

4. Building and deploying the sample EBP project for Heirloom PaaS

You should have already downloaded and installed Eclipse and the Heirloom plugin. To recap:

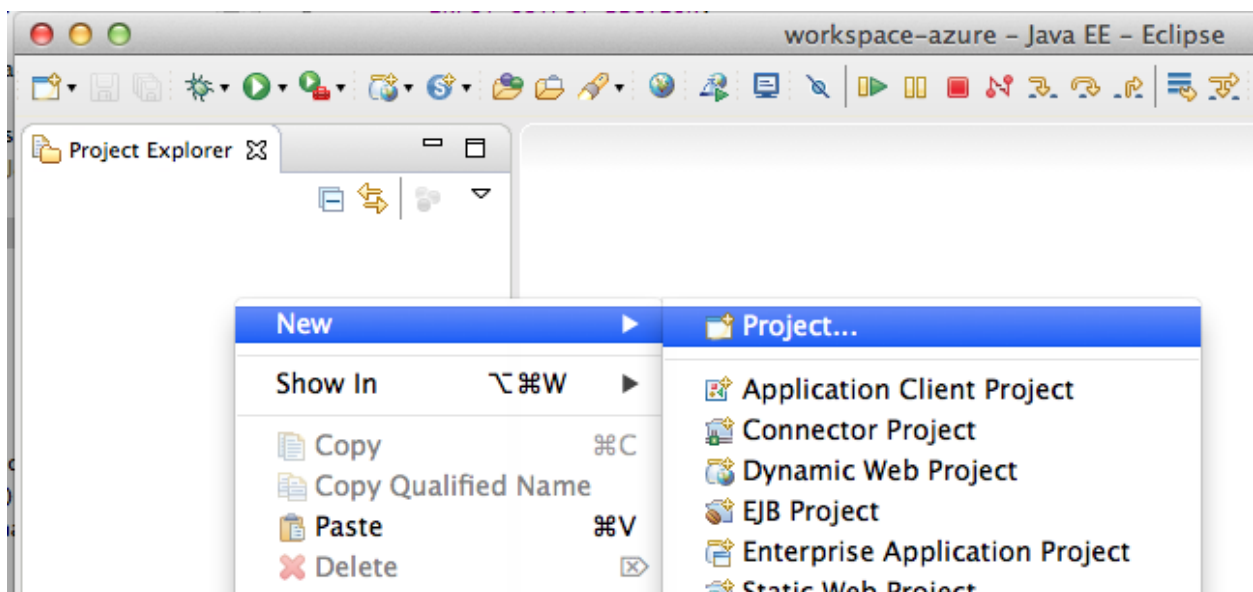
Download and install Eclipse for J2EE from here : <http://www.eclipse.org/downloads/>
Log into paas.heirloomcomputing.com and download our plugin, following instructions there for installation and license placement

Download Sample Source bundle from here: <http://www.elasticcobol.com/downloads/PaaSDemo.zip>

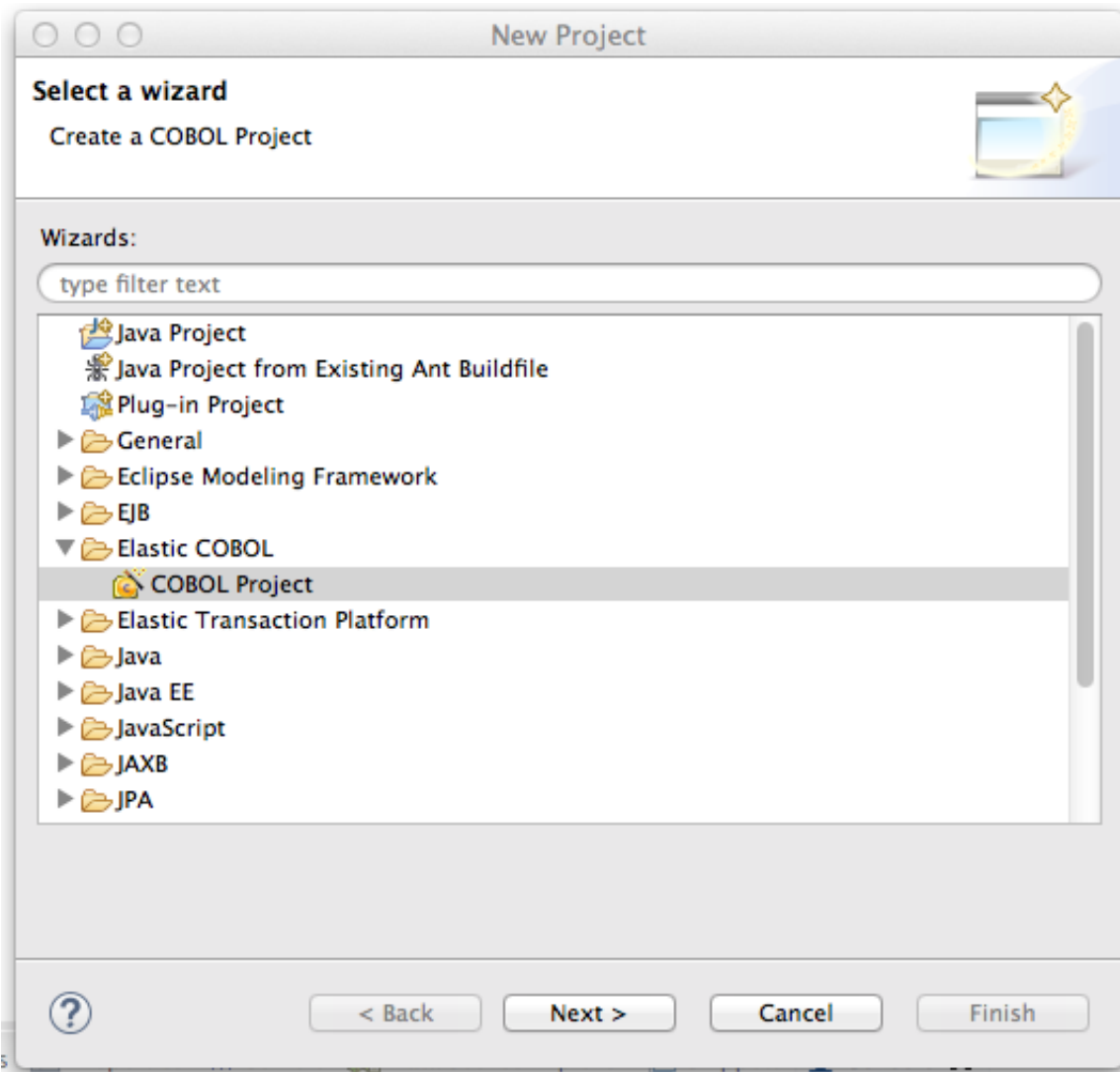
Build:

