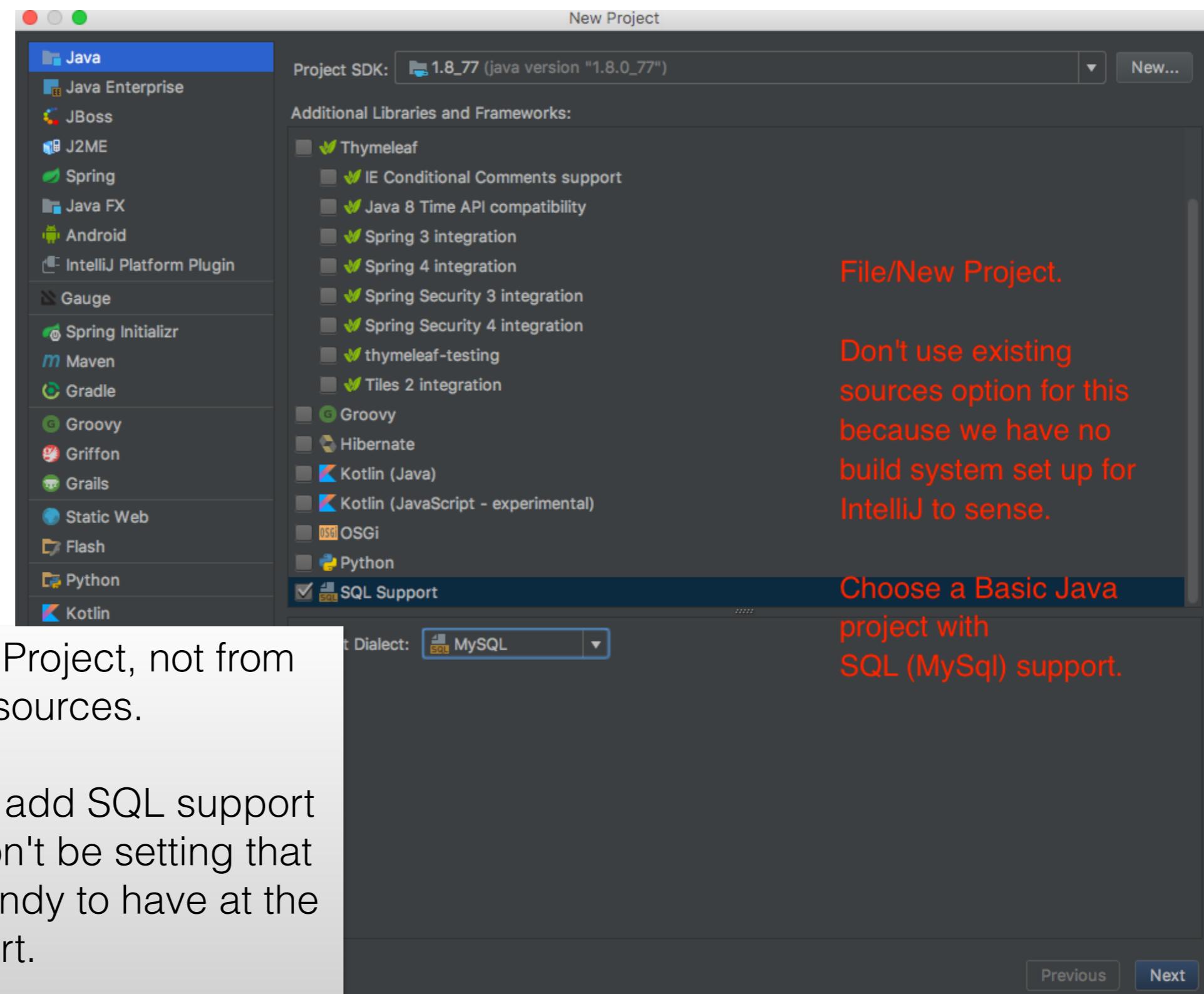


SimpleAffableBean

Setting up IntelliJ, JBoss/Wildfly and MySQL

“git clone git@github.com:nowucca/
SimpleAffableBean.git”

Set up a new IntelliJ project from scratch



Project name: SimpleAffableBean

Project location: /Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/project ...

Setting up the project folder underneath the SimpleAffableBean folder allows us to browse the output of IntelliJ later on.

Take special care to set up the project location, content root and module file location here. You can even put the project directory outside the source tree in general to avoid mixing source and generated files.

▼ More Settings

Module name: SimpleAffableBean

Content root: /Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean ...

Module file location: /Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/project ...

Project format: .ipr (file based) ▾

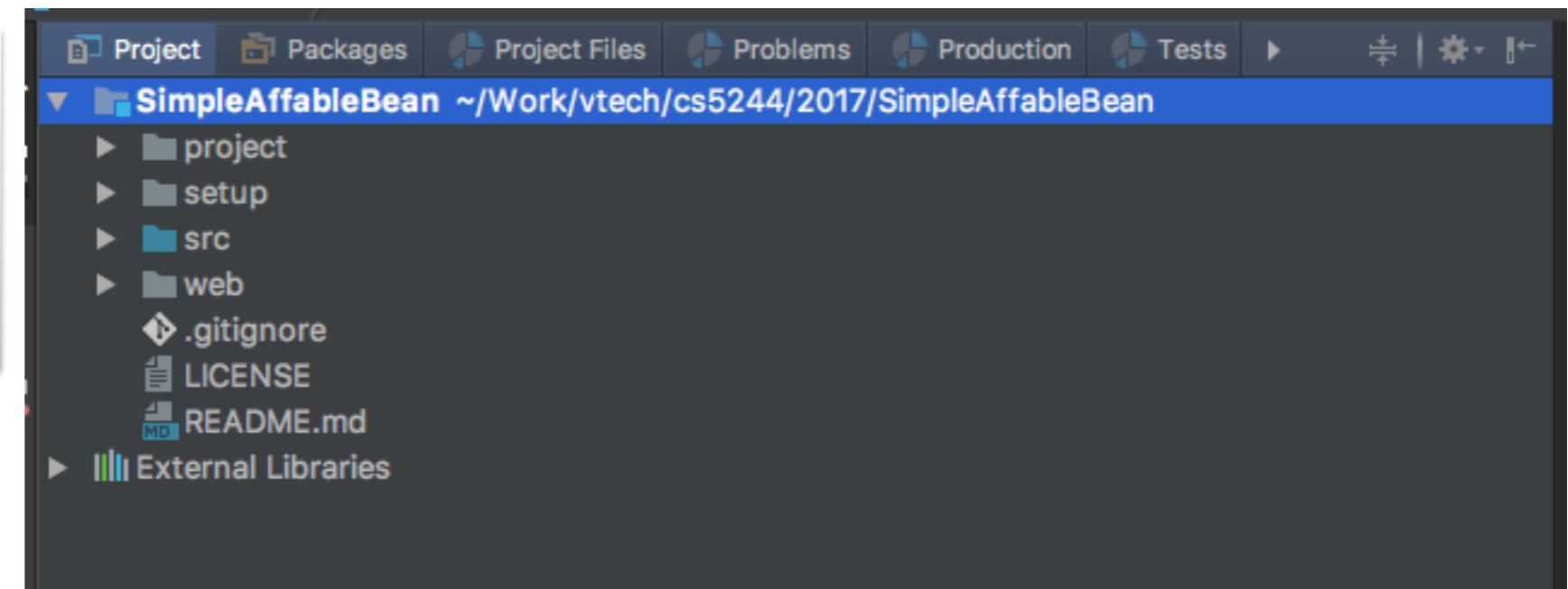
Help

Cancel

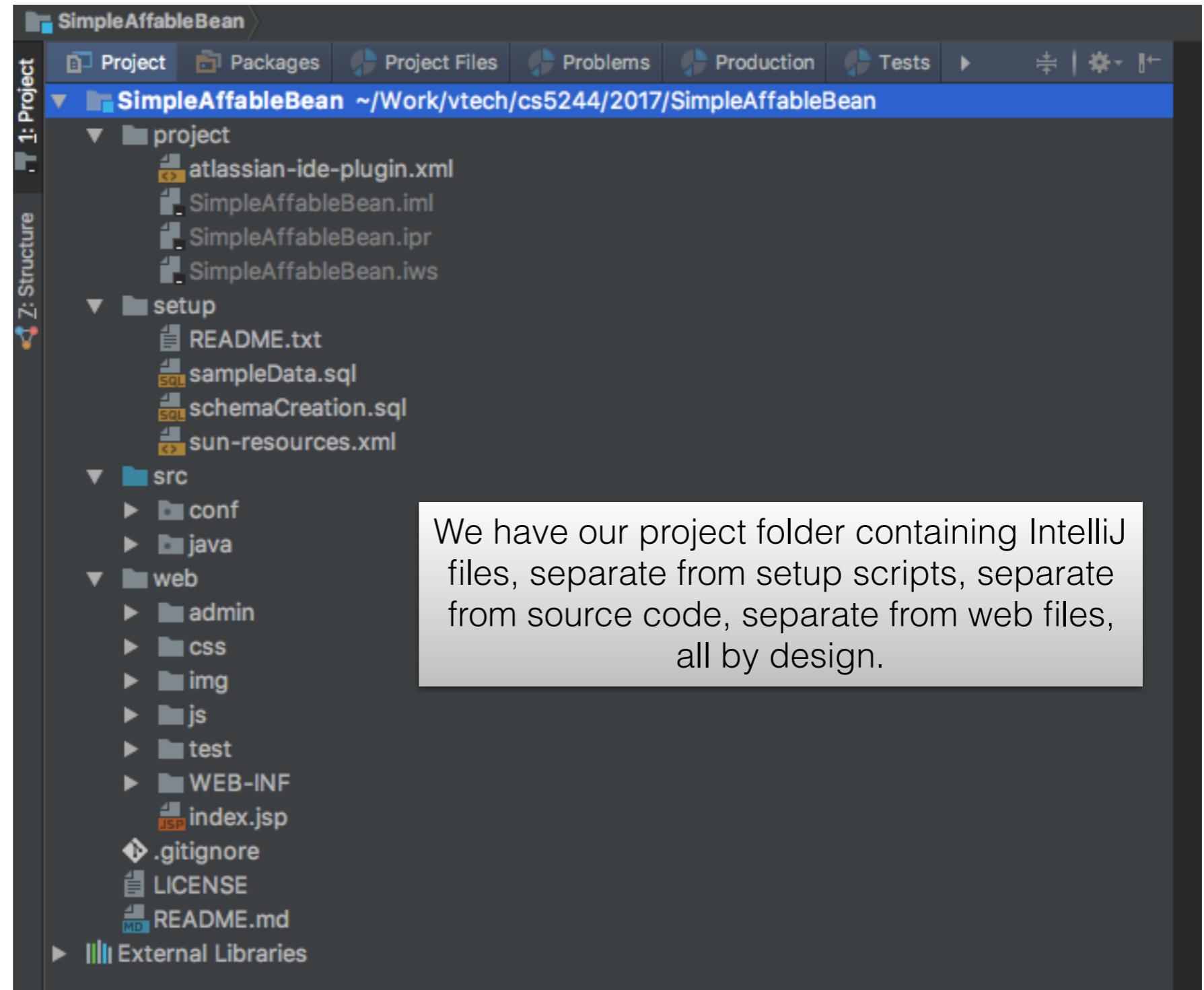
Previous

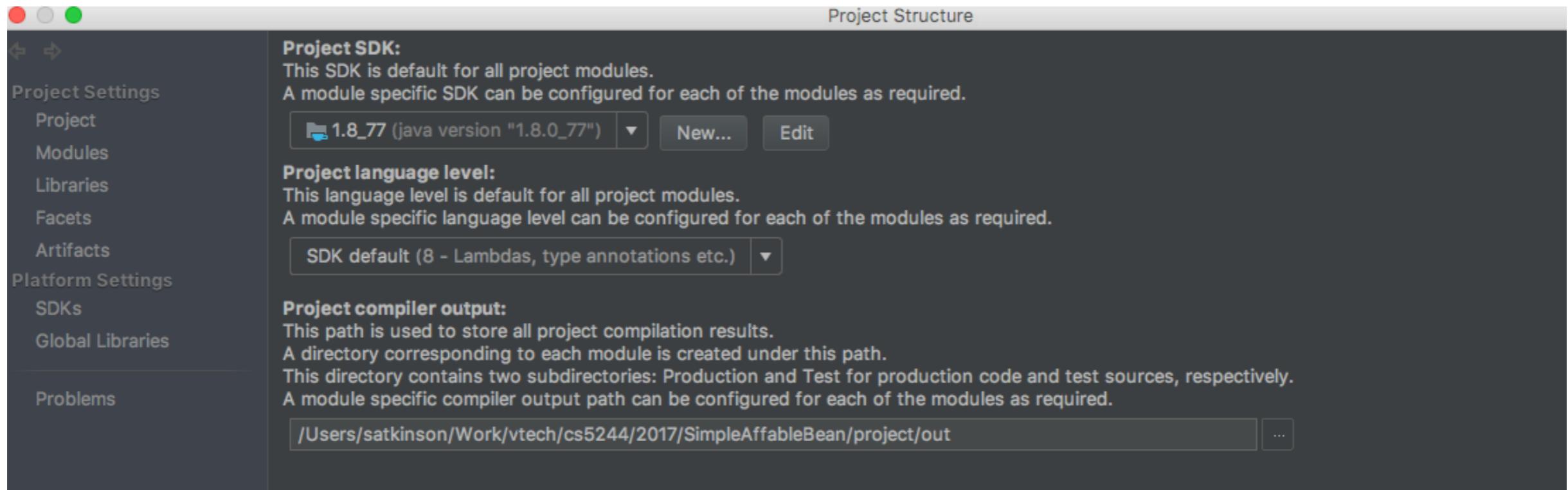
Finish

This shows the project structure after we establish the project.



