Cloud Developers Certification Training

Section 2.5 - Using CLI to Deploy and Manage Applications

Lab Exercise

Version:	1
Last modification date:	18-Mar-16
Owner:	IBM Ecosystem Development

Exercise 2.5.0 – Lab Prerequisites

- Have a IBM Bluemix account
    - Sign up for Bluemix http://bluemix.net
- A Web Browser supported by Bluemix
    - Chrome, latest version for your OS
    - Firefox, latest version for your OS and ESR 31 or ESR 38
    - Internet Explorer, version 10 and 11
    - Safari, latest version for Mac
- Cloud Foundry command line interface, Version 6.5.1 or later (Recommend using latest release)
- Have Git Bash installed (Recommended)
    - Download and Install Git Bash: https://git-scm.com/downloads

Exercise 2.5.1 – Install Cloud Foundry Command Line Interface

Skip this part if you have Cloud Foundry V6.5.1+ installed

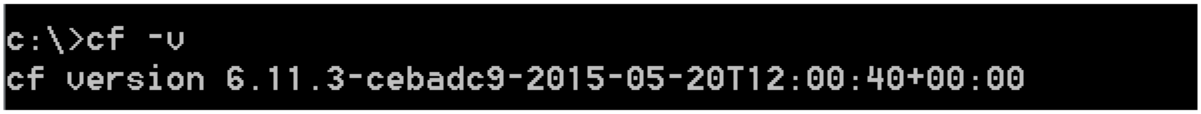
- CF-CLI download: https://github.com/cloudfoundry/cli/releases
- Windows Installation
    - Unpack the zip file.
    - Double click the cf executable.
    - When prompted, click Install, and then Close.
- MAC OSX and Linux Installation
    - Open the .pkg file.
    - In the installer wizard, click Continue.
    - Select an install destination and click Continue.
    - When prompted, click Install.

Exercise 2.5.2 – Understand Your Bluemix Account

In this exercise you will use the Command Line Interface tool to work with Bluemix and understand your Bluemix ID. You use this tool in a terminal or command window on your workstation.

- Start a command windows or Git bash shell window. The Cloud Foundry command line interface is not supported by Cygwin, avoid using Cygwin command line window.

- To verify your CLI release: ***cf –v***

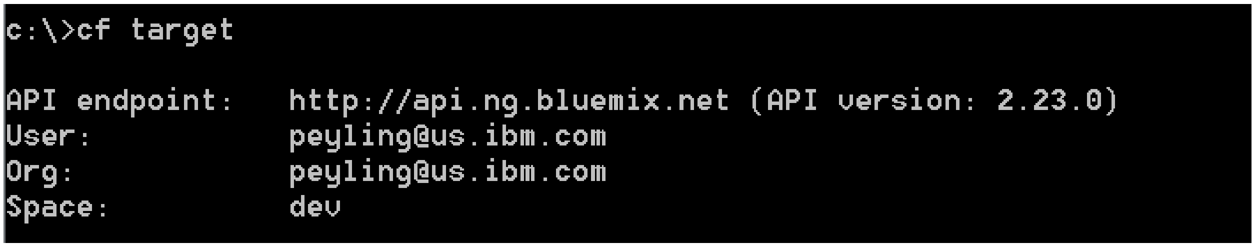


```
c:\>cf -v
cf version 6.11.3-cebadc9-2015-05-20T12:00:40+00:00
```

- To connect to Bluemix*:* ***cf api*** ***http://api.ng.bluemix.net*** (for US South region) or ***cf api*** **http://api.eu-gb.bluemix.net** (*for United Kingdom region)*

- To log in to Bluemix: ***cf login -u user_name -o org_name -s space_name*** or **cf login** then enter your Bluemix credentials (email & password) that you use to sign in to the Bluemix Web UI. Select the organization and space you want to work in if prompted.

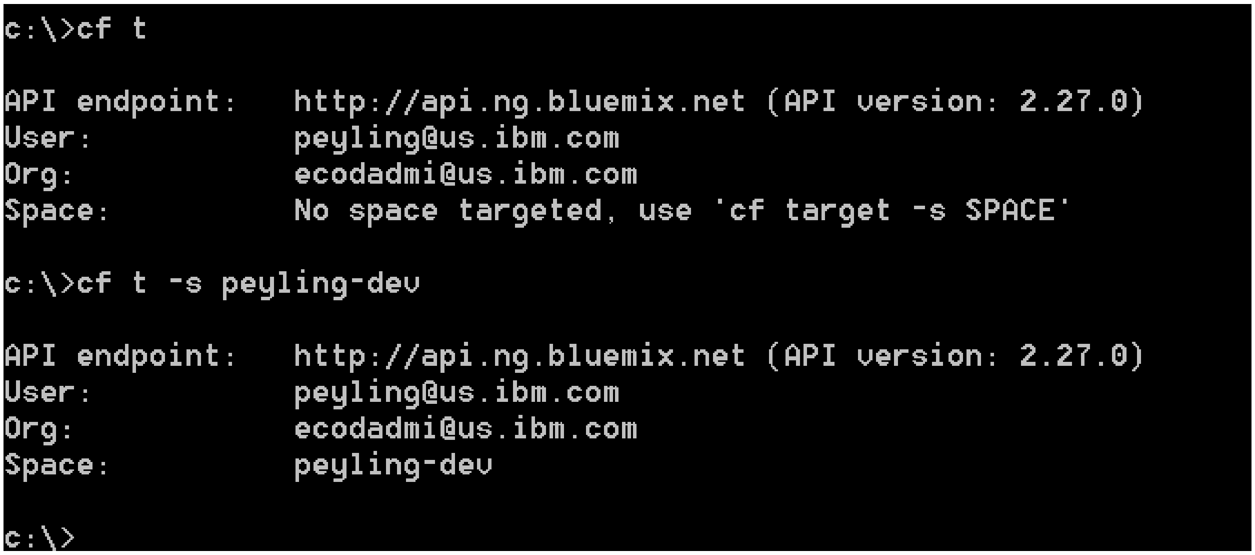
- To view your default organization, domain and space: ***cf target*** or ***cf t***



```
c:\>cf target

API endpoint:   http://api.ng.bluemix.net (API version: 2.23.0)
User:           peyling@us.ibm.com
Org:            peyling@us.ibm.com
Space:          dev
```

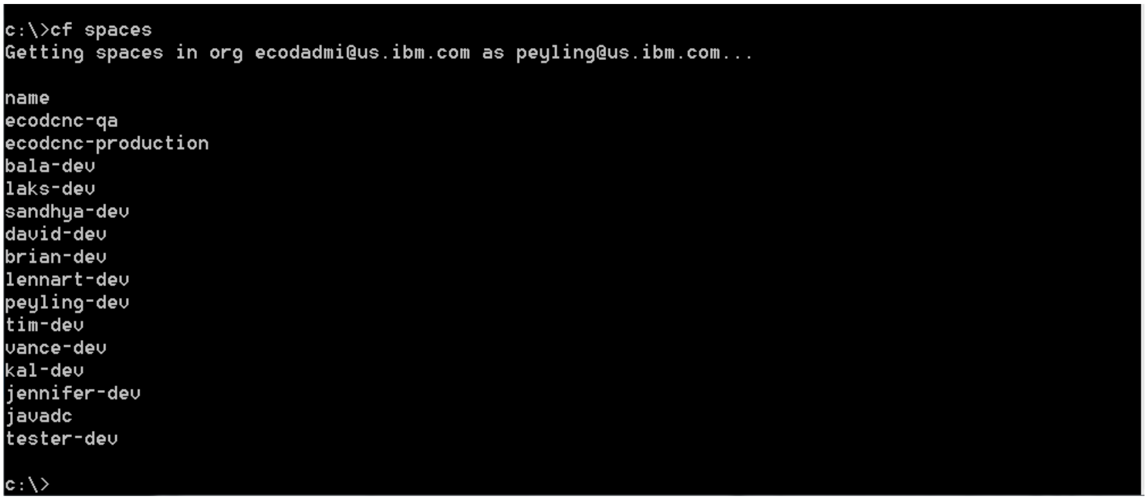
You will need information from this command result when you connect and log into Bluemix. In Bluemix, users must associate with an organization and have a space to work on their applications and services. When you create a Bluemix account, Bluemix creates a default organization for you, most likely is the same as your Bluemix ID and you are the owner of the organization. Your ID could be a member of multiple organizations and have access to or the owner of many spaces.

