

Experiment 6: Hadoop Reduce Side Join

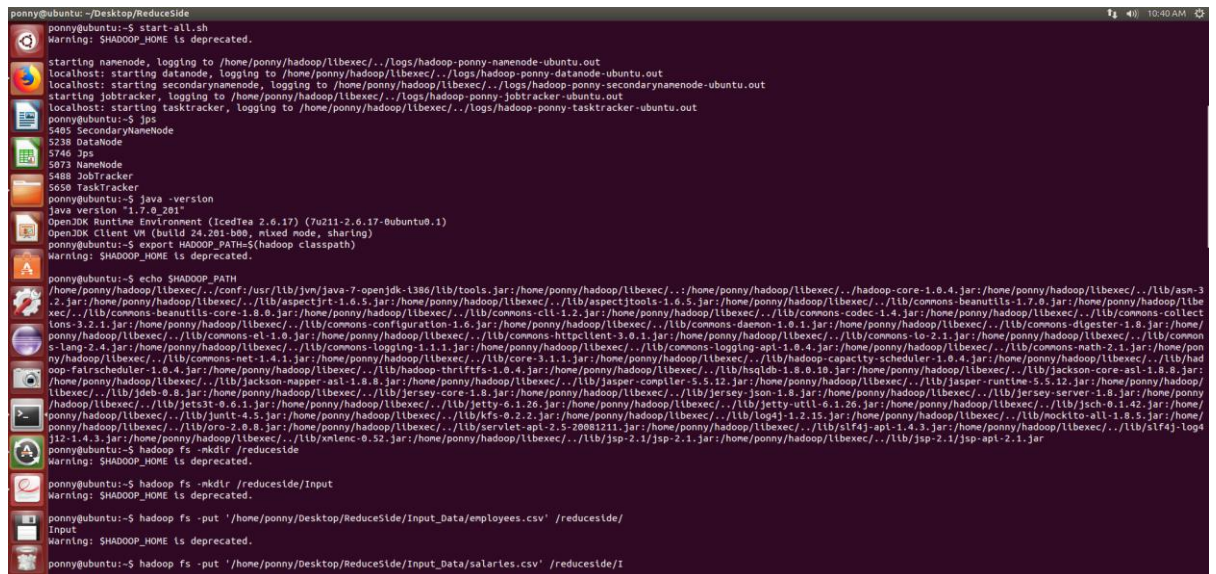
Namansh Singh Maurya
22MIA1034

Aim:

Implement the **Reduce Side Join** using Hadoop MapReduce.

Algorithm/Procedure:

1. Firstly, we will start our Hadoop system and check if it is working correctly or not, with that, we will export the environment variable of Hadoop and echo it back to the terminal.



```
ponny@ubuntu: ~/Desktop/ReduceSide
ponny@ubuntu:~$ start-all.sh
Warning: $HADOOP_HOME is deprecated.

starting namenode, logging to /home/ponny/hadoop/libexec/../logs/hadoop-ponny-namenode-ubuntu.out
localhost: starting datanode, logging to /home/ponny/hadoop/libexec/../logs/hadoop-ponny-datanode-ubuntu.out
localhost: starting secondarynamenode, logging to /home/ponny/hadoop/libexec/../logs/hadoop-ponny-secondarynamenode-ubuntu.out
starting jobtracker, logging to /home/ponny/hadoop/libexec/../logs/hadoop-ponny-jobtracker-ubuntu.out
localhost: starting tasktracker, logging to /home/ponny/hadoop/libexec/../logs/hadoop-ponny-tasktracker-ubuntu.out

ponny@ubuntu:~$ jps
5485 SecondaryNameNode
5238 DataNode
5746 Jps
5073 NameNode
5488 JobTracker
5650 TaskTracker

ponny@ubuntu:~$ java -version
java version "1.7.0_201"
OpenJDK Runtime Environment (IcedTea 2.6.17) (7u211-2.6.17-0ubuntu0.1)
OpenJDK Client VM (build 24.201-b00, mixed mode, sharing)

ponny@ubuntu:~$ export HADOOP_PATH=$(hadoop classpath)
Warning: $HADOOP_HOME is deprecated.

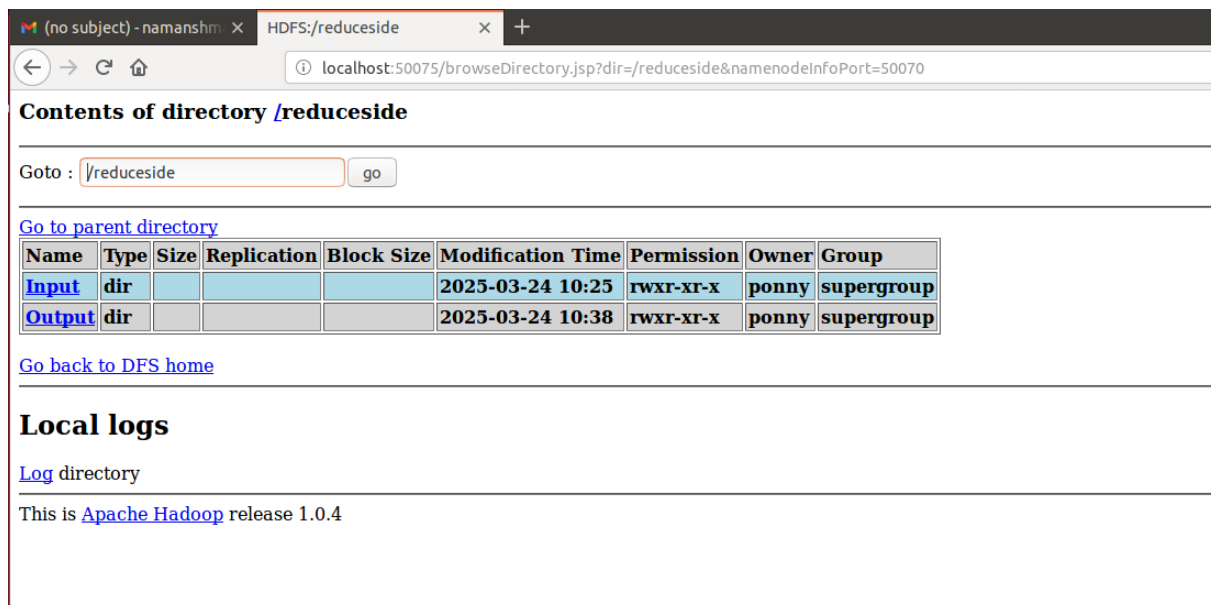
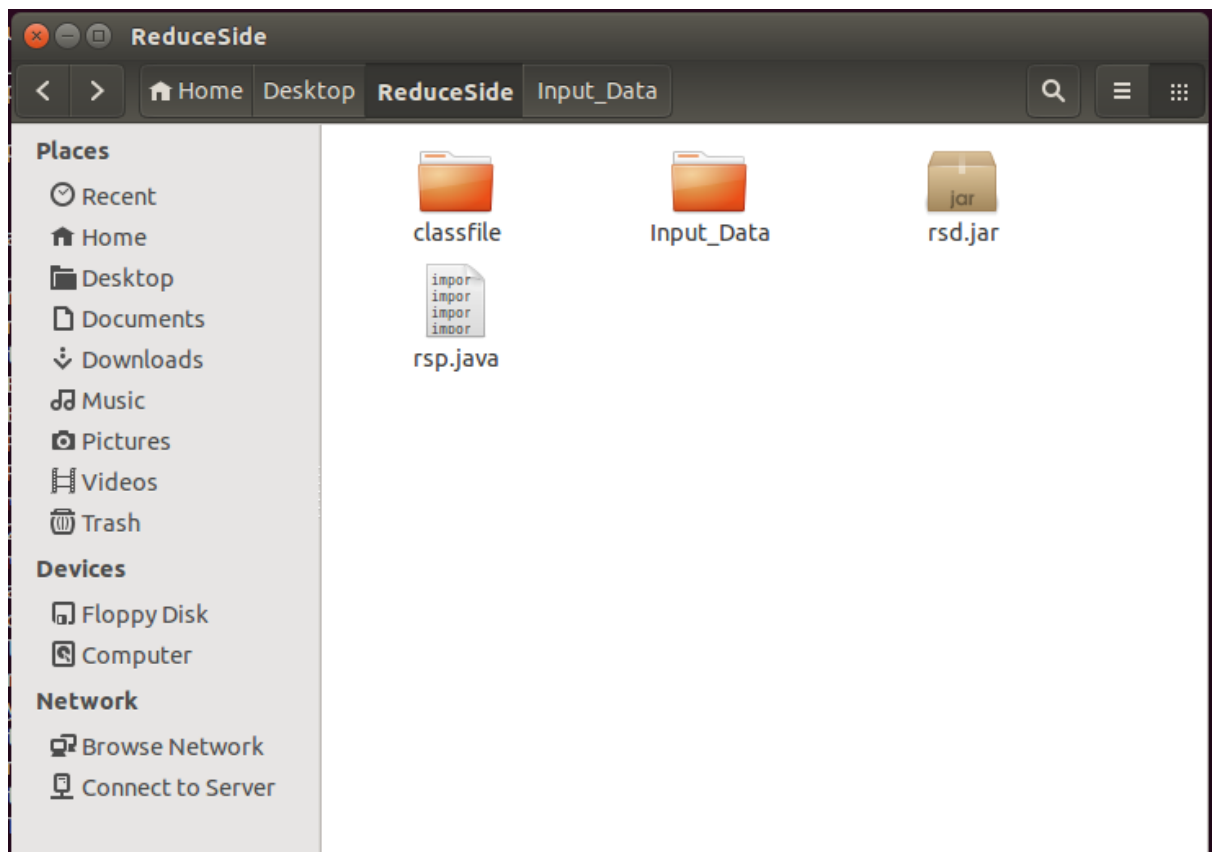
ponny@ubuntu:~$ echo $HADOOP_PATH
/home/ponny/hadoop/libexec/../conf:/usr/lib/jvm/java-7-openjdk-1.386/lib/tools.jar:/home/ponny/hadoop/libexec/../:/home/ponny/hadoop/libexec/../:/hadoop-core-1.0.4.jar:/home/ponny/hadoop/libexec/../lib/asn-3-2.jar:/home/ponny/hadoop/libexec/../lib/aspectjrt-1.6.5.jar:/home/ponny/hadoop/libexec/../lib/aspectjtools-1.6.5.jar:/home/ponny/hadoop/libexec/../lib/commons-beanutils-1.7.0.jar:/home/ponny/hadoop/libexec/../lib/commons-beanutils-core-1.8.0.jar:/home/ponny/hadoop/libexec/../lib/commons-cli-1.2.jar:/home/ponny/hadoop/libexec/../lib/commons-codec-1.4.jar:/home/ponny/hadoop/libexec/../lib/commons-collections-3.2.1.jar:/home/ponny/hadoop/libexec/../lib/commons-configuration-1.6.jar:/home/ponny/hadoop/libexec/../lib/commons-daemon-1.0.1.jar:/home/ponny/hadoop/libexec/../lib/commons-digester-1.8.jar:/home/ponny/hadoop/libexec/../lib/commons-el-1.0.jar:/home/ponny/hadoop/libexec/../lib/commons-httpclient-3.0.1.jar:/home/ponny/hadoop/libexec/../lib/commons-io-2.1.jar:/home/ponny/hadoop/libexec/../lib/commons-lang-2.4.jar:/home/ponny/hadoop/libexec/../lib/commons-logging-1.1.1.jar:/home/ponny/hadoop/libexec/../lib/commons-logging-api-1.0.4.jar:/home/ponny/hadoop/libexec/../lib/commons-math-2.1.jar:/home/ponny/hadoop/libexec/../lib/commons-net-1.4.1.jar:/home/ponny/hadoop/libexec/../lib/core-3.1.1.jar:/home/ponny/hadoop/libexec/../lib/hadoop-capacity-scheduler-1.0.4.jar:/home/ponny/hadoop/libexec/../lib/hadoop-fatrscheduler-1.0.4.jar:/home/ponny/hadoop/libexec/../lib/hadoop-thriftfs-1.0.4.jar:/home/ponny/hadoop/libexec/../lib/hsqldb-1.8.0.19.jar:/home/ponny/hadoop/libexec/../lib/jackson-core-asl-1.8.0.jar:/home/ponny/hadoop/libexec/../lib/jackson-mapper-asl-1.8.0.jar:/home/ponny/hadoop/libexec/../lib/jasper-compiler-5.5.12.jar:/home/ponny/hadoop/libexec/../lib/jasper-runtime-5.5.12.jar:/home/ponny/hadoop/libexec/../lib/jersey-json-1.8.jar:/home/ponny/hadoop/libexec/../lib/jersey-server-1.8.jar:/home/ponny/hadoop/libexec/../lib/jets3t-0.6.1.jar:/home/ponny/hadoop/libexec/../lib/jetty-6.1.26.jar:/home/ponny/hadoop/libexec/../lib/jetty-util-6.1.26.jar:/home/ponny/hadoop/libexec/../lib/jsch-0.1.42.jar:/home/ponny/hadoop/libexec/../lib/junit-4.5.jar:/home/ponny/hadoop/libexec/../lib/kfs-0.2.2.jar:/home/ponny/hadoop/libexec/../lib/log4j-1.2.15.jar:/home/ponny/hadoop/libexec/../lib/mockito-all-1.8.5.jar:/home/ponny/hadoop/libexec/../lib/oro-2.0.8.jar:/home/ponny/hadoop/libexec/../lib/servlet-api-2.5-20081211.jar:/home/ponny/hadoop/libexec/../lib/slf4j-api-1.4.3.jar:/home/ponny/hadoop/libexec/../lib/slf4j-log4j12-1.4.3.jar:/home/ponny/hadoop/libexec/../lib/xmlenc-0.52.jar:/home/ponny/hadoop/libexec/../lib/jsp-2.1/jsp-2.1.jar:/home/ponny/hadoop/libexec/../lib/jsp-2.1/jsp-api-2.1.jar

ponny@ubuntu:~$ hadoop fs -mkdir /reduceside
Warning: $HADOOP_HOME is deprecated.

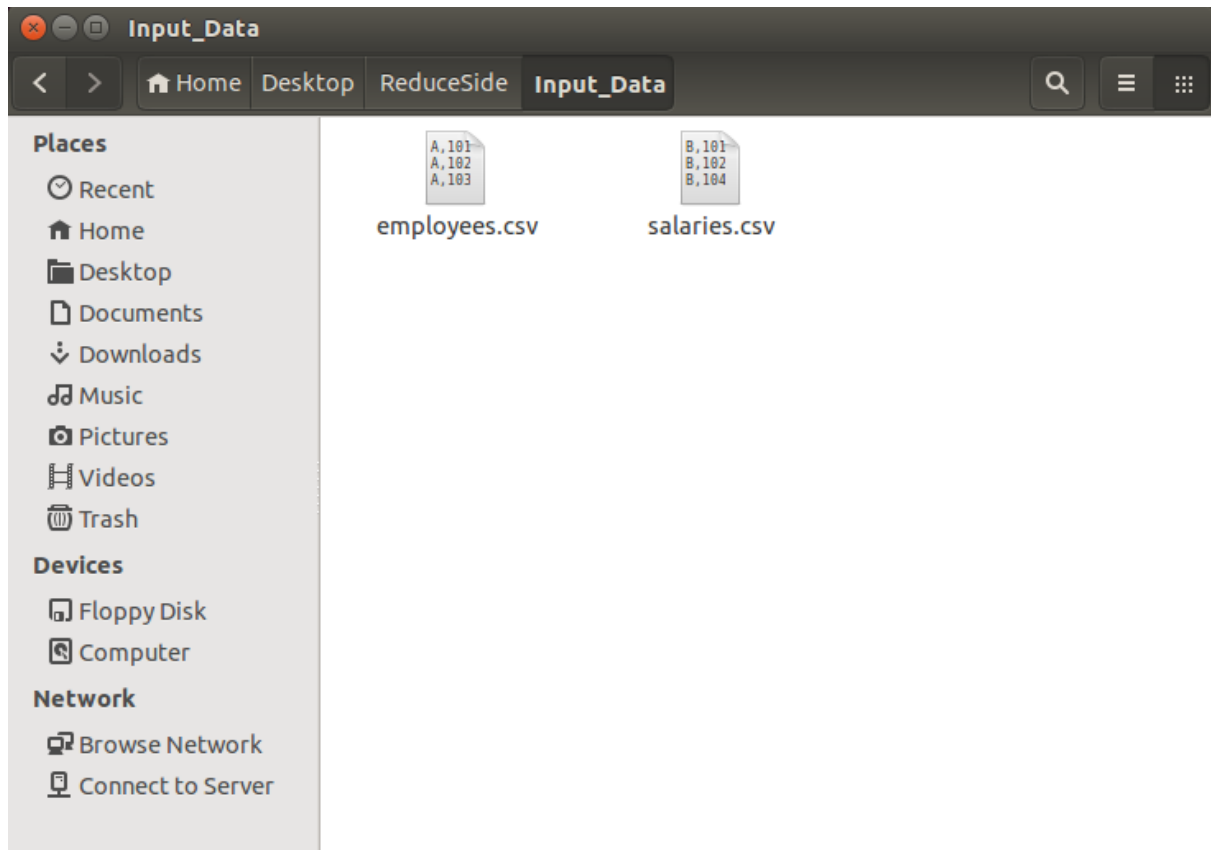
ponny@ubuntu:~$ hadoop fs -put '/home/ponny/Desktop/ReduceSide/Input_Data/employees.csv' /reduceside/
Input
Warning: $HADOOP_HOME is deprecated.

ponny@ubuntu:~$ hadoop fs -put '/home/ponny/Desktop/ReduceSide/Input_Data/salaries.csv' /reduceside/I
```

