SPARK INSTALLATION GUIDE

Guide Overview

The following document aims to guide you through Spark installation (python and Scala), both in Linux and Windows.

The installation process consists of:

- Checking out the prerequisites
- Installing Spark
- Installing Jupyter Notebook
- Installing Scala Kernel on Jupyter

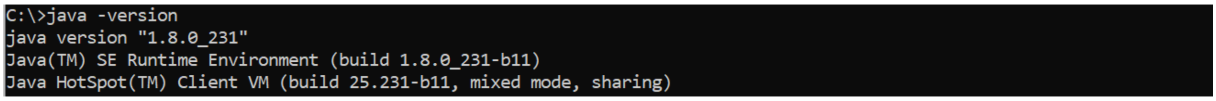
Prerequisites

Java

To know if java is already installed in your system run the following command

```
java -version
```

If Java is installed and configured to work from a Command Prompt, running the above command should print the information about the Java version to the console. See the following output for an example of what you should expect:



```
C:\>java -version
java version "1.8.0_231"
Java(TM) SE Runtime Environment (build 1.8.0_231-b11)
Java HotSpot(TM) Client VM (build 25.231-b11, mixed mode, sharing)
```

Else, you need to install it. To do so, complete the following steps:

For Linux

Run the following commands in a terminal:

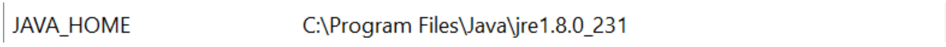
```
sudo apt update; sudo apt install oracle-java8-installer
```

Set Java environment variables :

```
sudo apt install oracle-java8-set-default
```

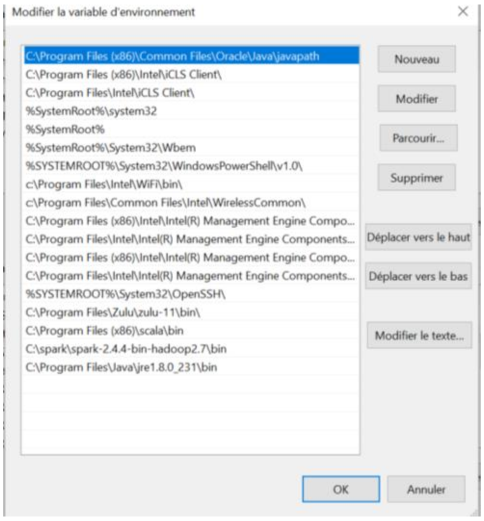
For Windows

1. Go to the Java download page. In case the download link has changed, search for `Java SE Runtime Environment` on the internet and you should be able to find the download page.
2. Click the *Download* button beneath JRE
3. Accept the license agreement and download the latest version of `Java SE Runtime Environment` installer. I suggest getting the exe for Windows x64 (such as `jre-8u92-windows-x64.exe`) unless you are using a 32 bit version of Windows in which case you need to get the *Windows x86 Offline* version.
4. Run the installer.
5. Set the JAVA_HOME environment variable as such: (replace with you Java directory)



```
JAVA_HOME                C:\Program Files\Java\jre1.8.0_231
```

6. Add Java to the PATH system Variable as such:



Modifier la variable d'environnement

```
C:\Program Files (x86)\Common Files\Oracle\Java\javapath
C:\Program Files (x86)\Intel\iCLS Client\
C:\Program Files\Intel\iCLS Client\
%SystemRoot%\system32
%SystemRoot%
%SystemRoot%\System32\Wbem
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\
c:\Program Files\Intel\WiFi\bin\
c:\Program Files\Common Files\Intel\WirelessCommon\
C:\Program Files (x86)\Intel\Intel(R) Management Engine Compo...
C:\Program Files\Intel\Intel(R) Management Engine Components...
C:\Program Files (x86)\Intel\Intel(R) Management Engine Compo...
C:\Program Files\Intel\Intel(R) Management Engine Components...
%SYSTEMROOT%\System32\OpenSSH\
C:\Program Files\Zulu\zulu-11\bin\
C:\Program Files (x86)\scala\bin
C:\spark\spark-2.4.4-bin-hadoop2.7\bin
C:\Program Files\Java\jre1.8.0_231\bin
```

After the installation is complete, close the Command Prompt if it was already open, open it and check if you can successfully run `java -version` command.

Python

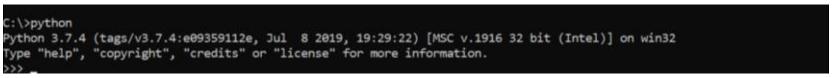
To check if python is already installed, complete the following :

Linux

Open a terminal and type “python”, in most Linux systems python is already installed. So you don’t need to install it.

Windows

Open a command prompt and type the following command “python”, if python is already installed you should have an output as such:

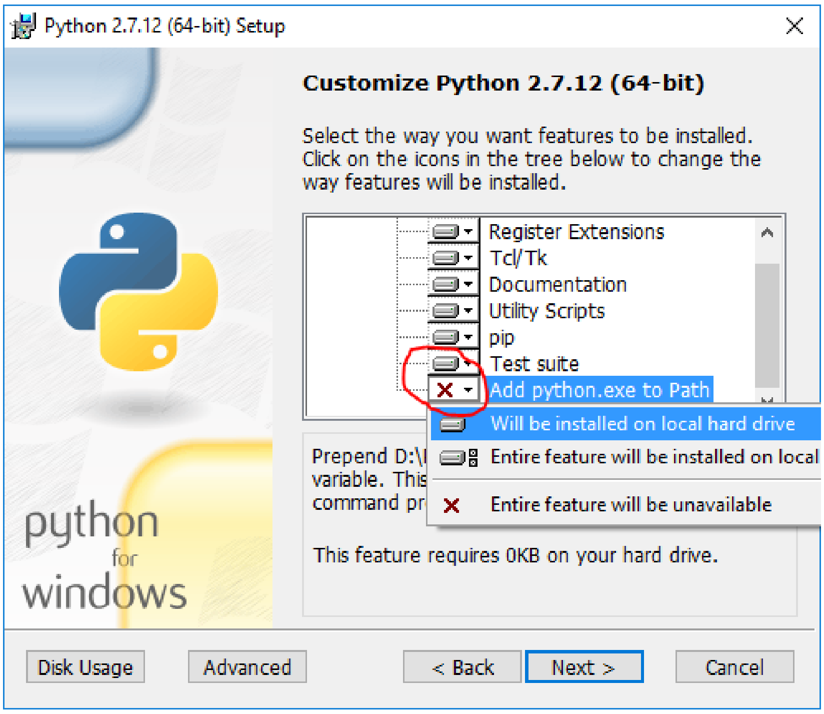


```
C:\>python
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul  8 2019, 19:29:22) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> _
```

You can also test if python is installed by running the command “python - - version”

If python is not installed, you need to install it. To do so :

1. Go to the Python download page.
2. Click the *Latest Python Release* link.
3. Download the `Windows x86-64 MSI installer` file. If you are using a 32 bit version of Windows download the `Windows x86 MSI installer` file.
4. When you run the installer, on the **Customize Python** section, make sure that the option *Add python.exe to Path* is selected. If this option is not selected, some of the PySpark utilities such as `pyspark` and `spark-submit` might not work.



Python 2.7.12 (64-bit) Setup

Customize Python 2.7.12 (64-bit)

Select the way you want features to be installed. Click on the icons in the tree below to change the way features will be installed.

Prepend D:\l
variable. This
command pr

This feature requires 0KB on your hard drive.

After the installation is complete, close the Command Prompt if it was already open, open it and check if you can successfully run `python --version` command.

7-zip (Windows only)

Install 7-zip from the following link : https://www.7-zip.org/

Installing Spark

Common steps for both OS

1. Go to the Spark download page.
2. For *Choose a Spark release*, select the latest stable release of Spark.
3. For *Choose a package type*, select a version that is pre-built for the latest version of Hadoop such as *Pre-built for Hadoop 2.6*.
4. For *Choose a download type*, select *Direct Download.*
5. Click the link next to *Download Spark* to download a zipped tarball file ending in .tgz extension such as `spark-1.6.2-bin-hadoop2.6.tgz`.
6. In order to install Apache Spark, there is no need to run any installer. To use Spark, you extract the files from the downloaded tarball in any folder of your choice. NOTE : Make sure that the folder path and the folder name containing Spark files do not contain any spaces.

