MIRACLE
SOFTWARE SYSTEMS

Deploying your first Node.js Application to IBM Bluemix

Miracle Summer of Code Virtual Labs Series

Pavani Gedala
MEAN Stack Developer
Miracle Software Systems, Inc.

April 26th, 2016

©2016 Miracle Software Systems, Inc.

Miracle's SoC Series : Deploying your first Node Application to IBM Bluemix

Overview

In this lab we will be creating a sample node application in Cloud9(A Online Editor) and will then deploy this application to IBM Bluemix by using the Cloud Foundry CLI.

Prerequisites

You will need the following to complete this lab successfully,

- Active email ID for registering with Bluemix
- Up to date browser to access Cloud9 and Bluemix
- Registered and active accounts with Cloud9 and Bluemix

Note : You will not be required to download (or) install anything on your laptops for this lab, but this lab can be recreated on your laptop instead of using C9.

Technology Involved

The following technologies will be covered in this lab,

- IBM Bluemix(PaaS)
- Node.js and NPM
- Express.js(Routing Framework)
- Cloud9(Web IDE)
- Cloud Foundry CLI(Open PaaS)
- HTML/CSS/JavaScript

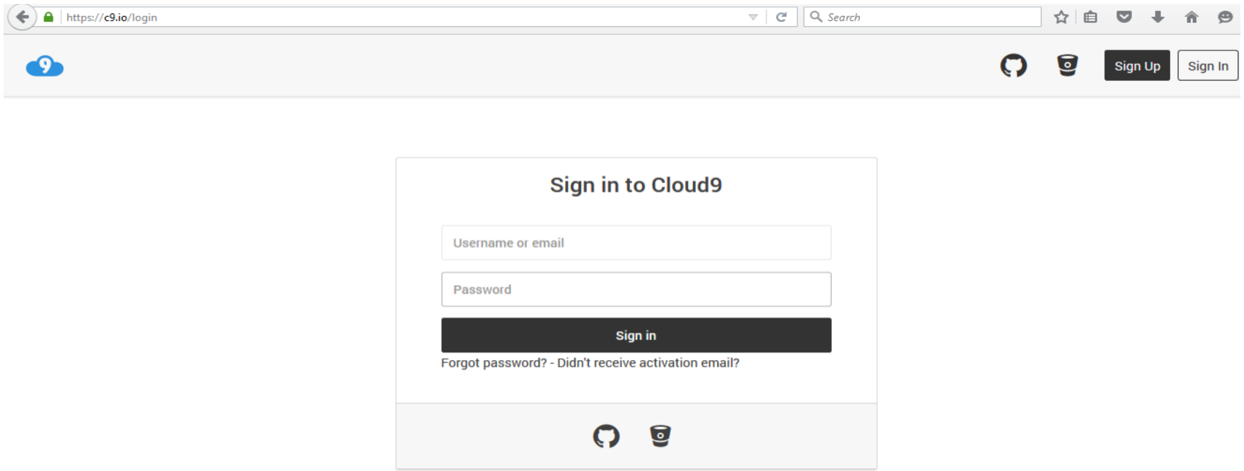
**Basic knowledge on web applications and using the command line will be beneficial but not entirely required*

Lab Steps

So, let us get started with the lab!

#1 | Access Cloud9

The first step would be to access Cloud9 and either register (or) login to your account. Login at https://c9.io/login (or) register at https://c9.io/signup.