2. Then we will create one folder named **reducejoin**, inside that, we will make **input_data** and **outputs** named folders and **rsp.java** file.



3. Inside input_data, we will create two text files named employees.csv and salaries.txt.



4. **After this, we will make the reducesidejoin directory in the Hadoop system.**
 We can view that using localhost, and we will put both the text files inside **reducesidejoin/Input** folder.
 Then we will enter the **ReduceSideJoin** directory for making required files inside **classfile**.

```

ponny@ubuntu: ~/Desktop/ReduceSide
ponny@ubuntu:~/Desktop$ cd Desktop
ponny@ubuntu:~/Desktop$ cd ReduceSide
ponny@ubuntu:~/Desktop/ReduceSide$ javac -classpath $(HADOOP_PATH) -d '/home/ponny/Desktop/ReduceSide/classfile' /home/ponny/Desktop/ReduceSide/rsp.java
ponny@ubuntu:~/Desktop/ReduceSide$ jar -cvf rsd.jar -C '/home/ponny/Desktop/ReduceSide/classfile' /
added manifest
adding: rsp.class(in = 1477) (out= 835)(deflated 43%)
adding: rsp$JoinReducer.class(in = 2006) (out= 891)(deflated 55%)
adding: rsp$JoinMapper.class(in = 1625) (out= 686)(deflated 57%)
ponny@ubuntu:~/Desktop/ReduceSide$ jar -cvf rsd.jar -C '/home/ponny/Desktop/ReduceSide/classfile' /
added manifest
adding: rsp.class(in = 1477) (out= 835)(deflated 43%)
adding: rsp$JoinReducer.class(in = 2006) (out= 891)(deflated 55%)
adding: rsp$JoinMapper.class(in = 1625) (out= 686)(deflated 57%)
ponny@ubuntu:~/Desktop/ReduceSide$ hadoop jar '/home/ponny/Desktop/ReduceSide/rsd.jar' /reduceside/Input
put /reduceside/Output
Warning: $HADOOP_HOME is deprecated.
Exception in thread "main" java.lang.ClassNotFoundException: /reduceside/Input
    at java.lang.Class.forName0(Native Method)
    at java.lang.Class.forName(Class.java:278)
    at org.apache.hadoop.util.RunJar.main(RunJar.java:149)
ponny@ubuntu:~/Desktop/ReduceSide$ hadoop jar '/home/ponny/Desktop/ReduceSide/rsd.jar' /reduceside/Input
put /reduceside/Output
Warning: $HADOOP_HOME is deprecated.
Exception in thread "main" java.lang.ClassNotFoundException: /reduceside/Input
    at java.lang.Class.forName0(Native Method)
    at java.lang.Class.forName(Class.java:278)
    at org.apache.hadoop.util.RunJar.main(RunJar.java:149)
ponny@ubuntu:~/Desktop/ReduceSide$ jar -cvf rsd.jar -C '/home/ponny/Desktop/ReduceSide/classfile' /
added manifest
adding: rsp.class(in = 1477) (out= 835)(deflated 43%)
adding: rsp$JoinReducer.class(in = 2006) (out= 891)(deflated 55%)
adding: rsp$JoinMapper.class(in = 1625) (out= 686)(deflated 57%)
adding: rsp.java(in = 2977) (out= 942)(deflated 68%)
adding: classfile/(in = 0) (out= 0)(stored 0%)
adding: classfile/rsp.class(in = 1477) (out= 835)(deflated 43%)
adding: classfile/rsp$JoinReducer.class(in = 2006) (out= 891)(deflated 55%)
adding: classfile/rsp$JoinMapper.class(in = 1625) (out= 686)(deflated 57%)
adding: Input_Data/(in = 0) (out= 0)(stored 0%)
adding: Input_Data/employees.csv(in = 36) (out= 32)(deflated 11%)
adding: Input_Data/salaries.csv(in = 36) (out= 26)(deflated 27%)
ponny@ubuntu:~/Desktop/ReduceSide$ hadoop jar '/home/ponny/Desktop/ReduceSide/rsd.jar' /reduceside/Input
put /reduceside/Output
Warning: $HADOOP_HOME is deprecated.

