**Deploy first application to PCF**

1. Run the first project (hello-world-rest-api) locally.
2. Let’s deploy above app on PCF.  
   a. cmd at root dir  
   b. login to PCF  
   c. cf push
3. Provide a name while pushing the app. Let’s say hello-world  
   cf push hello-world   
     
   **Routes should be globally unique**
4. cf push hello-world-rest-api -r hello-world-rest-api-001.cfapps.io  
   cf push hello-world-rest-api --random-route
5. When you push an app, CF automatically detects an appropriate build pack.  
   A buildpack is used to compile and prepare your app for launch.
6. Deploy as .jar  
   cf push hello-world-rest-api -p target/hello-world-rest-api.jar
7. Cf logs app\_name

**Understand Routes** 1. Host + Domain  
  
 2. cf routes  
 cf map-route hello-world-rest-api cfapps.io --hostname hello-world-rest-api-another-host-101  
  
  
**Check Logs and other commands**  
 1. Start/Stop app and check the logs  
  
 2. cf orgs  
 cf spaces  
  
  
**Creating manifest file** 1. Cf create-app-manifest app\_name  
  
 2. cf push  
  
 3. Rename manifest file  
  
 4. Update manifest file for the path and push the app.   
 path: target/hello-world-rest-api.jar

**CF PUSH** 1. Upload Package  
  
 2. Staging – Creation of Droplet (self-contained unit – package + Runtime + Configurations)  
 a. Determine right build pack(s)  
 b. Run the build pack(s) to create droplet  
  
 Droplet is the CF unit of execution.  
  
 3. Deployment – Deployment of droplet into a cell (Virtual Server in the cloud)  
  
 4. Cf v3-droplets app\_name

**Configure Buildpack for Spring Boot App**  
 1. Update manifest file to update for the buildpacks  
 buildpacks:   
 - Java Build pack git\_path  
  
 2. Timeout: 120

**Deploy web project – war file**