

Introduction to IoT: Build Your Own Basic IoT App

Watson Developer Conference Hands-On Workshop

Download this PDF and Node-RED flows at :

<https://github.com/johnwalicki/WatsonDevConf-IoT-Lab>

Author:

John Walicki | walicki@us.ibm.com

| @johnwalicki

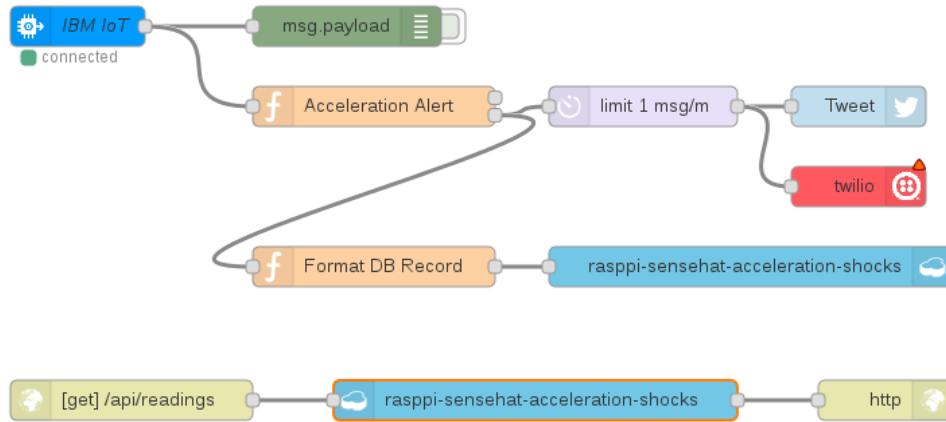


Table of Contents

Topics covered: Raspberry Pi, Node-RED, Watson Internet of Things Platform, SenseHAT, Twitter, Sentiment and Weather Company Data for IBM Bluemix.

In this lab, we will unbox and set up a Raspberry Pi and Raspberry Pi SenseHat and connect to the IBM Bluemix and Watson IoT Platform. We will query the temperature via the SenseHat and send data to the Watson IoT platform. Watson IoT Platform will report the temperature and alert maintenance of a high temperature. Using Node-RED, running on the Raspberry Pi and in Bluemix, the application will analyze the sentiment of a Twitter feed. The sentiment scores will be sent via Watson's Internet of Things Platform service from a Node-RED application hosted on IBM Bluemix to the local Raspberry Pi. The SenseHAT LED display will flash a color corresponding to the sentiment. We will retrieve outside weather data and display it on a LED screen connected to the SenseHat.

Raspberry Pi Set Up.....	3
Send data from the Raspberry Pi to Watson IoT.....	6
Local Node-RED Temperature Alert.....	10
Send Sensor data to a Bluemix Application.....	11
Send Twitter Feed Sentiment Alerts to a Raspberry Pi.....	16
Send Weather Alerts to a Raspberry Pi.....	21
Additional Raspberry Pi / SenseHAT Resources.....	25



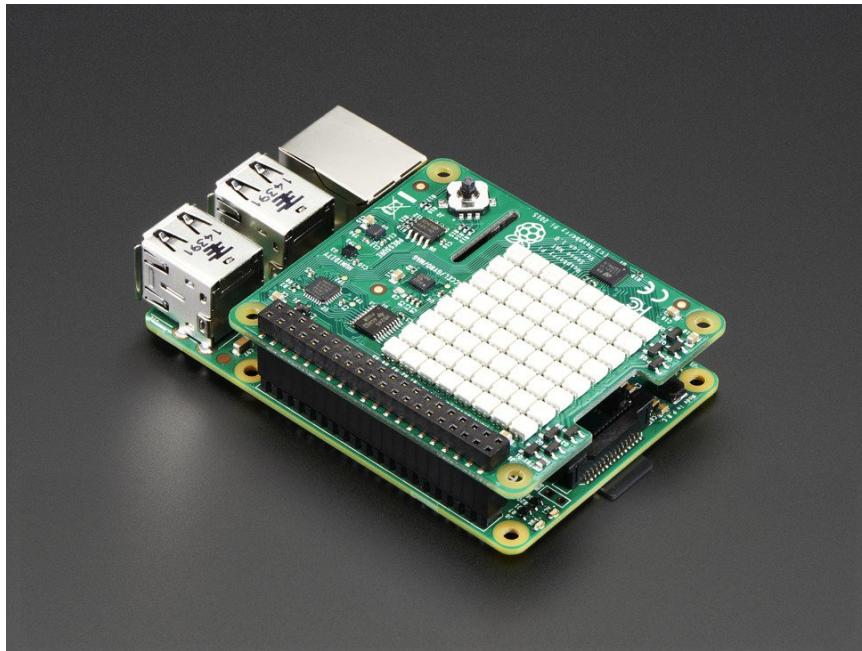
Lesson 1 – Raspberry Pi Set Up

Step 1 - Unbox the Raspberry Pi and Raspberry Pi SenseHAT

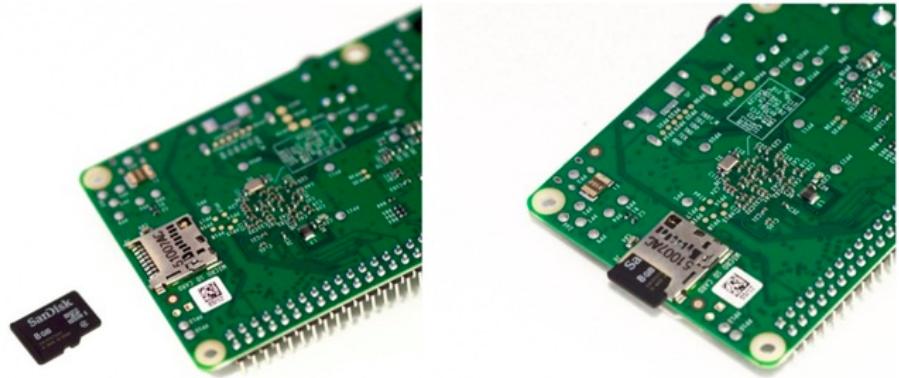


Step 2 – Connect the Raspberry Pi SenseHAT to the Raspberry Pi

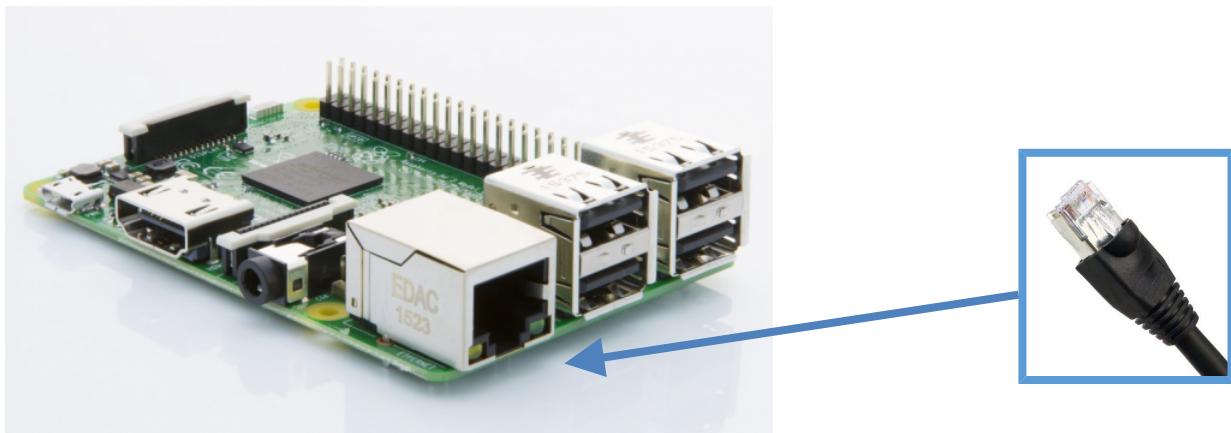
Connect your SenseHAT to the Raspberry Pi via the 40 GPIO Pins.



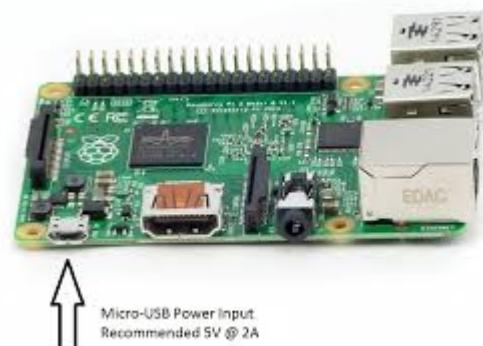
Step 3 - Insert the Raspbian Linux microSD card into the Raspberry Pi



Step 4 - Connect the Ethernet cable to the Raspberry



Step 5 - Power the Raspberry Pi via a microUSB Cable or 5v microUSB Power supply



Step 6 - Allow the Raspberry Pi to boot

The Raspberry Pi will display a red light if it is powered. After around 2 minutes, the IP addresses (Ethernet wired and WiFi) will scroll across the LED of the SenseHAT. The IP addresses will scroll for a few minutes and then turn off.

Write Down the IP address

Step 7 - Connect to the Raspberry Pi with a Terminal Client

In order to perform advanced configuration of the Raspberry Pi either a monitor and keyboard, or a Secure Shell (SSH) connection is required. On OSX and Linux there are default programs that can do this - Screen and SSH respectively. However on Windows no default exists, however PuTTY is light weight and easy to install and can be used to SSH into the Raspberry Pi.

For Windows Users:

- o PuTTY is installed on the lab laptops
- o or Visit the [PuTTY download page](#).
- o Under the "For Windows on Intel x86" heading, click on the "putty.exe" link to download the latest release version to your computer.



- o Double-click putty.exe on your computer to launch PuTTY.
- o Enter IP address of the Gateway
- o You can login with the username **pi** and the password **raspberry**

For Mac and Linux Users:

- o Open a Terminal
- o Type `$ ssh pi@<<IP Address>>`. Replace <<IP Address>> with the IP address of your gateway.
- o Enter **raspberry** as password

Step 8 - Open a Browser and enter `http://<ip-address>:1880`

The Node-RED graphical programming tool will open in your browser.