```

(no subject) - namanshm ×

HDFS:/reduceside/Input ×

+

localhost:50075/browseDirectory.jsp?dir=%2Freduceside%2FInput&namenodeInfoPort=50070

Contents of directory **/reduceside/Input**

Goto :

[Go to parent directory](#)

Name	Type	Size	Replication	Block Size	Modification Time	Permission	Owner	Group
employees.csv	file	0.04 KB	1	64 MB	2025-03-24 10:25	rw-r--r--	ponny	supergroup
salaries.csv	file	0.04 KB	1	64 MB	2025-03-24 10:25	rw-r--r--	ponny	supergroup

[Go back to DFS home](#)

Local logs

[Log](#) directory

This is [Apache Hadoop](#) release 1.0.4

(no subject) - namanshm ×

HDFS:/reduceside/Input/employees.csv ×

+

localhost:50075/browseBlock.jsp?blockId=-147379637715698525&blockSize=36&genstamp=1690&filename=%2Freduceside%2FInput%2Femployees.csv

File: **/reduceside/Input/employees.csv**

Goto :

[Go back to dir listing](#)
[Advanced view/download options](#)

A,101,John
A,102,Alice
A,103,Bob

[Download this file](#)
[Tail this file](#)

Chunk size to view (in bytes, up to file's DFS block size):

Total number of blocks: 1
-147379637715698525: [127.0.0.1:50010](#)

[Go back to DFS home](#)

(no subject) - namanshm x HDFS:/reduceside/Input/sal x +

localhost:50075/browseBlock.jsp?blockId=-1224756932014350680&blockSize=36&genstamp=16

File: /reduceside/Input/salaries.csv

Goto :

[Go back to dir listing](#)
[Advanced view/download options](#)

B,101,5000
B,102,6000
B,104,7000

[Download this file](#)
[Tail this file](#)

Chunk size to view (in bytes, up to file's DFS block size):

Total number of blocks: 1
-1224756932014350680: [127.0.0.1:50010](#)

[Go back to DFS home](#)

