Developer environment setup

This section explain how to install the "must have" programs on your computer. The installation instruction included with the links of the softwares. Please follow this instruction carefully.

Download links

Step-by-step guide

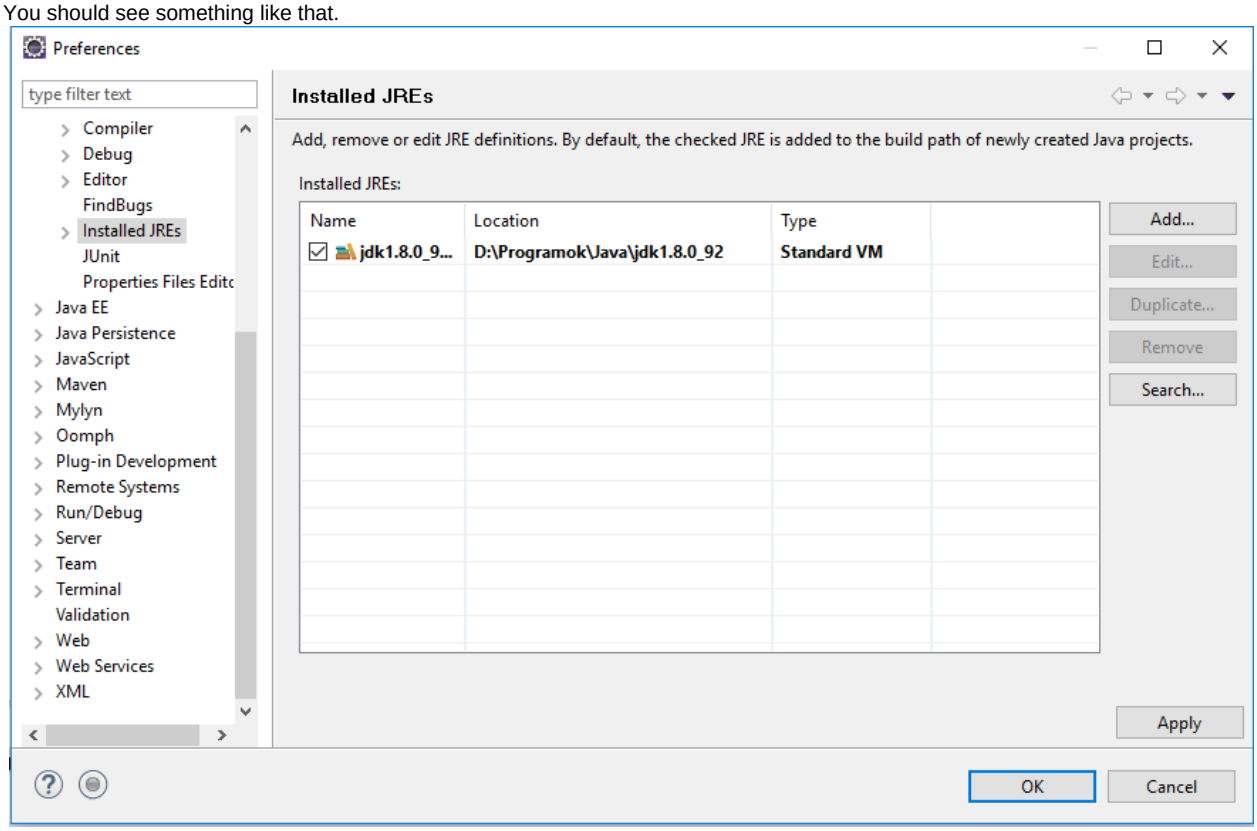
1. Run the executable file to install the JDK. After installing the software, you need to update the **PATH Environment Variable**.
2. Navigate to the **Control Panel System Advanced System Settings / System Properties Environment Variables**. Alternative: **Winkey+Pause**
3. If the **JAVA_HOME** environment variable does not exist, click the **New.** Set the Variable name to **JAVA_HOME** and the Variable value to **X:\Installation\directory\the\jdk8**
4. Edit the value of the **PATH** environment variable and add to the **PATH** the **%JAVA_HOME%\bin.** Click **OK.** Close all remaining windows.
5. Now let's check if it works or not. Window button on your keyboard and run **cmd.** Type in command line **java -version**. You should see on the console the version of the installed JDK.