Lesson 2 - Send data from the Raspberry Pi to Watson IoT

The Raspberry Pi Raspbian image provided in the Watson Developer Conference lab has a program installed and running that is automatically sending the CPU temperature of your Raspberry Pi to the IBM Watson IoT Platform QuickStart. In the terminal window, enter the **following two commands marked below in bold** and copy the device ID it reports.

```
pi@raspberrypi:~ $ sudo service iot restart
```

```
pi@raspberrypi:~ $ service iot getdeviceid
```

The device ID is b827eb6effa7

For Real-time visualization of the data, visit

<http://quickstart.internetofthings.ibmcloud.com/?deviceId=b827eb6effa7>

Open a browser tab to <http://quickstart.internetofthings.ibmcloud.com/?deviceId=<device id>>

The screenshot shows the IBM Watson IoT Platform Quickstart interface. At the top, there is a navigation bar with links for QUICKSTART, SERVICE STATUS, DOCUMENTATION, and BLOG, along with a SIGN IN button. The main content area has a title "Quickstart" with a subtitle "No sign-up required to see how easy it is to connect your device to Watson IoT Platform and view live sensor data". A checkbox labeled "I accept IBM's Terms of Use" is checked. Below this, a text input field contains "b827eb6effa7" and a "Go" button is next to it. To the right, a message says "Last message received at 10:05:52 PM".

The central part of the screen features a line graph titled "myPi" with the Y-axis labeled "status.cpuTemp". The graph shows fluctuating data points over time, with a notable peak around 22:05:33. Below the graph is a table of events:

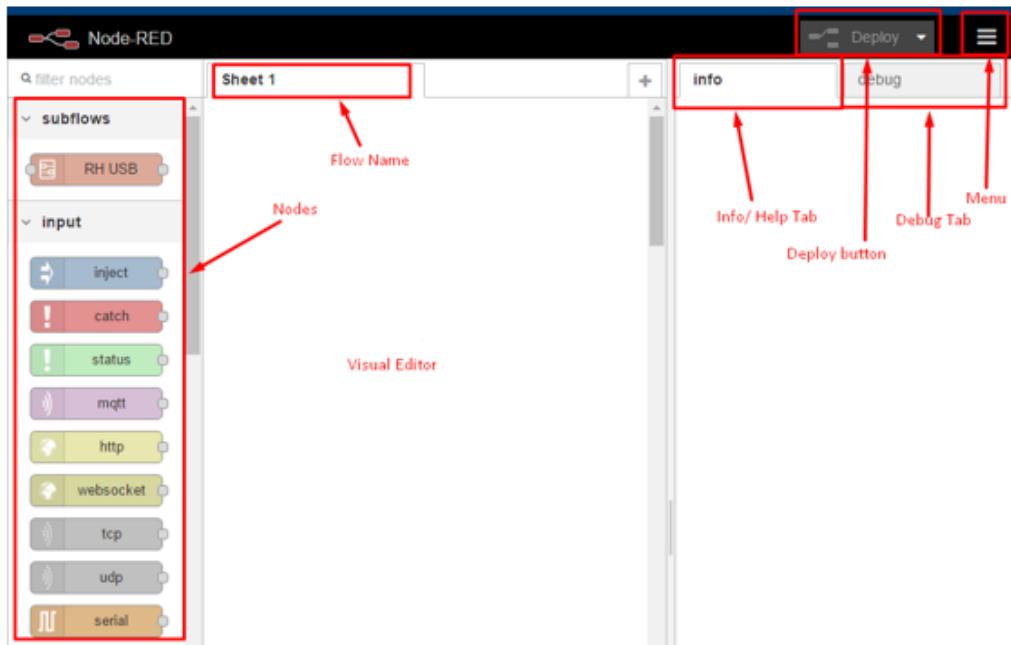
Event	Datapoint	Value	Time Received
status	myName	myPi	Oct 20, 2016 10:05:52 PM
status	cputemp	48.31	Oct 20, 2016 10:05:52 PM
status	cpuload	0.01	Oct 20, 2016 10:05:52 PM
status	sine	0.38	Oct 20, 2016 10:05:52 PM

On the right side, there is a sidebar with the heading "I've seen my data, what next?". It includes several options with icons and descriptions:

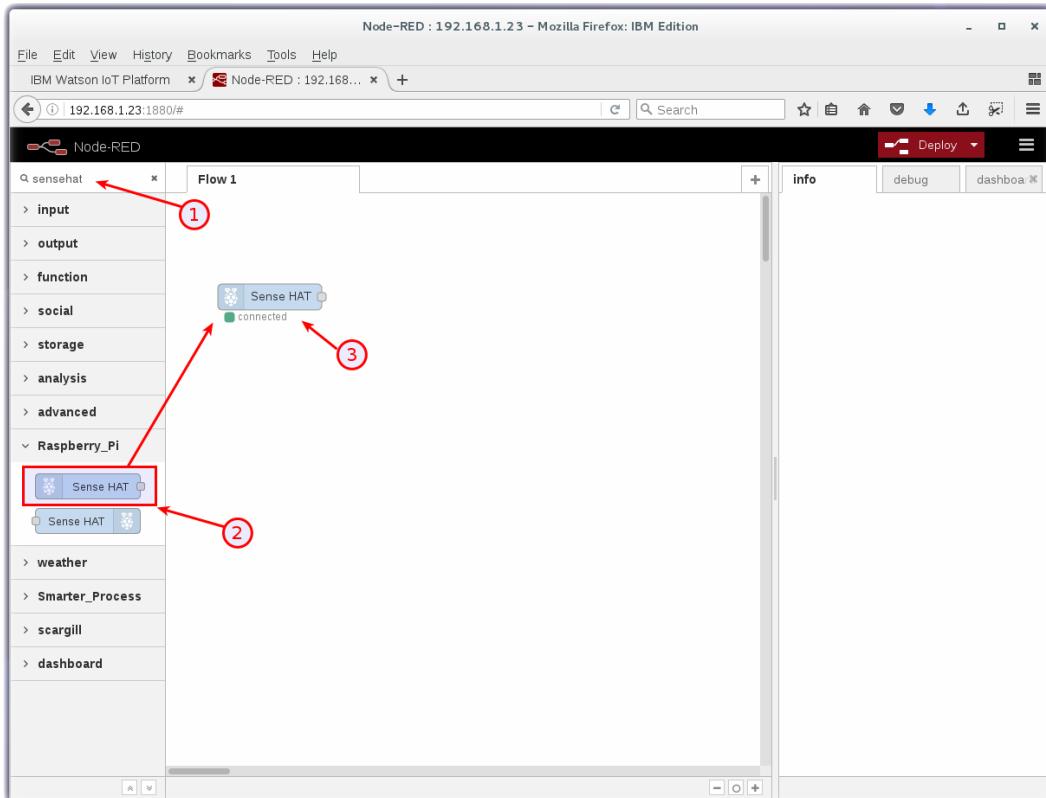
- "Use your device in an application created with IBM Bluemix." with a "Click here for more details." link.
- "Go to your Bluemix account" with "SIGN UP" and "LOG IN" buttons. A note says "Note: When you sign up for a trial you may have to wait up to 24 hours to receive your log-in information".
- "Create an app using the Internet of Things Starter from the Catalog" with a "CREATE APP" button. A note says "Note: You will have to name your app and wait for a few minutes for it to start running".
- "When your app is running, select the app URL or type it into the browser to open the Node-RED flow editor" with a URL "http://<appname>.mybluemix.net".
- "Import the flow for your device into the Node-RED flow editor" with a "IMPORT FLOW" button.

Sending SenseHAT data to IBM Watson IoT Platform Quickstart

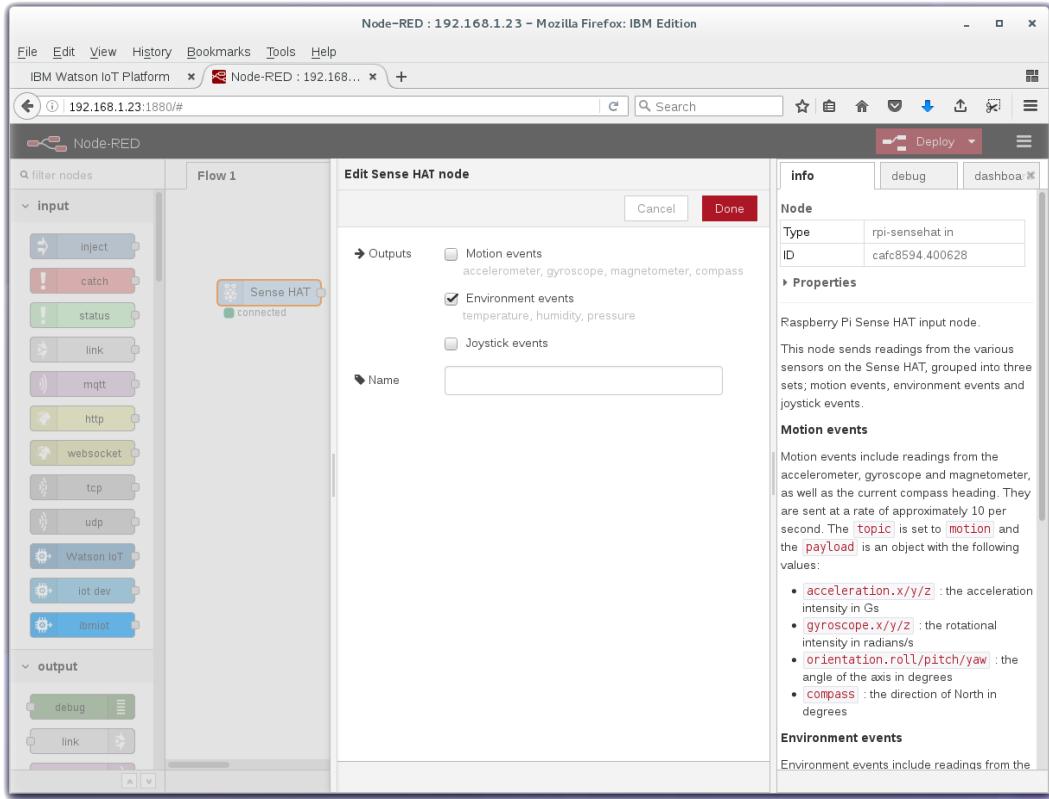
Now that we have sent the Raspberry Pi CPU temperature to Watson IoT, let's also send SenseHAT environmental sensor readings by using Node-RED. Create a new flow to program the rest of the exercises by clicking on the + button in the upper right corner of the Node-RED interface to open a Flow 1.



