Spark/Pyspark Setup with Windows

# Python Setup

Install PyCharm Community Edition is fine: <https://www.jetbrains.com/pycharm/download/download-thanks.html?platform=windowsARM64&code=PCC>

Setup JAVA and JDK and set the class path and hadoop dlls as mentioned below.

In the intellij - virtual env.

**pip install pyspark**

## Issues:

Pyspark python not found issue: pyspark not able to connect back

<https://stackoverflow.com/questions/53252181/python-worker-failed-to-connect-back>

Set Env PYSPARK\_PYTHON=python To Fix It.

### No module named numpy:

Open the virtualenv and run: pip install pandas

### Distutils not found:

This happens while running pyspark mllib

<https://stackoverflow.com/questions/69919970/no-module-named-distutils-util-but-distutils-installed>

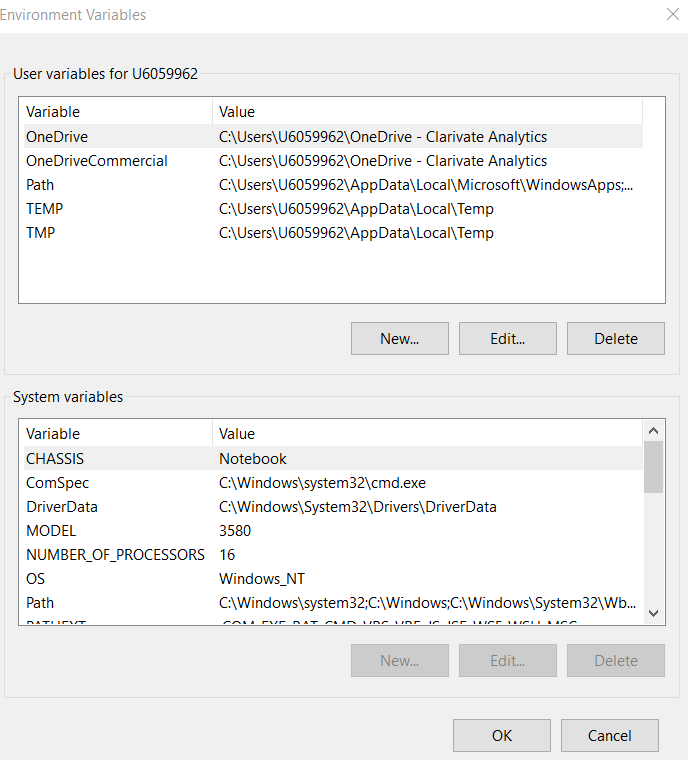
Open the virtualenv and run: pip install setuptools

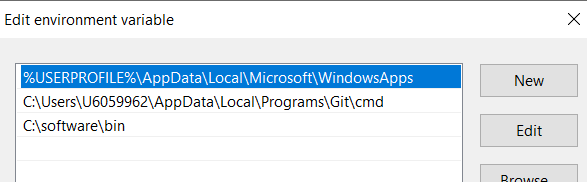
# Java Setup:

OpenJDK17 - windows build

<https://learn.microsoft.com/en-us/java/openjdk/download#openjdk-17>

Download the zip file. Unzip into any directory like C:\software

1. <https://github.com/donotech/dash_train/tree/master/hadooputils>: Download this directory to any local folder e.g. C:\software
2. Note: The original versions are here: [winutils/hadoop-3.0.0/bin at master · steveloughran/winutils · GitHub](https://github.com/steveloughran/winutils/tree/master/hadoop-3.0.0/bin). However no company allows downloading of exe or dll files. These files are renamed to bypass the firewall
3. Using command prompt go to the directory e.g. cd C:\Software\hadooputils
4. Rename the child directory bin-Copy to bin
5. Inside the bin directory rename like: ren hadoop.dll.txt hadoop.dll
6. Inside the bin directory rename like: ren winutils.exe.txt winutils.exe
7. So new directory structure would be C:\software\hadooputils\bin\{winutils.exe and hadoop.dll}
8. put hadoop.dll file into the C:/Windows/System32 folder
9. Open the user env setup in windows. For this search env and open the “change environment for this user” (don't open system user)
10. 
11. Change the path to add up to the bin directory in it. Like C:\Software\hadooputils\bin



1. Restart your pycharm/intellij
2. In your python spark program you need to add the JAVA path and HADOOP\_HOME like

os.environ['JAVA\_HOME'] = "C:\\software\\microsoft-jdk-17.0.10-windows-x64\\jdk-17.0.10+7"

os.environ['HADOOP\_HOME'] = 'C:\\software\\winutils'

## Spark Code Env Variables in Windows Machine

In Windows machine we need the Winutils to read and write

In the code set it up:

val winutilPath = "C:\\softwares\\hadoop3\_winutils" //"C:\\softwares\\winutils" //\\bin\\winutils.exe"; //bin\\winutils.exe";

if (System.*getProperty*("os.name").toLowerCase.contains("win")) {

System.*out*.println("Detected windows")

System.*setProperty*("hadoop.home.dir", winutilPath)

System.*setProperty*("HADOOP\_HOME", winutilPath)

}

## 

## Scala Build

Build in Scala SBT: In the build.sbt

ThisBuild / *version* := "0.1.0-SNAPSHOT"

ThisBuild / *scalaVersion* := "2.13.12"

lazy val root = (project in *file*("."))