You can set space if no space targeted or switch to another targeted Space for project in the organization by: ***cf target –s SpaceName***



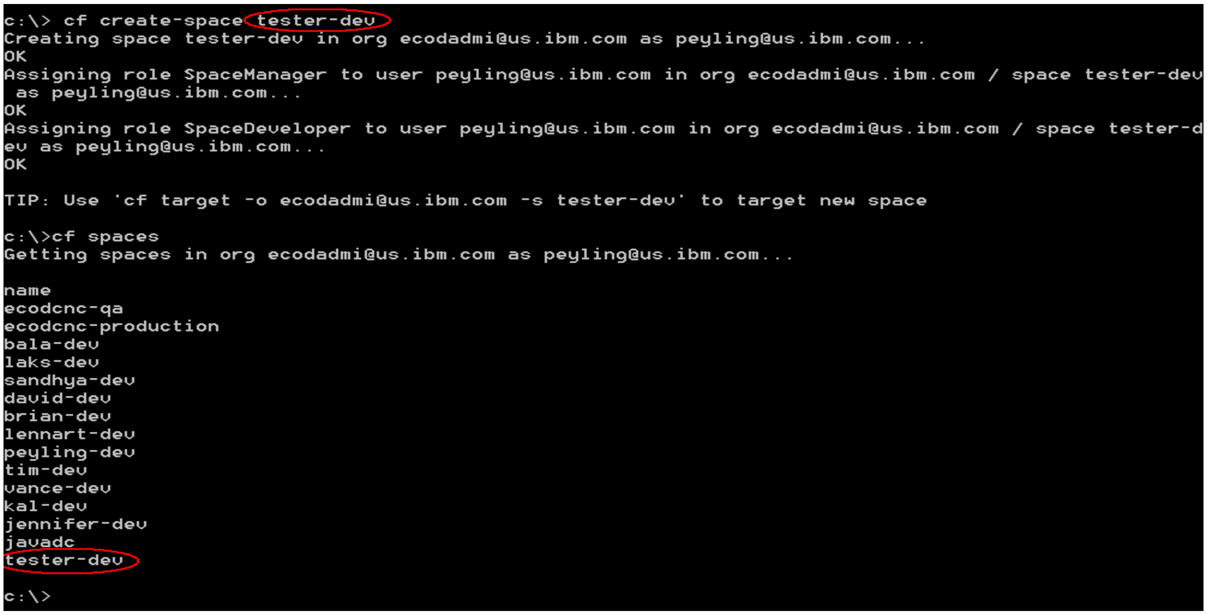
```
c:\>cf t

API endpoint:   http://api.ng.bluemix.net (API version: 2.27.0)
User:           peyling@us.ibm.com
Org:            ecodadmi@us.ibm.com
Space:          No space targeted, use 'cf target -s SPACE'

c:\>cf t -s peyling-dev

API endpoint:   http://api.ng.bluemix.net (API version: 2.27.0)
User:           peyling@us.ibm.com
Org:            ecodadmi@us.ibm.com
Space:          peyling-dev

c:\>
```

- To get a list of existing space in current organization: ***cf spaces***



```
c:\>cf spaces
Getting spaces in org ecodadmi@us.ibm.com as peyling@us.ibm.com...

name
ecodcnc-qa
ecodcnc-production
bala-dev
laks-dev
sandhya-dev
david-dev
brian-dev
lennart-dev
peyling-dev
tim-dev
vance-dev
kal-dev
jennifer-dev
javadc
tester-dev

c:\>
```

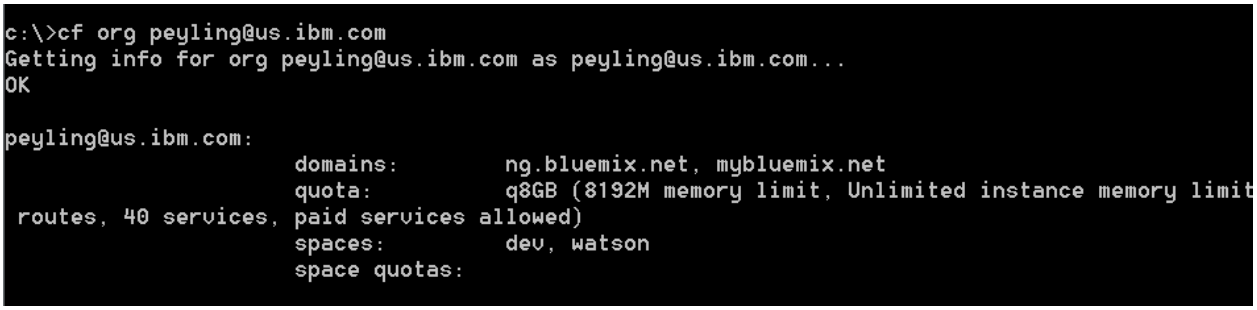
- To create a Space for a new project: **cf create-space *SpaceName***



```
c:\> cf create-space tester-dev
Creating space tester-dev in org ecodadmi@us.ibm.com as peyling@us.ibm.com...
OK
Assigning role SpaceManager to user peyling@us.ibm.com in org ecodadmi@us.ibm.com / space tester-dev
 as peyling@us.ibm.com...
OK
Assigning role SpaceDeveloper to user peyling@us.ibm.com in org ecodadmi@us.ibm.com / space tester-d
ev as peyling@us.ibm.com...
OK

TIP: Use 'cf target -o ecodadmi@us.ibm.com -s tester-dev' to target new space

c:\>cf spaces
Getting spaces in org ecodadmi@us.ibm.com as peyling@us.ibm.com...

name
ecodcnc-qa
ecodcnc-production
bala-dev
laks-dev
sandhya-dev
david-dev
brian-dev
lennart-dev
peyling-dev
tim-dev
vance-dev
kal-dev
jennifer-dev
javadc
tester-dev

c:\>
```

- To delete a space if your no longer need it: ***cf delete-space SpaceName***

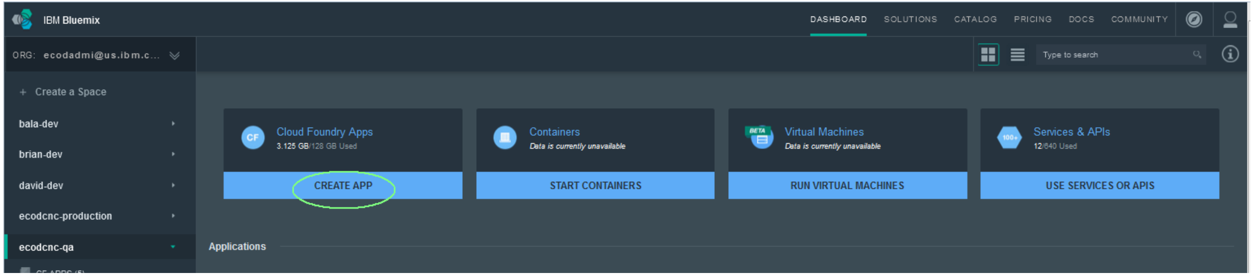
  - To view resource quota available to you: ***cf org yourORGname***



