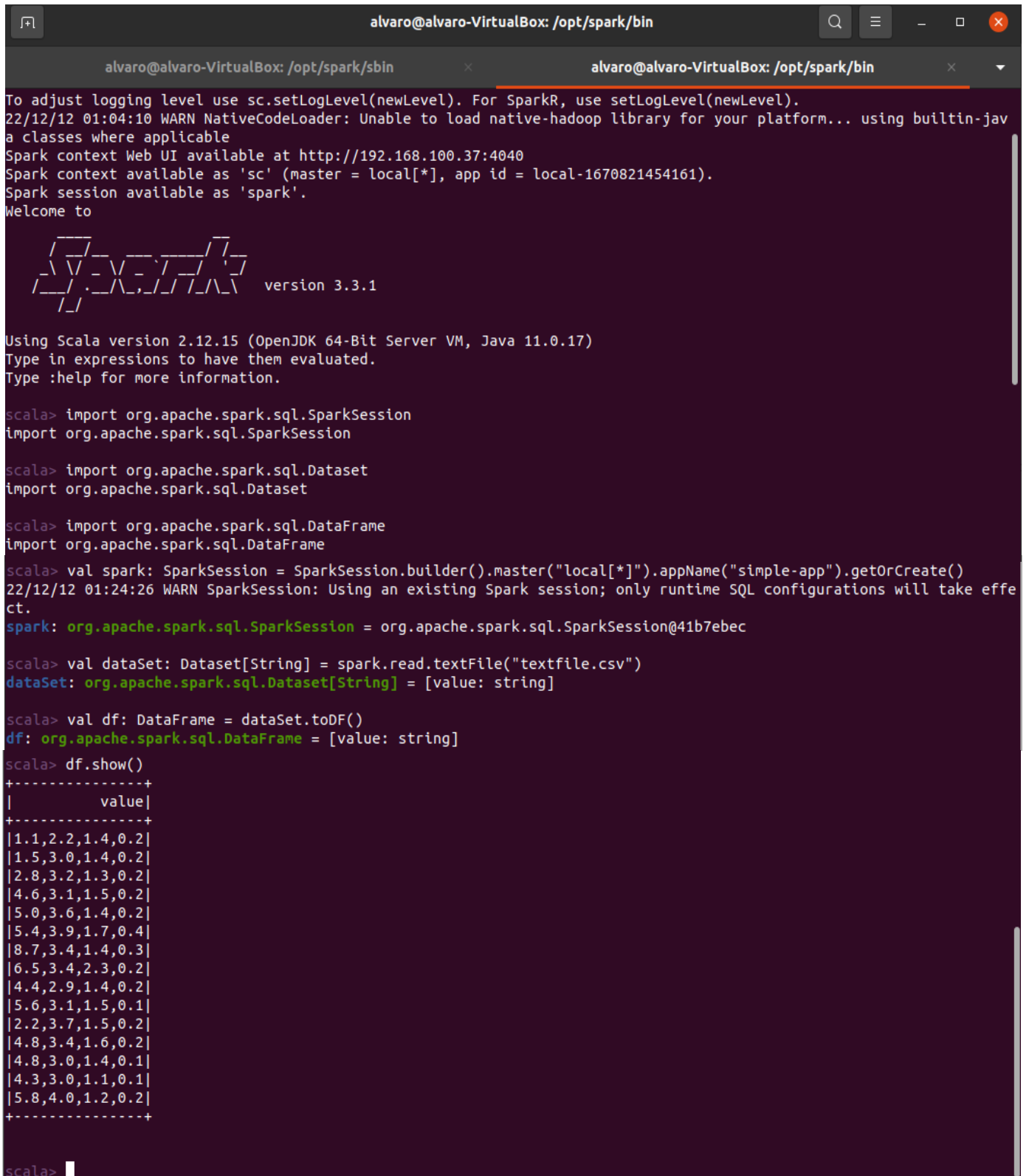


2. Realice el siguiente código, documente su funcionamiento en apache spark

Sesiones

```
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()
```



```
alvaro@alvaro-VirtualBox: /opt/spark/bin
alvaro@alvaro-VirtualBox: /opt/spark/sbin
alvaro@alvaro-VirtualBox: /opt/spark/bin

To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
22/12/12 01:04:10 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-jav
a classes where applicable
Spark context Web UI available at http://192.168.100.37:4040
Spark context available as 'sc' (master = local[*], app id = local-1670821454161).
Spark session available as 'spark'.
Welcome to

  ____      __
 / ___ |____/  \__\
| |  \| |  _ \|__ \| | | | |
| |___| | |_) |_) |
|_____|_|____|____|
version 3.3.1

Using Scala version 2.12.15 (OpenJDK 64-Bit Server VM, Java 11.0.17)
Type in expressions to have them evaluated.
Type :help for more information.

scala> import org.apache.spark.sql.SparkSession
import org.apache.spark.sql.SparkSession

scala> import org.apache.spark.sql.Dataset
