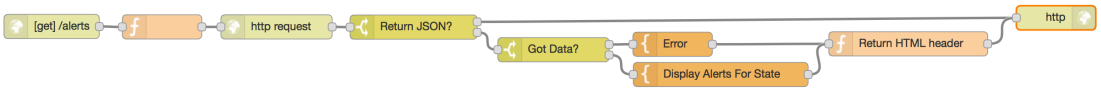
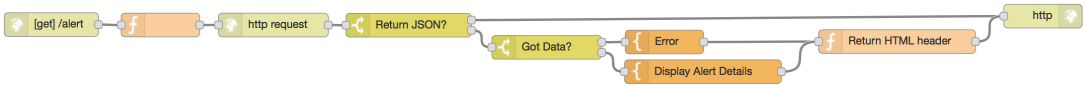
Weather Company Data for IBM Bluemix in Node-RED

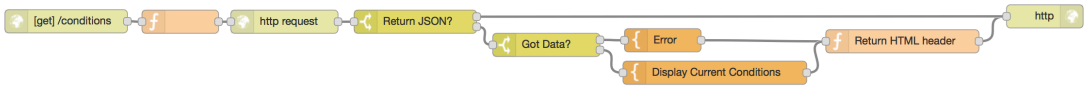
Hands-On Lab

JeanCarl Bisson | jbisson@us.ibm.com | @dothewww

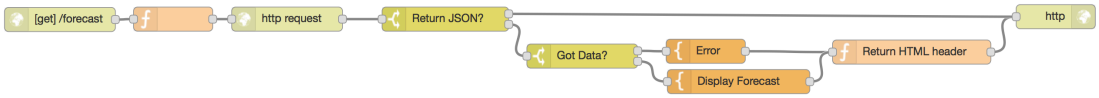


Get weather alerts by state and display weather alerts and advisories.
(see *Weather Alerts*)

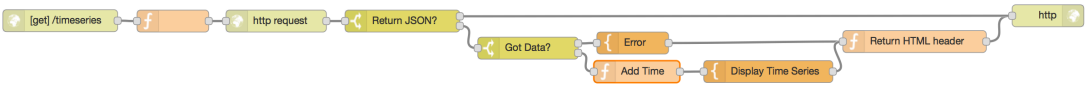




Get current weather conditions for a given location.
(see *Current Weather Conditions*)



Get daily forecasts for the next 3, 5, 7, or 10 days.
(see *Daily Forecast*)



Display past weather conditions over the last 24 hours.
(see *Time Series*)

Display weather trends over time for a given date range.
(see *Almanac*)

A digital copy of this lab and code snippets can be found at:
`http://ibm.biz/node-red-weather-company-data`

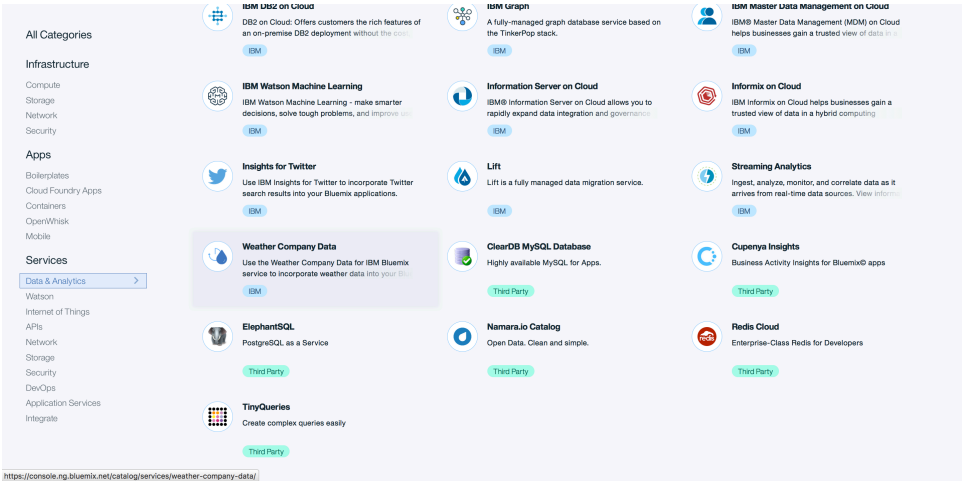
Add Weather Company Data API in IBM Bluemix

The Weather Company Data for IBM Bluemix service offers a variety of weather data via REST based API endpoints. In these tutorials, we'll get the current weather conditions, forecasted weather for the next ten days, retrieve weather alerts, use the weather almanac to find conditions over time, and use time series weather conditions for the last 24 hours to see how the weather has changed. Although the postal code is primarily used to look up weather in this tutorial, the API also accepts latitude/longitude geocode coordinates.

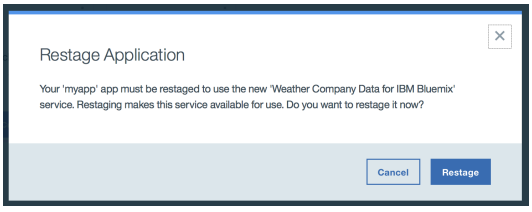
To get started using the API, create service credentials for the Weather Company Data for IBM Bluemix service.

1. Go to the Connections tab under the application overview for the Node-RED application in the IBM Bluemix dashboard and click on **Connect New**.

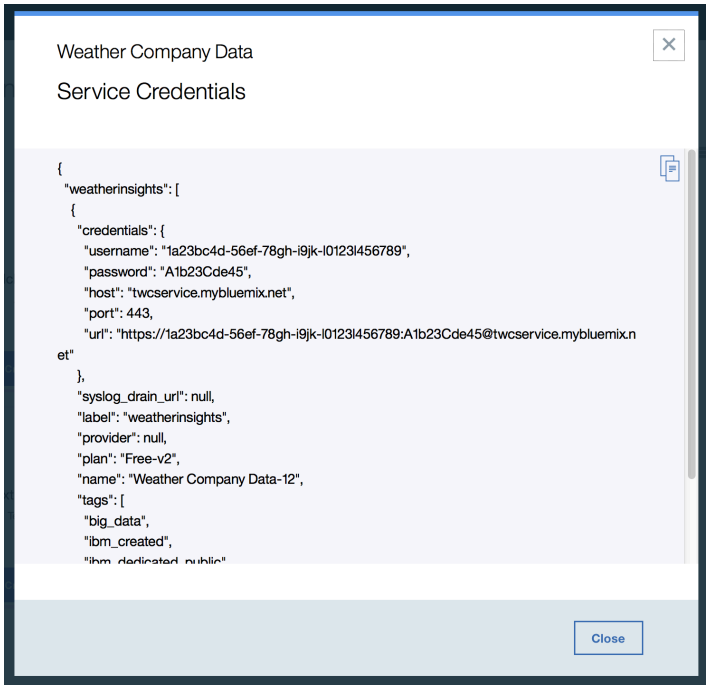
2. Click the **Weather Company Data for IBM Bluemix** tile under the Data & Analytics section. Click on **Create**.



3. IBM Bluemix will prompt to restage the application. Click on **Restage**. The application will restart and include the new service credentials in the environment.



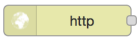
4. In the application overview in your Bluemix dashboard, click on **Connections** in the sidebar. Click on **View Credentials** underneath Weather Company Data for IBM Bluemix service. Copy the username and password values. These credentials will be needed later when making HTTP requests in the remaining sections of this tutorial.



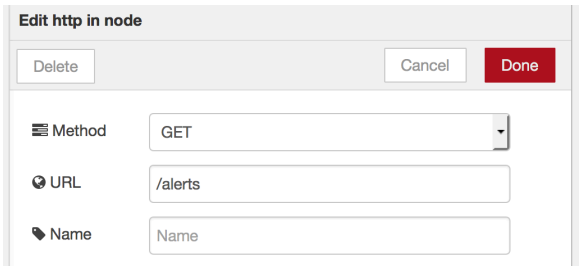
5. When the application has finished restaging, open the Node-RED Flow Editor. If you already have Node-RED open, refresh the page.

Weather Alerts

The Weather Company Data Alerts API returns active weather alert headlines that are related to severe thunderstorms, tornadoes, earthquakes, and floods. These APIs also return non-weather alerts such as child abduction alerts and law enforcement warnings. In this section, we'll create two flows. The first will list alert headlines for a specified state passed into the application via a URL query parameter. The second flow will show the alert details page, linked from the list generated in the first flow, and output a more detailed view of the alert.

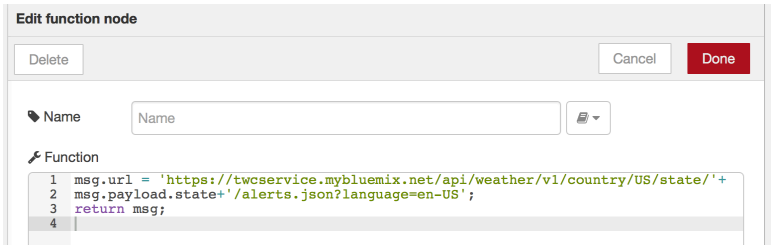
1. For the list webpage, add a 

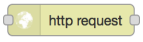
node as shown below.