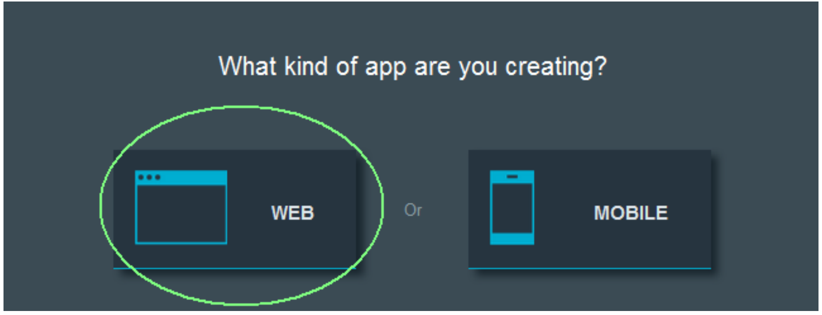
```
c:\>cf org peyling@us.ibm.com
Getting info for org peyling@us.ibm.com as peyling@us.ibm.com...
OK

peyling@us.ibm.com:
                      domains:        ng.bluemix.net, mybluemix.net
                      quota:          q8GB (8192M memory limit, Unlimited instance memory limit
 routes, 40 services, paid services allowed)
                      spaces:         dev, watson
                      space quotas:
```

Exercise 2.5.3 – Deploy your first application using Bluemix Web User Interface

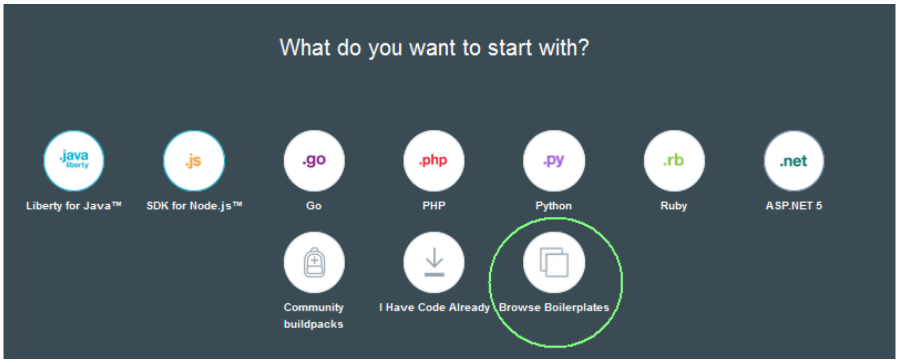
1. Open a browser navigate to https://bluemix.net and login to Bluemix using your Bluemix credentials. Once you log in, it should take you to your ‘default’ region and organization.
2. Create a web application by clicking on “CREATE APP” under Cloud Foundry Apps



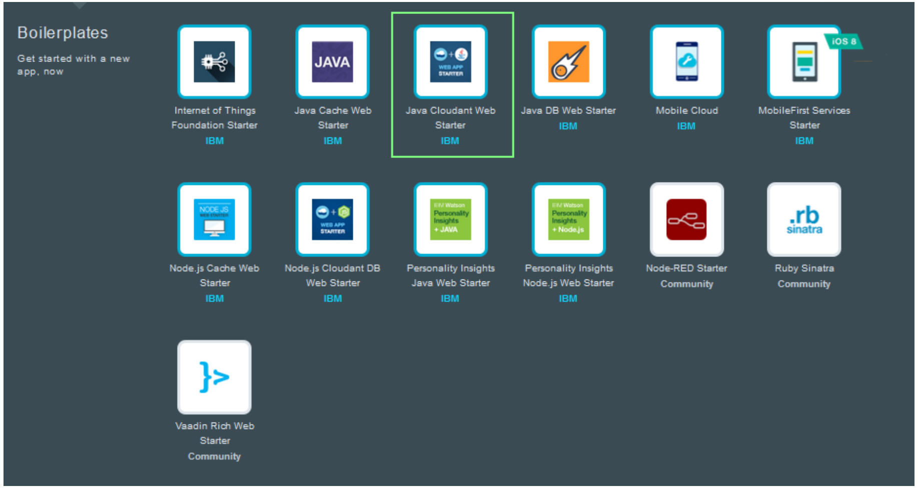
3. Select “Web” as your application template



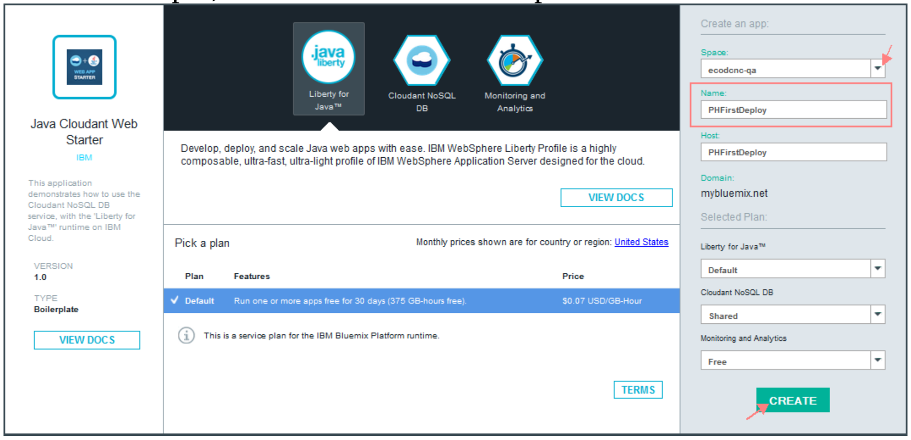
4. Choose a Boilerplate as starter by clicking on “Browse Boilerplate”



5. Select “Java Cloudant Web Starter” from Boilerplate catalog


Boilerplates
Get started with a new app, now
Internet of Things Foundation Starter
IBM
Java Cache Web Starter
IBM
Java Cloudant Web Starter
IBM
Java DB Web Starter
IBM
Mobile Cloud
IBM
MobileFirst Services Starter
IBM
iOS 8
Node.js Cache Web Starter
IBM
Node.js Cloudant DB Web Starter
IBM
Personality Insights Java Web Starter
IBM
Personality Insights Node.js Web Starter
IBM
Node-RED Starter
Community
Ruby Sinatra
Community
Vaadin Rich Web Starter
Community

6. Enter a name for your application. The host and application name will be the same and completed automatically. Then click on “Create”. The host name must be unique on Bluemix, so please choose a name with your company name or initials to try to make a unique name. You have option to pick **space** for this application from drop down list. In this example, we use current default space.



Java Cloudant Web Starter
IBM

This application demonstrates how to use the Cloudant NoSQL DB service, with the 'Liberty for Java™' runtime on IBM Cloud.

VERSION
1.0

TYPE
Boilerplate

Develop, deploy, and scale Java web apps with ease. IBM WebSphere Liberty Profile is a highly composable, ultra-fast, ultra-light profile of IBM WebSphere Application Server designed for the cloud.

Pick a plan

Monthly prices shown are for country or region: United States

Plan	Features	Price
✓ Default	Run one or more apps free for 30 days (375 GB-hours free).	$0.07 USD/GB-Hour

This is a service plan for the IBM Bluemix Platform runtime.