Opening the project up
to see more.

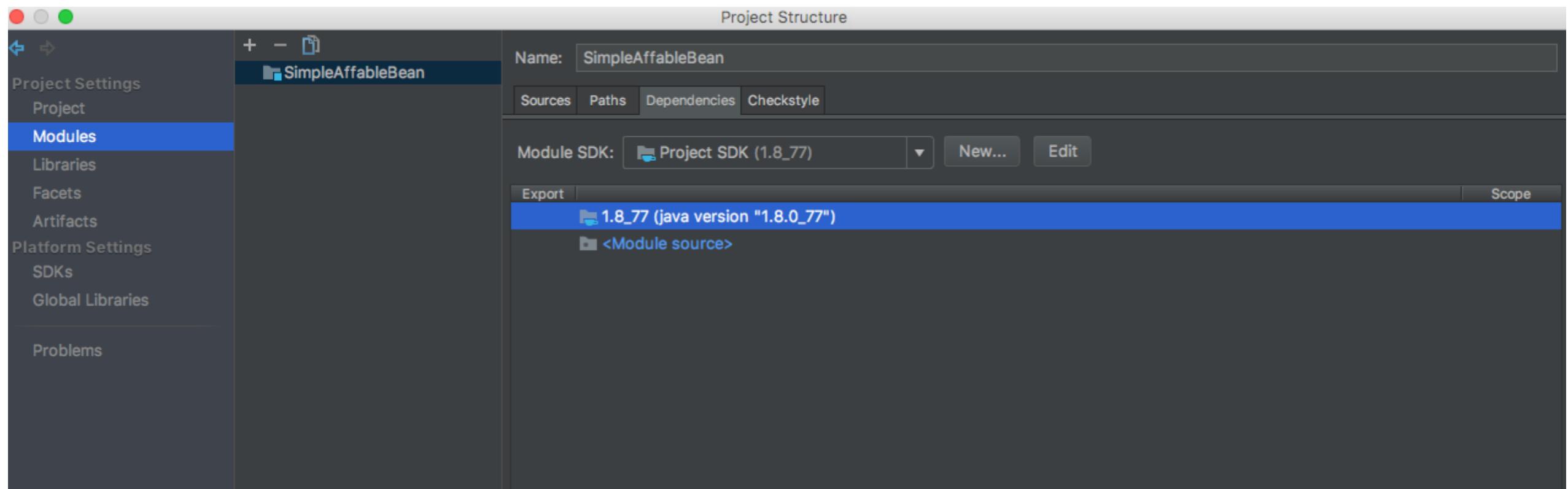




Open up the Project structure using Command-; or the Project Structure button on the toolbar, or the File/Project Structure menu item.

Note here we are using Java 8 with language level 8, so we can use lambdas etc.

The output of the project goes into project/out underneath the content root.



Looking at the sole module "SimpleAffableBean", we need to first tell IntelliJ that it has a Web "facet" to it so that IntelliJ knows that it's for a web application.

To do that, Right-click on the module name, and add a Web facet.

Project Structure

Project Settings

SimpleAffableBean

Web

Deployment Descriptors

Name: Web

Type: Web Module Deployment Descriptor

Path: /Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/project/web2/WEB-INF/web.xml

Note that IntelliJ gets the Web Module Deployment descriptor wrong here.

Add Application Server specific descriptor...

Web Resource Directories

Web Resource Directory: /Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/project/web2

Path Relative to Deployment Root: /

Source Roots

/Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/src

'Web' Facet resources are not included in an artifact

Create Artifact

Cancel Apply Ok

Project Structure

Name: Web

Deployment Descriptors

Type	Path
Web Module Deployment Descriptor	/Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/web/WEB-INF/web.xml

Add Application Server specific descriptor...

Web Resource Directories

Web Resource Directory	Path Relative to Deployment Root
/Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/project/web2	/

Source Roots

/Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/src

 'Web' Facet resources are not included in an artifact

Create Artifact

Cancel Apply OK

Project Settings

SimpleAffableBean

Web

Modules

Libraries

Facets

Artifacts

Platform Settings

SDKs

Global Libraries

Problems

?

Terminal Atlassian 6: TODO

Fix up the descriptor path
to web/WEB-INF/web.xml

Project Settings

Project

Modules

Libraries

Facets

Artifacts

Platform Settings

SDKs

Global Libraries

Problems

Project Structure

Name: Web

Deployment Descriptors

Type	Path
Web Module Deployment Descriptor	/Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/web/WEB-INF/web.xml

+ - ⚒

Add Application Server specific descriptor...

Web Resource Directories

Web Resource Directory	Path Relative to Deployment Root
/Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/web	/

+ - ⚒ ?

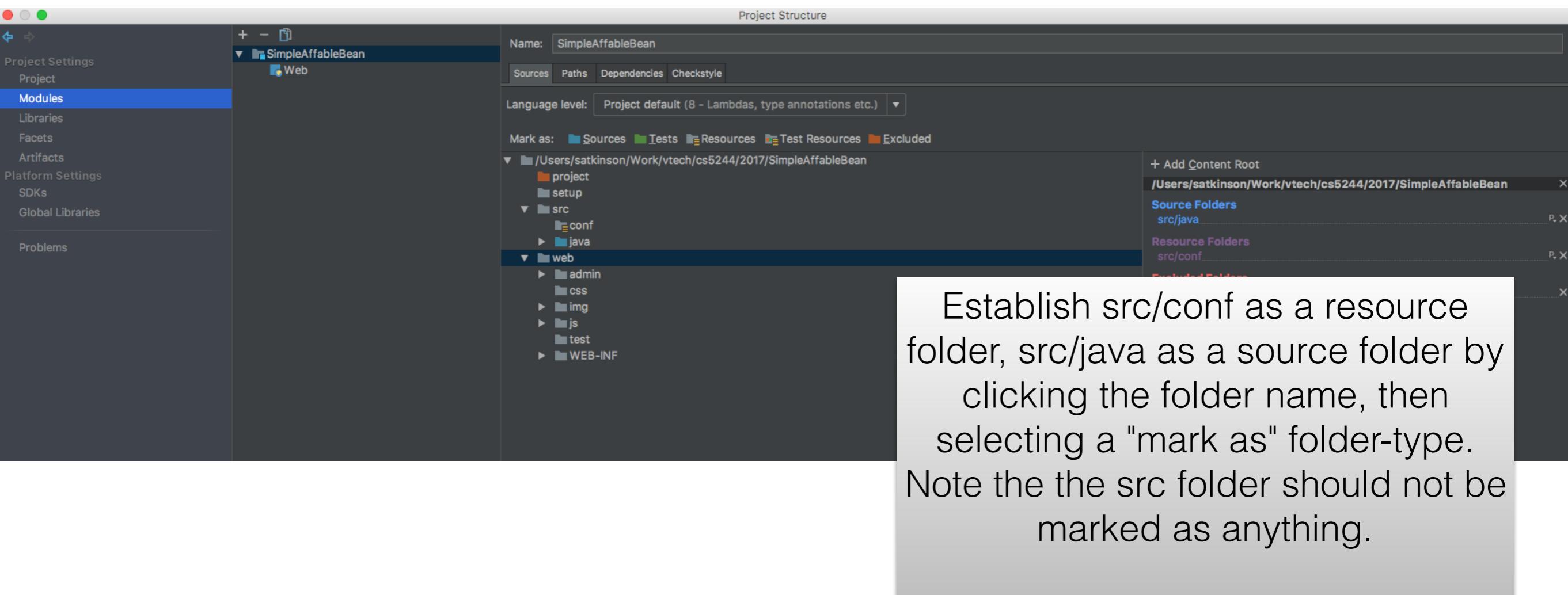
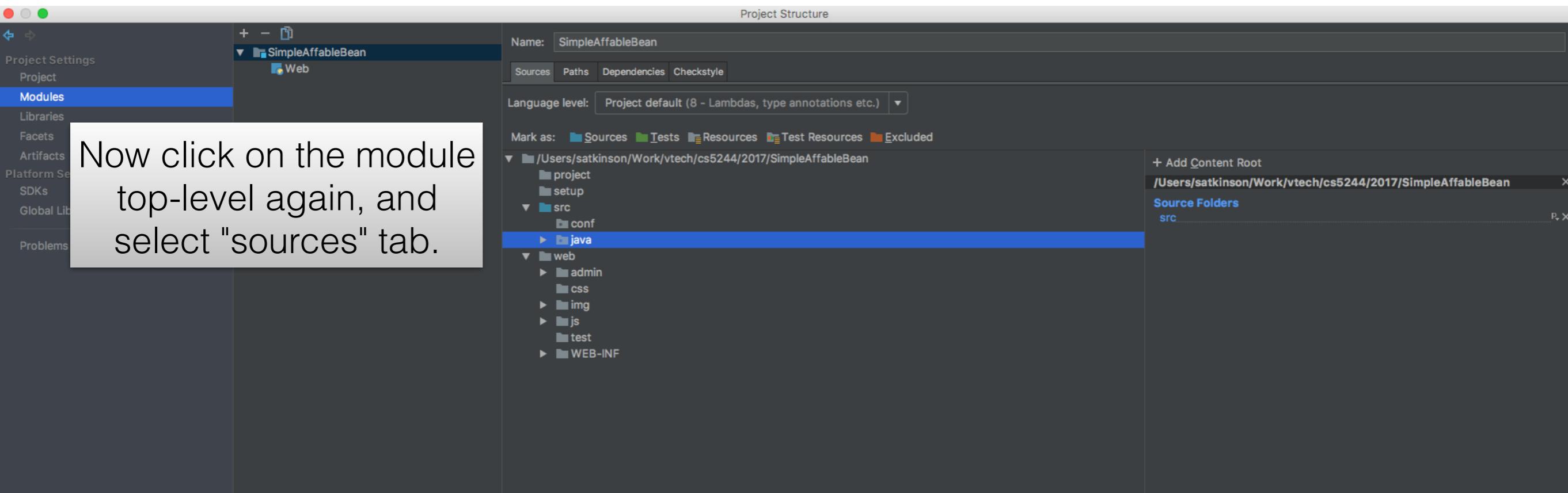
Source Roots

