■ 맵리듀스 기본 프로그래밍

1) 초창기 하둡 초창기에는 다른 툴이 없었음 / 처리 및 분석을 하기 위해서는 맵리듀스 프로그래밍으로

2) 이후 맵리듀스 프로그래밍이 어려워서 / 데이터과학자들은 java등 프로그래밍을 모르기 때문에 페이스북에서 hive를 만들었음

- SQL구문처럼 작성 가능

3) 맵퍼(Mapper)와 리듀서(Reducer) 구현후 실행하면 됨

4) WordCount.java 파일 생성

- test디렉토리 생성

```
[bigdata@server01 test]$ mkdir test
```

- test디렉토리로 이동

```
[bigdata@server01 test]$ cd test
```

- WordCount.java 파일 생성

```
[bigdata@server01 test]$ vi WordCount.java
```

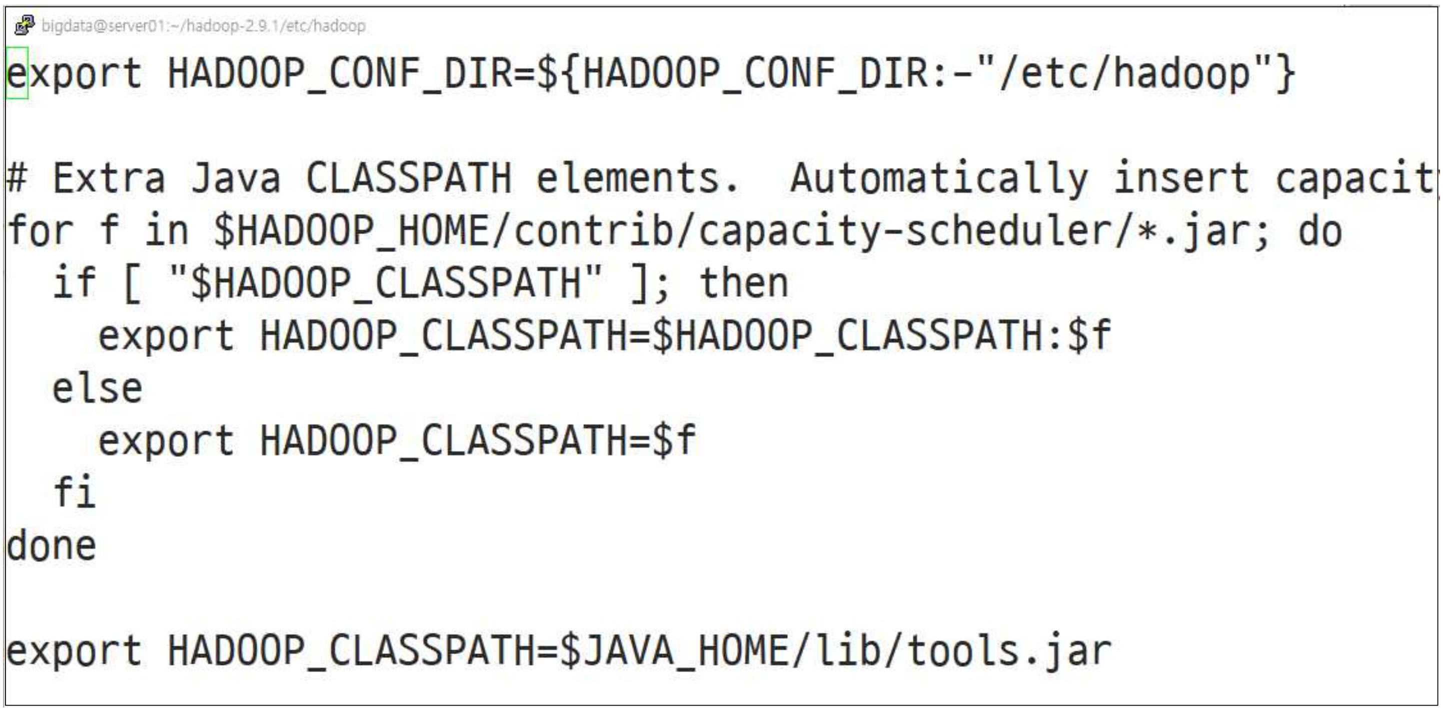
- WordCount.java

```
import java.io.IOException;
import java.util.StringTokenizer;

import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
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;

public class WordCount {

  public static class TokenizerMapper
       extends Mapper<Object, Text, Text, IntWritable>{

    private final static IntWritable one = new IntWritable(1);
    private Text word = new Text();

    public void map(Object key, Text value, Context context
                    ) throws IOException, InterruptedException {
      StringTokenizer itr = new StringTokenizer(value.toString());
      while (itr.hasMoreTokens()) {
        word.set(itr.nextToken());
        context.write(word, one);
      }
    }
  }

  public static class IntSumReducer
       extends Reducer<Text,IntWritable,Text,IntWritable> {
    private IntWritable result = new IntWritable();

    public void reduce(Text key, Iterable<IntWritable> values,
                       Context context
                       ) throws IOException, InterruptedException {
      int sum = 0;
      for (IntWritable val : values) {
        sum += val.get();
      }
      result.set(sum);
      context.write(key, result);
    }
  }

  public static void main(String[] args) throws Exception {
    Configuration conf = new Configuration();
    Job job = Job.getInstance(conf, "word count");
    job.setJarByClass(WordCount.class);
    job.setMapperClass(TokenizerMapper.class);
    job.setCombinerClass(IntSumReducer.class);
    job.setReducerClass(IntSumReducer.class);
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(IntWritable.class);
    FileInputFormat.addInputPath(job, new Path(args[0]));
    FileOutputFormat.setOutputPath(job, new Path(args[1]));
    System.exit(job.waitForCompletion(true) ? 0 : 1);
  }
}
```