Create an app:

Space: ecodcnc-qa

Name: PHFirstDeploy

Host: PHFirstDeploy

Domain: mybluemix.net

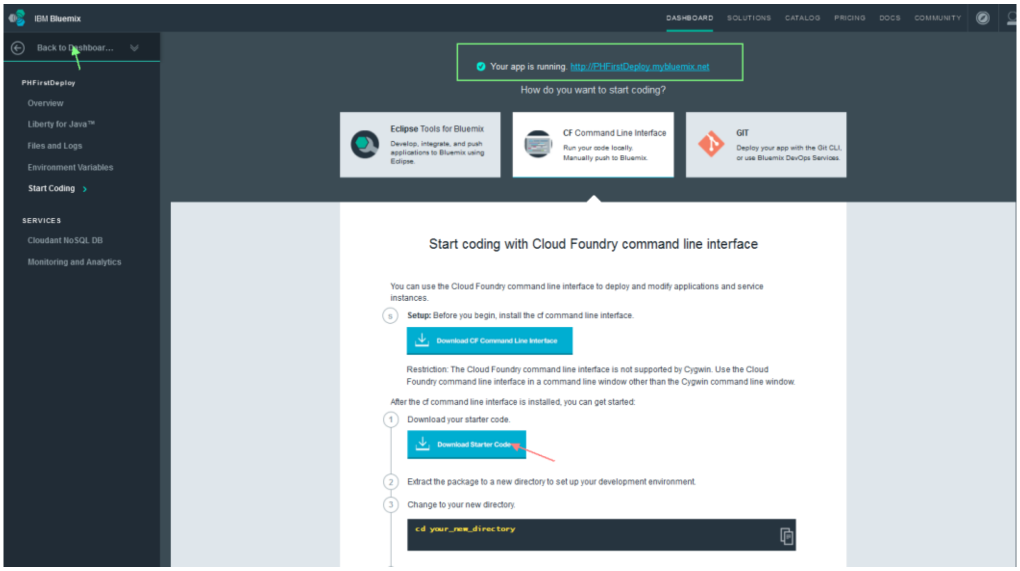
Selected Plan:

Liberty for Java™: Default

Cloudant NoSQL DB: Shared

Monitoring and Analytics: Free

7. When message shows “Your application is running” with the route to your application, click on “Download Starter Code” and save the zip file on your local workstation for later use. Then Click on “Back to Dashboard” on the top left panel to verify the application is up and running.



Your app is running. http://PHFirstDeploy.mybluemix.net

How do you want to start coding?

Eclipse Tools for Bluemix: Develop, integrate, and push applications to Bluemix using Eclipse.

CF Command Line Interface: Run your code locally. Manually push to Bluemix.

GIT: Deploy your app with the Git CLI, or use Bluemix DevOps Services.

Start coding with Cloud Foundry command line interface

You can use the Cloud Foundry command line interface to deploy and modify applications and service instances.

S Setup: Before you begin, install the cf command line interface.

Download CF Command Line Interface

Restriction: The Cloud Foundry command line interface is not supported by Cygwin. Use the Cloud Foundry command line interface in a command line window other than the Cygwin command line window.

After the cf command line interface is installed, you can get started:

1 Download your starter code.

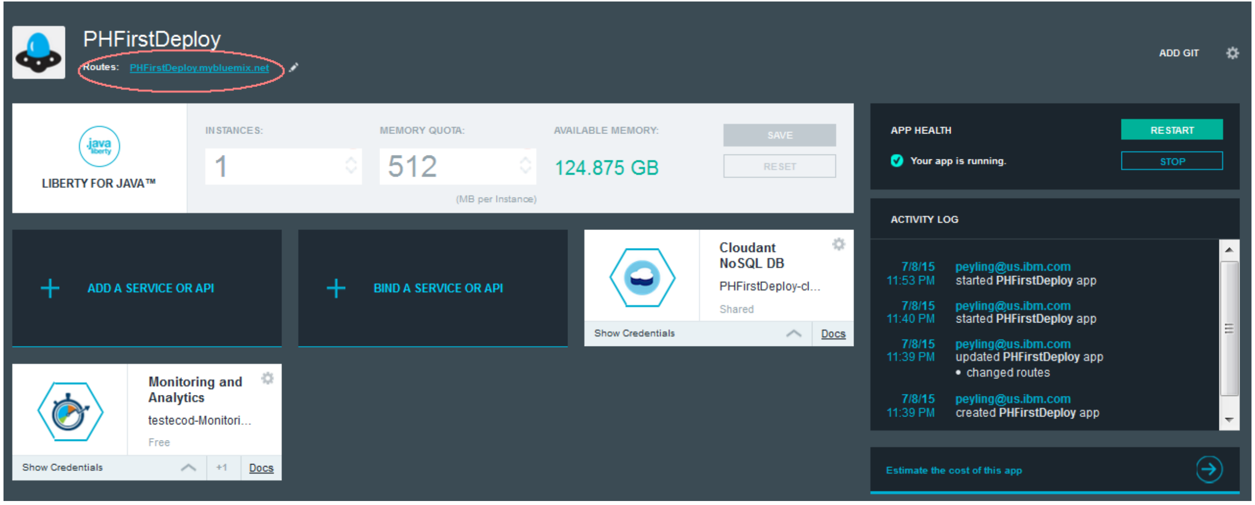
Download Starter Code

2 Extract the package to a new directory to set up your development environment.

3 Change to your new directory.

```
cd your_new_directory
```

8. You can see your application is up and running from Dashboard. Click on the tile of your application, it will show you the details about this application, including usage of service instances & resources, status and activity logs. You can access your application by clicking on the link of routes to ensure your starter code is working fine.



PHFirstDeploy

Routes: PHFirstDeploy.mybluemix.net

INSTANCES: 1

MEMORY QUOTA: 512 (MB per Instance)

AVAILABLE MEMORY: 124.875 GB

APP HEALTH

Your app is running.