Open Eclipse

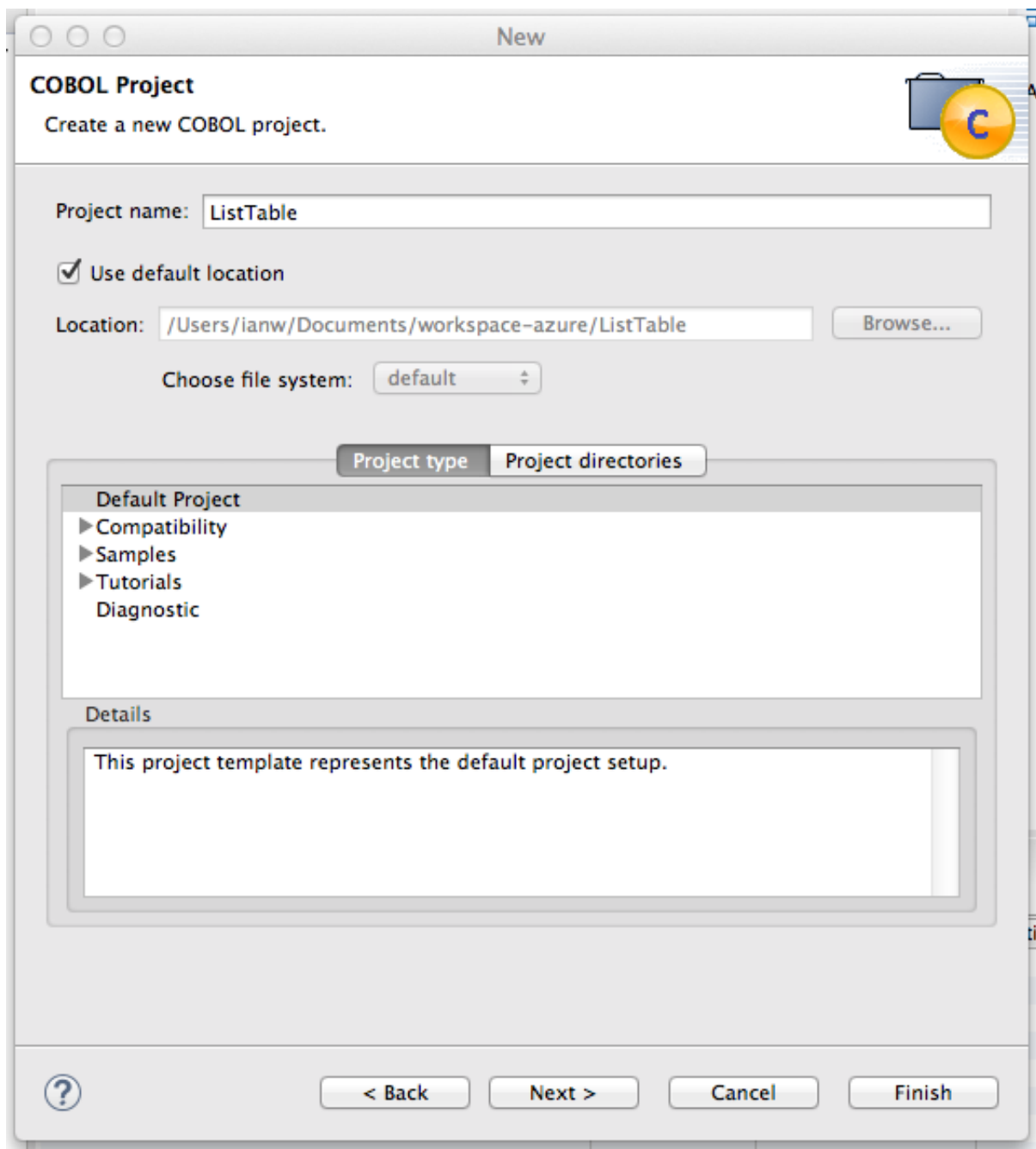
Right click the Project Explorer and select New->Project... :



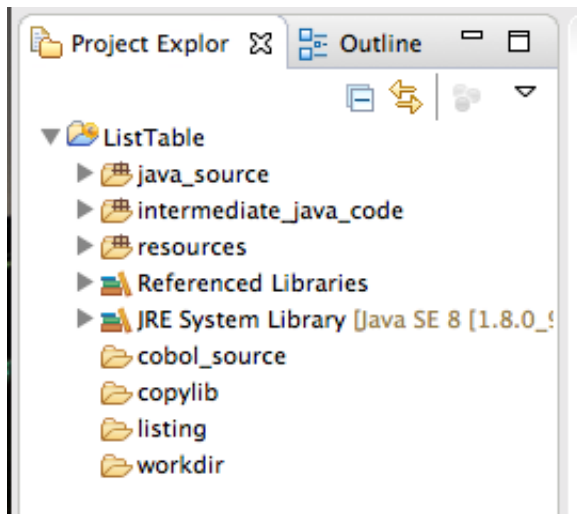
Choose COBOL Project and click Next:



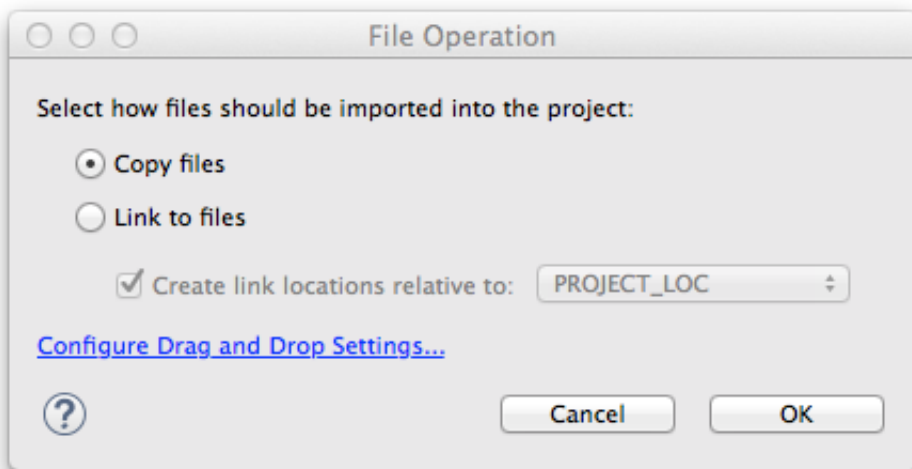
Name the project 'ListTable' , select 'Default Project' under Project Type and click Finish:



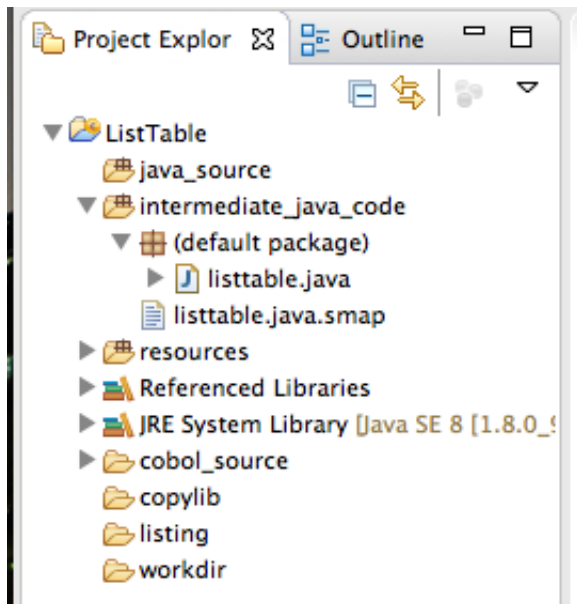
Expand the ListTable project:



