GitHub Repositories and Eclipse Setup
For Windows OS

Fork GitHub Repositories

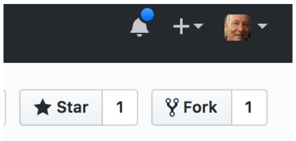
For this workshop you need to fork two GitHub repositories (see instructions below)

- https://github.com/sbl-sdsc/mmtf-spark
- https://github.com/sbl-sdsc/mmtf-workshop-2017

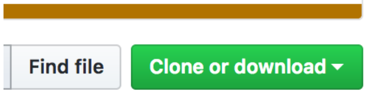
Forking will create your own copy of these two repositories in **your** GitHub account. See also https://help.github.com/articles/fork-a-repo/

Fork mmtf-spark

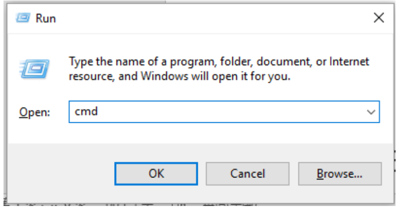
0. If your command shell supports git commands, skip this step.
   If not, you can install git using the following link:
   https://git-scm.com/download/win
1. Navigate to https://github.com/sbl-sdsc/mmtf-spark
2. In the top right corner, click Fork.



3. On GitHub, navigate to **your fork** of the mmtf-spark repository.
4. Under the repository name, click **Clone or download.**



5. In the Clone with HTTPs section, copy the clone URL for the repository.
6. Open command shell by pressing the **Windows** key and **R** and type “cmd”.



Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.

Open: cmd

7. Create a directory MMTF_Git in your home directory or location of your choice and cd into the directory by:

```
mkdir MMTF_Git
cd MMTF_Git
```

8. Create a clone of your fork, type

```
git clone https://github.com/<your github name>/mmtf-spark
```

Fork mmtf-workshop-2017

Now repeat this process for https://github.com/sbl-sdsc/mmtf-workshop-2017
Note, you already created the MMTF_Git directory, skip this step, but be sure to:

```
cd MMTF_Git
```

before you clone mmtf-workshop-2017

Create Eclipse Workspace

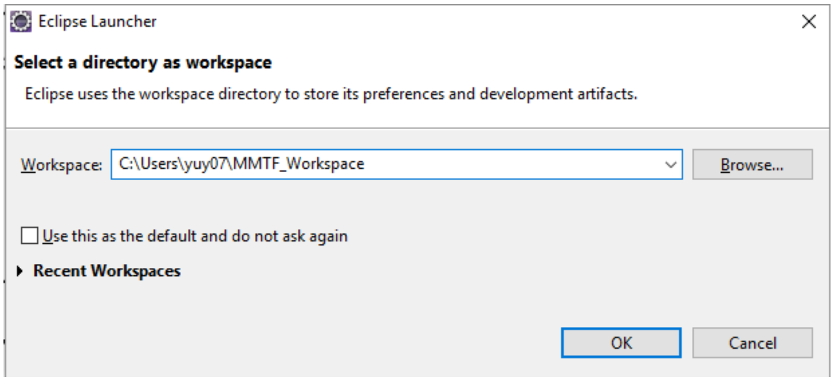
1. Open “cmd”.

2. In your home directory, or other location of your choice, create a directory called MMTF_Workspace

```
cd < directory >
mkdir MMTF_Workspace
```

3. Launch Eclipse.

4. When Eclipse starts up, it prompts for a workspace directory, click browse and navigate to MMTF_Workspace. Then click ok.



Select a directory as workspace

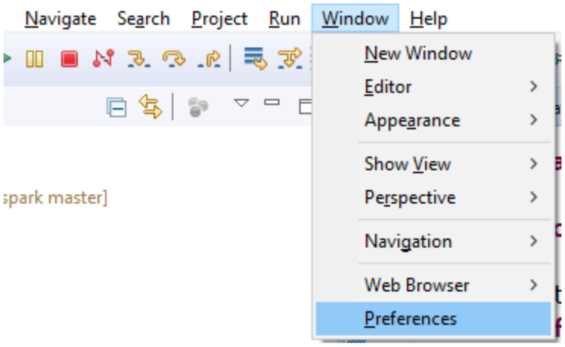
Eclipse uses the workspace directory to store its preferences and development artifacts.

