README

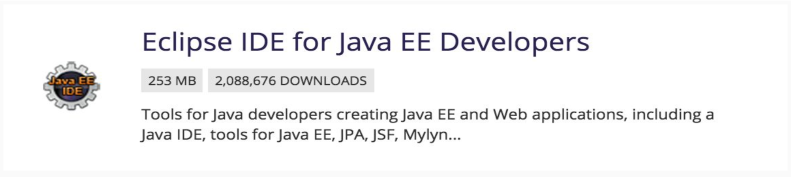
for Magnetics
Revised by: Sihan Ding, Yunny Chung

TECH USED

· Postgres 9.3
· Java v8u45
· Tomcat v8.0.23
· Bootstrap SASS v3.3.4
· Eclipse Luna

ALL SETUP INSTRUCTIONS

1. Install Eclipse Luna EE Edition (https://www.eclipse.org/downloads/)
   - It is really important to download Java EE Developers, not for Java Developers



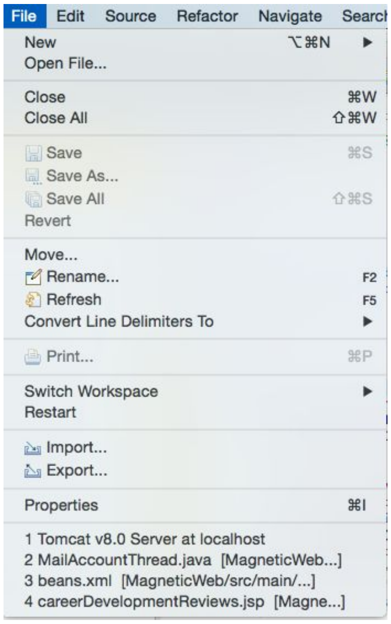
Eclipse IDE for Java EE Developers

253 MB 2,088,676 DOWNLOADS

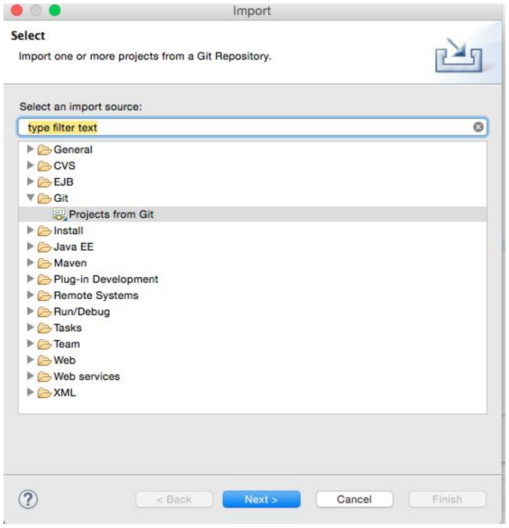
Tools for Java developers creating Java EE and Web applications, including a Java IDE, tools for Java EE, JPA, JSF, Mylyn...

2. Clone the MagneticWeb repo from git （the link will be provided by the company）

*How to clone files in the git into Eclipse
Open your Eclipse, click File -> Import


File Edit Source Refactor Navigate Searc
New ⌥⌘N ▶
Open File...
Close ⌘W
Close All ⇧⌘W
Save ⌘S
Save As...
Save All ⇧⌘S
Revert
Move...
Rename... F2
Refresh F5
Convert Line Delimiters To ▶
Print... ⌘P
Switch Workspace ▶
Restart
Import...
Export...
Properties ⌘I
1 Tomcat v8.0 Server at localhost
2 MailAccountThread.java [MagneticWeb...]
3 beans.xml [MagneticWeb/src/main/...]
4 careerDevelopmentReviews.jsp [Magne...]

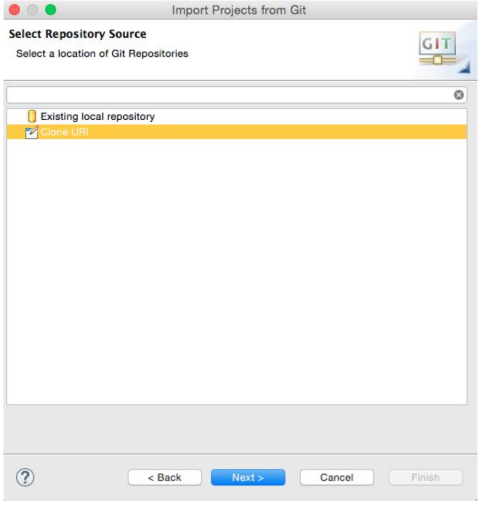
Select ‘Git’-> Projects from Git



Select

Import one or more projects from a Git Repository.