.settings(

*name* := "sparkl2scala",

*idePackagePrefix* := Some("com.bdec.training.sparkscala")

)

*libraryDependencies* += "org.apache.spark" %% "spark-core" % "3.5.0"

*libraryDependencies* += "org.apache.spark" %% "spark-sql" % "3.5.0"

*libraryDependencies* += "org.apache.spark" %% "spark-sql-kafka-0-10" % "3.5.0"

*libraryDependencies* += "org.apache.spark" %% "spark-mllib" % "3.5.0"

## Gradle Build

For gradle in build.gradle:

dependencies **{**

implementation 'com.google.guava:guava:29.0-jre'

testImplementation 'junit:junit:4.13'

implementation 'org.apache.spark:spark-sql\_2.12:3.0.0'

implementation 'org.apache.spark:spark-mllib\_2.12:3.0.0'

//testImplementation 'org.junit.jupiter:junit-jupiter-engine:5.8.2'

**}**

# 

# Issues in JDK Setup

Download JDK 17

<https://www.oracle.com/java/technologies/javase/jdk17-archive-downloads.html>

Add this to your intellij

You may get an illegal accessor issue: <https://stackoverflow.com/questions/69814385/exception-in-thread-main-java-lang-illegalaccesserror-class-org-apache-spark>

In the run configs add

--add-exports java.base/sun.nio.ch=ALL-UNNAMED

### NIO issues with Java 17

<https://stackoverflow.com/questions/67782975/how-to-fix-the-module-java-base-does-not-opens-java-io-to-unnamed-module>

org.gradle.jvmargs=-Xmx1536M \ --add-exports=java.base/sun.nio.ch=ALL-UNNAMED \ --add-opens=java.base/java.lang=ALL-UNNAMED \ --add-opens=java.base/java.lang.reflect=ALL-UNNAMED \ --add-opens=java.base/java.io=ALL-UNNAMED \ --add-exports=jdk.unsupported/sun.misc=ALL-UNNAMED

Add in JVM options in run configuration in IntelliJ as well

--add-exports=java.base/sun.nio.ch=ALL-UNNAMED --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.lang.reflect=ALL-UNNAMED --add-opens=java.base/java.io=ALL-UNNAMED --add-exports=jdk.unsupported/sun.misc=ALL-UNNAMED

## Unsatisfied Link Error

<https://sparkbyexamples.com/spark/spark-hadoop-exception-in-thread-main-java-lang-unsatisfiedlinkerror-org-apache-hadoop-io-nativeio-nativeiowindows-access0ljava-lang-stringiz/>

# 

# Running Kafka on Windows

<https://www.geeksforgeeks.org/how-to-install-and-run-apache-kafka-on-windows/>

Java 11 has to be installed.

<https://archive.apache.org/dist/kafka/2.8.2/>

Download the kafka2.8 tgz file: [kafka\_2.13-2.8.2.tgz](https://archive.apache.org/dist/kafka/3.0.0/kafka_2.13-3.0.0.tgz)

**(Do not use kafka 3 as it is not compatible with windows)**

Untar it

The properties files are under the config directory

Change the zookeeper.properties for data dir

dataDir=C:/softwares/kafka\_2.13-2.8.2/zkdata

Change the server.properties for log dir

log.dirs=C:/softwares/kafka\_2.13-2.8.2/kafka-logs

set JAVA\_HOME=C:\softwares\jdk-11.0.18

set PATH=%PATH%;%JAVA\_HOME%\bin

Start zookeeper

.\bin\windows\zookeeper-server-start.bat .\config\zookeeper.properties

In another terminal start kafka broker

.\bin\windows\kafka-server-start.bat .\config\server.properties

If you see an error like - kafka joining a wrong cluster or some other kafka leader election related errors then please delete the kafka logs folder

## Test the setup

Go into windows dir under kafka\bin

c:\softwares\kafka\_2.13-2.8.2\bin\windows>kafka-topics.bat --bootstrap-server localhost:9092 --create --topic socgenl2 --partitions 1

Created topic socgenl2.

c:\softwares\kafka\_2.13-2.8.2\bin\windows>kafka-console-producer.bat --bootstrap-server localhost:9092 --topic socgenl2

>hello world

>bye world

>Terminate batch job (Y/N)? y

c:\softwares\kafka\_2.13-2.8.2\bin\windows>kafka-console-consumer.bat --bootstrap-server localhost:9092 --topic socgenl2