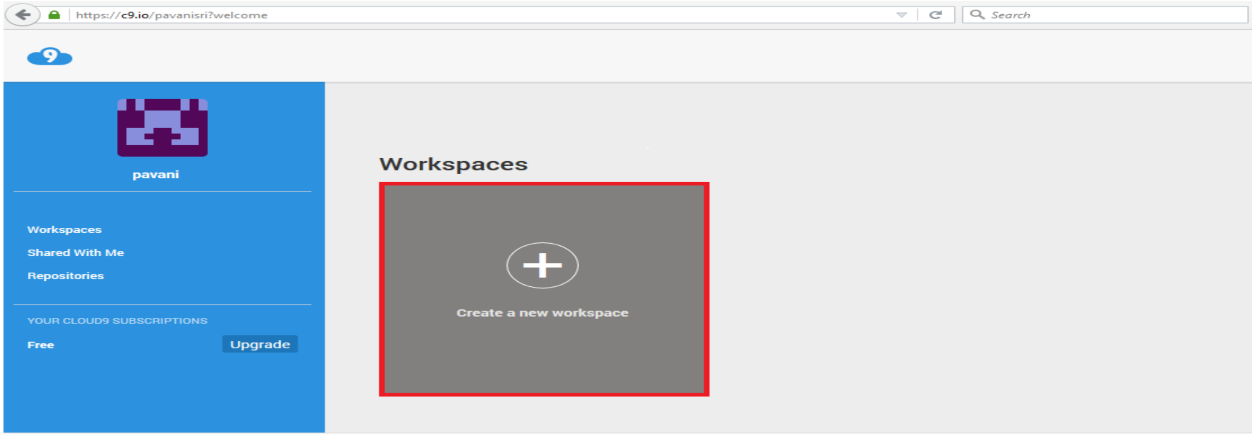


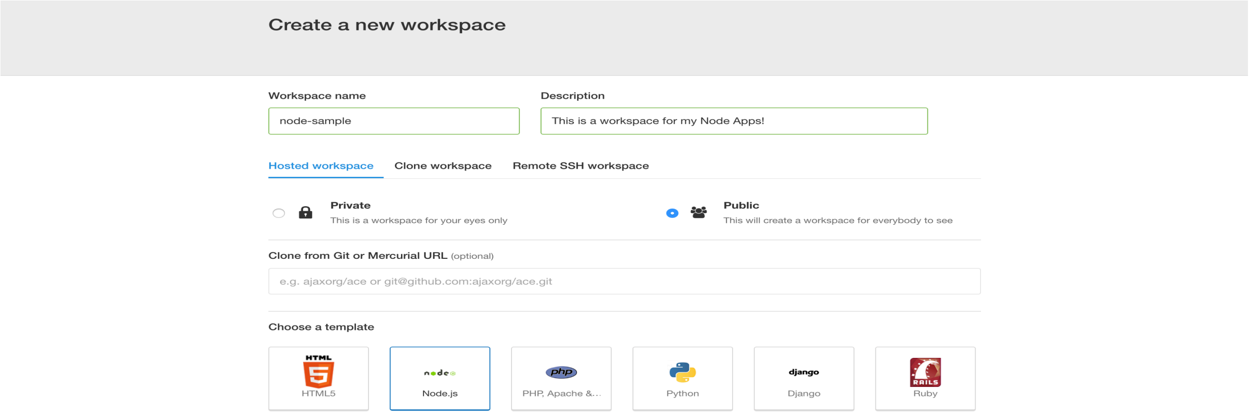
#2 | Create a new workspace

Create a new workspace in Cloud9 by clicking on the new workspace button(As Shown Below). When prompted fill in the following details,

- **Workspace Name** – Define what your workspace will contain
- **Description** – Describe the contents of your workspace
- Under Hosted Workspace, select the **public** option
- **Template** – Choose the Node.js template to get Node/NPM pre-installed

Click on **"Create Workspace"** to create your new workspace and initialize the terminal for you with Node.js and NPM pre-installed. You will be automatically taken to your new workspace.




Create a new workspace
Hosted workspace
Clone workspace
Remote SSH workspace
Private
This is a workspace for your eyes only
Public
This will create a workspace for everybody to see
Clone from Git or Mercurial URL (optional)
e.g. ajaxorg/ace or git@github.com:ajaxorg/ace.git
Choose a template
HTML5
Node.js
PHP, Apache &...
Python
Django
Ruby

Workspace name: node-sample

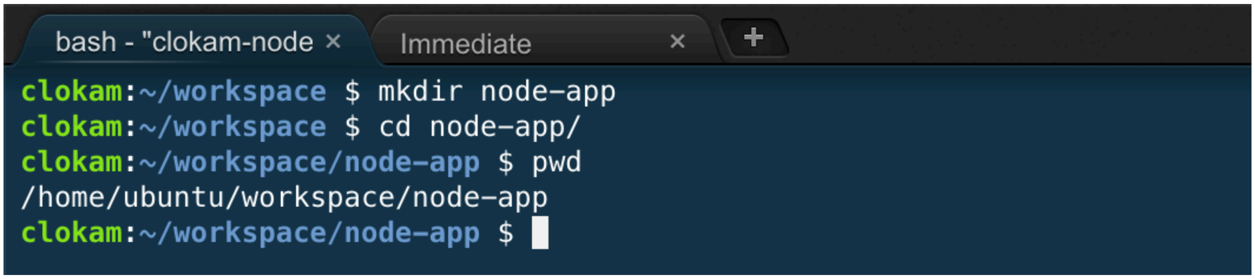
Description: This is a workspace for my Node Apps!

○ Private ● Public

Template: Node.js

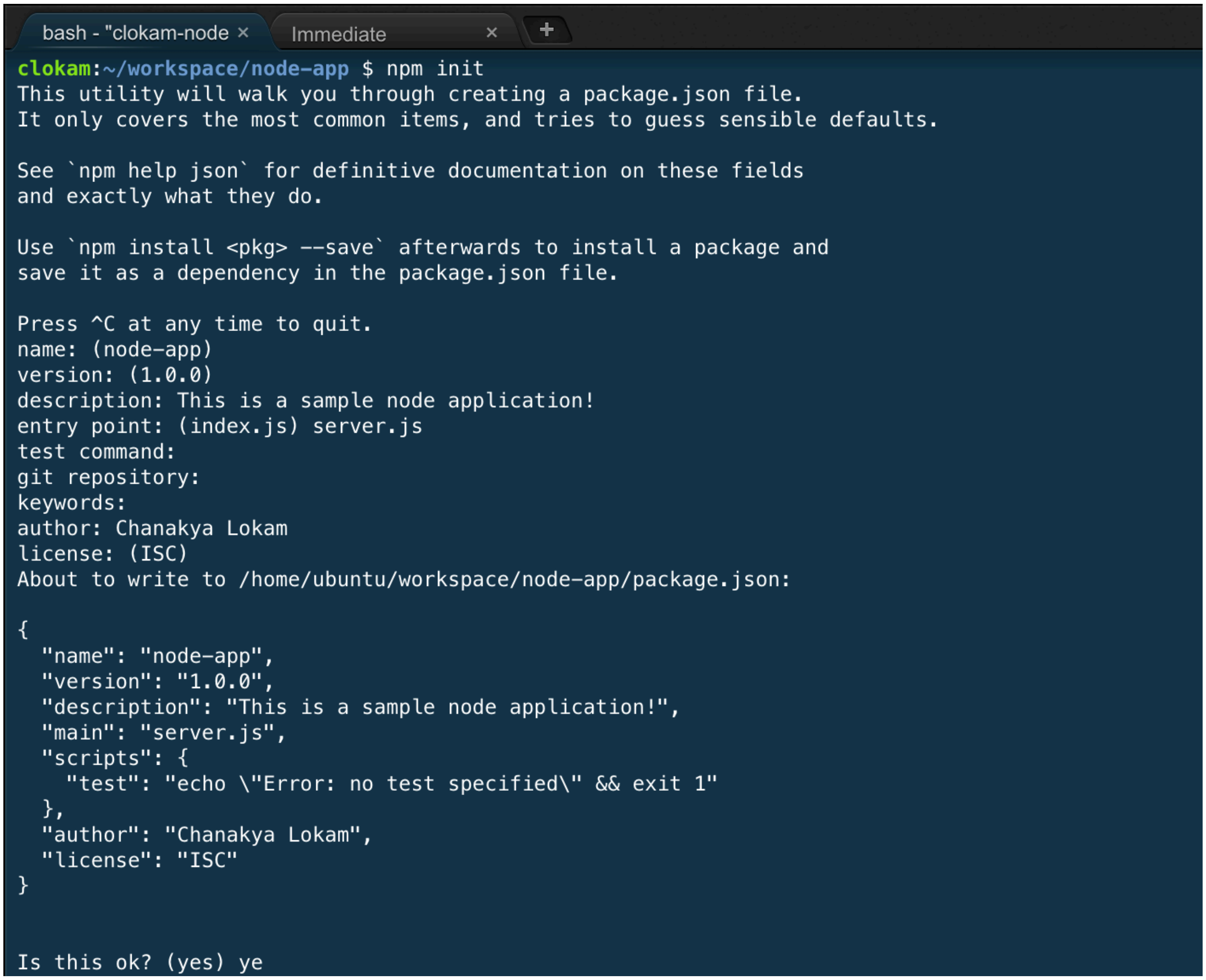
#3| Create a sample API Route using Express and Node