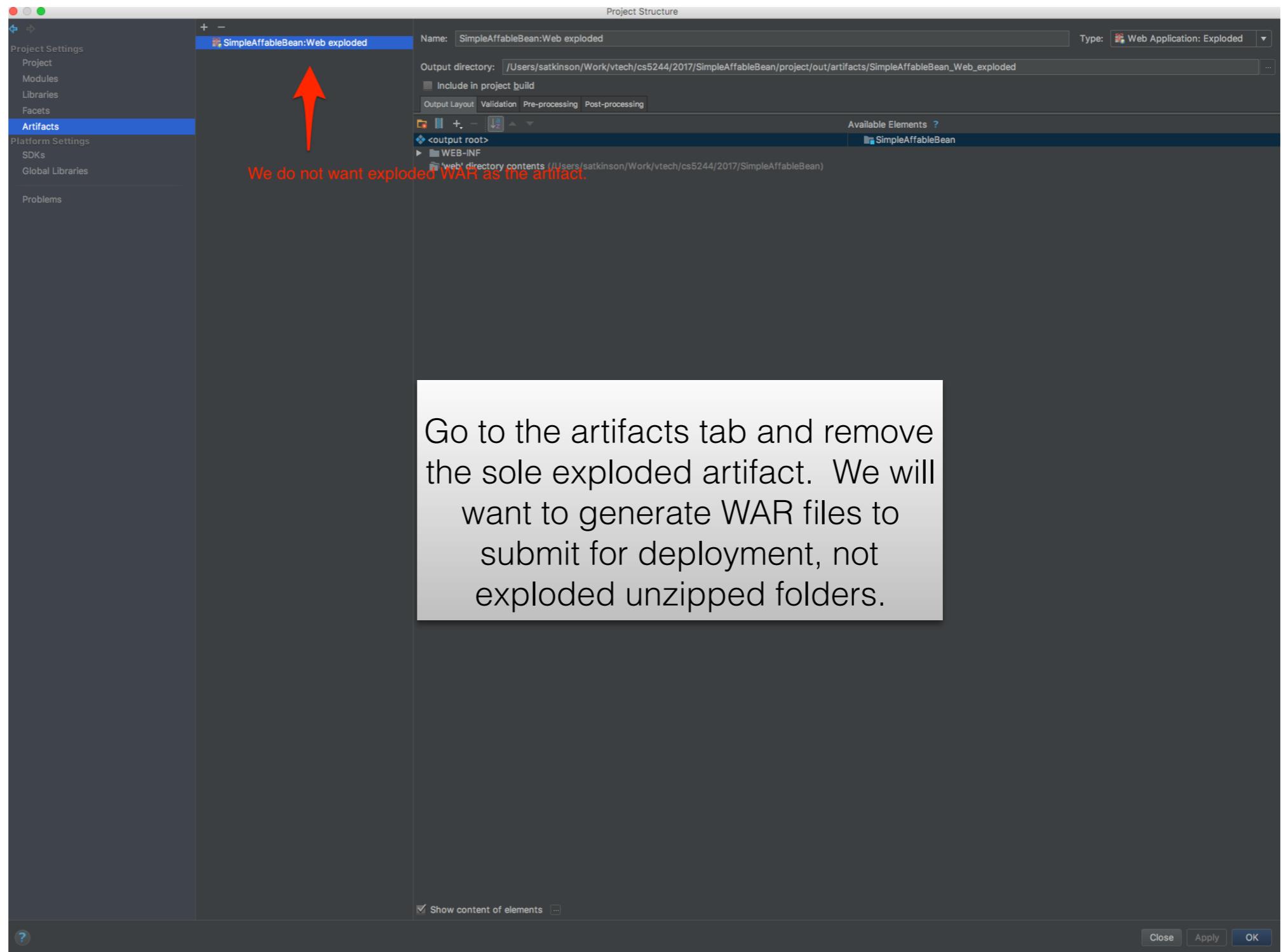
<input checked="" type="checkbox"/> /Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/src

?

Cancel Apply OK

Also tell it that
SimpleAffableBean/web is
the web resource directory





Project Settings

SimpleAffableBean

Name: SimpleAffableBean

Type: Web Application: Archive

Output directory: /Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/project/out/artifacts/SimpleAffableBean

Include in project build

Output Layout Validation Pre-processing Post-processing

SimpleAffableBean.war

Available Elements ?

SimpleAffableBean

'SimpleAffableBean' compile output

Web facet resources

Web Archive has been named and added, not an exploded artifact

META-INF/MANIFEST.MF file not found in 'unnamed.war'

Create Manifest... Use Existing Manifest...

Show content of elements ...

?

Close Apply OK

Project Settings

SimpleAffableBean

Name: SimpleAffableBean

Type: Web Application: A

Output directory: /Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/project/out/artifacts/SimpleAffableBean

Include in project build

Output Layout Validation Pre-processing Post-processing

Available Elements ?

SimpleAffableBean.war

META-INF

MANIFEST.MF (/Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/src/conf)

WEB-INF

classes

META-INF

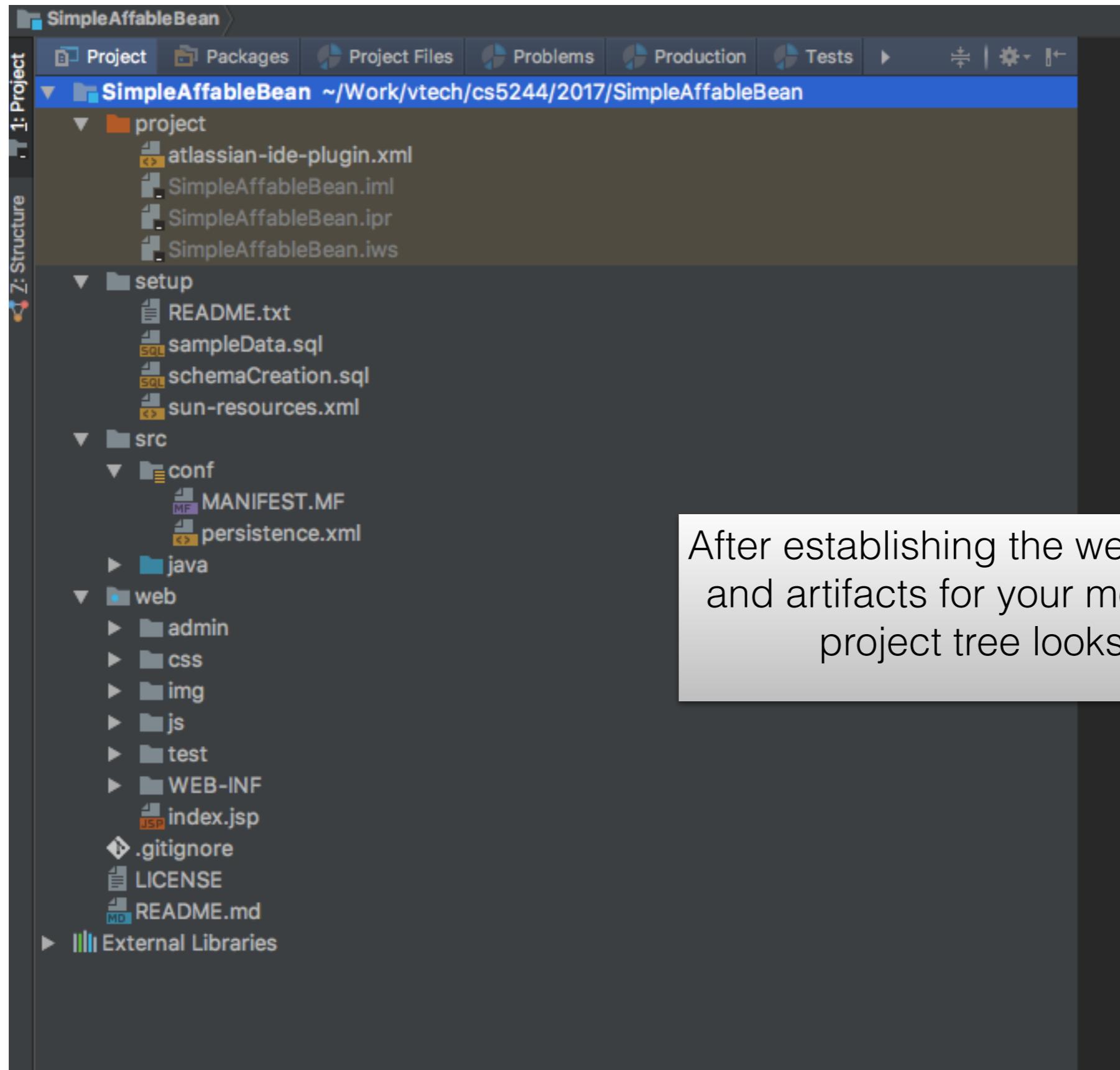
persistence.xml (/Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/src/conf)

web.xml (/Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/web/WEB-INF)

'web' directory contents (/Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean)

Play with the artifact builder.
Click on "Show content of elements" below and try to use the menu items to get your artifact output layout looking like the screenshot. Ignore "validation", and "processing" tabs.

Show content of elements



After establishing the web facet, sources and artifacts for your module, the main project tree looks like this.

JBoss IntelliJ Setup

Establishing a Run
Configuration

See [Jetbrains' JBoss setup pages](#) for more guidance



- ▼ **Defaults**
 - Android App
 - Android Tests
 - Ant Target
 - App Engine server
 - Applet
 - Application
 - Arquillian JUnit
 - Arquillian TestNG
 - Bash
 - Behave
 - bnd Bnd OSGi
 - Chromium Remote
 - Compound
 - dj Django server
 - dj Django tests
 - Firefox Remote
 - Flash App
 - Flash Remote Debug
 - FlexUnit
 - Gauge Execution
 - GlassFish Server
 - Gradle
 - Grails
 - Griffon
 - Groovy
 - Grunt.js
 - Gulp.js
 - J2ME
 - JAR Application
 - JBoss Server
 - JUnit
 - Java Scratch
 - JavaScript Debug
 - Kotlin
 - Kotlin (JavaScript - experimental)
 - Kotlin script
 - Lettuce
 - Maven
 - NW.js
 - OSGi Bundles
 - Plugin
 - Protractor
 - Pyramid server
 - Python
 - Python Remote Debug
 - Python tests
 - React Native

Press the + button to create a new configuration based on default settings

Choose Run/Edit
Configurations to get this
dialog

Confirm rerun with process termination

Temporary configurations limit:



Cancel Apply OK

Server Deployment Logs Code Coverage Startup/Connection

2) Add a local JBoss Server

- + Android App
- + Android Tests
- + Ant Target
- + App Engine server
- + Applet
- + Application
- + Arquillian JUnit
- + Arquillian TestNG
- + Bash
- + Behave
- + Bnd OSGi
- + Chromium Remote
- + Compound
- + Django server
- + Django tests
- + Firefox Remote
- + Flash App
- + Flash Remote Debug
- + FlexUnit
- + Gauge Execution
- + GlassFish Server
- + Gradle
- + Grails
- + Griffon
- + Groovy
- + Grunt.js
- + Gulp.js
- + J2ME
- + JAR Application
- + JBoss Server
 - + Local
 - + Remote
- + JUnit
- + Java Scratch
- + JavaScript Debug
- + Kotlin
- + Kotlin (JavaScript - experimental)
- + Kotlin script
- + Lettuce
- + Maven
- + NW.js
- + OSGi Bundles
- + Plugin
- + Protractor
- + Pyramid server
- + Python
- + Python Remote Debug

Application server:

Open browser: After launch ... with JavaScript debugger

<http://localhost:8080/>

VM options:

On 'Update' action: Show dialog

JRE: Default (1.8_77 - project SDK)

JBoss Server Settings

Username:

Password:

1) Add this to defaults

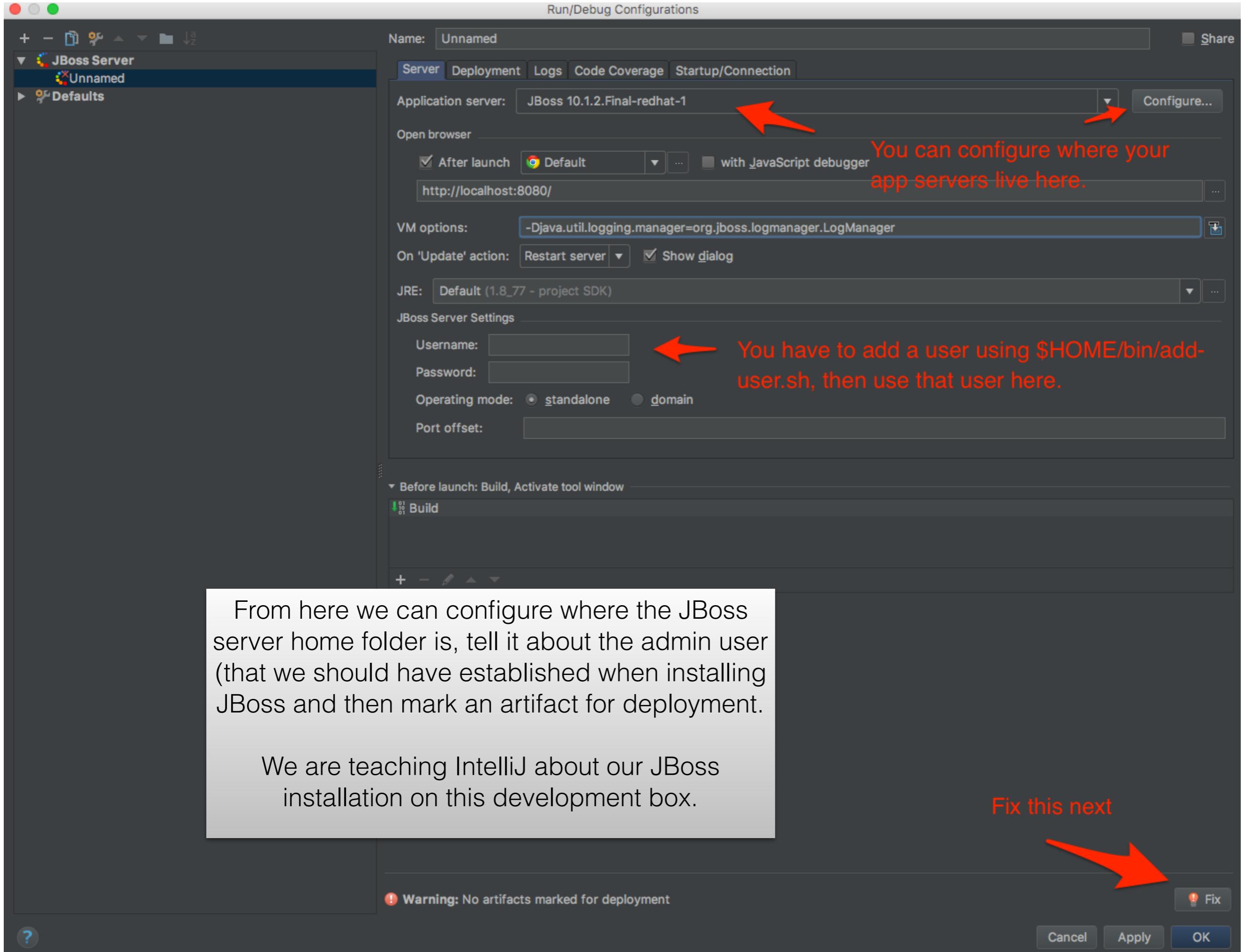
▼ Before launch: Build, Activate tool window

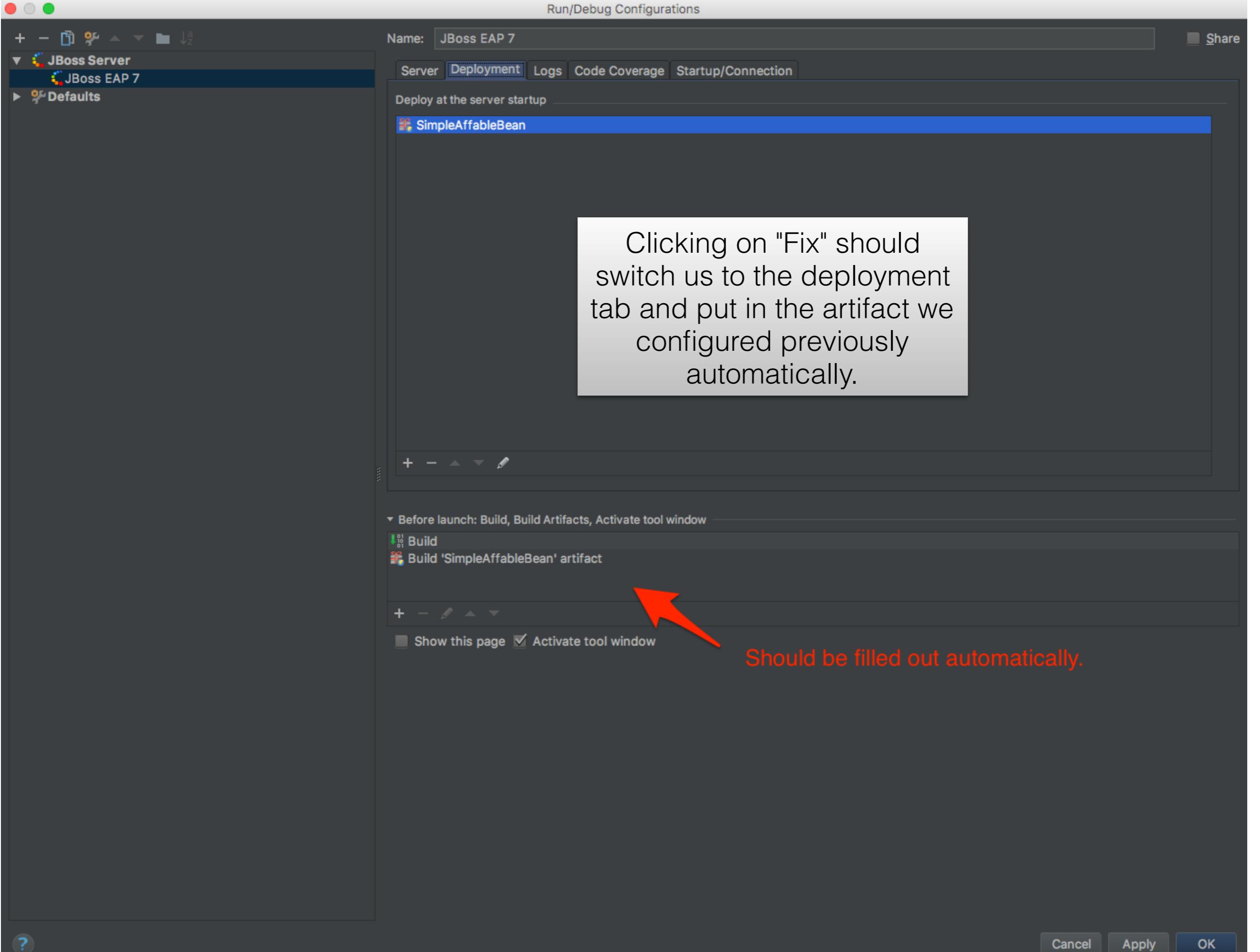
Build

+ -
 Show this page Activate tool window

You can adjust global defaults. JBoss needs to always be told about a logging manager, so we edit the default. Then hit the "+" to make a new configuration from the default.

Cancel Apply OK





Run \$JBOSS_HOME/bin/standalone.sh to start the server from the command line so we can access the admin screens.

The screenshot shows the Red Hat JBoss Enterprise Application Platform Admin Console. At the top, there's a navigation bar with icons for back, forward, and search, followed by a 'RED HAT' logo and a 'Home' link. Below the navigation bar, there are two tabs: 'Access Control' and 'Patching'. The main content area is titled 'Red Hat JBoss Enterprise Application Platform' and 'New to JBoss'. It features several sections: 'Deployments' (with 'Create a Datasource' highlighted by a red arrow), 'Configuration' (with 'Create a Datasource' also present), 'Runtime' (with 'Monitor the Server'), and 'Access Control' (with 'Assign User Roles'). Each section has a brief description and a 'Start' button.

Red Hat JBoss Enterprise Application Platform

New to JBoss

Deployments

Add and manage deployments

[Deploy an Application](#) | [Start](#)

Deploy an application to the server

1. Use the 'Add Deployment' wizard to deploy the application
2. Enable the deployment

Configuration

Configure subsystem settings

[Create a Datasource](#) | [Start](#)

Define a datasource to be used by deployed applications. The proper JNDI name will be generated and registered.

1. Select the Datasources subsystem
2. Add a Non-XA or XA datasource
3. Use the 'Create Datasource' wizard to configure the datasource settings

[Create a JMS Queue](#) | [Start](#)

Runtime

Monitor server status

[Monitor the Server](#) | [Start](#)

Access Control

Manage user and group permissions for management operations

[Assign User Roles](#) | [Start](#)

The screenshot shows the MySQL Workbench interface. On the left, the 'SCHEMAS' tree view has 'afb1' expanded, with several tables like 'category', 'customer', and 'product' listed under it. A red arrow points from the text in the top right towards this schema selection. The top navigation bar shows 'Local instance 3306'.

I created an "afb1" database and user using MySQL workbench.

The screenshot shows the 'Administration - Users and Privileges' dialog. It lists existing users: 'afb1' (from host '%'), 'cs5244' (from host '%'), 'mysql.sys' (from host 'localhost'), and 'root' (from host 'localhost'). A new user 'afb1' is being created with the following details:

User	From Host
afb1	%

Red annotations highlight the 'User' field as 'afb1' and the 'From Host' field as '%'. A large red arrow points from the text in the top right towards the 'User' field of the new user entry.

user afb1/afb1 defined and given full access to the database

localhost:9990/console/App.html#profile/ds-finder

RED HAT JBOSS ENTERPRISE APPLICATION PLATFORM

Home Deployments Configuration Runtime Access Control Patching

Configuration	Subsystem (28)	Type	Datasource
Subsystems >	Q []	Non-XA >	ExampleDS
Interfaces	JCA	XA >	tinybookstore
Socket Binding			
Paths			
System Properties			
Mail			
Transactions			
EJB 3			
EE			
JPA			
Batch			
IO			
Logging			
Deployment Scann			
JMX			
Remoting			
Infinispan			
Security			
Web Services			
Web/HTTP - Undertow	>		
BeanValidation			
Jaxrs			
Jdr			

Create Datasource

Step 1/3: Datasource Attributes

Name*: jdbc/affablebean

JNDI Name*: java:/jdbc/affablebean

Required fields are marked with an asterisk (*)

Need Help?

Create Datasource

Step 2/3: JDBC Driver

Select one of the installed JDBC driver. Don't see your driver? Please make sure it's deployed as a module and properly registered.

Specify Driver Detected Driver

Name
mysql
h2

1-2 of 2

Create Datasource

Step 3/3: Connection Settings

Connection URL*: jdbc:mysql://localhost:3306/afb

Username: afb1

Password: ***

Security Domain:

Required fields are marked with an asterisk (*).

Cancel << Back Next >>

Custom

H2 Datasource

PostgreSQL Datasource

MySQL Datasource

Oracle Datasource

Microsoft SQLServer Datasource

IBM DB2 Datasource

Sybase Datasource

Cancel << Back Next >>

localhost:9990/console/App.html#profile/ds-finder/datasources;name=jdbc%25C0affablebean

RED HAT JBOSS' ENTERPRISE APPLICATION PLATFORM

« Back Configuration: Subsystems > Subsystem: Datasources > Type: Non-XA > **Datasource: jdbc/affablebean**

DATASOURCES

JDBC datasource 'jdbc/affablebean' (enabled)

JDBC datasource configurations.

Disable

Attributes Connection Pool Security Properties Validation Timeouts Statements

Edit

Connection URL: jdbc:mysql://localhost:3306/afb1

New Connection Sql:

Transaction Isolation:

Use JTA?: true

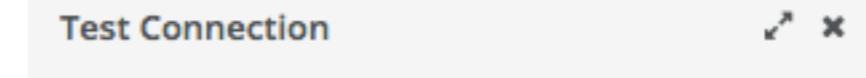
Use CCM?: true

Test Connection Need Help?



Test Connection

Need Help?



i Successfully created JDBC connection.

Successfully connected to database
jdbc/affablebean.

OK

You MUST see this success message before you go on, otherwise it will not work.

Fixing Library Dependencies and Compile Errors

Also adding a JPA facet
before we forget....