> **You may see a PATH environment variable similar to the following**
> C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\;C:\Program Files (x86)\Skype\Phone\;%JAVA_HOME%\bin

Preparing Eclipse for Java development

In this section, you will verify that Eclipse is properly set up for Java development.

1. Run the **eclipse** executible file.
2. The top of the eclipse window, on the navigation bar click **Window Preferences.** You can set in this dialog how you want Eclipse to operate.
3. On the left menu click **Java Installed JREs** or use the searching field.
4. Select the installed JREs and **Remove**
5. Click the **Add** button Select the **Standard VM** Next Directory Select the folder of the jdk **Finish**
6. You should see something like that.



Maven based on **Java,** so ensure **JAVA_HOME** environment variable is set. See the instructions above.

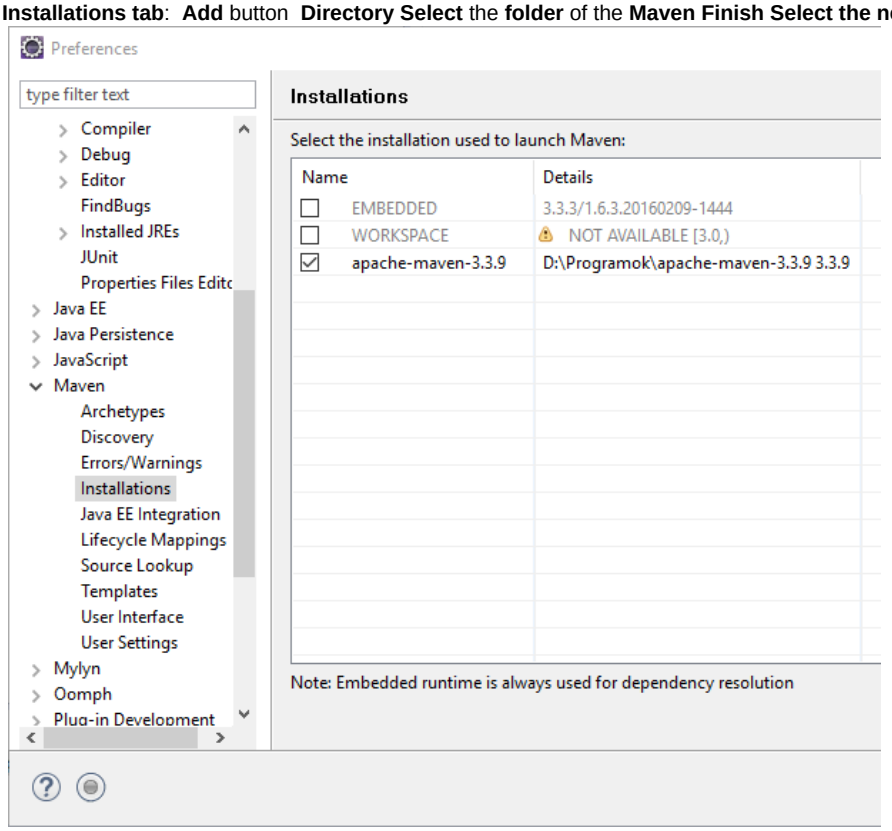
1. The installation of Apache Maven is a simple process of extracting the archive in any directory.
2. Navigate to the **Control Panel System Advanced System Settings / System Properties Environment Variables**. Alternative: **W inkey+Pause**
3. If the **M2, M2_HOME** and **MAVEN_OPTS** does not exists, click the **New.**
    a. Set the Variable name to **M2_HOME** and the Variable value to **X:\Installation\directory\apache-maven**
    b. Set the Variable name to **M2** and the Variable value to **%M2_HOME%\bin**
4. Add the **%M2%** to the **PATH.** Click **OK.**
5. Now let's check if it works or not. Window button on your keyboard and run **cmd.** Type in command line **mvn -version**. You should see on the console the version of the installed maven.