- 'i'키 누르고 입력후 :wq 으로 저장

- java 파일을 컴파일하고 jar로 압축하기 위하여 hadoop-env.sh 에 HADOOP_CLASSPATH 지정

```
[bigdata@server01]$ vi hadoop-2.9.1/etc/hadoop/hadoop-env.sh
```



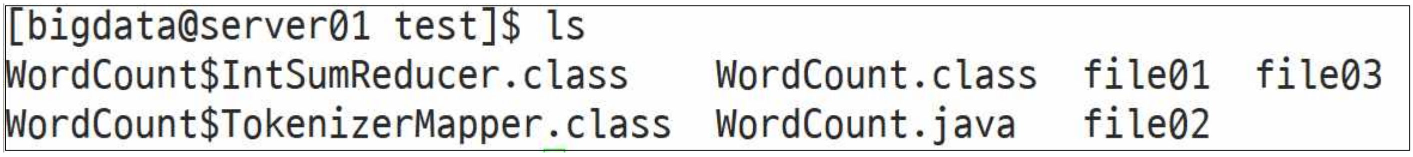
```
export HADOOP_CONF_DIR=${HADOOP_CONF_DIR:-"/etc/hadoop"}

# Extra Java CLASSPATH elements.  Automatically insert capacit
for f in $HADOOP_HOME/contrib/capacity-scheduler/*.jar; do
  if [ "$HADOOP_CLASSPATH" ]; then
    export HADOOP_CLASSPATH=$HADOOP_CLASSPATH:$f
  else
    export HADOOP_CLASSPATH=$f
  fi
done

export HADOOP_CLASSPATH=$JAVA_HOME/lib/tools.jar
```

- WordCount.java 를 컴파일

```
[bigdata@server01]$ ./hadoop-2.9.1/bin/hadoop com.sun.tools.javac.Main
                     /home/bigdata/test/WordCount.java
```

