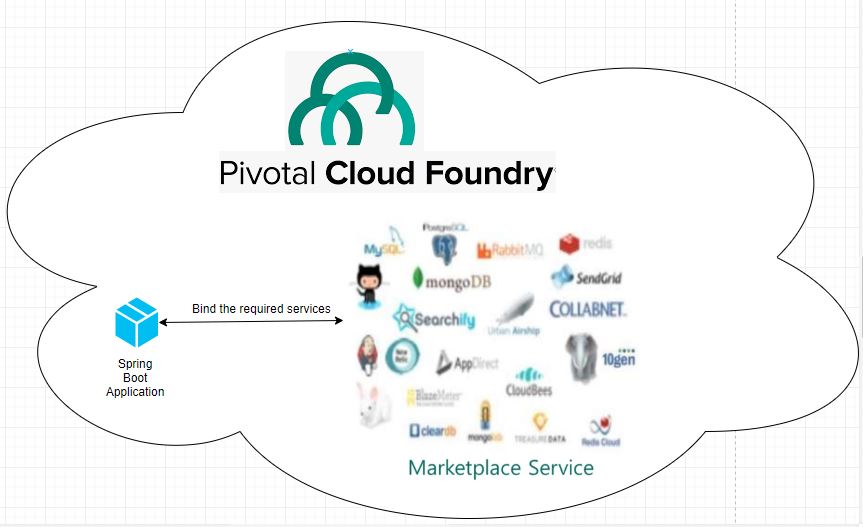
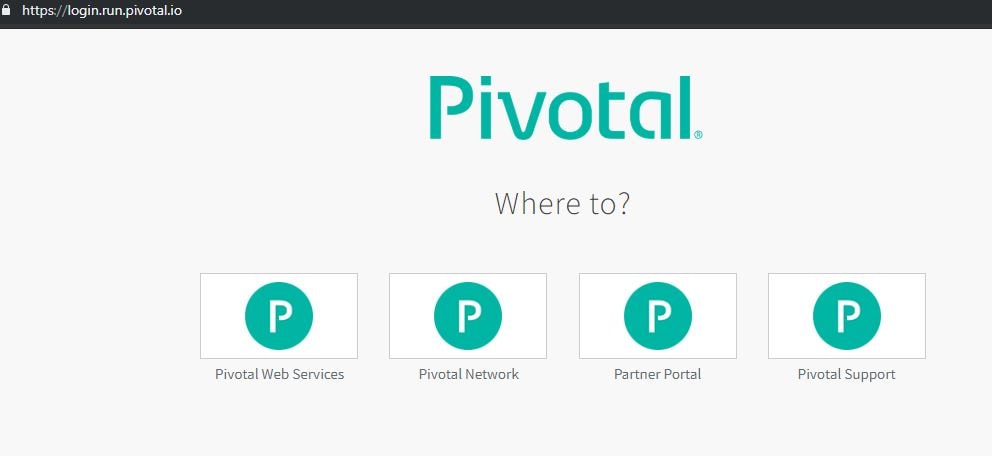
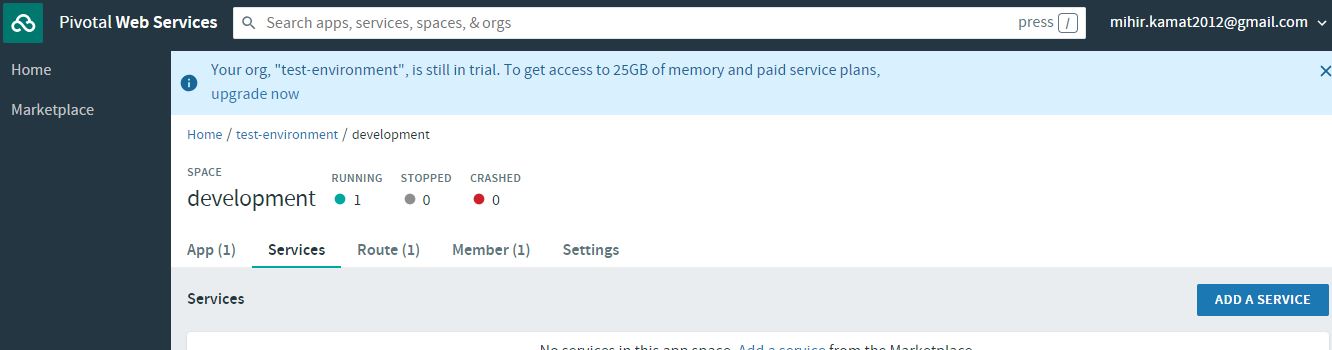
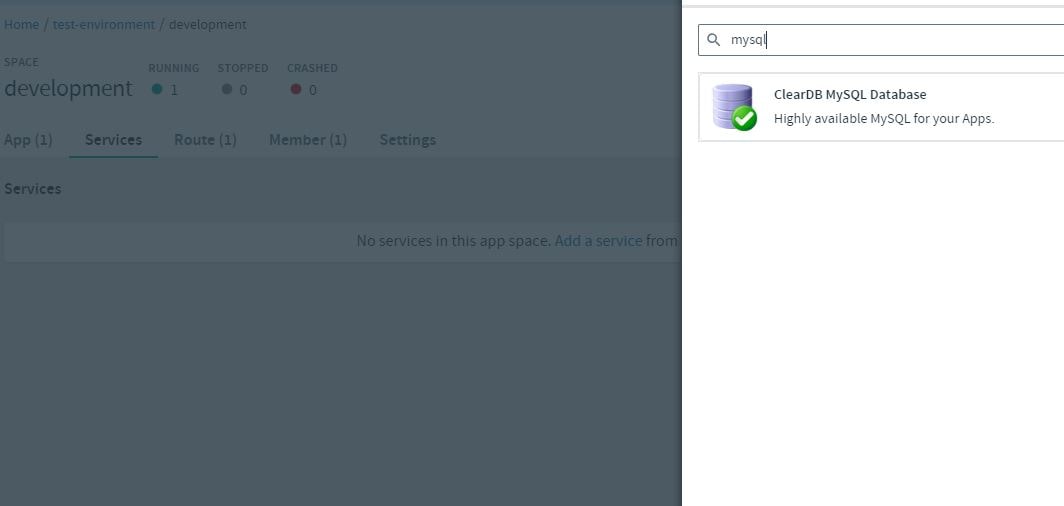
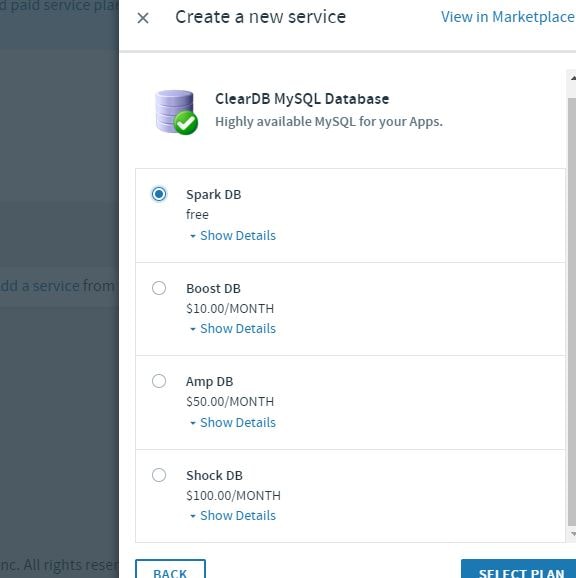
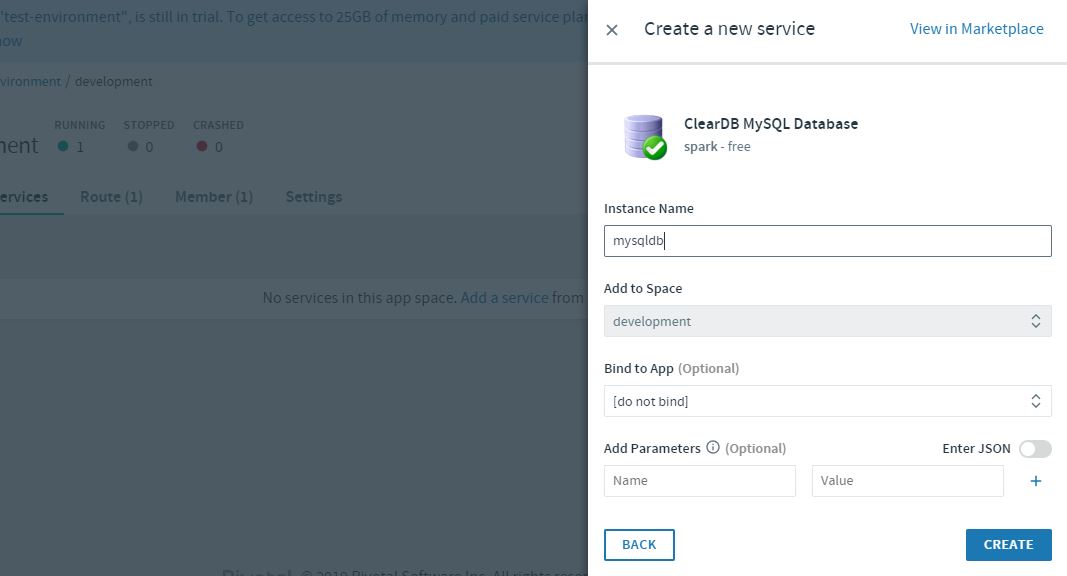
## In previous tutorial we had deployed a [**Spring Boot Application to PCF**](https://www.javainuse.com/pcf/pcf-hello). In this tutorial we will develop a Spring Boot + MySQL Application and deploy it to PCF. We will be creating a Spring Boot + MySQL Application using JDBC and testing it locally. We have already seen [**Spring Boot MYSQL JDBC basics in a previous tutorial.**](https://www.javainuse.com/spring/bootjdbc) For deploying on PCF, we dont need to install MySQL on the PCF cloud. On PCF there is a concept of services where on demand resources like MySQL and RabbitMQ can ordered and directly bound to the application. We will see how this is done. Spring Boot SQL PCF Push PCF Deployment-