Create a new directory using the “**mkdir** <directory name>” command in the terminal provided at the bottom. Then navigate to that directory in current bash terminal using **cd** <directory name>.



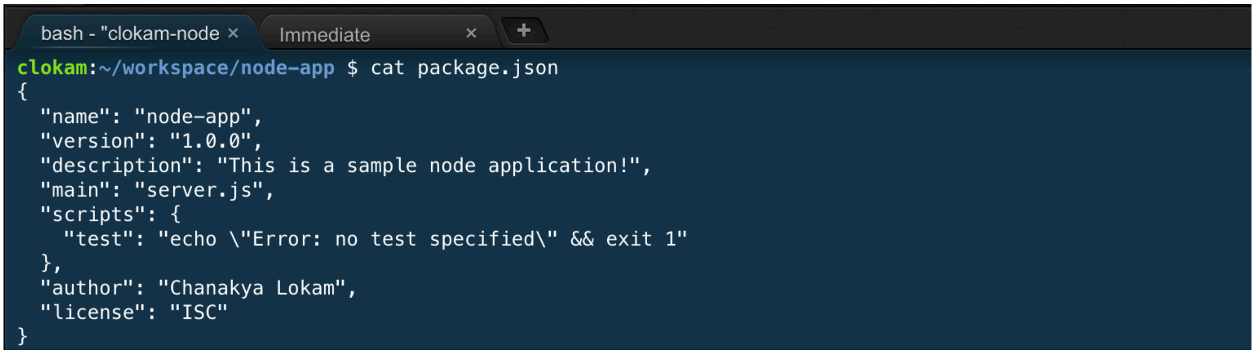
```
clokam:~/workspace $ mkdir node-app
clokam:~/workspace $ cd node-app/
clokam:~/workspace/node-app $ pwd
/home/ubuntu/workspace/node-app
clokam:~/workspace/node-app $ 
```

Then initialize your Node Project by creating the **package.json** file with the **npm init** command. When prompted enter more information about your application.



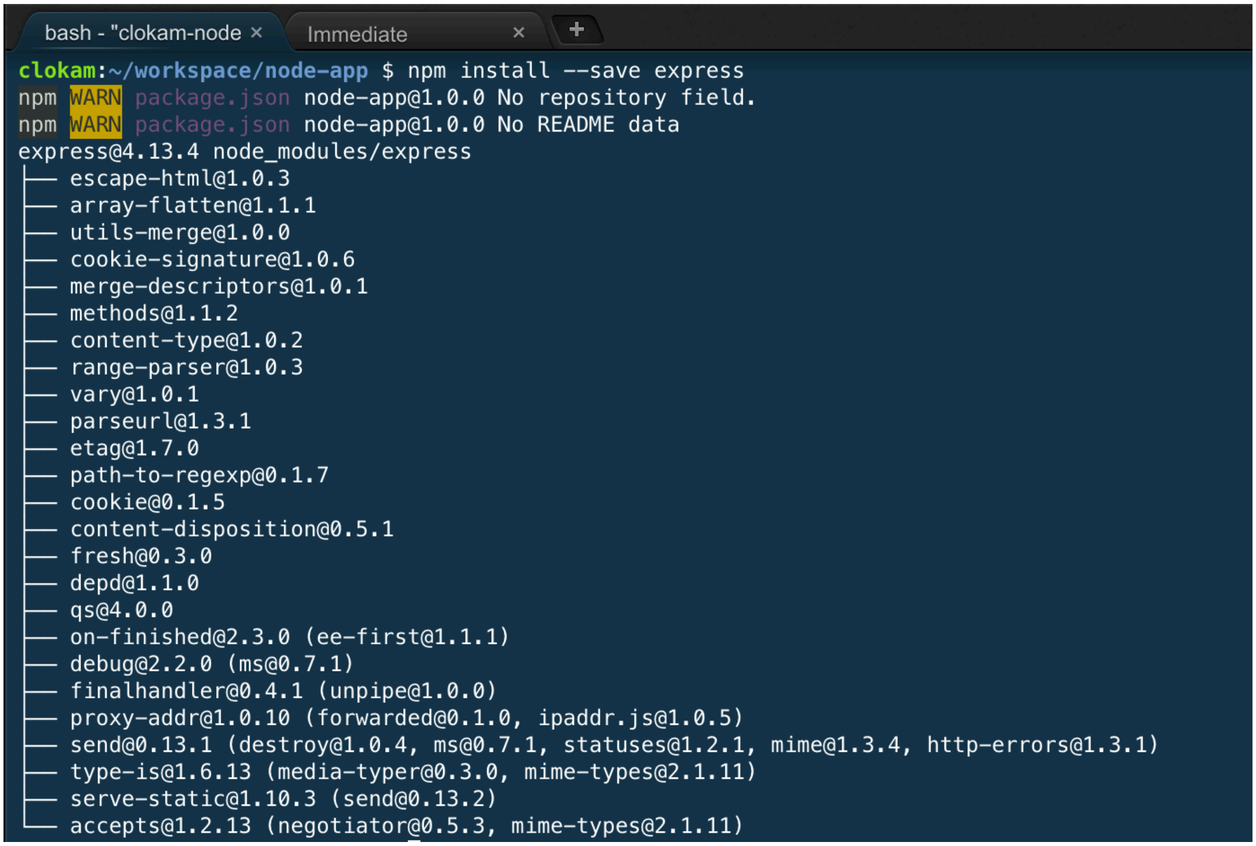
```
clokam:~/workspace/node-app $ npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See `npm help json` for definitive documentation on these fields
and exactly what they do.

Use `npm install <pkg> --save` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
name: (node-app)
version: (1.0.0)
description: This is a sample node application!
entry point: (index.js) server.js
test command:
git repository:
keywords:
author: Chanakya Lokam
license: (ISC)
About to write to /home/ubuntu/workspace/node-app/package.json:

{
  "name": "node-app",
  "version": "1.0.0",
  "description": "This is a sample node application!",
  "main": "server.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "Chanakya Lokam",
  "license": "ISC"
}


Is this ok? (yes) ye
```