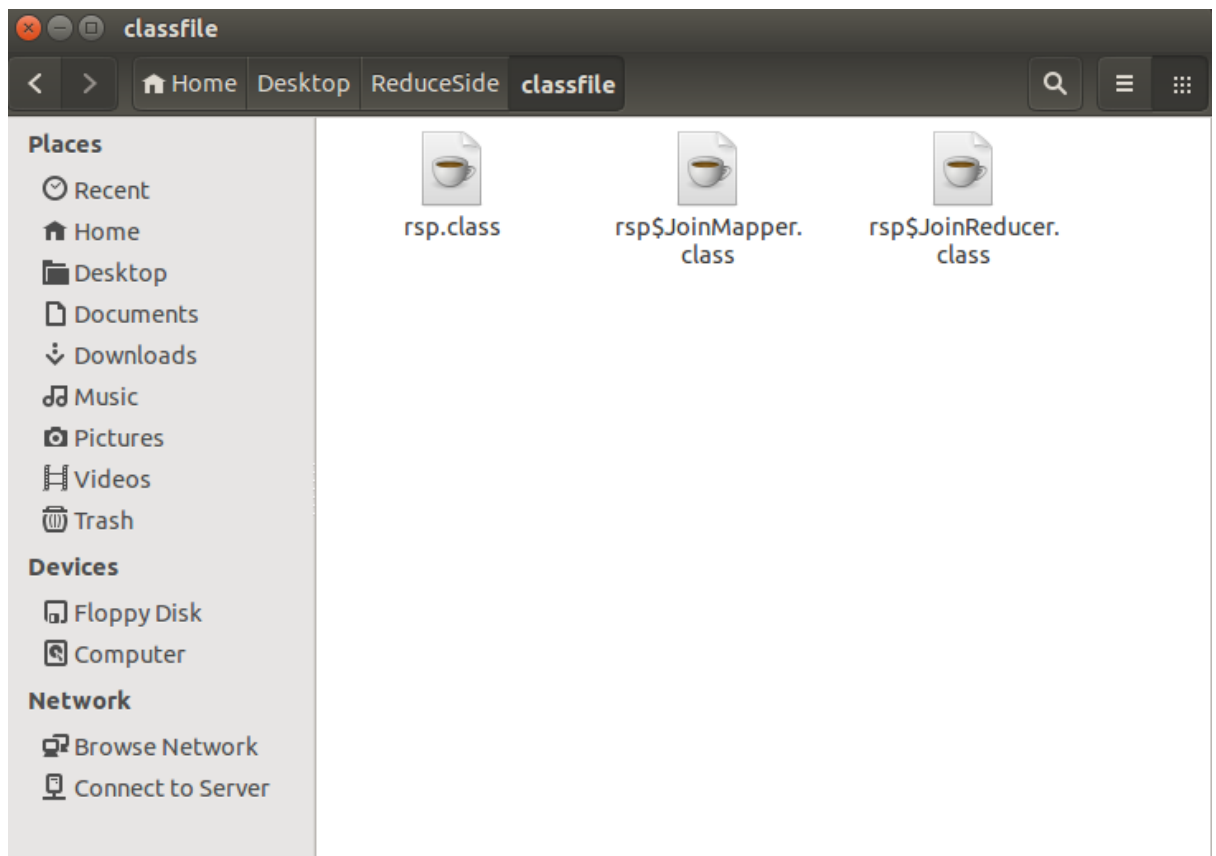
5. Now using this command:

```
javac -classpath ${HADOOP_CLASSPATH} -d /home/user/ReduceSideJoin/classfile  
/home/user/ReduceSideJoin/ReduceSideJoin.java
```

we will get the following three files in the **classfile** directory.

```
ponny@ubuntu: ~/Desktop/ReduceSide
at java.lang.Class.forName(Native Method)
at java.lang.Class.forName(Class.java:278)
at org.apache.hadoop.util.RunJar.main(RunJar.java:149)
ponny@ubuntu:~/Desktop/ReduceSide$ hadoop jar '/home/ponny/Desktop/ReduceSide/rsd.jar' rsp /reducedside/
Input /reducedside/output
warning: SHADOOP_HOME is deprecated.

25/03/24 10:37:43 WARN mapred.JobClient: Use GenericOptionsParser for parsing the arguments. Applications should implement Tool for the same.
25/03/24 10:37:43 INFO Input.FileInputFormat: Total input paths to process : 2
25/03/24 10:37:43 INFO util.NativeCodeLoader: Loaded the native-hadoop library
25/03/24 10:37:43 WARN mapred.JobClient: SnappyLoadSnappy: Snappy native library not loaded
25/03/24 10:37:44 INFO mapred.JobClient: Running Job: job_202503241020_0001
25/03/24 10:37:45 INFO mapred.JobClient: map 0% reduce 0%
25/03/24 10:37:58 INFO mapred.JobClient: map 100% reduce 0%
25/03/24 10:38:13 INFO mapred.JobClient: map 100% reduce 100%
25/03/24 10:38:18 INFO mapred.JobClient: Job complete: job_202503241020_0001
25/03/24 10:38:18 INFO mapred.JobClient: Counters: 29
25/03/24 10:38:18 INFO mapred.JobClient: Job Counters
25/03/24 10:38:18 INFO mapred.JobClient: Launched reduce tasks=1
25/03/24 10:38:18 INFO mapred.JobClient: SLOTS_MILLIS_MAPS=14885
25/03/24 10:38:18 INFO mapred.JobClient: Total time spent by all reduces waiting after reserving
25/03/24 10:38:18 INFO mapred.JobClient: slots (ms)=0
25/03/24 10:38:18 INFO mapred.JobClient: Total time spent by all maps waiting after reserving slots (ms)=0
25/03/24 10:38:18 INFO mapred.JobClient: Launched map tasks=2
25/03/24 10:38:18 INFO mapred.JobClient: Data-local map tasks=2
25/03/24 10:38:18 INFO mapred.JobClient: SLOTS_MILLIS_REDUCES=12553
25/03/24 10:38:18 INFO mapred.JobClient: File Output Format Counters
25/03/24 10:38:18 INFO mapred.JobClient: Bytes Written=31
25/03/24 10:38:18 INFO mapred.JobClient: FileSystemCounters
25/03/24 10:38:18 INFO mapred.JobClient: FILE_BYTES_READ=84
25/03/24 10:38:18 INFO mapred.JobClient: HDFS_BYTES_READ=307
25/03/24 10:38:18 INFO mapred.JobClient: FILE_BYTES_WRITTEN=63980
25/03/24 10:38:18 INFO mapred.JobClient: HDFS_BYTES_WRITTEN=31
25/03/24 10:38:18 INFO mapred.JobClient: File Input Format Counters
25/03/24 10:38:18 INFO mapred.JobClient: Bytes Read=72
25/03/24 10:38:18 INFO mapred.JobClient: Map-Reduce Framework
25/03/24 10:38:18 INFO mapred.JobClient: Map output materialized bytes=90
25/03/24 10:38:18 INFO mapred.JobClient: Map input records=6
25/03/24 10:38:18 INFO mapred.JobClient: Reduce shuffle bytes=90
25/03/24 10:38:18 INFO mapred.JobClient: Spilled Records=12
25/03/24 10:38:18 INFO mapred.JobClient: Map output bytes=86
25/03/24 10:38:18 INFO mapred.JobClient: Total committed heap usage (bytes)=337780736
25/03/24 10:38:18 INFO mapred.JobClient: CPU time spent (ms)=540
25/03/24 10:38:18 INFO mapred.JobClient: Combine input records=0
25/03/24 10:38:18 INFO mapred.JobClient: SPLIT_RAW_BYTES=235
25/03/24 10:38:18 INFO mapred.JobClient: Reduce input records=6
```



