



Creating CRUD Application with Cloudfant, IBM Bluemix and Node JS

AP Cloud Lab | Miracle Innovation Labs

Team Miracle

Miracle Software Systems, Inc.

November 9th, 2017

Creating CRUD Application with Cloudant, IBM Bluemix and Node JS

Goal

In this lab we will guide you how to create a CRUD application with Cloudant using Node.js for backend, Bootstrap and HTML for frontend. You will then deploy the application to IBM Bluemix.

Pre-Requisites

The following installations will need to be completed for this lab to be run successfully,

- Account with IBM Bluemix
- Node.js and NPM Installed
- Cloudant account for storing data
- Text Editor such as Sublime Text (or) Notepad ++

Technology Involved

- Server Side - NodeJS
- Client Side Technologies (HTML, CSS, Bootstrap)
- Cloud Technologies – IBM Bluemix
- Database - IBM Cloudant


Lab Steps


Let's get started with the lab!

#1 | Creating an IBM Cloudant account

For creating a new Cloudant account visit the below link and click on Sign Up button,

<https://cloudant.com/sign-up/>

 IBM Cloudant

Product ▾ Resources ▾ Contact Us ▾ [Sign In](#) [Sign Up](#) 

Create New Account - Existing Cloudant Enterprise Customers Only!

Starting on July 24, 2017, all new signups for Cloudant are through IBM Bluemix leveraging the new and improved [pricing model](#)! Please go to [IBM Bluemix](#) to sign up/in today!

If you are an existing Cloudant Enterprise user with a dedicated cluster and need a new account on it, please fill out the form to the right. The account will be provisioned on a temporary staging environment. Once created, login to the Cloudant Dashboard and

Username .cloudant.com
Must include only alphanumeric or dash, and at least three characters

Password
Must be a strong password of at least sixteen characters


First Name


Last Name

Company or Institution

Email

Provide the details and click on Create Account button for new registration to Cloudant. Click on Sign in for logging into the account.

 IBM Cloudant

Product ▾ Resources ▾ Contact Us ▾ [Sign In](#) [Sign Up](#) 

Sign In to Your Account

Note to existing Cloudant users: Starting on July 24, 2017, all new signups for Cloudant Shared plan will be through IBM Bluemix leveraging the new and improved [pricing model](#). Please go to [IBM Bluemix](#) to sign up/in today. Existing users can continue to login to their accounts here.

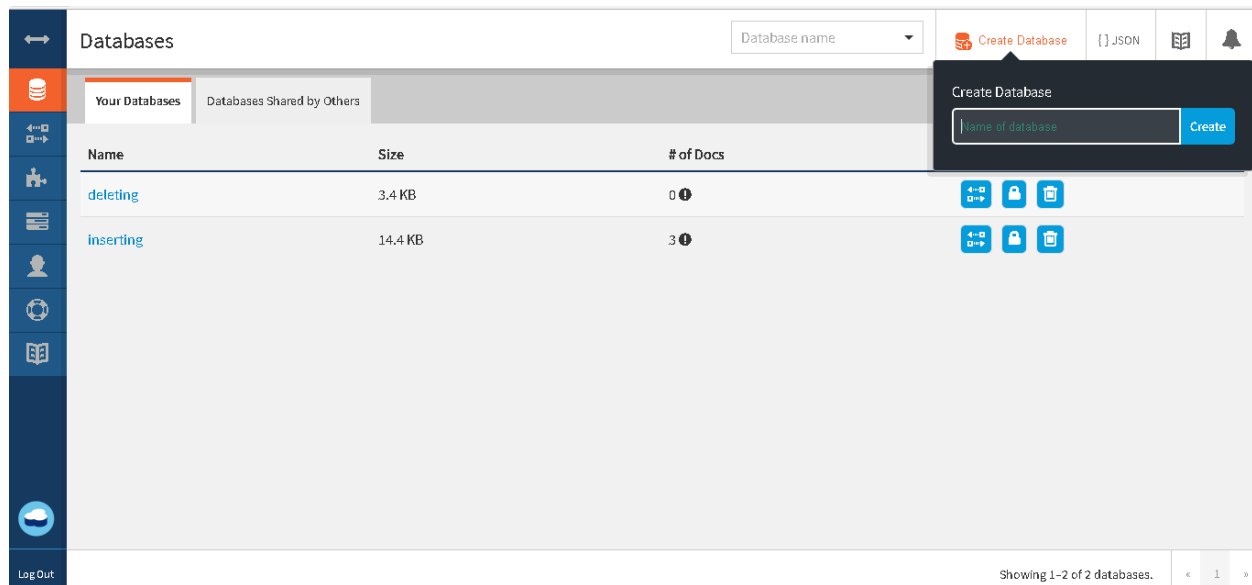
Username .cloudant.com

Password

[Forgot your password?](#)

[Sign In](#) [Don't have an account?](#)

Provide Cloudant credentials and login to your account. After signing in you should be able to see the below dashboard.



By clicking on create database (you can find Create Database option on top right corner), we can create a new database, by clicking that option, a popup will appear on the screen which asks to enter database name.

Also we can do the same through Node JS code from server side. Below are the steps to perform Cloudant CRUD operations using NodeJS.

