

REPORT

빅데이터 처리 시스템 개발
정기 수행평가

과 목 명	통합구현
능 력 단 위	빅데이터 처리 시스템 개발
성 명	장경준

수행평가 실습 보고서			
능력단위 요소	빅데이터 처리 시스템 개발	작 성 자	장경준
평가방법	서술형	작성일자	
단 계 명	구현	문서번호	#1

1.SDK(System Development Kit) 또는 개발 도구 정보

구분	내용
SDK(개발도구) 명칭	CentOS
SDK(개발도구) 버전	CentOS 8
SDK(개발도구)설명 (주요기능의 설명)	<ul style="list-style-type: none"> ● 리눅스 서버 1인자인 RHEL을 철저히게 반영한 오픈소스 플랫폼. ● 최고의 서버 운영체제.
SDK(개발도구) 사용방법 (가격, 절차 등)	오픈소스

2. 평가문항 풀이

1. 맵리듀스 실습

1-1. 하둡 실행 및 디렉터리 생성

```
[root@Server101 ~]# start-all.sh
```

```
[root@Server101 ~]# hdfs dfs -mkdir /MapReduce
```

1-2. 자바 파일 프로그래밍

[main]

```
1 package sub1;
2
3 import java.io.IOException;
4
5 import org.apache.hadoop.conf.Configuration;
6 import org.apache.hadoop.fs.Path;
7 import org.apache.hadoop.io.IntWritable;
8 import org.apache.hadoop.io.Text;
9 import org.apache.hadoop.mapreduce.Job;
10 import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
11 import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
12 import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
13 import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;
14
15 public class KeywordCountMain {
16     public static void main(String[] args) throws IOException, ClassNotFoundException, InterruptedException {
17
18         Configuration conf = new Configuration();
19         Job job = new Job(conf, "KeywordCount");
20
21         job.setJarByClass(KeywordCountMain.class);
22         job.setMapperClass(KeywordCountMapper.class);
23         job.setReducerClass(KeywordCountReducer.class);
24
25         job.setInputFormatClass(TextInputFormat.class);
26         job.setOutputFormatClass(TextOutputFormat.class);
27
28         job.setOutputKeyClass(Text.class);
29         job.setOutputValueClass(IntWritable.class);
30
31         FileInputFormat.addInputPath(job, new Path(args[0]));
32         FileOutputFormat.setOutputPath(job, new Path(args[1]));
33
34         job.waitForCompletion(true);
35
36         System.out.println("KeywordCount MapReduce End...");
37     }
38 }
```

[mapper]

```
1 package sub1;
2
3 import java.io.IOException;
4 import java.util.StringTokenizer;
5
6 import org.apache.hadoop.io.IntWritable;
7 import org.apache.hadoop.io.LongWritable;
8 import org.apache.hadoop.io.Text;
9 import org.apache.hadoop.mapreduce.Mapper;
10
11 public class KeywordCountMapper extends Mapper<LongWritable, Text, Text, IntWritable>{
12
13     private IntWritable one = new IntWritable(1);
14     private Text txt = new Text();
15
16     @Override
17     protected void map(LongWritable key, Text value, Mapper<LongWritable, Text, Text, IntWritable>.Context context) throws IOException, InterruptedException {
18
19         String line = value.toString();
20         String[] tokens = line.split(",");
21
22         txt.set(tokens[1]);
23         context.write(txt, one);
24     }
25 }
```