Cloudant NoSQL DB
PHFirstDeploy-cl...
Shared

Monitoring and Analytics
testecod-Monitori...
Free

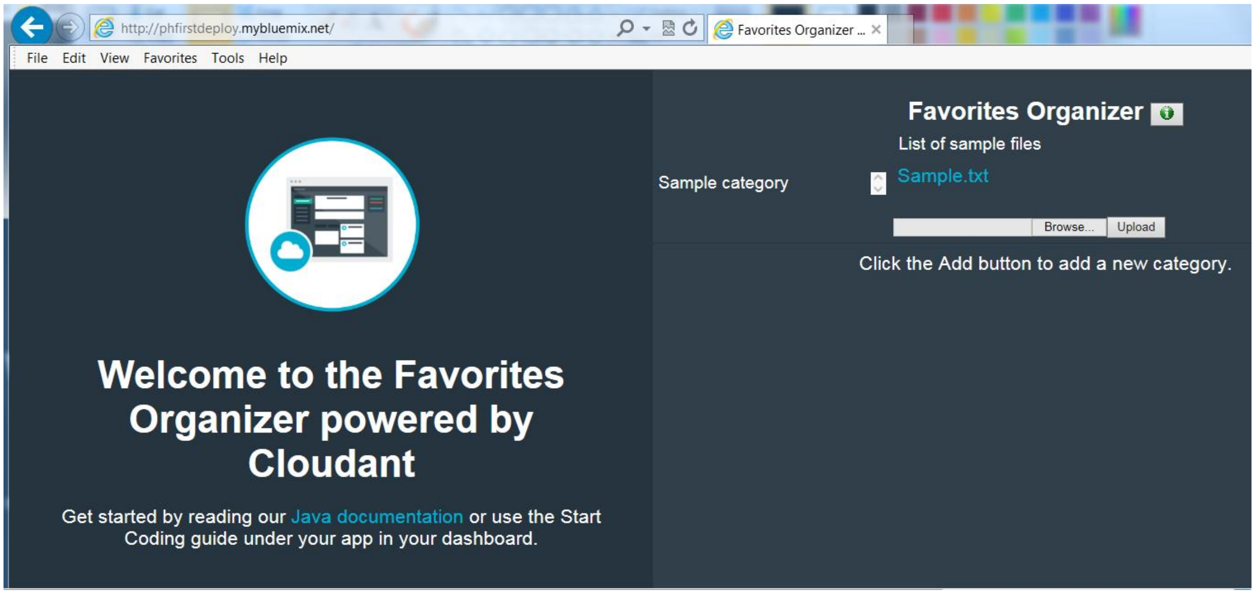
ACTIVITY LOG

7/8/15 11:53 PM peyling@us.ibm.com started PHFirstDeploy app

7/8/15 11:40 PM peyling@us.ibm.com started PHFirstDeploy app

7/8/15 11:39 PM peyling@us.ibm.com updated PHFirstDeploy app
- changed routes

7/8/15 11:39 PM peyling@us.ibm.com created PHFirstDeploy app



Favorites Organizer

List of sample files

Sample category

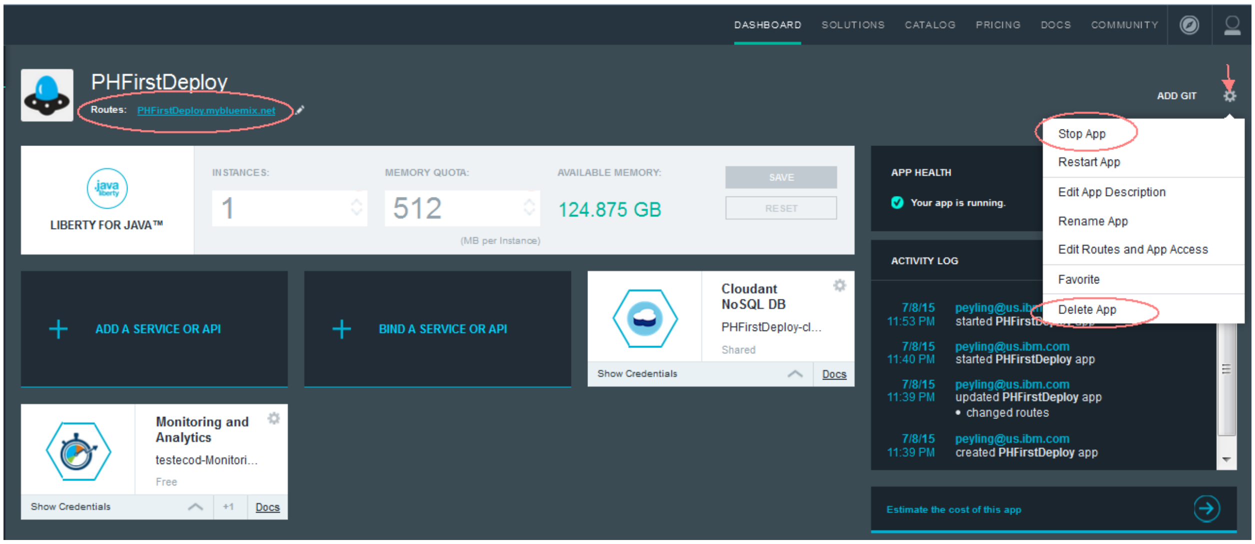
Sample.txt

Click the Add button to add a new category.

Welcome to the Favorites Organizer powered by Cloudant

Get started by reading our Java documentation or use the Start Coding guide under your app in your dashboard.

9. You can administrate your application from dashboard on menu or your application “Overview “section.



PHFirstDeploy

Routes: PHFirstDeploy.mybluemix.net

LIBERTY FOR JAVA™

INSTANCES: 1

MEMORY QUOTA: 512 (MB per Instance)

AVAILABLE MEMORY: 124.875 GB

SAVE RESET

APP HEALTH

Your app is running.

ACTIVITY LOG

7/8/15 11:53 PM peyling@us.ibm.com started PHFirstDeploy app

7/8/15 11:40 PM peyling@us.ibm.com started PHFirstDeploy app

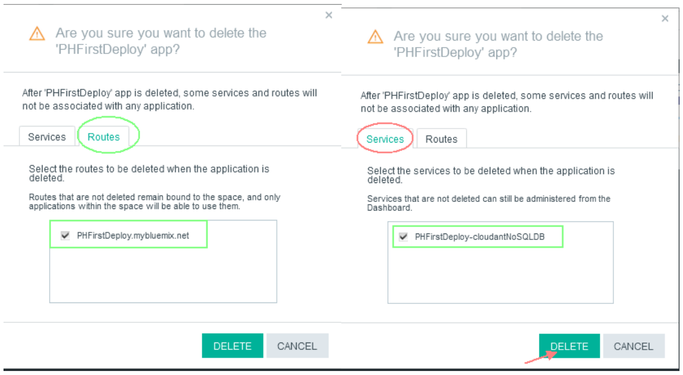
7/8/15 11:39 PM peyling@us.ibm.com updated PHFirstDeploy app
- changed routes

7/8/15 11:39 PM peyling@us.ibm.com created PHFirstDeploy app

Cloudant NoSQL DB PHFirstDeploy-cl... Shared

Monitoring and Analytics testecod-Monitori... Free

10. Click on"Stop App"and then"Delete App"to delete the application you just recreated and access. You want to delete the Service and the Route with the application, so ensure the checkbox in both services tab and the Routes tab are checked. Then click on "DELETE".



Are you sure you want to delete the 'PHFirstDeploy' app?

After 'PHFirstDeploy' app is deleted, some services and routes will not be associated with any application.

Select the routes to be deleted when the application is deleted.

Routes that are not deleted remain bound to the space, and only applications within the space will be able to use them.

☑ PHFirstDeploy.mybluemix.net

Are you sure you want to delete the 'PHFirstDeploy' app?

After 'PHFirstDeploy' app is deleted, some services and routes will not be associated with any application.

Select the services to be deleted when the application is deleted.

Services that are not deleted can still be administered from the Dashboard.

☑ PHFirstDeploy-cloudantNoSQLDB

After complete the Step 10, the application your deleted should be no longer displayed on Bluemix dashboard. We will deploy the same application from Command Line Interface by using the application zip file we just downloaded.

Exercise 2.5.4 – Deploy then update an application using the CLI

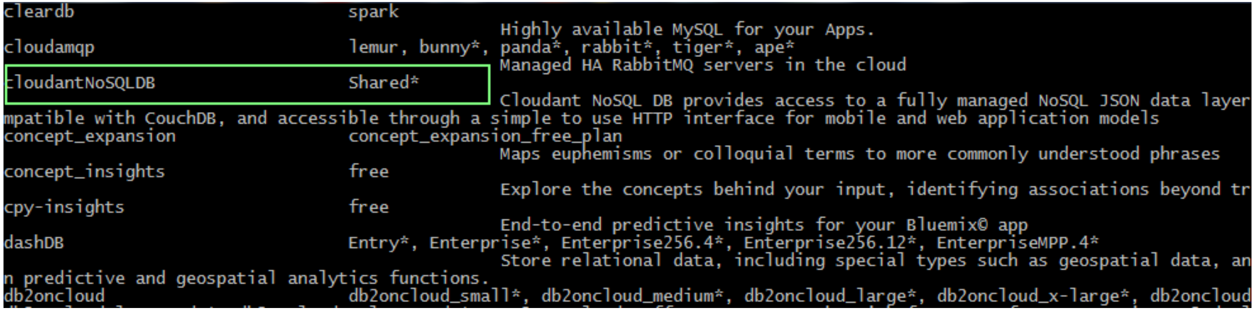
We will deploy and update the same sample application in previous section by using the code in *yourApplication.zip* file you downloaded.