#2 | Initializing Cloudant in Node JS

Below are the steps to establish the connection between Cloudant and Node JS

1. Install the Cloudant NPM module in Node JS
2. Import the module in the code
3. Provide your Cloudant credentials (Username and Password) in Node JS

Installing Cloudant Module

Open Command Prompt and give the following command,

npm install cloudant --save

Import Cloudant Module

To import this module in NodeJS, use the below statement in the script file

```
var Cloudant = require('cloudant');
```

Database Connection

Below is the code snippet for connecting to Cloudant database.

```
var user = <'your Cloudant user name'>;  
var pwd= <'your Cloudant password'>;  
var cloudant = Cloudant ({account : user, password: pwd});
```

#3 | Creating Cloudant API's for CRUD operations

To create and insert data into Cloudant database using Node.js follow the below steps,

Inserting Data

Copy and paste the code of Create.html in GitHub Repo **crud-app** folder for front end code and for API creation - below is code snippet,

```
app.get('/create', function(req, res) {  
  var obj = { <JSON Data Object>;  
  cloudant.db.create('inserting', function() {  
    var alice = cloudant.db.use('inserting');  
    alice.insert(obj, function(err) {  
      if (err) {  
        return console.log('[alice.insert] ', err.message);  
      }  
      res.send("Inserted Successfully");  
    });  
  });  
});
```

Retrieving Data

Copy and paste the code of Retrieve.html in GitHub Repo **crud-app** folder for front end code and for API creation - below is code snippet

```
app.get('/ret', function(req, res) {  
  var id = req.query.ss;  
  var alice = cloudant.db.use('inserting');
```

```
var bb = {
  "selector": {
    _id: id
  }
};
alice.find(bb, function(err, result) {
  if (err)
    throw err;
  var obb = result.bookmark;
  if (obb === "nil") {
    res.send("no data found");
  } else {
    res.send(result)
  }
}) });
```

Updating Data

Copy and paste the code of Update.html in GitHub Repo **crud-app** folder for front end code and for API creation - below is code snippet

```
app.get('/update', function(req, res) {
  var id = req.query.ss;
  var alice = cloudant.db.use('inserting');
  var bb = {
    "selector": {
      _id: id
    }
  };
  alice.find(bb, function(err, result) {
    if (err)
      throw err;
    var obb = result.bookmark;
    if (obb === "nil") {
      res.send("no data found");
    } else {
      res.send(result);
    }
  });
});
```

```
    }  
  })  
});
```

Deleting Data

Copy and paste the code of Delete.html in GitHub Repo **crud-app** folder for front end code and for API creation - below is code snippet

```
app.get('/delete', function(req, res) {  
  var email1 = req.query.ss;  
  var alice = cloudant.db.use('inserting');  
  var bb = {  
    "selector": {  
      _id: email1  
    }  
  };  
  alice.find(bb, function(err, result) {  
    if (err)  
      throw err;  
    var obb = result.bookmark;  
    if (obb === "nil") {  
      res.send("no data found");  
    } else {  
      alice.destroy(b, a, function(err) {  
        if (err) {  
          throw err;  
        }  
        res.send("Deleted Successfully");  
      })  
    }  
  })  
});
```

#4 | Rendering HTML Pages

For rendering HTML pages in Node JS, we need to add the following snippet in our Server code.

```
app.use(express.static(__dirname + '/', {  
  index: 'Create.html'  
}));
```

#5 | Consuming CRUD API's

In the above portion we created APIs for performing CRUD operations, to make use of those APIs we need to consume API's in front-end using JavaScript.

In this application we created 4 APIs named Create, Update, Retrieve and Delete.

Here is the code snippet for consuming API in JavaScript

```
var xhttp = new XMLHttpRequest();  
xhttp.open("POST", "Your Rest URL Here", true);  
xhttp.setRequestHeader("Content-type", "application/json");  
xhttp.send();  
var response = JSON.parse(xhttp.responseText);
```

You can find the complete code in GitHub repository



#6 | Run the application

Navigate to the workspace folder where the code exists, and open command prompt

Run **node app.js**


```
C:\Windows\System32\cmd.exe - node app.js
D:\IoT\Manasa SAP Cloud\CRUDApp\HTMLScreen-Final>node app.js
running at 3001
```

Application is running at <http://localhost:3001/>
Open in browser, then you can see the below page.



MEAN CRUD Application Demo

CreateRetrieveDeleteUpdate

Name

Author

ID

Title

Volume

Keywords

Pages

SubmitReset

© 2017 Miracle Software Systems, Inc.

Built by [Miracle's Innovation Labs](#)

Insert the data as below. After inserting the data will be stored in Cloudant database

C Programming
Dennis Ritchie
1
The C Programming Language
1
C
250

Submit

Reset

For retrieving the data, click on Retrieve button.

Open IBM Cloudant Dashboard and check whether the data is inserted or not.

The screenshot shows the IBM Cloudant Dashboard interface. On the left is a sidebar with navigation links: All Documents, Query, Permissions, Changes, Design Documents, and Log Out. The main area displays a document with ID 1. The document structure is as follows:

id	key	value
1	1	{ "rev": "8-b38764adfa077598a4eb27708d..." }

At the bottom of the dashboard, it indicates "Showing document 1 - 1. Documents per page: 20".

