Instalación de PySpark en Windows

PySpark es una biblioteca Spark escrita en Python para ejecutar la aplicación Python usando las capacidades de Apache Spark. por lo que no hay una biblioteca de PySpark para descargar.
Sigue los pasos a continuación para instalar PySpark en Windows.

PASO 1. Instalar la distribución Python o Anaconda

El primer paso es descargar e instalar Python desde Python.org o la distribución Anaconda que incluye Python, Spyder IDE y Jupyter notebook.
Link anaconda: https://docs.anaconda.com/anaconda/install/windows/
Link Python: https://www.python.org/downloads/

PASO 1.2 Instalar JDK y definir variables de entorno

También debemos asegurarnos de que tenemos java instalado y debemos asegurarnos de que las variables de entorno están correctamente establecidas.
Link descarga:
https://www.oracle.com/es/java/technologies/javase/javase8-archive-downloads.html

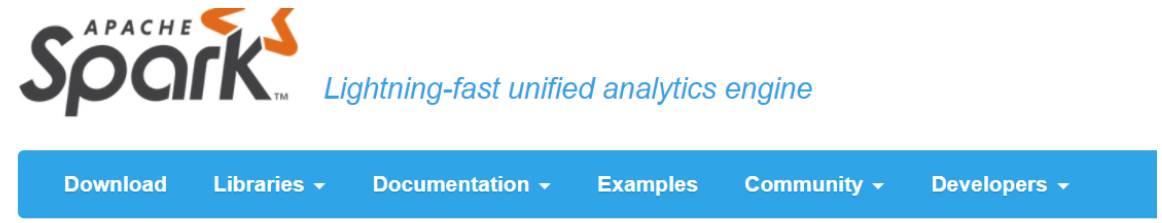
Establecemos las variables de entorno:

```
JAVA_HOME = C:\Program Files\Java\jdk1.8.0_201
```

```
PATH = %PATH%;C:\Program Files\Java\jdk1.8.0_201\bin
```

PASO 2. Instalación de PySpark en Windows

En la página de descarga de Spark(https://spark.apache.org/downloads.html) , seleccione el enlace "Descargar Spark (punto 3)" para descargar. Si desea utilizar una versión diferente de Spark & Hadoop, seleccione la que deseaba de los menús desplegables y el enlace en el punto 3 cambia a la versión seleccionada y le proporciona un enlace actualizado para descargar.



Una vez lo hemos descargado, descomprime el binario usando 7zip o winrar y copia la carpeta de dentro spark-3.0.0-bin-hadoop2.7 en la ruta que quieras por ejemplo **c:\apps**

Ahora crearemos las variables de entorno.

SPARK_HOME = C:\apps\spark-3.0.0-bin-hadoop2.7
HADOOP_HOME = C:\apps\spark-3.0.0-bin-hadoop2.7

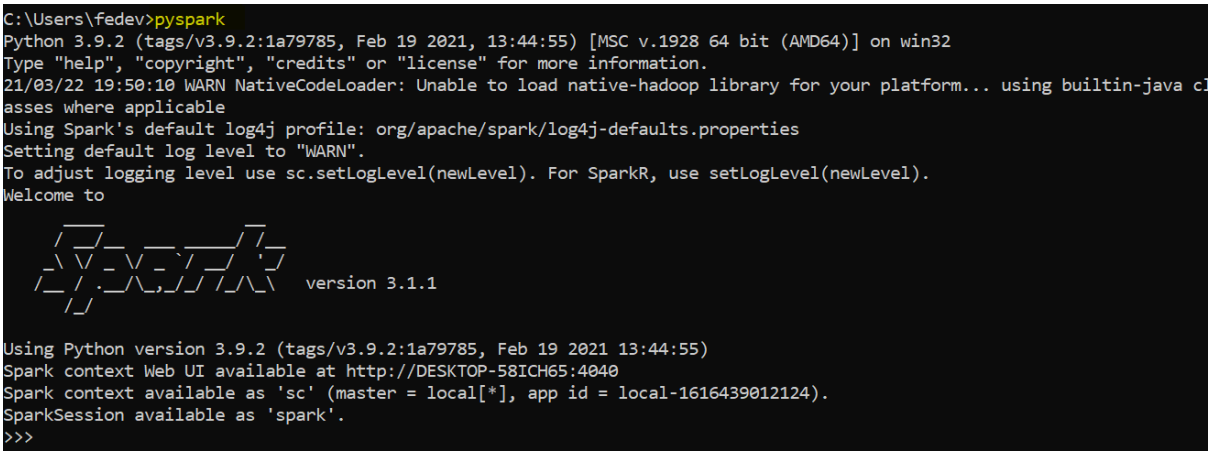
PATH=%PATH%;C:\apps\spark-3.0.0-bin-hadoop2.7\bin

PASO 3. Instale winutils.exe en Windows

Descargue el archivo wunutils.exe de winutils y cópialo en la %SPARK_HOME%\bin\carpeta. Winutils son diferentes para cada versión de Hadoop, por lo tanto, descarga la versión correcta de https://github.com/steveloughran/winutils

PASO 4. Ejecuta PYSPARK

Ya lo tendríamos todo, ahora solo tendremos que abrir una consola y ejecutar: **pyspark**



Y ya tendremos pyspark en windows funcionando

Paso 5. PySpark con Jupyter notebook

Instalamos conda findpark, para acceder a la instancia de Spark desde el cuaderno jupyter.

Pip install jupyter

En una consola de anaconda escribir:

conda install -c conda-forge Findspark

O desde una consola cmd ejecutar:
pip install findspark

Podemos trabajar desde aquí o instalar jupyter
Ejecutar para arrancar
jupyter notebook
http://localhost:8888/

Anexo: Si no tengo PIP instalado

Las siguientes instrucciones deberían funcionar en Windows 7, Windows 8.1 y Windows 10:

1. Descarga el script del instalador get-pip.py. Si estás en Python 3.2, necesitarás esta versión de get-pip.py. En caso de tener Python 3.3 o 3.4 usar estas versiones de PiP correspondientemente Python 3.3 get-pip.py o Python 3.4 get-pip.py. De cualquier manera, haga clic derecho en el enlace y seleccione Guardar como y guárdelo en cualquier carpeta del pc, como su carpeta de Descargas.
2. Abra el símbolo del sistema y navegue hasta el archivo get-pip.py.
3. Ejecute el siguiente comando: `python get-pip.py`

Probar jupyter notebook con spark

Probar por ejemplo esto:

```
import findspark
findspark.init()
findspark.find()
import pyspark

from pyspark import SparkContext, SparkConf
from pyspark.sql import SparkSession
sc = SparkContext.getOrCreate()

nums = sc.parallelize([1,2,3,4])
```

nums

In [3]:
```
import findspark
findspark.init()
findspark.find()
import pyspark


from pyspark import SparkContext, SparkConf
from pyspark.sql import SparkSession
sc = SparkContext.getOrCreate()

nums = sc.parallelize([1,2,3,4])
nums
```

Out[3]:
```
ParallelCollectionRDD[2] at readRDDFromFile at PythonRDD.scala:274
```

In [ ]: