

5. MapReduce & HDFS 명령어

목 차

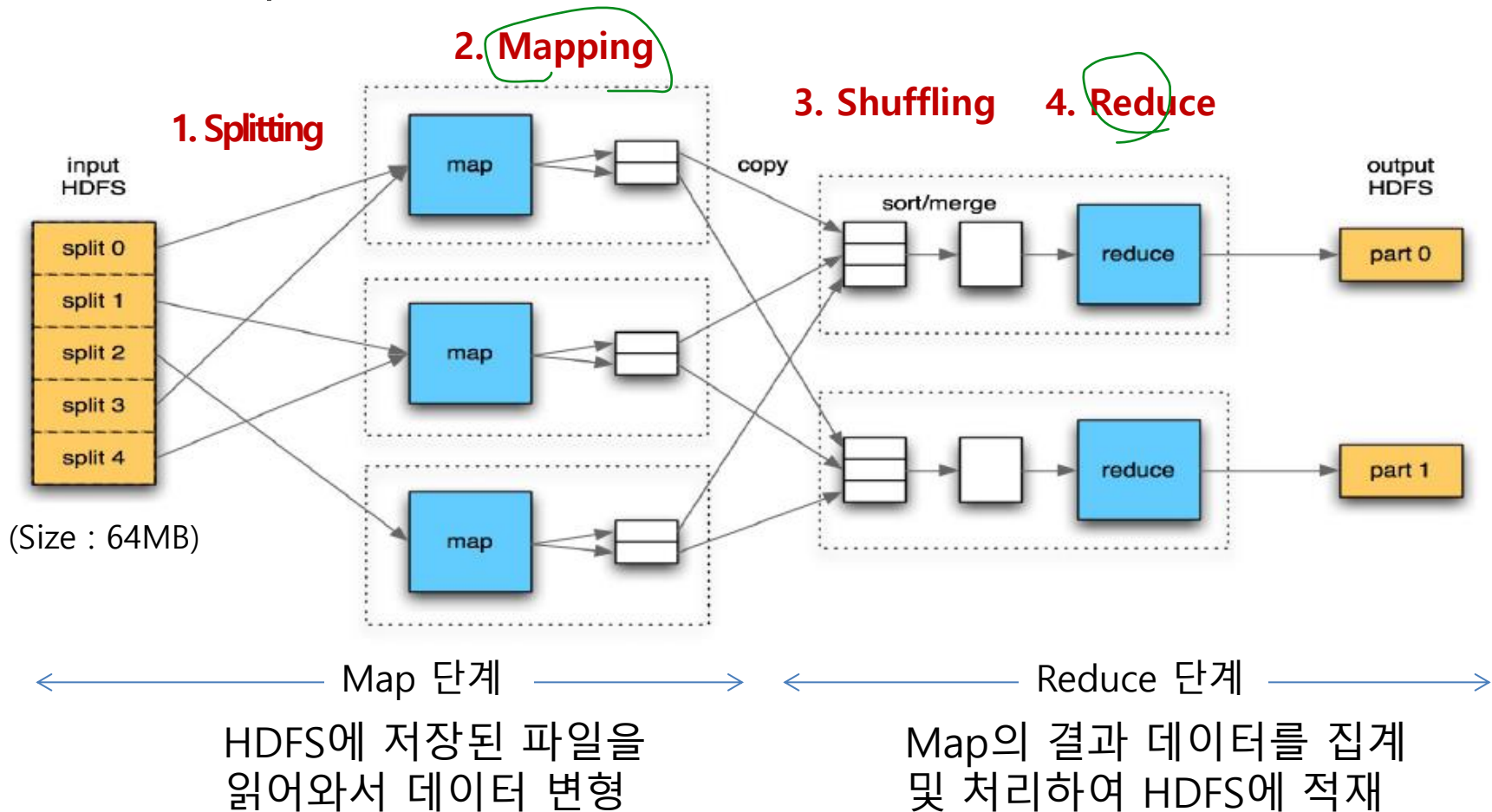
1. MapReduce 개요
2. Master/Slave 서버 ON
3. Hadoop/Yarn/Historyserver 시작
4. HDFS 명령어
5. MapReduce : Word Count 실습
6. HDFS 명령어 실습
7. Hadoop/Yarn/Historyserver 종료