> **You may see a PATH environment variable similar to the following**
> C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\;C:\Program Files (x86)\Skype\Phone\;%JAVA_HOME%\bin;%M2%

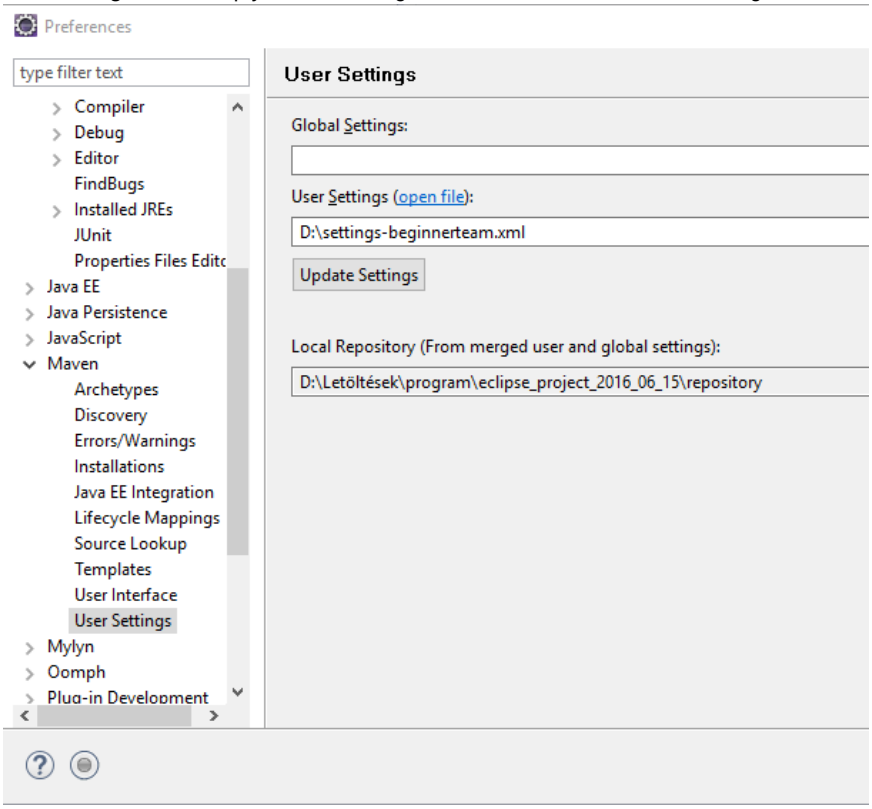
Preparing Eclipse for use Maven

1. Run the **eclipse** executible file.
2. The top of the eclipse window, on the navigation bar click **Window Preferences.** You can set in this dialog how you want Eclipse to operate.
3. On the left open the Maven menu. Let's go through each point:

a. **Maven tab**: select the Download Artifact Sources and the Download Artifact JavaDoc fields
b. **Installations tab**: **Add** button **Directory Select** the **folder** of the **Maven Finish Select the newly added maven.**



a. **User Settings tab**: Set up your own setting.xml file. You should use the following: Maven settings.xml

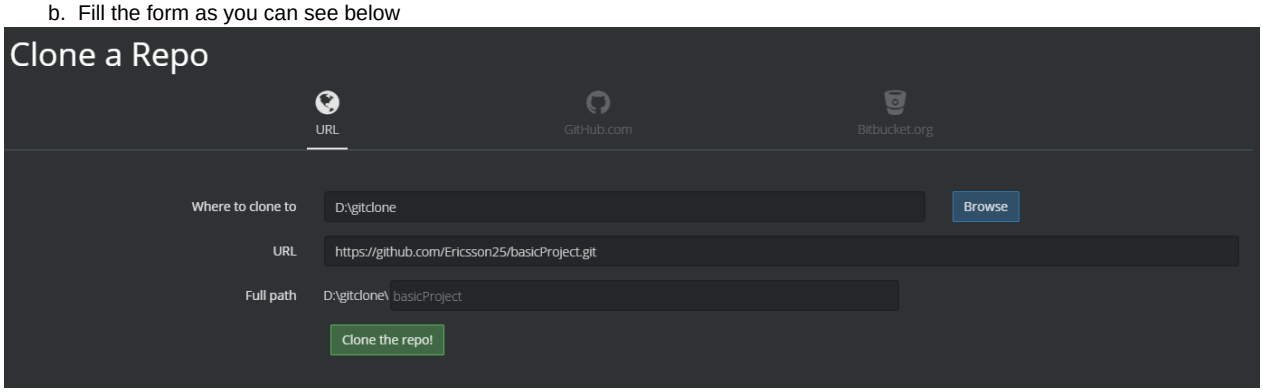


GitKraken

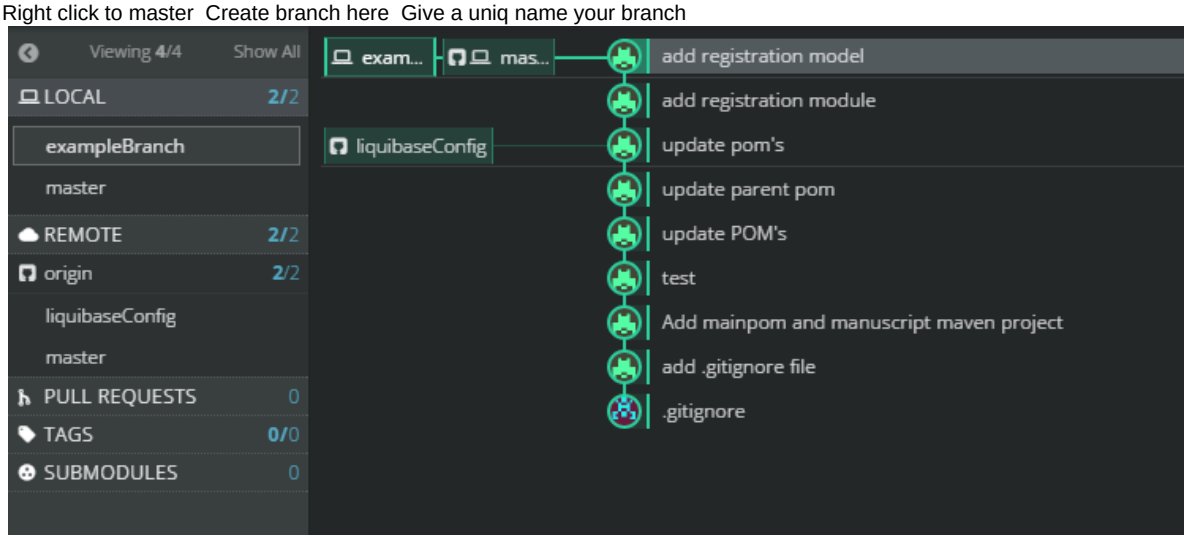
In this section we will discuss about push, commit and branch process.

Please **never** use the **master** branch to **push.** Always **create** your **own branch** with a unique name, and **use** it to **commit** and **push.** If you have **finished** your work, **merg** your branch **into** the **master** with carefully.