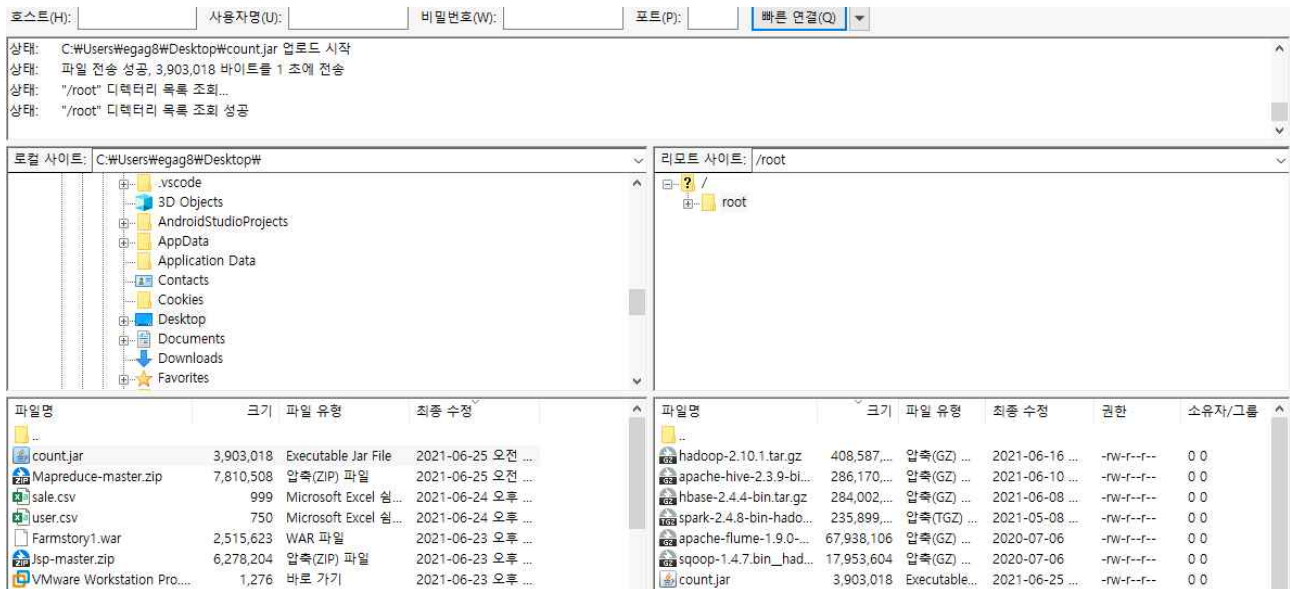
[reducer]


```

16 public class KeywordCountReducer extends Reducer<Text, IntWritable, Text, IntWritable>{
17
18     private Map<String, Integer> map = new HashMap<>();
19     @Override
20     protected void reduce(Text key, Iterable<IntWritable> values, Reducer<Text, IntWritable, Text, IntWritable>.Context context) throws IOException, InterruptedException {
21
22         int sum = 0;
23         for(IntWritable val : values) {
24             sum += val.get();
25         }
26         map.put(key.toString(), sum);
27     }
28
29     @Override
30     protected void cleanup(Reducer<Text, IntWritable, Text, IntWritable>.Context context) throws IOException, InterruptedException {
31         // TODO: Write the map to the output
32         List<String> list = new ArrayList<>();
33         list.addAll(map.keySet());
34         Collections.sort(list, new Comparator<String>() {
35             @Override
36             public int compare(String o1, String o2) {
37
38                 Integer v1 = map.get(o1);
39                 Integer v2 = map.get(o2);
40
41                 int result = v1.compareTo(v2);
42                 return result;
43             }
44         });
45
46         Collections.reverse(list);
47
48         Iterator<String> iter = list.iterator();
49
50         while(iter.hasNext()) {
51
52             String word = iter.next();
53             int sum = map.get(word);
54
55             context.write(new Text(word), new IntWritable(sum));