Once created you can see the file with the **cat package.json** command.



```
clokam:~/workspace/node-app $ cat package.json
{
  "name": "node-app",
  "version": "1.0.0",
  "description": "This is a sample node application!",
  "main": "server.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "Chanakya Lokam",
  "license": "ISC"
}
```

We will be using Express in this example to route our application and serve static HTML pages. To install dependencies and also add them to the package.json file, use the npm install command with the --save option. You can install express with the **npm install --save express** command.



```
clokam:~/workspace/node-app $ npm install --save express
npm WARN package.json node-app@1.0.0 No repository field.
npm WARN package.json node-app@1.0.0 No README data
express@4.13.4 node_modules/express
├── escape-html@1.0.3
├── array-flatten@1.1.1
├── utils-merge@1.0.0
├── cookie-signature@1.0.6
├── merge-descriptors@1.0.1
├── methods@1.1.2
├── content-type@1.0.2
├── range-parser@1.0.3
├── vary@1.0.1
├── parseurl@1.3.1
├── etag@1.7.0
├── path-to-regexp@0.1.7
├── cookie@0.1.5
├── content-disposition@0.5.1
├── fresh@0.3.0
├── depd@1.1.0
├── qs@4.0.0
├── on-finished@2.3.0 (ee-first@1.1.1)
├── debug@2.2.0 (ms@0.7.1)
├── finalhandler@0.4.1 (unpipe@1.0.0)
├── proxy-addr@1.0.10 (forwarded@0.1.0, ipaddr.js@1.0.5)
├── send@0.13.1 (destroy@1.0.4, ms@0.7.1, statuses@1.2.1, mime@1.3.4, http-errors@1.3.1)
├── type-is@1.6.13 (media-typer@0.3.0, mime-types@2.1.11)
├── serve-static@1.10.3 (send@0.13.2)
└── accepts@1.2.13 (negotiator@0.5.3, mime-types@2.1.11)
```

You can now again see the **package.json** file and will notice that the express dependency has been added to the dependencies section.

#4 | Let's start coding!

Create the following files in your app directory, and add the given code. These files will form the base of your application.

- **index.html** : Your web page entry point
- **server.js** : Your Node.js Server File that will use Express to serve the index.html page

index.html

```
<!DOCTYPE html>
<html>
<meta http-equiv="Content-Type" Content="text/html;charset=utf-8"/>
<meta content="utf-8" http-equiv="encoding">
<head>
<body style="text-align:center">
<div style="background-color:#2368a0; color:white;  padding:10px; width:40%;
margin-left:30%;margin-top:10%; ">
<h1><center>MIRACLE SOFTWARE SYSTEMS</center></h1>
</div>
<div style="background-color:#00aae7 ; color:white;  padding:10px; width:40%;
margin-left:30%; text-align:center;  ">
<p align="justify">Founded in 1994, Miracle Software Systems, Inc. is a private
minority firm headquartered in Novi, MI(USA).For the past 2 decades teams at
Miracle have helped numerous customers rapidly transition their IT to a Service
Oriented Architecture. With over 1500 employees at Global Development Centers
in 8 countries, Miracle has been able to carve a mark into niche IT Services.</p>
</div>
</body>
</head>
</html>
```

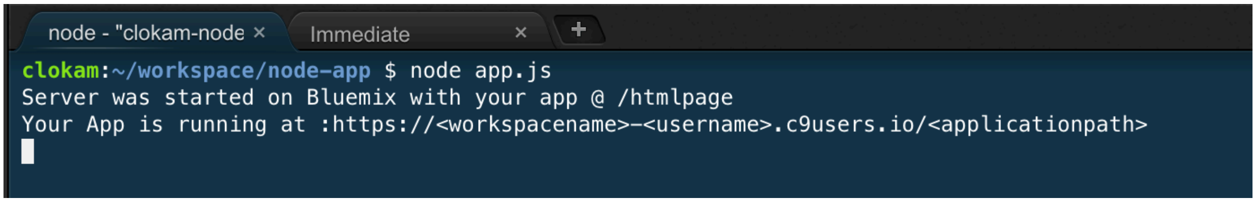
----

server.js

```
var express = require('express');
var app = express();
app.get("/htmlpage", function(req,res){
res.sendFile(__dirname + "/" + "index.html");
});

app.listen(process.env.PORT, process.env.IP, function(){
 console.log("Server was started on Bluemix with your app @ /htmlpage");
 console.log("You  App is running at :
https://<workspacename><username>.c9users.io/<applicationpath>");
});
-----------------------------------------------------------------------------------------------
```

#5 | Testing the application locally on C9

You can run your application from the terminal with the **node server.js** command.