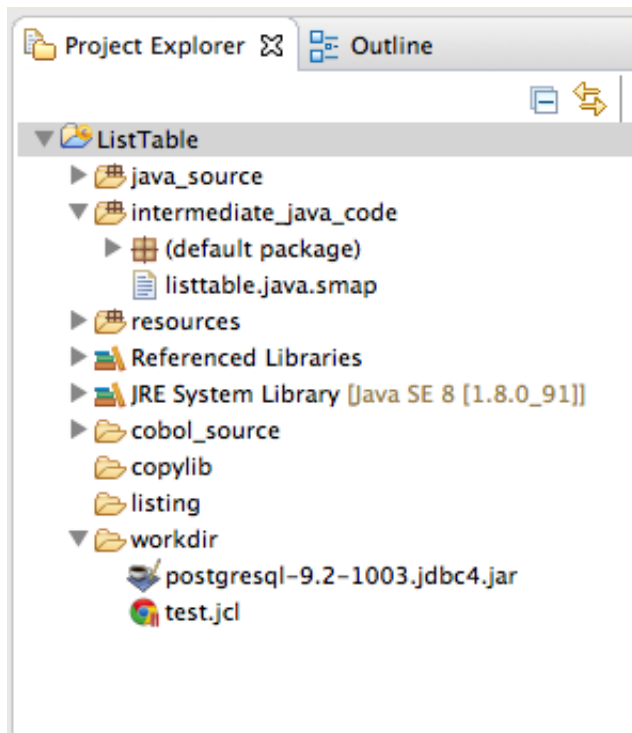
From your source bundle drag the ListTable.cbl file into the cobol_source folder and choose to 'Copy files' and click OK:



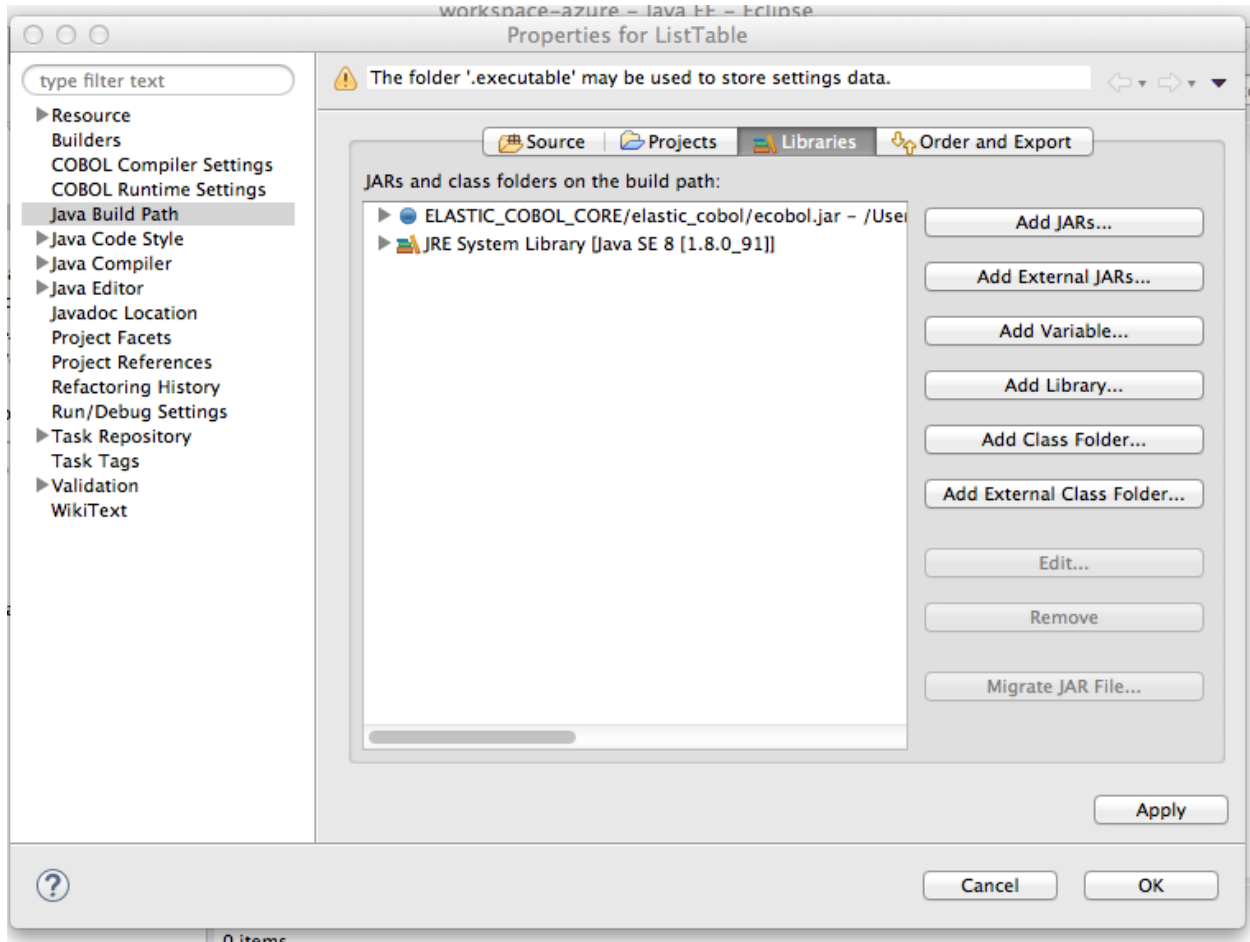
Expand the intermediate_java_code folder and note that Java has been generated from the COBOL:



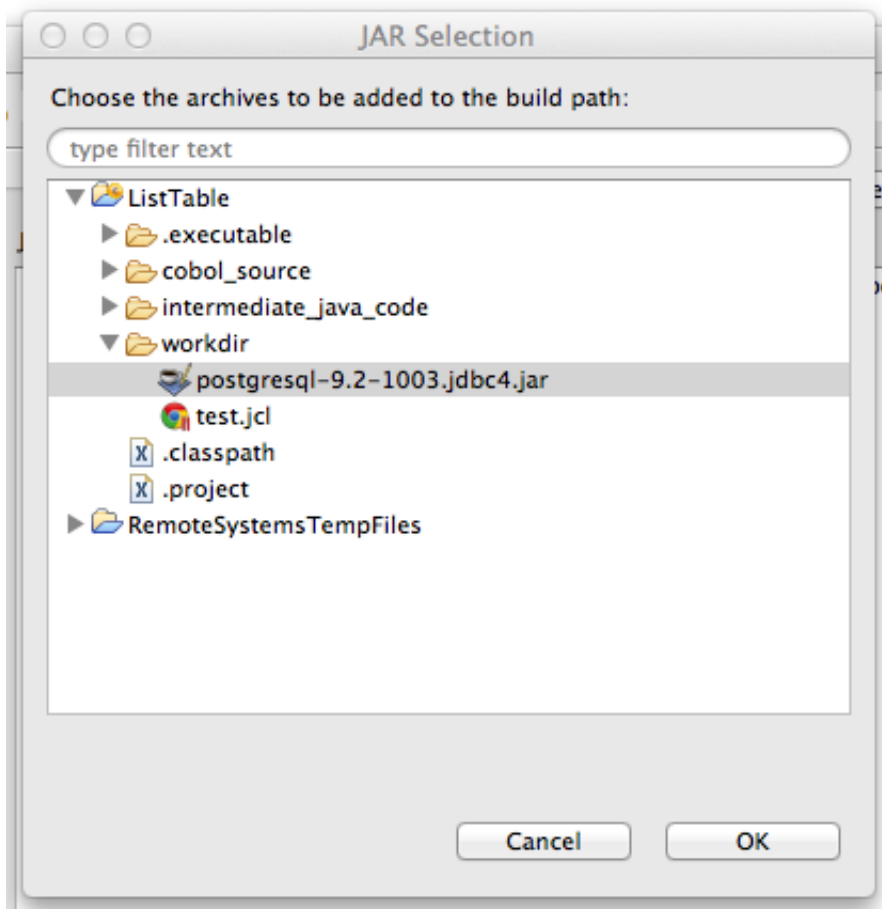
Drag the file test.jcl into the workdir folder and postgresql-9.2-1003.jdbc4.jar into the workdir folder, again choosing to 'Copy files' :