- Now we will compile the Java file, then we will get the JAR file, and we will execute the code and get the required output inside the **output** folder on Hadoop.

```
ponny@ubuntu: ~/Desktop/ReduceSide
25/03/24 10:37:44 INFO mapred.JobClient: Running job: job_202503241020_0001
25/03/24 10:37:45 INFO mapred.JobClient: map 0% reduce 0%
25/03/24 10:37:58 INFO mapred.JobClient: map 100% reduce 0%
25/03/24 10:38:13 INFO mapred.JobClient: map 100% reduce 100%
25/03/24 10:38:18 INFO mapred.JobClient: Job complete: job_202503241020_0001
25/03/24 10:38:18 INFO mapred.JobClient: Counters: 29
25/03/24 10:38:18 INFO mapred.JobClient:   Job Counters
25/03/24 10:38:18 INFO mapred.JobClient:     Launched reduce tasks=1
25/03/24 10:38:18 INFO mapred.JobClient:     SLOTS_MILLIS_MAPS=14885
25/03/24 10:38:18 INFO mapred.JobClient:     Total time spent by all reduces waiting after reserving
25/03/24 10:38:18 INFO mapred.JobClient:     slots (ms)=0
25/03/24 10:38:18 INFO mapred.JobClient:     Total time spent by all maps waiting after reserving slo
25/03/24 10:38:18 INFO mapred.JobClient:     ts (ms)=0
25/03/24 10:38:18 INFO mapred.JobClient:     Launched map tasks=2
25/03/24 10:38:18 INFO mapred.JobClient:     Data-local map tasks=2
25/03/24 10:38:18 INFO mapred.JobClient:     SLOTS_MILLIS_REDUCES=12553
25/03/24 10:38:18 INFO mapred.JobClient:   File Output Format Counters
25/03/24 10:38:18 INFO mapred.JobClient:     Bytes Written=31
25/03/24 10:38:18 INFO mapred.JobClient:   FileSystemCounters
25/03/24 10:38:18 INFO mapred.JobClient:     FILE_BYTES_READ=84
25/03/24 10:38:18 INFO mapred.JobClient:     HDFS_BYTES_READ=307
25/03/24 10:38:18 INFO mapred.JobClient:     FILE_BYTES_WRITTEN=63980
25/03/24 10:38:18 INFO mapred.JobClient:     HDFS_BYTES_WRITTEN=31
25/03/24 10:38:18 INFO mapred.JobClient:   File Input Format Counters
25/03/24 10:38:18 INFO mapred.JobClient:     Bytes Read=72
25/03/24 10:38:18 INFO mapred.JobClient:   Map-Reduce Framework
25/03/24 10:38:18 INFO mapred.JobClient:     Map output materialized bytes=90
25/03/24 10:38:18 INFO mapred.JobClient:     Map input records=6
25/03/24 10:38:18 INFO mapred.JobClient:     Reduce shuffle bytes=90
25/03/24 10:38:18 INFO mapred.JobClient:     Spilled Records=12
25/03/24 10:38:18 INFO mapred.JobClient:     Map output bytes=66
25/03/24 10:38:18 INFO mapred.JobClient:     Total committed heap usage (bytes)=337780736
25/03/24 10:38:18 INFO mapred.JobClient:     CPU time spent (ms)=540
25/03/24 10:38:18 INFO mapred.JobClient:     Combine input records=0
25/03/24 10:38:18 INFO mapred.JobClient:     SPLIT_RAW_BYTES=235
25/03/24 10:38:18 INFO mapred.JobClient:     Reduce input records=0
25/03/24 10:38:18 INFO mapred.JobClient:     Reduce input groups=4
25/03/24 10:38:18 INFO mapred.JobClient:     Combine output records=0
25/03/24 10:38:18 INFO mapred.JobClient:     Physical memory (bytes) snapshot=342876160
25/03/24 10:38:18 INFO mapred.JobClient:     Virtual memory (bytes) snapshot=1155051520
25/03/24 10:38:18 INFO mapred.JobClient:     Map output records=6
25/03/24 10:38:18 INFO mapred.JobClient:     Map output records=6
ponny@ubuntu:~/Desktop/ReduceSide$ hadoop fs -cat /reduceside/Output/part-r-00000
Warning: $HADOOP_HOME is deprecated.
101 John, 5000
102 Alice, 6000
ponny@ubuntu:~/Desktop/ReduceSide$
```

