Intellij IDEA

- Checklist
- Checkout the project
- · Setting up the environment
 - Add manual dependencies to maven
 - Setup derby database
- Setup the project
 - Initial project setup
 - Run/Debug configurations
 - Webapp Yes Shop
 - Webapp REST API
 - Webapp JAM server
- Running the project
 - Derby
 - JAM Client
- Final configurations
 - Configuring shared image vault
 - IDE recommended settingsImport demo data
- Tips and Tricks
 - javac source release 1.X requires target release
 - 1.X
 - where are all the log files
 - Error creating bean with name 'dataSource' defined in class path resource
 - Unable to import maven project: See logs for details

Checklist

Ensure all required libraries and software is installed as per install from source guide

Ensure all required path variables are setup as per install from source guide

Checkout the project

Navigate to directory of your choice where you want the sources to reside on your file system. For this guide we will be using:

/development/projects/java

Create a new directory for project.

For this guide we will be using:

yc-idea

The repository is located at GutHub https://github.com/inspire-software/yes-cart.

Please refer to instruction on GutHub https://help.github.com/articles/which-remote-url-should-i-use/ on the cloning/forking options available.

Setup IDE (18:24) Play Part 1: Setting up development environment Cloning is the typical way of getting repository for developers. For changes to be pushed to repository you need write permissions. If you plan create your own version of the platform you should consider forking repository instead (see GitHub guide https://help.git hub.com/articles/fork-a-repo/).

The easiest way to clone is to use GitHub application but you will need a GitHub account for that.

Alternatively you can use command line but you will need to setup git (see guide https://help.github.com/articles/set-up-git/). And then run:

\$ mkdir yc-idea
\$ cd yc-idea
\$ git clone git@github.com:inspire-software/yes-cart.git .

This will put all sources to the yc-idea directory which will look something like this:

<pre>denispavlov@denis-pavlovs-macbook-pro: /development/projects/java/yc-idea\$ 11</pre>						
total 936						
drwxr-xr-x@	denispav	lov admin	816B Jun	8	19:31	
drwxr-xr-x@	denispav	lov admin	782B Jun	8	18:22	
drwxr-xr-x	denispav	lov admin	102B Apr	16	22:48	.idea
drwxr-xr-x@	denispav	lov admin	340B Jun	8	19:31	.svn
drwxr-xr-x@	denispav	lov admin	238B Jun	8	15:50	core
drwxr-xr-x	denispav	lov admin	238B Jun	8	16:00	core-dto
drwxr-xr-x@	denispav	lov admin	544B Jun	8	15:49	core-modules
drwxr-xr-x@	denispav	lov admin	238B Jun	8	15:48	domain-api
drwxr-xr-x@	denispav	lov admin	476B Jun	4	20:36	env
drwxr-xr-x@	denispav	lov admin	238B Dec	4	2014	imagevault
drwxr-xr-x@	denispav	lov admin	204B Jun	5	11:29	installer
drwxr-xr-x@	denispav	lov admin	272B Jun	8	11:42	manager
drwxr-xr-x@	denispav	lov admin	238B Jun	8	15:48	payment-api
drwxr-xr-x@	denispav	lov admin	272B Jun	8	15:48	persistence
-rw-rr	denispav	lov admin	27K Jun	8	15:27	pom.xml
drwxr-xr-x	denispav	lov admin	102B Jun	8	15:48	target
drwxr-xr-x@	denispav	lov admin	238B Jun	8	16:02	theme
drwxr-xr-x@	denispav	lov admin	306B Mar	23	21:41	util
drwxr-xr-x@	denispav	lov admin	306B Jun	8	16:02	web
-rwxr-xr-x	denispav	lov admin	9.2K Jun	8	14:23	yc.bat
-rwxr-xr-x	denispav	lov admin	9.0K Jun	8	14:22	yc.sh
-rw-rr	denispav	lov admin	4.4K Apr	21	10:46	yes-cart.iml
-rw-rr	denispav	lov admin	187K Jun	8	08:06	yes-cart.ipr
-rw-rr	denispav	lov admin	217K Jun	8	19:31	yes-cart.iws
denispavlov@	nis-pavlo	vs-macbook-p	oro: /devei	lopr	nent/pr	rojects/java/yc-idea\$

Setting up the environment

On *nix systems you need to add executable permission to ./yc.sh script like so: "chmod +x yc.sh"

Use the yc.sh or yc.bat file in order to setup the environment.