Open the ID and view the data that is available

inserting > 1

Save Changes Cancel

Upload Attachment Clone Document Delete

1 {
2 "_id": "1",
3 "_rev": "8-b38764adfa077598a4eb27708d120a7e",
4 "dname": "C Programmin",
5 "dauthor": "Dennis Ritchie",
6 "dtitle": "The C Programming Language",
7 "dvolume": "1",
8 "dkeywords": "C",
9 "dpages": "250"
10 }

Log Out

Get Data

Provide the ID for which you want to retrieve the data. Here we are retrieving ID '1', the data will be displayed as below,

Get Data

C Programming

Denis Ritchie

1

The C Programming Language

1

C

250


Data retrieved successfully!

For updating the data, click on Update button.

Get Data

Provide the ID of the data which you want to change the values. After updating the data those updated values will be stored in database successfully.

MIRACLE
SOFTWARE SYSTEMS



MEAN CRUD Application Demo

CreateRetrieveDeleteUpdate

Get Data

Update


Reset

© 2017 Miracle Software Systems, Inc.

Built by **Miracle's Innovation Labs**

For deleting the document from database, click on Delete tab.

MIRACLE
SOFTWARE SYSTEMS



MEAN CRUD Application Demo



CreateRetrieveDeleteUpdate

Delete Data

© 2017 Miracle Software Systems, Inc.

Built by **Miracle's Innovation Labs**

Provide the ID value which you want to delete.



MEAN CRUD Application DemoCreateRetrieveDeleteUpdate

Delete Data

Data deleted successfully

© 2017 Miracle Software Systems, Inc.

Built by [Miracle's Innovation Labs](#)

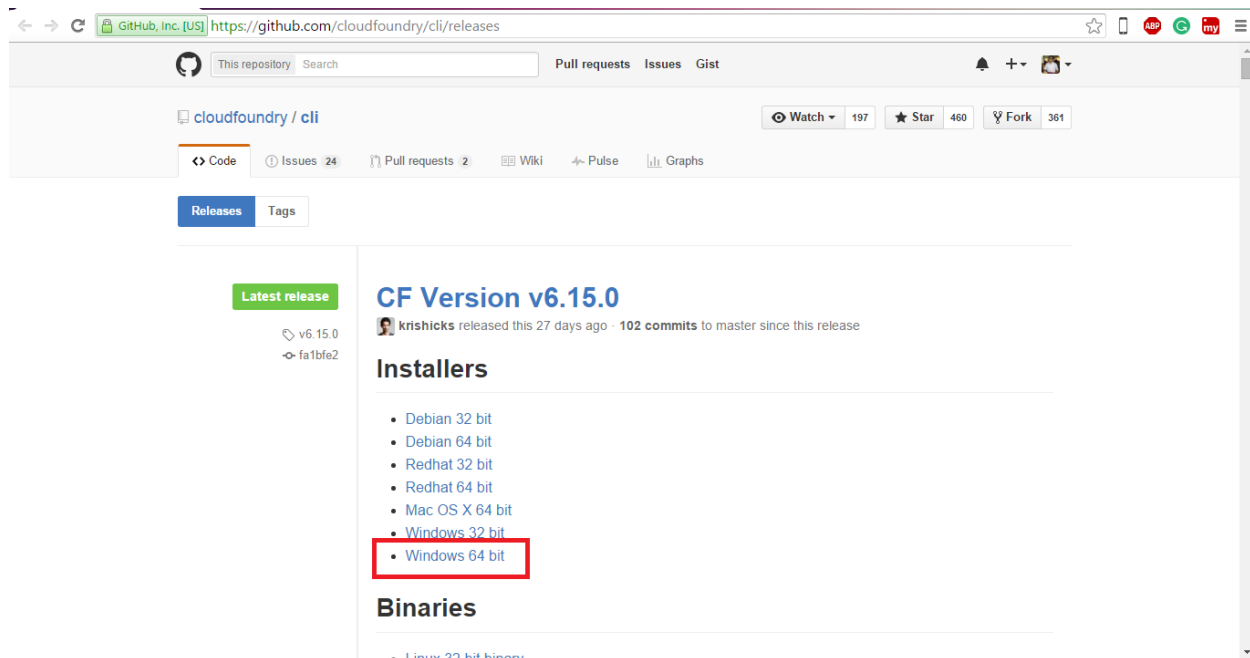
This data will be permanently deleted in the database.

#7 | Installing the CF CLI

At this point you will be asked how you want to start coding your application. For this lab, we will be using the **Cloud Foundry (CF) CLI** option.