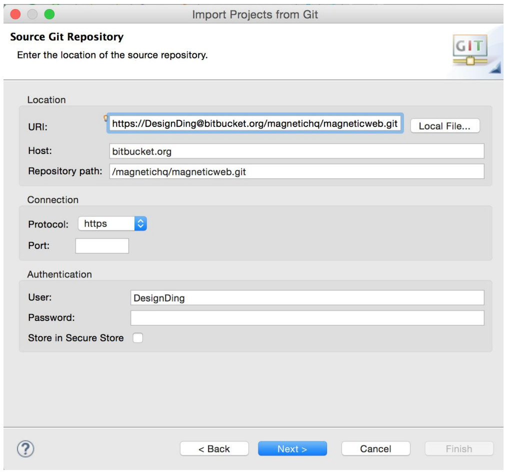
Click ‘Clone URL’



Select Repository Source

Select a location of Git Repositories

Copy the URL of the Git (eclipse will automatically fill the rest of information once you input the url)



Source Git Repository

Enter the location of the source repository.

Location

URI: https://DesignDing@bitbucket.org/magnetichq/magneticweb.git

Host: bitbucket.org

Repository path: /magnetichq/magneticweb.git

Connection

Protocol: https

Port:

Authentication

User: DesignDing

Password:

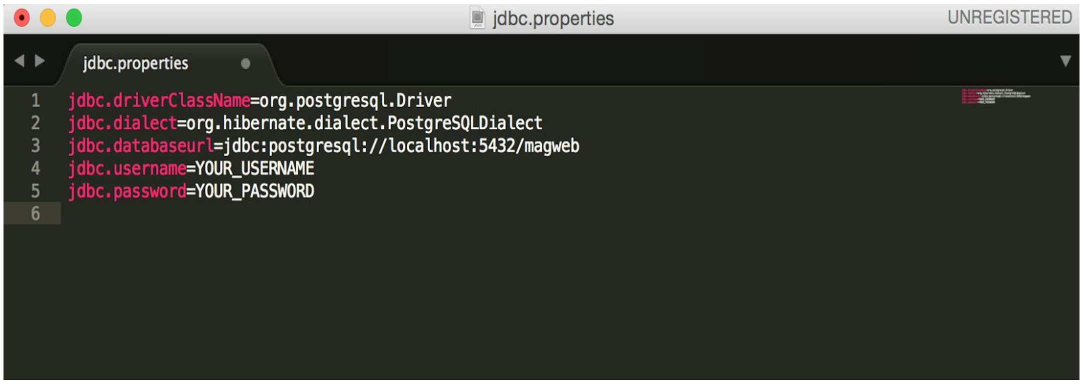
Store in Secure Store □

DONE!

3. Install Postgres 9.3
    - Remember your username and password. You will need them for the next step.

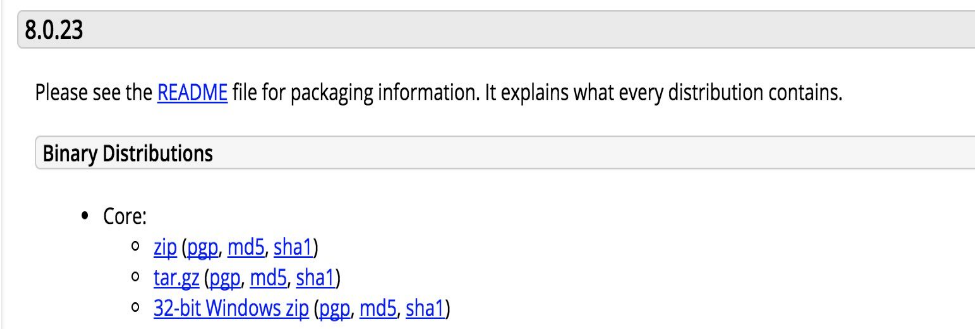
4. Create a database called **magweb** with *username YOUR_USERNAME and password YOUR_PASSWORD* of your Postgres (it's usually the username and password of your computer)