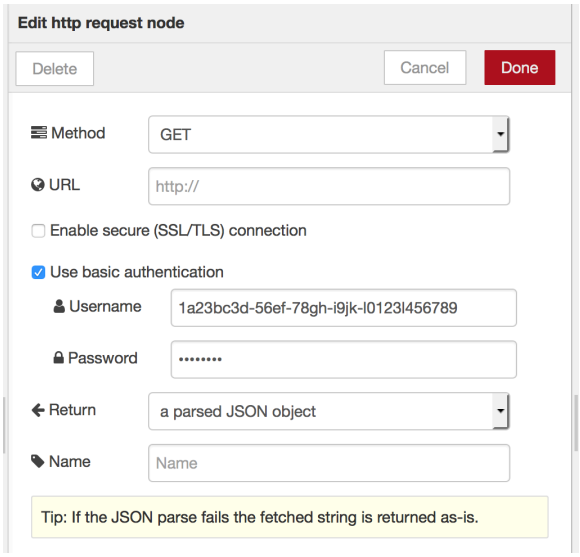
2. Add a 

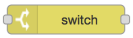
node as shown below. This will take a two-letter state abbreviation from URL query parameter, *state,* and place it in the URL used to call the Weather Company Data API.



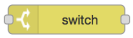
3. Add a 

node as shown below. After checking the **Use basic authentication** box, enter the username and password from the Weather Company Data service credentials in Step #4 of the **Add Weather Company Data API in IBM Bluemix** section.



4. To make this application versatile, we'll split the flow so the application will return the list of headlines in two formats. The first option will be where the results are returned in JSON format, great for use in applications that can call the web endpoint and consume the JSON. The second option will be a webpage showing the list of alert headlines in a human friendly format. Add a 

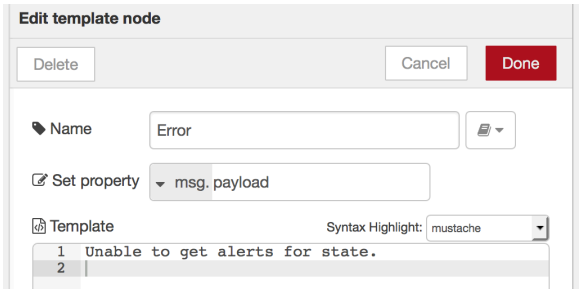
node as shown below.

5. For the flow where the JSON should be returned, the application can simply return the contents of *msg.payload*. We'll connect this to the HTTP response node in step #10.
6. For the flow where a webpage should be displayed, the application should first check if there was a successful response from the API. Add a 

node as shown below.

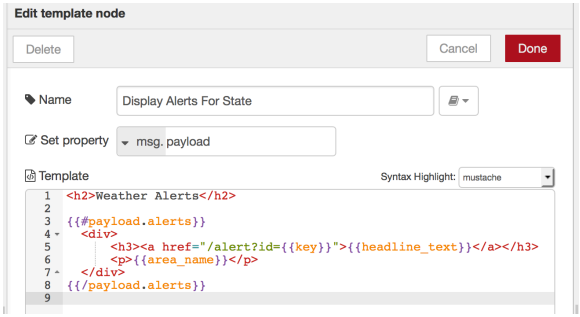
7. If a HTTP status code other than 200 is returned, the application should display an error message. Add a 

node as shown below.



8. For the flow where a webpage should be displayed, add a 

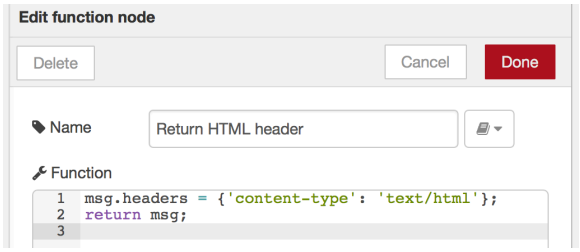
node with the HTML in the file named 1-display-alerts-for-state.html

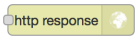


Get the code:
`ibm.biz/Bdr4TW`

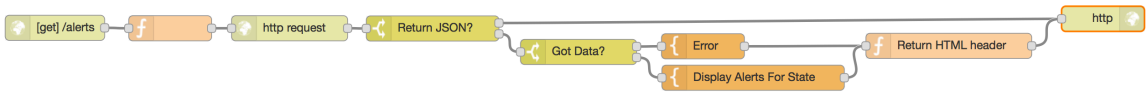
9. Add a 

node to return the correct content type HTTP headers to display a webpage.



10. Add a 

node. Connect the nodes together as shown below.



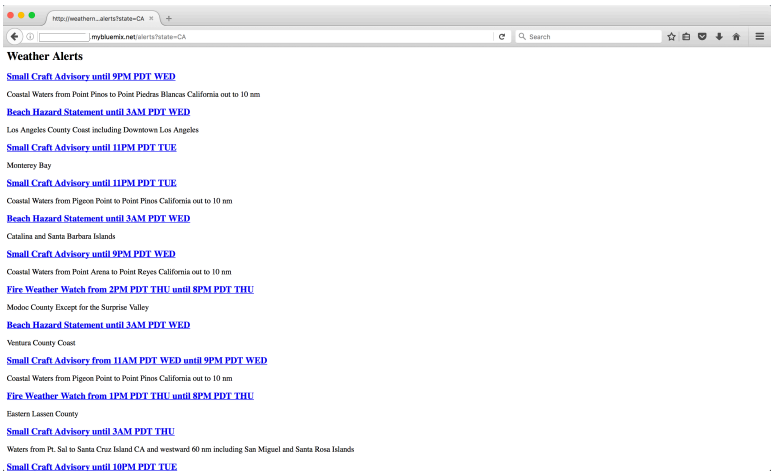
11. Click on the red 

button in the upper-right corner of the screen to save and deploy the changes.
12. Open a new browser tab and visit the application's endpoint, passing in the URL query parameter *state* to filter alerts for that state:

`http://<<MY-APP>>.mybluemix.net/alerts?state=<<STATE>>`

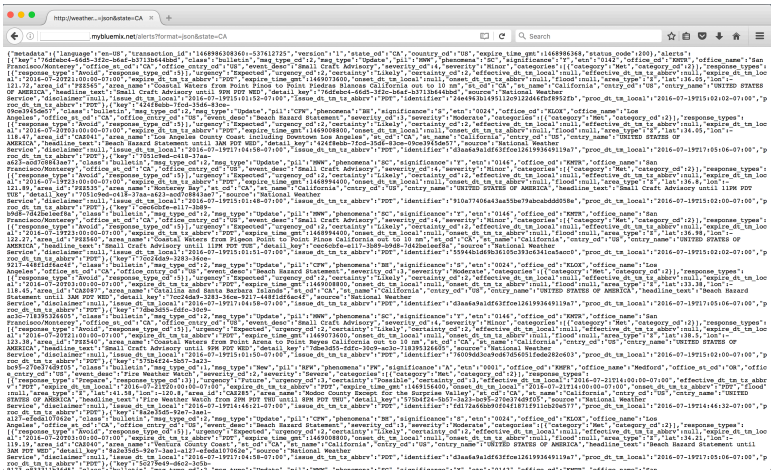
- Replace <<MY-APP>> with the host of the Node-RED application you chose.
- Replace <<STATE>> with the two-letter abbreviation for a state (such as CA for California, TX for Texas)

If there are active weather alerts for the state, a list of alerts similar to the one shown below will be displayed.

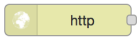


13. To see the JSON representation of the list, insert *format=json* in the URL query string:

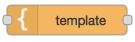
    `http://<<MY-APP>>.mybluemix.net/alerts?`**`format=json`**`&state=<<STATE>>`

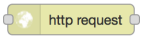


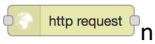
14. The headlines of the weather alerts are useful, but the real value is in the details of a specific alert. We'll add another flow that will take the id of a weather alert, make an API call to the Weather Company Data API, and display the alert details. Select the flow you've created by clicking and dragging a rectangle around the flow. Copy and paste it below the first flow.

    There are three nodes that need to be changed, the 

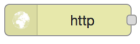
node, the 

, and the 

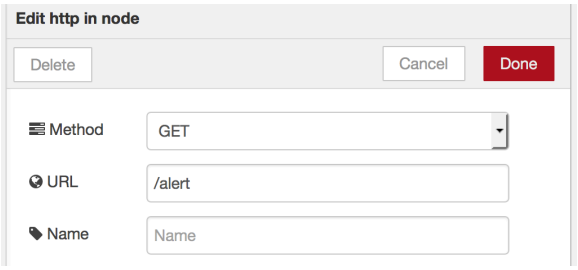
node. Some nodes, such as the 

node, specially handle credentials and do not transfer credentials when nodes are copied. The 

node needs the credentials added in the username and password fields.

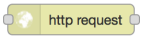
    Edit the 

node as shown below.

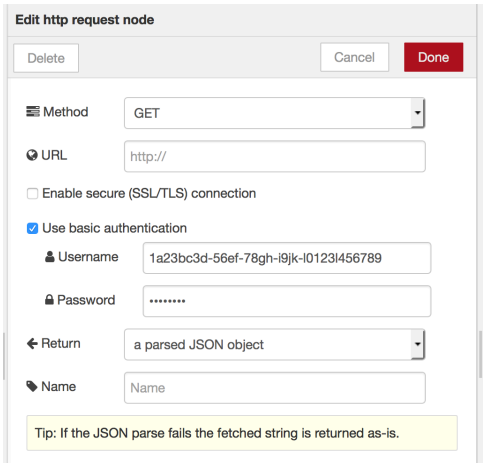


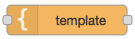
15. Change the 

node as shown below. This will take the alert ID from the URL query parameter, *id,* and place it in the URL used to call the Weather Company Data API.

