

# Program JAR Files

All the required jar files are available in the Lab @ the following path -

**/data/trendytech/mapreduce\_jars**

**Note :** These JAR Files are placed in your local system i.e., the gateway node

You can view the jar files by using the following command

**ls /data/trendytech/mapreduce\_jars**

---

```
[itv005857@g01 ~]$ ls /data/trendytech/mapreduce_jars/  
mapreduce_prog_0_reducer.jar  mapreduce_prog_2_reducer.jar  mapreduce_prog_combiner.jar  mapreduce_prog_cpartitioner.jar  mapreduce_prog.jar  
[itv005857@g01 ~]$
```

## Here is how you can run the programs-

**Pre steps before running the Jars**

**-Create input folder named input under your home directory (/user/<username>/data/) in HDFS**

**Ex :** `hadoop fs -mkdir /user/itv005857/data/input`

**-Create a file named inputfile.txt on your local gateway node and insert some text. Save and Quit**

**Ex :** `vi inputfile.txt`

**-Copy the file inputfile.txt to the input directory (created in the first step) in HDFS**

**Ex:** `hadoop fs -put inputfile.txt /user/itv005857/data/input`

(The username itv005857 has to be replaced with your respective usernames)

Now, you can run the following command -

**hadoop jar /data/trendytech/mapreduce\_jars/<jarname> <input\_file\_path>  
<output\_folder>**

(Note - Replace <jarname> <input\_file\_path> and <output\_folder> with their respective values for each of the example programs to be executed)

Ex: `hadoop jar /data/trendytech/mapreduce_jars/mapreduce_prog.jar /user/itv005857/data/input/inputfile.txt /user/itv005857/data/output`

```
itv005857@g01:~$ hadoop jar /data/trendytech/mapreduce_jars/mapreduce_prog.jar /user/itv005857/data/input/inputfile.txt /user/itv005857/data/output
2023-04-23 07:55:17,663 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at m02.itviversity.com/172.16.1.104:8032
2023-04-23 07:55:17,854 INFO client.AHSPProxy: Connecting to Application History server at m01.itviversity.com/172.16.1.103:10200
2023-04-23 07:55:18,230 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/itv005857/.staging/job_1675999795986_292
2023-04-23 07:55:23,886 INFO input.FileInputFormat: Total input files to process : 1
2023-04-23 07:55:24,136 INFO mapreduce.JobSubmitter: number of splits:1
2023-04-23 07:55:24,690 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1675999795986_29268
2023-04-23 07:55:24,690 INFO mapreduce.JobSubmitter: Executing with tokens: []
2023-04-23 07:55:24,905 INFO conf.Configuration: resource-types.xml not found
2023-04-23 07:55:24,905 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2023-04-23 07:55:24,950 INFO impl.YarnClientImpl: Submitted application application_1675999795986_29268
2023-04-23 07:55:24,969 INFO mapreduce.Job: The url to track the job: http://m02.itviversity.com:19088/proxy/application_1675999795986_29268/
2023-04-23 07:55:24,970 INFO mapreduce.Job: Running job: job_1675999795986_29268
2023-04-23 07:55:30,025 INFO mapreduce.Job: Job job_1675999795986_29268 running in uber mode : false
2023-04-23 07:55:30,026 INFO mapreduce.Job: map 0% reduce 0%
2023-04-23 07:55:34,099 INFO mapreduce.Job: map 100% reduce 0%
2023-04-23 07:55:39,131 INFO mapreduce.Job: map 100% reduce 100%
2023-04-23 07:55:40,141 INFO mapreduce.Job: Job job_1675999795986_29268 completed successfully
2023-04-23 07:55:40,203 INFO mapreduce.Job: Counters: 54
    File System Counters
      FILE: Number of bytes read=327
      FILE: Number of bytes written=532967
      FILE: Number of read operations=0
      FILE: Number of large read operations=0
      FILE: Number of write operations=0
      HDFS: Number of bytes read=281
      HDFS: Number of bytes written=152
```

Here is boiler plate code to create Spark Session-

```
from pyspark.sql import SparkSession
import getpass
username = getpass.getuser()
spark = SparkSession. \
    builder. \
    config('spark.ui.port','0'). \
    config("spark.sql.warehouse.dir", f"/user/{username}/warehouse"). \
    enableHiveSupport(). \
    master('yarn'). \
    getOrCreate()
```