This script uses mvn executable therefore if you have abnormal maven installation please open the script and ensure that MVN

Add manual dependencies to maven

Use the script command to add 3rd party dependencies that are not available through maven central:

```
$ ./yc.sh i3rd
```

Run the maven install

Use standard maven install command from the root of project directory to ensure that all modules are built.

```
$ mvn clean install -PdevIntellijIDEA,derby,ftEmbededLucene,paymentAll
-DskipTests=true
```

If you are planning to use IDEA setup as your primary development option you must always use **devIntellijIDEA** when building from command line

For this setup we are using the devIntellijIDEA profile for a derby build. Please consult this page for details on the available profiles

The build process will take some time. Note that we are using **clean install** as clean phase will ensure that all node_modules dependencies for jam-jaclient are correctly installed. skipTests option will allow us to save some time by skipping the tests during this build

Setup derby database

If you encounter java.security.AccessControlException: access denied ("java.net.SocketPermission" "localhost:1527" "listen,resolve") refer to Derby quick startguide.

Start the derby in background

\$./yc.sh derbygob

Ensure that you see the following message that signifies that Derby has started and ready:

```
Starting Derby database (background mode)
Derby server started on port 1527...
Mon Dec 31 14:38:06 GMT 2012 : Security manager installed using the
Basic server security policy.
Mon Dec 31 14:38:11 GMT 2012 : Apache Derby Network Server - 10.8.1.2 -
(1095077) started and ready to accept connections on port 1527
```

Run the dbi (database initialisation)

\$./yc.sh dbiderby

This will generate the following message:

```
Initialise Derby database

Setting Derby environment variables

Running init script

/development/projects/java/yc-idea/env/setup/dbi/derby/dbinit.sql

Initialisation complete. See log:

/development/projects/java/yc-idea/derbyinit.log
```

Ensure that derbyinit.log file does not have any critical errors in it (e.g. Table not created or insert failed).

At this point you have a blank Derby database.

We recommend backing up the database files so that a clean version is always available.

When working with Derby files ALWAYS shutdown the Derby server prior doing the backup - or the database will become corrupt.

Shutdown derby

\$./yc.sh derbyend

Ensure that you see that shutdown message:

```
Stopping Derby database

Mon Dec 31 14:45:28 GMT 2012 : Apache Derby Network Server - 10.8.1.2 -

(1095077) shutdown

Derby server stopped...

Mon Dec 31 14:45:29 GMT 2012 : Apache Derby Network Server - 10.8.1.2 -

(1095077) shutdown
```

Optionally backup the files

The Derby files will be located in

```
/env/derby/lib/yes - the main database
/env/derby/lib/yespay - payment transactions database
```

In order to back up just copy these two directories. When you need to reinitialise the database just replace these with backed up versions. AL WAYS do this with Derby shutdown otherwise the database files will become corrupt.

Setup the project

Initial project setup

Intellij IDEA comes bundled with Maven 3, we highly recommend not to use it and reconfigure Maven home to point to your local installation that you use to perform command line builds (Preferences > Build, Execution, Deployment > Build Tools > Maven)

Open IDEA and select Import Project



Point to the **pom.xml** in your project root directory

O O Select File or Directory to Import
Select directory with existing sources , Eclipse project (.project) or classpath (.classpath) file, Maven project file (pom.xml), Bnd/Bndtools workspace (workspace/cnf), Flash Builder project file (.project, *.fxp, *.fxpl), Gradle build script (*.gradle).
😭 📼 🕽 🕞 🗶 🍠 🔳 🛛 Hide path
/development/projects/java/yc-idea/pom.xml
 sandbox yc yc-idea core core-dto core-modules domain-api env installer manager payment-api persistence target theme util web
? Cancel OK

Leave the directory path as is and click next

Import Project from Maven
Root directory /development/projects/java/yc-idea
Search for projects recursively
Project format: .ipr (file based)
Keep project files in:
Import Maven projects automatically
Create IntelliJ IDEA modules for aggregator projects (with 'pom' packaging)
Create module groups for multi-module Maven projects
✓ Keep source and test folders on reimport
Exclude build directory (%PROJECT_ROOT%/target)
✓ Use Maven output directories
Generated sources folders: Detect automatically +
Phase to be used for folders update: process-resources \$
IDEA needs to execute one of the listed phases in order to discover all source folders that are configured via Maven plugins. Note that all test-* phases firstly generate and compile production sources.
Automatically download: 🗌 Sources 🗌 Documentation
Dependency types: jar, test-jar, maven-plugin, ejb, ejb-client, jboss-har, jboss-sar, war, ear, bundle
Comma separated list of dependency types that should be imported
Environment settings
? Cancel Previous Next

Select then following profiles: derby, devIntellijIDEA, ftEmbededLucene and paymentAll. Make sure that other profiles are unticked.

