Setup guide to Install Apache Spark on Windows

**Introduction**

Apache Spark is an open-source framework that processes large volumes of stream data from multiple sources. Spark is used in distributed computing with machine learning applications, data analytics, and graph-parallel processing.

Apache Spark is initially written in a Java Virtual Machine(JVM) language called Scala, whereas Pyspark is like a Python API which contains a library called Py4J. This allows dynamic interaction with JVM objects.

This guide will show you **how to install Apache Spark on Windows 10**and test the installation**.**

**Prerequisites**

* A system running Windows 10
* A user account with administrator privileges (required to install software, modify file permissions, and modify system PATH)
* Command Prompt or Powershell
* A tool to extract .tar files, such as 7-Zip
* Eclipse Pre-installed.

**Install Apache Spark on Windows**

If you already have Java 8 and Python 3 installed, you can skip the first two steps.

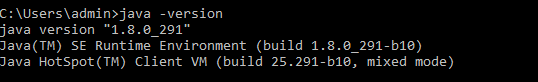
Apache Spark requires Java 8. You can check to see if Java is installed using the command prompt.

Open the command line by clicking **Start** > type *cmd* > click **Command Prompt**.

Type the following command in the command prompt:

java -version

If Java is installed, it will respond with the following output:



If you don’t have Java installed:

## Java installation

## Step1. Visit Oracle's website for the download of the Java Development Kit(JDK).

## Link - [Java SE Development Kit 8 - Downloads | Oracle India](https://www.oracle.com/in/java/technologies/javase/javase-jdk8-downloads.html)

## Step2. Open the installer file, and the download begins

## 

## Step3. Go to "Command Prompt" and type "java -version" to know the version and know whether it is installed or not.

## 

## Step4. Add the Java path

## 

## Step5. Go to the search bar and "EDIT THE ENVIRONMENT VARIABLES.

## 

## Step6. Click into the "Environment Variables'

## 

## Step7. Click into "New" to create your new Environment variable.

## 

## Step8. Use Variable Name as "JAVA\_HOME' and your Variable Value as 'C:\Program Files (x86)\Java\jdk1.8.0\_251'. This is your location of the Java file. Click 'OK' after you've finished the process.

## 

## Step9. Let's add the User variable and select 'Path' and click 'New' to create it

## 

## Step10. Add the Variable name as 'PATH' and path value as 'C:\Program Files (x86)\Java\jdk1.8.0\_251\bin', which is your location of Java bin file. Click 'OK' after you've finished the process.

## 

## Python Installation

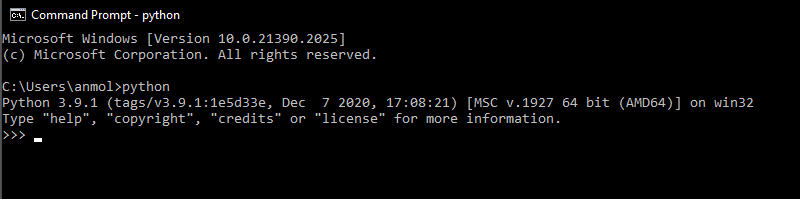
Apache Spark requires python. You can check to see if python is installed using the command prompt.

Open the command line by clicking **Start** > type *cmd* > click **Command Prompt**.

Type the following command in the command prompt:

python

If python is installed, it will respond with the following output:

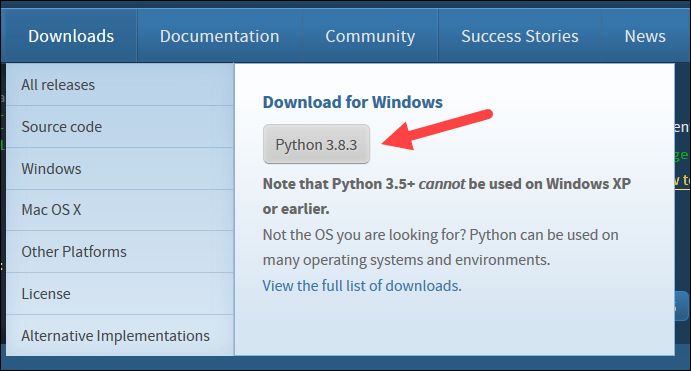


If Python is not installed then follow the following steps:

### **Step11.** Install Python

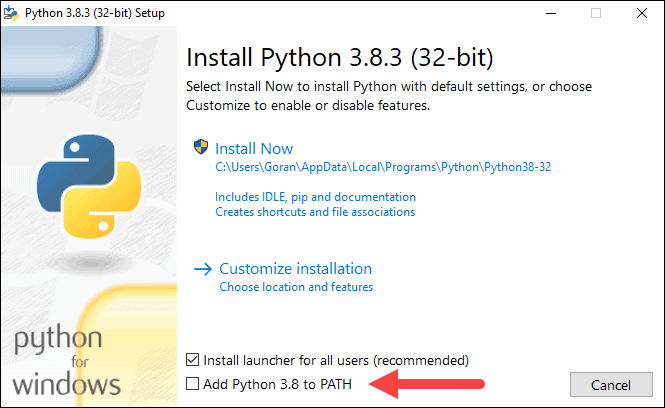
1. To install the Python package manager, navigate to <https://www.python.org/> in your web browser.

2. Mouse over the **Download** menu option and click **Python 3.8.3**. 3.8.3. is the latest version at the time of writing the article.



3. Once the download finishes, run the file.

**Step12.** Near the bottom of the first setup dialog box, check off Add Python 3.8 to PATH. Leave the other box checked.



**Step13.** Click **Install Now**, and let the installation complete.

If you have a command prompt open, restart it. Verify the installation by checking the version of Python:

python --version

The output should print **Python 3.8.3**.

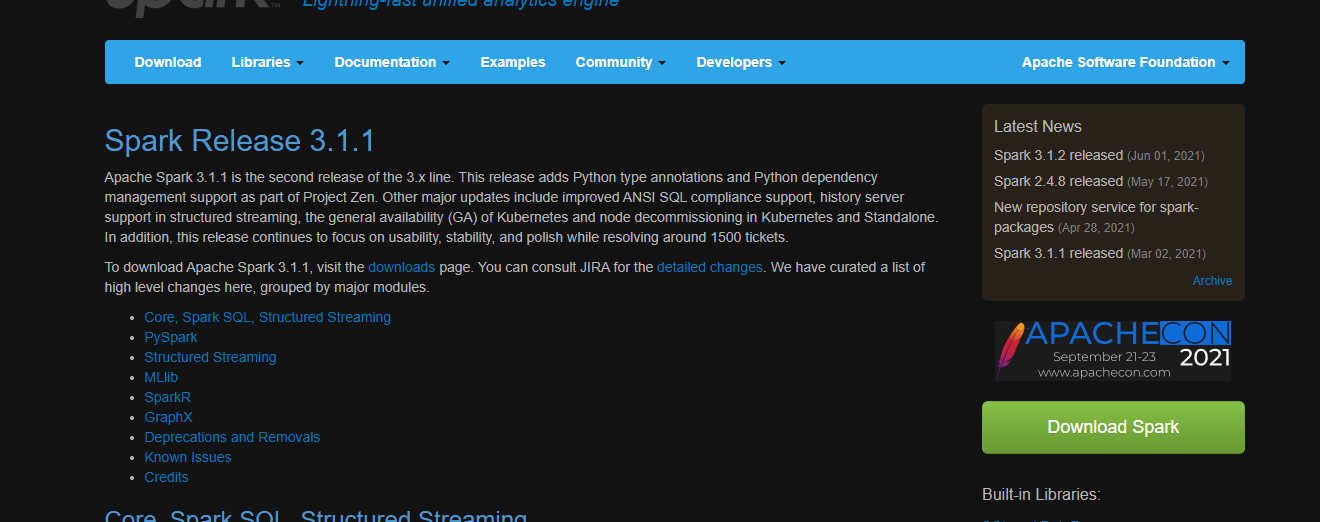
### **Download Apache Spark**

**Step14**. Head over to the [Spark homepage](https://spark.apache.org/downloads.html).

**Step15.** Select the Spark release and package type as following and download the .tgz file.

In our case, I am using ***Spark release “spark-3.1.1-bin-hadoop2.7”.***

You can download spark version according to your project requirements.