- Open a command windows or Git Bash shell window and change directory to the location you unzipped the downloaded sample application.
- Log in to Bluemix and specify which Bluemix region you want to work with by using **api endpoint –a** option with region URL. so issue one of the following commands, choose region you have been using in Bluemix UI:
  cf l -a https://api.ng.bluemix.net ***(for Region: US South)***
  cf l -a https://api.eu-gb.bluemix.net ***(for Region: United Kingdom)***

Then enter your email and password that you use to sign in to the Bluemix Web UI. Select the organization and space you want to work in if prompted.

- Before we deploy the application we need to deploy a Cloudant database, so we can look at the available services using:
  cf marketplace

Your will get a list of all the services, the one we are interested in is the cloudantNoSQLDB.



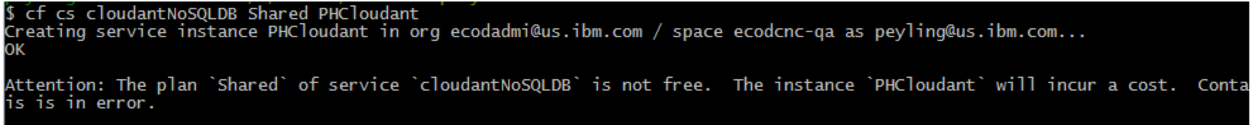
```
cleardb                         spark
                                              Highly available MySQL for your Apps.
cloudamqp                       lemur, bunny*, panda*, rabbit*, tiger*, ape*
                                              Managed HA RabbitMQ servers in the cloud
cloudantNoSQLDB                 Shared*
                                              Cloudant NoSQL DB provides access to a fully managed NoSQL JSON data layer
mpatible with CouchDB, and accessible through a simple to use HTTP interface for mobile and web application models
concept_expansion               concept_expansion_free_plan
                                              Maps euphemisms or colloquial terms to more commonly understood phrases
concept_insights                free
                                              Explore the concepts behind your input, identifying associations beyond tr
cpy-insights                    free
                                              End-to-end predictive insights for your Bluemix© app
dashDB                          Entry*, Enterprise*, Enterprise256.4*, Enterprise256.12*, EnterpriseMPP.4*
                                              Store relational data, including special types such as geospatial data, an
n predictive and geospatial analytics functions.
db2oncloud                      db2oncloud_small*, db2oncloud_medium*, db2oncloud_large*, db2oncloud_x-large*, db2oncloud
```

- To create the service, use: ***cf create-service NameOfService Plan YourNameOfServiceInstance*** command. For example:

  cf cs cloudantNoSQLDB Shared PHCloudant

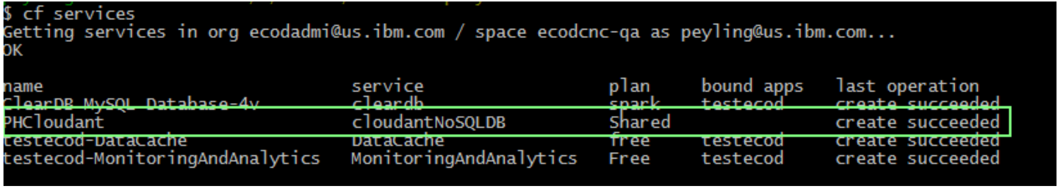
where:

- CloudantNoSQLDB is the name of the service from the cf marketplace command
- Shared and standard are the name of the service plans we want to use from the cf marketplace command
- PHCloudant is the name of the service instances we want to use – please choose your own name rather than PHCloudant – you will need to use this name when connecting (binding) the service to the application.



```
$ cf cs cloudantNoSQLDB Shared PHCloudant
Creating service instance PHCloudant in org ecodadmi@us.ibm.com / space ecodcnc-qa as peyling@us.ibm.com...
OK

Attention: The plan `Shared` of service `cloudantNoSQLDB` is not free.  The instance `PHCloudant` will incur a cost.  Conta
is is in error.
```

- To verify creation of the Cloudant service by using **cf services** command to see a list of service instances you created. The service instances remain unbounded to the application. You will bound the services to application later.



```
$ cf services
Getting services in org ecodadmi@us.ibm.com / space ecodcnc-qa as peyling@us.ibm.com...
OK

name                              service                   plan     bound apps   last operation
ClearDB MySQL Database-4v         cleardb                   spark    testecod     create succeeded
PHCloudant                        cloudantNoSQLDB           Shared                create succeeded
testecod-DataCache                DataCache                 free     testecod     create succeeded
testecod-MonitoringAndAnalytics   MonitoringAndAnalytics    Free     testecod     create succeeded
```

- We can now deploy the application using **cf push *YourAppName*** command. Ensure you are in the directory for your application. You should have the files and directories as following example:

  JavaCloudantDB.war WebContent build.xml instructions.md src README.txt bin dep-jar manifest.yml

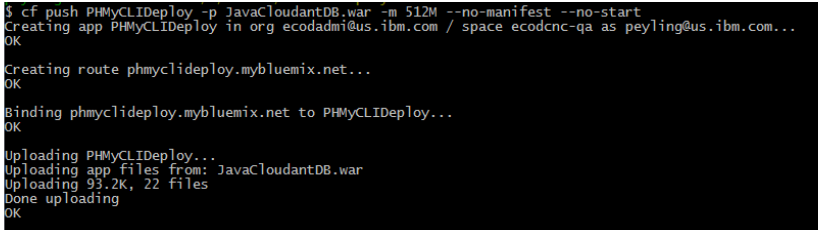
  Enter the following command as example. Use your unique application name which should not be *PHCLIDeploy* for your application.

  cf push PHMyCLIDeploy -p JavaCloudantDB.war -m 512M --no-manifest --no-start

where:

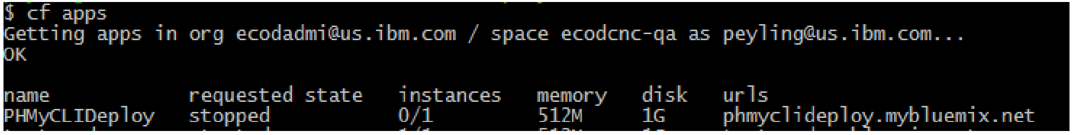
- PHMyCLIDeploy will be the application name and hostname
- -p specifies the path or file (war file) containing the application

- -m specifies the amount of memory to allocate each application instance (1GB is default)
- --no-manifest instructs to CLI tool to ignore the default manifest file. A minimal manifest file requires only application name. Normally it defines services, host, memory, disk-quota, domain, application command, path, instances used by the application for consistency and reproducibility. Ensure the content in the manifest.yml match the application you are deploying if you want to use manifest for deployment.
- --no-start instructs to CLI tool not to automatically start the application. We don't want the application to automatically start because we need to link the Cloudant database instance to the application before starting the application.



```
$ cf push PHMyCLIDeploy -p JavaCloudantDB.war -m 512M --no-manifest --no-start
Creating app PHMyCLIDeploy in org ecodadmi@us.ibm.com / space ecodcnc-qa as peyling@us.ibm.com...
OK

Creating route phmyclideploy.mybluemix.net...
OK

Binding phmyclideploy.mybluemix.net to PHMyCLIDeploy...
OK

Uploading PHMyCLIDeploy...
Uploading app files from: JavaCloudantDB.war
Uploading 93.2K, 22 files
Done uploading
OK
```