Ensure that you select ALL relevant profiles for the version you are using, as new versions may introduce new mandatory profiles that must be selected

0	0	Import Project from Maven		
Sele	ect profiles:			
	ssl paymentCybersourceOf derby dev paymentBase uat paymentAuthorizeOff buildInstaller paymentAll systest mysql paymentPaypalOff paymentCybersource prod paymentPaypal paymentLiqPayOff devIntellijIDEA paymentAuthorize	f		
?	Cancel		Previous Next	

Select the maven project to import:

00	Import Project from Maven
Select Maven projects to impo	rt
org.yes:yes-cart:3.0.0-:	HAPSHOT
	Select all Unselect all
Open Project Structure	after import
? Cancel	Previous Next

Select project SDK:

$\Theta \cap \Theta$	Import Project from Maven		
Please select project SDK. This SDK will be used by default by all project modules.			
+ -	Name: 1.8		
1.7	JDK home path: <1.8.0_45.jdk/Contents/Home		
Rex_sdk_4.6	Classpath Sourcepath Annotations Documentation Paths /Library/Java/JavaVirtualMachines/jdk1.8.0_45.jdk/Conte		
? Cancel	Previous		

Leave the project name as is and finish:

00	Import Project from Maven		
Please enter a name to create a new IntelliJ IDEA project.			
Project name:	yes-cart		
Project file location:	/development/projects/java/yc-idea		
? Cancel	Previ	ous Finish	

Let IDEA process the files (this will take some time).

Run/Debug configurations

Webapp Yes Shop

Open configurations window: Run > Edit Configurations

Click "+" in the top left corner and select "Tomcat Server > Local" from the menu

000	Run/Debug Configuration	ns
+ - 🖻 🛠 »		
Add New Configuration	button to create a new configuration bas	ed on default settings
Applet	button to create a new configuration bas	ed on default settings
Application		
Chromium Remote		
👼 Cucumber java		
Firefox Remote		
🗲 Flash App		
🔏 Flash Remote Debug		
🛵 FlexUnit		
GlassFish Server		
💽 Gradle		
😡 Grails		
Oriffon		
Groovy		
🦉 Grunt.js		
🏺 Gulp.js		
JAR Application		
🔄 JavaScript Debug		
JUnit		
🏶 Maven		
🟟 NW.js		
Remote		
🚔 Spy-js		
😤 Spy-js for Node.js		
NØ TestNG		
👷 Tomcat Server 🔹 🕨	Add New 'Tomcat Server' Configuration	
T XSLT	👷 Local	
20 items more (irrelevant)	Remote	
(?)		Cancel Apply OK

In the window specify:

- Name: yc-shop
 Select Tomcat 7 as the application server (add it using configure button if not listed)
 Set VM options: -Xmx512m -Xms256m
 Select on 'Update' action as "Update classes and Resources"

You need to come back from "deployment tab" (next step) to set this option as it is only available once WAR files are added to deployment.

- Uncheck open browser after launchEnsure HTTP port number is 8080 and JMX is 1099

000	Run/Debug Configurations
+ - 🛅 🗚 »	Name: yc-shop Share
 Flash App Tomcat Serve 	Server Deployment Logs Code Coverage Startup/Connection
 Section 2 Section 2	Application server: Tomcat 7 ‡ Configure
	Open browser
	After launch OF Default + with JavaScript debugger
	http://localhost:8080/yes-shop/
	VM options: -Xmx512m -Xms256m
	On 'Update' action: Update classes and resources 💠 🗹 Show dialog
	On frame deactivation: Do nothing \$
	Use alternative JRE:
	Tomcat Server Settings
	HTTP port: 8080 Deploy applications configured in Tomcat instance
	HTTPs port: Preserve sessions across restarts and redeploys
	JMX port: 1099
	AJP port:
	Before launch: Make, Build Artifacts, Maven Goal
?	Close Apply OK

Click "Apply" and select deployment tab the following:

- Click "+", then "Artifact" and select "yes-shop-wicket:war exploded"
 Set application context to / (i.e. root) Note: 3.4.0+ now has default mapping of storefront to root (and not /yes-shop)

00	Run/Debug Configurations	
+ - 🛅 🌮 »	Name: yc-shop	🗌 Share
 Flash App Tomcat Serve vc-shop 	Server Deployment Logs Code Coverage Startup/Connection	
Pefaults	Deploy at the server startup	
	stypes-shop-wicket:war exploded Application context: //yes-shop	v
	+ - A - P	
	Before launch: Make, Build Artifacts	
	** Make	
(?)	Cancel Apply	OK

On the "Before launch" section at the bottom of the window:

- Click "+", "Run maven goal"
- Select working directory as "YC_HOME/yc-idea/web/store-wicket"
- Type "process-resources" into Command Line



Click "Apply", then "OK" to save configuration

Now you will be able to run storefront from IDEA by using the Run menu choice **yc-shop** or from the Application Servers window (View > Tool Windows > Application Servers).

Webapp REST API

Open configurations window: Run > Edit Configurations

Click "+" in the top left corner and select "Tomcat Server > Local" from the menu



In the window specify:

- Name: yc-api
- Select Tomcat 7 as the application server (add it using configure button if not listed)
- Set VM options: -Xmx512m -Xms256m
- Select on 'Update' action as "Update classes and Resources"

You need to come back from "deployment tab" (next step) to set this option as it is only available once WAR files are added to deployment

- Uncheck open browser after launch
- Ensure HTTP port number is 8081 and JMX is 1100

000	Run/Debug Configurations
+ - 🖺 🛠 » N	Name: yc-api Share
► Flash App ▼ R Tomcat Serve	Server Deployment Logs Code Coverage Startup/Connection
interest and a second	Application server: Tomcat 7 Configure
	Open browser
	After launch 💿 Default 💠 🔤 with JavaScript debugger
	http://localhost:8081/yes-api/
	VM options: -Xmx512m -Xms256m
	On 'Update' action: Update classes and resources 💠 🗹 Show dialog
	On frame deactivation: Do nothing \$
	Use alternative JRE:
	Tomcat Server Settings
	HTTP port: 8081 Deploy applications configured in Tomcat instance
	HTTPs port: Preserve sessions across restarts and redeploys
	JMX port: 1100
	AJP port:
	Defens lowely Males Daild Artifacts
	Make
?	Close Apply OK

Click "Apply" and select deployment tab the following:

- Click "+", then "Artifact" and select "yes-api:war exploded"
 Set application context to /yes-api

00	Run/Debug Configurations	
+ - 🖺 😤 👋	Name: yc-api	Share
 ▶ Flash App ▼ ₹ Tomcat Serve 	Server Deployment Logs Code Coverage Startup/Connection	
₩yc-api	Deploy at the server startup	
Pefaults	ses-api:war exploded	-
	Application context. Tyes-api	
E.		
Ŧ		
	Before launch: Make, Build Artifacts	
	↓ Make	
?	Close Apply	OK

On the "Before launch" section at the bottom of the window:

- Click "+", "Run maven goal"
 Select working directory as "YC_HOME/yc-idea/web/api"
 Type "process-resources" into Command Line

Before launch: Make, Build Artifacts, Maven Goal		
↓ ⁰¹ ₀₁ Make		
👬 Build 'yes-api:war exploded' artifact		
m Run Maven Goal 'Yes cart storefront API: process-resources'		
$ + - \mathscr{I} \perp =$		

Click "Apply", then "OK" to save configuration

Now you will be able to run REST API from IDEA by using the Run menu choice **yc-api** or from the Application Servers window (View > Tool Windows > Application Servers).

Webapp JAM server

Open configurations window: Run > Edit Configurations

Click "+" in the top left corner and select "Tomcat Server > Local" from the menu



In the window specify:

- Name: yc-manager
- Select Tomcat 7 as the application server (add it using configure button if not listed)
- Set VM options: -Xmx512m -Xms256m
- Select on 'Update' action as "Update classes and Resources"

You need to come back from "deployment tab" (next step) to set this option as it is only available once WAR files are added to deployment.