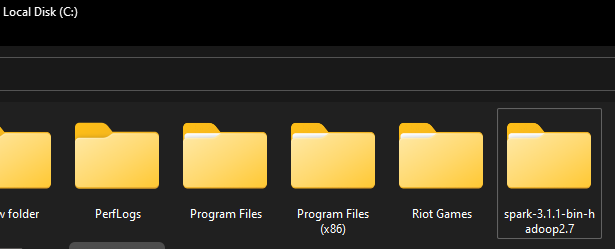


You can make a new folder called 'spark' in the C directory and extract the given file by using ‘7-zip’, which will be helpful afterward.

**Or**

You can download spark from the link provided and extract that folder into your C:/ drive.

Link: <https://drive.google.com/file/d/1JsbC7gRt0Jw_vDWzSVoLQMzFcmAPo_nb/view?usp=sharing>



TO check spark version in windows

Cd to C:/ directory

And go to cd spark-3.1.1-hadoop-2.7

Cd to bin folder

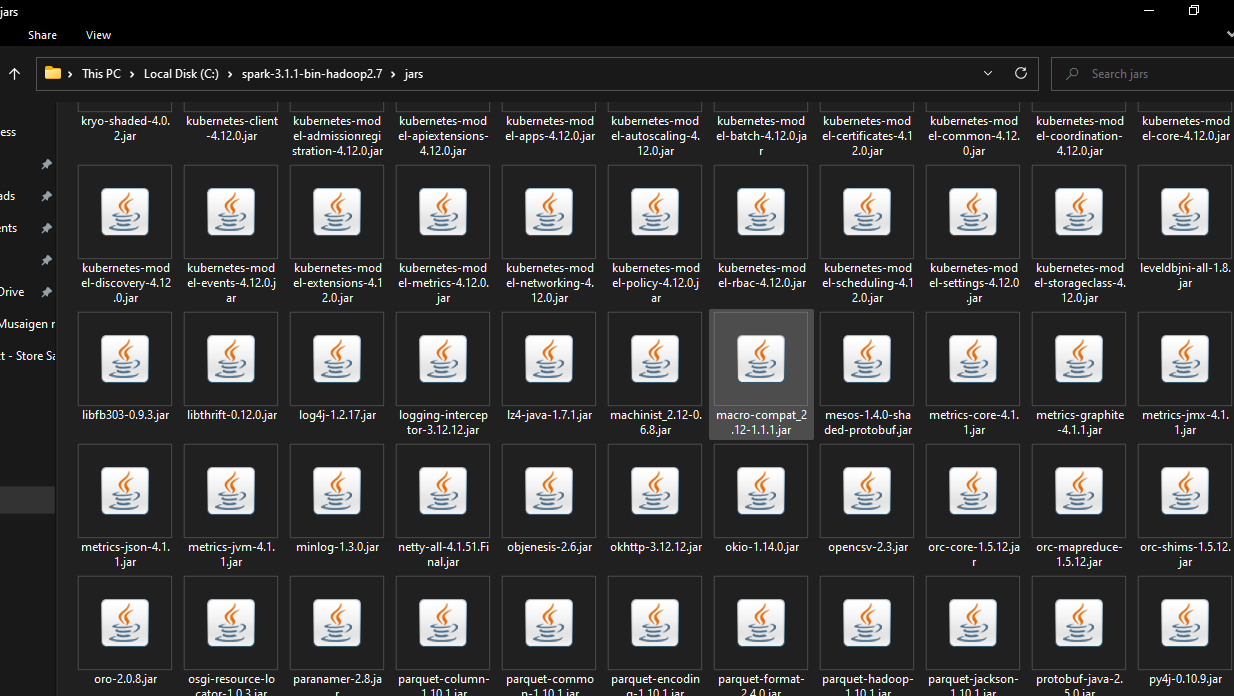
And type the command

Spark-shell

**Step16.** Now download sqljdbc42.jar, postgresql-42.2.20.jre7.jar, mssql-jdbc-7.2.0.jre8.jar. Paste this jar files at this path “C:\spark-3.1.1-bin-hadoop2.7\jars”

Link : https://drive.google.com/file/d/1R6XBzLfm1m-lc\_95Ag1RjAql3fTGf9o\_/view?usp=sharing

1. sqljdbc42.jar is required for SQL connection with spark.
2. postgresql-42.2.20.jre7.jar is required for Postgre SQL connection with spark.
3. mssql-jdbc-7.2.0.jre8.jar is required for mysql connection with spark.

