Práctica BM006 – Bluemix Node-RED Starter

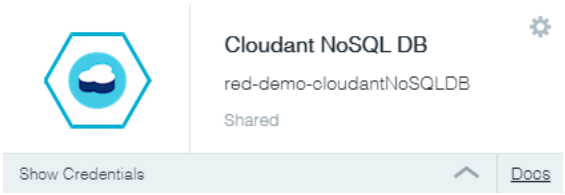
Marzo 2015

Node-RED es una herramienta de código abierto que te permite construir aplicaciones simplemente ligando elementos entre sí en un ambiente gráfico. Estos elementos pueden representar dispositivos de hardware, APIs web o servicios accesibles en internet.

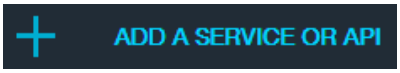
En Bluemix es muy sencillo crear un ambiente Node-RED que pueda acceder a los servicios de la plataforma. En este documento te mostraremos como construir tu primera aplicación usando esta herramienta.

Creando la aplicación

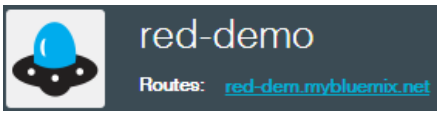
1. En el catálogo de servicios, selecciona el ícono Node-RED Starter de la sección Boilerplates para crear una nueva aplicación **y asígnale un nombre único.**
2. Agrega los servicios que utilizará tu aplicación.
   a. El boilerplate incluye una base de datos **Cloudant NoSQL** que utilizaremos en este ejemplo (recuerda el nombre, lo vas a utilizar después).



   b. Puedes agregar otros servicios a la aplicación para que sean accesibles en el entorno de Node-RED.



3. Accede al editor gráfico.
   a. Accede a la URL generada para tu aplicación http://<myapp>.mybluemix.net
   b. Se desplegará una página con información sobre Node-RED y un botón con la leyenda **"Go to your Node-RED flow editor"** que te llevará al editor.



Estás listo para comenzar a crear flujos.

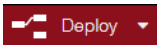
Tu primer flujo

En el panel izquierdo encontrarás diferentes nodos para manejar entradas, salidas y funciones que puedes utilizar en tus flujos. La sección central presenta espacio en blanco para diseñar los flujos. Finalmente el panel de la derecha tiene una pestaña para desplegar información del nodo seleccionado y una pestaña para desplegar mensajes de debug.

Todo flujo comienza con una entrada. La sección **input** del panel de nodos contiene diferentes formas de ingresar datos a la aplicación. Comenzaremos con la forma más simple, una entrada manual desde el mismo editor.

1. Agrega un botón **inject** arrastrándolo a la sección central y da doble click sobre éste para modificar las opciones.
   a. En la opción **payload** selecciona **string**, y en el seguiente campo escribe la cadena "ejemplo".

2. Agrega un nodo **debug** que será la salida de este primer flujo. Este nodo escribirá en la consola la propiedad payload del mensaje que se le envía.
3. Liga la salida del nodo inject con la entrada del nodo debug, de esta manera el primer nodo enviará su contenido al siguiente cuando sea activado.



4. Despliega el flujo pulsando el botón **Deploy** de la esquina superior derecha.

El flujo ya está activo. Para enviar un mensaje por medio del nodo inject pulsa el botón en el lado izquierdo del nodo. En el panel derecho, la pestaña **debug** mostrará la cadena enviada.

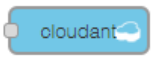
Estos son los pasos básicos para crear un flujo y desplegarlo. Ahora agreguemos algo de funcionalidad.

Agregando funcionalidad

Ahora comenzaremos a interactuar con el servicio de Cloudant NoSQL DB que se agregó a la aplicación.

1. Agrega un nodo **function** al flujo. Este tipo de nodos definen funciones JavaScript para manipular los mensajes entre nodos. Este nodo tomará como parámetro de entrada el objeto **msg** del nodo que lo invoca y enviará como mensaje al nodo siguiente un objeto o arreglo de objetos.
2. En las propiedades del nodo escribe la siguiente función que guardará las propiedades “cadena” y “fecha” en el campo `payload` del objeto que devolverá:

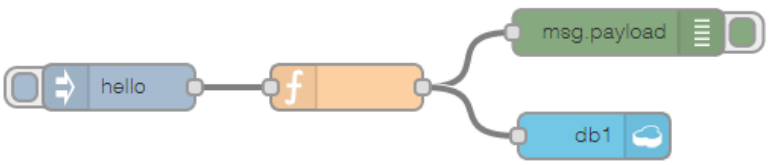
```
var date = new Date();
var obj = {
    "cadena":msg.payload,
    "fecha":date};
msg.payload=obj;
return msg;
```



3. Agrega un nodo **cloudant out** por medio del cual se insertará el objeto recién creado a la base de datos (ten cuidado de no confundirlo con el nodo cloudant in, el ícono es muy parecido). Configura las siguientes propiedades del nodo:

   Service: selecciona el nombre del servicio que está ligado a la aplicación
   Database: db1
   Operation: insert
   Only store msg.payload object? : checked

4. Liga la salida del nodo inject a la función y la salida de ésta al nodo Cloudant out. El flujo quedará ahora de esta manera:



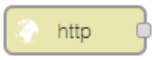
5. Despliega la aplicación.