The screenshot shows the JBoss Tools IDE interface. On the left is the Project Explorer (1: Project) and Problems (2: Structure) panes. The Project Explorer lists the project structure, including setup, src, web, and various Java and XML files. The src folder contains sub-folders like cart, controller, entity, filter, resources, and session, which further contain facade classes like AbstractFacade, CategoryFacade, CustomerFacade, CustomerOrderFacade, OrderedProductFacade, OrderManager, and ProductFacade. The web folder contains admin, css, img, js, test, WEB-INF (with jspf and view sub-folders), and various JSP files like cart.jsp, category.jsp, checkout.jsp, confirmation.jsp, sun-web.xml, and web.xml. The Problems pane shows a list of errors and warnings, primarily related to the OrderManager.java file.

```
8 package session;
9
10 import cart.*;
11 import entity.*;
12 import java.math.BigDecimal;
13 import java.util.ArrayList;
14 import java.util.HashMap;
15 import java.util.List;
16 import java.util.Map;
17 import java.util.Random;
18 import javax.annotation.Resource;
19 import javax.ejb.EJB;
20 import javax.ejb.SessionContext;
21 import javax.ejb.Stateless;
22 import javax.ejb.TransactionAttribute;
23 import javax.ejb.TransactionAttributeType;
24 import javax.ejb.TransactionManagement;
25 import javax.ejb.TransactionManagementType;
26 import javax.persistence.EntityManager;
27 import javax.persistence.PersistenceContext;
28
29 /**
30  * 
31  * @author tgiunipero
32  */
33 @Stateless
34 @TransactionManagement(TransactionManagementType.CONTAINER)
35 public class OrderManager {
36
37     @PersistenceContext(unitName = "AffableBeanPU")
38     private EntityManager em;
39     @Resource
40     private SessionContext context;
41     @EJB
42 }
```

Need to add libraries to project to get this to work. The compile/build failed.

Information: java: Errors occurred while compiling module 'SimpleAffableBean'
Information: javac 1.8.0_77 was used to compile java sources
Information: 1/24/17, 10:14 PM - Compilation completed with 100 errors and 0 warnings in 2s 258ms

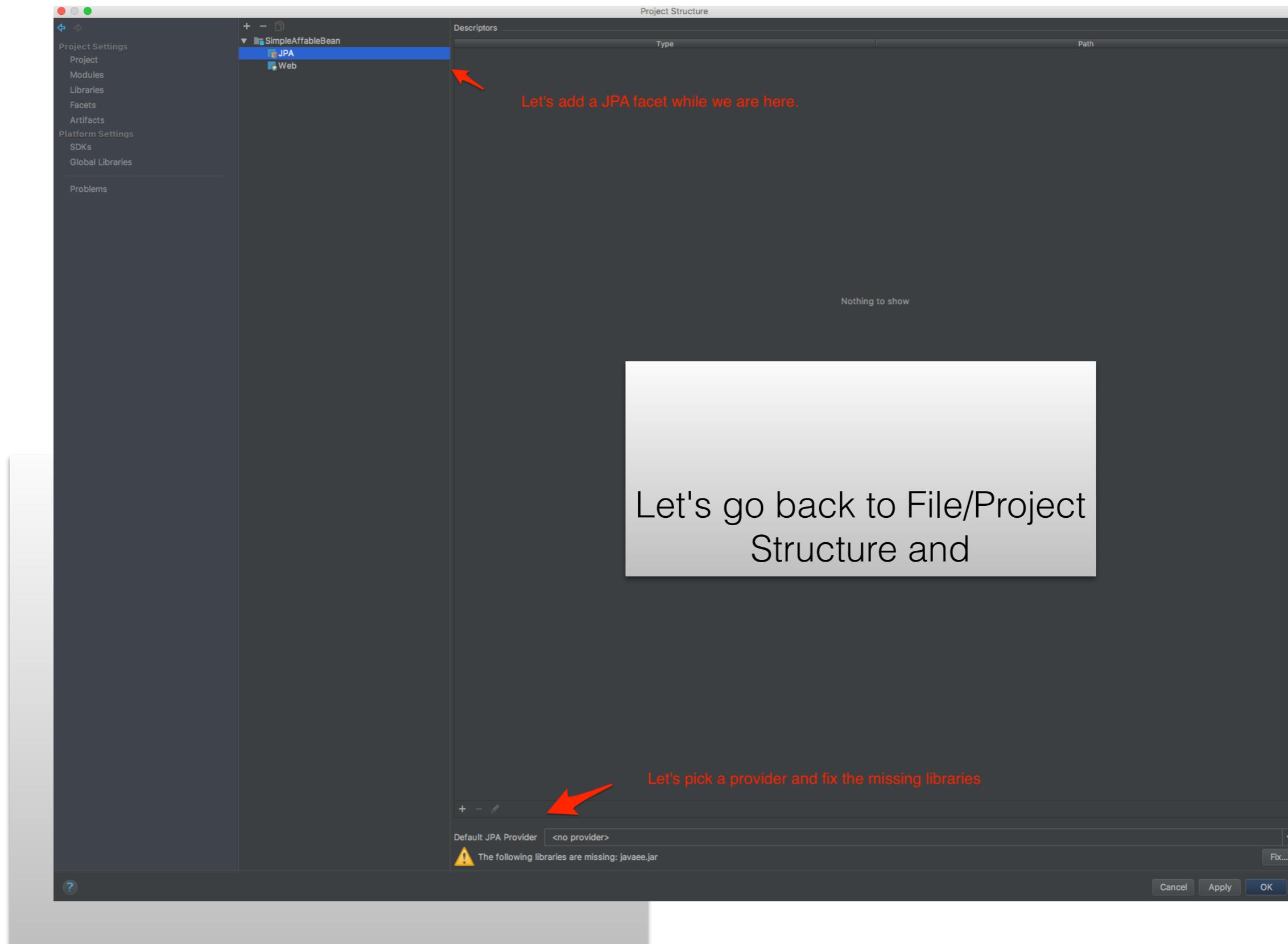
/Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/src/java/session/OrderManager.java

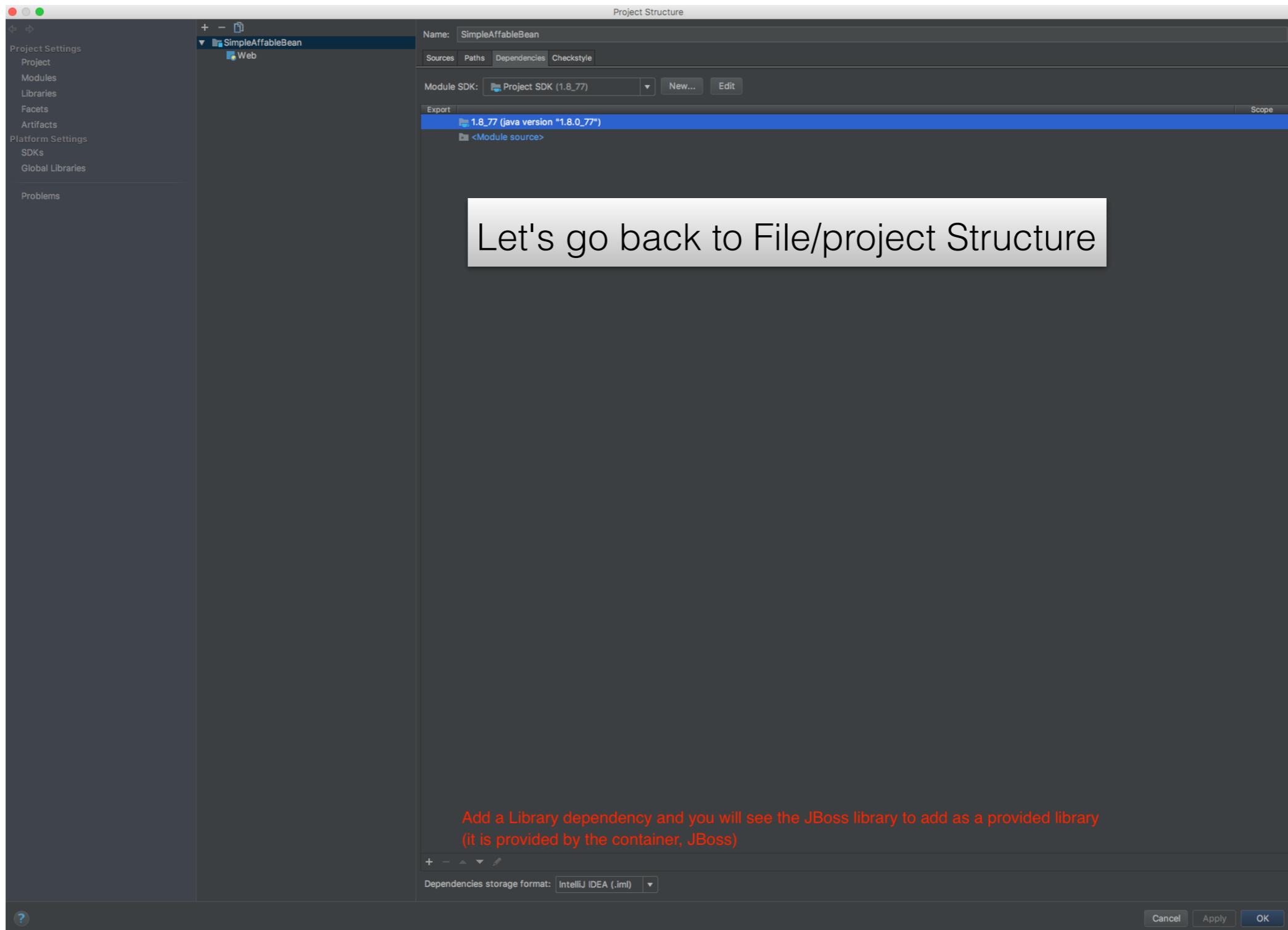
- Error:(20, 17) java: package javax.ejb does not exist
- Error:(21, 17) java: package javax.ejb does not exist
- Error:(22, 17) java: package javax.ejb does not exist
- Error:(23, 17) java: package javax.ejb does not exist
- Error:(24, 17) java: package javax.ejb does not exist
- Error:(25, 17) java: package javax.ejb does not exist
- Error:(26, 17) java: package javax.ejb does not exist
- Error:(27, 25) java: package javax.persistence does not exist
- Error:(28, 25) java: package javax.persistence does not exist
- Error:(34, 2) java: cannot find symbol
symbol: class Stateless
- Error:(35, 2) java: cannot find symbol
symbol: class TransactionManagement
- Error:(39, 13) java: cannot find symbol
symbol: class EntityManager
- Error:(41, 13) java: cannot find symbol
symbol: class SessionContext

I clicked on the Run/Run JBoss Local menu item.

The project should build and then deploy automatically to JBoss.