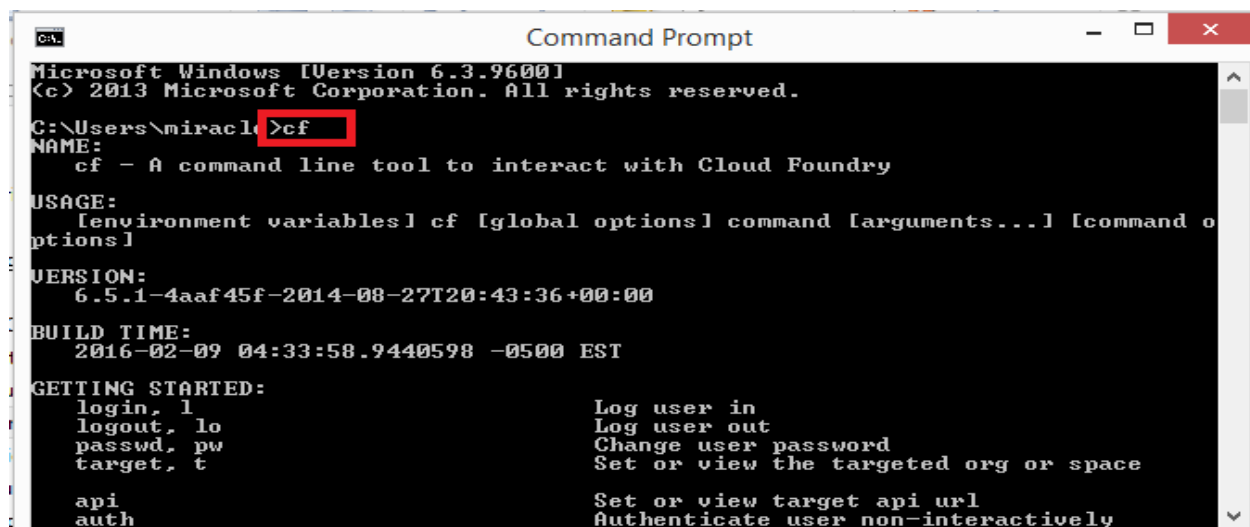
Open this link for installing and downloading,

<https://github.com/cloudfoundry/cli>



Then you will get a zip file. After extracting the zip file, you can find a **.exe** file inside. Install it.

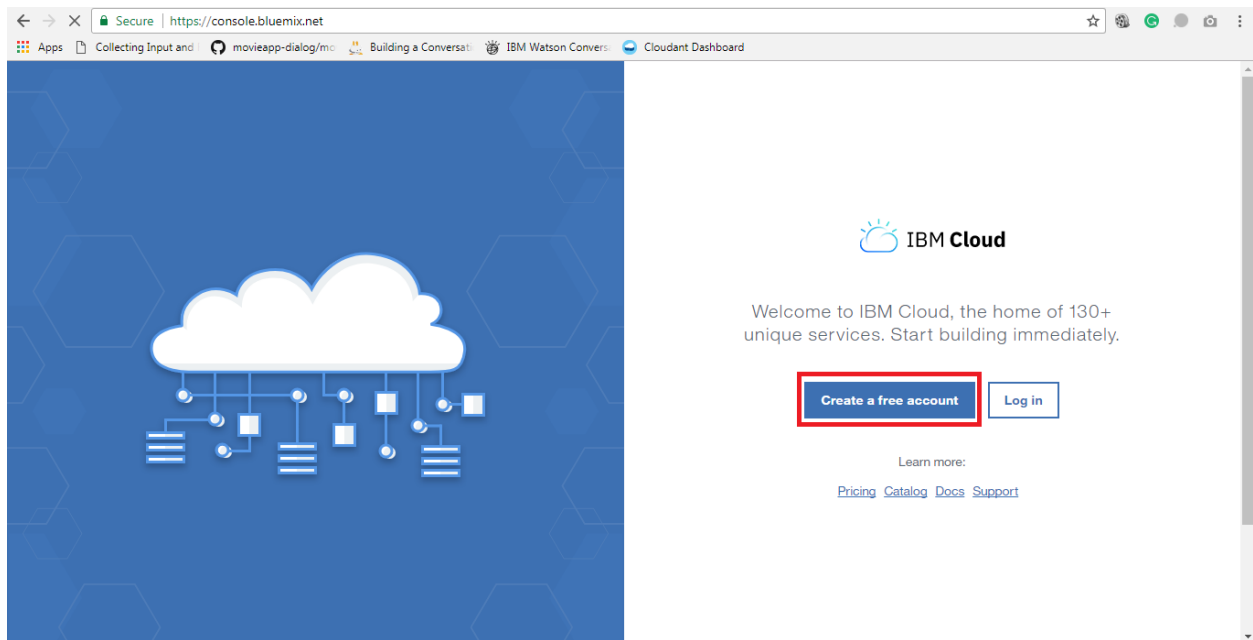
To check whether **CF** is installed properly or not, open command prompt and execute CF command. Then it will show you a set of **CF** commands, which indicates that CF is successfully installed on your machine.




#8 | Creating an IBM Bluemix account

The next step will be to make sure that we have access to the IBM Bluemix Console with either the free trial option (or) the paid subscription option.

Login to Bluemix at <http://bluemix.net> (or) Register today at <https://console.ng.bluemix.net/registration/>



Click on Create a free account, and the fields as required.

 IBM Cloud

Sign up for an IBMid and create your IBM Cloud account


Build on IBM Cloud for free with no time restrictions

Guaranteed free development with Lite plans
Develop worry-free and at no cost with cap based Lite plan services for as long as you like.

Start on your projects right away
Skip entering your credit card info and get working in just a few short steps.

Get \$200 on us to try paid services
Ease into cloud pricing or try something new with \$200 in credit available for 1 month upon upgrade.

Ready to get started? Sign up today!

 IBM Cloud

Sign up for an IBMid and create your IBM Cloud account

Build on IBM Cloud for free with no time restrictions

Guaranteed free development with Lite plans
Develop worry-free and at no cost with cap based Lite plan services for as long as you like.