- Uncheck open browser after launch
- Ensure HTTP port number is 8082 and JMX is 1101

00	Run/Debug Configurations
+ - 🖺 🛠 »	Name: yc-manager Share
 Flash App Tomcat Serve 	Server Deployment Logs Code Coverage Startup/Connection
₩yc-api	Application server: Tomcat 7 Configure
₩yc-manage P Pefaults	Open browser
	After launch 📀 Default 💠 🔤 with JavaScript debugger
	http://localhost:8082/yes-manager/
	VM options: -Xmx512m -Xms256m
	On 'Update' action: Update classes and resources 💠 🗹 Show dialog
	On frame deactivation: Do nothing \$
	Use alternative JRE:
	Tomcat Server Settings
	HTTP port: 8082 Deploy applications configured in Tomcat instance
	HTTPs port: Preserve sessions across restarts and redeploys
	JMX port: 1101
	AJP port:
	Berore launch: Make, Build Artifacts
?	Close Apply OK

Click "Apply" and select deployment tab the following:

- Click "+", then "Artifact" and select "jam:war exploded"
 Set application context to /yes-manager

00	Run/Debug Configurations	
+ - 🗈 🗚 »	Name: yc-api	Share
Flash App Flash Serve Tomcat Serve	Server Deployment Logs Code Coverage Startup/Connection	
≪yc-shop ≪yc-api	Deploy at the server startup	
Pefaults	yes-api:war exploded	
	↓ Make	
?	Close Apply	ОК

On the "Before launch" section at the bottom of the window:

- Click "+", "Run maven goal"
 Select working directory as "YC_HOME/yc-idea/yc-idea/manager/jam"
 Type "process-resources" into Command Line

* Before launch: Make, Build Artifacts, Maven Goal		
Make		
🏶 Build 'yes-shop-wicket:war exploded' artifact		
m Run Maven Goal 'Yes cart storefront (Wicket): process-resources'		
$+$ $ \mathscr{I}$ \blacktriangle \neg		

Click "Apply", then "OK" to save configuration

Now you will be able to run JAM server from IDEA by using the Run menu choice **yc-manager** or from the Application Servers window (View > Tool Windows > Application Servers).

Running the project

When running the platform locally using above configuration you have the following components:

Component	Dependency	Notes
Derby database		Derby server is started and stopped using yc.sh/yc.bat from command line. You must have derby server running before starting up any of the webapps
Storefront	Derby	Client facing storefront that is launched from IDEA run option yc-shop
REST API	Derby	REST API that is launched from IDEA run option yc-api
JAM server	Derby, Storefront (partial), REST API (partial)	Java Angular Manager (JAM) is administrative application server that is launched from IDEA run option y c-manager . Some operations in JAM are communicating update messages to Storefront and REST applications, so those need to be running for some operations to be performed (e.g. Indexing)

Derby

Navigate to \$YC_HOME in command line and start Derby database using command:

```
$ ./yc.sh derbygob
If you need to work with DB directly then connect using command:
    $ ./yc.sh derbycon
When you finished working you can shutdown Derby using command:
```

1-----

\$./yc.sh derbyend

Webapps

Running all web apps is straightforward from IDEA, simply launch the appropriate Run configuration.

Remember that all web apps require Derby DB running.

· Follow dependency table above to see in which order web apps need to be started

Webapp	IDEA Run	URL	Notes
	configuration		

Storefront	yc-shop	http://localhost:8080/	You will need to set up connection to smtp server in order to see customer registration and order details update. Look for config-mail.properties in YC_HOME/env/maven/(yourenvironment) , default environment is dev
REST API	ус-арі	http://localhost:8081/yes-api/	Use REST client of your choice to work with API (some examples: Chrome REST plugin, CoocoaRestClient, SOAP UI). Data formats supported are: XML, JSON
JAM server	jam	http://localhost:8082/yes-manager/	This URL should redirect to client in appropriate language. If the page is blank it could be that maven build was not done and client Flex app was not build. You can run it from IDEA as well.

If you encounter problems because missing resources, such as mail templates or swf, this could be because IDEA did a clean build and those files are not available. You need to copy those files either by running "mvn validate -Ptemplates" (see maven profiles section), or you could use "\$./yc.sh cpres" command that will copy all necessary resources to all webapps

JAM Client

JAM consists of two components: server (jam) and client (jam-jsclient).

The server must be running when developing the client. In order to track your modifications **npm start** command must be run from the **mana ger/jam-jsclient/src/main/typescript** directory. This will **only track** the modifications you make and compile the typescript into javascript. In order to update the files on the running server you must run **mvn validate -Pdev,nodejs,derby,ftEmbededLucene** from **manager/jam** which will update the resources on your currently running server.