****

**Step17**. Download winutils.exe and paste that file in

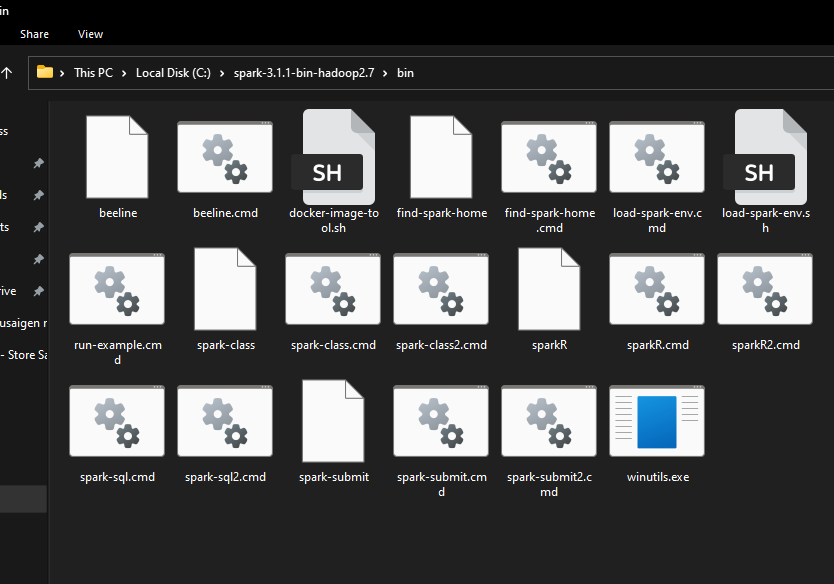
path : C:\spark-3.1.1-bin-hadoop2.7\bin

https://drive.google.com/file/d/1knb1ov87BgpqQLj7u9uXfeImGacHxWZ1/view?usp=sharing

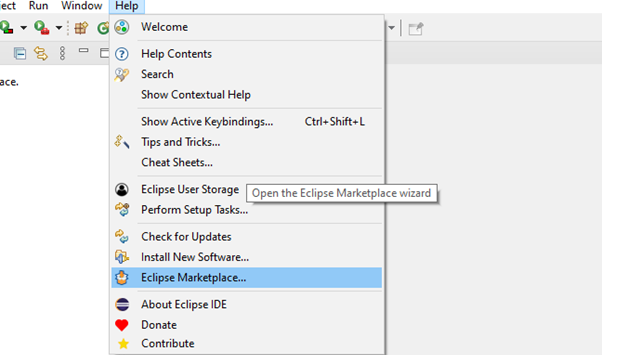
Hadoop requires native libraries on Windows to work properly -that includes accessing the file:// filesystem, where Hadoop uses some Windows APIs to implement postfix like file access permissions.

This is implemented in HADOOP.DLL and WINUTILS.EXE.

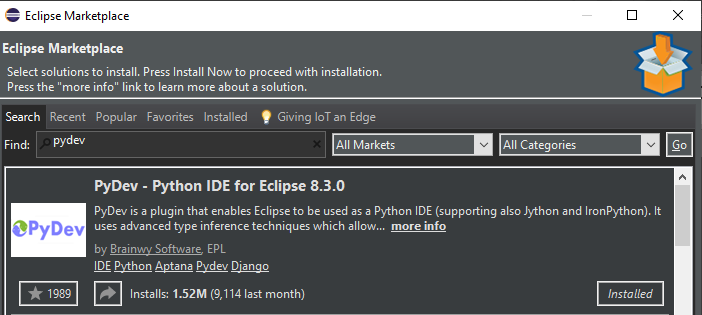
In particular, %HADOOP\_HOME%\BIN\WINUTILS.EXE must be locatable



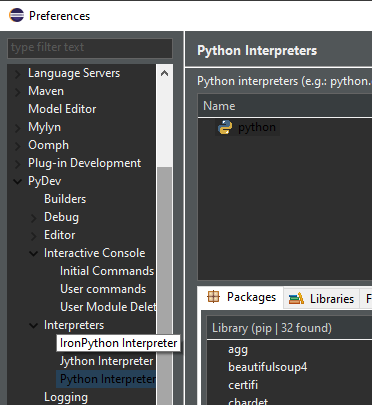
**Step18.** Now**,** open Eclipse in your system and click on help in the status bar >> Click on Eclipse Marketplace