5. Ensure the variables in jdbc.properties file match your database (manually change them if not)
    - find jdbc.properties in magneticweb/src/main/resources/



```
jdbc.driverClassName=org.postgresql.Driver
jdbc.dialect=org.hibernate.dialect.PostgreSQLDialect
jdbc.databaseurl=jdbc:postgresql://localhost:5432/magweb
jdbc.username=YOUR_USERNAME
jdbc.password=YOUR_PASSWORD

```

6. Download Tomcat and install in an appropriate folder
    - Download link for Tomcat: https://tomcat.apache.org/download-80.cgi
    - download the tar.gz file under "core" (you can do zip)



8.0.23

Please see the README file for packaging information. It explains what every distribution contains.

Binary Distributions

- Core:
    - zip (pgp, md5, sha1)
    - tar.gz (pgp, md5, sha1)
    - 32-bit Windows zip (pgp, md5, sha1)

MAC SETUP INSTRUCTIONS

1. Type the following in your terminal

(if you have not installed brew yet) **Install brew** (package manager for Mac)

- run **brew install gradle** (install gradle)
- run **gradle eclipse**
- run **brew install maven** (install maven)
- run **gem install bootstrap-sass** (install bootstrap-sass)

2. Install JDK8 from http://www.oracle.com/technetwork/java/javase/downloads/index.html

3.

- In your terminal type the following
  - **/usr/libexec**
    (the url folder is hidden, to make all the hidden files show up please follow the steps in this website: http://ianlunn.co.uk/articles/quickly-showhide-hidden-files-mac-os-x-mavericks/)
- Check java_home file exists
- If it does, open it by typing **open java_home**
  - You will see **/Library/Java/JavaVirtualMachines/jdk1.8.0_40.jdk/Contents/Home** in the file.

**Need more help: Check where JDK8 installed :
 /usr/libexec/java_home Add JDK dir to eclipse for Java VM -
/Library/Java/JavaVirtualMachines/jdk1.8.0_40.jdk/Contents/Home

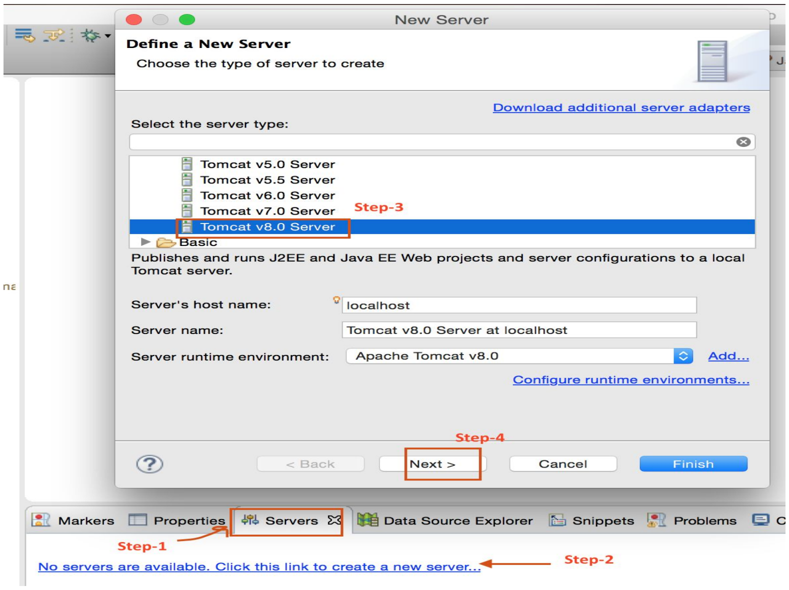
4. Add CATALINA_HOME to your environment variables; type the following lines

- **echo “export CATALINA_HOME=/directory/to/apache-tomcat-8.0.20” >> ~/.bash_profile**
  (*the blue line: /director/to/apache-tomcat-8.0.20 should be where you installed apache-tomcat-8.0.20 or 23 in your local compute,
  for example: Users/YOURUSERNAME/DOWNLOADS/apache-tomcat-8.0.20)

  then**,**
- **source ~/.bash_profile** (this line is for updating the shells with new variables)

