboard group **Display Text** as shown below. Click **Add**.




3. Customize the text input node as shown below. When finished, click **Done**.



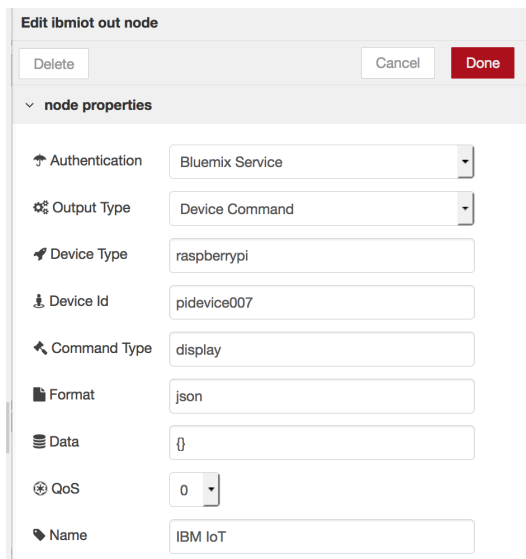
4. Add a  change node as shown below.



The 'Edit change node' dialog shows the 'node properties' section. Under 'Rules', a rule is defined: 'Move' from 'msg.payload' to 'msg.payload.text'. Buttons for 'Delete', 'Cancel', and 'Done' are at the top.

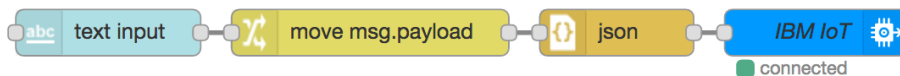
5. Add a  json node to convert the payload object into a JSON string.

6. Add a  ibmiot node as shown below.

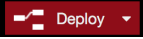


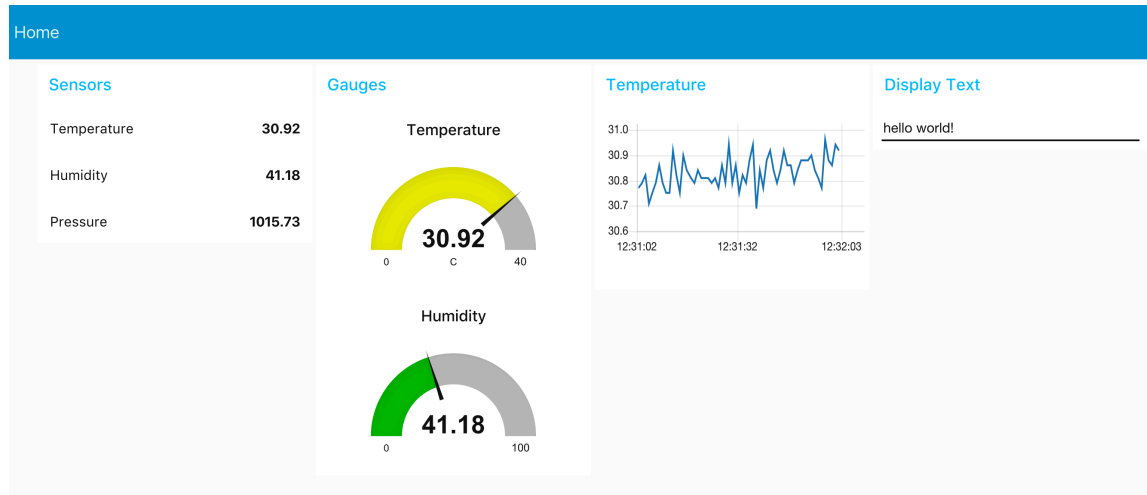
The 'Edit ibmiot out node' dialog shows the 'node properties' section. Fields include: Authentication (Bluemix Service), Output Type (Device Command), Device Type (raspberrypi), Device Id (pidevice007), Command Type (display), Format (json), Data ({}), QoS (0), and Name (IBM IoT). Buttons for 'Delete', 'Cancel', and 'Done' are at the top.