- 확인

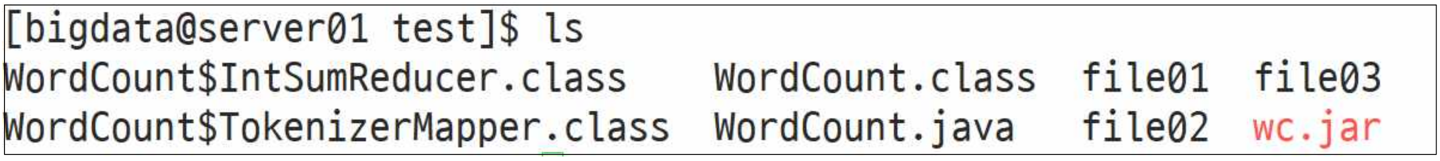
```
[bigdata@server01]$ cd test
[bigdata@server01 test]$ ls
```



```
[bigdata@server01 test]$ ls
WordCount$IntSumReducer.class    WordCount.class  file01  file03
WordCount$TokenizerMapper.class  WordCount.java   file02
```

- jar파일로 압축

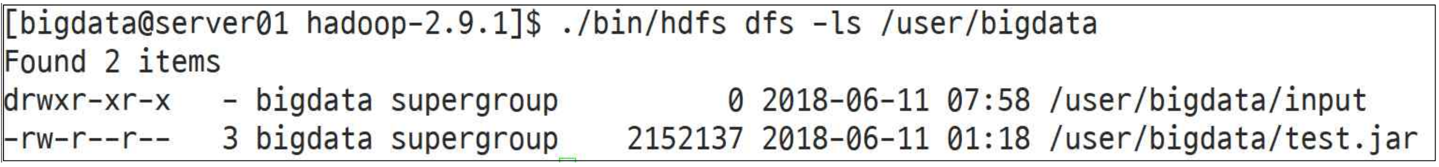
```
[bigdata@server01 test]$ jar cf wc.jar WordCount*.class
```



```
[bigdata@server01 test]$ ls
WordCount$IntSumReducer.class    WordCount.class  file01  file03
WordCount$TokenizerMapper.class  WordCount.java   file02  wc.jar
```

- 하둡에 input 디렉토리 생성

```
[bigdata@server01 ]$ ./hadoop-2.9.1/bin/hdfs dfs -mkdir /user/bigdata/input
```



```
[bigdata@server01 hadoop-2.9.1]$ ./bin/hdfs dfs -ls /user/bigdata
Found 2 items
drwxr-xr-x   - bigdata supergroup          0 2018-06-11 07:58 /user/bigdata/input
-rw-r--r--   3 bigdata supergroup    2152137 2018-06-11 01:18 /user/bigdata/test.jar
```

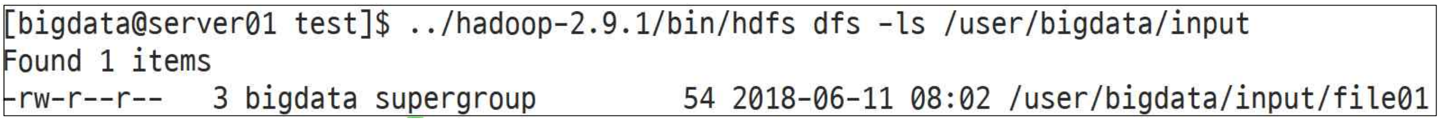
- 단어 숫자를 세기위한 파일(들) 생성

```
[bigdata@server01 test]$ vi file01
```

```
Hello Hadoop
Hi MapReduce
Hello Java
Hadoop Fighting
```

- input 디렉토리에 파일 입력(put)