5. Add tomcat server to Eclipse by following steps.

- Go to Eclipse
- Click ‘Window’ -> ‘Show View’ -> ‘Server’
- Double Click ‘No server available’ (continued on the following page)
- Click Tomcat v8.0 Server and Next (image available below)



New Server

Define a New Server

Choose the type of server to create

Download additional server adapters

Select the server type:

Tomcat v5.0 Server
Tomcat v5.5 Server
Tomcat v6.0 Server
Tomcat v7.0 Server
Tomcat v8.0 Server
Basic

Publishes and runs J2EE and Java EE Web projects and server configurations to a local Tomcat server.

Server's host name: localhost

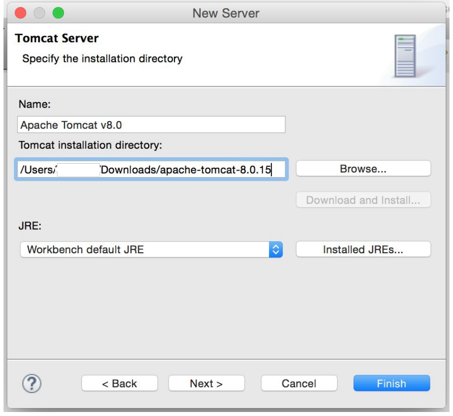
Server name: Tomcat v8.0 Server at localhost

Server runtime environment: Apache Tomcat v8.0

Configure runtime environments...

No servers are available. Click this link to create a new server...

- Select Apache installation Directory and click Finish.



Tomcat Server

Specify the installation directory

Name: Apache Tomcat v8.0

Tomcat installation directory: /Users/[redacted]Downloads/apache-tomcat-8.0.15

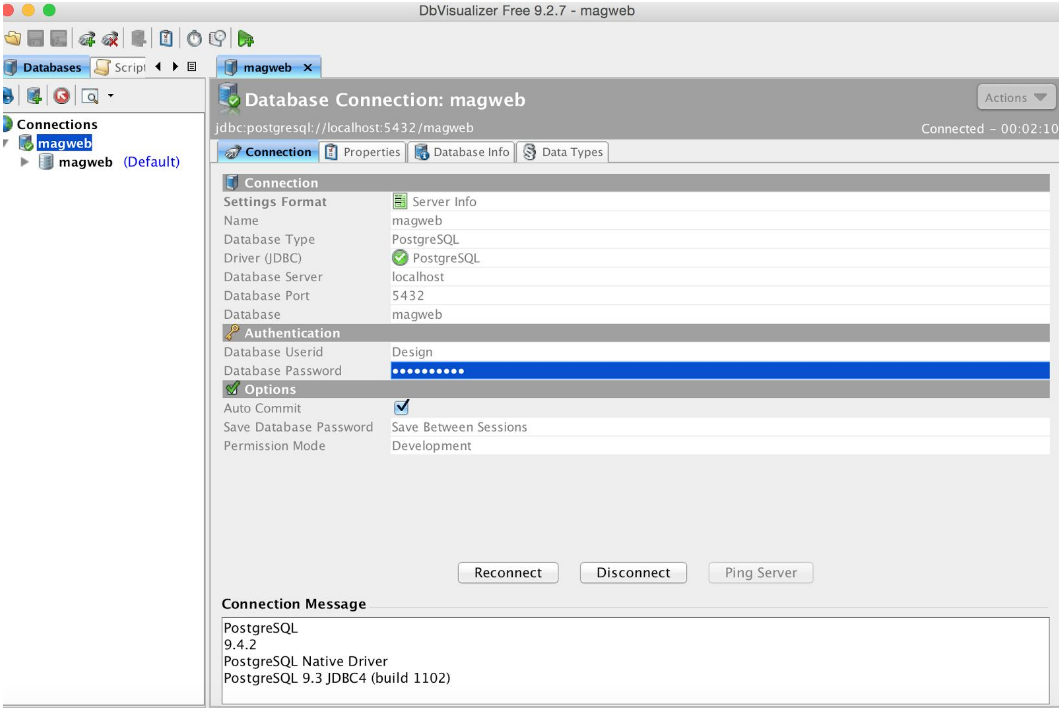
JRE: Workbench default JRE

(images are avilable: http://crunchify.com/step-by-step-guide-to-setup-and-install-apache-tomcat-server-in-eclipse-development-environment-ide/)

INSTRUCTIONS FOR ALL ENVIRONMENTS

- Ensure your server location in Eclipse is using your tomcat installation AND the deploy path is using your tomcat installation/webapps.
- Add **org.apache.jasper.compiler.Parser.STRICT_QUOTE_ESCAPING=false** to catalina.properties in the Server Project in Eclipse

1. Download the DbVisualiser tool and set up, once you finish it should look like the picture below: (link: https://www.dbvis.com/download/)



Database Connection: magweb

jdbc:postgresql://localhost:5432/magweb

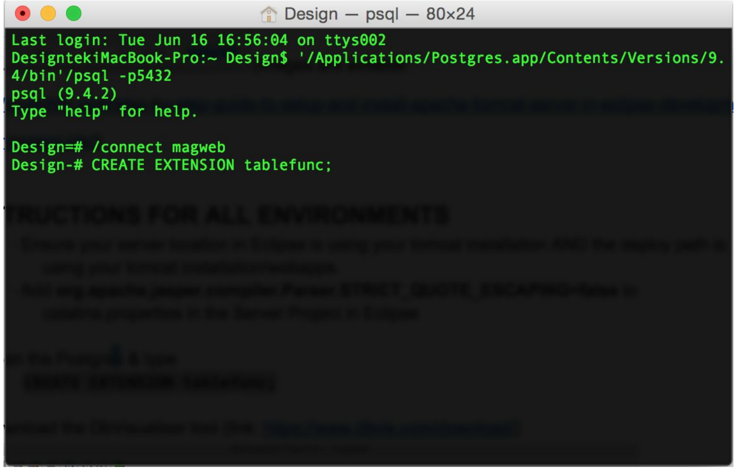
Connected - 00:02:10

Connection	
Settings Format	Server Info
Name	magweb
Database Type	PostgreSQL
Driver (JDBC)	✓ PostgreSQL
Database Server	localhost
Database Port	5432
Database	magweb
Authentication	
Database Userid	Design
Database Password	••••••••••
Options	
Auto Commit	☑
Save Database Password	Save Between Sessions
Permission Mode	Development

Connection Message

```
PostgreSQL
9.4.2
PostgreSQL Native Driver
PostgreSQL 9.3 JDBC4 (build 1102)
```

2.
open your postgres, type in /connect YOUR_DATABASE_NAME, and then type in `CREATE EXTENSION tablefunc;` It installs various functions that return tables.