```

1-3. jar 파일 전송

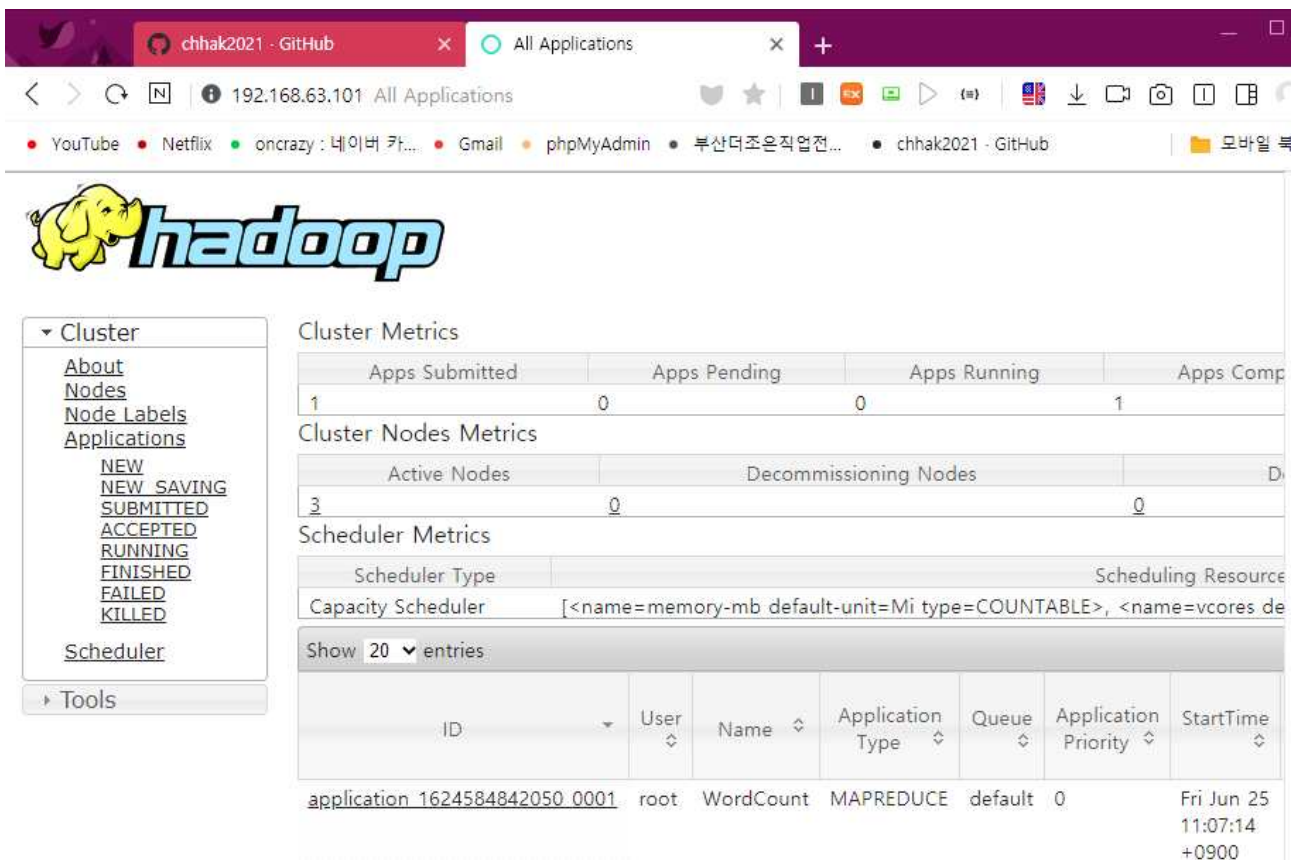


1-4. 맵리듀스 실행

```
[root@Server101 ~]# yarn jar count.jar sub1.WordCountMain /MapReduce/sample.txt /MapReduce/output
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/home/bigdata/hadoop-2.10.1/share/hadoop/common/lib/hadoop-auth-2.10.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
21/06/24 22:07:09 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
21/06/24 22:07:10 INFO client.RMPProxy: Connecting to ResourceManager at Server101/192.168.63.101:8032
21/06/24 22:07:11 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
21/06/24 22:07:13 INFO input.FileInputFormat: Total input files to process : 1
21/06/24 22:07:13 INFO mapreduce.JobSubmitter: number of splits:1
21/06/24 22:07:13 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1624584842050_0001
21/06/24 22:07:14 INFO conf.Configuration: resource-types.xml not found
21/06/24 22:07:14 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
21/06/24 22:07:14 INFO resource.ResourceUtils: Adding resource type - name = memory-mb, units = Mi, type = COUNTABLE
21/06/24 22:07:14 INFO resource.ResourceUtils: Adding resource type - name = vcores, units = , type = COUNTABLE
21/06/24 22:07:15 INFO impl.YarnClientImpl: Submitted application application_1624584842050_0001
21/06/24 22:07:15 INFO mapreduce.Job: The url to track the job: http://Server101:8088/proxy/application_1624584842050_0001/
21/06/24 22:07:15 INFO mapreduce.Job: Running job: job_1624584842050_0001
```