Workspace: C:\Users\yuy07\MMTF_Workspace

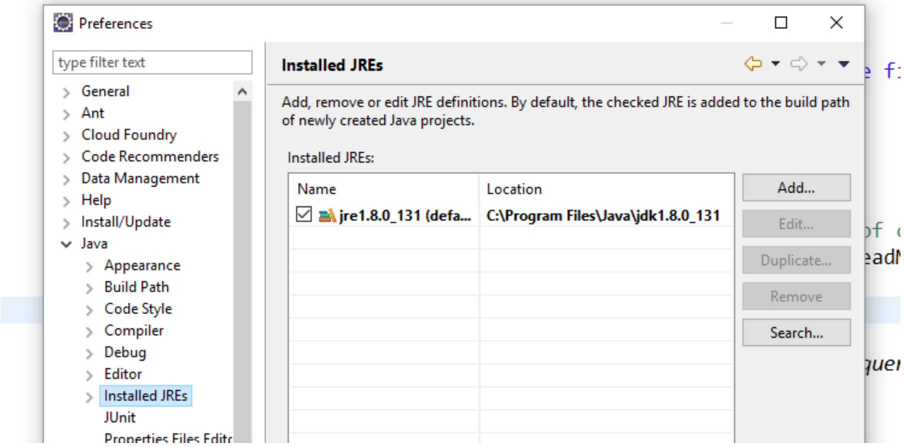
□ Use this as the default and do not ask again

Set Eclipse Preferences

1. In the Eclipse top menu bar, navigate to: Window > Preferences



2. In the Preferences window, expand Java and click on Installed JREs.



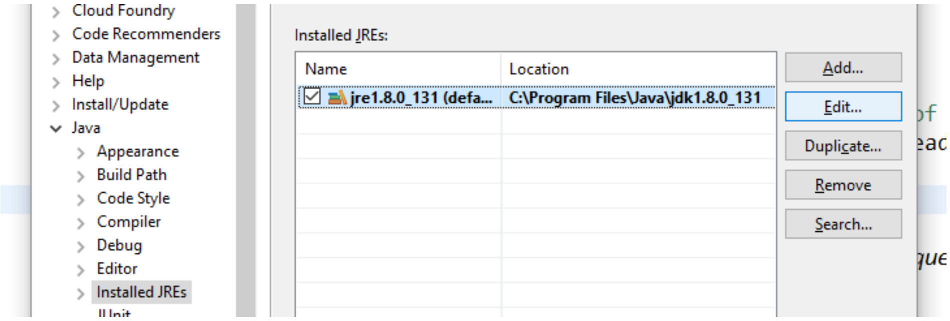
Installed JREs

Add, remove or edit JRE definitions. By default, the checked JRE is added to the build path of newly created Java projects.

Installed JREs:

Name	Location
☑ jre1.8.0_131 (defa...	C:\Program Files\Java\jdk1.8.0_131

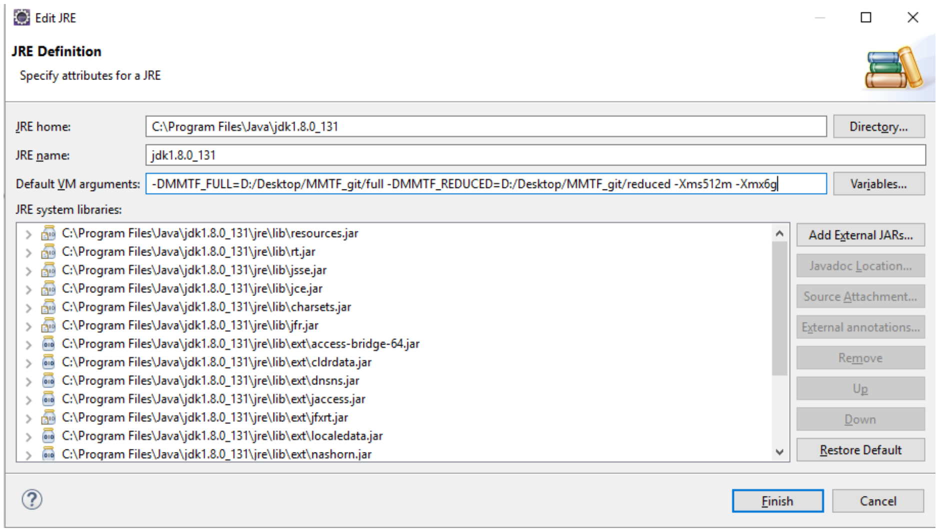
3. Click on the latest version of the Java JDK (1.8.0_131) to select it (blue highlight) and then click the Edit button.