Start on your projects right away
Skip entering your credit card info and get working in just a few short steps.

Get \$200 on us to try paid services
Ease into cloud pricing or try something new with \$200 in credit available for 1 month upon upgrade.

Ready to get started? Sign up today!

Already have an IBM Cloud account? [Log in](#)

Email*

*Enter an email address.

First Name*

Last Name*

Company

Country or Region*

United States

Phone Number*

miracle software systems

Country or Region*

United States

Phone Number*

9493415290


Password*


Keep me informed of products, services, and offerings from IBM companies worldwide.

☒ By email ☐ By telephone

By clicking Create Account, I accept the [IBM Cloud privacy policy](#) and [IBM Cloud terms](#).

Create Account

 Privacy - Terms

 Privacy - Terms

After Clicking on "**Create Account**", confirmation mail will be sent to the registered mail id. Click on Confirm account and then Login to your Bluemix account.

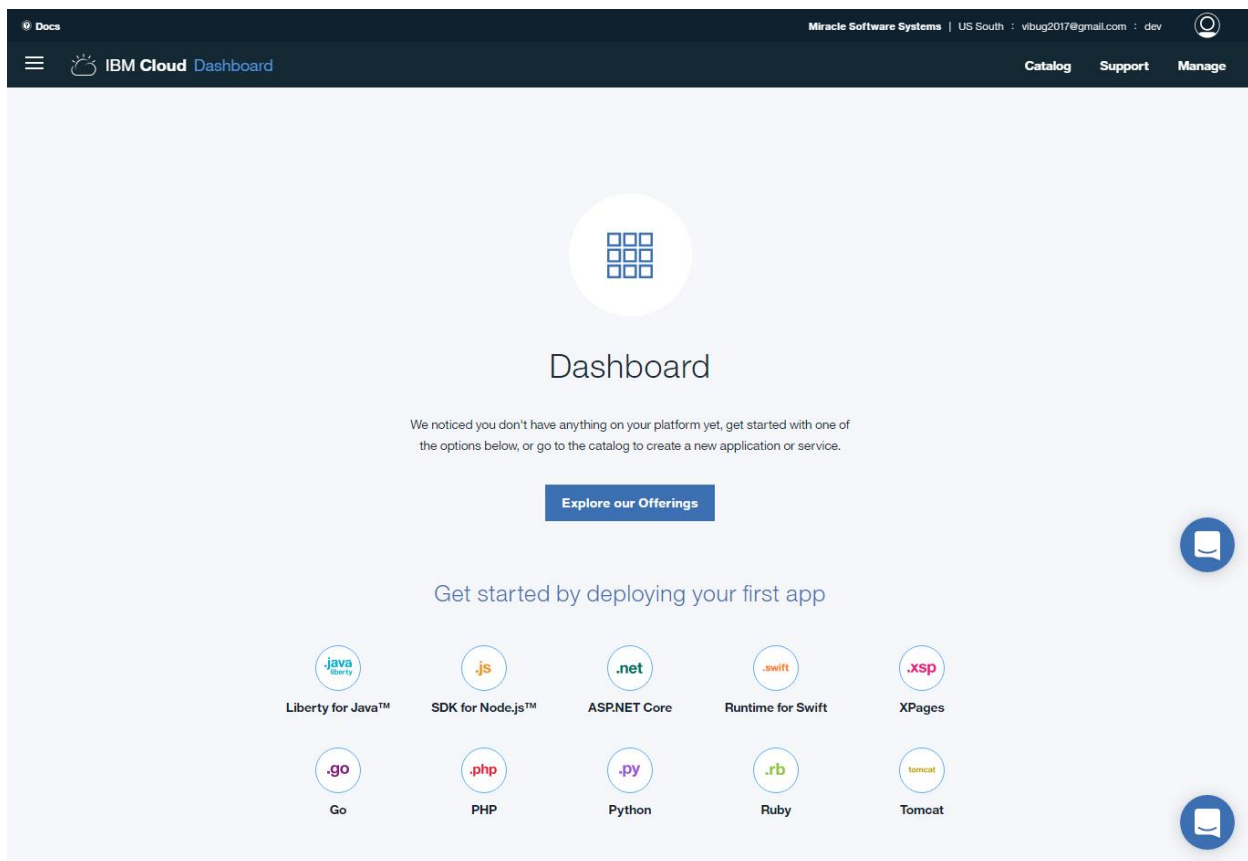
Log into IBM Cloud

Enter Email or IBMid: [Forgot your IBMid?](#)

Continue

New? [Create an IBM Cloud account](#)

After you login, you can see the dashboard where you can take a look at your applications and services.



The screenshot shows the IBM Cloud Dashboard interface. At the top, there's a dark navigation bar with 'Docs' on the left and 'Miracle Software Systems | US South : vibug2017@gmail.com : dev' on the right. Below this, a secondary bar contains a hamburger menu, the 'IBM Cloud Dashboard' title, and links for 'Catalog', 'Support', and 'Manage'. The main content area has a large circular icon with a 3x3 grid of squares. Below the icon, the word 'Dashboard' is centered. A message states: 'We noticed you don't have anything on your platform yet, get started with one of the options below, or go to the catalog to create a new application or service.' A blue button labeled 'Explore our Offerings' is positioned below the message. Further down, the heading 'Get started by deploying your first app' is followed by a grid of ten circular icons representing different technologies: Liberty for Java™, SDK for Node.js™, ASP.NET Core, Runtime for Swift, XPages, Go, PHP, Python, Ruby, and Tomcat. On the right side of the dashboard, there are two floating chat bubbles.

#9 | Deploying the application to IBM Bluemix

The next step will be to take your application and deploy it back to Bluemix so that you can share it with your friends.

Add manifest file, for pushing the application to IBM Bluemix.

applications:

- path: .

memory: 256M

instances: 1

domain: mybluemix.net

name: crud-node-app

host: crud-node-app

disk_quota: 1024M

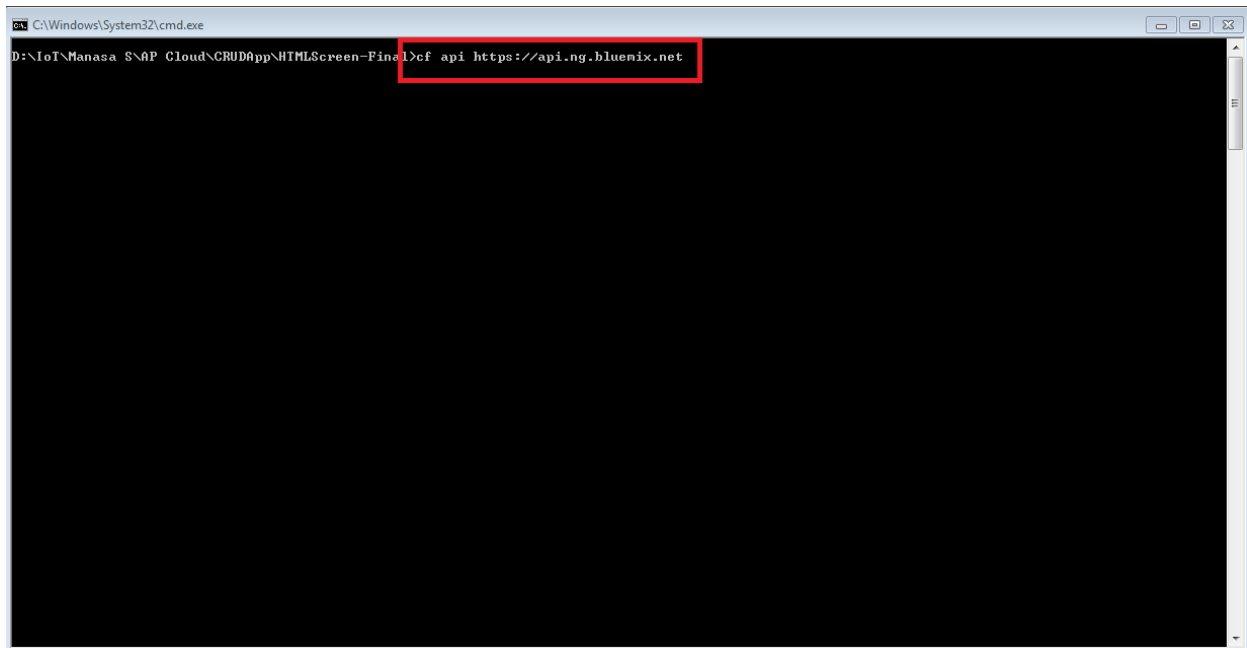
Name the file as manifest.yml and save in the same folder.

Open the **Command Prompt** and go to the location where you have your workspace. Then, connect to Bluemix using one of the following commands (Depends on which region you selected in your profile).

For Sydney: cf api <https://api.au-syd.bluemix.net>

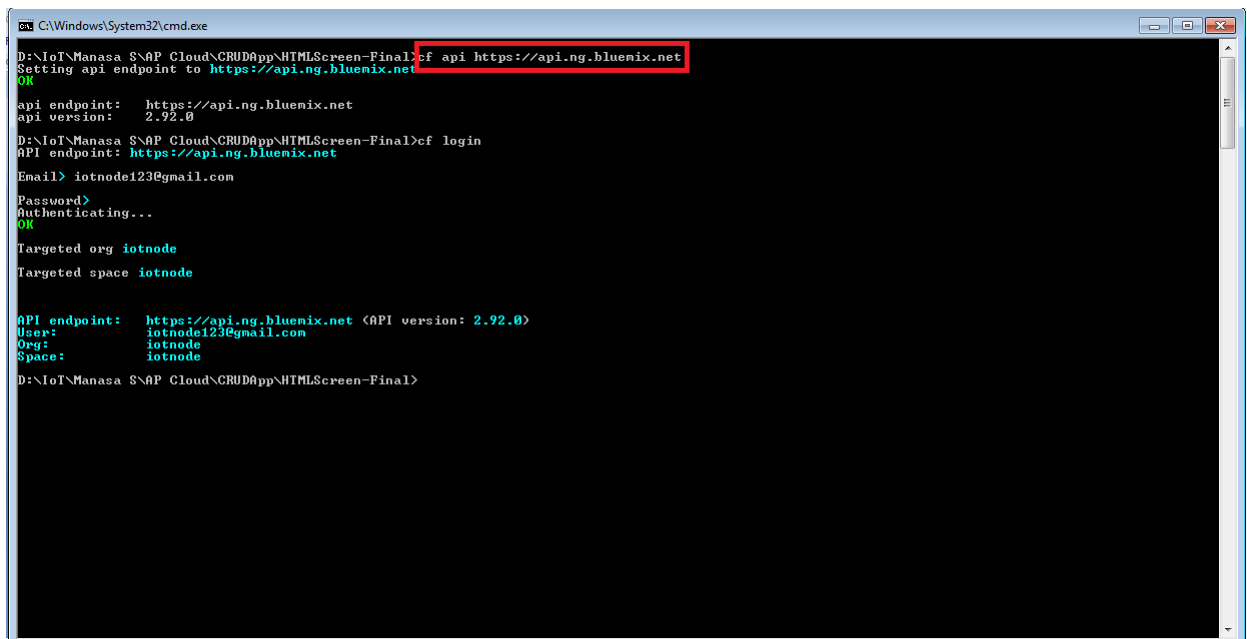
For US South: cf api <https://api.ng.bluemix.net>

For United Kingdom: cf api <https://api.eu-gb.bluemix.net>



```
C:\Windows\System32\cmd.exe
D:\IoT\Manasa S\AP Cloud\CRUDApp\HTMLScreen-Final>cf api https://api.ng.bluemix.net
```

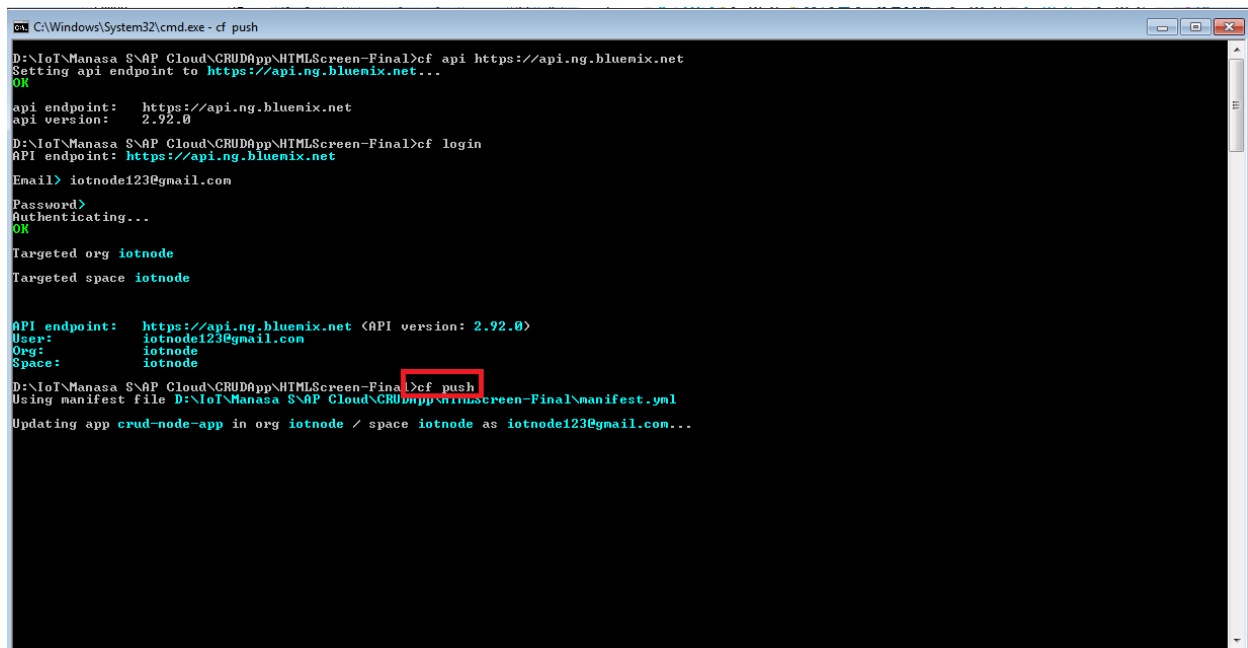
Login to Bluemix using the “**cf login**” command, and when prompted enter your user ID and password to login.



```
C:\Windows\System32\cmd.exe
D:\IoT\Manasa S\AP Cloud\CRUDApp\HTMLScreen-Final>cf api https://api.ng.bluemix.net
Setting api endpoint to https://api.ng.bluemix.net
OK
api endpoint: https://api.ng.bluemix.net
api version: 2.92.0
D:\IoT\Manasa S\AP Cloud\CRUDApp\HTMLScreen-Final>cf login
API endpoint: https://api.ng.bluemix.net
Email> iotnode123@gmail.com
Password>
Authenticating...
OK
Targeted org iotnode
Targeted space iotnode

API endpoint: https://api.ng.bluemix.net (API version: 2.92.0)
User: iotnode123@gmail.com
Org: iotnode
Space: iotnode
D:\IoT\Manasa S\AP Cloud\CRUDApp\HTMLScreen-Final>
```

Make sure that you are within your application's directory and use the “**cf push**” command to push your application to your Bluemix Organization.



```

C:\Windows\System32\cmd.exe - cf push

D:\IoT\Manasa $\AP Cloud\CRUDApp\HTMLScreen-Final>cf api https://api.ng.bluemix.net
Setting api endpoint to https://api.ng.bluemix.net...
OK
api endpoint: https://api.ng.bluemix.net
api version: 2.92.0

D:\IoT\Manasa $\AP Cloud\CRUDApp\HTMLScreen-Final>cf login
API endpoint: https://api.ng.bluemix.net
Email> iotnode123@gmail.com
Password>
Authenticating...
OK
Targeted org iotnode
Targeted space iotnode

API endpoint: https://api.ng.bluemix.net (API version: 2.92.0)
User: iotnode123@gmail.com
Org: iotnode
Space: iotnode

D:\IoT\Manasa $\AP Cloud\CRUDApp\HTMLScreen-Final>cf push
Using manifest file D:\IoT\Manasa $\AP Cloud\CRUDApp\HTMLScreen-Final\manifest.yml
Updating app crud-node-app in org iotnode / space iotnode as iotnode123@gmail.com...
```

Note: This process might take around 3 to 5 minutes for completion

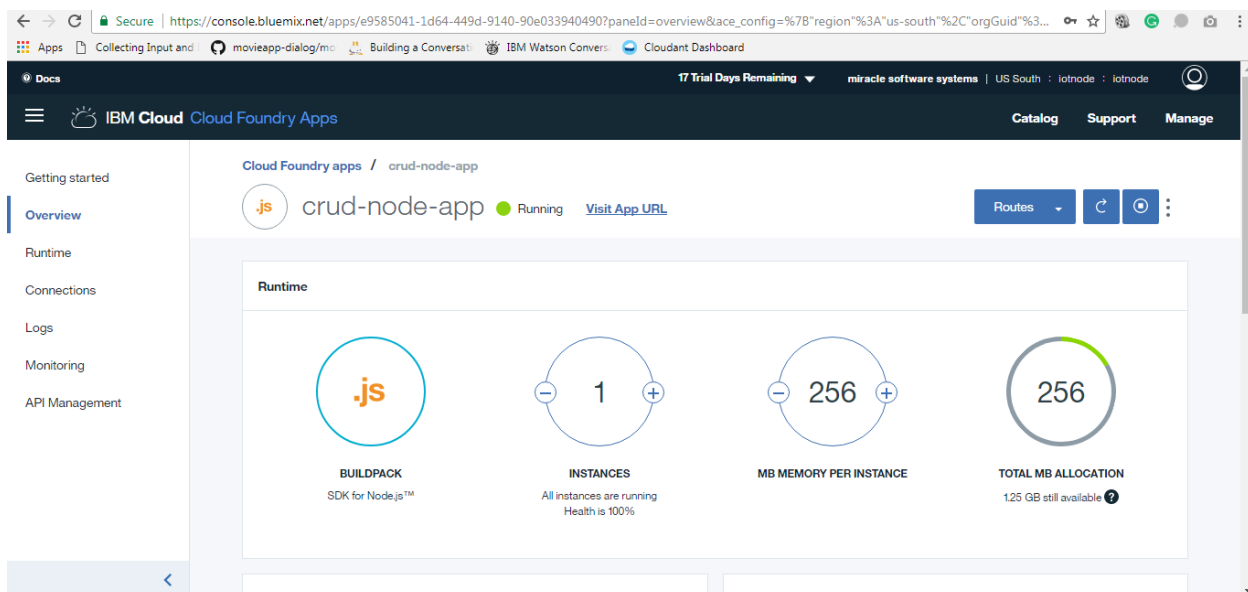
Once your application is pushed, your Command Prompt should look as shown below,

```

C:\Windows\System32\cmd.exe
-----> - hower_components (not cached - skipping)
-----> Building dependencies
-----> Installing App Management
-----> Caching build
-----> Clearing previous node cache
Saving 2 cacheDirectories (default):
- node_modules
- hower_components (nothing to cache)
-----> Build succeeded!
    cloudant@1.8.0
    dotenv@4.0.0
    express@4.16.2
    multer@1.3.0
Exit status 0
Staging complete
Uploading droplet, build artifacts cache...
Uploading build artifacts cache...
Uploading droplet...
Uploaded build artifacts cache (3.5M)
Uploaded droplet (22.7M)
Uploading complete
Stopping instance e3931f31-eef5-40e8-8b5e-d45fc581a960
Destroying container
Successfully destroyed container
1 of 1 instances running
App started
OK
App crud-node-app was started using this command './vendor/initial_startup.rb'
Showing health and status for app crud-node-app in org iotnode / space iotnode as iotnode123@gmail.com...
OK
requested state: started
instances: 1/1
usage: 256M x 1 instances
urls: crud-node-app.nybluemix.net
last uploaded: Thu Nov 9 11:21:25 UTC 2017
stack: cflinuxfs2
buildpack: SDK for Node.js(TM) (ibm-node.js-6.11.4, buildpack-v3.15-20171024-1528)
#0 state since cpu memory disk details
running 2017-11-09 05:22:27 AM 0.3% 29.2M of 256M 96.7M of 1G
D:\IoI\Manasa $AP Cloud\CRUDApp\HTMLScreen-Final>

```

Now, you can go back to your Bluemix account in the browser and access your applications URL through **Dashboard->Application Overview->Application URL**.



‘Your very own application, that you created and deployed in IBM Bluemix, should now be available as shown below!’

Name
Author
ID
Title
Volume
Keywords
Pages

Submit

Reset

Now, perform the same operations as above.