1-5 실행 확인

```
For more detailed output, check the application tracking page: http://Server101:8088/cluster/
app/application_1624584842050_0001 Then click on links to logs of each attempt.
. Failing the application.
21/06/24 22:07:35 INFO mapreduce.Job: Counters: 0
WordCount MapReduce 종료...
[root@Server101 ~]# yarn application -list
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.Kerberos
Util (file:/home/bigdata/hadoop-2.10.1/share/hadoop/common/lib/hadoop-auth-2.10.1.jar) to met
hod sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.auth
entication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access op
erations
WARNING: All illegal access operations will be denied in a future release
21/06/24 22:08:02 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your p
latform... using builtin-java classes where applicable
21/06/24 22:08:03 INFO client.RMPProxy: Connecting to ResourceManager at Server101/192.168.63.
101:8032
Total number of applications (application-types: [], states: [SUBMITTED, ACCEPTED, RUNNING] a
nd tags: []):0
      Application-Id      Application-Name      Application-Type      User
Queue      State      Final-State      Progress
Tracking-URL
[root@Server101 ~]#
```



The screenshot shows a web browser window with the Hadoop web interface. The browser tabs include 'chhak2021 - GitHub' and 'All Applications'. The address bar shows '192.168.63.101 All Applications'. The page features the Hadoop logo and a sidebar with navigation links: Cluster (About, Nodes, Node Labels, Applications), Scheduler, and Tools. The main content area displays 'Cluster Metrics' with a table showing 1 App Submitted, 0 Apps Pending, 0 Apps Running, and 1 App Completed. Below this is 'Cluster Nodes Metrics' showing 3 Active Nodes and 0 Decommissioning Nodes. The 'Scheduler Metrics' section shows the Capacity Scheduler. At the bottom, a table lists applications, with the first entry being 'application_1624584842050_0001' submitted by 'root' for the 'WordCount' application type, in the 'MAPREDUCE' state, on the 'default' queue, with a priority of 0, and a start time of 'Fri Jun 25 11:07:14 +0900'.

Apps Submitted	Apps Pending	Apps Running	Apps Completed
1	0	0	1

Active Nodes	Decommissioning Nodes
3	0

ID	User	Name	Application Type	Queue	Application Priority	StartTime
application_1624584842050_0001	root	WordCount	MAPREDUCE	default	0	Fri Jun 25 11:07:14 +0900

2. 하이브 실습

2-1. 계정 생성 및 디렉터리 생성

```
MariaDB [(none)]> CREATE DATABASE hive_metastore_db;
ERROR 1007 (HY000): Can't create database 'hive_metastore_db'; database exists
MariaDB [(none)]> CREATE USER 'hive'@'localhost' IDENTIFIED BY '1234';
ERROR 1396 (HY000): Operation CREATE USER failed for 'hive'@'localhost'
MariaDB [(none)]> CREATE USER 'hive'@'%' IDENTIFIED BY '1234';
```

```
[root@Server101 ~]# hdfs dfs -mkdir /hive/warehouse
```

2-2. 하이브 설치

```
[root@Server101 ~]# cd /home/bigdata/
[root@Server101 bigdata]# ll
합계 0
drwxr-xr-x  7 root root 187  6월 16 02:53 apache-flume-1.9.0-bin
drwxr-xr-x 10 root root 184  6월 22 01:55 apache-hive-2.3.9-bin
lrwxrwxrwx  1 root root  23  6월 16 02:55 flume -> apache-flume-1.9.0-bin/
lrwxrwxrwx  1 root root  14  6월 15 22:01 hadoop -> hadoop-2.10.1/
drwxr-xr-x 11 1000 1000 172  6월 16 01:10 hadoop-2.10.1
lrwxrwxrwx  1 root root  12  6월 17 02:56 hbase -> hbase-2.4.4/
drwxr-xr-x  9 root root 216  6월 17 03:10 hbase-2.4.4
lrwxrwxrwx  1 root root  22  6월 22 01:55 hive -> apache-hive-2.3.9-bin/
lrwxrwxrwx  1 root root  26  6월 23 01:52 spark -> spark-2.4.8-bin-hadoop2.7/
drwxr-xr-x 13  501 1000 211  5월  8 05:22 spark-2.4.8-bin-hadoop2.7
lrwxrwxrwx  1 root root  30  6월 17 01:45 sqoop -> sqoop-1.4.7.bin__hadoop-2.6.0/
drwxr-xr-x  9 1000 1000 318 12월 18 2017 sqoop-1.4.7.bin__hadoop-2.6.0
```