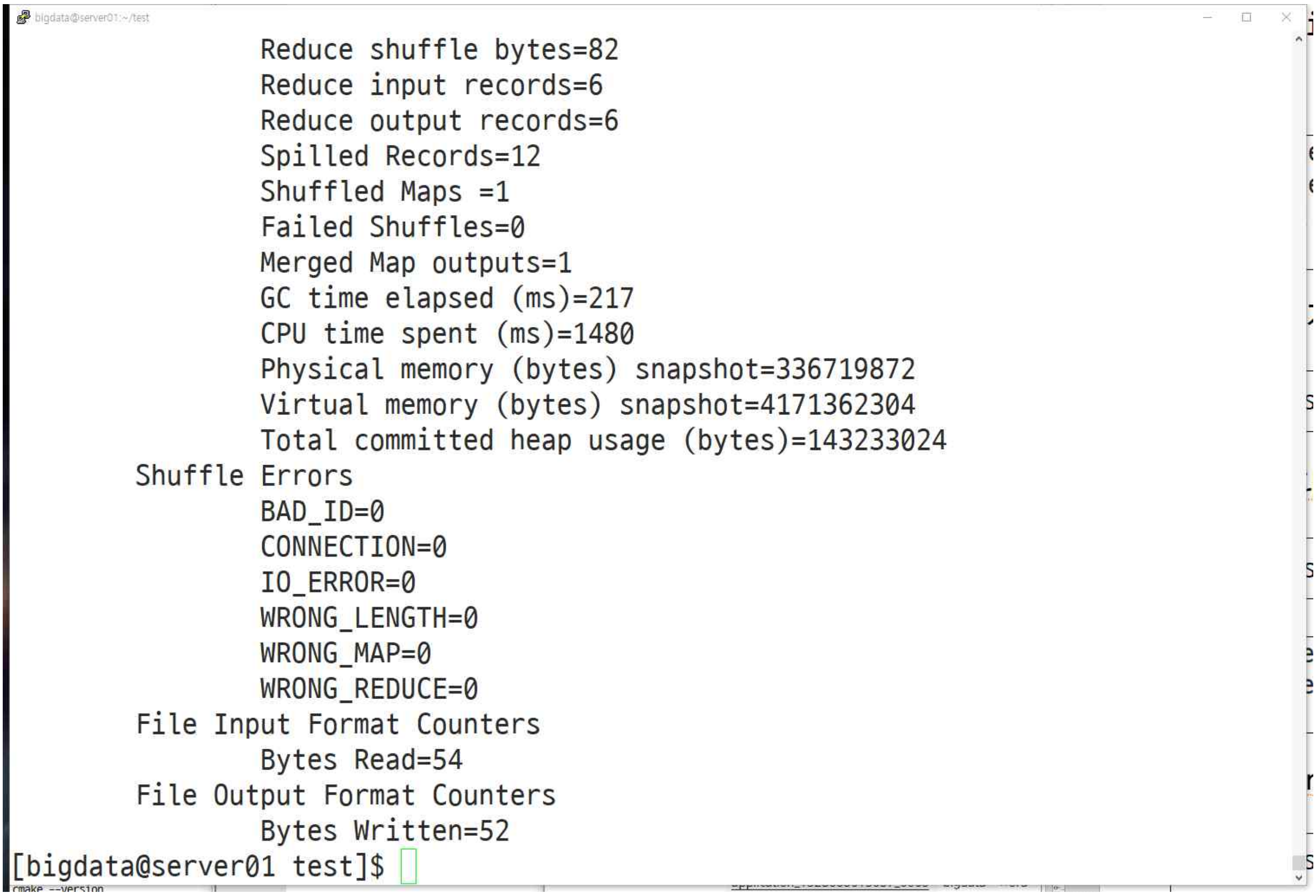
```
[bigdata@server01 test]$ ../hadoop-2.9.1/bin/hdfs dfs -put file01 /user/bigdata/input
```



```
[bigdata@server01 test]$ ../hadoop-2.9.1/bin/hdfs dfs -ls /user/bigdata/input
Found 1 items
-rw-r--r--   3 bigdata supergroup         54 2018-06-11 08:02 /user/bigdata/input/file01
```