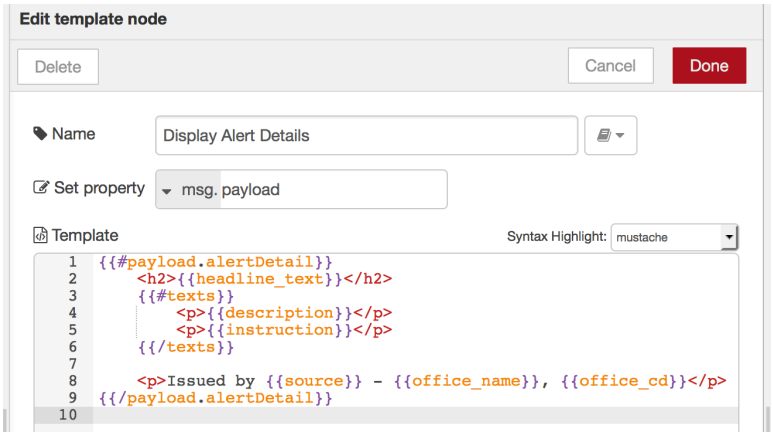
16. Change the 

node as shown below. After checking the **Use basic authentication** box, enter the username and password from the Weather Company Data service credentials in Step #4 of the **Add Weather Company Data API in IBM Bluemix** section.



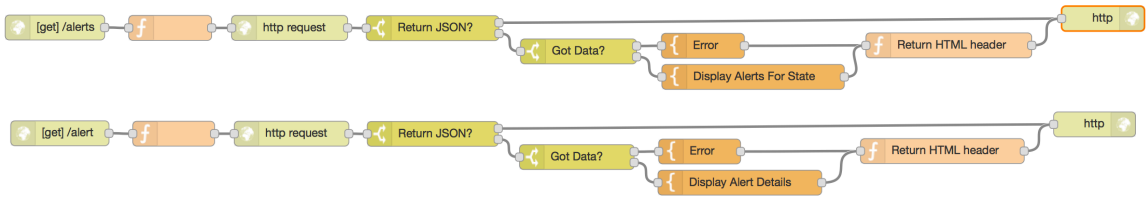
17. For the flow where a webpage should be displayed, replace the HTML in the 

node with the HTML in the file named 2-display-alert-details.html



Get the code:
ibm.biz/Bdr4wr

18. Connect the nodes together as shown below.

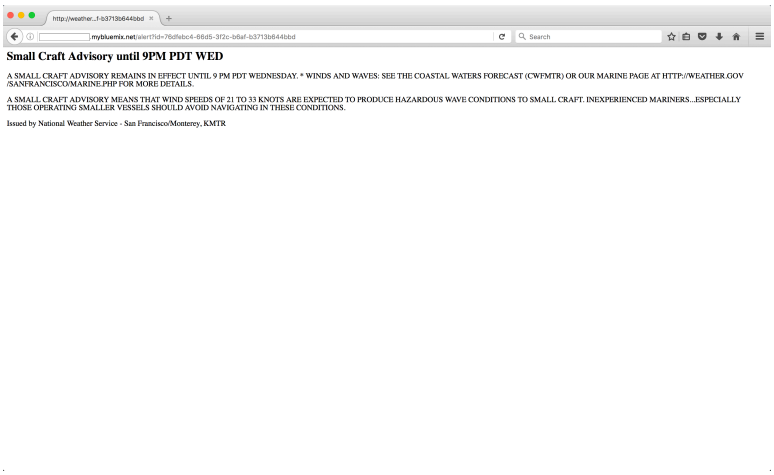


19. Click on the red  button in the upper-right corner of the screen to save and deploy the changes.
20. Open a new browser tab and visit the application's endpoint, passing in the alert ID from one of those from step #13:

    `http://<<MY-APP>>.mybluemix.net/alert?id=<<ID>>`

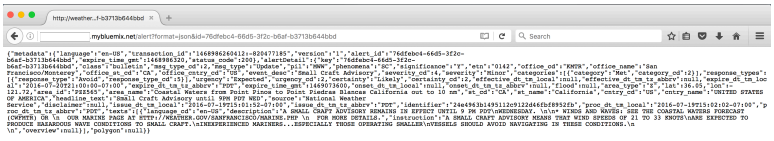
    - Replace <<MY-APP>> with the host of the Node-RED application you chose.
    - Replace <<ID>> with the alert ID from one of the weather alerts in the list from step #13.

    The details page displays the contents of the single matching alert.



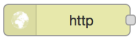
21. To see the JSON representation of the alert, insert *format=json* in the URL query string:

    `http://<<MY-APP>>.mybluemix.net/alert?format=json&id=<<ID>>`

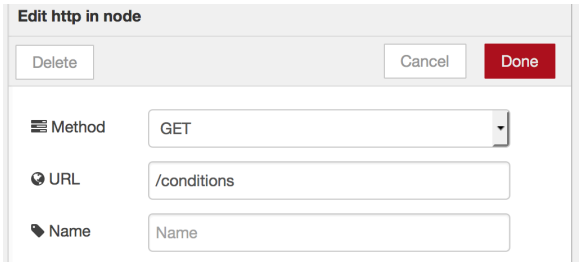


Current Weather Conditions

The Weather Company Data for IBM Bluemix API offers a variety of weather attributes including: temperature, humidity, barometric pressure, rain and snowfall, and cloud cover. In this section, we'll create a flow that takes a zip code input and display the current weather observation for that location. Please refer to the **Add Weather Company Data API in IBM Bluemix** section to create the Weather Company Data service credentials for the application.

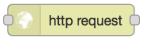
1. Add a 

node as shown below.

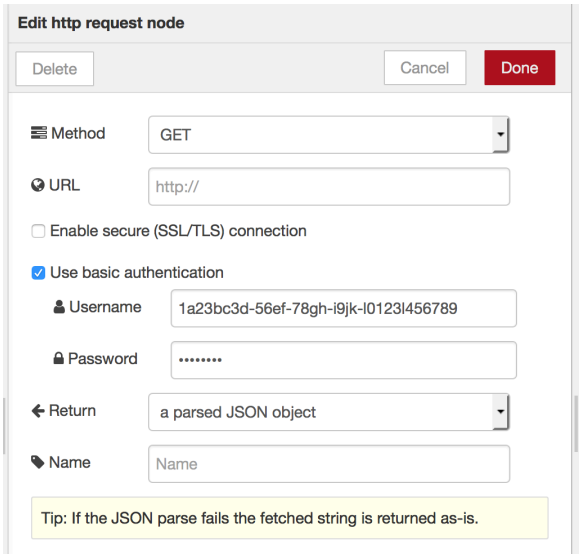


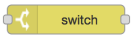
2. Add a 

node as shown below. This will take the URL query parameter, *zipcode,* and place it in the URL used to call the Weather Company Data API.

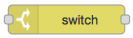
3. Add a 

node. After checking the **Use basic authentication** box, enter the username and password from the Weather Company Data service credentials in Step #4 of the **Add Weather Company Data API in IBM Bluemix** section.



4. To make this application versatile, we'll split the flow so the application will return the results in two formats. The first option will be where the result is returned in JSON format, great for use in applications that can call the web endpoint and consume the JSON. The second option will be a webpage showing the current conditions in a human readable format. Add a 

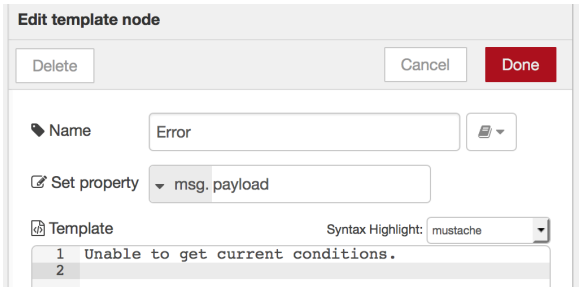
node as shown below.

5. For the flow where the JSON should be returned, the application can simply return the contents of *msg.payload*. We'll connect this to the HTTP response node in step #10.
6. For the flow where a webpage should be displayed, the application should first check if there was a successful response from the API. Add a 

node as shown below.

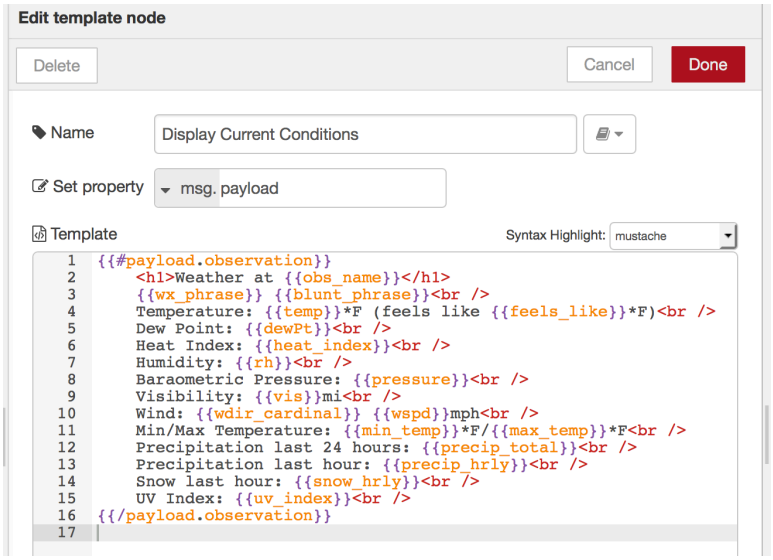
7. If a HTTP status code other than 200 is returned, the application should display an error message. Add a 

node as shown below.