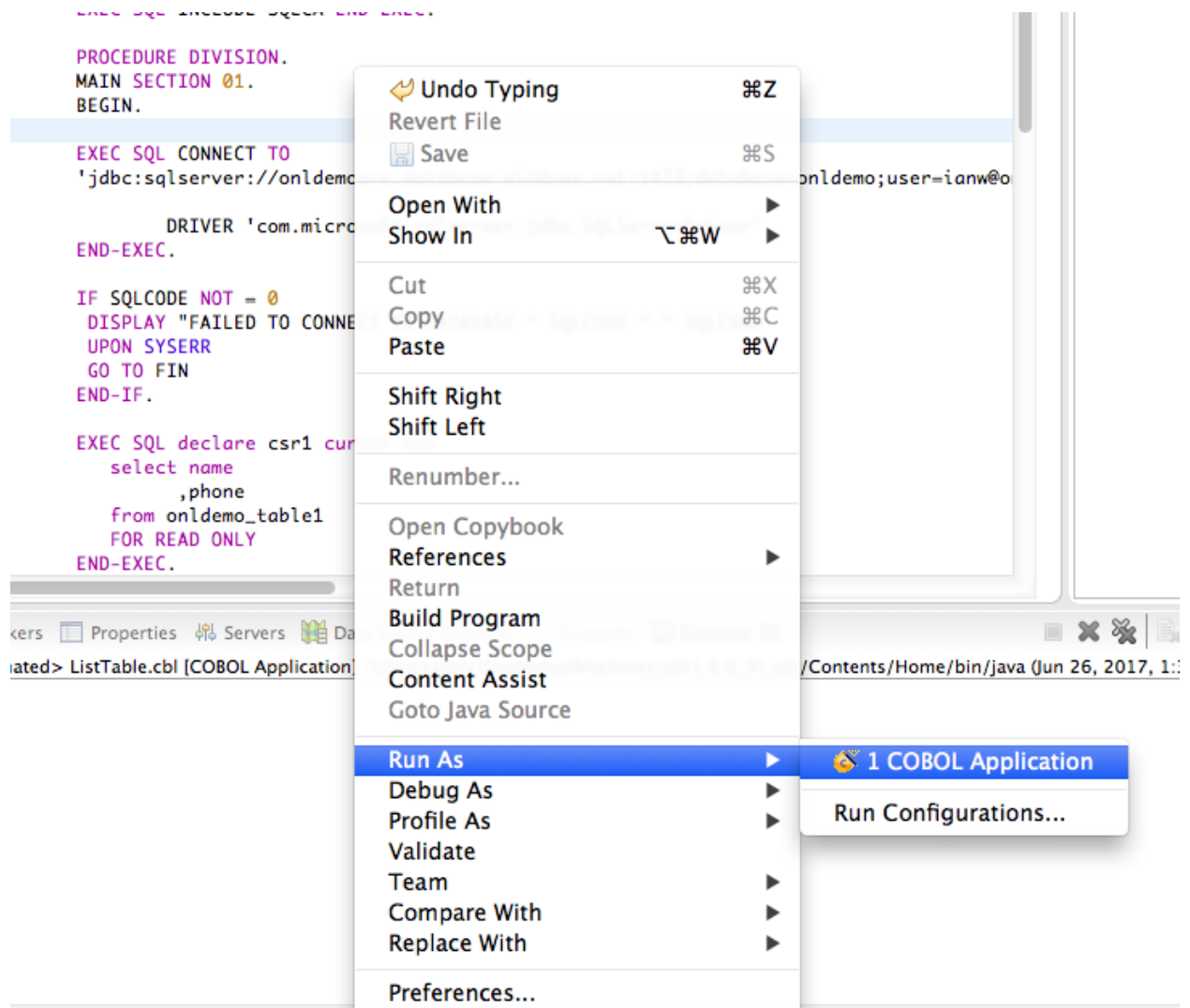
Right click the ListTable Project and choose properties, then click 'Java Build Path':



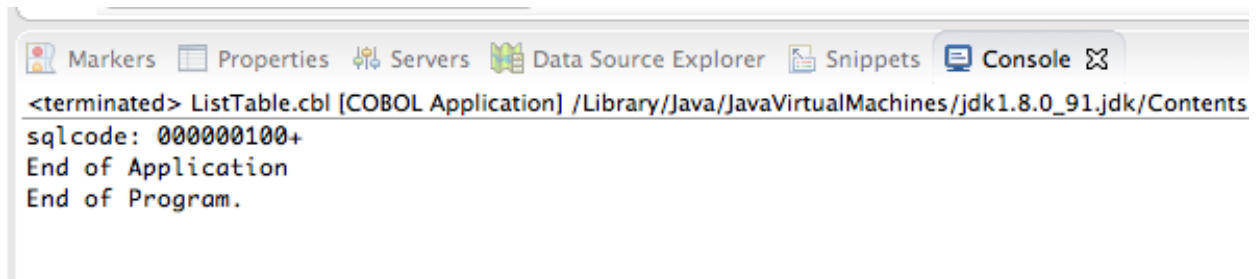
Click 'Add JARS' and expand the ListTable project to the workdir, select postgresql-9.2-1003.jdbc4.jar and click OK:



Open the cobol_source folder , double click ListTable.cbl to bring up the source code. Edit the jdbc driver information with the jdbc connection string you copied from the Amazon RDS console properties. Remember to add in the user and password:

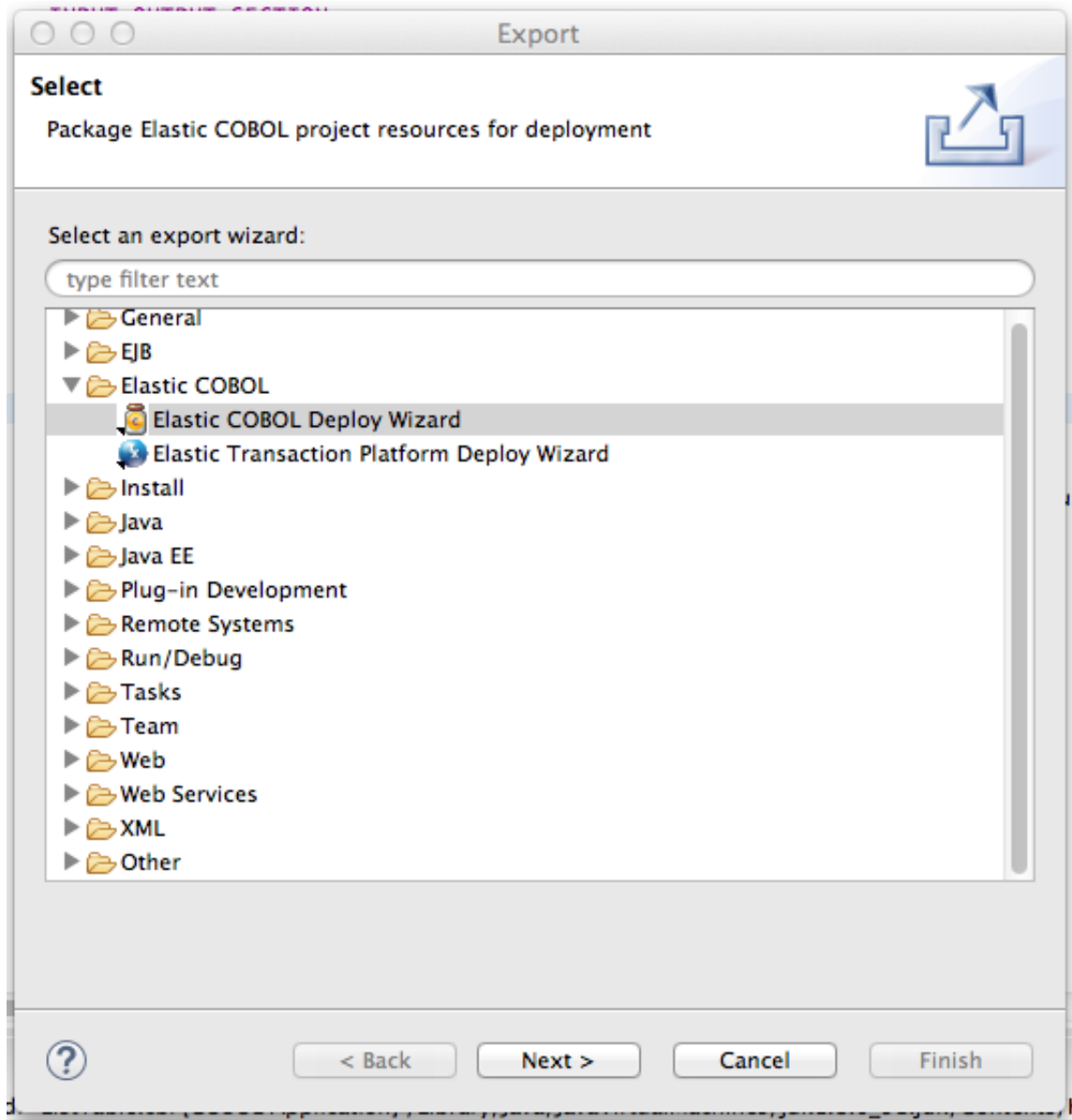


If you have set up the database and table you should see a SQL Code 100 in the console window:



This confirms the program will work correctly after it is exported and run under EBP.

To export this application so it is ready to deploy to EBP later on, right click the ListTable project and choose 'Export':



Choose Elastic COBOL Deploy Wizard and click Next:

COBOL Deploy Wizard

Archive Package Specification

Define what resources to package

Resources to export:

- [ListTable] listtable\$Wrk.class
- [ListTable] listtable.class
- [ListTable] listtable.java.smap

Add...
Remove
▲
▼

Deployment Options

Traditional Cloud

Select the export destination:

JAR or WAR file: Browse...

Specify the deployment license location:

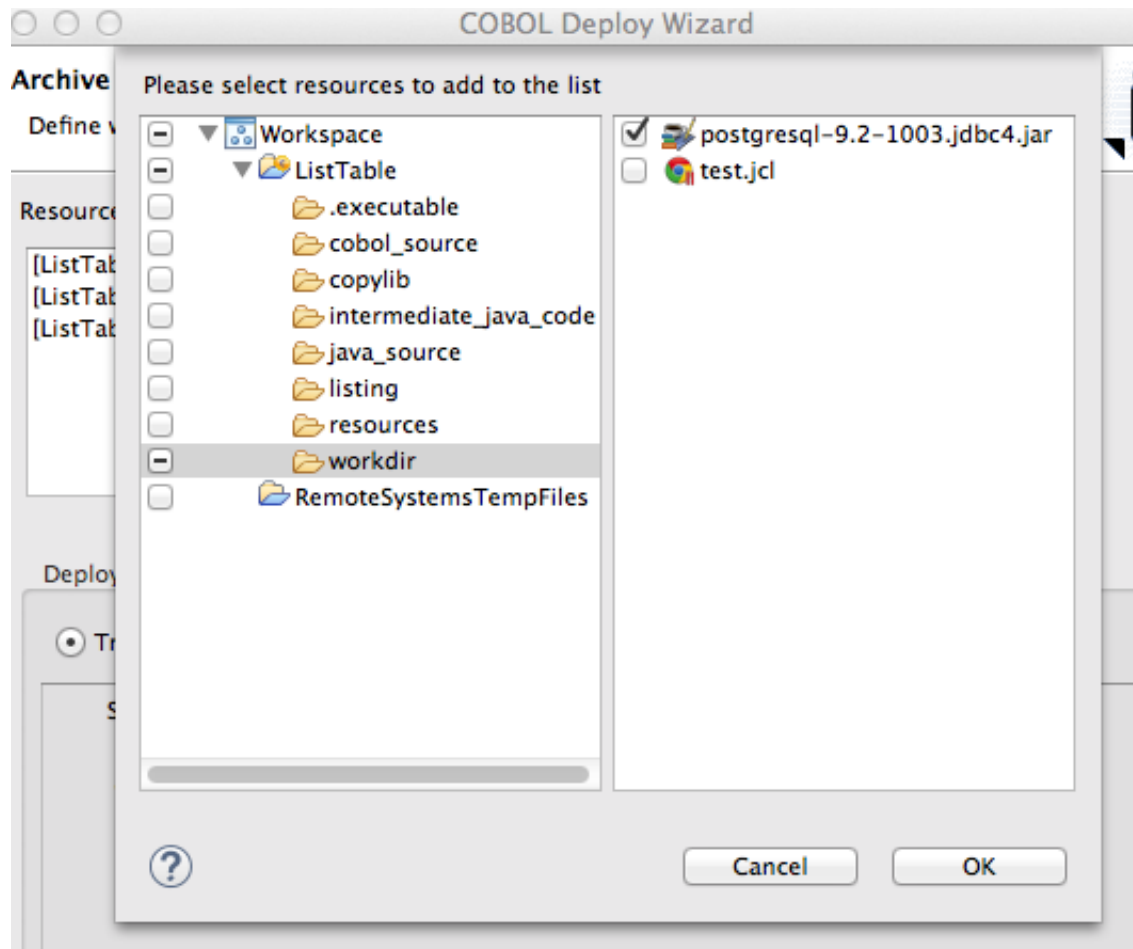
Browse...

