**Examen Final**

Nombre: Jhoel Williams Cruz Quilli

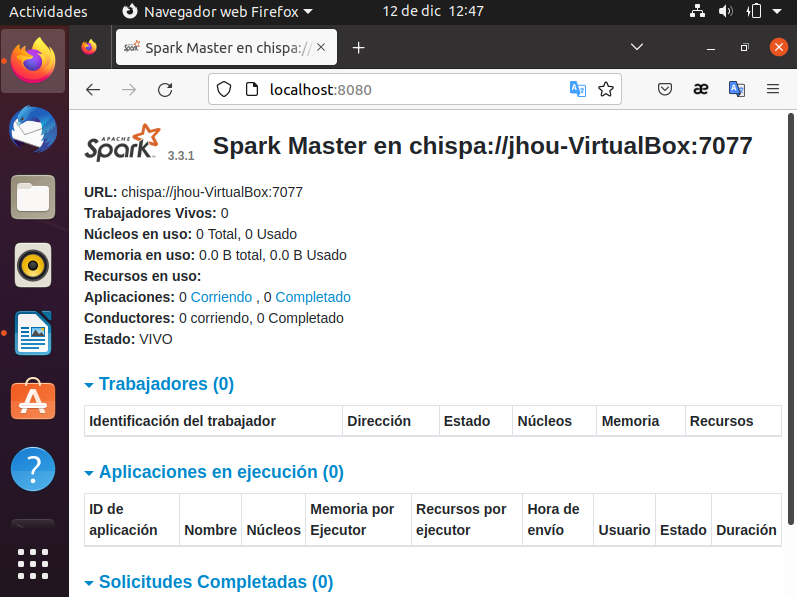
C.I: 13373216

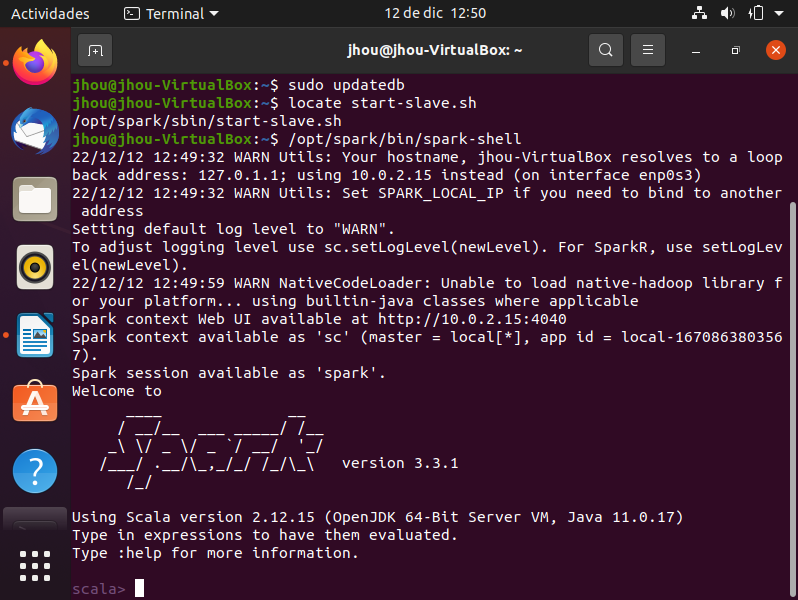
1. En una máquina virtual realice la configuración de apache spark, puede guiarse en cualquier tutorial o el proporcionado por el docente.

url: <https://computingforgeeks.com/how-to-install-apache-spark-on-ubuntu-debian/>

Con el shell podra ejecutar scala por defecto

Instale Python para spark





1. Realice el siguiente código, documente su funcionamiento en apache spark

Sesiones

Para que el codigo funcione debemos ejecutar las siguientes librerías:

import org.apache.spark.sql.SparkSession

Este código en Spark permite la interacción directa con la API de SQL estructurado con Spark Session. También implica especificar la configuración de la aplicación Spark.