8. For the flow where a webpage should be displayed, add a 

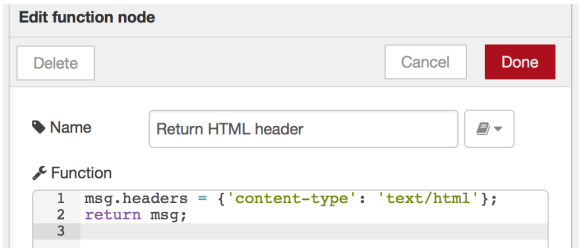
node with the HTML in the file named 3-display-current-conditions.html



Get the code:
ibm.biz/Bdr4w2

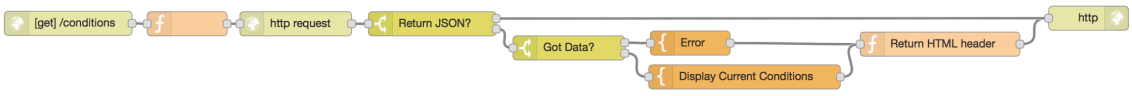
9. Add a 

node to return the correct content type HTTP headers to display a webpage.



10. Add a 

node. Connect the nodes together as shown below.



11. Click on the red 

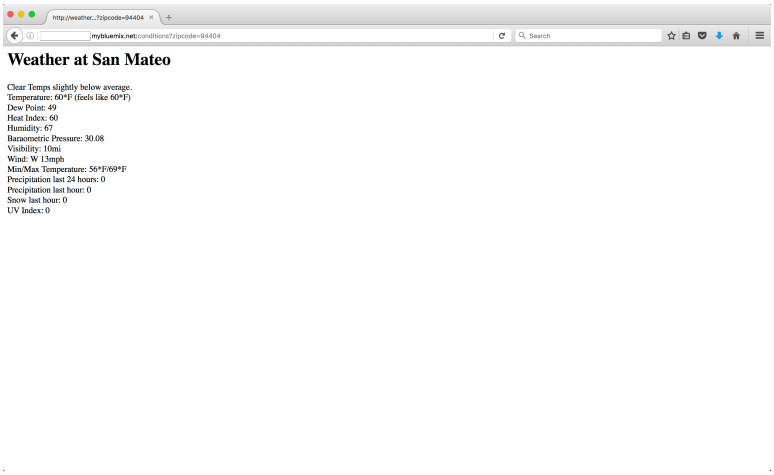
button in the upper-right corner of the screen to save and deploy the changes.

12. Open a new browser tab and visit the application's endpoint, passing in the zip code of a location for the current conditions:

    `http://<<MY-APP>>.mybluemix.net/conditions?zipcode=<<ZIP-CODE>>`

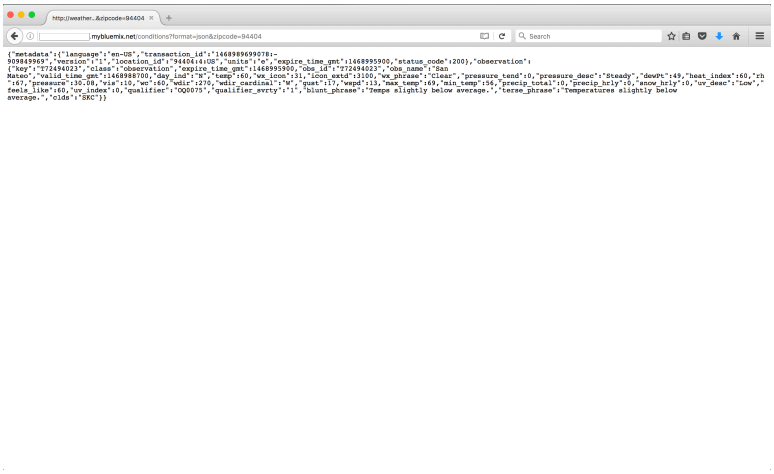
    - Replace <<MY-APP>> with the host of the Node-RED application you chose.
    - Replace <<ZIP-CODE>> with the five-digit zip code of a location.

    The current weather conditions are displayed on the webpage.



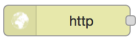
13. To see the JSON representation of the current weather conditions, insert *format=json* in the URL query string:

    `http://<<MY-APP>>.mybluemix.net/conditions?`**`format=json`**`&zipcode=<<ZIP-CODE>>`

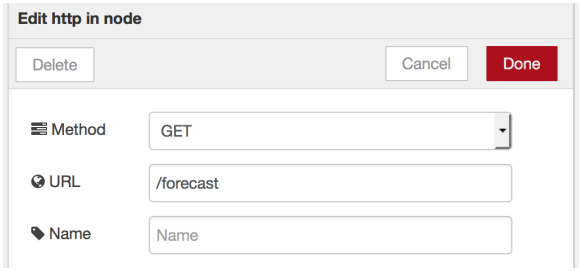


Daily Forecast

The Weather Company Data for IBM Bluemix API offers a daily forecast API that can contain up to ten days of daily forecasts for each location. Each day of a forecast can contain up to three separate forecasts. For any given forecast day the API can return day, night, and 24-hour forecasts. In this section, we will get the ten-day forecast. You can also choose to use the three-, five-, or seven-day API endpoint for a fewer number of days. Please refer to the **Add Weather Company Data API in IBM Bluemix** section to create the Weather Company Data service credentials for the application.

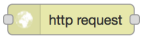
1. Add a 

node as shown below.

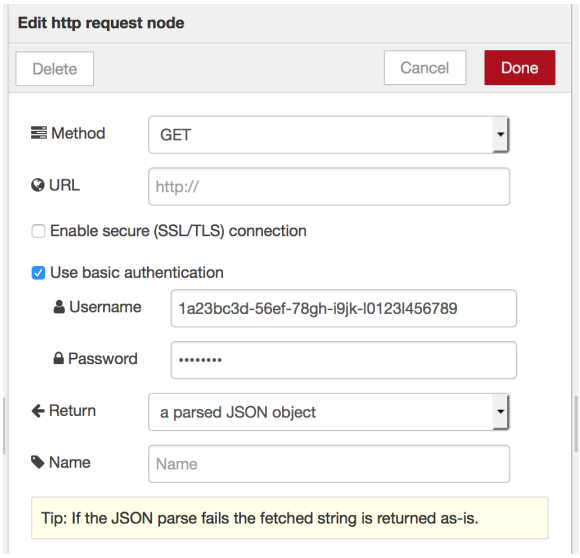


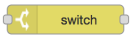
2. Add a 

node as shown below. This will take the URL query parameter, *zipcode,* and place it in the URL used to call the Weather Company Data API.

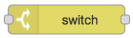
3. Add a 

node. After checking the **Use basic authentication** box, enter the username and password from the Weather Company Data service credentials in Step #4 of the **Add Weather Company Data API in IBM Bluemix** section.



4. To make this application versatile, we'll split the flow so the application will return the results in two formats. The first option will be where the forecasts are returned in JSON format, great for use in applications that can call the web endpoint and consume the JSON. The second option will be a webpage showing the forecasts in a human friendly list. Add a 

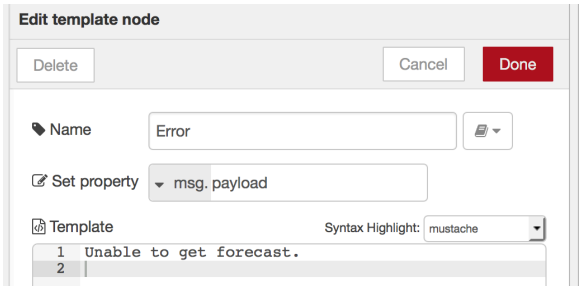
node as shown below.

5. For the flow where the JSON should be returned, the application can simply return the contents of *msg.payload*. We'll connect this to the HTTP response node in step #10.
6. For the flow where a webpage should be displayed, the application should first check if there was a successful response from the API. Add a 

node as shown below.

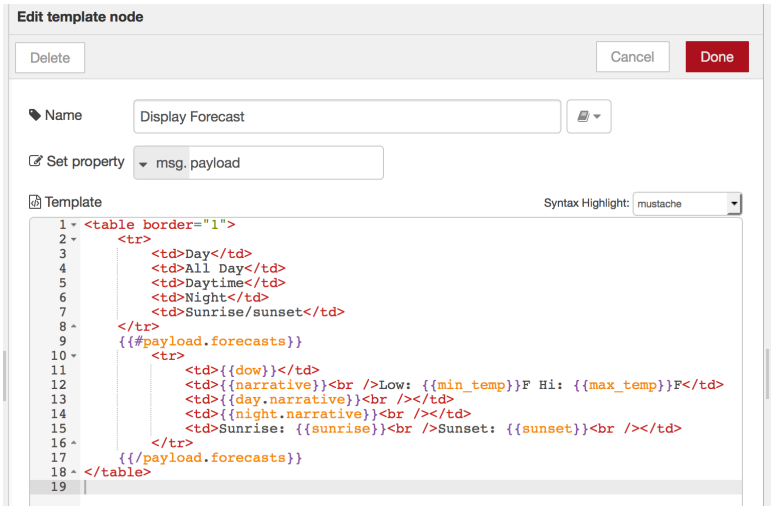
7. If a HTTP status code other than 200 is returned, the application should display an error message. Add a 

node as shown below.