Linux

To extract the tarball file, use the following command in a terminal :

```
tar zxvf spark-1.6.2-bin-hadoop2.6.tgz
```

tell your bash (or zsh, etc.) where to find Spark. To do so, configure your $PATH variables by adding the following lines in your `~/.bashrc` (or `~/.zshrc`) file:

```
export SPARK_HOME=[Path to your spark]
export PATH=$SPARK_HOME/bin:$PATH
```

Windows

To extract the tarball file, you can use any tool of your choice. The extraction can be done using the 7Zip tool mentioned in the prerequisites.

Create a folder called `spark` on your C drive and extract the zipped tarball in a folder called `spark-1.6.2-bin-hadoop2.6`. So all Spark files are in a folder called `C:\spark\spark-1.6.2-bin-hadoop2.6`. Set the `SPARK_HOME` environment variable to this folder.

To test if your installation was successful, open a Command Prompt, change to SPARK_HOME directory and type `bin\pyspark`. This should start the PySpark shell which can be used to interactively work with Spark. I got the following messages in the console after running bin\pyspark command.

Installing winutils

Let's download the winutils.exe and configure our Spark installation to find winutils.exe.

1. Create a `hadoop\bin` folder inside the SPARK_HOME folder.

2. Download the winutils.exe for the version of hadoop against which your Spark installation was built for. In this document the hadoop version is 2.6.0. So the winutils.exe for hadoop 2.6.0 should be downloaded and copied to the hadoop\bin folder in the SPARK_HOME folder.
3. Create another system environment variable in Windows called HADOOP_HOME that points to the hadoop folder inside the SPARK_HOME folder.
4. Since the hadoop folder is inside the SPARK_HOME folder, it is better to create HADOOP_HOME environment variable using a value of %SPARK_HOME%\hadoop. That way you don't have to change HADOOP_HOME if SPARK_HOME is updated.

If you now run the bin\pyspark script from a Windows Command Prompt, the error messages related to winutils.exe should be gone.

Installing Jupyter Notebook

Install Jupyter notebook:

```
$ pip install jupyter
```

You can run a regular jupyter notebook by typing:

```
$ jupyter notebook
```

Installing Scala Kernel for Jupyter

Step1: install the package

```
pip install spylon-kernel
```

Step2: create a kernel spec

This will allow us to select the scala kernel in the notebook.

```
python -m spylon_kernel install
```

Step3: start the jupyter notebook

```
ipython notebook
```

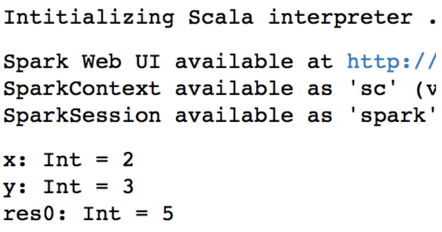
And in the notebook we select New -> spylon-kernel . This will start our scala kernel.

Step4: testing the notebook

Let's write some scala code:

```
val x = 2
val y = 3x+y
```

The output should be something similar with the result in the following. As you can see it also starts the spark components. For this please make sure you have SPARK_HOME set up.



```
Intitializing Scala interpreter .

Spark Web UI available at http://
SparkContext available as 'sc' (v
SparkSession available as 'spark'

x: Int = 2
y: Int = 3
res0: Int = 5
```

Executing Spark in a Jupyter Notebook (pySpark)

FindSpark package

Note : findSpark package is not specific to Jupyter Notebook, you can use this trick in your favorite IDE too.

To install findspark:

```
$ pip install findspark
```

Launch a regular Jupyter Notebook:

```
$ jupyter notebook
```

Create a new Python [default] notebook and write the following script:

```
import findspark
findspark.init()

import pyspark

sc = pyspark.SparkContext(appName="Pi")
[Use your spark context as desired]
sc.stop()
```

Resources

This document was made by combining these well-made resources:

- http://deelesh.github.io/pyspark-windows.html
- executing scala in jupyter notebook : https://medium.com/@bogdan.cojocar/how-to-run-scala-and-spark-in-the-jupyter-notebook-328a80090b3b
- executing spark in python jupyter notebook : https://www.sicara.ai/blog/2017-05-02-get-started-pyspark-jupyter-notebook-3-minutes