2-3. 환경변수 설정

```
13 export JAVA_HOME=/usr/lib/jvm/java-11-openjdk
14 export PATH=$PATH:$JAVA_HOME/bin
15 export HADOOP_HOME=/home/bigdata/hadoop
16 export HADOOP_MAPRED_HOME=$HADOOP_HOME
17 export HADOOP_COMMON_HOME=$HADOOP_HOME
18 export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
19 export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"
20 export YARN_HOME=$HADOOP_HOME
21 export HADOOP_CONF_DIR=$HADOOP_HOME/etc/hadoop
22 export YARN_CONF_DIR=$HADOOP_HOME/etc/hadoop
23 export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin
24 export CLASS_PATH=$JAVA_HOME/lib:$CLASS_PATH
25 export FLUME_HOME=/home/bigdata/flume
26 export FLUME_CONF_DIR=/home/bigdata/flume/conf
27 export SQOOP_HOME=/home/bigdata/sqoop
28 export SQOOP_CONF_DIR=/home/bigdata/sqoop/conf
29 export PATH=$PATH:$SQOOP_HOME/bin
30 export HBASE_HOME=/home/bigdata/hbase
31 export PATH=$PATH:$HBASE_HOME/bin
32 export HIVE_HOME=/home/bigdata/hive
33 export PATH=$PATH:$HIVE_HOME/bin
34 export SPARK_HOME=/home/bigdata/spark
35 export PATH=$PATH:$SPARK_HOME/bin
```


2-4. 주석해제

```
48 HADOOP_HOME=/home/bigdata/hadoop
```

2-5. hive-site.xml 수정

```
1 <configuration>
2   <property>
3     <name>hive.metastore.warehouse.dir</name>
4     <value>/hive/warehouse</value>
5   </property>
6   <property>
7     <name>javax.jdo.option.ConnectionURL</name>
8     <value>jdbc:mysql://localhost:3306/hive_metastore_db?
9       createDatabaseIfNotExist=true</value>
10  </property>
11  <property>
12    <name>javax.jdo.option.ConnectionDriverName</name>
13    <value>com.mysql.jdbc.Driver</value>
14  </property>
15  <property>
16    <name>javax.jdo.option.ConnectionUserName</name>
17    <value>hive</value>
18  </property>
19  <property>
20    <name>javax.jdo.option.ConnectionPassword</name>
21    <value>1234</value>
22  </property>
23 </configuration>
```

2-6. 커넥터 추가

```
/home/bigdata/hive/lib/mysql-connector-java-5.1.49-bin.jar
[root@Server101 lib]# find 49-bin $HIVE_HOME/lib
```

2-7. metastore 초기화

```
[root@Server101 lib]# schematool -initSchema -dbType mysql
```

2-8. 테이블 생성

```
hive> CREATE TABLE `Naver_in`(  
  > rank int,  
  > keyword String,  
  > rdate String  
  > )  
  > row format delimited  
  > fields terminated by "."  
  > tblproperties("skip.header.line.count"="1");  
OK  
Time taken: 6.334 seconds
```

2-9. 데이터 로드

```
hive> LOAD DATA INPATH '/naver/20-07-19/*' OVERWRITE INTO TABLE Naver_in;
```

2-10. 집계

```
hive> select keyword, sum(1) as total from Naver_in  
  > group by keyword order by total desc;  
□
```