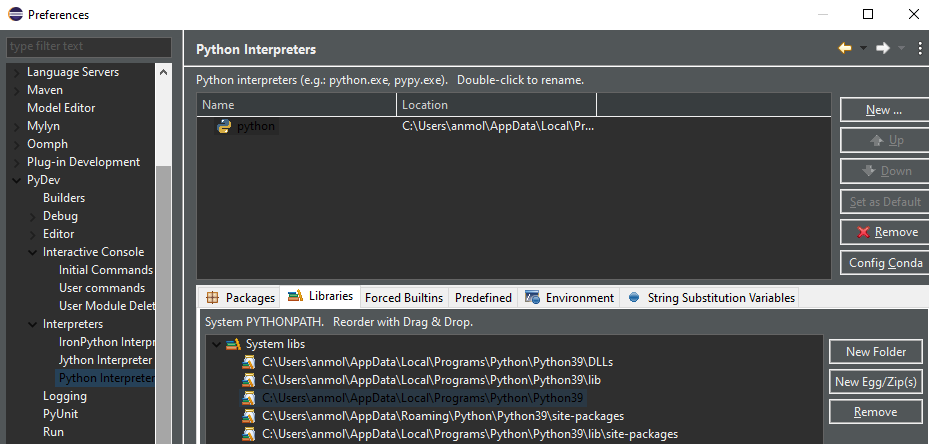
**Step19**. Search for “pydev” in eclipse marketplace and install it.



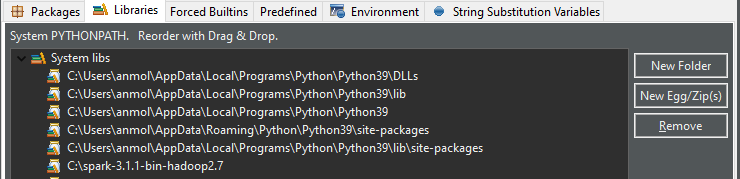
**Step20.** Once pydev is installed, go to preferences >> PyDev >> Interpreters >> Python Interpreter.



**Step21**. Go to Libraries section and make sure these files are installed. Go to new in python interpreters and select python.exe

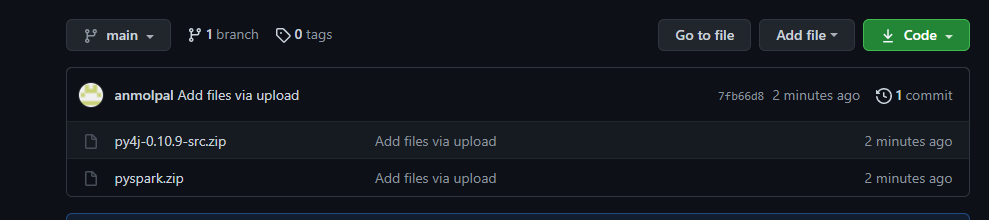


**Step22**. Now add the spark directory path in Libraries. On the right-hand side, click on New Folder and add the path of the folder.

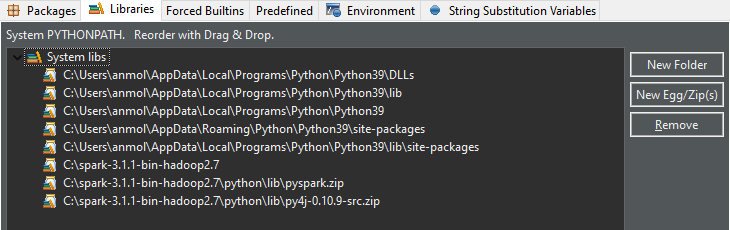


**Step23.** Download pyspark.zip and py4j-0.10.9-src.zip from the below mentioned GitHub link

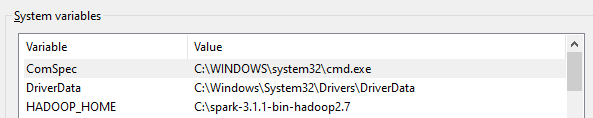
**https://github.com/anmolpal/Zip.git**

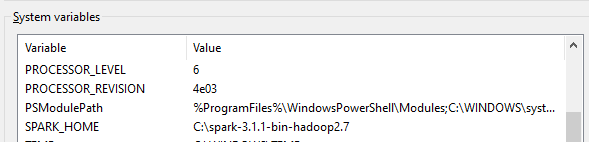
****

**Step24.** Now on the right-hand side there is an option “New Egg/Zip”. Click on it and add the two downloaded zip files into your Libraries section.