8. For the flow where a webpage should be displayed, add a 

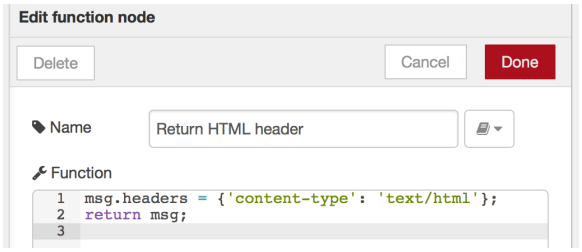
node with the HTML in the file named 4-display-forecast.html

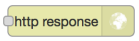


Get the code:
ibm.biz/Bdr4ww

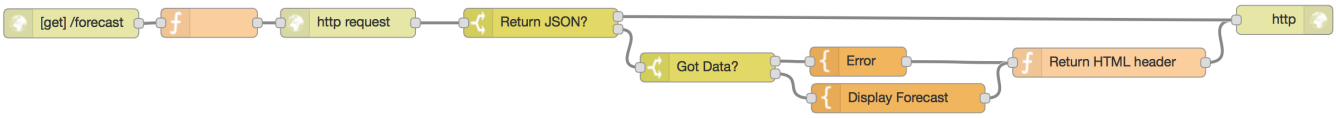
9. Add a 

node to return the correct content type HTTP headers to display a webpage.



10. Add a 

node. Connect the nodes together as shown below.



11. Click on the red 

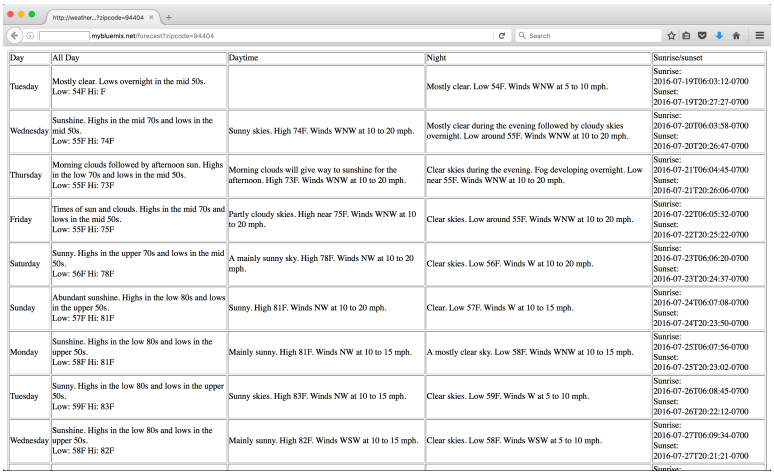
button in the upper-right corner of the screen to save and deploy the changes.

12. Open a new browser tab and visit the application's endpoint, passing in the zip code of a location to find the forecast:

    `http://<<MY-APP>>.mybluemix.net/forecast?zipcode=<<ZIP-CODE>>`

    - Replace <<MY-APP>> with the host of the Node-RED application you chose.
    - Replace <<ZIP-CODE>> with the five-digit zip code of a location

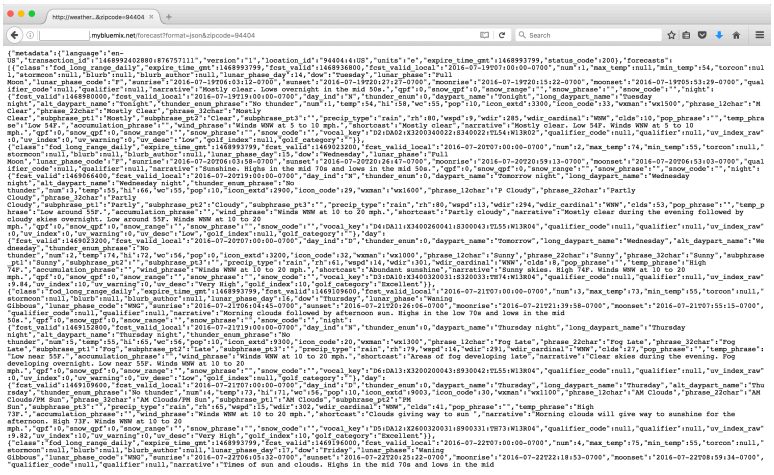
    A table of the next ten days and the weather forecasts for the 24-hour, day and night periods, along with the sunrise and sunset times should be displayed, similar to the one shown below.



Day	All Day	Daytime	Night	Sunrise/sunset
Tuesday	Mostly clear. Lows overnight in the mid 50s. Low: 54F Hi: F		Mostly clear. Low 54F. Winds WNW at 5 to 10 mph.	Sunrise: 2016-07-19T06:03:12-0700 Sunset: 2016-07-19T20:27:27-0700
Wednesday	Sunshine. Highs in the mid 70s and lows in the mid 50s. Low: 55F Hi: 74F	Sunny skies. High 74F. Winds WNW at 10 to 20 mph.	Mostly clear during the evening followed by cloudy skies overnight. Low around 55F. Winds WNW at 10 to 20 mph.	Sunrise: 2016-07-20T06:03:58-0700 Sunset: 2016-07-20T20:26:47-0700
Thursday	Morning clouds followed by afternoon sun. Highs in the low 70s and lows in the mid 50s. Low: 55F Hi: 73F	Morning clouds will give way to sunshine for the afternoon. High 73F. Winds WNW at 10 to 20 mph.	Clear skies during the evening. Fog developing overnight. Low near 55F. Winds WNW at 10 to 20 mph.	Sunrise: 2016-07-21T06:04:45-0700 Sunset: 2016-07-21T20:26:06-0700
Friday	Times of sun and clouds. Highs in the mid 70s and lows in the mid 50s. Low: 55F Hi: 75F	Partly cloudy skies. High near 75F. Winds WNW at 10 to 20 mph.	Clear skies. Low around 55F. Winds WNW at 10 to 20 mph.	Sunrise: 2016-07-22T06:05:32-0700 Sunset: 2016-07-22T20:25:22-0700
Saturday	Sunny. Highs in the upper 70s and lows in the mid 50s. Low: 56F Hi: 78F	A mainly sunny sky. High 78F. Winds NW at 10 to 20 mph.	Clear skies. Low 56F. Winds W at 10 to 20 mph.	Sunrise: 2016-07-23T06:06:20-0700 Sunset: 2016-07-23T20:24:37-0700
Sunday	Abundant sunshine. Highs in the low 80s and lows in the upper 50s. Low: 57F Hi: 81F	Sunny. High 81F. Winds NW at 10 to 20 mph.	Clear. Low 57F. Winds W at 10 to 15 mph.	Sunrise: 2016-07-24T06:07:08-0700 Sunset: 2016-07-24T20:23:50-0700
Monday	Sunshine. Highs in the low 80s and lows in the upper 50s. Low: 58F Hi: 81F	Mainly sunny. High 81F. Winds NW at 10 to 15 mph.	A mostly clear sky. Low 58F. Winds WNW at 10 to 15 mph.	Sunrise: 2016-07-25T06:07:56-0700 Sunset: 2016-07-25T20:23:02-0700
Tuesday	Sunny. Highs in the low 80s and lows in the upper 50s. Low: 59F Hi: 83F	Sunny skies. High 83F. Winds NW at 10 to 15 mph.	Clear skies. Low 59F. Winds W at 5 to 10 mph.	Sunrise: 2016-07-26T06:08:45-0700 Sunset: 2016-07-26T20:22:12-0700
Wednesday	Sunshine. Highs in the low 80s and lows in the upper 50s. Low: 58F Hi: 82F	Mainly sunny. High 82F. Winds WSW at 10 to 15 mph.	Clear skies. Low 58F. Winds WSW at 5 to 10 mph.	Sunrise: 2016-07-27T06:09:34-0700 Sunset: 2016-07-27T20:21:21-0700

13. To see the JSON representation of the forecasts, insert *format=json* in the URL query string:

    `http://<<MY-APP>>.mybluemix.net/forecast?`**`format=json`**`&zipcode=<<ZIP-CODE>>`



EXTRA!

You can also limit the number of days forecasted? Use the following endpoints in Step #2 for 3-, 5-, and 7-day forecasts:

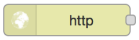
```
msg.url = 'https://twcservice.mybluemix.net:443/api/weather/v1/location/'+
          msg.payload.zipcode+':4:US/forecast/daily/3day.json?units=e';

msg.url = 'https://twcservice.mybluemix.net:443/api/weather/v1/location/'+
          msg.payload.zipcode+':4:US/forecast/daily/5day.json?units=e';

msg.url = 'https://twcservice.mybluemix.net:443/api/weather/v1/location/'+
          msg.payload.zipcode+':4:US/forecast/daily/7day.json?units=e';
```

Time Series

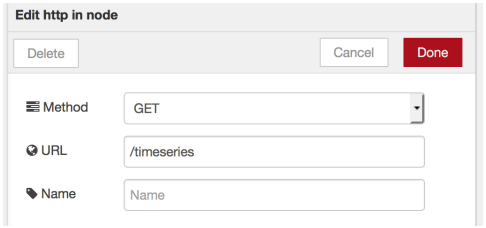
Now that we have the current weather observations and daily forecast, let's go back in time and see what the weather was hour-by-hour. The Weather Company Data for IBM Bluemix API offers a time series API that returns past observations that occurred up to and including the last 24 hours for the location requested.