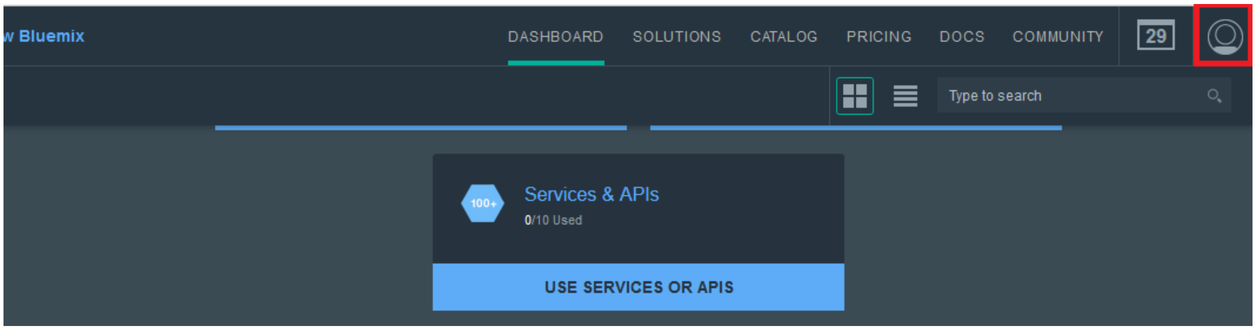
```
clokam:~/workspace/node-app $ node app.js
Server was started on Bluemix with your app @ /htmlpage
Your App is running at :https://<workspacename>-<username>.c9users.io/<applicationpath>
```

Specify your workspace name, user name and application path in the below URL format and see your application in your browser.

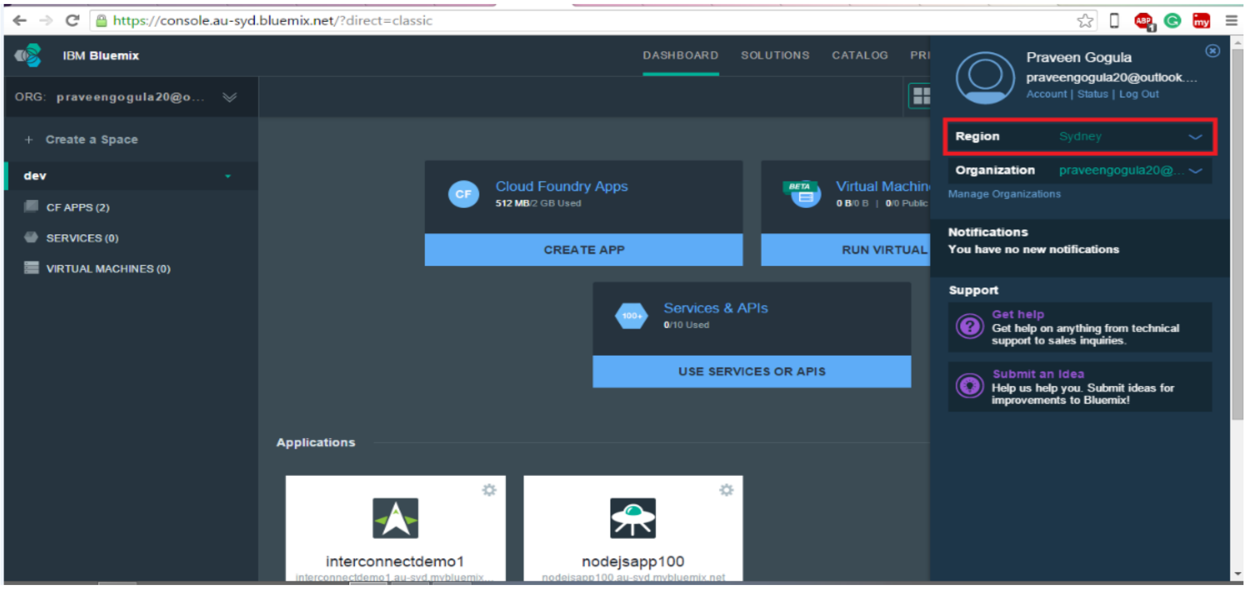
https://<workspacename>-<username>.c9users.io/<applicationpath>

#6 | Access Bluemix

The next step is to make sure that you have access for IBM Bluemix Console with either the free trial option (or) the paid subscription option. Login to Bluemix at **http://bluemix.net** (or) Create a new account today! After logging in you can see your dashboard with your services and applications.



You can also access your profile and check your region as shown below,



https://console.au-syd.bluemix.net/?direct=classic

Region Sydney

Organization praveengogula20@...

Notifications
You have no new notifications

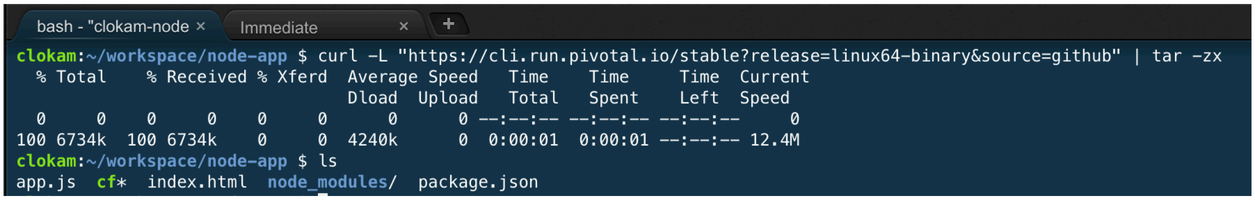
Get help
Get help on anything from technical support to sales inquiries.

Submit an Idea
Help us help you. Submit ideas for improvements to Bluemix!