import org.apache.spark.sql.Dataset

scala> import org.apache.spark.sql.DataFrame
import org.apache.spark.sql.DataFrame

scala> val spark: SparkSession = SparkSession.builder().master("local[*]").appName("simple-app").getOrCreate()
22/12/12 01:24:26 WARN SparkSession: Using an existing Spark session; only runtime SQL configurations will take effe
ct.
spark: org.apache.spark.sql.SparkSession = org.apache.spark.sql.SparkSession@41b7ebec

scala> val dataSet: Dataset[String] = spark.read.textFile("textfile.csv")
dataSet: org.apache.spark.sql.Dataset[String] = [value: string]

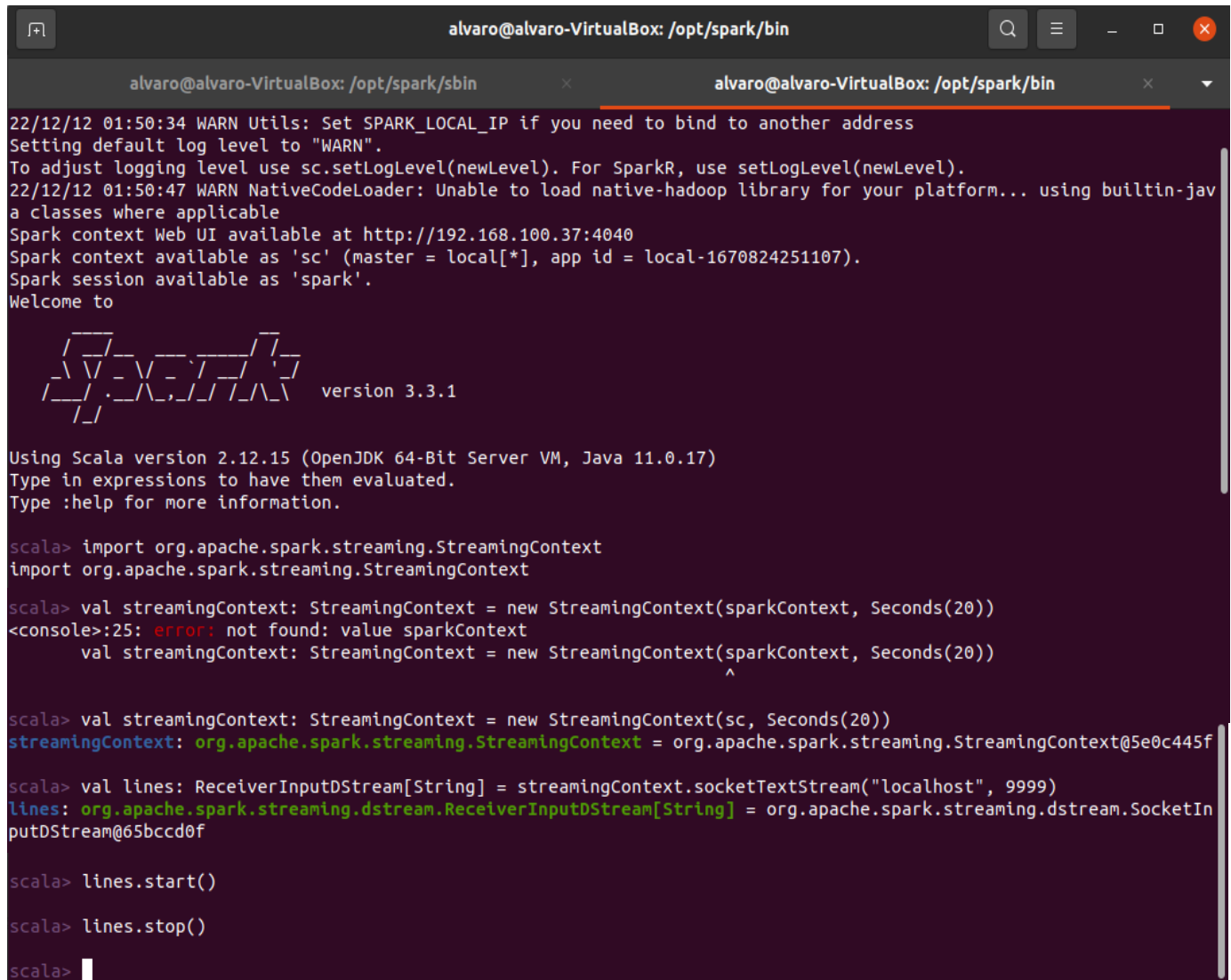
scala> val df: DataFrame = dataSet.toDF()
df: org.apache.spark.sql.DataFrame = [value: string]

scala> df.show()
+-----+
|          value|
+-----+
|1.1,2.2,1.4,0.2|
|1.5,3.0,1.4,0.2|
|2.8,3.2,1.3,0.2|
|4.6,3.1,1.5,0.2|
|5.0,3.6,1.4,0.2|
|5.4,3.9,1.7,0.4|
|8.7,3.4,1.4,0.3|
|6.5,3.4,2.3,0.2|
|4.4,2.9,1.4,0.2|
|5.6,3.1,1.5,0.1|
|2.2,3.7,1.5,0.2|
|4.8,3.4,1.6,0.2|
|4.8,3.0,1.4,0.1|
|4.3,3.0,1.1,0.1|
|5.8,4.0,1.2,0.2|
+-----+

scala>
```

