4_textfile

February 21, 2022

0.0.1 Create an RDD using textFile

[1]:
```
# Import SparkContext and SparkConf
from pyspark import SparkContext, SparkConf
```

[2]:
```
# Initialize spark
conf = SparkConf().setAppName("RDDFromTextfile")
sc = SparkContext(conf=conf)
```

```
22/02/21 11:55:51 WARN Utils: Your hostname, ThinkCentre resolves to a loopback
address: 127.0.1.1; using 10.180.5.223 instead (on interface eno1)
22/02/21 11:55:51 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another
address
22/02/21 11:55:51 WARN NativeCodeLoader: Unable to load native-hadoop library
for your platform… using builtin-java classes where applicable
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use
setLogLevel(newLevel).
```

[ ]:
```
# Create an RDD of market arrivals of various items (Karnataka)
marketRDD = sc.textFile("./CommMktArrivals.csv")
```

[ ]:
```
# Signature: sc.textFile(name, minPartitions=None, use_unicode=True)

# Read a text file from HDFS, a local file system (available on all
# nodes), or any Hadoop-supported file system URI, and return it as an
# RDD of Strings.

# If use_unicode is False, the strings will be kept as `str` (encoding
# as `utf-8`), which is faster and smaller than unicode. (Added in
# Spark 1.2)

sc.textFile
```

[ ]:
```
marketRDD.collect()
```