Este es el enfoque para iniciar una sesión de Spark y crear un conjunto de datos y un marco de datos con la sesión.

val spark: SparkSession = SparkSession.builder()

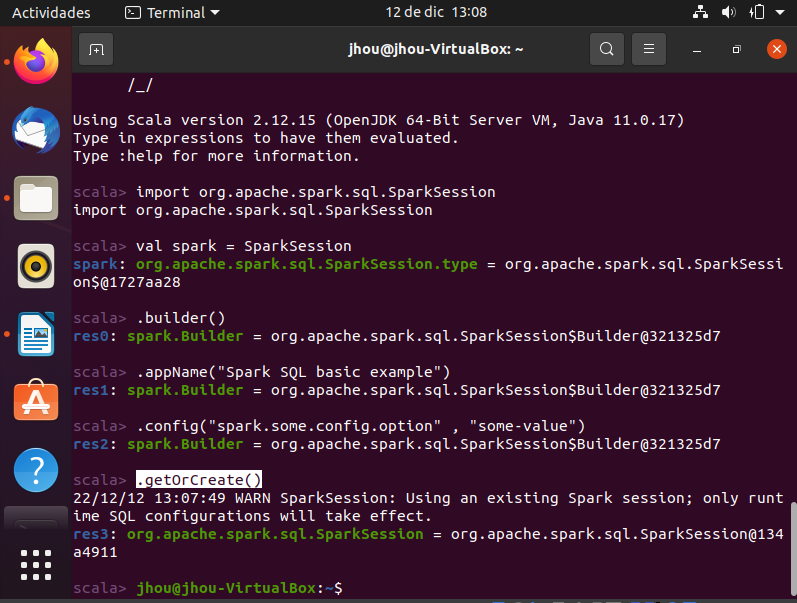
.master("local[\*]")

.appName("simple-app")

.getOrCreate()

val dataSet: Dataset[String] = spark.read.textFile("textfile.csv")

val df: DataFrame = dataSet.toDF()



Streaming

Debemos importar las siguientes librerías:

import org.apache.spark.\_

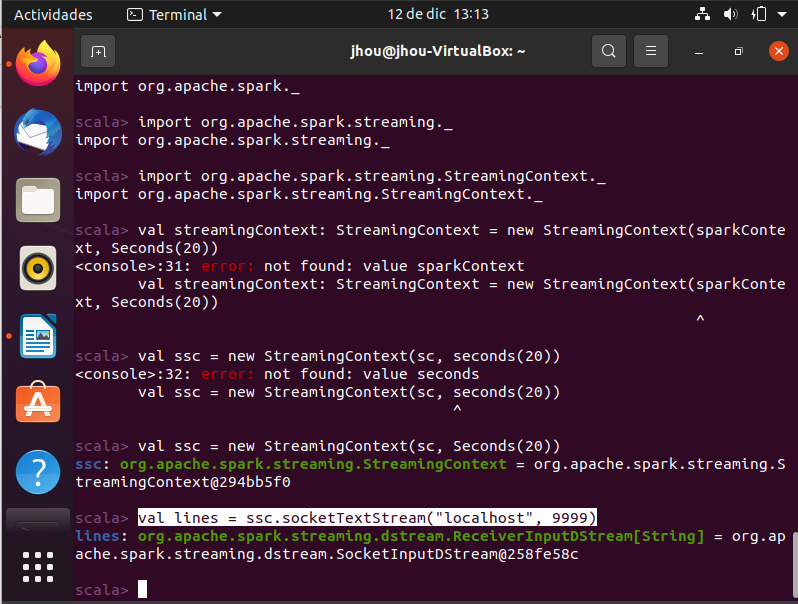
import org.apache.spark.streaming.\_

import org.apache.spark.streaming.StreamingContext.\_

El otro punto de entrada a Spark es usar el contexto de transmisión al interactuar con datos en tiempo real. Se puede crear una instancia de contexto de transmisión a partir de una configuración de Spark o un contexto de Spark. Esto se puede hacer con este codigo

val streamingContext: StreamingContext = new StreamingContext(sparkContext, Seconds(20))

val lines: ReceiverInputDStream[String] = streamingContext.socketTextStream("localhost", 9999)



RDD

val cadenas = Array(“Docentes”, “inteligenciaArtificial”, “quefinal”)

val cadenasRDD = sc . parallelize (cadenas)

cadenasRDD.collect()

file.collect()

val filtro = cadenasRDD.filter(line => line.contains(“quefinal”))

val fileNotFound = sc.textFile(“/7añljdlsjd/alkls/”, 6)

fileNotFound.collect()