#7 | Install CloudFoundry CLI in Cloud9

Download the **CloudFoundry CLI** in C9 terminal using **CURL**,

curl -L "https://cli.run.pivotal.io/stable?release=linux64-binary&source=github" | tar -zx

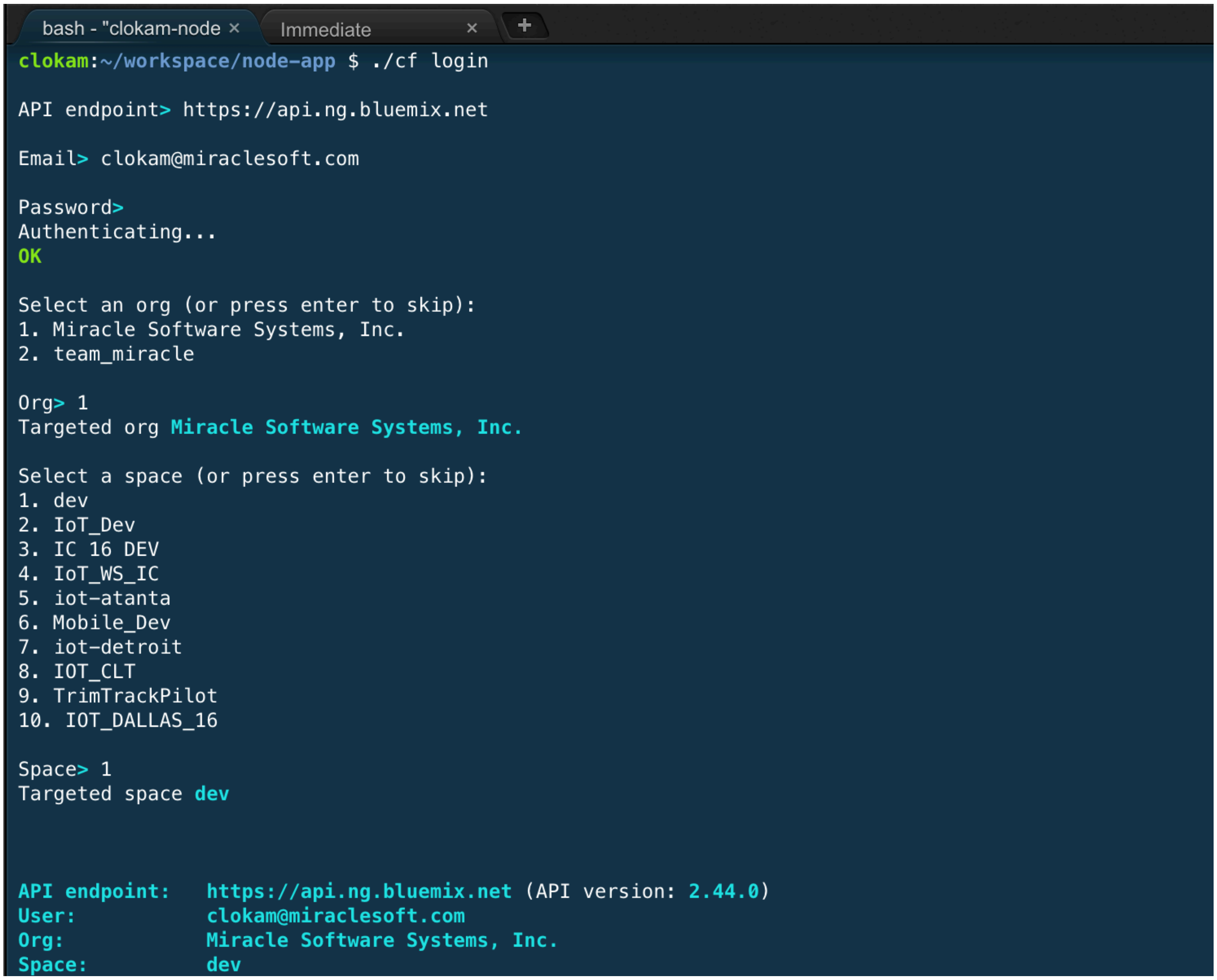


```
clokam:~/workspace/node-app $ curl -L "https://cli.run.pivotal.io/stable?release=linux64-binary&source=github" | tar -zx
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
100 6734k  100 6734k    0     0  4240k      0  0:00:01  0:00:01 --:--:-- 12.4M
clokam:~/workspace/node-app $ ls
app.js  cf*  index.html  node_modules/  package.json
```

Login to Bluemix from C9 using "**./cf login** ", and when prompted enter the API endpoint along with Bluemix credentials.

- **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

©2016 Miracle Software Systems, Inc.



```
clokam:~/workspace/node-app $ ./cf login

API endpoint> https://api.ng.bluemix.net

Email> clokam@miraclesoft.com

Password>
Authenticating...
OK

Select an org (or press enter to skip):
1. Miracle Software Systems, Inc.
2. team_miracle

Org> 1
Targeted org Miracle Software Systems, Inc.

Select a space (or press enter to skip):
1. dev
2. IoT_Dev
3. IC 16 DEV
4. IoT_WS_IC
5. iot-atanta
6. Mobile_Dev
7. iot-detroit
8. IOT_CLT
9. TrimTrackPilot
10. IOT_DALLAS_16

Space> 1
Targeted space dev



API endpoint:   https://api.ng.bluemix.net (API version: 2.44.0)
User:           clokam@miraclesoft.com
Org:            Miracle Software Systems, Inc.
Space:          dev
```