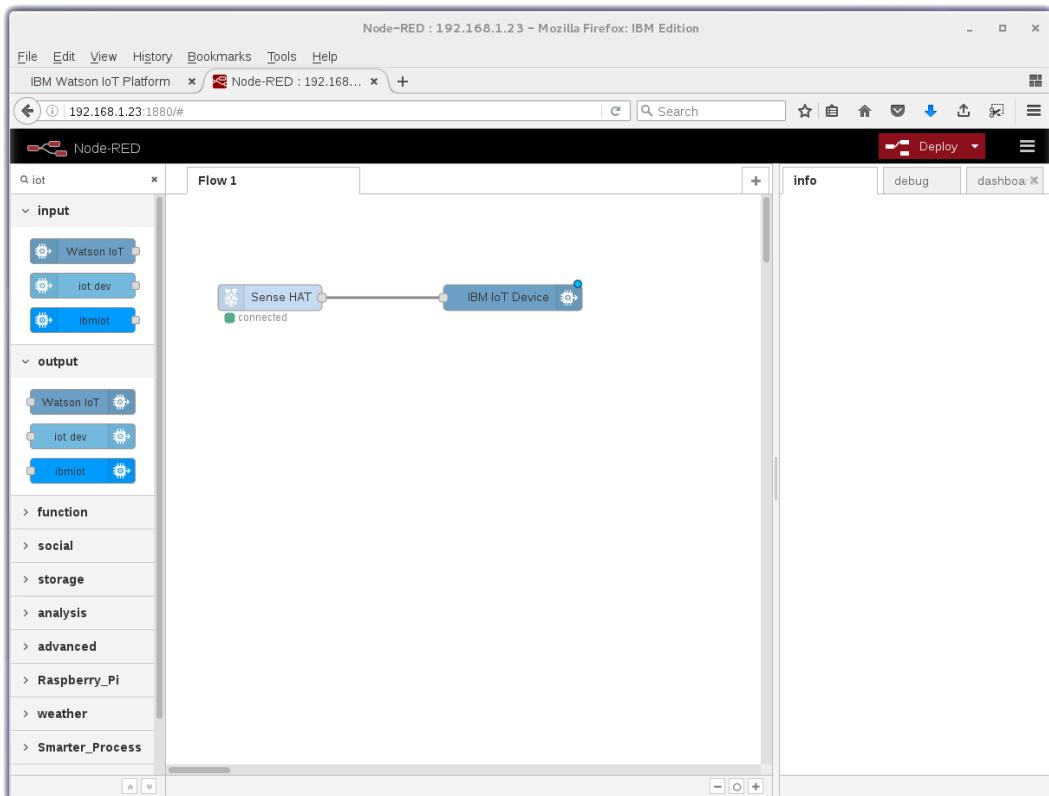
- Find a SenseHAT node by searching in the filter node box (1). Drag the SenseHAT node (2) from the palette onto the flow (3).



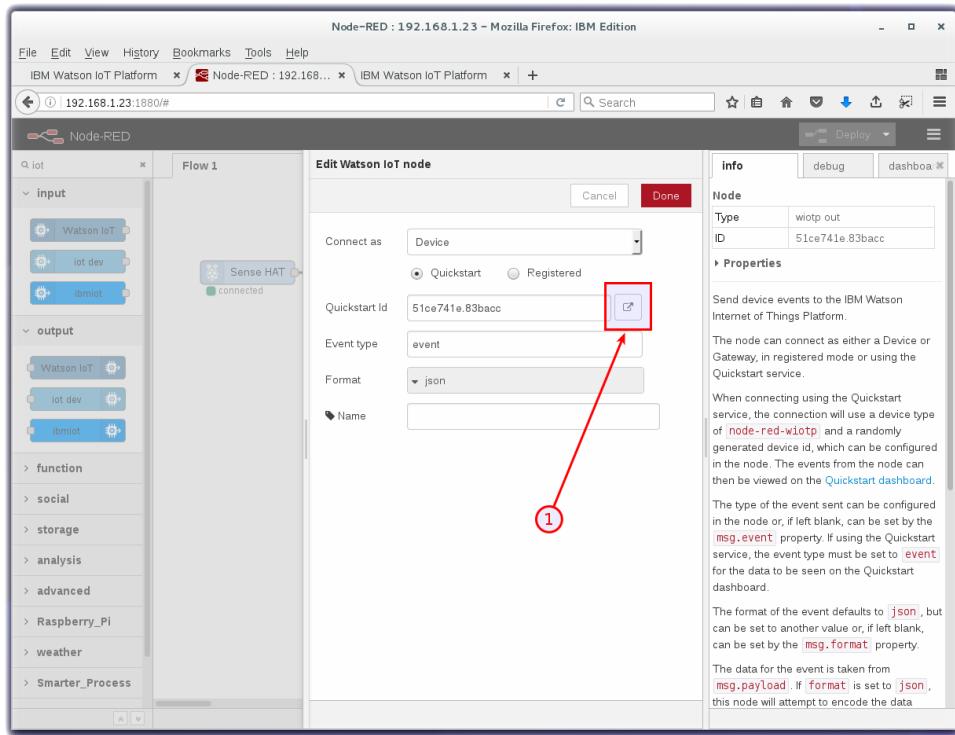
- Double click on the SenseHAT node on the Flow and select the “Environment events” checkbox and press the Done button.



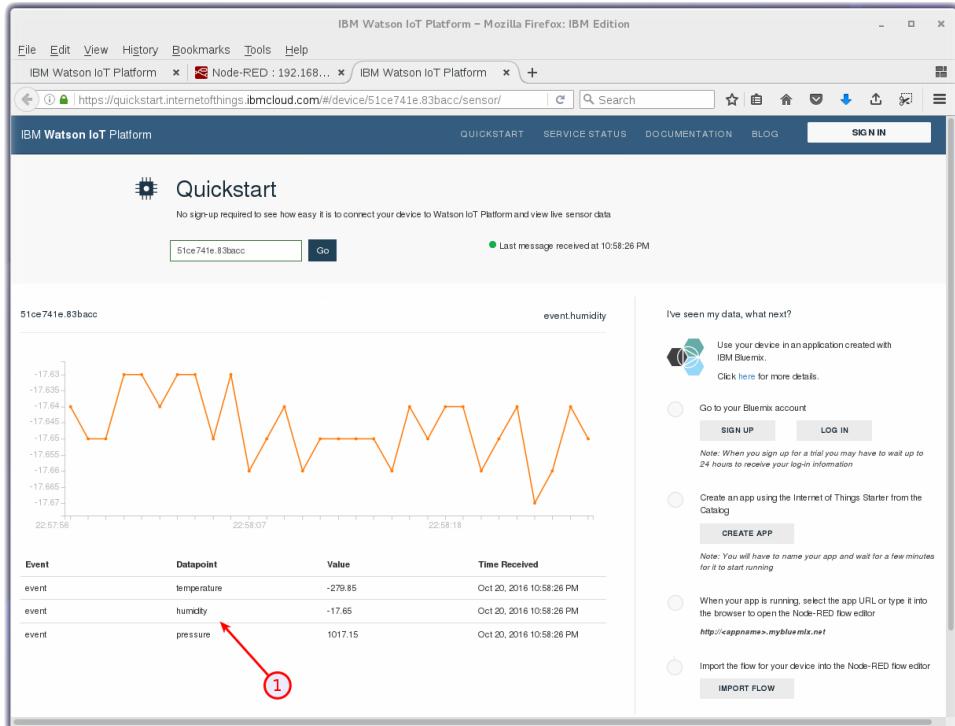
- Search for a Watson IoT output node on the palette and drag it onto the flow. Wire it to the SenseHAT node by dragging your mouse from one node point to the other. A wire will connect the two nodes.



- Click the Deploy  button on the top of menu bar to deploy the Node-RED flow.
- Doubleclick on the IBM IoT Device node and then click on the box (1) to the right of the Quickstart Id. A new browser tab will open. **Note the unique Quickstart Id**. This Id will be used in Lesson 4 and 5.

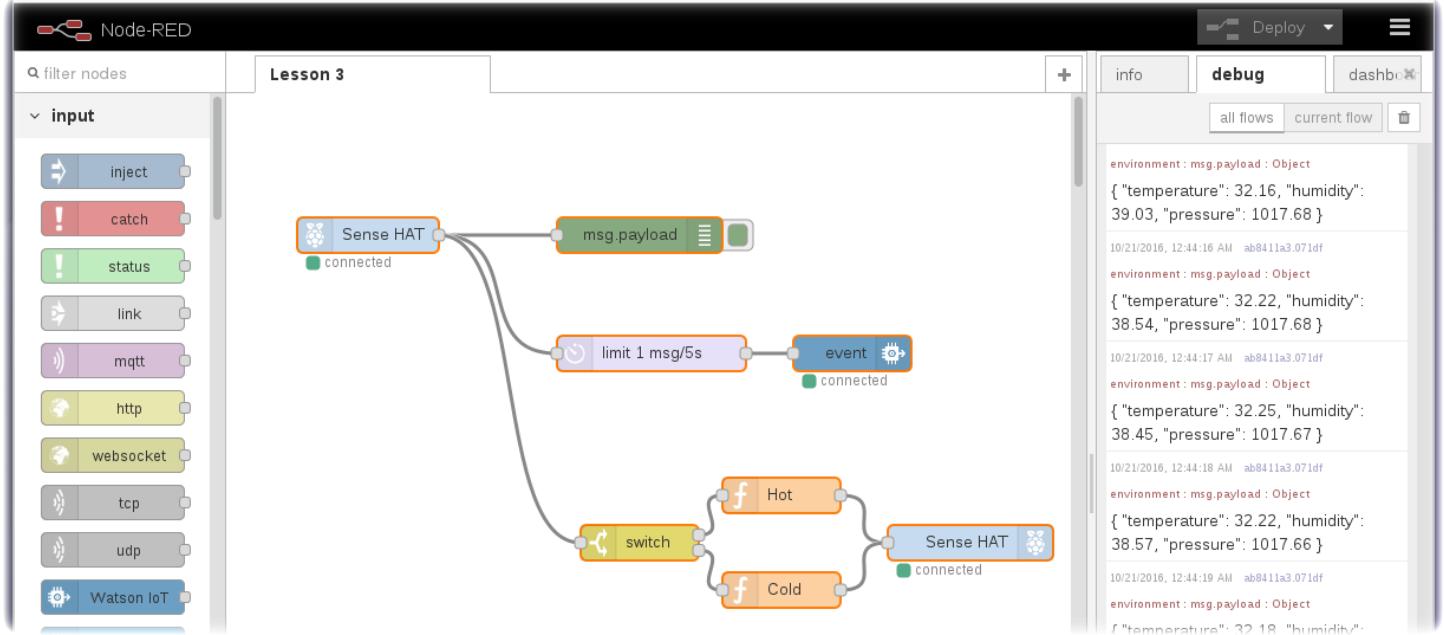


- Room temperature, humidity and air pressure sensor data from the SenseHAT is now being sent to the Watson IoT Platform and is being plotted in the Quickstart landing page. You can click (1) on any of the three datapoints to plot them.