The recent observations data is continuously updated and replaced with a first-in/first-out methodology (rotating data with newest observation and moving the oldest observations to the archive storage) based on date/time stamping of the observations. The amount of data that is retained and available from any station can be more than 24 individual observation reports. The number of observations is determined by the type of observation it is.

Please refer to the **Add Weather Company Data API in IBM Bluemix** section to create the Weather Company Data service for the application.

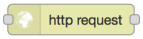
1. Add a 

node as shown below.

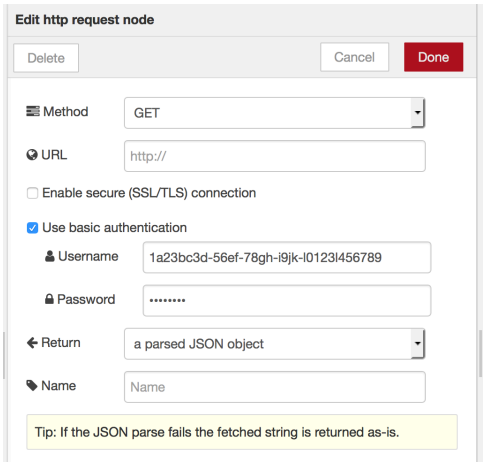


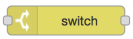
2. Add a 

node as shown below. This will take the URL query parameter, *zipcode,* and place it in the URL used to call the Weather Company Data API. A second URL query parameter, *hours*, is used to limit the number of hours of conditions the API should return.

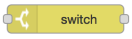
3. Add a 

node. After checking the **Use basic authentication** box, enter the username and password from the Weather Company Data service credentials in Step #4 of the **Add Weather Company Data API in IBM Bluemix** section.

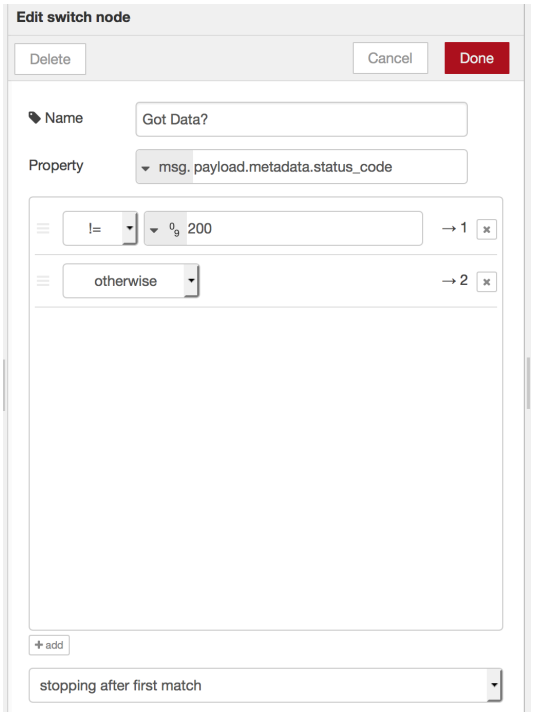


4. To make this application versatile, we'll split the flow so the application will return the results in two formats. The first option will be where the observations are returned in JSON format, great for use in applications that can call the web endpoint and consume the JSON. The second option will be a webpage showing a list of observations in a human friendly report. Add a 

node as shown below.

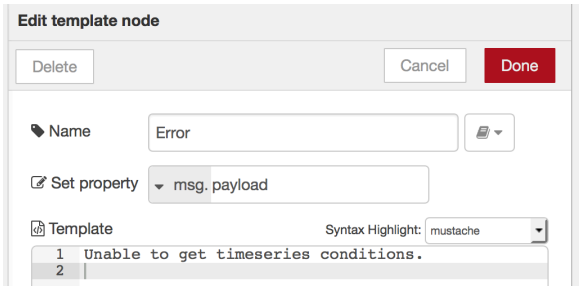
5. For the flow where the JSON should be returned, the application can simply return the contents of *msg.payload*. We'll connect this to the HTTP response node in step #10.
6. For the flow where a webpage should be displayed, the application should first check if there was a successful response from the API. Add a 

node as shown below.



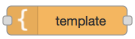
7. If a HTTP status code other than 200 is returned, the application should display an error message. Add a 

node as shown below.

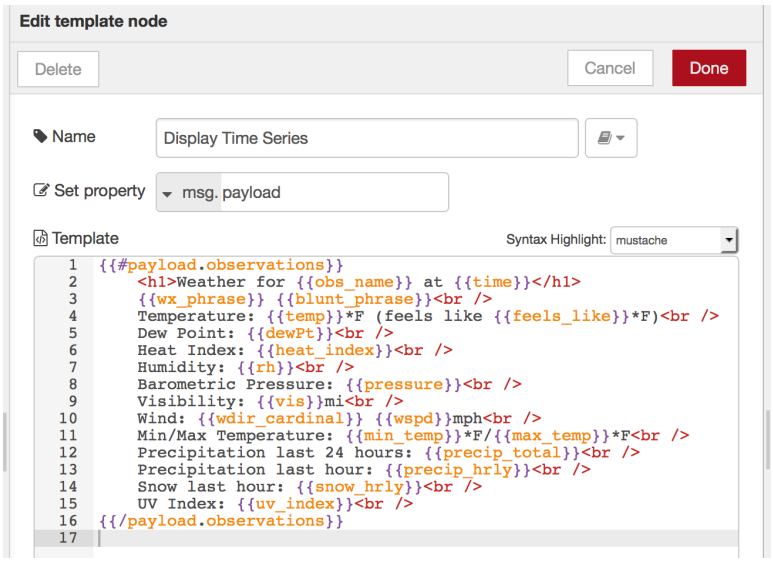


8. For the flow where a webpage should be displayed, we need to add a time friendly format to each observation. Add a 

as shown below. This will add a property named *time* with the JavaScript formatted time that will be used in the template in step #9.

9. Add a 

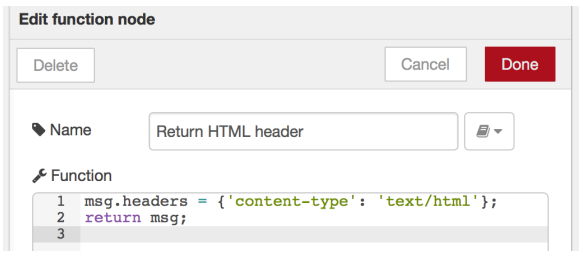
node with the HTML in the file named 5-display-time-series.html

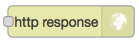


Get the code:
ibm.biz/Bdr4wt

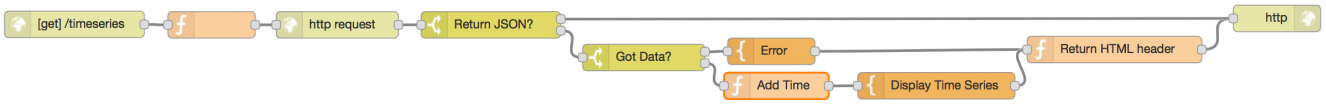
10. Add a 

node to return the correct content type HTTP headers to display a webpage.



11. Add a 

node. Connect the nodes together as shown below.



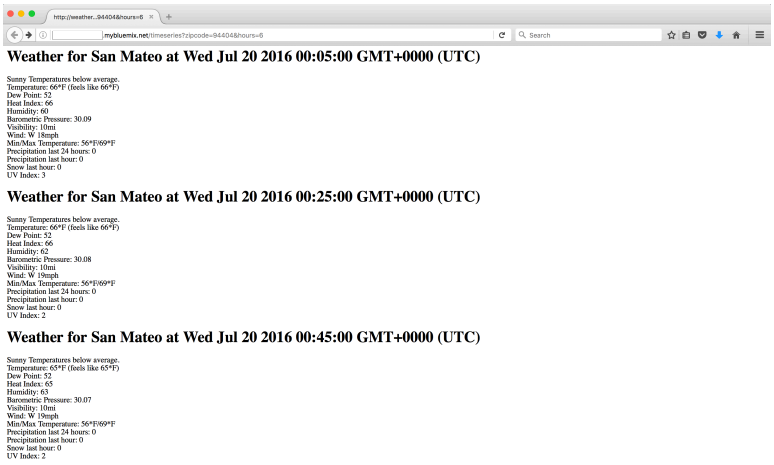
12. Click on the red 

button in the upper-right corner of the screen to save and deploy the changes.
13. Open a new browser tab and visit the application's endpoint, passing in the zip code of a location and the number of hours of observations to display:

```
http://<<MY-APP>>.mybluemix.net/timeseries?zipcode=<<ZIP-CODE>>&hours=<<HOURS>>
```

- Replace <<MY-APP>> with the host of the Node-RED application you chose.
- Replace <<ZIP-CODE>> with the five-digit zip code of a location
- Replace <<HOURS>> with the number of hours of results to be returned (0-23)

The weather observations, along with the time of each observation, are shown for any observations recorded within the specified number of hours at that location.