#8 | Create manifest.yml file and Push Application

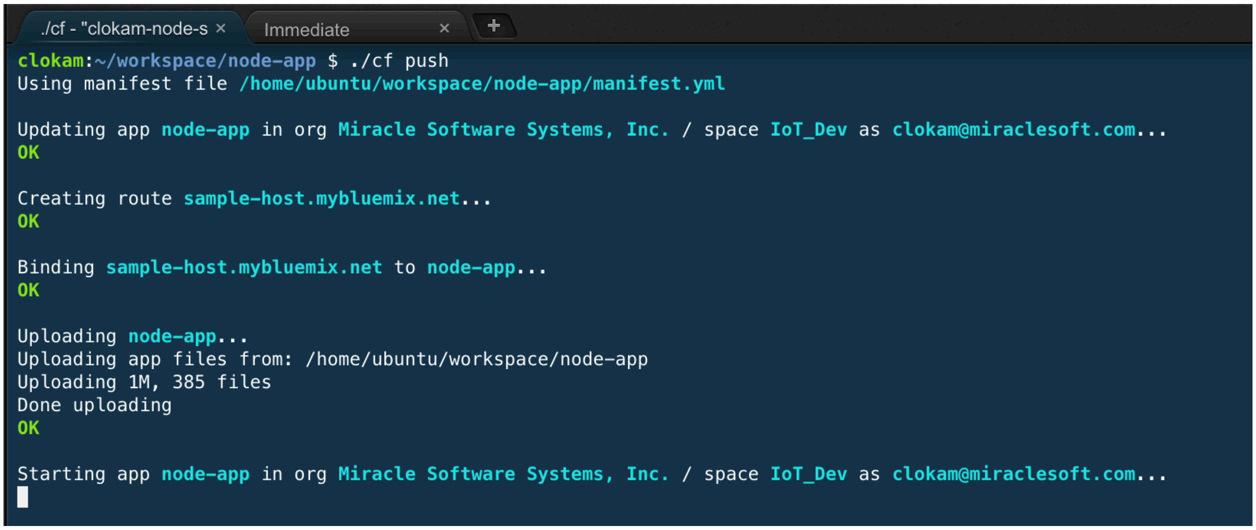
You will next need to create a manifest file for Bluemix to recognize your application and deploy it to the Node Runtime. Bluemix is based on Cloud Foundry, and within CloudFoundry the Node Buildpack will be able to detect, compile and run the Node.js Application.

manifest.yml

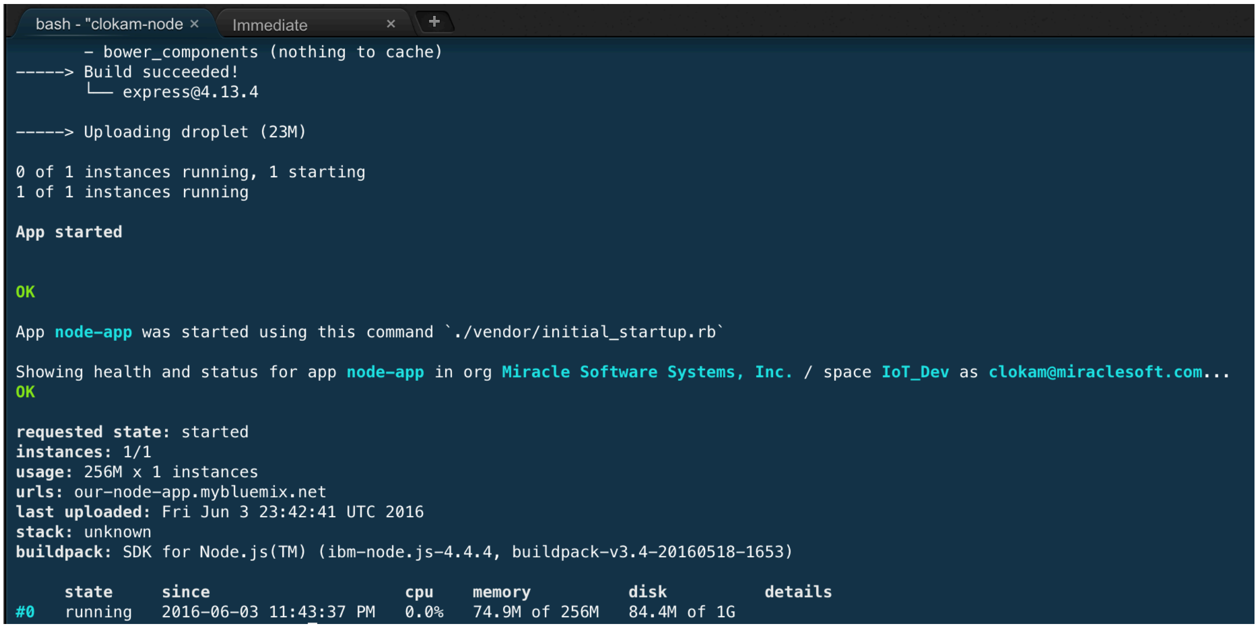
```
----------------------------------------------------------------------------------------------------------
applications:
- path: .
  memory: 256M
```

instances: 1
domain: mybluemix.net
name: node-app
host: sample-host
disk_quota: 1024M