Lesson 3 – Local Node-RED Temperature Alert

To practice with Node-RED visual programming, we might want to limit the amount of sensor data that is sent to Watson IoT Quickstart to 1 msg / 5 secs. It is also interesting to watch the SenseHAT environmental data by reporting it to a debug node before it is sent to Quickstart. Depending on the temperature, display if the room is hot or cold by changing the SenseHAT LEDs to red / green.



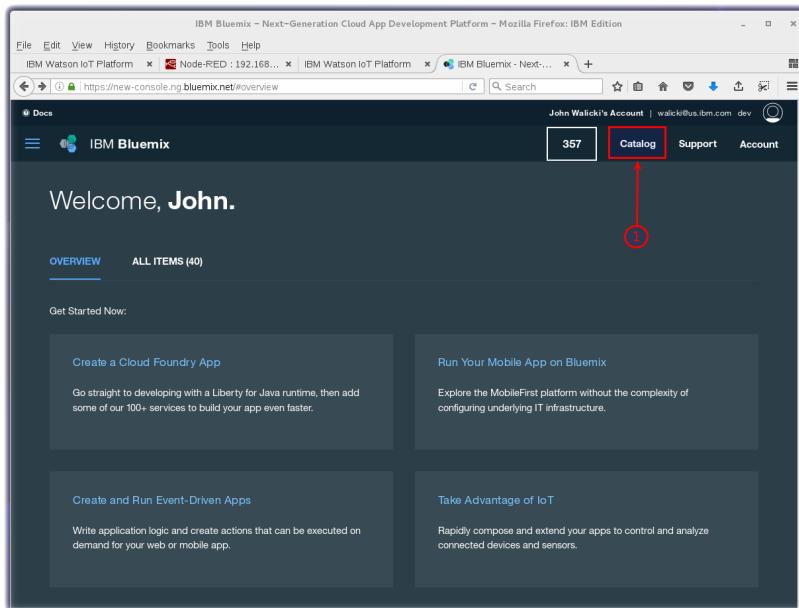
To create the above program, follow the URL below and download the Lesson3.flow. The text is a Node-RED representation of the above graphical flow. You can copy it from the browser to the clipboard and paste it in your Node-RED browser tab. From the **Node-RED menu, select Import → Clipboard, paste the flow and press the red Import button**. Finally press the Deploy button.

This flow is available at : <https://github.com/johnwalicki/WatsonDevConf-IoT-Lab>

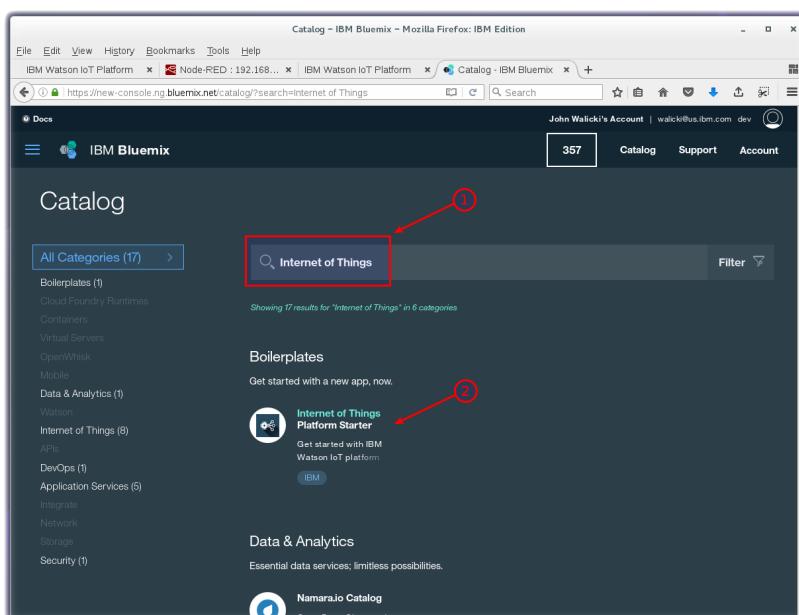
Lesson 4 - Send sensor data to a Bluemix Application

Quickstart is a fast way to send and see sensor data on the Watson IoT Platform. The next step is to create a Bluemix Watson IoT application that receives the Raspberry Pi sensor data via the Quickstart service and takes actions on it.

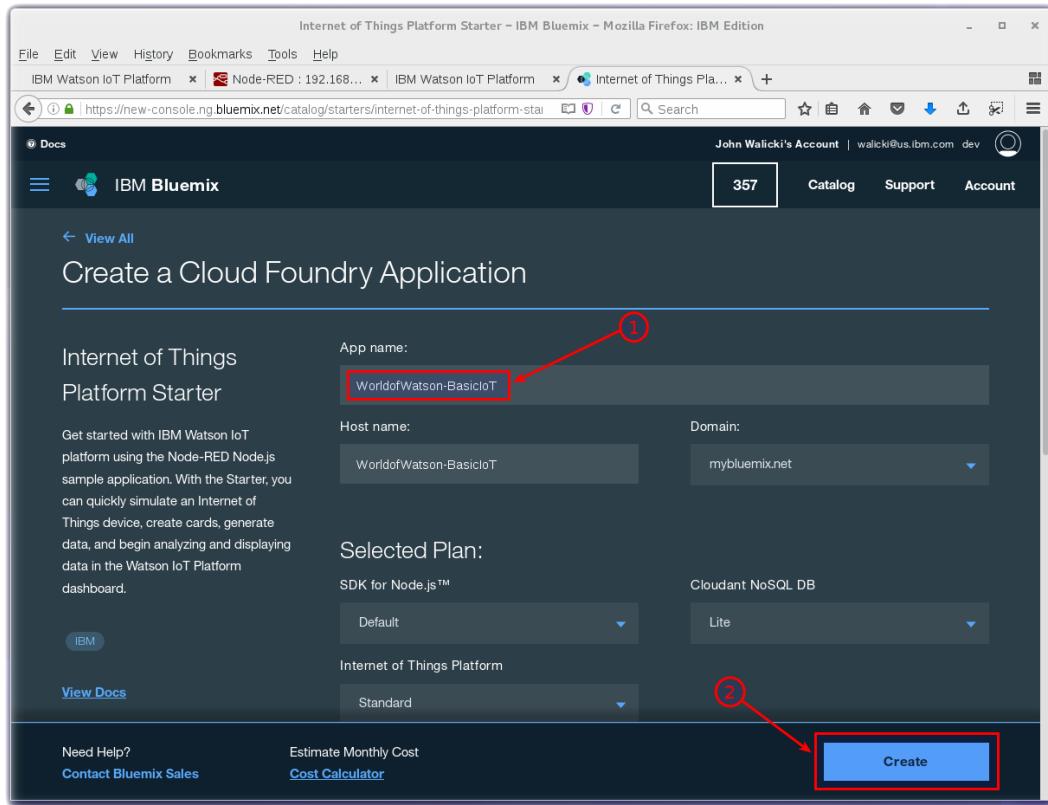
- If you do not yet have a Bluemix account, visit <http://bluemix.net/registration> and enter the requested information. Check your email to confirm the Bluemix account creation.
- See the lab instructor for a promocode to extend your 30 day free trial.
- Login into <http://bluemix.net> with your new account userid and password.
- Click on the word **Catalog** on the upper right side.



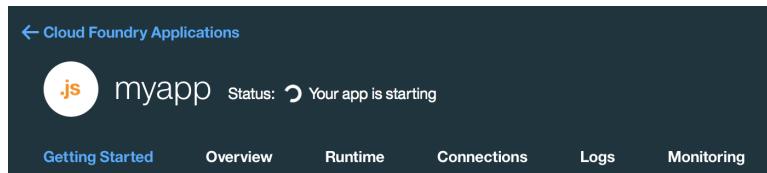
- Search for **Internet of Things Platform Starter** – The boilerplates are designed with pre-assembled services that work together. The Internet of Things Platform Starter includes a Node-RED Node.js web server, Cloudant database to store sensor data, and the IoT platform service so you can connect devices



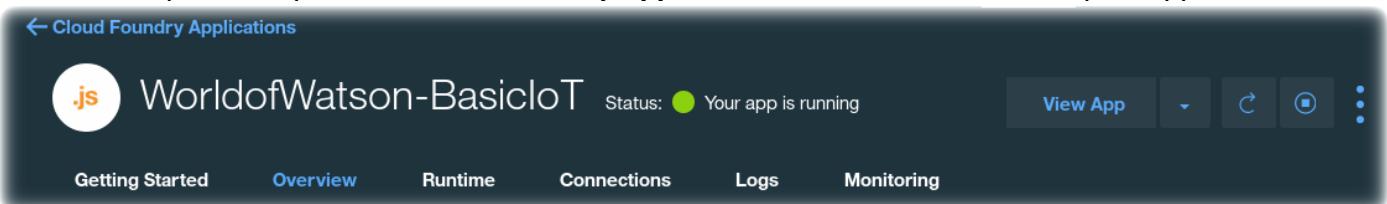
- Name your application something unique (1). If you choose **myapp**, your application will be located at <http://myapp.mybluemix.net>. There can only be one “**myapp**” application registered in IBM Bluemix. You can try adding your initials in front of the host if the host you choose is already taken by someone else. Click on **Create** (2) to create the application instance.