Installed JREs:

Name	Location
☑ jre1.8.0_131 (defa...	C:\Program Files\Java\jdk1.8.0_131

4. Under the Default VM arguments enter the path to your full and reduced Hadoop Sequence files as shown below and add the -Xms512m and -Xmx6g memory options like this:
   -DMMTF_FULL=< *directory to full* > -DMMTF_REDUCED=< *directory to reduced* > -Xms512m -Xmx6g
   Note that only 64bits system supports "-Xmx6g", so be sure you installed the java(64bits) and eclipse(64bits).



Edit JRE

JRE Definition

Specify attributes for a JRE

JRE home: C:\Program Files\Java\jdk1.8.0_131

JRE name: jdk1.8.0_131

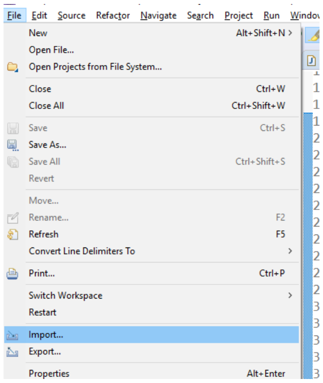
Default VM arguments: -DMMTF_FULL=D:/Desktop/MMTF_git/full -DMMTF_REDUCED=D:/Desktop/MMTF_git/reduced -Xms512m -Xmx6g

JRE system libraries:

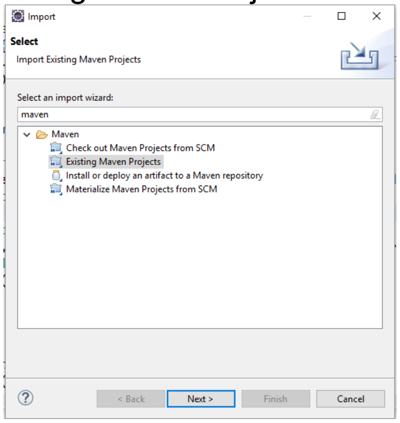
```
C:\Program Files\Java\jdk1.8.0_131\jre\lib\resources.jar
C:\Program Files\Java\jdk1.8.0_131\jre\lib\rt.jar
C:\Program Files\Java\jdk1.8.0_131\jre\lib\jsse.jar
C:\Program Files\Java\jdk1.8.0_131\jre\lib\jce.jar
C:\Program Files\Java\jdk1.8.0_131\jre\lib\charsets.jar
C:\Program Files\Java\jdk1.8.0_131\jre\lib\jfr.jar
C:\Program Files\Java\jdk1.8.0_131\jre\lib\ext\access-bridge-64.jar
C:\Program Files\Java\jdk1.8.0_131\jre\lib\ext\cldrdata.jar
C:\Program Files\Java\jdk1.8.0_131\jre\lib\ext\dnsns.jar
C:\Program Files\Java\jdk1.8.0_131\jre\lib\ext\jaccess.jar
C:\Program Files\Java\jdk1.8.0_131\jre\lib\ext\jfxrt.jar
C:\Program Files\Java\jdk1.8.0_131\jre\lib\ext\localedata.jar
C:\Program Files\Java\jdk1.8.0_131\jre\lib\ext\nashorn.jar
```

Import mmtf-spark into Eclipse

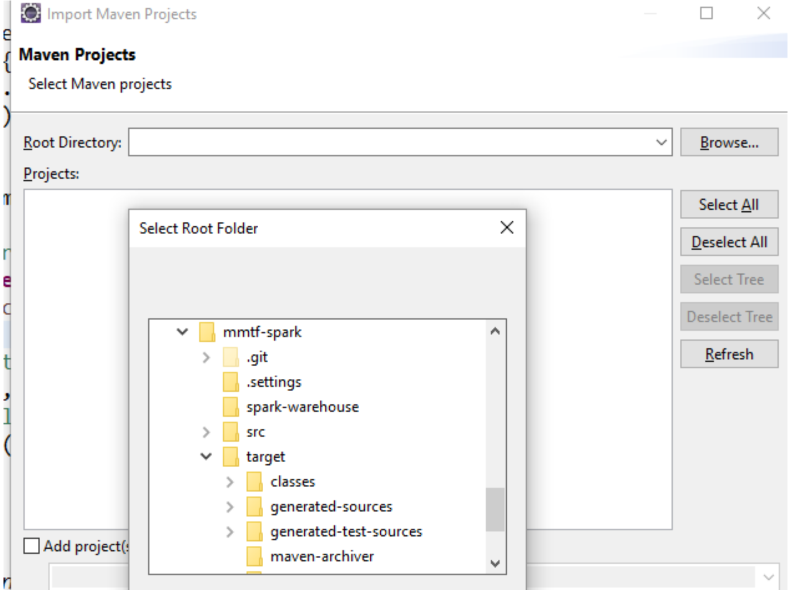
1. Select Import from the File menu



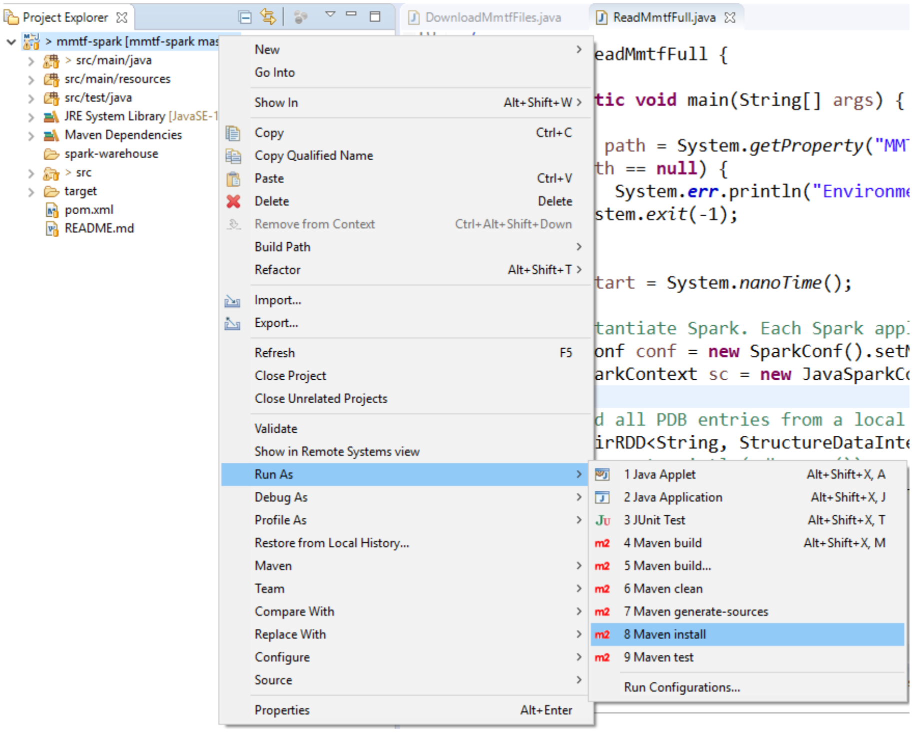
2. Navigate to Maven > Existing Maven Project and click Next.


Import
Select
Import Existing Maven Projects
Select an import wizard:
maven
Maven
Check out Maven Projects from SCM
Existing Maven Projects
Install or deploy an artifact to a Maven repository
Materialize Maven Projects from SCM
< Back
Next >
Finish
Cancel

3. Click on Browse, navigate to your MMTF_Git directory, select mmtf-spark, and click open.



4. In Eclipse, right click on imported mmtf-spark project and select Run As > Maven Install from the menu.



```
eadMmtfFull {

tic void main(String[] args) {

 path = System.getProperty("MMT
th == null) {
  System.err.println("Environme
stem.exit(-1);


tart = System.nanoTime();

tantiate Spark. Each Spark app
onf conf = new SparkConf().setM
arkContext sc = new JavaSparkCo

d all PDB entries from a local
irRDD<String, StructureDataInte
```

5. Maven will install all dependencies and run unit tests on this project. This may take several minutes.
6. There should be a Build success message in the Eclipse Console if the installation succeeded.

Import mmtf-workshop-2017 into Eclipse

Now follow the same steps to import mmtf-workshop-2017 into Eclipse.

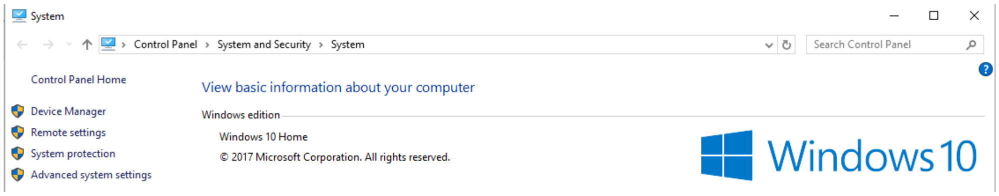
In Eclipse, right click **on each of the mmtf-workshop-2017 projects** and Run As > Maven Install from the menu.

Running Spark on Windows

The official Hadoop release does not include the native libraries to work properly on Windows. So when you try to run spark on your Windows machine, you may receive a java.lang.NullPointerException. This issue is caused by a missing winutils.exe file that Spark needs to use hadoop functionality.