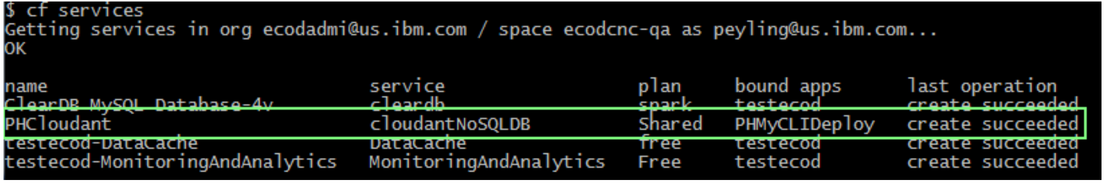
You can use **cf apps** command to list and verify your applications.



```
$ cf apps
Getting apps in org ecodadmi@us.ibm.com / space ecodcnc-qa as peyling@us.ibm.com...
OK

name            requested state   instances   memory   disk   urls
PHMyCLIDeploy   stopped           0/1         512M     1G     phmyclideploy.mybluemix.net
```

- To bind the database service instance to application, use: *cf bs yourAppName yourNameOfServiceInstance* command. For example:
  cf bs PHMyCLIDeploy PHCloudant
  where:
  - PHMyCLIDeploy is the application name used when deploying the application
  - PHCloudant is the service instance name used when deploying the service

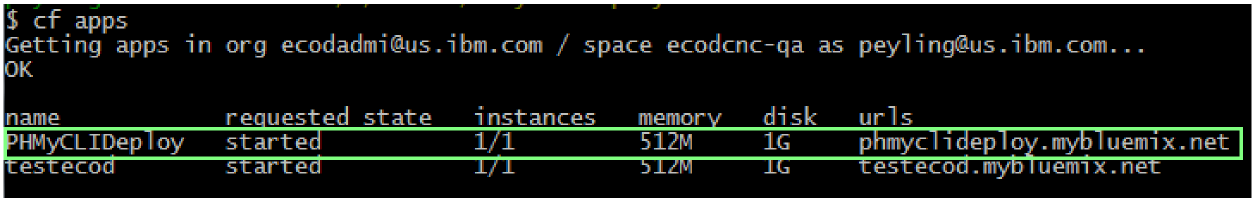
- Type: **cf services** command to verify bind-service result. You should see the application and service now linked, but the application is still stopped.



```
$ cf services
Getting services in org ecodadmi@us.ibm.com / space ecodcnc-qa as peyling@us.ibm.com...
OK

name                             service                  plan     bound apps      last operation
ClearDB MySQL Database-4v        cleardb                  spark    testecod        create succeeded
PHCloudant                       cloudantNoSQLDB          Shared   PHMyCLIDeploy   create succeeded
testecod-DataCache               DataCache                free     testecod        create succeeded
testecod-MonitoringAndAnalytics  MonitoringAndAnalytics   Free     testecod        create succeeded
```

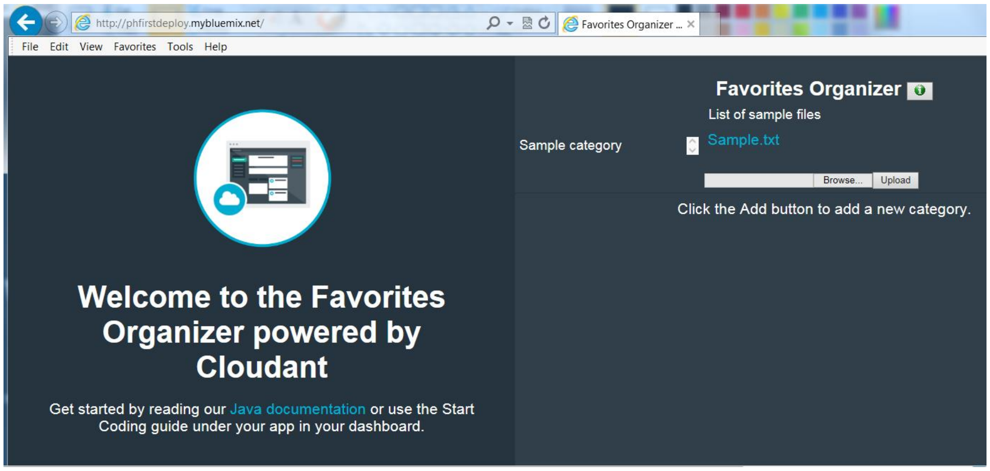
- To start an application use: ***cf start YourAppName*** command. For example:
  cf start PHMyCLIDeploy
  where:
  - PHMyCLIDeploy is the application name you want to start

- Type: **cf apps** command to see the application status has changed from "*stopped* " to "*started*".



```
$ cf apps
Getting apps in org ecodadmi@us.ibm.com / space ecodcnc-qa as peyling@us.ibm.com...
OK

name            requested state   instances   memory   disk   urls
PHMyCLIDeploy   started           1/1         512M     1G     phmyclideploy.mybluemix.net
testecod        started           1/1         512M     1G     testecod.mybluemix.net
```

The application can be accessed via browser by using the urls listed from the result. For example: http://phmyclideploy.mybluemix.net



Click the Add button to add a new category.

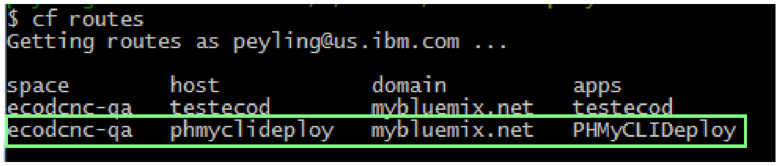
Welcome to the Favorites Organizer powered by Cloudant

Get started by reading our Java documentation or use the Start Coding guide under your app in your dashboard.

- Make some changes to the application and rebuild the application by using **ant**
    1. Edit the file *webContent/index.html* and change "*Welcome to the Favorite Organizer powered by Cloudant*" to "*Welcome to the Lab 2.5 Organizer*" or any text you would like to display. Then save the file.
    - Rebuild the WAR file to reflect the changes by issuing the **ant** command in the root directory of the project (contains build.xml). Ensure you have ant installed, JAVA_HOME and ANT_HOME defined, PATH setup properly in your local workstation.
- Redeploy the updated WAR file with the push command. This time no need to include the --no-start or memory parameter
  cf push PHMyCLIDeploy -p JavaCloudantDB.war --no-manifest

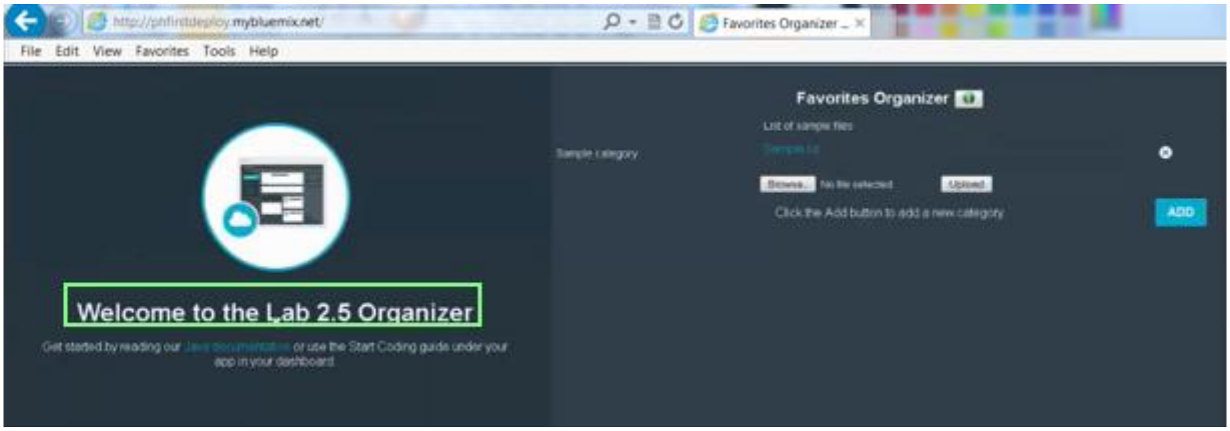
Once the application has restarted, test the application via a browser to ensure your changes are displayed properly.