- IBM Bluemix will create an application in your account based on the services in the boilerplate. This is called staging an application. It can take a few minutes for this process to complete. While you wait, you can click on the **Logs** tab and see activity logs from the platform and Node.js runtime.

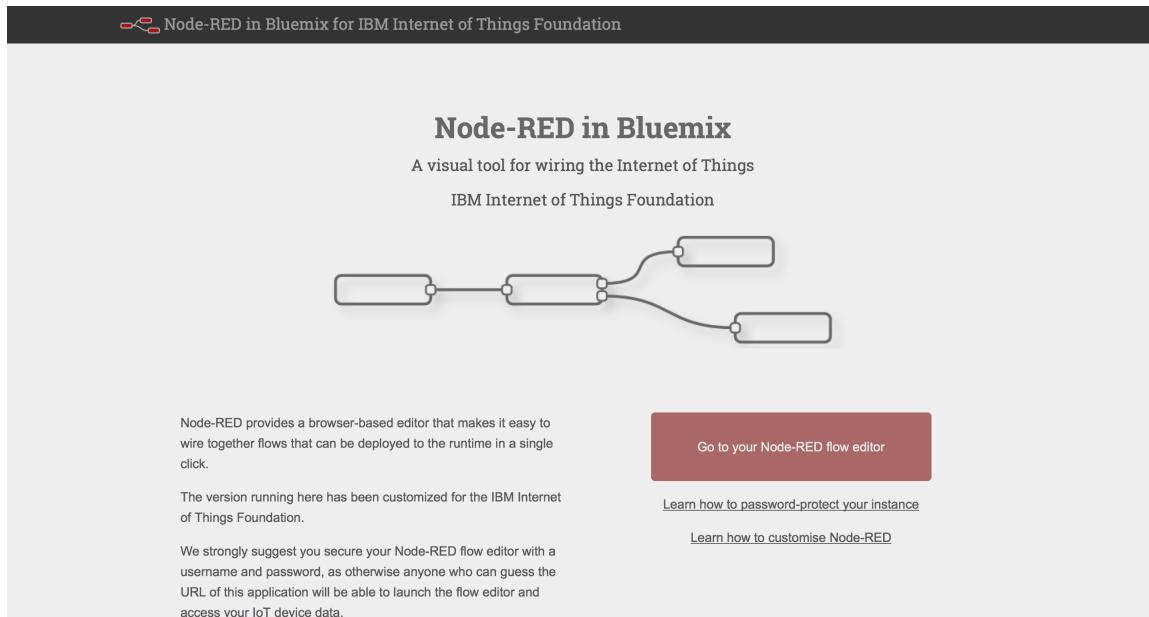


- You may need to press the **Cloud Foundry Applications** link and then re-enter your application.

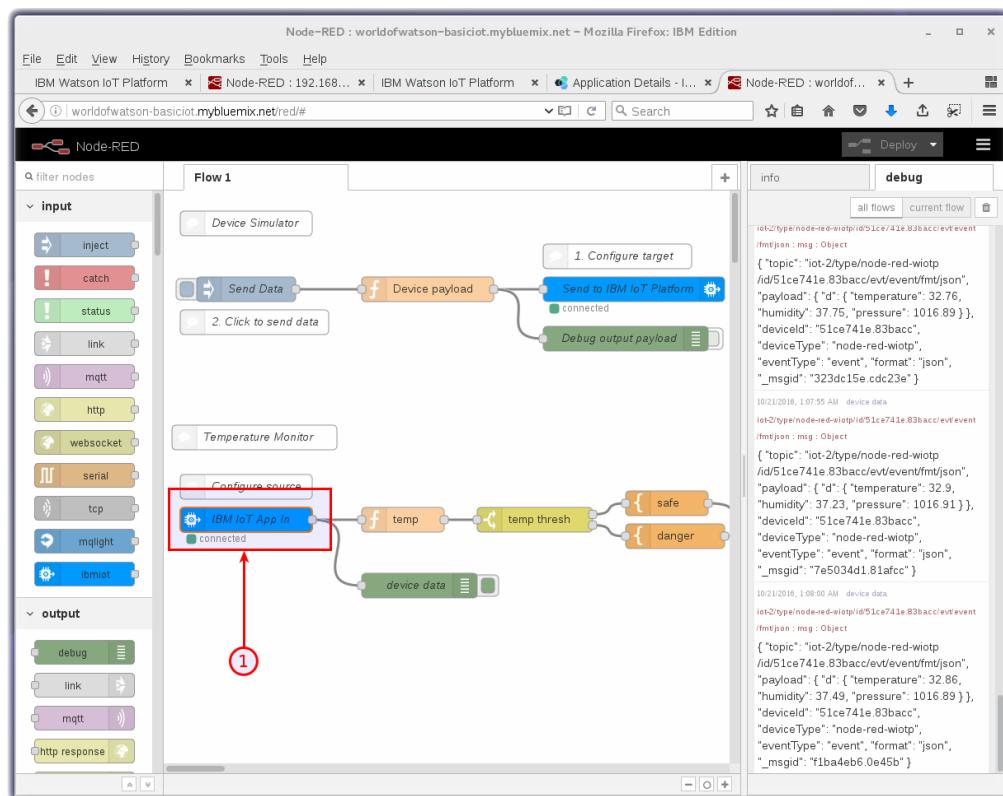


- Open your application by pressing the **View App** button.

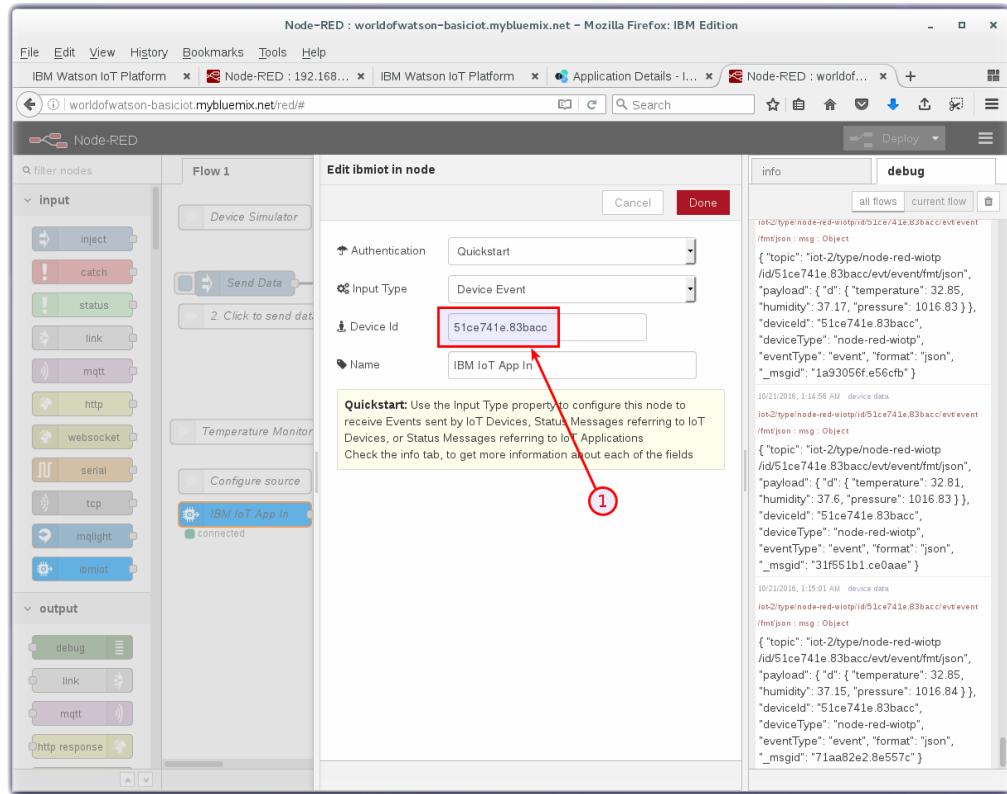
- A new browser tab will open to the Node-RED start page. Node-RED is an open-sourced Node.js application that provides a visual editor that makes it easy to wire together flows. Click on the red button **Go to your Node-RED flow editor** to launch the editor.



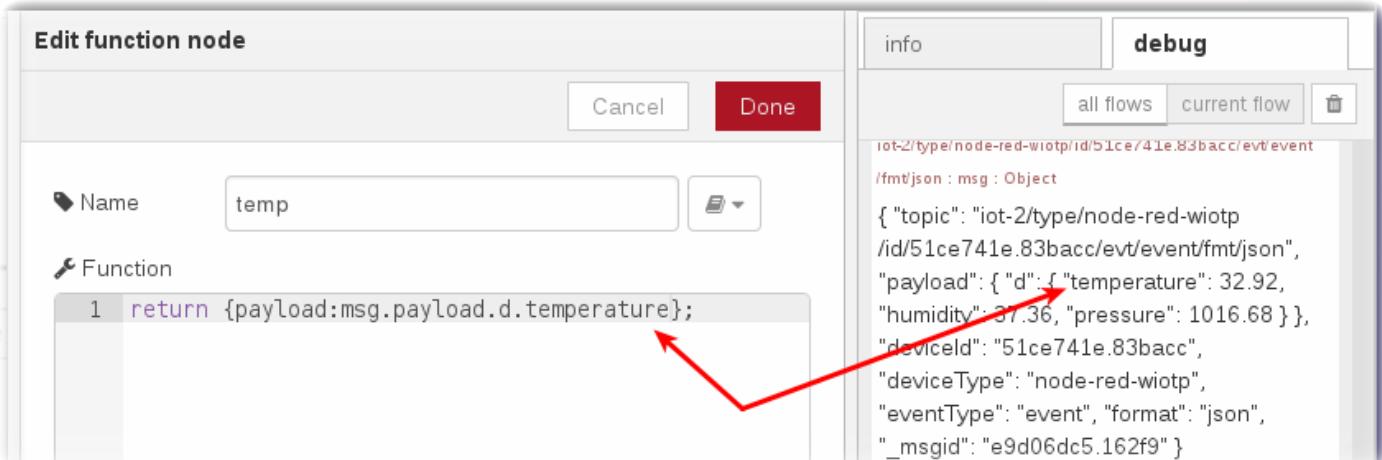
- The IoT Starter app has a default flow that can be modified. To receive the Raspberry Pi SenseHAT data from your **Quickstart ID** within our new Bluemix application, click on the **IBM IoT App In** node (1).