Options:

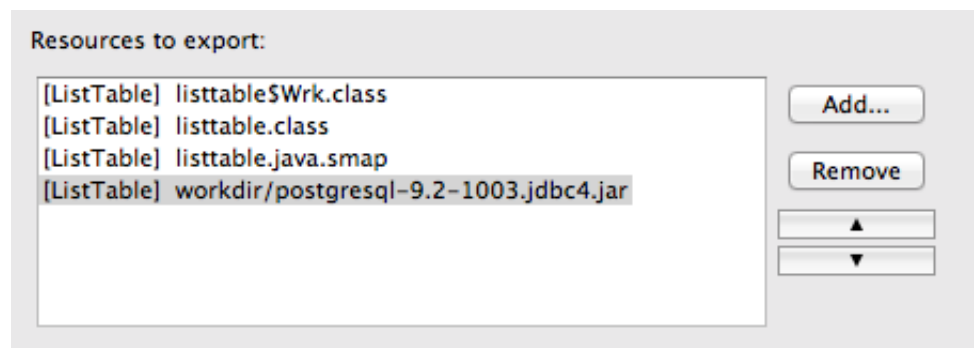
- Compress the contents of the archive
- Overwrite existing files without warning

? < Back Next > Cancel Finish

Fill in the name of the file to export (seen here is /Users/ianw/LISTTABLE.jar) NOTE - The filename should be all lower or all upper case.
Click 'Add..' at the top to add the SQL driver to the export.
Open the workspace/project and workdir folder and select the postgresql-9.2-1003.jdbc4.jar, then click OK:



This will add the driver to the export list:



Click Next and select ListTable.cbl as the Run Configuration:

COBOL Deploy Wizard

Archive Packaging Options

Define required runtime elements and other options

Select runtime elements required by application:

<input checked="" type="checkbox"/> Elastic COBOL Runtime Elements	<input checked="" type="checkbox"/> Core Runtime Elements (Required)
	<input checked="" type="checkbox"/> Printing API
	<input checked="" type="checkbox"/> File Protocol Support
	<input checked="" type="checkbox"/> Graphical Screen Section Support
	<input checked="" type="checkbox"/> JCurse Support (Platform Depend
	<input checked="" type="checkbox"/> XML Support
	<input checked="" type="checkbox"/> Environment Support

Include the selected runtime elements within the export archive

Select the settings to use for launching the application:

Run Configuration:

Select the class of the application entry point:

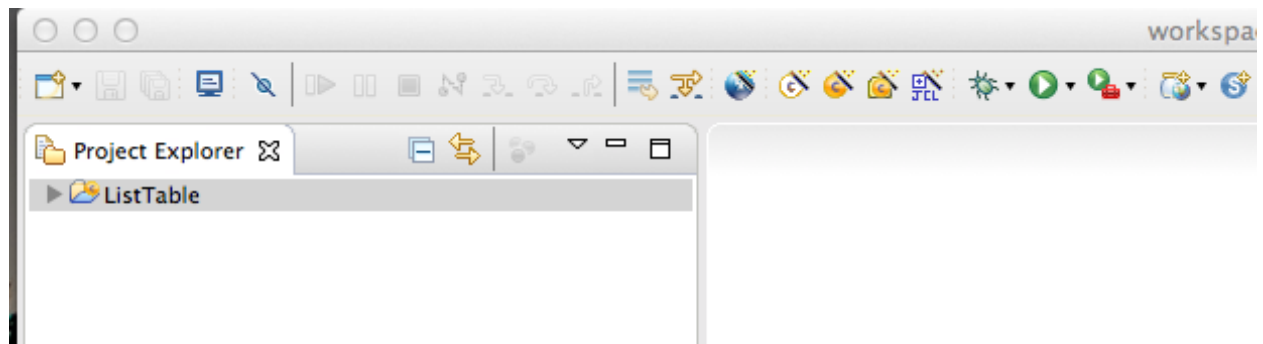
Main Class:

Other Options (JAR only):

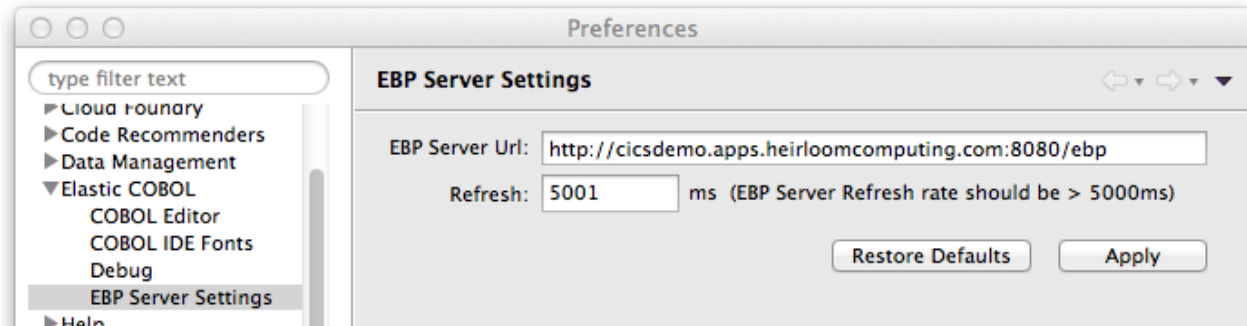
- Copy JAR dependencies from the Elastic COBOL directory to the export directory
- Generate script-based application launcher stub
- Generate HTML-based application launcher stub
 - Generate WAR package from JAR and HTML stub
 - Use Java plugin

Click Finish, the jar file will be created.

Minimize the ListTable project and close the ListTable.cbl source file:

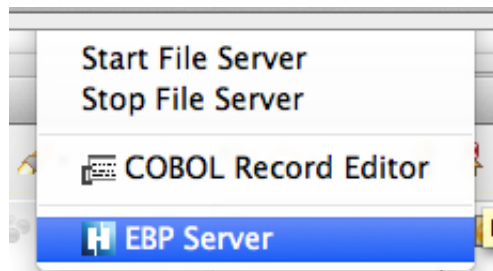


Go to Eclipse - > Preferences, Elastic COBOL/EBP Server settings:



Enter your server URL, set the Refresh rate to be 5001 and click Apply.

From the tools menu bring up the EBP server:



Deploy:

In order to deploy the jar file to the instance we will use the file explorer built into the platform.

