**Goals**

●      Use CF Plugin for Eclipse / STS to deploy an application

●      Push a simple Java/Spring application to Cloud Foundry

●      Observe a running application

**Prerequisites**

To complete these steps, you must have STS installed, a Cloud Foundry account (Such as Pivotal Web Services)

**Steps**

1.     Open STS or Eclipse.  (For simplicity, the following instructions will simply refer to either as “Eclipse”.

2.     Add a “Cloud Foundry” Server (if you do not already have one)

a.     Select the “Servers” view; this is typically located in the lower left corner of the workspace.  (If it is not present, go to Window / Show View / Other / Server / Servers.)

b.     Within the Servers view, right click, select Pivotal / Cloud Foundry.  Enter the email and password used during sign up

c.     If prompted, take the default values for Organization / Space.

d.     Click Finish.  Note that we have setup the server, but we have not yet deployed any applications.

3.     Import the spring-mvc-demo project into Eclipse

a.     Right-click within the Package Explorer and select “Import…”

b.     Select Maven / Existing Maven Project)

c.     Use “Browse” and locate the folder where you originally installed the spring-mvc-demo project.  Select the project and click Finish.

4.     Deploy the spring-mvc-demo project on Cloud Foundry

a.     Select the spring-mvc-demo project (there may be an outer folder, if so open the folder to find the project within it).

b.     Drag the spring-mvc-demo project and drop on the “Pivotal Cloud Foundry” server.  (Or, right-click on the project, select Run As / Run on Server, and select the Cloud Foundry server)

c.     When prompted, enter a name.  We recommend you choose a short name that is easy to type when working with CF’s command line interface. The name should be unique within your CF account. Leave “buildpack” empty, and check “Save to Manifest”.  Next.

d.     Alter the value of subdomain.  The default value (based on application name) is unlikely to be unique within the cfapps.io domain, so create a longer name by prefixing/suffixing spring-mvc-demo with another value, such as your initials or name, i.e. “keyser.soze. spring-mvc-demo”.  The goal is to make Deployed URL unique across the entire cfapps.io domain.  Next.

e.     Click “next” through the remaining screens.  You won’t need to change any values, but note the type of things that can be controlled during deployment.

                                               i.     Note:  A 400 error encountered at this stage typically indicates a non-unique application name.  Remove your application and try to create a unique name.

5.     Observe the running application

a.     Double click on the “Pivotal Cloud Foundry” server.  Go to the “Applications and Services” tab.  Click on your spring-mvc-demo application.

b.     Click on the “Mapped URLs” link to see your application.  If desired open your app in another browser.

c.     Note the console is updated with the application’s activity.

d.     If you have reached this point, congratulations, you have completed the lab!

Bonus: If you have time, see if you can complete the following:

6.     Scale the application:

a.     In the console view, use the “close console” button to close all consoles that may be open.  There may be several open, so click this button several times.

b.     Change the number of instances from 1 to 2.  Click Restart.

c.     Note the “Instances” section is updated with two running instances (it may take a moment for both instances to restart).

d.     Right click on each instance and select “Show Console”

e.     In the console view, use the “Display Selected Console” button to toggle between the two instances.  Also note that you can use the “Open Console” button to open two consoles at once.

f.      In the browser, refresh the web page.  See if you can determine which instance handled the request

g.     Change instances back to 1 and Restart.

7.     Explore the manifest.

a.     Find the “Manifest” prompt and click the “Save” button.  Click “OK” if prompted about merging.

b.     Right-click the spring-mvc-demo project and select Refresh.  Notice that a “manifest.yml” file is now present within the project.

c.     Open manifest.yml.  Explore the settings and format of this file.

d.     (Note that any changes made to the manifest will not take effect when using the Eclipse interface to Cloud Foundry)

8.      Alter memory setting.

a.     Within Eclipse, alter the memory allocation for spring-mvc-demo to 128 MB.  Note that this will not be enough memory to run the application.

b.     Click Update and Restart.  Observe the console to see what kinds of problems will be reported.  Does the application ever start?  Do you get a clear message about memory?