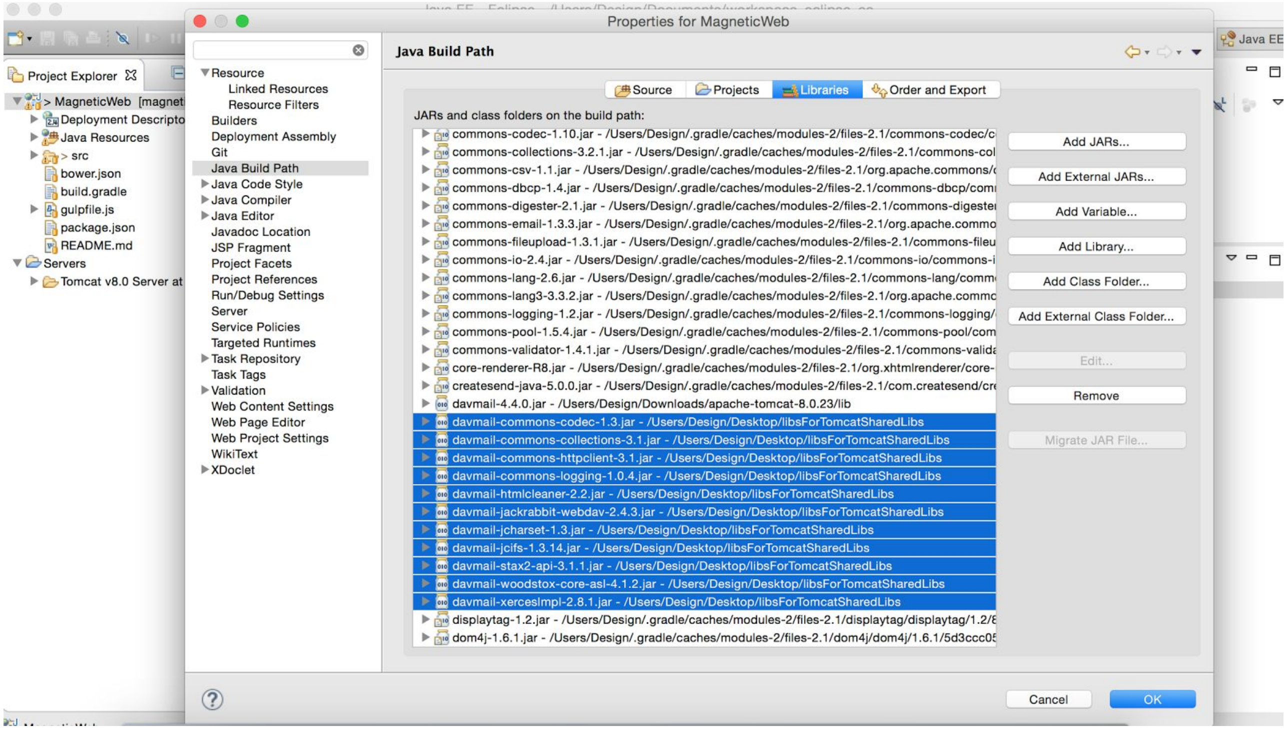
```
Last login: Tue Jun 16 16:56:04 on ttys002
DesigntekiMacBook-Pro:~ Design$ '/Applications/Postgres.app/Contents/Versions/9.
4/bin'/psql -p5432
psql (9.4.2)
Type "help" for help.

Design=# /connect magweb
Design-# CREATE EXTENSION tablefunc;
```

3. To enable SCSS files in Eclipse follow steps in this link: http://stackoverflow.com/questions/7614612/is-there-an-eclipse-editor-for-sasss-scss-files-or-syntax-coloring-plugin

4. Ensure in Eclipse -> Preferences -> General -> Workspace that **Refresh using native hooks and polling** and **Refresh on access** are enabled

5. Get the **davmail-4.4.0.jar file** from the company and add it to the tomcat installation /lib folder

- import other 11 davmail jar files to your library.(get from the company)