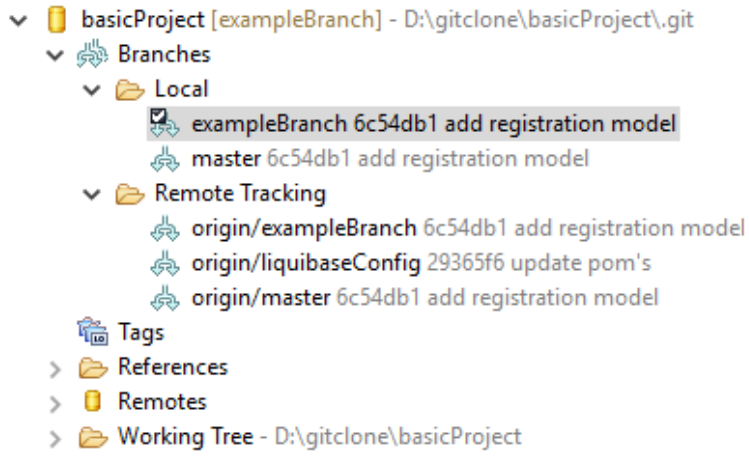
1. Download and install the GitKraken software.
2. Start the GitKraken and perform the tutorial.
3. Clone the git repository
    a. Use the **ctrl+n** shortcut.
    b. Fill the form as you can see below



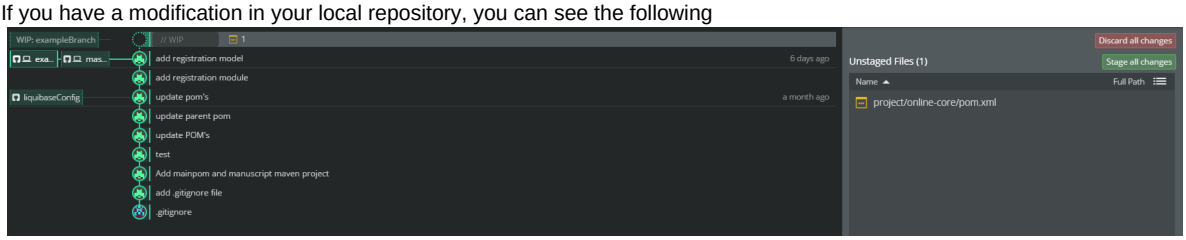
4. Import the project into the Eclipse
5. **Create a branch**
    a. Right click to master Create branch here Give a uniq name your branch



    b. Right click the newly create branch and Push Submit on the top of the window
    c. Now you pushed your branch to the remote repository
    d. Make sure you have selected your newly created local branch in Eclipse

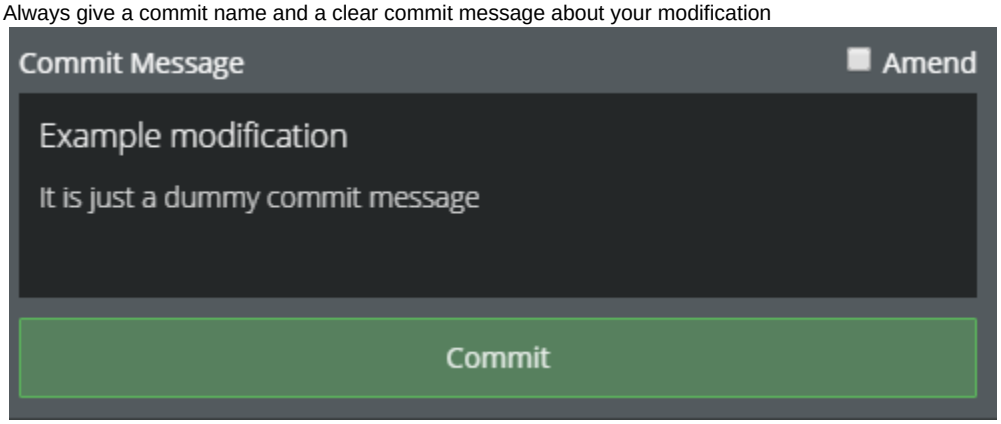


e. If you have a modification in your local repository, you can see the following

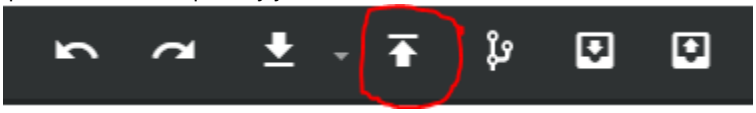


f. On the right side you need to accept or discard the changes

g. Always give a commit name and a clear commit message about your modification



h. Use the Commit button. It is not upload nothing the remote repository. It is visible your local repository. To finalize and upload the remote repository your modification use the Push button.