7. Connect the nodes as shown below.



Get the code:
`ibm.biz/BdiW8d`

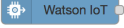
8. Click on  to save and deploy your changes. The dashboard now displays a text input that will be used to send an IoT command to the IoT Platform.

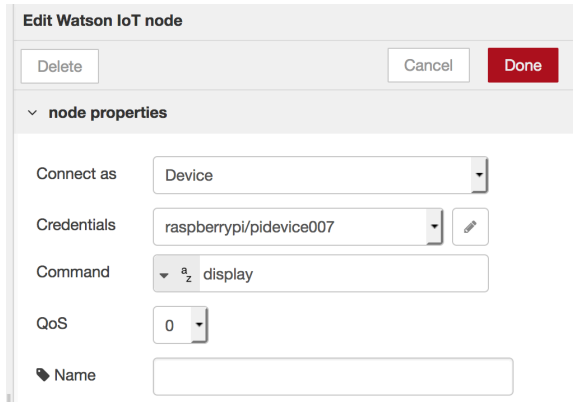


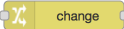
In the next section, we'll modify the Raspberry Pi to subscribe to this command and take action.

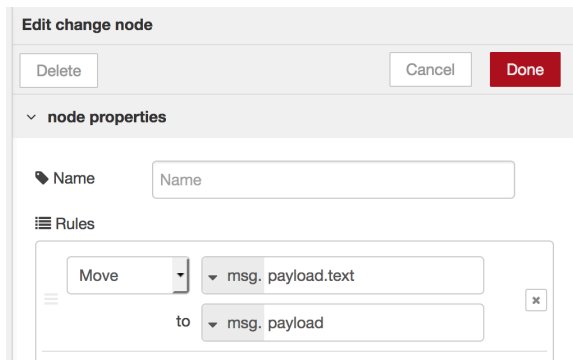
Subscribe to Commands in Node-RED

In this section, we'll create a flow that subscribes to the display command in Node-RED running on the Raspberry Pi, and then command the Sense HAT display to scroll the message on the 8x8 LED display.

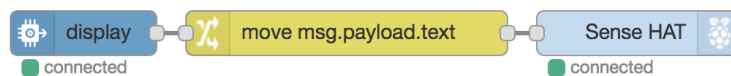
1. Add a  node. This node will subscribe to the `display` commands for the device we created previously.



2. Add a  node as shown below. It moves the contents of the command to the `msg.payload` so the Sense HAT node in the next step can use it.



3. Add a  node. Connect the nodes together as shown.



Get the code:
ibm.biz/BdiW8R

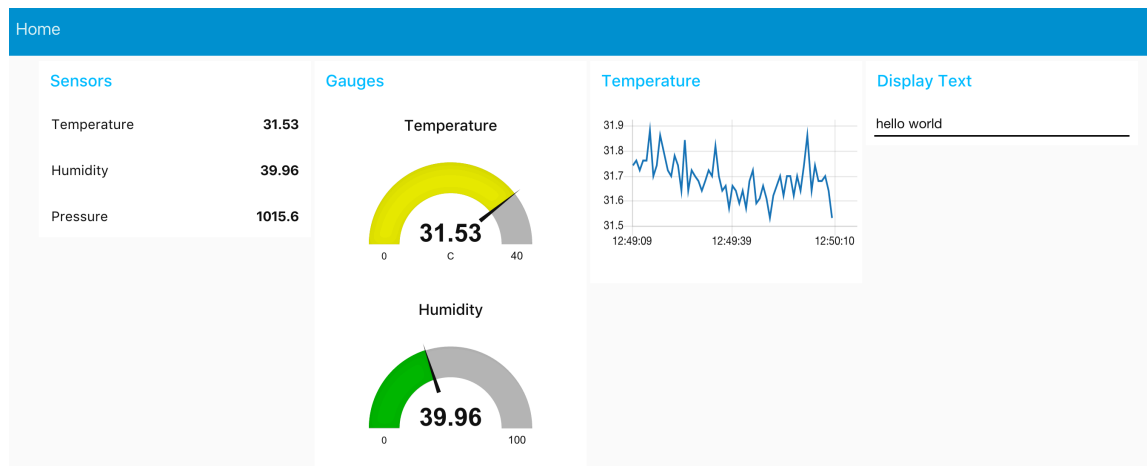
4. Click on  to save and deploy your changes.