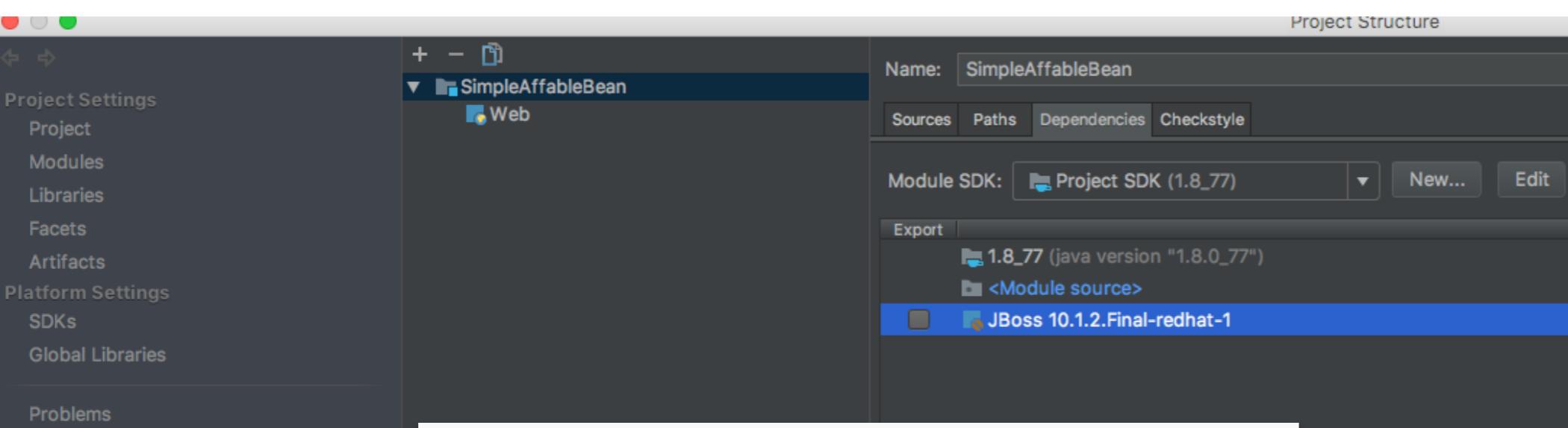
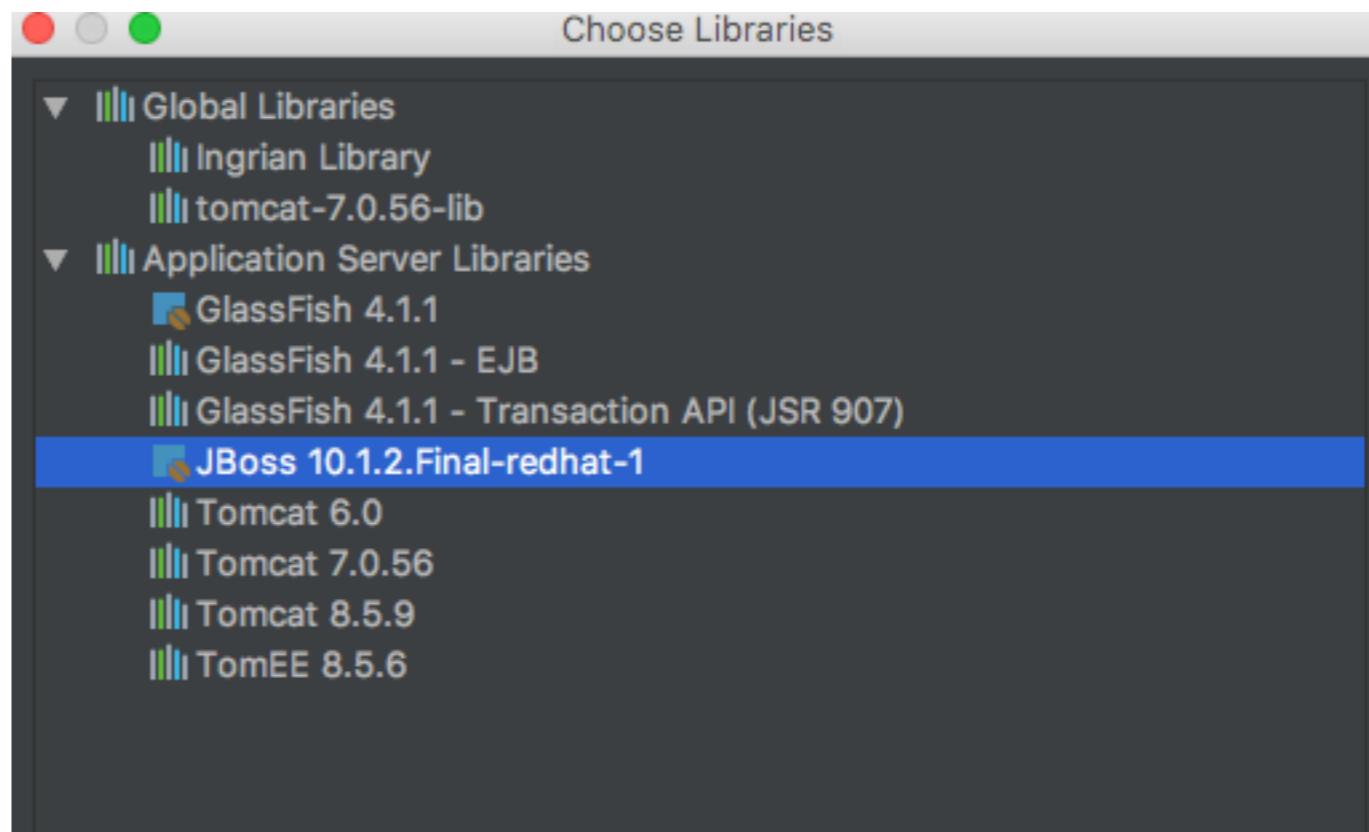
However, it fails to compile due to lack of libraries.





Let's hit the Dependencies tab
and the "+" down the bottom and
pick "Libraries" option.

We can then choose our configured application server libraries.



And the library appears as a dependency now.

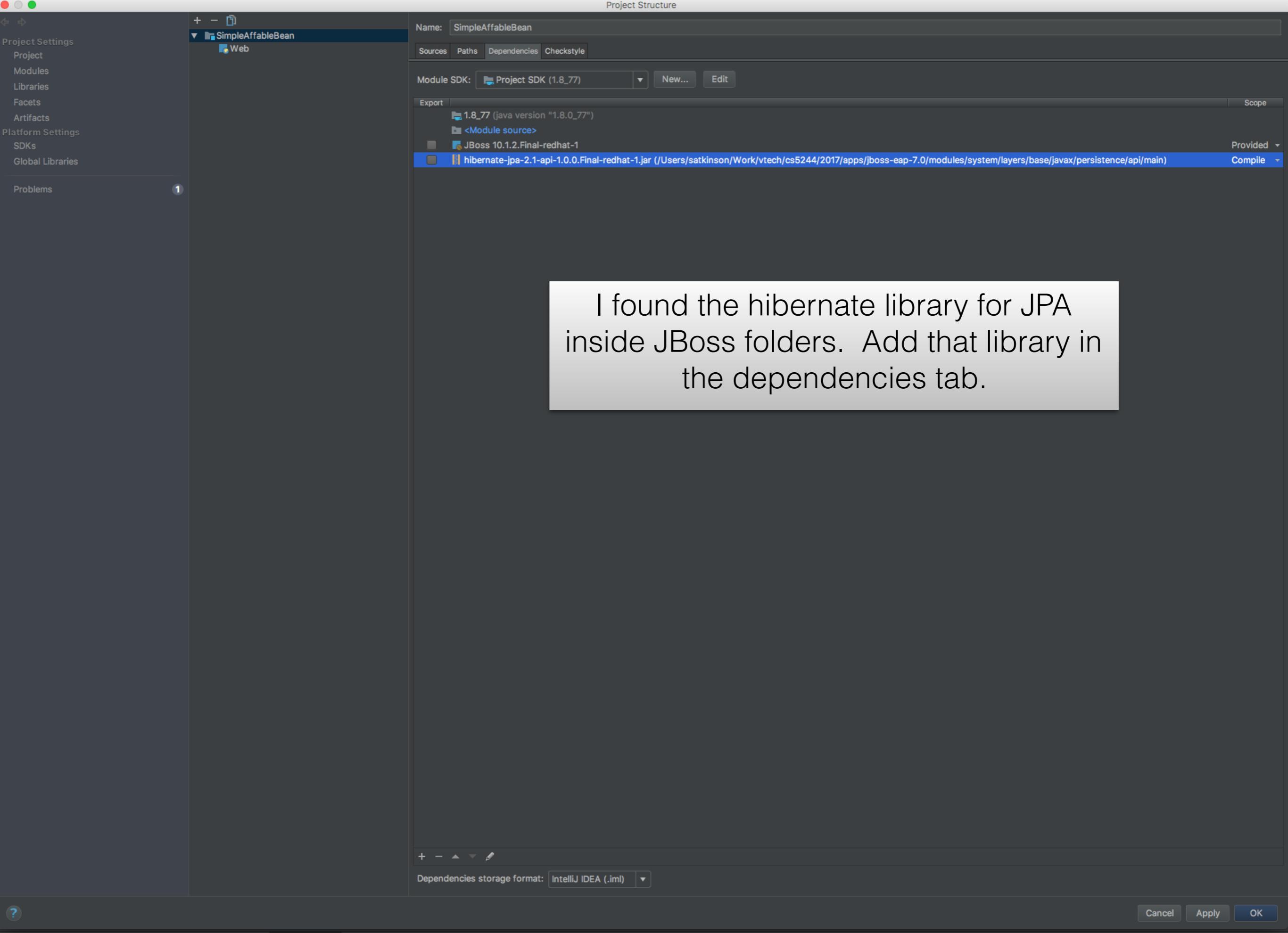
The screenshot shows an IDE interface with the following details:

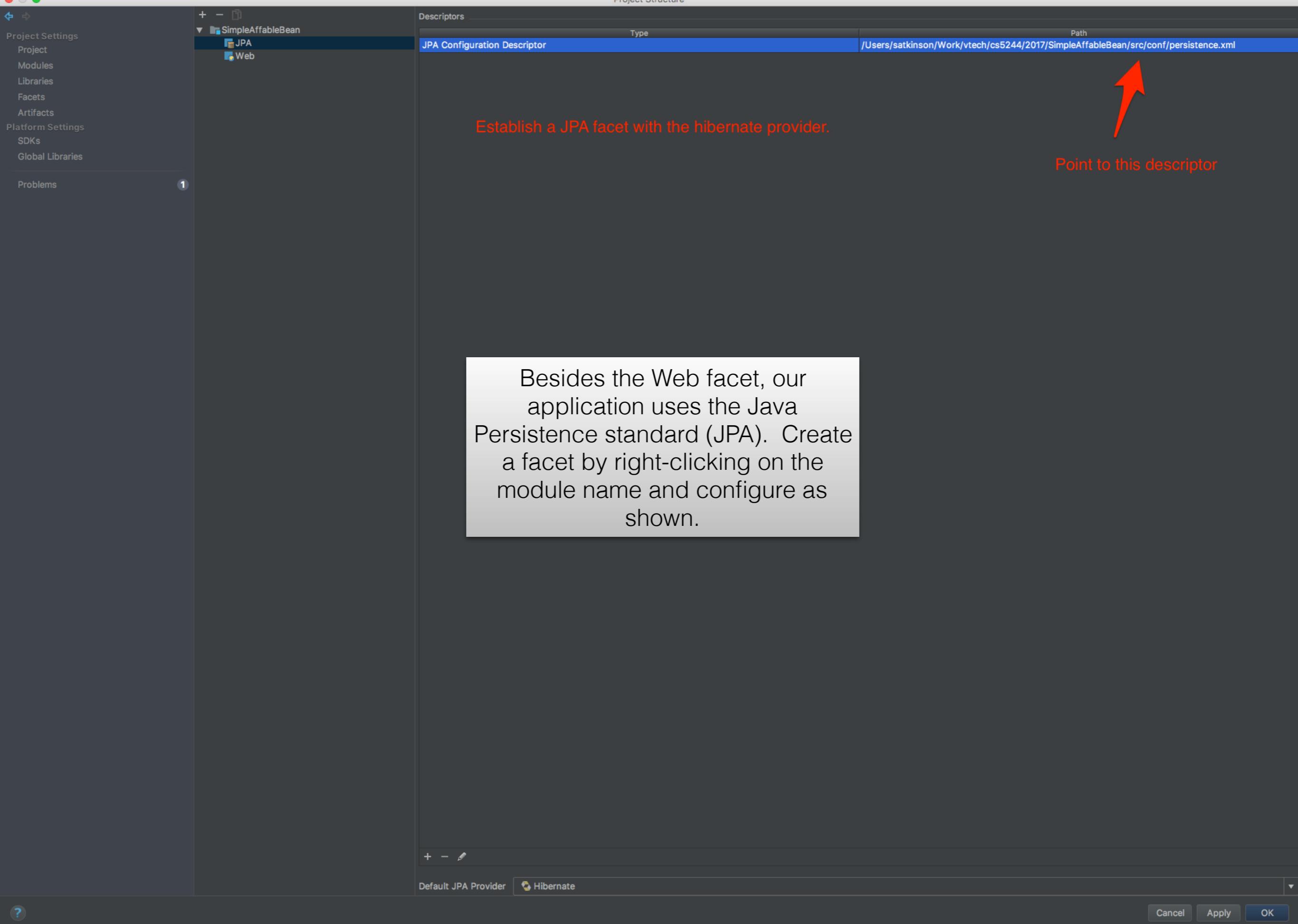
- Project Structure:** The project is named "SimpleAffableBean". It contains a "src" directory with "session" and "java" packages. The "session" package contains classes like AbstractFacade, CategoryFacade, CustomerFacade, CustomerOrderFacade, OrderedProductFacade, OrderManager, and ProductFacade. The "java" package contains a "cart" package and a "controller" package.
- Code Editor:** The main editor window displays the "OrderManager.java" file. The code is a session bean definition with annotations like `@Stateless`, `@TransactionManagement(TransactionManagementType.CONTAINER)`, and `@EJB`. It imports various Java and EJB packages.
- Build Errors:** The bottom-left pane shows a list of compilation errors for "OrderManager.java". The errors are mostly "java: package ... does not exist" for various EJB-related classes like `Stateless`, `TransactionManagement`, and `EntityManager`.
- Right Panel:** A sidebar on the right shows the contents of the "persistence.xml" file, which includes configuration for "sun-web.xml" and "web.xml".