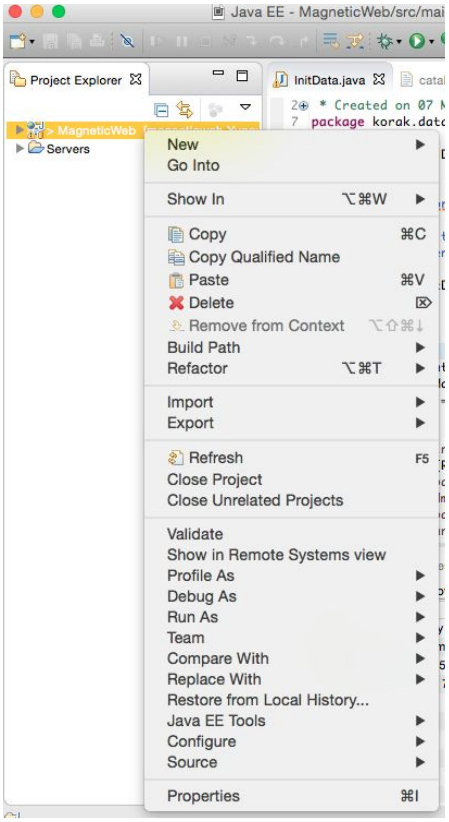
Properties for MagneticWeb

Java Build Path

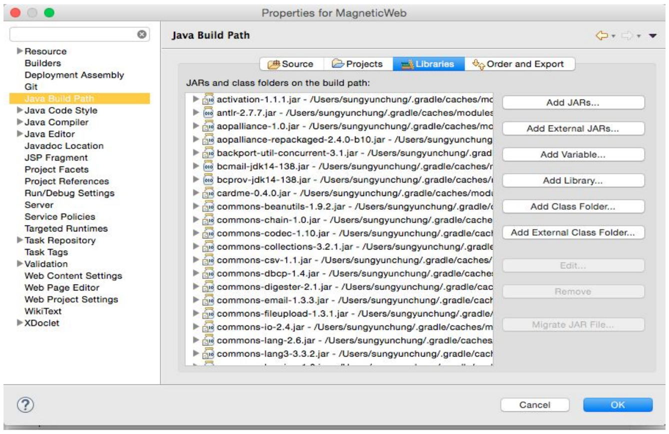
JARs and class folders on the build path:

```
commons-codec-1.10.jar - /Users/Design/.gradle/caches/modules-2/files-2.1/commons-codec/c
commons-collections-3.2.1.jar - /Users/Design/.gradle/caches/modules-2/files-2.1/commons-col
commons-csv-1.1.jar - /Users/Design/.gradle/caches/modules-2/files-2.1/org.apache.commons/c
commons-dbcp-1.4.jar - /Users/Design/.gradle/caches/modules-2/files-2.1/commons-dbcp/com
commons-digester-2.1.jar - /Users/Design/.gradle/caches/modules-2/files-2.1/commons-digeste
commons-email-1.3.3.jar - /Users/Design/.gradle/caches/modules-2/files-2.1/org.apache.commo
commons-fileupload-1.3.1.jar - /Users/Design/.gradle/caches/modules-2/files-2.1/commons-fileu
commons-io-2.4.jar - /Users/Design/.gradle/caches/modules-2/files-2.1/commons-io/commons-i
commons-lang-2.6.jar - /Users/Design/.gradle/caches/modules-2/files-2.1/commons-lang/comm
commons-lang3-3.3.2.jar - /Users/Design/.gradle/caches/modules-2/files-2.1/org.apache.commo
commons-logging-1.2.jar - /Users/Design/.gradle/caches/modules-2/files-2.1/commons-logging/
commons-pool-1.5.4.jar - /Users/Design/.gradle/caches/modules-2/files-2.1/commons-pool/com
commons-validator-1.4.1.jar - /Users/Design/.gradle/caches/modules-2/files-2.1/commons-valida
core-renderer-R8.jar - /Users/Design/.gradle/caches/modules-2/files-2.1/org.xhtmlrenderer/core-
createsend-java-5.0.0.jar - /Users/Design/.gradle/caches/modules-2/files-2.1/com.createsend/cre
davmail-4.4.0.jar - /Users/Design/Downloads/apache-tomcat-8.0.23/lib
davmail-commons-codec-1.3.jar - /Users/Design/Desktop/libsForTomcatSharedLibs
davmail-commons-collections-3.1.jar - /Users/Design/Desktop/libsForTomcatSharedLibs
davmail-commons-httpclient-3.1.jar - /Users/Design/Desktop/libsForTomcatSharedLibs
davmail-commons-logging-1.0.4.jar - /Users/Design/Desktop/libsForTomcatSharedLibs
davmail-htmlcleaner-2.2.jar - /Users/Design/Desktop/libsForTomcatSharedLibs
davmail-jackrabbit-webdav-2.4.3.jar - /Users/Design/Desktop/libsForTomcatSharedLibs
davmail-jcharset-1.3.jar - /Users/Design/Desktop/libsForTomcatSharedLibs
davmail-jcifs-1.3.14.jar - /Users/Design/Desktop/libsForTomcatSharedLibs
davmail-stax2-api-3.1.1.jar - /Users/Design/Desktop/libsForTomcatSharedLibs
davmail-woodstox-core-asl-4.1.2.jar - /Users/Design/Desktop/libsForTomcatSharedLibs
davmail-xercesImpl-2.8.1.jar - /Users/Design/Desktop/libsForTomcatSharedLibs
displaytag-1.2.jar - /Users/Design/.gradle/caches/modules-2/files-2.1/displaytag/displaytag/1.2/8
dom4j-1.6.1.jar - /Users/Design/.gradle/caches/modules-2/files-2.1/dom4j/dom4j/1.6.1/5d3ccc05
```