Program:

```
import java.io.IOException;
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

/**
 * Reduce Side Join Example for Employee and Salary Data
 */
public class rsp {

    // Mapper Class
    public static class JoinMapper extends Mapper<Object, Text, Text, Text> {
        public void map(Object key, Text value, Context context) throws
IOException, InterruptedException {
            String[] fields = value.toString().split(",");

            // Ensure valid record
            if (fields.length >= 3) {
                String recordType = fields[0].trim(); // "A" for Employee, "B"
for Salary
                String joinKey = fields[1].trim(); // Employee ID (Join
Key)
                String details = fields[2].trim(); // Employee Name or
Salary
            }
        }
    }
}
```



```

        context.write(new Text(joinKey), new Text(recordType + "," +
details));
    }
}

// Reducer Class
public static class JoinReducer extends Reducer<Text, Text, Text, Text> {
    public void reduce(Text key, Iterable<Text> values, Context context)
throws IOException, InterruptedException {
        String employeeName = null;
        String salary = null;

        // Iterate over values
        for (Text val : values) {
            String[] tokens = val.toString().split(",");
            if (tokens.length == 2) {
                if (tokens[0].equals("A")) {
                    employeeName = tokens[1]; // Employee Name
                } else if (tokens[0].equals("B")) {
                    salary = tokens[1]; // Employee Salary
                }
            }
        }

        // Output only if both values exist
        if (employeeName != null && salary != null) {
            context.write(key, new Text(employeeName + ", " + salary));
        }
    }
}

// Driver Method
public static void main(String[] args) throws Exception {
    if (args.length < 2) {
        System.err.println("Usage: ReduceSideJoin <input path> <output
path>");
        System.exit(-1);
    }

    Configuration conf = new Configuration();
    Job job = new Job(conf);
    job.setJarByClass(rsp.class);
    job.setMapperClass(JoinMapper.class);
    job.setReducerClass(JoinReducer.class);
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(Text.class);

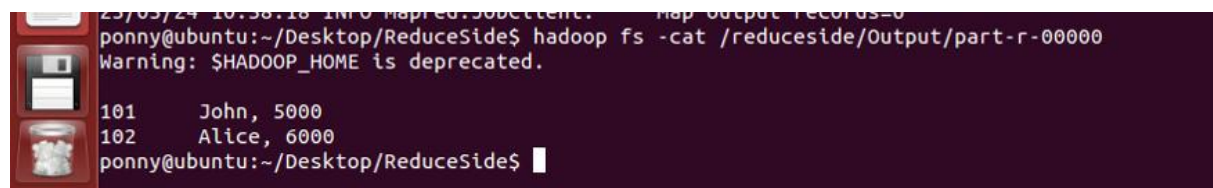
```



```
FileInputFormat.addInputPath(job, new Path(args[0]));
FileOutputFormat.setOutputPath(job, new Path(args[1]));

System.exit(job.waitForCompletion(true) ? 0 : 1);
}
}
```

Output in the terminal:



Output in the Localhost:

(no subject) - namanshm xHDFS:/reduceside/Output x+

localhost:50075/browseDirectory.jsp?dir=%2Freduceside%2FOutput&namenodeInfoPort=50070

Contents of directory /reduceside/Output

Goto :

[Go to parent directory](#)

Name	Type	Size	Replication	Block Size	Modification Time	Permission	Owner	Group
_SUCCESS	file	0 KB	1	64 MB	2025-03-24 10:38	rw-r--r--	ponny	supergroup
_logs	dir				2025-03-24 10:37	rw-r--r--	ponny	supergroup
part-r-00000	file	0.03 KB	1	64 MB	2025-03-24 10:38	rw-r--r--	ponny	supergroup

[Go back to DFS home](#)

Local logs

[Log](#) directory

This is [Apache Hadoop](#) release 1.0.4



(no subject) - namanshm

HDFS:/reduceside/Output/

localhost:50075/browseBlock.jsp?blockId=4348233580577086036&blockSize=31&genstamp=1

File: /reduceside/Output/part-r-00000

Goto : /reduceside/Output go

[Go back to dir listing](#)
[Advanced view/download options](#)

101	John, 5000
102	Alice, 6000

[Download this file](#)
[Tail this file](#)

Chunk size to view (in bytes, up to file's DFS block size): 32768 Refresh

Total number of blocks: 1
4348233580577086036: [127.0.0.1:50010](#)

[Go back to DFS home](#)

Result:

The **Reduce Side Join** in Hadoop MapReduce was successfully implemented.

The program merged **customer details with their respective transaction amounts** using the distributed computing framework, enabling efficient large-scale data processing.