Weather for San Mateo at Wed Jul 20 2016 00:05:00 GMT+0000 (UTC)

Sunny Temperatures below average.
Temperature: 66°F (feels like 66°F)
Dew Point: 52
Heat Index: 66
Humidity: 60
Barometric Pressure: 30.09
Visibility: 10mi
Wind: W 18mph
Min/Max Temperature: 56°F/69°F
Precipitation last 24 hours: 0
Precipitation last hour: 0
Snow last hour: 0
UV Index: 3

Weather for San Mateo at Wed Jul 20 2016 00:25:00 GMT+0000 (UTC)

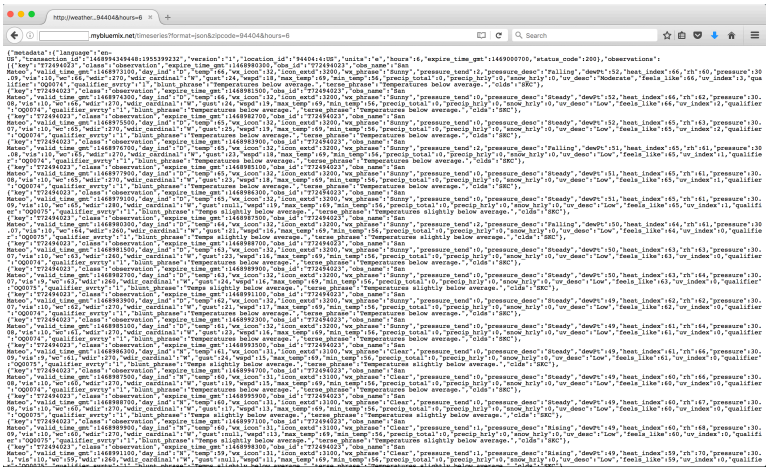
Sunny Temperatures below average.
Temperature: 66°F (feels like 66°F)
Dew Point: 52
Heat Index: 66
Humidity: 62
Barometric Pressure: 30.08
Visibility: 10mi
Wind: W 19mph
Min/Max Temperature: 56°F/69°F
Precipitation last 24 hours: 0
Precipitation last hour: 0
Snow last hour: 0
UV Index: 2

Weather for San Mateo at Wed Jul 20 2016 00:45:00 GMT+0000 (UTC)

Sunny Temperatures below average.
Temperature: 65°F (feels like 65°F)
Dew Point: 52
Heat Index: 65
Humidity: 63
Barometric Pressure: 30.07
Visibility: 10mi
Wind: W 19mph
Min/Max Temperature: 56°F/69°F
Precipitation last 24 hours: 0
Precipitation last hour: 0
Snow last hour: 0
UV Index: 2

14. To see the JSON representation of the time series of observations, insert *format=json* in the URL query string:

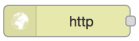
```
http://<<MY-APP>>.mybluemix.net/timeseries?format=json&zipcode=<<ZIP-CODE>>&hours=<<HOURS>>
```



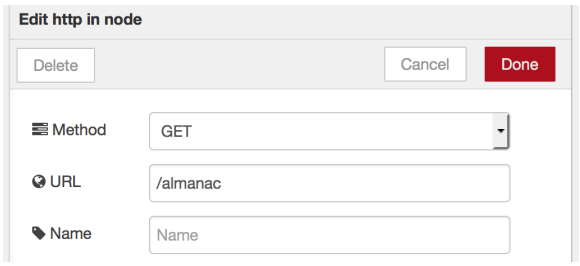
Almanac

The Weather Company Data for IBM Bluemix API offers an almanac API that returns the daily almanac for the location identified by location. The almanac information for this API is sourced from National Weather Service observations stations from a time period spanning 10 to 30 years or more. The information is gathered and provided by the National Climatic Data Center (NCDC). In this section, we will add an endpoint that returns the weather almanac data for a period of dates and see min/max temperatures and the years they occurred, average and mean temperatures, and amounts of precipitation and snowfall.

Please refer to the **Add Weather Company Data API in IBM Bluemix** section to create the Weather Company Data service credentials for the application.

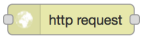
1. Add a 

node as shown below.

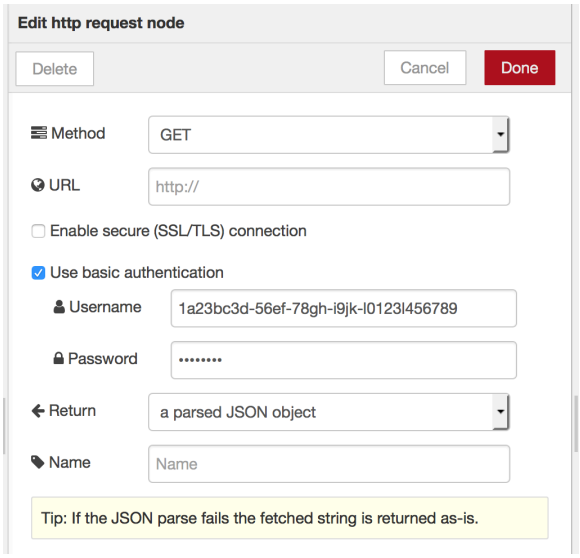


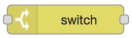
2. Add a 

node as shown below. This will take the URL query parameter, *zipcode,* and place it in the URL used to call the Weather Company Data API.

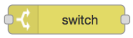
3. Add a 

node. After checking the **Use basic authentication** box, enter the username and password from the Weather Company Data service credentials in Step #4 of the **Add Weather Company Data API in IBM Bluemix** section.



4. To make this application versatile, we'll split the flow so the application will return the results in two formats. The first option will be where the results are returned in JSON format, great for use in applications that can call the web endpoint and consume the JSON. The second option will be a webpage showing the list in a human readable format. Add a 

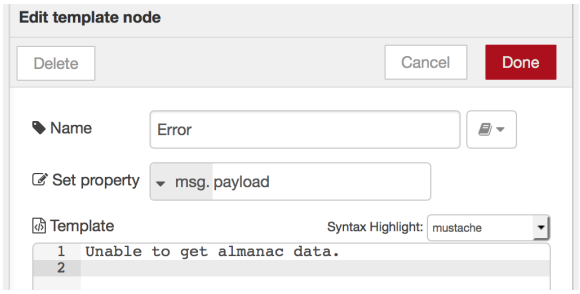
node as shown below.

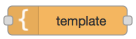
5. For the flow where the JSON should be returned, the application can simply return the contents of *msg.payload*. We'll connect this to the HTTP response node in step #10.
6. For the flow where a webpage should be displayed, the application should first check if there was a successful response from the API. Add a 

node as shown below.

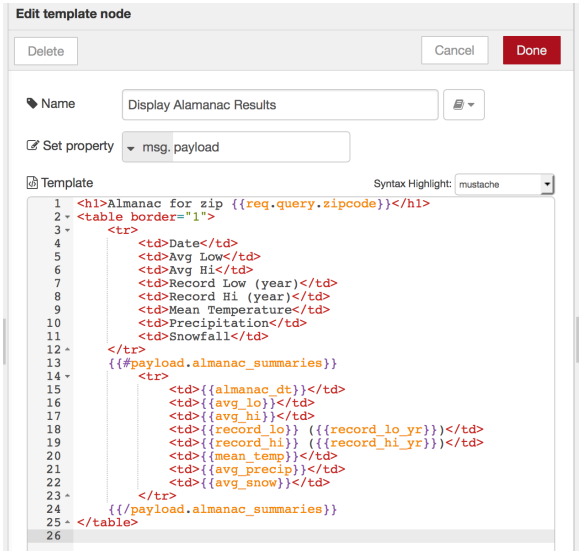
7. If a HTTP status code other than 200 is returned, the application should display an error message. Add a 

node as shown below.



8. For the flow where a webpage should be displayed, add a 

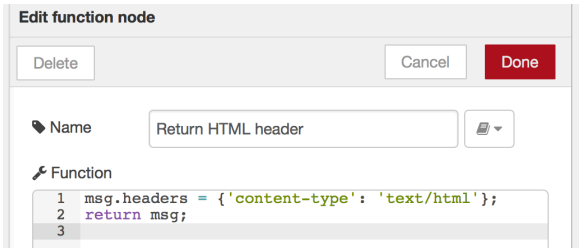
node with the HTML in the file named 6-display-almanac-results.html

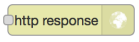


Get the code:
ibm.biz/Bdr4w5

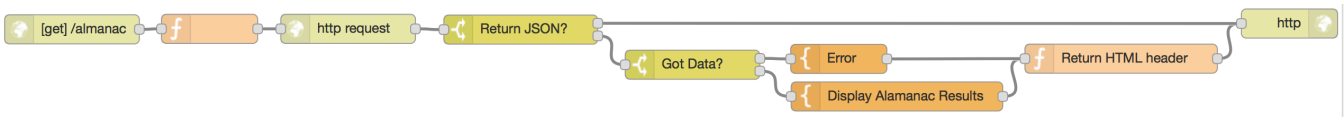
9. Add a 

node to return the correct content type HTTP headers to display a webpage.



10. Add a 

node. Connect the nodes together as shown below.