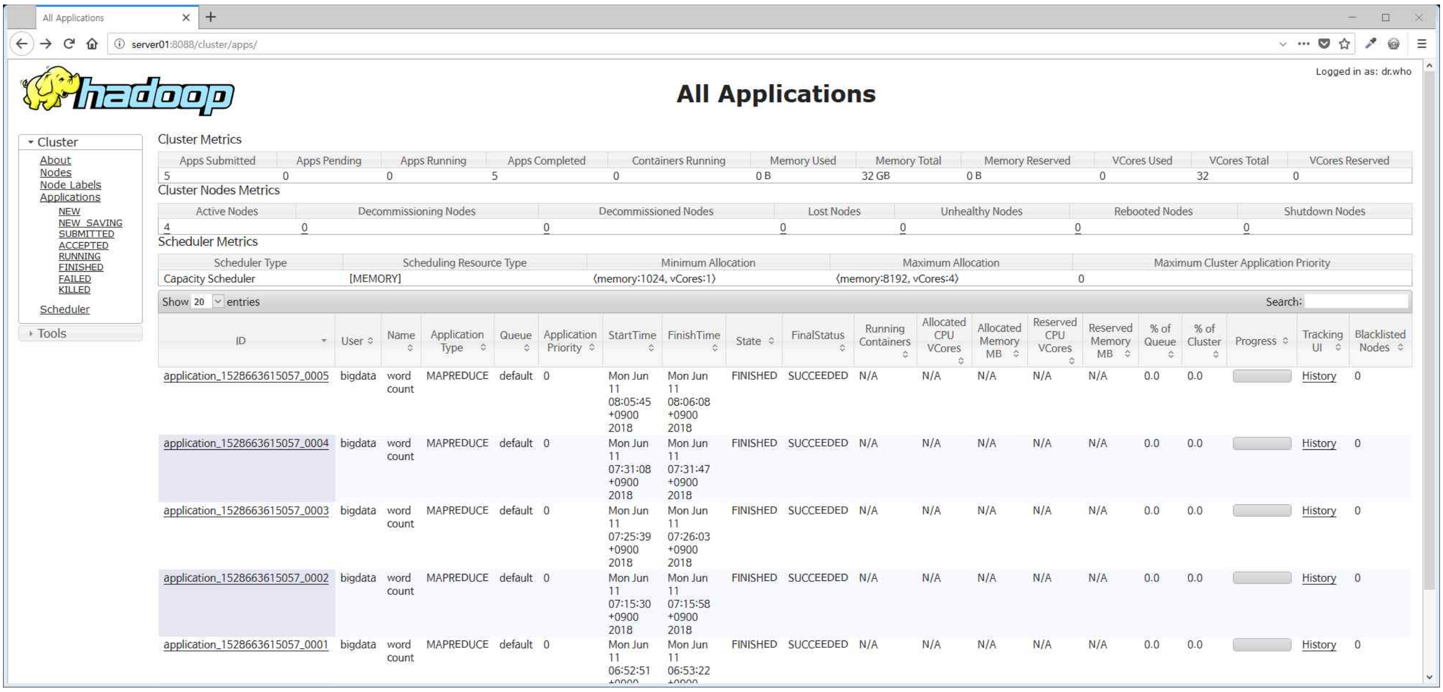
- WordCount 실행

```
[bigdata@server01 test]$ ../hadoop-2.9.1/bin/hadoop jar wc.jar WordCount
/user/bigdata/input /user/bigdata/output
```



```
                Reduce shuffle bytes=82
                Reduce input records=6
                Reduce output records=6
                Spilled Records=12
                Shuffled Maps =1
                Failed Shuffles=0
                Merged Map outputs=1
                GC time elapsed (ms)=217
                CPU time spent (ms)=1480
                Physical memory (bytes) snapshot=336719872
                Virtual memory (bytes) snapshot=4171362304
                Total committed heap usage (bytes)=143233024
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=54
        File Output Format Counters
                Bytes Written=52
[bigdata@server01 test]$
```

- server01:8088에서 yarn 애플리케이션 실행 확인



Logged in as: dr.who

All Applications

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved
5	0	0	5	0	0 B	32 GB	0 B	0	32	0

Cluster Nodes Metrics

Active Nodes	Decommissioning Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Rebooted Nodes	Shutdown Nodes
4	0	0	0	0	0	0

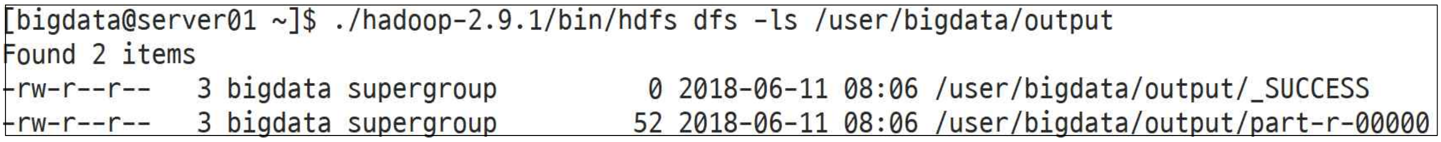
Scheduler Metrics

Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation	Maximum Cluster Application Priority
Capacity Scheduler	[MEMORY]	⟨memory:1024, vCores:1⟩	⟨memory:8192, vCores:4⟩	0