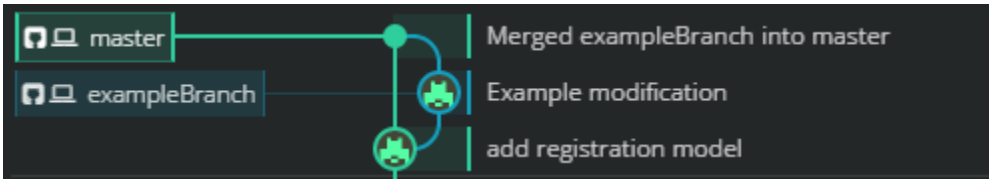


6. **Merging in git**

a. Make sure you have selected the branch that you want to merg with the master.

b. Right click on **master** and select the **Merge {yourBranchName} into master.** If there is no conflict between the branches you will not see error message.

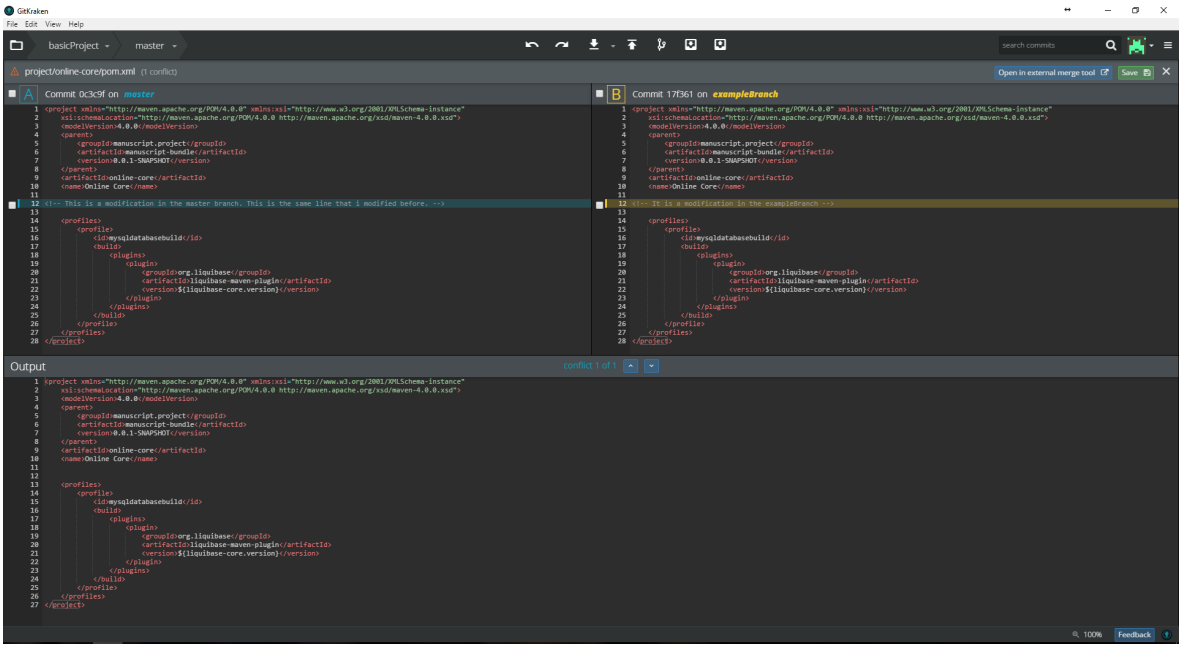
c. Select the **master** and **Push.**



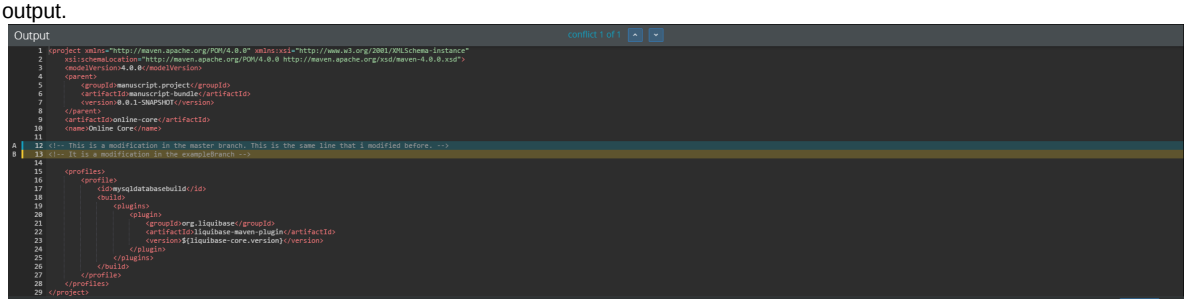
7. **Resolving a merge conflict**

a. I have created two push to remote repository. One push on master and one push on exampleBranch. I modified the same line.

b. If we would like to merge the **master** and the **exampleBranc** we should see **conflict.**



c. We can accept the modification A or B or both. I am going to accept the both modification. We can see the following on the output.



d. **Save Commit and Merge Push**

e. We can see the following in the remote repository

29 lines (27 sloc) | 945 Bytes — Raw | Blame | History

```xml
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
        <modelVersion>4.0.0</modelVersion>
        <parent>
                <groupId>manuscript.project</groupId>
                <artifactId>manuscript-bundle</artifactId>
                <version>0.0.1-SNAPSHOT</version>
        </parent>
        <artifactId>online-core</artifactId>
        <name>Online Core</name>

<!-- This is a modification in the master branch. This is the same line that i modified before. -->
<!-- It is a modification in the exampleBranch -->

        <profiles>
                <profile>
                        <id>mysqldatabasebuild</id>
                        <build>
                                <plugins>
                                        <plugin>
                                                <groupId>org.liquibase</groupId>
                                                <artifactId>liquibase-maven-plugin</artifactId>
                                                <version>${liquibase-core.version}</version>
                                        </plugin>
                                </plugins>
                        </build>
                </profile>
        </profiles>
</project>
```

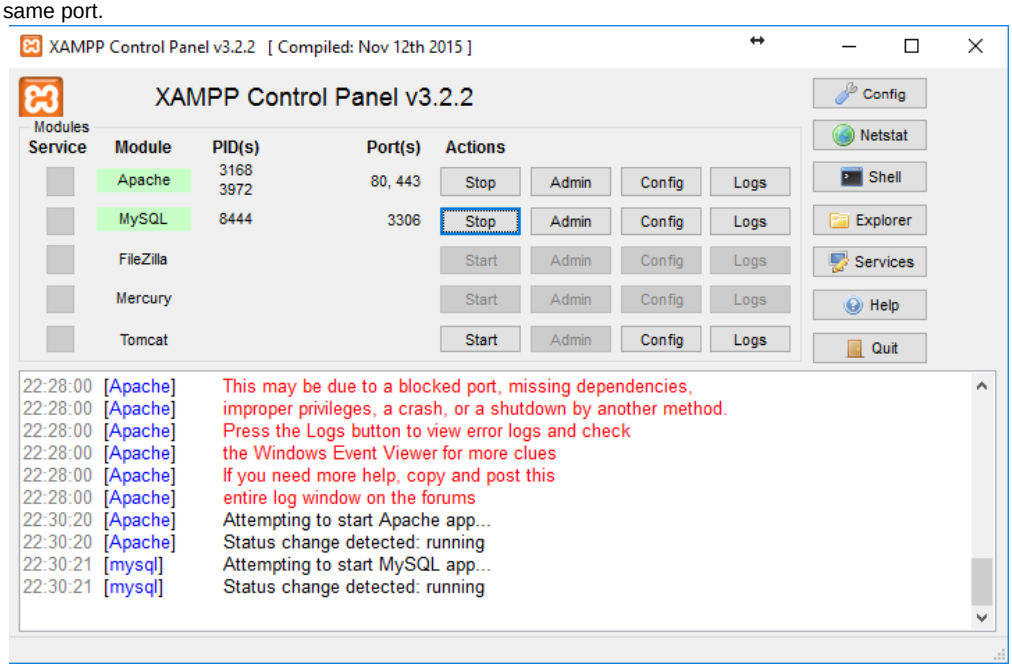
f. **Make sure you have not overwrite an another changes!**

