

pipe() Transformation

Pipe each partition of the RDD through a shell command, e.g. a Perl or bash script. RDD elements are written to the process's stdin and lines output to its stdout are returned as an RDD of strings.

In [1]: **from** pyspark.sql **import** SparkSession

```
spark = SparkSession \
    .builder \
    .master("local[*]") \
    .appName("pipe Transformation") \
    .enableHiveSupport() \
    .getOrCreate()
```

Setting default log level to "WARN".

To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).

22/10/14 09:50:32 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable

In [2]: *#Reading data from a file on the local machine*

```
data_file_path = "/Users/vaishaliyasala/Desktop/Github/Spark/Exercise_Dependencies/SalesJan2009.csv"
```

```
df = spark.read.csv(data_file_path, header = True )
```

```
df1 = df.select(df["Name"]).repartition(4)
```

```
print(df1.printSchema())
```

#From DataFrame to RDD and view of sample data

```
names_rdd = df1.rdd.map(lambda x: (x[0]))
```

```
print("Names Input=", names_rdd.take(10))
```

```
print("Number of Partitions =", names_rdd.getNumPartitions())
```

```
root
```

```
|-- Name: string (nullable = true)
```

```
None
```

```
Names Input= ['Heloise ', 'Melissa', 'Maria Macarena', 'Harriet', 'William', 'Mary', 'Mike', 'Tracey', 'Leila', 'Ellen']
```

```
Number of Partitions = 4
```

In [3]: *#Shell script Path*

```
shell_script_path = "/Users/vaishaliyasala/Desktop/Github/Spark/Exercise_Dependencies/pipe.sh"
```

#In this example, shell script converts the input from each partition of RDD to upper case and returns it

```
pipe_names_rdd = names_rdd.pipe(shell_script_path)
```

```
print("Names in UPPER CASE =", pipe_names_rdd.take(5))
```

```
print("Number of Partitions =", pipe_names_rdd.getNumPartitions())
```

```
Names in UPPER CASE = ['HELOISE', 'MELISSA', 'MARIA MACARENA', 'HARRIET', 'WILLIAM']
```

```
Number of Partitions = 4
```