To resolve this problem, you need to:

1. Download Hadoop 2.6.0 compiled for Windows(64-bit) from here:
   http://static.barik.net/software/hadoop-2.6.0-dist/hadoop-2.6.0.tar.gz
   For information about this issue see:
   https://www.barik.net/archive/2015/01/19/172716/
   NOTE: there is a different winutils.exe file for the 64-bit Windows and it will not work on the 32-bit Windows.
2. Unzip the archive and copy them into a folder like C:\hadoop\
   You can download 7zip to unzip and untar the file if you don't have any unzip software:
   http://www.7-zip.org/download.html
3. Set up HADOOP_HOME environment variable pointing to the above directory (without bin).
   For example: if you copied the winutils.exe to C:\hadoop\bin, set HADOOP_HOME to C:\hadoop
   If you are not familiar with how to set an environment variable, follow these steps, else, skip to step 4:
   - Open "cmd".
   - Type "control system" and press enter. You should see something like this:



System

Control Panel > System and Security > System

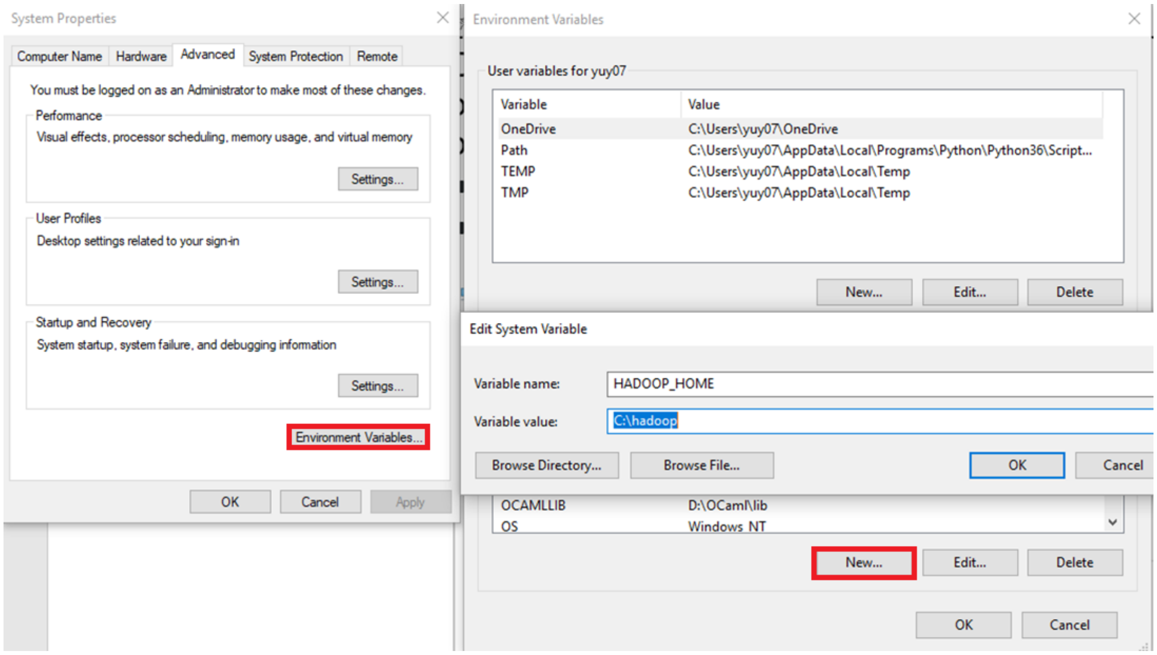
View basic information about your computer

Windows edition

Windows 10 Home

© 2017 Microsoft Corporation. All rights reserved.

   - Click on Advanced system settings > Environment Variables > New, and add the new environment variable.



System Properties

You must be logged on as an Administrator to make most of these changes.

Performance: Visual effects, processor scheduling, memory usage, and virtual memory

User Profiles: Desktop settings related to your sign-in

Startup and Recovery: System startup, system failure, and debugging information

Environment Variables

User variables for yuy07

Variable	Value
OneDrive	C:\Users\yuy07\OneDrive
Path	C:\Users\yuy07\AppData\Local\Programs\Python\Python36\Script...
TEMP	C:\Users\yuy07\AppData\Local\Temp
TMP	C:\Users\yuy07\AppData\Local\Temp

Edit System Variable

Variable name: HADOOP_HOME

Variable value: C:\hadoop

OCAMLLIB	D:\OCaml\lib
OS	Windows_NT

4. If you are running your Spark application in Eclipse, restart Eclipse