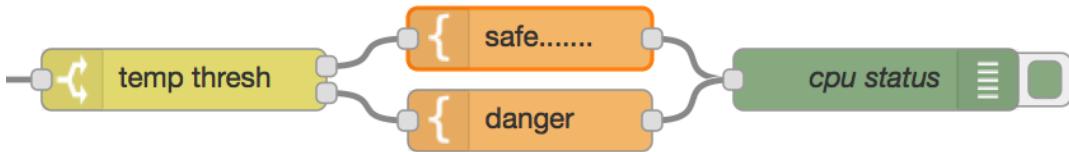
- Authentication by default is set to Quickstart. Provide the Device Id (1) from the local Node-RED Watson IoT node in Lesson 2. You can Copy and Paste the Device Id between browser tabs.



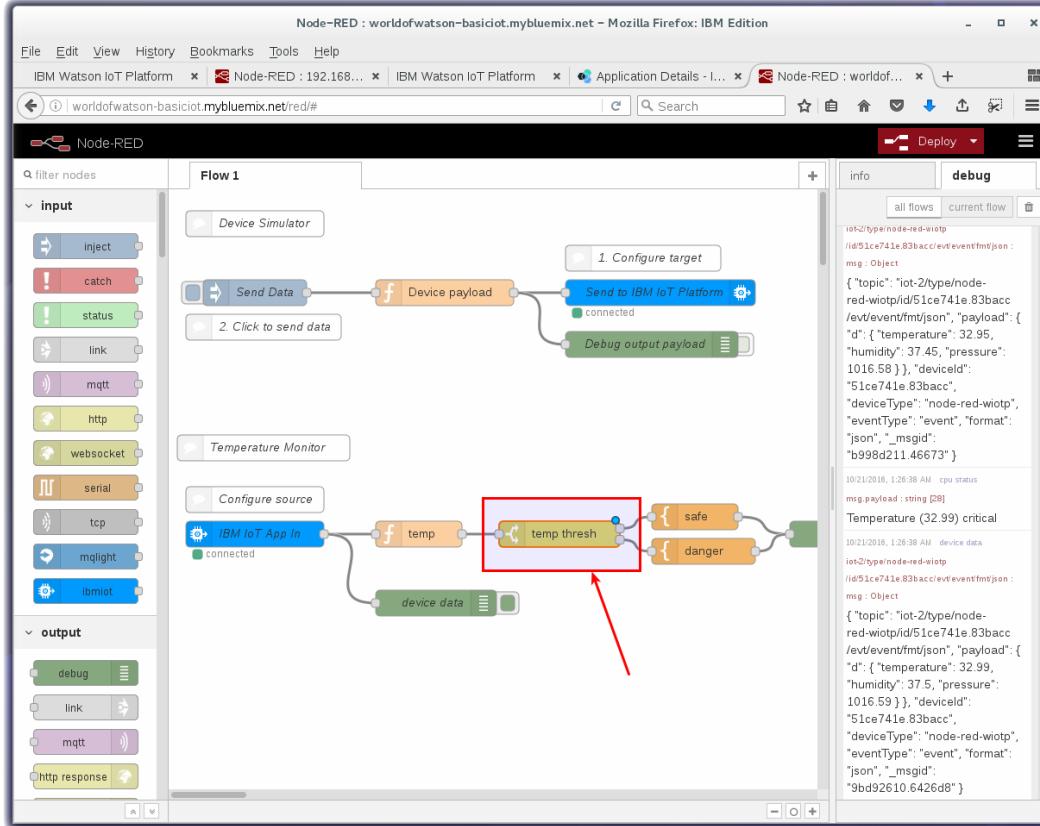
- Click on the “temp” Function node. Fix up the temp function so that it extracts the **msg.payload.d.temperature** value that is being sent from the Raspberry Pi SenseHAT.



- The yellow node is called a node. You can program logic using a switch node and split a flow into two or more flows based on a property's value. In this example, if the temperature is less than or equal to 30°C, it is considered "safe" and continues with the flow to the template labeled safe. If the temperature is greater than 30°C, it is considered "danger[ous]" and continues with the flow to the template labeled danger.



- Set the temp threshold calibrated to your SenseHAT by clicking on the temp thresh node.



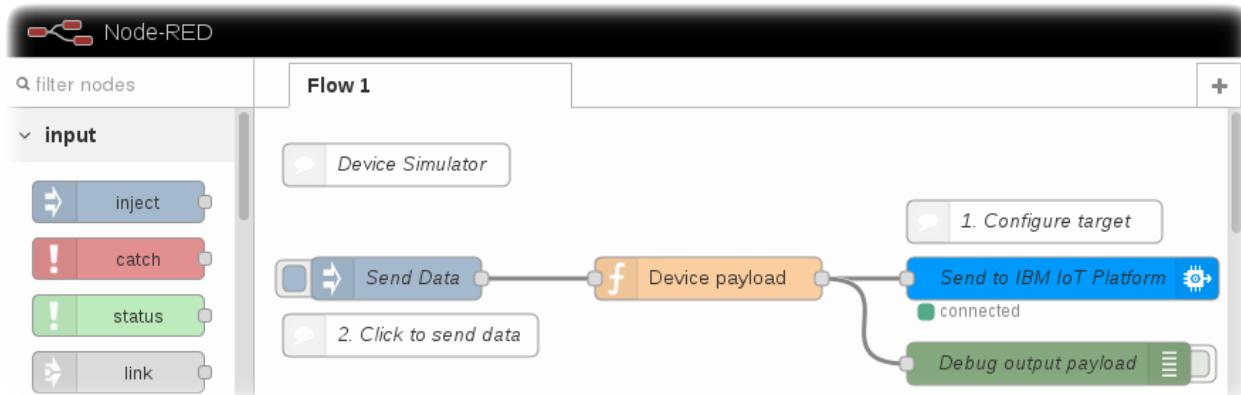
- Click on the button in the top right of the screen to save and deploy your changes.
- Click on the debug tab in the right-hand pane. Every 5 second, the SenseHAT emits a device event to the Watson IoT platform with temperature and other data. The Node-RED application subscribes to these events and the IBM IoT In node triggers the flow with the temperature data in the message. When the debug nodes are processed, contents of the message object are in the debug tab. Adding debug nodes can be helpful when something doesn't work right and you want to see the values being passed around. The Quickstart sensor data should appear in the debug tab and the Temperature is reported as Critical or Safe within our Bluemix Watson IoT application.

Lesson 5 – Send Twitter Feed Sentiment Alerts to a Raspberry Pi

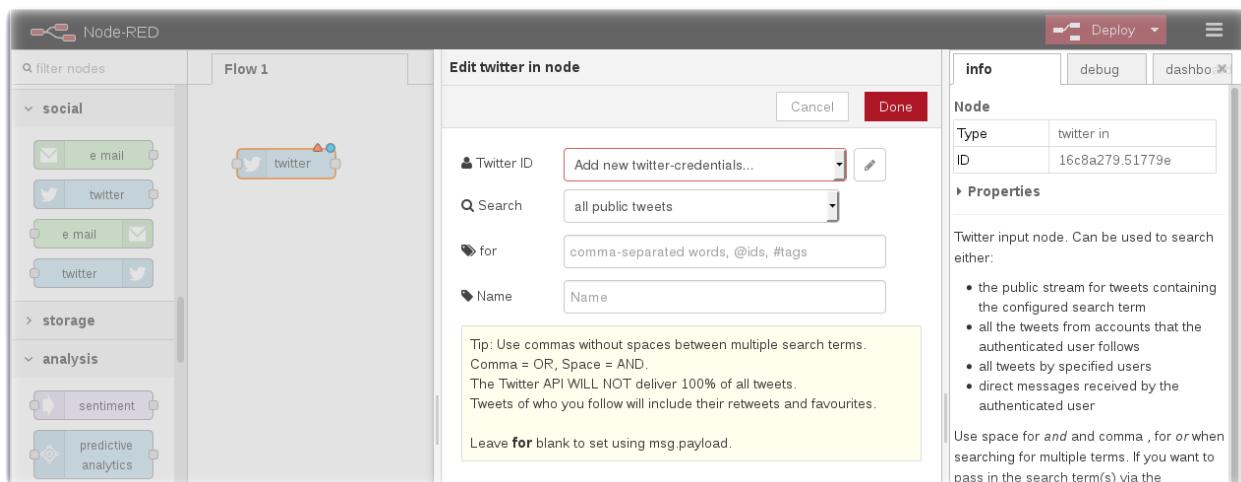
Until this point, the lessons have sent data **from** the Raspberry Pi **to** Watson IoT. Lesson 5 will send command data to the Raspberry Pi. Recall that in Lesson 3, the SenseHat LED red/green alert colors were set if the temperature exceeded a threshold value inside the switch statement. In this section, we will connect a Twitter account and analyze the Twitter feed using the Watson Sentiment API. The Sentiment score of the tweets will be sent to the Raspberry Pi and the SenseHat LED will display a range of sentiment colors.

Warning: These modifications will require changes on both the Bluemix Node-RED and on the local Node-RED flows. Be careful as you make changes in the two browser tabs. It is easy to become disoriented by which flow you are editing.

1. Let's start the modifications on the Bluemix Watson IoT Application side. Recall that the default flow for a Watson IoT Starter Application looks like this.



2. The “Device Simulator” and “2. Click to send data” comment nodes and “Send Data” inject nodes can be deleted and will be replaced by a Twitter input node.
3. Sign up for a Twitter account at <http://twitter.com>. If you already have a Twitter account, proceed to step 4.
4. Add a **twitter** input node from the palette “social” section. Click on the twitter node and then click on the “Add new twitter-credentials...” pencil button. Authenticate with Twitter. The account you sign in with will be used to receive tweets.





Authorize Node RED to use your account?