Open a webpage to the home page of the application instance:



Heirloom PaaS deployment instance: cicsdemo.apps.heirloomcomputing.com

Instance: [Secure](#) | [File Explorer](#) | [DB Explorer](#) | [Tomcat](#) | [EBP JES/JCL Console](#) | [EBP JES/JCL Configuration](#)

External: [Heirloom Computing](#) | [Heirloom PaaS](#) | [Support](#)

Copyright (c) 2017 [Heirloom Computing, Inc.](#)

Click the File Explorer link and sign in with your PaaS Portal credentials:



Login

Username:

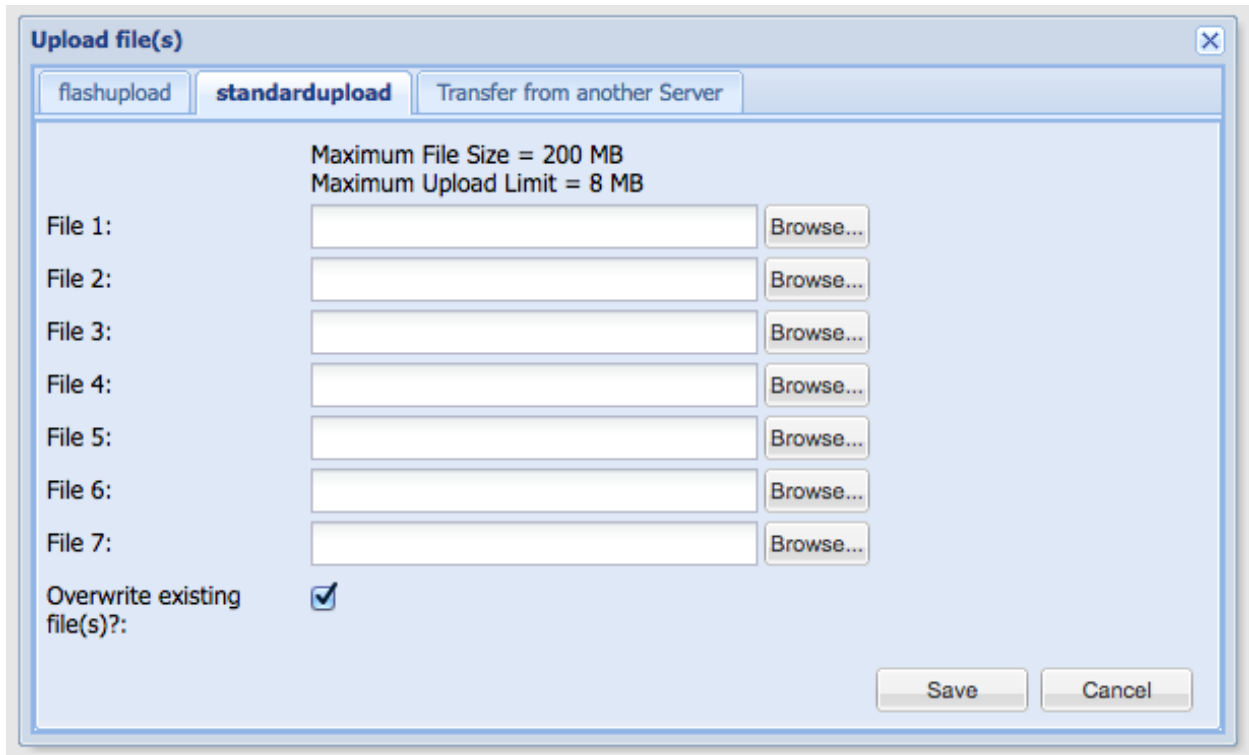
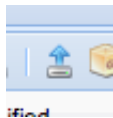
Password:

Language: ▼

Click the 'Data' directory:

Name	Size	Type	Modified	Perms	Owner
data	4 KB	Directory	2015/05/07 11:42	777 (rwxrwxrwx)	n/a
geronimo	4 KB	Directory	2012/02/09 07:31	777 (rwxrwxrwx)	n/a
h2	4 KB	Directory	2015/05/07 11:59	755 (rwxr-xr-x)	n/a

Click the upload icon to bring up the file upload dialog:

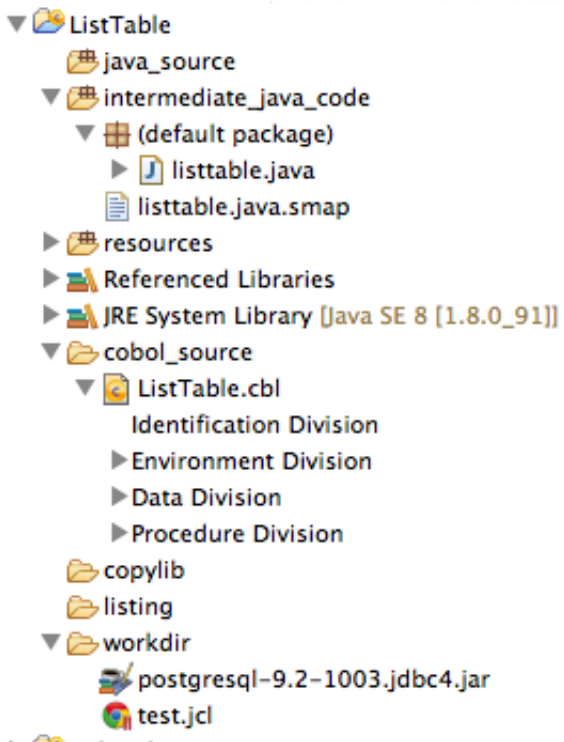


Click the 'Browse' button next to the 'File 1:' field and select your LISTTABLE.jar file, then click Save:

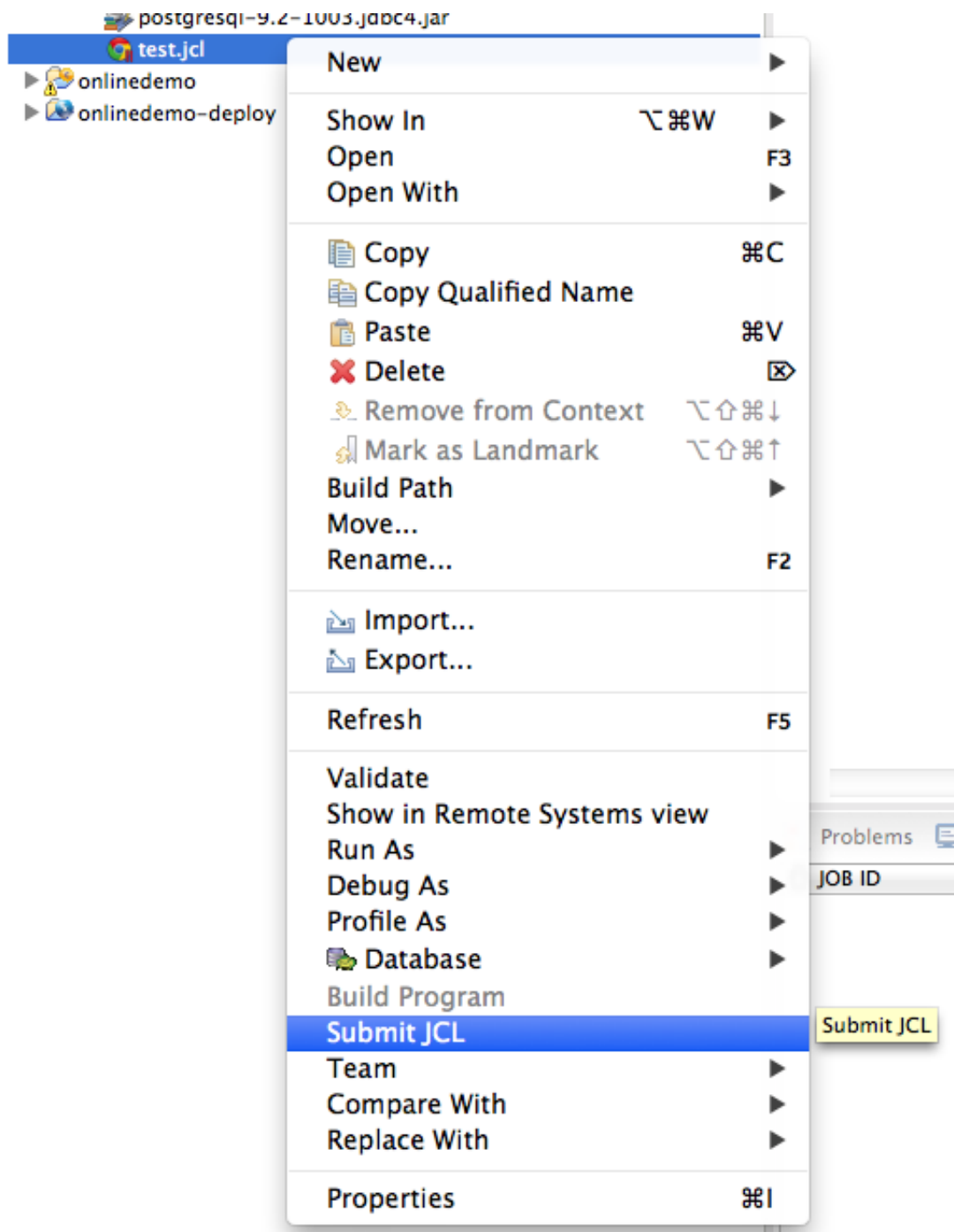
Name	Size	Type	Modified	Perms	Owner
spool	4 KB	Directory	2014/10/19 19:25	777 (rwxrwxrwx)	n/a
tape	4 KB	Directory	2014/10/19 19:25	777 (rwxrwxrwx)	n/a
HCI00.TESTALL.LOG	28 B	File	2015/05/07 11:42	644 (rw-r--r--)	n/a
LISTTABLE.jar	4.15 MB	Jar	2017/08/11 15:22	644 (rw-r--r--)	n/a