- You might need to add more external jars if necessary.
- You can add external jars by following steps
    - Right click the project -> Click ‘Properties’



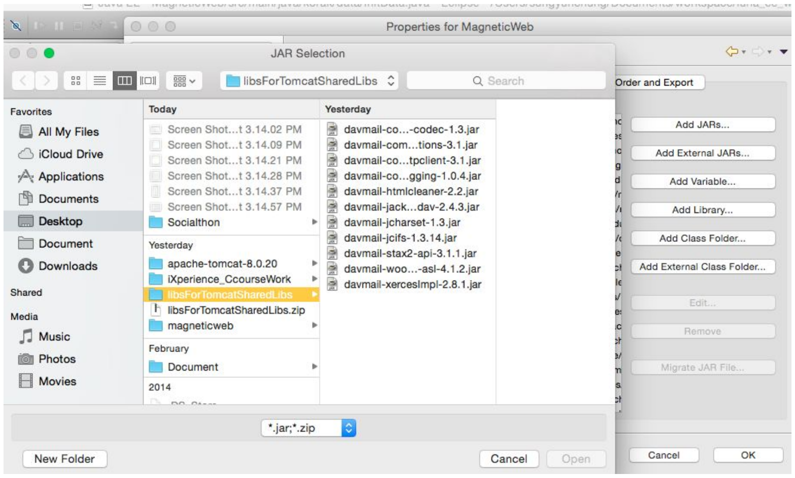
- Click ‘Resources’ -> ‘Java Build Path’ -> ‘Add External JARs’



JARs and class folders on the build path:

activation-1.1.1.jar - /Users/sungyunchung/.gradle/caches/mo
antlr-2.7.7.jar - /Users/sungyunchung/.gradle/caches/modules
aopalliance-1.0.jar - /Users/sungyunchung/.gradle/caches/mo
aopalliance-repackaged-2.4.0-b10.jar - /Users/sungyunchung
backport-util-concurrent-3.1.jar - /Users/sungyunchung/.grad
bcmail-jdk14-138.jar - /Users/sungyunchung/.gradle/caches/r
bcprov-jdk14-138.jar - /Users/sungyunchung/.gradle/caches/
cardme-0.4.0.jar - /Users/sungyunchung/.gradle/caches/modu
commons-beanutils-1.9.2.jar - /Users/sungyunchung/.gradle/
commons-chain-1.0.jar - /Users/sungyunchung/.gradle/cache
commons-codec-1.10.jar - /Users/sungyunchung/.gradle/cac
commons-collections-3.2.1.jar - /Users/sungyunchung/.gradle
commons-csv-1.1.jar - /Users/sungyunchung/.gradle/caches/
commons-dbcp-1.4.jar - /Users/sungyunchung/.gradle/cache
commons-digester-2.1.jar - /Users/sungyunchung/.gradle/cac
commons-email-1.3.3.jar - /Users/sungyunchung/.gradle/cac
commons-fileupload-1.3.1.jar - /Users/sungyunchung/.gradle/
commons-io-2.4.jar - /Users/sungyunchung/.gradle/caches/m
commons-lang-2.6.jar - /Users/sungyunchung/.gradle/caches
commons-lang3-3.3.2.jar - /Users/sungyunchung/.gradle/cac

- Choose any JARS that are necessary to be added



6.
Install latest versions node, click “install” in https://nodejs.org/
This will install both node and npm.
However, npm updates more frequently than node, so you can type:

- sudo npm install npm -g

in your terminal to get the latest version of npm.
(You can also follow the vedio tutori: https://docs.npmjs.com/getting-started/installing-node)

7. In the root directory , run the following (This can be tricky. Ask for help if you were stuck)

① sudo npm install -g bower
② bower install
③ sudo npm install
④ npm install -g gulp
⑤ gulp

-> if your terminal shows you warning and told you “Cannot find module 'FILE_NAME' ”, you can install this missing module by running the following commands repeatedly until the waring ends (the last file should be gulp-watch)

- ❖ **sudo npm install -g “file name”**
- ❖ **gulp**

RUNNING THE WEBAPP

· Run InitData.java in the data package, to populate the database with test data.
· Run **gulp** to compile SCSS and JS
· Run **gulp watch** to automatically compile SCSS and JS when changes are made