[Authorize app](#)

[Cancel](#)



Node RED

nodered.org

Node-RED Twitter node

This application will be able to:

- Read Tweets from your timeline.
- See who you follow, and follow new people.
- Update your profile.
- Post Tweets for you.
- Access your direct messages.

Will not be able to:

- See your Twitter password.

5. Return back to the node configuration. The settings for the Twitter node should have your username set. Click **Add**.



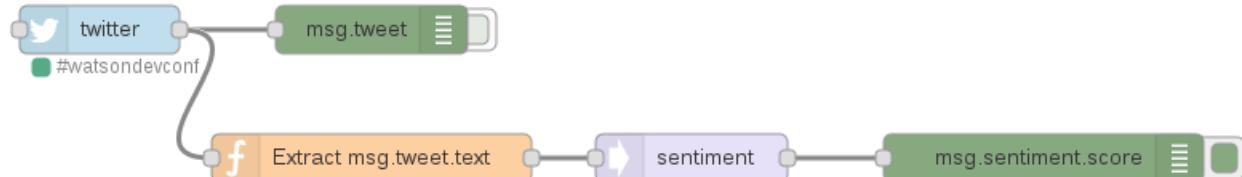
6. Configure the twitter feed to “Search all public tweets” for your favorite #hashtag. You might want to analyze the sentiment of your local sports team – NFL or College Football for instance – or the Watson Developer Conference hashtag - #watsondevconf

Be aware that there is a Twitter API rate limit to the number of tweets your application can receive. Following politics or trending topics might exceed your Twitter API limit and your application will pause until tomorrow. It's quite fun to watch however after Election day.

7. Confirm that the Bluemix application is receiving tweets by wiring the Twitter node to a Debug node that prints out msg(tweet). Press the Deploy button.



8. Pass the Twitter messages to a function that extracts the msg(tweet.text) Wire the function node to the Watson Sentiment node for analysis. Wire the Sentiment node to a Debug node which reports the sentiment. Press the Deploy button. A score greater than zero is positive and less than zero is negative.



9. Recall that the default IoT Starter application sends simulated data that the default “Device payload” function generates. This function must be modified to send the sentiment score to the Raspberry Pi. The Raspberry Pi local Node-RED flow will set the color of the SenseHat LED display based on the sentiment of the tweets. If you double click on the “Device payload” function, select all of the javascript code and delete it. Replace it with this function.

Quickstart can only accept JSON events.

Example JSON device event: {"d": {"myName": "RaspberryPi", "temperature": 989}}

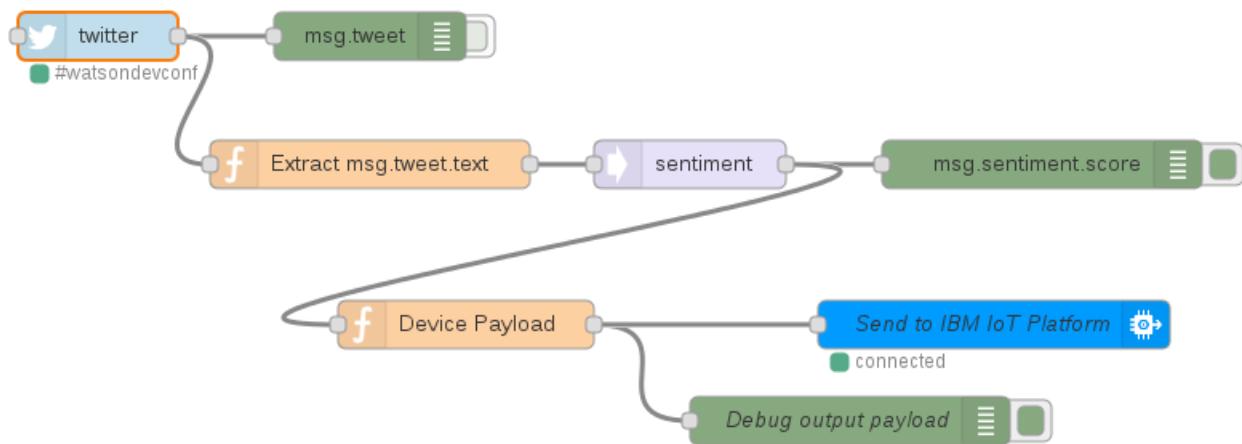
```
// Create MQTT message in JSON
msg = {
  payload: JSON.stringify(
    {
      d:{
        "sentiment" : msg.sentiment.score
      }
    }
  );
}
return msg;
```

10. This flow is available at <https://github.com/johnwalicki/WatsonDevConf-IoT-Lab>

11. The **Send to IBM IoT Platform** node needs to be configured to send the data to your Quickstart Id. Click on the Send to IBM IoT Platform node and paste in the **Quickstart Device Id** from Lesson 2 and Lesson 4.

12. Remember to  Deploy this flow.

13. The new flow might look like this:



Receiving Bluemix Sentiment Score Alerts on a Raspberry Pi

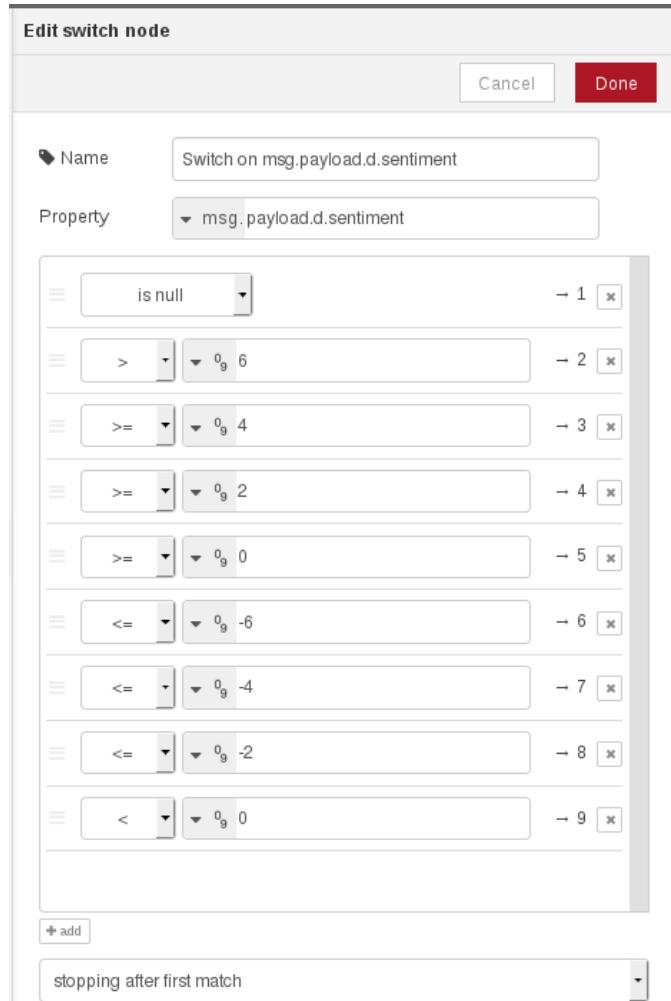
The local Raspberry Pi half of Lesson 5 will receive MQTT messages from your Bluemix application and act on the sentiment scores from a twitter feed. The Raspberry Pi SenseHat LED will display different colors depending on the sentiment.

A switch node and a series of function nodes will set the color to be sent to the Raspberry Pi SenseHat LED display. An alternative technique could implement the switch statement in a single function node and set the color, but the resulting picture (below) would be boring and not as snazzy in a workshop lab.

1. On the local Node-RED browser tab, a new Lesson 5 flow will be added. Create a new flow (press the + sign)
2. Drag a **IBM IoT** input node from the palette onto your flow.
3. Double Click on the IBM IoT Node to configure it with your Quickstart device ID from Lesson 2 and 5
4. Drag a Debug node and wire it to the IBM IoT Node. Click on Deploy to verify the flow is receiving sentiment scores.



5. Drag a **switch** node to your flow. Double click on the switch node and press the “+ add” button nine (9) times. Click on “stopping after the first match” and set the ranges as displayed in the figure below.

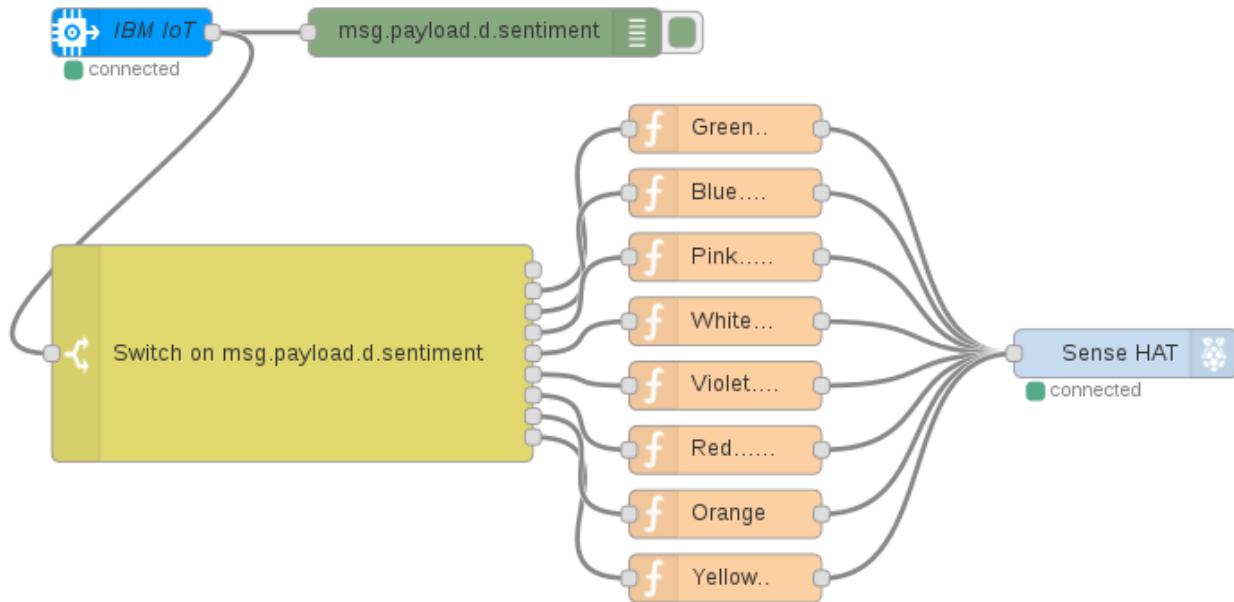


6. Drag eight (8) Function nodes to your flow and configure them to send, respectively; green, blue, pink, white, violet, red, orange, yellow color patterns to the SenseHat LED.

```
msg.payload = "*,* ,green";  
return msg;
```

7. Drag a Sense HAT Sense HAT output node to your flow.

8. Wire the nodes like this illustration.



9. Click on Deploy to save and deploy your changes

10. This flow is available at <https://github.com/johnwalicki/WatsonDevConf-IoT-Lab>

11. Menu → Import → Clipboard paste into Node-RED

Note: If your LED display starts blinking RED / GREEN, it is possible you didn't disable part of the Lesson 3 flow. Turn to that flow and delete the output wire to the SenseHat node. Don't delete the entire flow however. You might still want to send environmental sensor data to Watson IoT Platform.

