

CPSC 476 Problemset 1
Daniel Jordan
daniel_jordan@csu.fullerton.edu

1. Download and install a Java 7 JDK. Verify that it works by running `javac -version`.

JDK downloaded from Oracle's website

```
dan@ubuntu: ~  
File Edit View Search Terminal Help  
dan@ubuntu:~$ javac -version  
javac 1.7.0_40  
dan@ubuntu:~$
```

2. Type in the Java code for `VanillaDataReaderClient` and friends from the Object Coupling Problem section of *Just Spring* Chapter 1 and convince it to build and run successfully from the command line.

```
dan@ubuntu: ~/school/CPSC476/problemset1  
File Edit View Search Terminal Help  
dan@ubuntu:~/school/CPSC476/problemset1$ javac -d classes -sourcepath src src/com/problemset1/apps/VanillaDataReaderClient.java  
dan@ubuntu:~/school/CPSC476/problemset1$ java -cp classes com.problemset1.apps.VanillaDataReaderClient  
Got data using no-spring: text,file,for,cpsc476,problem,set,number,1,  
dan@ubuntu:~/school/CPSC476/problemset1$
```

3. Download and install Apache Ant 1.9.2. Verify that it works by running `ant -version`.

I downloaded Ant through Ubuntu repositories with “`sudo apt-get install ant`”

```
dan@ubuntu: ~  
File Edit View Search Terminal Help  
dan@ubuntu:~$ ant -version  
Apache Ant(TM) version 1.8.2 compiled on December 3 2011  
dan@ubuntu:~$
```

4. Create an Ant build.xml to build and run `VanillaDataReaderClient`.

```
dan@ubuntu: ~/school/CPSC476/problemset1  
File Edit View Search Terminal Help  
dan@ubuntu:~/school/CPSC476/problemset1$ ant run-coupled  
Buildfile: /home/dan/school/CPSC476/problemset1/build.xml  
  
init:  
[mkdir] Created dir: /home/dan/school/CPSC476/problemset1/classes  
  
compile:  
[javac] Compiling 4 source files to /home/dan/school/CPSC476/problemset1/classes  
  
run-coupled:  
[java] Got data using no-spring: text,file,for,cpsc476,problem,set,number,1,  
  
BUILD SUCCESSFUL  
Total time: 1 second  
dan@ubuntu:~/school/CPSC476/problemset1$
```

5. Download and install the Maven Ant Tasks. Read about the Dependencies task, then create an Ant build.xml to install Spring Framework 3.2.4.RELEASE.

Maven task to download spring 3.2.4. XML namespace defined by antlib:org.apache.maven.artifact.ant.

```
<target name="init">
  <mkdir dir="${classes}" />
  <artifact:dependencies pathId="dependency.classpath">
    <dependency groupId="org.springframework"
      artifactId="spring-context"
      version="3.2.4.RELEASE"/>
  </artifact:dependencies>
</target>
```

Maven downloading dependencies, including spring-context-3.2.4.RELEASE.jar.

```
dan@ubuntu: ~/school/CPSC476/problemset1
File Edit View Search Terminal Help
[artifact:dependencies] Downloading: org/springframework/spring-context/3.2.4.RELEASE/spring-context-3.2.4.RELEASE.jar from repository central
1 at http://repo1.maven.org/maven2
```

6. Type in the Java code for DecoupledDataReader and friends from the Introducing Spring section of Just Spring Chapter 1 and add compile and run targets to your build.xml file.

```
dan@ubuntu: ~/school/CPSC476/problemset1
File Edit View Search Terminal Help
dan@ubuntu:~/school/CPSC476/problemset1$ ant run-decoupled
Buildfile: /home/dan/school/CPSC476/problemset1/build.xml

init:
[mkdir] Created dir: /home/dan/school/CPSC476/problemset1/classes

compile:
[javac] Compiling 4 source files to /home/dan/school/CPSC476/problemset1/classes

run-decoupled:
[java] Sep 28, 2013 4:11:32 PM org.springframework.context.support.AbstractApplicationContext prepareRefresh
[java] INFO: Refreshing org.springframework.context.support.ClassPathXmlApplicationContext@243250b1: startup date [Sat Sep 28 16:11:32 P
T 2013]; root of context hierarchy
[java] Sep 28, 2013 4:11:32 PM org.springframework.beans.factory.xml.XmlBeanDefinitionReader loadBeanDefinitions
[java] INFO: Loading XML bean definitions from class path resource [./beans.xml]
[java] Sep 28, 2013 4:11:32 PM org.springframework.beans.factory.support.DefaultListableBeanFactory preInstantiateSingletons
[java] INFO: Pre-instantiating singletons in org.springframework.beans.factory.support.DefaultListableBeanFactory@f13357: defining beans
[reader]; root of factory hierarchy
[java] Using Decoupled Data Client, Got data: text,file,for,cpsc476,problem,set,number,1,

BUILD SUCCESSFUL
Total time: 2 seconds
dan@ubuntu:~/school/CPSC476/problemset1$
```

7. Download create-qanda-tables.sql, insert-qanda-data.sql, and the CPSC 476 HSQLDB Package. Extract the package, start the database, and populate it. Keep this database for use with Problem Set 2.

Starting the database

```
dan@ubuntu: ~/school/CPSC476/problemset1/db
File Edit View Search Terminal Help
dan@ubuntu:~/school/CPSC476/problemset1/db$ ant start
Buildfile: /home/dan/school/CPSC476/problemset1/db/build.xml

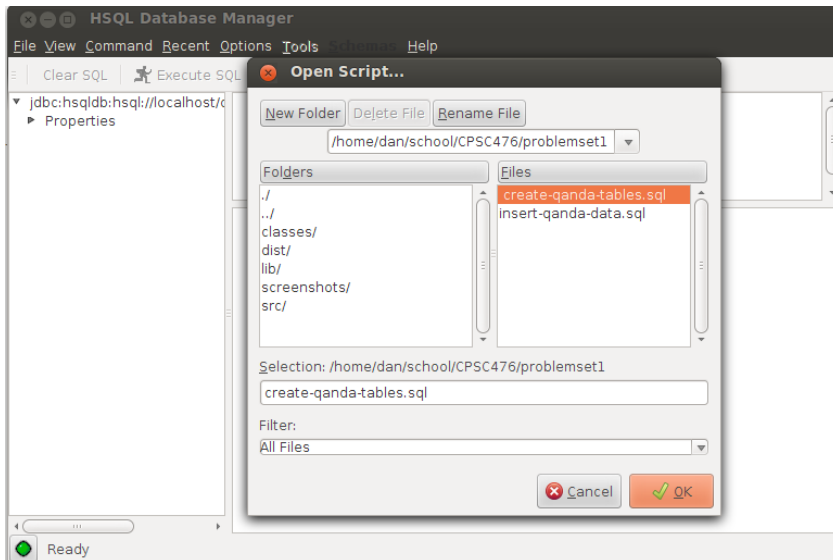
check-db-exists:

init:

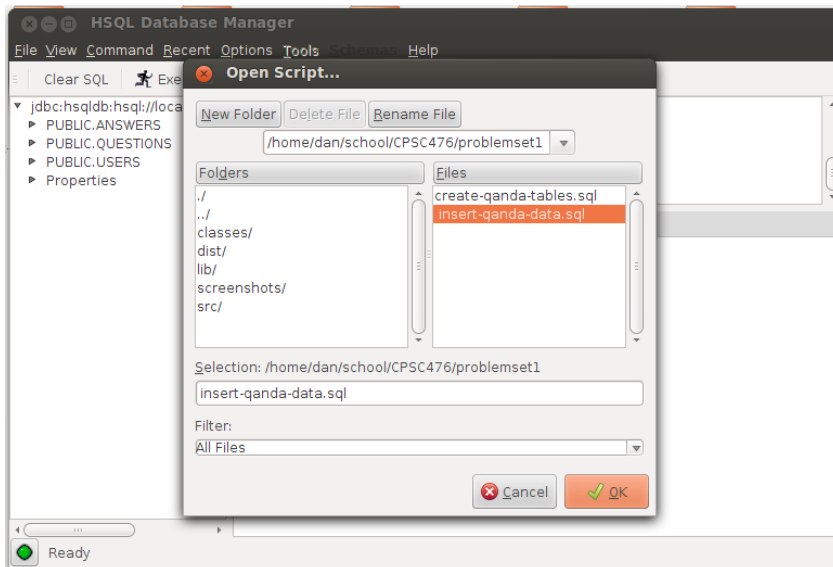
start:
[echo] HSQLDB started

BUILD SUCCESSFUL
Total time: 0 seconds
dan@ubuntu:~/school/CPSC476/problemset1/db$
```

Creating the database



Populating database



Verifying data in database