- To see the route of your application from command line by using ***cf routes*** or ***cf apps*** command.



```
$ cf routes
Getting routes as peyling@us.ibm.com ...

space        host            domain         apps
ecodcnc-qa   testecod        mybluemix.net  testecod
ecodcnc-qa   phmyclideploy   mybluemix.net  PHMyCLIDeploy
```

In showing example, the route of PHMyCLIDeploy application is **host.domain**. For example, the application can be accessed from browser via ***http://phmyclideploy.mybluemix.net***



Click the Add button to add a new category.

Welcome to the Lab 2.5 Organizer

Get started by reading our Java documentation or use the Start Coding guide under your app in your dashboard.

- To stop the running application, use: ***cf stop yourAppName command.*** For example:

cf sp PHMyCLIDeploy

- To delete the application routes and services, use: ***cf d yourAppName –r*** command and ***cf ds yourAppName*** command. For example:
  cf d PHMyCLIDeploy –r
  where:
  - PHmyCLIDeploy is the application name to be deleted
  - -r instructs Bluemix to also delete the routes attached to the application

  cf ds PHCloudant
  where
  - PHCloudant is the name of the service instance to be deleted

  Note : you will be asked to confirm the delete of the application and service answer y to confirm you want to delete

You have explored the application deployment using Bluemix CLI command. We only use some commonly use commands in the lab. You can find out more commands and their options by typing: **cf** command from your windows or command shell. The following tables provide a summary of IBM Bluemix CLI commands as reference for building, deployment and management of your applications.

Bluemix CF command summary

GLOBAL OPTIONS COMMAND			
--version, -v	Print the version	**--help, -h**	Show help
GETTING STARTED COMMAND			
login, l	Log user in	**logout, lo**	Log user out
passwd, pw	Change user password	**target, t**	Set or view the targeted org or space
api	Set or view target api url	**auth**	Authenticate user non-interactively
APP COMMAND			
apps, a	List all apps in the target space	**app**	Display health and status for app
push, p	Push a new app or sync changes to an existing app	**scale**	Change or view the instance count, disk space limit, and memory limit for an app
delete, d	Delete an app	**rename**	Rename an app
start, st	Start an app	**stop, sp**	Stop an app
restart, rs	Restart an app	**restage, rg**	Restage an app
restart-app-instance	Terminate the running application Instance at the given index and instantiate a new instance of the application with the same index	**events**	Show recent app events
files, f	Print out a list of files in a directory or the content of a specific file	**logs**	Tail or show recent logs for an app
env, e	Show all env variables for an app	**set-env, se**	Set an env variable for an app
unset-env	Remove an env variable	**stacks**	List all stacks (a stack is a pre-built file system, including an operating system, that can run apps)
stack	Show information for a stack (a stack is a pre-built file system,	**copy-source**	Make a copy of app source code from one application to another. Unless overridden, the copy-

	including an operating system, that can run apps)		source command will restart the application.

BUILDPACKS			
buildpacks	List all buildpacks	**create-buildpack**	Create a buildpack
update-buildpack	Update a buildpack	**rename-buildpack**	Rename a buildpack
delete-buildpack	Delete a buildpack		

SERVICES COMMAND			
marketplace, m	List available offerings in the marketplace	**services, s**	List all service instances in the target space
service	Show service nstance info	**create-service, cs**	Create a service instance
update-service	Update a service inst.	**delete-service, ds**	Delete a service instance
rename-service	Rename a service instance	**create-service-key, csk**	Create key for a service instance
service-keys, sk	List keys for a service instance	**service-key**	Show service key info
delete-service-key, dsk	Delete a service key	**bind-service, bs**	Bind a service instance to an app
unbind-service, us	Unbind a service instance from an app	**create-user-provided-service, cups**	Make a user-provided service instance available to cf apps
update-user-provided-service, uups	Update user-provided service instance name value pairs		

ROUTES			
routes, r	List all routes in the current space or the current organization	**create-route**	Create a url route in a space for later use
check-route	Perform a simple check to determine whether a route currently exists or not.	**map-route**	Add a url route to an app
unmap-route	Remove a url route from an app	**delete-route**	Delete a route
delete-orphaned-route	Delete all orphaned routes (e.g.: those that are not mapped to an app)		
ORGANIZATIONS			
orgs, o	List all orgs	**org**	Show org info
create-org, co	Create an org	**delete-org**	Delete an org
SPACES			
spaces	List all spaces in an org	**space**	Show space info
create-space	Create a space	**delete-space**	Delete a space
rename-space	Rename a space		
DOMAINS			
domains	List domains in the target org	**create-domain**	Create a domain in an org for later use
delete-domain	Delete a domain	**create-shared-domain**	Create a domain that can be used by all orgs (admin-only)

USER ADMIN COMMAND			
create-user	Create a new user	**delete-user**	Delete a user
org-users	Show org users by role	**set-org-role**	Assign an org role to a user
unset-org-role	Remove an org role from a user	**space-users**	Show space users by role
set-space-role	Assign a space role to a user	**unset-space-role**	Remove a space role from a user
ORGANIZATION ADMIN COMMAND			
quotas	List available usage quotas	**quota**	Show quota info
set-quota	Assign a quota to an org	**create-quota**	Define a new resource quota
delete-quota	Delete a quota	**update-quota**	Update an existing resource quota
share-private-domain	Share a private domain with an org	**unshare-private-domain**	Unshare a private domain with an org
SPACE ADMIN COMMAND			
space-quotas	List available space resource quotas	**space-quota**	Show space quota info
create-space-quota	Define a new space resource quota	**update-space-quota**	update an existing space quota
delete-space-quota	Delete a space quota definition and unassign the space quota from all spaces	**set-space-quota**	Assign a space quota definition to a space
SERVICE ADMIN COMMAND			
service-auth-tokens	List service auth tokens	**create-service-auth-token**	Create a service auth token
update-service-auth-token	Update a service auth token	**delete-service-auth-token**	Delete a service auth token
service-brokers	List service brokers	**create-Service-broker**	Create a service broker
update-Service-broker	Update a service broker	**delete-service-broker**	Delete a service broker

rename-service-broker	Rename a service broker	**migrate-service-instances**	Migrate service instances from one service plan to another
purge-service-offering	Recursively remove a service and child objects from Cloud Foundry database without making requests to a service broker	**service-access**	List service access settings
enable-service-access	Enable access to a service or service plan for one or all orgs	**disable-service-access**	Disable access to a service or service plan for one or all orgs