In Cloud Foundry, services are on demand resources that users can provision and use for their deployed application.  
Examples of resources services provide include databases on a shared or dedicated server, or accounts on a SaaS application. These resources are known as service instances and the systems that deliver and operate these resources are known as Services. Think of a service as a factory that delivers service instances.  
Services can either be

* User Defined Service
* MarketPlace Service

For this example we will be needing the MySQL MarketPlace service.  
  


Provisioning MySql Service using PCF Web Console

* Login to Pivotal Cloud Foundry with your credentials  
  
* Go to Services tab for our development space  
    
  
* Click on Add New Service and type MySQL.  
    
  
* Select ClearDB MySql  
  
* Select the free plan free Spark DB.  
  

Creating Manifest Configuration file to Bind our Maven Project to PCF SQL Service

[Previously when deploying the application to PCF using the cf push command](https://www.javainuse.com/pcf/pcf-hello), we had given the required configuration like location if the jar file as command line arguments.

cf push test-environment -p target\employee-producer-0.0.1-SNAPSHOT.jar

There are other configurations like memory, application name,PCF services to be bound which can be specified as command line arguments.  
  
A much better way to specify this is as a configuration file named manifest.yml. Manifest is the blueprint of the application to be deployed in PCF. Manifests provide consistency and reproducibility, and can help us automate deploying apps.  
  
In our eclipse jdbc project create a manifest configuration file as follows-

name: SpringBootMySql

path: target\boot-jdbc-0.0.1-SNAPSHOT.jar

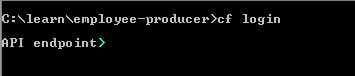
memory: 1G

services:

-mysqldb

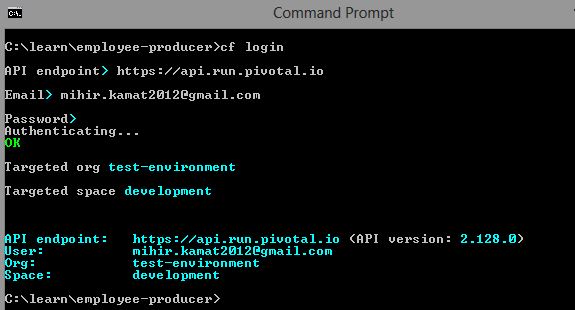
Push the application to PCF

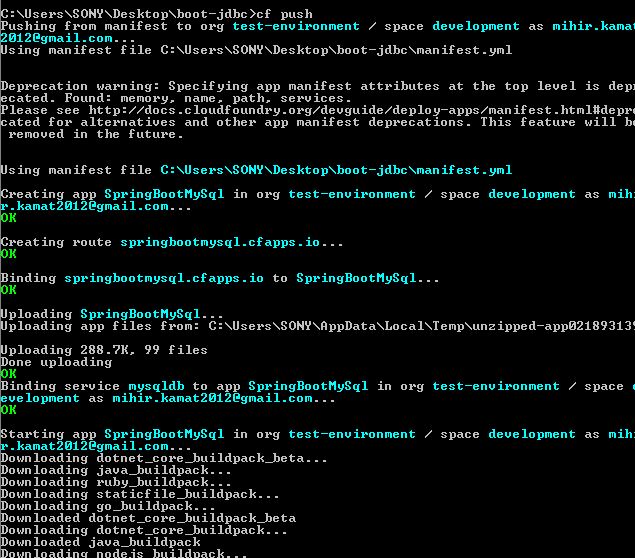
* Open the command terminal and use the following command
* cf login

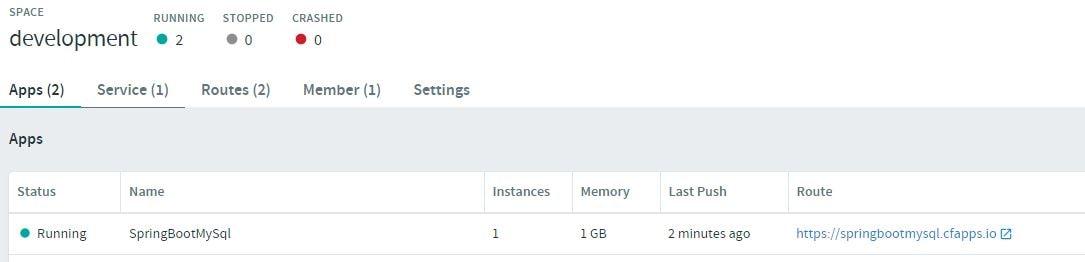


* It will ask for Cloud Foundry API. Enter The following API value-
* https://api.run.pivotal.io

Cloud Foundry Login API

* Next it will ask you for the Cloud Foundry credentials  
  
* Use the command cf push-
* cf push



The application is deployed on PCF. Go the PCF Web Console. Our new application will be up and running.  
  
  
Select the application and go to the specified route. Append /employees to the route url. Our application is up and running.  
  