You are now ready to test the jar file using JCL.

Switch back to Eclipse and expand the ListTable project and its workdir folder:



Right click the 'test.jcl' and click 'Submit JCL':

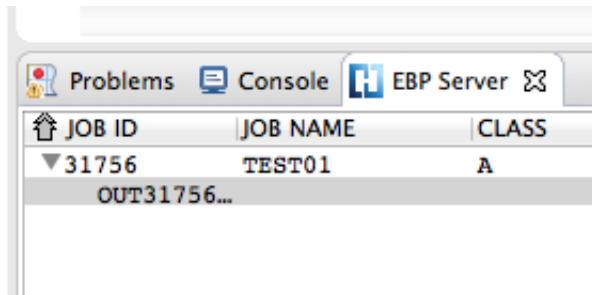


This will send the JCL to EBP, click the EBP Server tab:

JOB ID	JOB NAME	CLASS	TYPE	USER	STATE	SUBMITTED	INITIATED	COMPLETED	TIME	MESSAGE
31756	TEST01	A	JCL	EBP	TERMINATED	2017-08-13T0...	2017-08-13T0...	2017-08-13T0...	0	WITH CC 0

The tab will refresh until you see the job listed and you will see a Message of 'WITH CC 0' which means the job completed correctly.

Click the left hand arrow to drop down the outputs of the job:



Double click the OUTnnnn file and it will open up in Eclipse:

```

JOB:31756::MSG
1 2017-08-13 03:12:32 HEIRLOOM COMPUTING INC - ELASTIC BATCH PLATFORM - JES/JCL (v17.6.26)
2
3 2017-08-13 03:12:32 $HASP111I JOB 31756 TEST01   STARTED                               ON CLASS A
4 2017-08-13 03:12:32 $HASP117I JCL SCAN START
5 2017-08-13 03:12:32 $HASP118I                /* :....1.....2.....3.....4.....5.....6.....7.
6 2017-08-13 03:12:32 $HASP118I                //TEST01  JOB (HCIACCT),'EBP Table Test',CLASS=A,MSGCLASS=A,
7 2017-08-13 03:12:32 $HASP118I                //                MSGLEVEL=(2,2),NOTIFY=USERID
8 2017-08-13 03:12:32 $HASP118I                //STEPID01 EXEC PGM=ListTable
9 2017-08-13 03:12:32 $HASP118I                //SYSIN   DD   DUMMY
10 2017-08-13 03:12:32 $HASP118I               //SYSPRINT DD  SYSOUT=*
11 2017-08-13 03:12:32 $HASP118I               //SYSOUT  DD  SYSOUT=*
12 2017-08-13 03:12:32 $HASP118I               //SYSERR  DD  SYSOUT=*
13 2017-08-13 03:12:32 $HASP117I JCL SCAN END
14
15 2017-08-13 03:12:32 $HASP125I JOB ANALYSIS START
16 2017-08-13 03:12:32 $HASP126I                JOB ATTRIBUTE NAME: TEST01
17 2017-08-13 03:12:32 $HASP126I                JOB ATTRIBUTE CLASS: A
18 2017-08-13 03:12:32 $HASP126I                JOB ATTRIBUTE ACCOUNT: HCIACCT
19 2017-08-13 03:12:32 $HASP126I                JOB ATTRIBUTE PROGRAMMER: EBP Table Test
20
21 2017-08-13 03:12:32 $HASP126I                STEP ATTRIBUTE NAME: STEPID01
22 2017-08-13 03:12:32 $HASP126I                STEP ATTRIBUTE PGM: ListTable
23 2017-08-13 03:12:32 $HASP126I                DD ATTRIBUTE NAME: SYSIN
24 2017-08-13 03:12:32 $HASP126I                DD ATTRIBUTE NAME: SYSPRINT
25 2017-08-13 03:12:32 $HASP126I                DD ATTRIBUTE NAME: SYSOUT
26 2017-08-13 03:12:32 $HASP126I                DD ATTRIBUTE NAME: SYSERR
27
28 2017-08-13 03:12:32 $HASP125I JOB ANALYSIS STOP
29
30
31 2017-08-13 03:12:32 $HASP135I JOB 31756 TEST01  STEP STEPID01          STARTED
32 2017-08-13 03:12:34 $HASP136I JOB 31756 TEST01  STEP STEPID01          TERMINATED          WITH CC 0
33
34 2017-08-13 03:12:34 $HASP112I JOB 31756 TEST01  TERMINATED                WITH CC 0
35 2017-08-13 03:12:34 =====
36 2017-08-13 03:12:34 $HASP174I OUTPUT FROM STEP 'STEPID01' DD NAME 'SYSPRINT'
37 2017-08-13 03:12:34 =====
38 2017-08-13 03:12:34 $HASP174I OUTPUT FROM STEP 'STEPID01' DD NAME 'SYSOUT'
39 name: ian                : 123456
40 name: IAN                : 512-555-1234
41 name: BOB                : 408-333-2222
42 sqlcode: 000000100+
43 End of Program.
44 2017-08-13 03:12:34 =====
45 2017-08-13 03:12:34 $HASP174I OUTPUT FROM STEP 'STEPID01' DD NAME 'SYSERR'
46

```

Congratulations!

You just configured,built,deployed,and tested a CICS application, and a JCL/JES batch application to the Heirloom PaaS