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

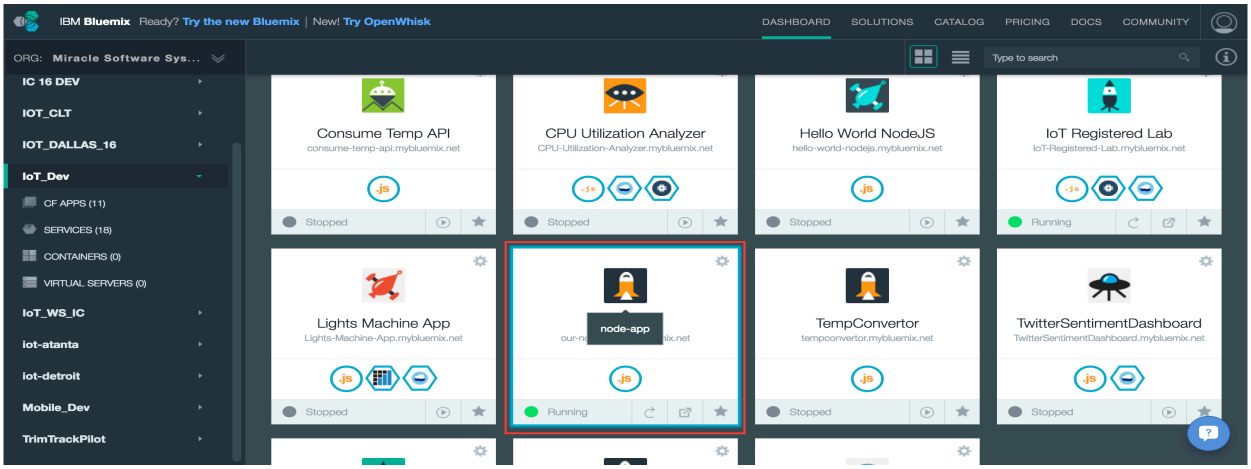


```
clokam:~/workspace/node-app $ ./cf push
Using manifest file /home/ubuntu/workspace/node-app/manifest.yml

Updating app node-app in org Miracle Software Systems, Inc. / space IoT_Dev as clokam@miraclesoft.com...
OK

Creating route sample-host.mybluemix.net...
OK

Binding sample-host.mybluemix.net to node-app...
OK

Uploading node-app...
Uploading app files from: /home/ubuntu/workspace/node-app
Uploading 1M, 385 files
Done uploading
OK

Starting app node-app in org Miracle Software Systems, Inc. / space IoT_Dev as clokam@miraclesoft.com...
```

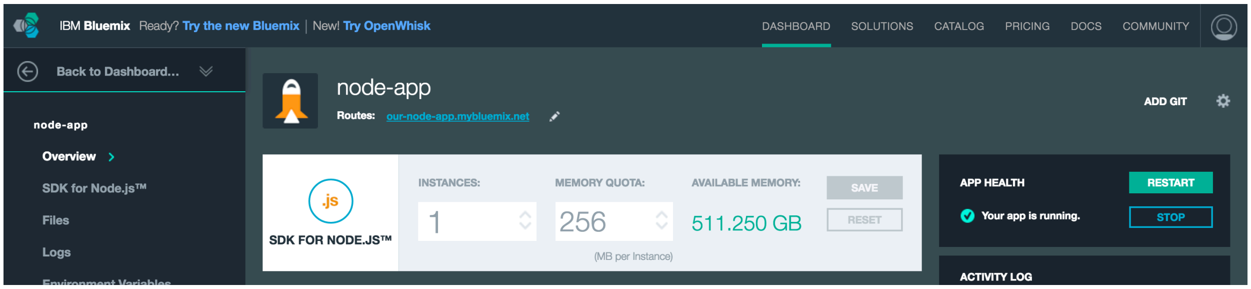


```
       - bower_components (nothing to cache)
-----> Build succeeded!
       └── express@4.13.4

-----> Uploading droplet (23M)

0 of 1 instances running, 1 starting
1 of 1 instances running

App started


OK

App node-app was started using this command `./vendor/initial_startup.rb`

Showing health and status for app node-app in org Miracle Software Systems, Inc. / space IoT_Dev as clokam@miraclesoft.com...
OK

requested state: started
instances: 1/1
usage: 256M x 1 instances
urls: our-node-app.mybluemix.net
last uploaded: Fri Jun 3 23:42:41 UTC 2016
stack: unknown
buildpack: SDK for Node.js(TM) (ibm-node.js-4.4.4, buildpack-v3.4-20160518-1653)

     state     since                    cpu    memory         disk          details
#0   running   2016-06-03 11:43:37 PM   0.0%   74.9M of 256M   84.4M of 1G
```

Go back to your Bluemix account in the browser and find your node application.


IBM Bluemix Ready? Try the new Bluemix | New! Try OpenWhisk
DASHBOARD SOLUTIONS CATALOG PRICING DOCS COMMUNITY
ORG: Miracle Software Sys...
Type to search
IC 16 DEV
IOT_CLT
IOT_DALLAS_16
IoT_Dev
CF APPS (11)
SERVICES (18)
CONTAINERS (0)
VIRTUAL SERVERS (0)
IoT_WS_IC
iot-atanta
iot-detroit
Mobile_Dev
TrimTrackPilot
Consume Temp API
consume-temp-api.mybluemix.net
Stopped
CPU Utilization Analyzer
CPU-Utilization-Analyzer.mybluemix.net
Stopped
Hello World NodeJS
hello-world-nodejs.mybluemix.net
Stopped
IoT Registered Lab
IoT-Registered-Lab.mybluemix.net
Running
Lights Machine App
Lights-Machine-App.mybluemix.net
Stopped
node-app
Running
TempConvertor
tempconvertor.mybluemix.net
Stopped
TwitterSentimentDashboard
TwitterSentimentDashboard.mybluemix.net
Stopped

Click on your app and select your **app route** in the overview page.



node-app

Routes: our-node-app.mybluemix.net

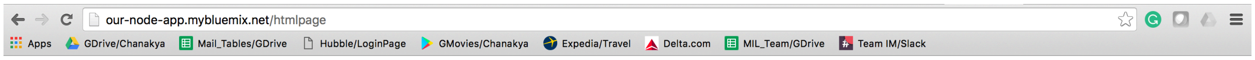
INSTANCES: 1

MEMORY QUOTA: 256

AVAILABLE MEMORY: 511.250 GB

Your app is running.

Add **/htmlpage** to your Application Route to view your completed and deployed application. You mentioned /htmlpage as your express route in your server.js file and that is why you have to go to that route to access your application.



our-node-app.mybluemix.net/htmlpage

MIRACLE SOFTWARE SYSTEMS

Founded in 1994, Miracle Software Systems, Inc. is a private minority firm headquartered in Novi,MI(USA).For the past 2 decades teams at Miracle have helped numerous customers rapidly transition their IT to a Service Oriented Architecture.With over 1500 employees at Global Development Centers in 8 countries, Miracle has been able to carve a mark into niche IT Services.