Default user/password to connect is:

username	admin@yes-cart.com
password	1234567

Final configurations

Configuring shared image vault

By default the image vault is configured in the environment specific configuration file **yc-config.properties** to point to **theme/imagevault/src/ main/resources/default/**which contains the default layout for this directory. We highly recommend copying this outside of the project directory structure and changing the paths in yc-config.properties

IDE recommended settings

We recommend to enable the following settings:

Tool Windows bars

From preferences: Appearance & Behaviour > Appearance Click "Show tool window bars"

This will create buttons on edges of the IDE window to allow you quickly navigate between Tools Windows.

Import demo data

Follow steps on Demo Data Import

Tips and Tricks

javac source release 1.X requires target release 1.X

When Intellij IDEA imports maven projects it sets command line flag for the compiler to be compatible with source and target versions set in pom.xml files.Symptoms is the following message during compilation:

Mess	sages Make		
••		Information: Compilation completed with 1 error and 0 warnings	
н.	-	Information: 1 error Information: 0 warnings	
×	<u>n</u>	Error: javac: source release 1.7 requires target release 1.7	

To fix this open Preferences > Compiler > Java Compiler and change the "Additional command line parameters" as necessary.

$\Theta \bigcirc \Theta$	Settings
 Project Settings [yes-cart] Code Style ColdFusion Compiler Excludes Java Compiler Annotation Processors 	Settings Compiler > Java Compiler Use compiler: Javac Javac Options Image: Ima
RMI Compiler Android DX Compiler Flex Compiler Groovy Compiler Validation Copyright	Additional command line parameters: -target 1.7
Coverage CSS Dialects Deployment File Colors File Encodings Gant Cradle	
GUI Designer Inspections	Cancel Apply OK

where are all the log files

Intellij IDEA generates a temporary directory for Tomcat which mimics its structure to load web applications via run/debug/application server windows.

The logs are kept there along with temporary files.

On MacOSx the directory is in:

/Users/[yourusername]/Library/Caches/IntelliJIdea11/tomcat/[webappname]/
logs

where [yourusername] is your login and [webappname] is auto generated name for your run configuration.

On Windows the directory is in:

C:\Users\User.IntelliJIdea14\system\tomcat\[webappname]\logs

java.lang.lllegalArgumentException Could not resolve placeholder

Sometimes when you switch between branch and master in Git Intellij IDEA files may go out of synchronisation.

Simple fix is to:

- 1. Find project in Maven window in Intellij IDEA
- 2. Click skipTests button
- 3. Run install goal for the web apps module

If above does not work it is recommended to do a clean install from command line.

Ensure that you are using local installation of Maven3 in Preferences > Build, Execution, Deployment > Build Tools > Maven

it is not recommended to use bundled one.

Perform a clean maven build from command line:

```
mvn clean install -PdevIntellijIDEA,derby,ftEmbededLucene,paymentAll
-DskipTests=true
```

Then synchronise files in Intellij IDEA: File > Synchronise

Error creating bean with name 'dataSource' defined in class path resource

Sometimes when you switch between branch and master in Git Intellij IDEA files may go out of synchronisation.

Simple fix is to:

1. Find project in Maven window in Intellij IDEA

- 2. Click skipTests button
- 3. Run install goal for the web apps module

If above does not work it is recommended to do a clean install from command line.

Ensure that you are using local installation of Maven3 in Preferences > Build, Execution, Deployment > Build Tools > Maven

it is not recommended to use bundled one.

Perform a clean maven build from command line:

mvn clean install -PdevIntellijIDEA,derby,ftEmbededLucene,paymentAll
-DskipTests=true

Then synchronise files in Intellij IDEA: File > Synchronise

Unable to import maven project: See logs for details

Sometimes when you switch between branch and master in Git idea files may become corrupted, especially when modules structure is different.

This causes IDEA to fail to load maven modules properly and display the following message when trying to re-import:



If you got into this condition and did not notice then during deploying of web application you will encounter the various errors during deployment of webapps.

If you are asked to remove some modules - do so and then re-import all modules in the Maven window.

If this does not help unfortunately the only way to solve this is to delete corrupted .idea, *.iml, *.ipr and *.iws files and reimport

- 1. Close Intellij IDEA
- 2. Navigate to .idea directory and delete it
- 3. Re-open Intellij IDEA
- 4. Reconfigure workspace