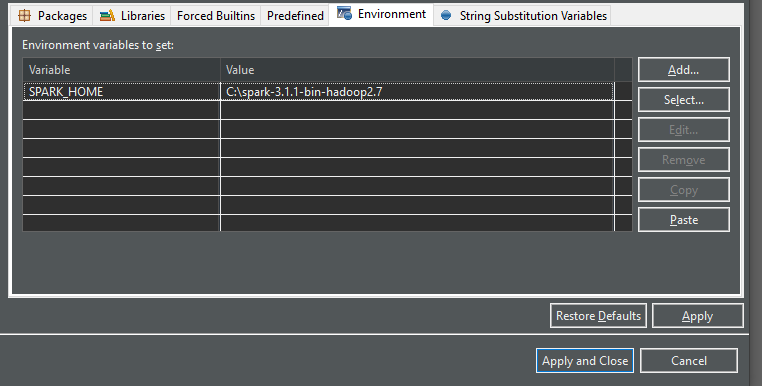


**Step25**. Go back to the environment variables in your windows and add SPARK\_HOME and HADOOP\_HOME path.





**Step26**. Now get back to your eclipse IDE and go to the Environment section and add “SPARK\_HOME” path in it.



And that’s it spark is setup and ready to use.

**Troubleshooting:**

**If this error occurs >>**

C:\spark>spark-shell

Exception in thread "main" java.lang.ExceptionInInitializerError

at org.apache.hadoop.util.StringUtils.<clinit>(StringUtils.java:80)

at org.apache.hadoop.security.SecurityUtil.getAuthenticationMethod(SecurityUtil.java:611)

at org.apache.hadoop.security.UserGroupInformation.initialize(UserGroupInformation.java:273)

at org.apache.hadoop.security.UserGroupInformation.ensureInitialized(UserGroupInformation.java:261)

at org.apache.hadoop.security.UserGroupInformation.loginUserFromSubject(UserGroupInformation.java:791)

at org.apache.hadoop.security.UserGroupInformation.getLoginUser(UserGroupInformation.java:761)

at org.apache.hadoop.security.UserGroupInformation.getCurrentUser(UserGroupInformation.java:634)

at org.apache.spark.util.Utils$$anonfun$getCurrentUserName$1.apply(Utils.scala:2464)

at org.apache.spark.util.Utils$$anonfun$getCurrentUserName$1.apply(Utils.scala:2464)

at scala.Option.getOrElse(Option.scala:121)

at org.apache.spark.util.Utils$.getCurrentUserName(Utils.scala:2464)

at org.apache.spark.SecurityManager.<init>(SecurityManager.scala:222)

at org.apache.spark.deploy.SparkSubmit$.secMgr$lzycompute$1(SparkSubmit.scala:393)

at org.apache.spark.deploy.SparkSubmit$.org$apache$spark$deploy$SparkSubmit$$secMgr$1(SparkSubmit.scala:393)

at org.apache.spark.deploy.SparkSubmit$$anonfun$prepareSubmitEnvironment$7.apply(SparkSubmit.scala:401)

at org.apache.spark.deploy.SparkSubmit$$anonfun$prepareSubmitEnvironment$7.apply(SparkSubmit.scala:401)

at scala.Option.map(Option.scala:146)

at org.apache.spark.deploy.SparkSubmit$.prepareSubmitEnvironment(SparkSubmit.scala:400)

at org.apache.spark.deploy.SparkSubmit$.submit(SparkSubmit.scala:170)

at org.apache.spark.deploy.SparkSubmit$.main(SparkSubmit.scala:136)

at org.apache.spark.deploy.SparkSubmit.main(SparkSubmit.scala)

Caused by: java.lang.StringIndexOutOfBoundsException: begin 0, end 3, length 2

at java.base/java.lang.String.checkBoundsBeginEnd(String.java:3107)

at java.base/java.lang.String.substring(String.java:1873)

at org.apache.hadoop.util.Shell.<clinit>(Shell.java:52)

... 21 more

Solution : Check the Java version you are using. Install Java 8 and this issue will be resolved.

**PyDev not available:** Pydev is not available for older versions of eclipse. Download the latest version of eclipse from the link below.

[Eclipse Downloads | The Eclipse Foundation](https://www.eclipse.org/downloads/)