ID	User	Name	Application Type	Queue	Application Priority	StartTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU VCores	Allocated Memory MB	Reserved CPU VCores	Reserved Memory MB	% of Queue	% of Cluster	Progress	Tracking UI	Blacklisted Nodes
application_1528663615057_0005	bigdata	word count	MAPREDUCE	default	0	Mon Jun 11 08:05:45 +0900 2018	Mon Jun 11 08:06:08 +0900 2018	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	0.0	0.0		History	0
application_1528663615057_0004	bigdata	word count	MAPREDUCE	default	0	Mon Jun 11 07:31:08 +0900 2018	Mon Jun 11 07:31:47 +0900 2018	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	0.0	0.0		History	0
application_1528663615057_0003	bigdata	word count	MAPREDUCE	default	0	Mon Jun 11 07:25:39 +0900 2018	Mon Jun 11 07:26:03 +0900 2018	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	0.0	0.0		History	0
application_1528663615057_0002	bigdata	word count	MAPREDUCE	default	0	Mon Jun 11 07:15:30 +0900 2018	Mon Jun 11 07:15:58 +0900 2018	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	0.0	0.0		History	0
application_1528663615057_0001	bigdata	word count	MAPREDUCE	default	0	Mon Jun 11 06:52:51	Mon Jun 11 06:53:22	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	0.0	0.0		History	0

- /user/bigdata/output 폴더 확인

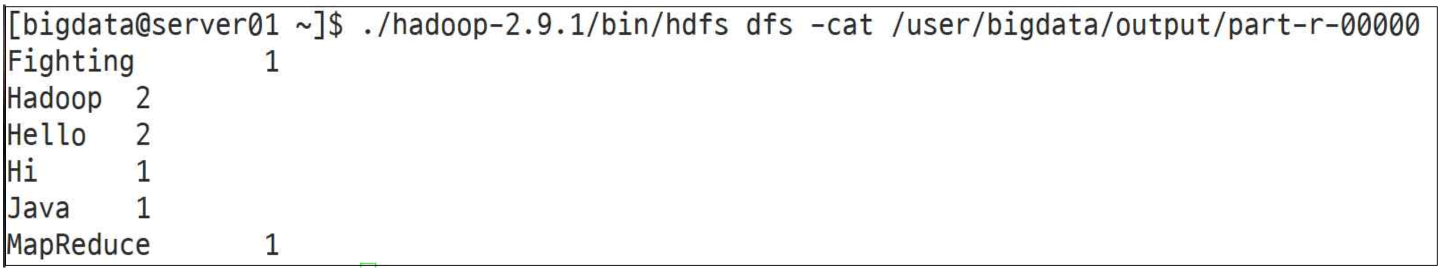
```
[bigdata@server01 ~]$ ./hadoop-2.9.1/bin/hdfs dfs -ls /user/bigdata/output
```



```
[bigdata@server01 ~]$ ./hadoop-2.9.1/bin/hdfs dfs -ls /user/bigdata/output
Found 2 items
-rw-r--r--   3 bigdata supergroup          0 2018-06-11 08:06 /user/bigdata/output/_SUCCESS
-rw-r--r--   3 bigdata supergroup         52 2018-06-11 08:06 /user/bigdata/output/part-r-00000
```

- 실제 작동 확인

```
[bigdata@server01 ~]$ ./hadoop-2.9.1/bin/hdfs dfs -cat
                      /user/bigdata/output/part-r-00000
```



```
[bigdata@server01 ~]$ ./hadoop-2.9.1/bin/hdfs dfs -cat /user/bigdata/output/part-r-00000
Fighting        1
Hadoop  2
Hello   2
Hi      1
Java    1
MapReduce       1
```