11. Click on the red 

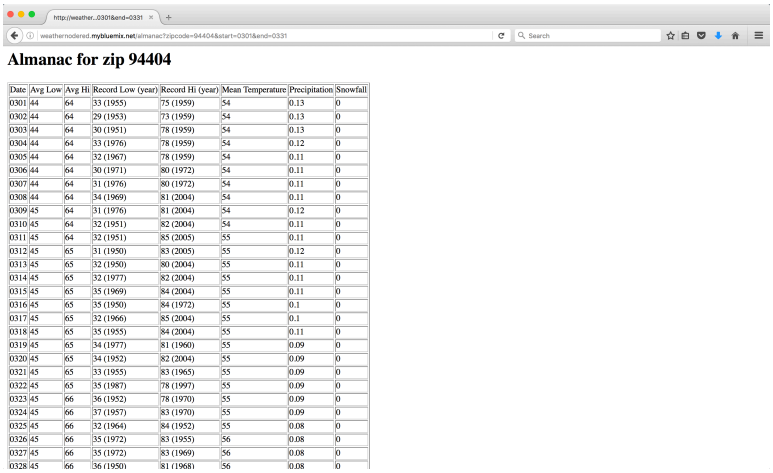
button in the upper-right corner of the screen to save and deploy the changes.

12. Open a new browser tab and visit the application's endpoint, passing in the zip code of a location to find the current conditions:

    `http://<<MY-APP>>.mybluemix.net/almanac?zipcode=<<ZIP-CODE>>&start=<<MMDD>>&end=<<MMDD>>`

    - Replace <<MY-APP>> with the host of the Node-RED application you chose.
    - Replace <<ZIP-CODE>> with the five-digit zip code of a location.
    - Replace <<MMDD>> with the start and end dates in the two-digit month (single digit months should be preceded by a zero), two digit day format. March 14th would be 0314.

    For each day in the date period specified, the recorded low and high temperatures, along with other weather attributes, are shown in the table.

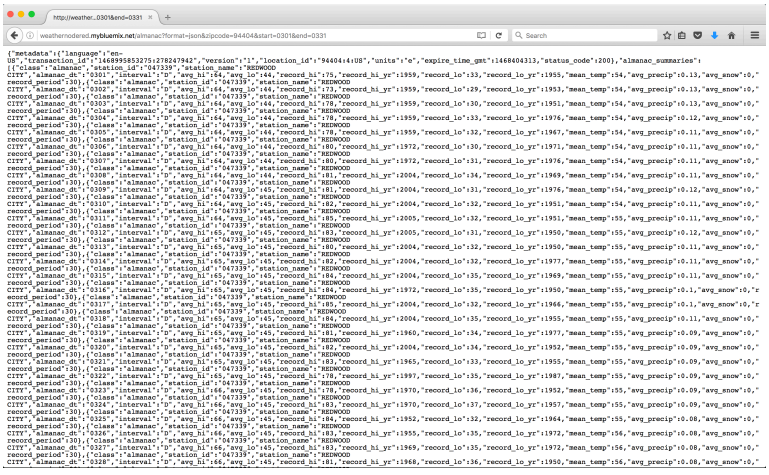


Almanac for zip 94404

Date	Avg Low	Avg Hi	Record Low (year)	Record Hi (year)	Mean Temperature	Precipitation	Snowfall
0301	44	64	33 (1955)	75 (1959)	54	0.13	0
0302	44	64	29 (1953)	73 (1959)	54	0.13	0
0303	44	64	30 (1951)	78 (1959)	54	0.13	0
0304	44	64	33 (1976)	78 (1959)	54	0.12	0
0305	44	64	32 (1967)	78 (1959)	54	0.11	0
0306	44	64	30 (1971)	80 (1972)	54	0.11	0
0307	44	64	31 (1976)	80 (1972)	54	0.11	0
0308	44	64	34 (1969)	81 (2004)	54	0.11	0
0309	45	64	31 (1976)	81 (2004)	54	0.12	0
0310	45	64	32 (1951)	82 (2004)	54	0.11	0
0311	45	64	32 (1951)	85 (2005)	55	0.11	0
0312	45	65	31 (1950)	83 (2005)	55	0.12	0
0313	45	65	32 (1950)	80 (2004)	55	0.11	0
0314	45	65	32 (1977)	82 (2004)	55	0.11	0
0315	45	65	35 (1969)	84 (2004)	55	0.11	0
0316	45	65	35 (1950)	84 (1972)	55	0.1	0
0317	45	65	32 (1966)	85 (2004)	55	0.1	0
0318	45	65	35 (1955)	84 (2004)	55	0.11	0
0319	45	65	34 (1977)	81 (1960)	55	0.09	0
0320	45	65	34 (1952)	82 (2004)	55	0.09	0
0321	45	65	33 (1955)	83 (1965)	55	0.09	0
0322	45	65	35 (1987)	78 (1997)	55	0.09	0
0323	45	66	36 (1952)	78 (1970)	55	0.09	0
0324	45	66	37 (1957)	83 (1970)	55	0.09	0
0325	45	66	32 (1964)	84 (1952)	55	0.08	0
0326	45	66	35 (1972)	83 (1955)	56	0.08	0
0327	45	66	35 (1972)	83 (1969)	56	0.08	0
0328	45	66	36 (1950)	81 (1968)	56	0.08	0

13. To see the JSON representation of the almanac data, insert *format=json* in the URL query string:

`http://<<MY-APP>>.mybluemix.net/almanac?`**`format=json`**`&zipcode=<<ZIP-CODE>>&start=<<MMDD>>&end=<<MMDD>>`



Appendix

The table shown below lists API endpoints offered by the Weather Company Data for IBM Bluemix API. For more information, please visit https://twcservice.mybluemix.net.

10-Day Daily Forecast by Geocode	/v1/geocode/{latitude}/{longitude}/forecast/daily/10day.json
10-Day Daily Forecast by Postal Code	/v1/location/{locationId}/forecast/daily/10day.json
10-Day Intraday Forecast by Geocode	/v1/geocode/{latitude}/{longitude}/forecast/intraday/10day.json
10-Day Intraday Forecast by Postal Code	/v1/location/{locationId}/forecast/intraday/10day.json
3-Day Daily Forecast by Geocode	/v1/geocode/{latitude}/{longitude}/forecast/daily/3day.json
3-Day Daily Forecast by Postal Code	/v1/location/{locationId}/forecast/daily/3day.json
3-Day Intraday Forecast by Geocode	/v1/geocode/{latitude}/{longitude}/forecast/intraday/3day.json
3-Day Intraday Forecast by Postal Code	/v1/location/{locationId}/forecast/intraday/3day.json
48-Hour Hourly Forecast by Geocode	/v1/geocode/{latitude}/{longitude}/forecast/hourly/48hour.json
48-Hour Hourly Forecast by Postal Code	/v1/location/{locationId}/forecast/hourly/48hour.json
5-Day Daily Forecast by Geocode	/v1/geocode/{latitude}/{longitude}/forecast/daily/5day.json
5-Day Daily Forecast by Postal Code	/v1/location/{locationId}/forecast/daily/5day.json
5-Day Intraday Forecast by Geocode	/v1/geocode/{latitude}/{longitude}/forecast/intraday/5day.json
5-Day Intraday Forecast by Postal Code	/v1/location/{locationId}/forecast/intraday/5day.json
7-Day Daily Forecast by Geocode	/v1/geocode/{latitude}/{longitude}/forecast/daily/7day.json
7-Day Daily Forecast by Postal Code	/v1/location/{locationId}/forecast/daily/7day.json
7-Day Intraday Forecast by Geocode	/v1/geocode/{latitude}/{longitude}/forecast/intraday/7day.json
7-Day Intraday Forecast by Postal Code	/v1/location/{locationId}/forecast/intraday/7day.json
Alert Details	/v1/alert/{detail_key}/details.json
Alert Headlines by Country and State Codes	/v1/country/{countrycode}/state/{statecode}/alerts.json
Alert Headlines by Country Code	/v1/country/{countrycode}/alerts.json
Alert Headlines by Country Code and Area ID Code	/v1/country/{countrycode}/area/{areaid}/alerts.json
Alert Headlines by Geocode	/v1/geocode/{latitude}/{longitude}/alerts.json
Almanac-Daily by Geocode	/v1/geocode/{latitude}/{longitude}/almanac/daily.json
Almanac-Daily by Postal Code	/v1/location/{locationId}/almanac/daily.json
Almanac-Monthly by Geocode	/v1/geocode/{latitude}/{longitude}/almanac/monthly.json
Almanac-Monthly by Postal Code	/v1/location/{locationId}/almanac/monthly.json
Location Services-Point by Postal Code	/v3/location/point
Location Services-Search by City Name	/v3/location/search
Site-Based Current Conditions Forecast by Geocode	/v1/geocode/{latitude}/{longitude}/observations.json
Site-Based Current Conditions Forecast by Postal Code	/v1/location/{locationId}/observations.json
Site-Based Time Series Observations by Geocode	/v1/geocode/{latitude}/{longitude}/observations/timeseries.json
Site-Based Time Series Observations by Postal Code	/v1/location/{locationId}/observations/timeseries.json

{{locationId}}
Format: PostalCode:4:CountryCode
Where:

- PostalCode is any valid postal code for a TWC supported country
- The digit 4 is the Location Type 4
- CountryCode is any valid country code for a TWC supported country
- Example: 30339:4:US