Lesson 6 - Send Weather Alerts to a Raspberry Pi

In this section, we will display the temperature of your favorite home town. The outside temperature will be retrieved via the **Weather Company Data for IBM Bluemix** service.

1. Return to the **Connections** tab of your Internet of Things Starter Application
2. Click on the **Connect New** button (1) of your Node-RED application in IBM Bluemix.

The screenshot shows the IBM Bluemix Cloud Foundry Apps dashboard. On the left, there's a sidebar with links: Getting Started, Overview, Runtime, and Connections (which is highlighted in blue). The main area displays the app 'WatsonDevConf-BasicIoT'. It includes a 'View App' button, a refresh icon, a stop icon, and a more options icon. Below these are two buttons: 'Connect Existing' and 'Connect New'. A red circle with the number '1' is drawn around the 'Connect New' button. To the right of the app name, it says 'Status: Your app is running' with a green dot.

3. Select the **Weather Company Data for IBM Bluemix** under the Data & Analytics category.

The screenshot shows the IBM Bluemix Catalog. In the search bar, 'weather' is typed. Below the search bar, there are categories: All Categories (2), Apps, Mobile, Services (2), Data & Analytics (1), Watson, Internet of Things (1), APIs, Storage, Security, and DevOps. On the right, the search results for 'weather' are shown, with a count of 2 results in 2 categories. Under the 'Data & Analytics' heading, the 'Weather Company Data' service is listed. It has a brief description: 'Essential data services; limitless possibilities.' and a link: 'Use the Weather Company Data for IBM Bluemix service to'. There's also an 'IBM' logo.

4. Click **Create** to add the service to your application.

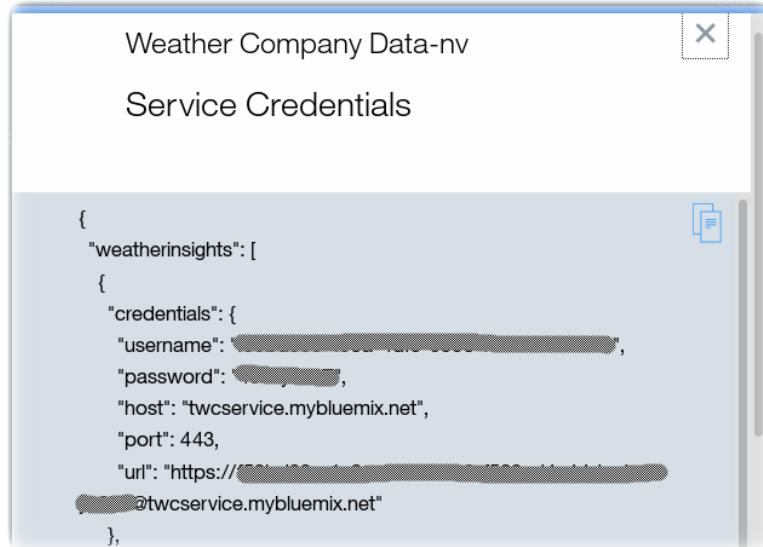
- When prompted to restage the application, click **Restage** to restart the application and update the environment with the credentials to the Weather service.
- When the application has restarted, a third service tile will appear in the **Connections** tab. Click on **View Credentials** in the Weather Company Data service tile to show the service credentials.

The screenshot shows the IBM Bluemix Cloud Foundry Apps dashboard for the application 'WatsonDevConf-BasicIoT'. The 'Connections' tab is active. Three service tiles are listed:

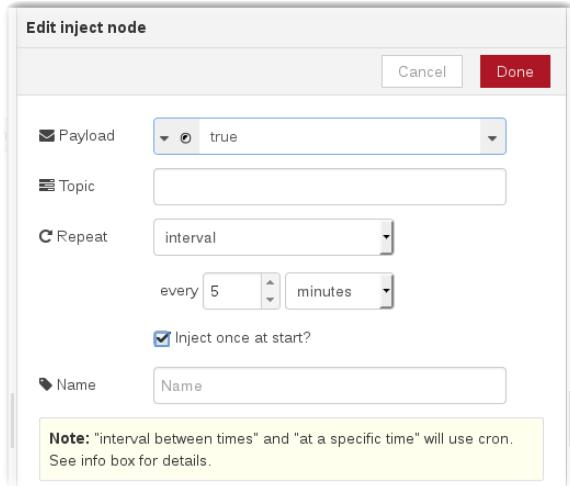
- WatsonDevConf-BasicIoT (Cloudant NoSQL DB Lite)
- WatsonDevConf-BasicIoT (Internet of Things Platform iotf-service-standard)
- Weather Company Data (Weather Company Data Free-v2)

The 'Weather Company Data' tile has its 'View Credentials' button highlighted.

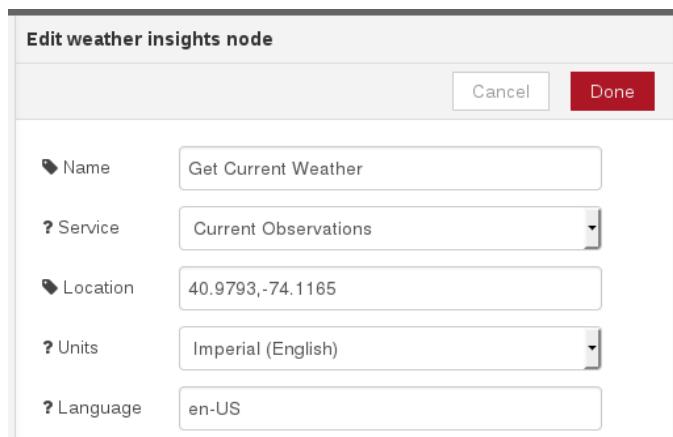
- Copy the username and password values and save them for use in step #10.



8. Return to your Node-RED application flows on Bluemix and add a new flow tab by clicking the +. Name the flow **Weather**
9. Add a  node as shown below. This will trigger the flow once at startup, and then every 5 minutes.

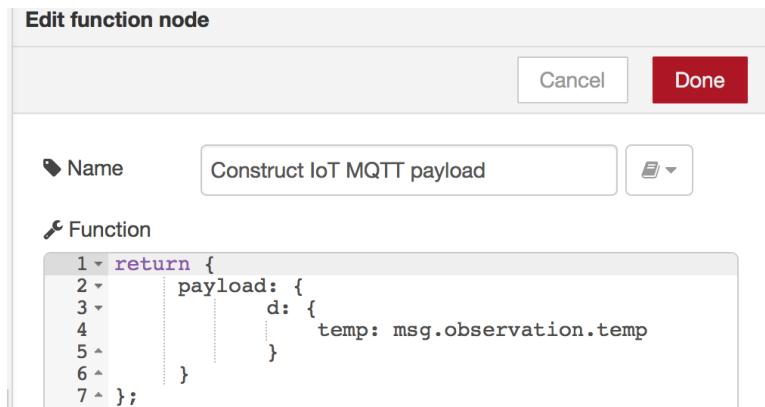


10. Add a  node as shown below.



Look up the Geocode of your home town and enter the Longitude,Latitude in the Location field. Do not enter any spaces. Note the – minus symbol / negative sign if you live in the western hemisphere.

11. Add a  node as shown below.



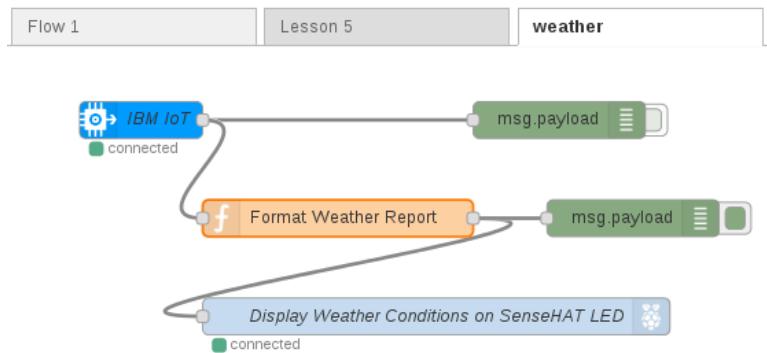
This will format the temperature in a MQTT message payload, for use in step #11. If the weather condition changes every 5 minutes, the value will be sent to the Raspberry Pi for display on the SenseHAT LED

12. On IBM Bluemix, connect the nodes together as shown below.



Get the Code:
<https://github.com/johnwalicki/WatsonDevConf-IoT-Lab>

13. On the Raspberry Pi, connect the nodes together as shown.



Get the Code:
<https://github.com/johnwalicki/WatsonDevConf-IoT-Lab>

14. Click on  to save and deploy your changes.

15. **Note:** You will need to **disable** the SenseHAT output nodes created in Lesson 3 and Lesson 5 so that those flows do not overwrite the LED. The LED should display the outside temperature and will update the temperature from the Weather service every 5 minutes.

Additional Raspberry Pi / SenseHAT Resources

In this final section, we will challenge the workshop attendees to implement additional SenseHAT projects.

There are many Raspberry Pi Recipes available at:

<https://developer.ibm.com/recipes>

An excellent Marble Maze project is great fun:

Connecting a Sense HAT to Watson IoT using Node-RED

<https://developer.ibm.com/recipes/tutorials/connecting-a-sense-hat-to-watson-iot-using-node-red/>