Ahora, cuando actives el nodo inject un objeto con la cadena que se le ha pasado y la fecha actual se enviará al servicio de base de datos. El contenido también se enviará a la consola.

¿Cómo sabes que realmente se insertó el objeto? Vamos a desplegar el contenido de la base de datos.

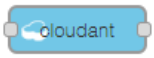
Desplegando los datos

Dejemos de interactuar sólo con el editor gráfico y despleguemos el contenido de la base de datos en una página web.



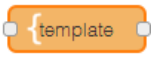
1. Crea un nodo **http in**. Este constituye otra forma de entrada así que será el inicio de otro flujo. Configura las siguientes propiedades:

   Method: GET
   url: /db1



2. Agrega un nodo **cloudant in** que usaremos para consultar los objetos en la base de datos. Configura las siguientes propiedades:

   Service: selecciona el nombre del servicio que está ligado a la aplicación
   Database: db1
   Search by: all documents



3. Crea un nodo **template** con el que se construirá el contenido de la página. Este nodo utiliza la notación **mustache** para construir el contenido. Si no estás familiarizado con esta notación puedes obtener más información en http://mustache.github.io/mustache.5.html

   Agrega el siguiente código en las propiedades del nodo:

```
<b>Contenido:</b><br>
{{#payload}}
   {{fecha}} : {{cadena}} <br>
{{/payload}}
```

4. Crea un nodo **http response** para generar una salida tipo HTTP del flujo.

5. Liga los nodos de la siguiente manera:

6. Despliega la aplicación.

El flujo será invocado al acceder la dirección http://<my-app>.mybluemix.net/db1 en un browser. Se desplegará una lista de los objetos insertados en la base de datos.

Completando la aplicación

Para finalizar agregaremos una forma para insertar los objetos desde la misma página. Primero vamos a substituir la entrada del flujo de inserción por una llamada HTTP POST.

1. En el flujo de inserción (el primero que creaste) agrega un nodo **http in** y configura las siguientes propiedades:

   Method: POST
   url: /db1

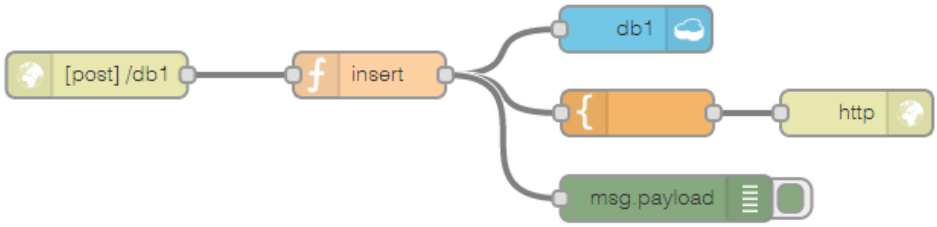
2. Modifica el nodo **function**, ahora el texto para incluir en el objeto vendrá de una forma HTML. El código ahora debe quedar así:

```
var obj = {
    "cadena":msg.req.body["texto"],
    "fecha":date};
```

3. Inserta un nodo **template** para crear el contenido que devolverá la página. En este caso se insertará el contenido de la llamada GET a `/db1`. El contenido del nodo queda como sigue:

```
<iframe src="/db1" frameborder="0"
  style="width: 100%; height: 100%">
</iframe>
```

4. Agrega un nodo **http response** para substituir el nodo *inject* y liga los nodos de la siguiente manera:



Finalmente, de vuelta al flujo de consulta (el segundo que construiste):

5. Agrega al contenido del nodo **template** el siguiente código que mostrará una forma HTML después de la lista de objetos en la base de datos.

```
<form action="/db1"
    method="post" target="_top">
  <input type="text" name="texto"/>
  <button type="submit">Enviar</button>
</form>
```

6. Despliega la aplicación

Al invocar de nuevo la URL http://<my-app>.mybluemix.net/db1 se desplegará la lista de objetos en la base de datos además de una forma para insertar uno nuevo. Al enviar el formulario se realizará una llamada POST que iniciará el flujo de inserción.

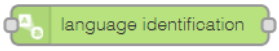
La nueva lista desplegada mostrará el nuevo registro creado.

¡Felicidades! Has creado una aplicación que utiliza servicios de Bluemix usando Node-RED.

¡Es tu turno!

La mejor forma de comprobar lo que has aprendido es haciendo un nuevo flujo por ti mismo, ¿qué te parece usar un servicio de IBM Watson para esto?

Este es el requerimiento:



Construye una forma HTML para ingresar un texto y utiliza el nodo **language identification** para identificar y desplegar el lenguaje en el que está escrito dicho texto.

Tip: En el entorno de Bluemix agrega a tu aplicación el servicio Language Identification de la sección de servicios Watson.

Bonus: Si el texto enviado no está en español, tradúcelo utilizando los servicios de IBM Watson en Bluemix.