Streaming

```
val streamingContext: StreamingContext = new StreamingContext(sparkContext, Seconds(20))
val lines: ReceiverInputDStream[String] = streamingContext.socketTextStream("localhost", 9999)
```



The screenshot shows a terminal window titled 'alvaro@alvaro-VirtualBox: /opt/spark/bin'. It displays the Spark shell startup logs, including warnings about SPARK_LOCAL_IP and native-hadoop library loading. The Spark version 3.3.1 logo is shown. The user enters Scala code to create a StreamingContext and a ReceiverInputDStream. An error is shown for the first attempt due to a missing 'sparkContext' variable. The second attempt is successful, showing the object creation for 'streamingContext' and 'lines'. Finally, the user calls 'lines.start()' and 'lines.stop()'.

```
alvaro@alvaro-VirtualBox: /opt/spark/bin
22/12/12 01:50:34 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
22/12/12 01:50:47 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-jav
a classes where applicable
Spark context Web UI available at http://192.168.100.37:4040
Spark context available as 'sc' (master = local[*], app id = local-1670824251107).
Spark session available as 'spark'.
Welcome to

  ____
 /    \
/_      \
 \      /
  \    /
   \  /
    \/

 version 3.3.1

Using Scala version 2.12.15 (OpenJDK 64-Bit Server VM, Java 11.0.17)
Type in expressions to have them evaluated.
Type :help for more information.

scala> import org.apache.spark.streaming.StreamingContext
import org.apache.spark.streaming.StreamingContext

scala> val streamingContext: StreamingContext = new StreamingContext(sparkContext, Seconds(20))
<console>:25: error: not found: value sparkContext
      val streamingContext: StreamingContext = new StreamingContext(sparkContext, Seconds(20))
                                                                    ^

scala> val streamingContext: StreamingContext = new StreamingContext(sc, Seconds(20))
streamingContext: org.apache.spark.streaming.StreamingContext = org.apache.spark.streaming.StreamingContext@5e0c445f

scala> val lines: ReceiverInputDStream[String] = streamingContext.socketTextStream("localhost", 9999)
lines: org.apache.spark.streaming.dstream.ReceiverInputDStream[String] = org.apache.spark.streaming.dstream.SocketIn
putDStream@65bccd0f

scala> lines.start()

scala> lines.stop()

scala>
```

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("/7añljdsjd/alkls/", 6)
fileNotFound.collect()
```

[illegible]