Text on Screen:

Still looks like a persistence library is needed.
The persistence stuff is a standard api-level
jar, should be able to find it somewhere in the
application server modules.

But we are still getting compile errors if we try to run again. We must need more libraries.





CustomerOrderFacade.java - SimpleAffableBean - [~]

SimpleAffableBean Project Packages Project Files Problems Production Tests

1: Project 2: Structure

SimpleAffableBean src java session CustomerOrderFacade

SimpleAffableBean.iml SimpleAffableBean.ipr SimpleAffableBean.iws

setup README.txt sampleData.sql schemaCreation.sql sun-resources.xml

src conf MANIFEST.MF persistence.xml

java cart controller entity filter resources session AbstractFacade CategoryFacade CustomerFacade CustomerOrderFacade OrderedProductFacade OrderManager ProductFacade validate

web admin css img js test WEB-INF jspf view cart.jsp category.jsp checkout.jsp confirmation.jsp sun-web.xml web.xml index.jsp .gitignore LICENSE

Messages Build

Information: java: Errors occurred while compiling module 'SimpleAffableBean'
Information: javac 1.8.0_77 was used to compile java sources
Information: Module "SimpleAffableBean" was fully rebuilt due to project configuration/dependencies changes
Information: 1/24/17, 10:27 PM - Compilation completed with 2 errors and 0 warnings in 2s 167ms

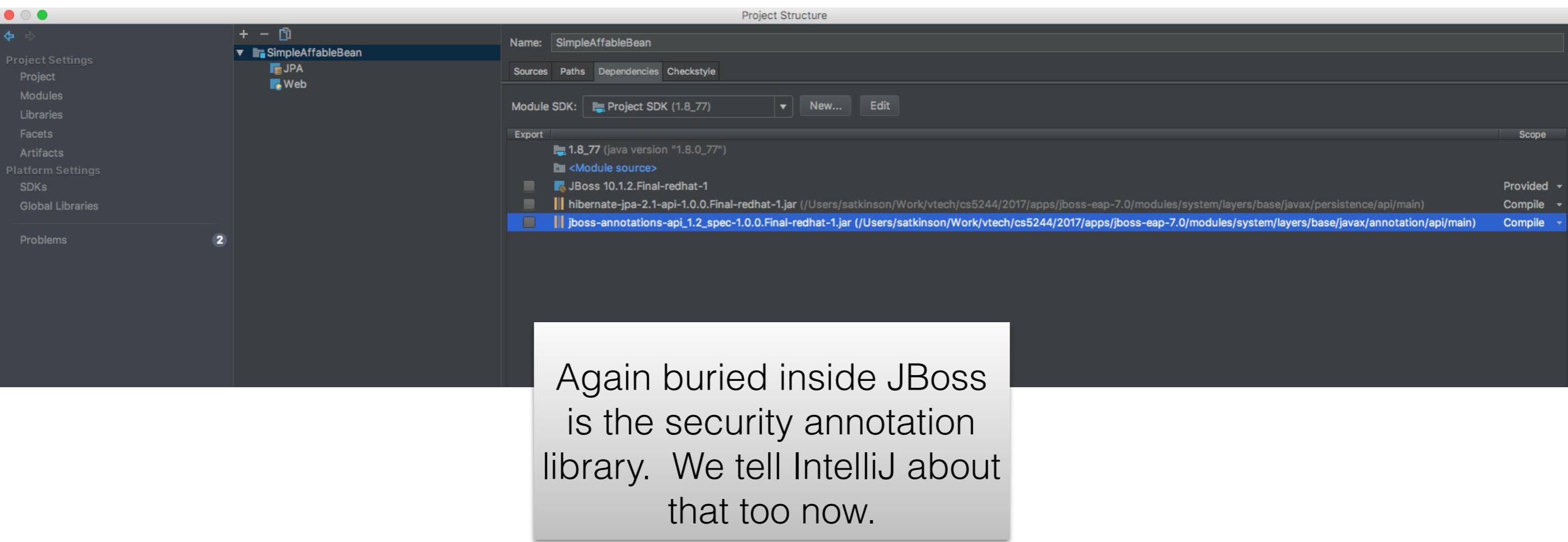
/Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/src/java/session/CustomerOrderFacade.java
Error:(12, 33) java: package javax.annotation.security does not exist
Error:(43, 6) java: cannot find symbol
symbol: class RolesAllowed
location: class session.CustomerOrderFacade

/Users/satkinson/Work/vtech/cs5244/2017/SimpleAffableBean/src/java/session/OrderManager.java
Information: java: Some input files use unchecked or unsafe operations.
Information: java: Recompile with -Xlint:unchecked for details.

```
1 +/.../
8
9     package session;
10
11    import entity.CustomerOrder;
12    import javax.annotation.security.RolesAllowed;
13    import javax.ejb.Stateless;
14    import javax.persistence.EntityManager;
15    import javax.persistence.PersistenceContext;
16
17    /**
18     * Need a security jar.
19     *
20     * @author tgiunipero
21     */
22    @Stateless
23    public class CustomerOrderFacade extends AbstractFacade<CustomerOrder>
24        @PersistenceContext(unitName = "AffableBeanPU")
25        private EntityManager em;
26
27    protected EntityManager getEntityManager() { return em; }
28
29    public CustomerOrderFacade() { super(CustomerOrder.class); }
30
31
32    // overridden - refresh method called to refresh
33    public CustomerOrder find(Object id) {
34        CustomerOrder order = em.find(CustomerOrder.class, id);
35        em.refresh(order);
36        return order;
37    }
38
39
40    // in this implementation, there is only one order per customer
41    // the data model however allows for multiple orders per customer
42    @RolesAllowed("affableBeanAdmin")
43    public CustomerOrder findByCustomer(Object customer) {
44        return (CustomerOrder) em.createNamedQuery("CustomerOrder.f...
```

Need a security jar.
So let's find it and
add another library.

Still not done. Running
yields another compile error
due to a missing security
annotation library.

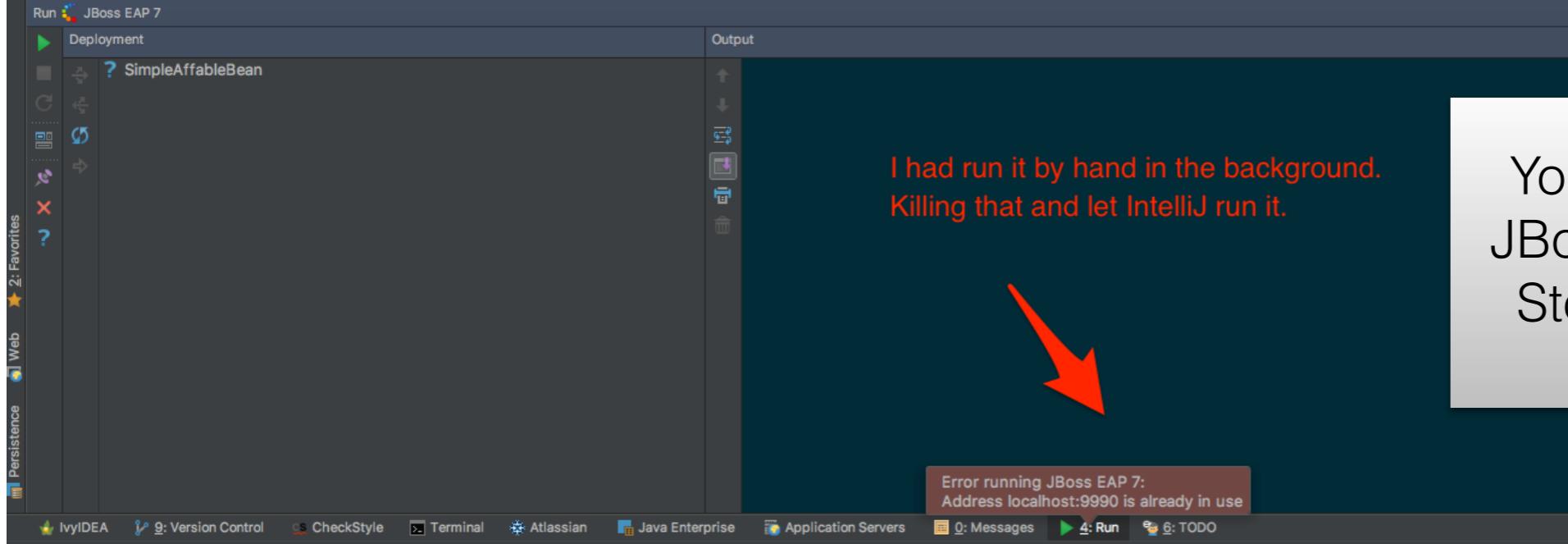


Attempting to run JBoss within IntelliJ

Are we ready to run JBoss
and deploy the web app
automatically?

The screenshot shows the IntelliJ IDEA interface with the project 'SimpleAffableBean' open. The left sidebar displays the project structure, including the 'src' directory which contains 'java', 'session', and 'CustomerOrderFacade'. The main editor window shows the 'CustomerOrderFacade.java' file:

```
CustomerOrderFacade
1 +/...
8
9 package session;
10
11 import entity.CustomerOrder;
12 import javax.annotation.security.RolesAllowed;
13 import javax.ejb.Stateless;
14 import javax.persistence.EntityManager;
15 import javax.persistence.PersistenceContext;
16
17 /**
18 * ...
19 * @author tgiunipero
20 */
21 @Stateless
22 public class CustomerOrderFacade extends AbstractFacade<CustomerOrder> {
23     @PersistenceContext(unitName = "AffableBeanPU")
24     private EntityManager em;
25
26     protected EntityManager getEntityManager() { return em; }
27
28     public CustomerOrderFacade() { super(CustomerOrder.class); }
29
30     // overridden - refresh method called to retrieve order id from database
31     public CustomerOrder find(Object id) {
32         CustomerOrder order = em.find(CustomerOrder.class, id);
33         em.refresh(order);
34         return order;
35     }
36
37     // in this implementation, there is only one order per customer
38     // the data model however allows for multiple orders per customer
39     @RolesAllowed("affableBeanAdmin")
40     public CustomerOrder findByCustomer(Object customer) {
41         return (CustomerOrder) em.createNamedQuery("CustomerOrder.findByC
42
43
44
45
```



You may still be running
JBoss in the background.
Stop JBoss - IntelliJ will
start it for you.

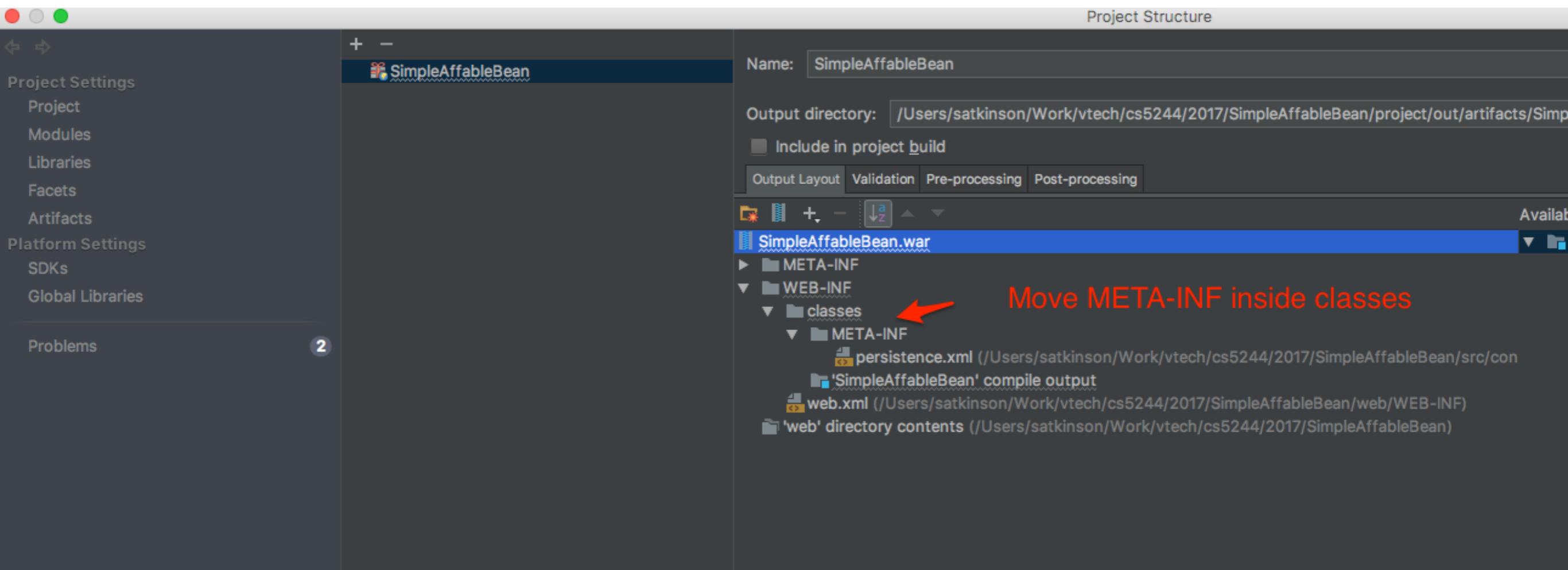
```
22:30:17,870 ERROR [org.jboss.msc.service.fail] (MSC service thread 1-3) MSC000001: Failed to start service jboss.deployment.unit."SimpleAffableBean.war".INSTALL: org.jboss.msc.service.StartException in service jboss.deployment.unit."SimpleAffableBean.war".INSTALL  
at org.jboss.as.server.deployment.DeploymentUnitPhaseService.start(DeploymentUnitPhaseService.java:154)  
at org.jboss.msc.service.ServiceControllerImpl$StartTask.startService(ServiceControllerImpl.java:1948)  
at org.jboss.msc.service.ServiceControllerImpl$StartTask.run(ServiceControllerImpl.java:1881) <2 internal calls>  
at java.lang.Thread.run(Thread.java:745)  
Caused by: org.jboss.as.server.deployment.DeploymentUnitProcessingException: WFLYEE0041: Component class session.ProductFacade for component ProductFacade has errors:  
WFLYJPA0033: Can't find a persistence unit named AffableBeanPU in deployment "SimpleAffableBean.war"  
at org.jboss.as.ee.component.deployers.ModuleJndiBindingProcessor$1.handle(ModuleJndiBindingProcessor.java:157)  
at org.jboss.as.ee.component.ClassDescriptionTraversal.run(ClassDescriptionTraversal.java:54)  
at org.jboss.as.ee.component.deployers.ModuleJndiBindingProcessor.processClassConfigurations(ModuleJndiBindingProcessor.java:186)  
at org.jboss.as.ee.component.deployers.ModuleJndiBindingProcessor.deploy(ModuleJndiBindingProcessor.java:143)  
at org.jboss.as.server.deployment.DeploymentUnitPhaseService.start(DeploymentUnitPhaseService.java:147)  
... 5 more  
  
22:30:17,889 ERROR [org.jboss.as.controller.management-operation] (management-handler-thread - 2) WFLYCTL0013: Operation ("deploy") failed - address: [[{"deployment" => "SimpleAffableB  
"WFLYCTL0080: Failed services" => {"jboss.deployment.unit.\\"SimpleAffableBean.war\\".INSTALL" => "org.jboss.msc.service.StartException in service jboss.deployment.unit.\\"SimpleAffab  
Caused by: org.jboss.as.server.deployment.DeploymentUnitProcessingException: WFLYEE0041: Component class session.ProductFacade for component ProductFacade has errors:  
WFLYJPA0033: Can't find a persistence unit named AffableBeanPU in deployment \\"SimpleAffableBean.war\\",  
"WFLYCTL0180: Services with missing/unavailable dependencies" => [  
    "jboss.deployment.unit.\\"SimpleAffableBean.war\\".weld.weldClassIntrospector is missing [jboss.deployment.unit.\\"SimpleAffableBean.war\\".beanmanager]",  
    "jboss.deployment.unit.\\"SimpleAffableBean.war\\".batch.environment is missing [jboss.deployment.unit.\\"SimpleAffableBean.war\\".beanmanager]"  
]  
}  
]  
22:30:17,891 ERR  
{  
    "WFLYCTL0080:  
Caused by: o  
WFLYJPA0033: Can  
"WFLYCTL0180:  
    "jboss.d  
    "jboss.d  
}  
]
```

It seems we are not finding persistence unit "AffableBeanPU".

This was established in persistence.xml.

After much swearing, I realize that persistence.xml is not packaged properly in the artifact....

of deployment "SimpleAffableBean.war" was rolled back with the following failure message:
ALL" => "org.jboss.msc.service.StartException in service jboss.deployment.unit.\\"SimpleAffab
Component class session.ProductFacade for component ProductFacade has errors:
Bean.war\\",
ng [jboss.deployment.unit.\\"SimpleAffableBean.war\\".beanmanager]",
.deployment.unit.\\"SimpleAffableBean.war\\".beanmanager]"



Move META-INF inside classes

Return to Project Structure/Artifacts.

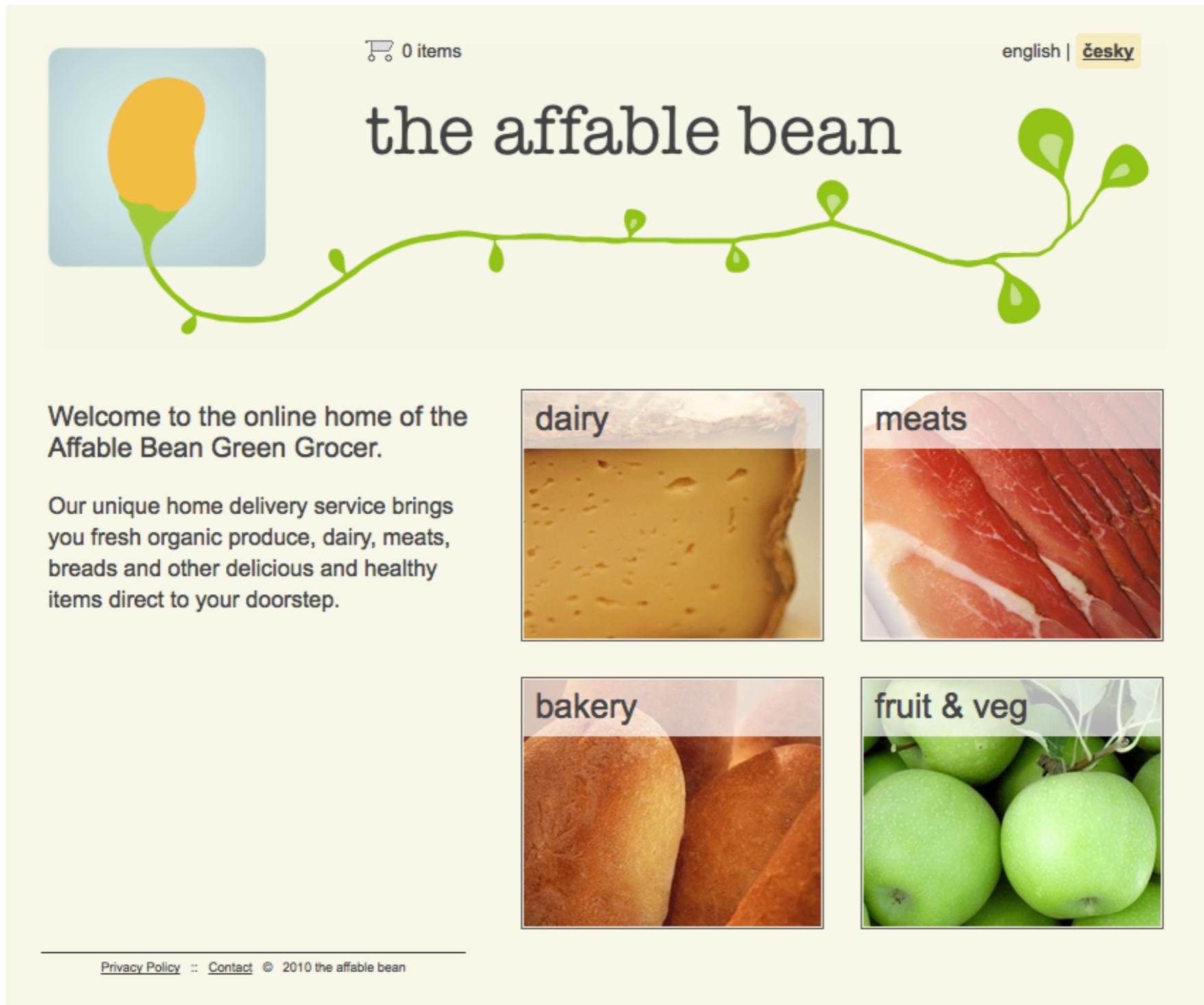
There the WEB-INF/META-INF folder that had persistence.xml needs to be inside WEB-INF/classes/META-INF according to the JPA packaging rules.

```
java:global/SimpleAffableBean/OrderManager!session.OrderManager
java:app/SimpleAffableBean/OrderManager!session.OrderManager
java:module/OrderManager!session.OrderManager
java:global/SimpleAffableBean/OrderManager
java:app/SimpleAffableBean/OrderManager
java:module/OrderManager

22:40:50,301 INFO [org.infinispan.configuration.cache.EvictionConfigurationBuilder] (ServerService Thread Pool -- 25) ISPN000152: Passivation configured without an eviction policy being sele
22:40:50,303 INFO [org.infinispan.configuration.cache.EvictionConfigurationBuilder] (ServerService Thread Pool -- 25) ISPN000152: Passivation configured without an eviction policy being sele
22:40:50,303 INFO [org.infinispan.configuration.cache.EvictionConfigurationBuilder] (ServerService Thread Pool -- 25) ISPN000152: Passivation configured without an eviction policy being sele
22:40:50,304 INFO [org.infinispan.configuration.cache.EvictionConfigurationBuilder] (ServerService Thread Pool -- 25) ISPN000152: Passivation configured without an eviction policy being sele
22:40:50,317 INFO [org.jboss.weld.deployer] (MSC service thread 1-6) WFLYWELD0006: Starting Services for CDI deployment: SimpleAffableBean.war
22:40:50,357 INFO [org.jboss.weld.Version] (MSC service thread 1-6) WELD-000900: 2.3.3 (redhat)
22:40:50,403 INFO [org.jboss.weld.deployer] (MSC service thread 1-8) WFLYWELD0009: Starting weld service for deployment SimpleAffableBean.war
22:40:50,571 INFO [org.jboss.as.jpa] (ServerService Thread Pool -- 58) WFLYJPA0010: Starting Persistence Unit (phase 2 of 2) Service 'SimpleAffableBean.war#AffableBeanPU'
22:40:50,648 INFO [org.jboss.as.clustering.infinispan] (ServerService Thread Pool -- 25) WFLYCLINF0002: Started client-mappings cache from ejb container
22:40:50,758 ERROR [stderr] (ServerService Thread Pool -- 58) Tue Jan 24 22:40:50 PST 2017 WARN: Establishing SSL connection without server's identity verification is not recommended. Accordin
22:40:51,010 INFO [org.hibernate.dialect.Dialect] (ServerService Thread Pool -- 58) HHH000400: Using dialect: org.hibernate.dialect.MySQL5Dialect
22:40:51,055 INFO [org.hibernate.envers.boot.internal.EnversServiceImpl] (ServerService Thread Pool -- 58) Envers integration enabled? : true
22:40:51,607 INFO [org.hibernate.hql.internal.QueryTranslatorFactoryInitiator] (ServerService Thread Pool -- 58) HHH000397: Using ASTQueryTranslatorFactory
22:40:52,407 INFO [javax.enterprise.resource.webcontainer.jsf.config] (ServerService Thread Pool -- 20) Initializing Mojarra 2.2.12-jbossorg-2 for context '/SimpleAffableBean'
22:40:53,150 INFO [org.wildfly.extension.undertow] (ServerService Thread Pool -- 20) WFLYUT0021: Registered web context: /SimpleAffableBean
22:40:53,164 INFO [org.jboss.as.server] (management-handler-thread - 2) WFLYSRV0010: Deployed "SimpleAffableBean.war" (runtime-name : "SimpleAffableBean.war")
[2017-01-24 10:40:53,177] Artifact SimpleAffableBean: Artifact is deployed successfully
[2017-01-24 10:40:53,177] Artifact SimpleAffableBean: Deploy took 3,842 milliseconds
```

Woot!

Running the application now deploys successfully!



And the browser popped up straight away!