In the next section, we'll send a command to display text on the 8x8 LED display.

Commanding the Sense HAT

In the previous two sections, we created a UI to capture text and converted it into an IoT command. The Raspberry Pi subscribes to these IoT commands and relays the messages to the Sense HAT. In this section, we'll send a couple of different messages to command the 8x8 LED display to show. The first is to scroll a message.

1. Return to the Node-RED dashboard in IBM Bluemix. Enter text into the text input. Press **ENTER**.



When the Raspberry Pi receives the command, it sends the message to the Sense HAT node which scrolls the text on the 8x8 LED display.

2. You can also change the color of the individual pixels with the format: `<x>,<y>,<color>`

As the documentation for the Sense HAT node explains:

Set the colour of individual pixels

Format: `<x>,<y>,<colour>`

`x` and `y` must either be a value from 0 to 7, a `*` to indicate the entire row or column, or a range such as `3-6`.

`colour` must be one of:

- the well-known HTML colour names - eg `red` or `aquamarine`,
- the CheerLights colour names,
- a HEX colour value - eg `#aa9900`
- an RGB triple - `190,255,0`
- or simply `off`

To set the entire screen to red: `*,*,red`

To set the four corners of the display to red, green (`#00ff00`), yellow and blue (`0,0,255`):

`0,0,red,0,7,#00ff00,7,7,yellow,7,0,0,0,255`

To set a 3-pixel wide column to purple: `4-6,*,purple`

Send the following commands to see what each one does.

```
*,*,red
```

```
0,0,red,0,7,#00ff00,7,7,yellow,7,0,0,0,255
```

```
4-6,*,purple
```

3. If you feel adventurous, you can customize the text color, background, and speed of the scrolling text. This exercise is left for you to complete. However, here are a couple of hints. Add additional text inputs (or even other UI elements like a dropdown) in the Node-RED dashboard and send those properties in the IoT command. On the Raspberry Pi, modify the change node to move the values to the appropriate msg properties for the Sense HAT node to use. Here's the Sense HAT documentation:

Scroll a message

If `msg.payload` is not recognised as any of the above commands, it is treated as a text message to be scrolled across the screen.

If the text is a single character, it will be displayed without scrolling. To scroll a single character, append a blank space after it - "A ".

The following message properties can be used to customise the appearance:

- `msg.color` - the colour of the text, default: `white`
- `msg.background` - the colour of the background, default: `off`
- `msg.speed` - the scroll speed. A value in the range 1 (slower) to 5 (faster), default: `3`

What's Next?

This lab demonstrated how you can manually send commands to an IoT device. However, you can automate the process! For example, using the IoT Starter lab, send an IoT command when the temperature is "dangerous" or higher than the temperature threshold and change the color of the 8x8 LED display to red to signal an issue.

Use outside factors to make decisions on what should be done in certain scenarios. The Weather Company Data for IBM Bluemix API provides weather observations and forecasts. IBM Watson analyzes natural language text (analyze emotions using Tone Analyzer), audio (capture audio via a microphone and use Speech to Text to transcribe what is being said, or play audio via a speaker with Text to Speech to play audio alerts), and images (take a photo using a Raspberry Pi camera and recognize objects with Visual Recognition).

For more labs using Node-RED, please visit ibm.biz/node-red-labs