and

XAMPP

In this section I will explane how to create mySQL database with XAMPP and how you can manage it with DbVisualizer. Furthermore I will speak about the liquibase maven plugin, settings.xml and its usage.

1. **XAMPP Control Panel and database creation**
   a. Start the Apache and MySQL modul. If you use Skype turn it off before start the Apache, because the Apache is using the same port.

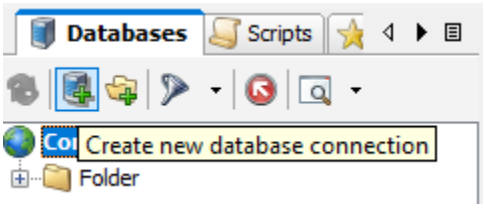


b. Use the **admin** button to navigate the **phpMyAdmin**. *Please devote time to study the panel.*
c. If you wish you can create new user on the admin panel.
d. Create the database. Example script:

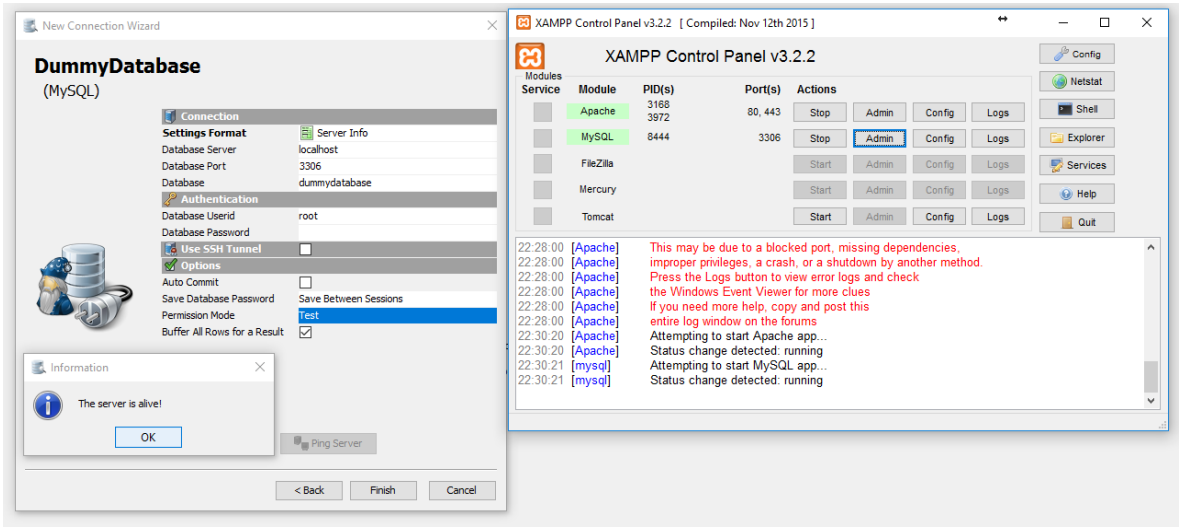
Create database

```
CREATE DATABASE IF NOT EXISTS DUMMYDATABASE;
USE DUMMYDATABASE;
```

2. **Using DbVisualizer**
   a. Create new database Use Wizard



   b. Give a name of you database
   c. Select the corresponding driver. In this case select MySql driver.
   d. Fill the form



3. **Liquibase maven plugin**