1. MapReduce 개요

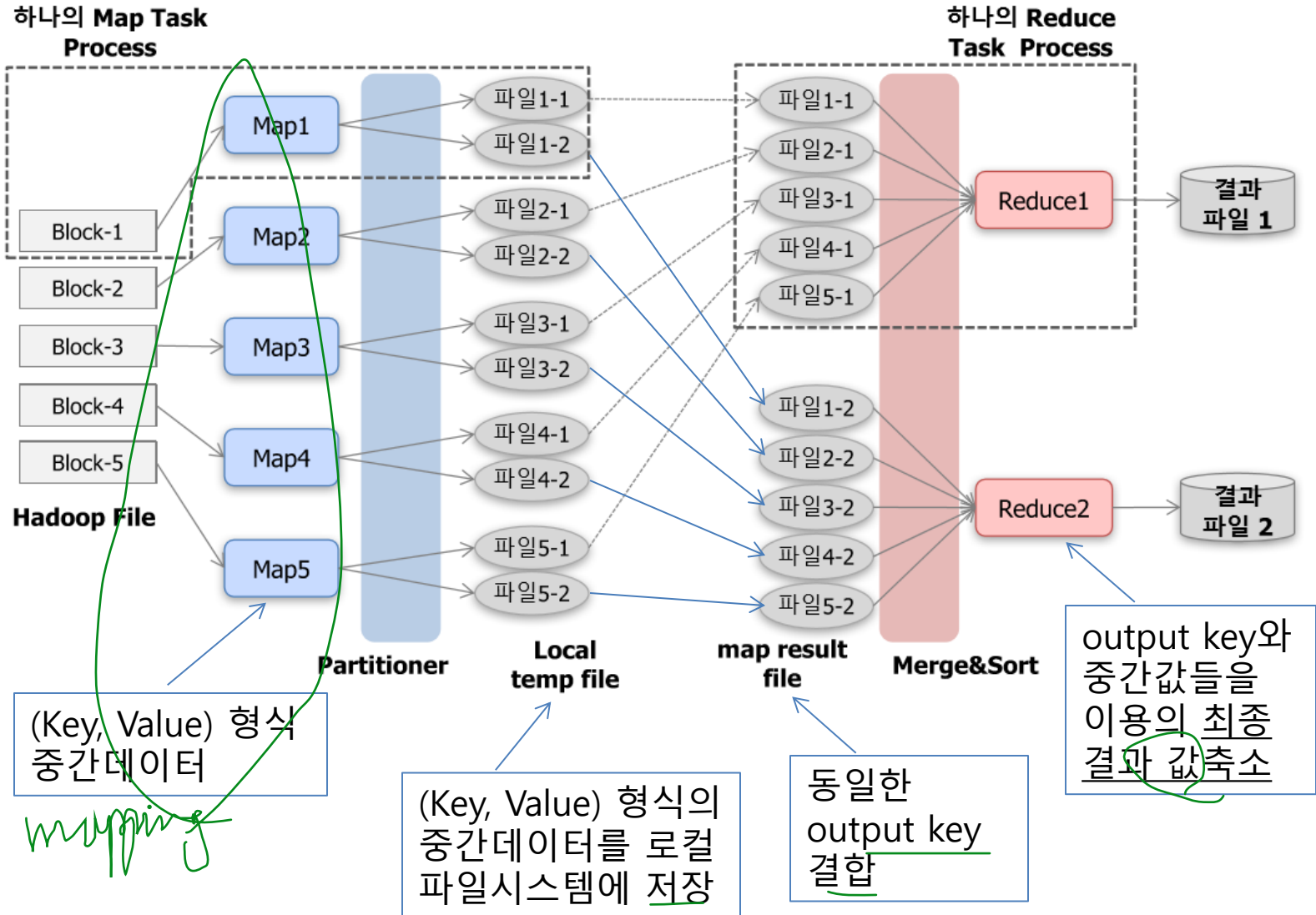
- HDFS 파일 대상 분산배치분석 지원 프레임워크
- 애플리케이션 구현 시 데이터 전송, 분산 처리, 내고장성 등의 복잡한 처리 담당
- 맵(Map)과 리듀스(Reduce) 두 단계 처리
 - ✓ 맵 : 입력 파일 한 줄 읽기 → 데이터 변형
 - ✓ 리듀스 : 맵의 결과 집계(Aggregation)
- 애플리케이션 예
 - ✓ Word Counter

1. MapReduce 개요

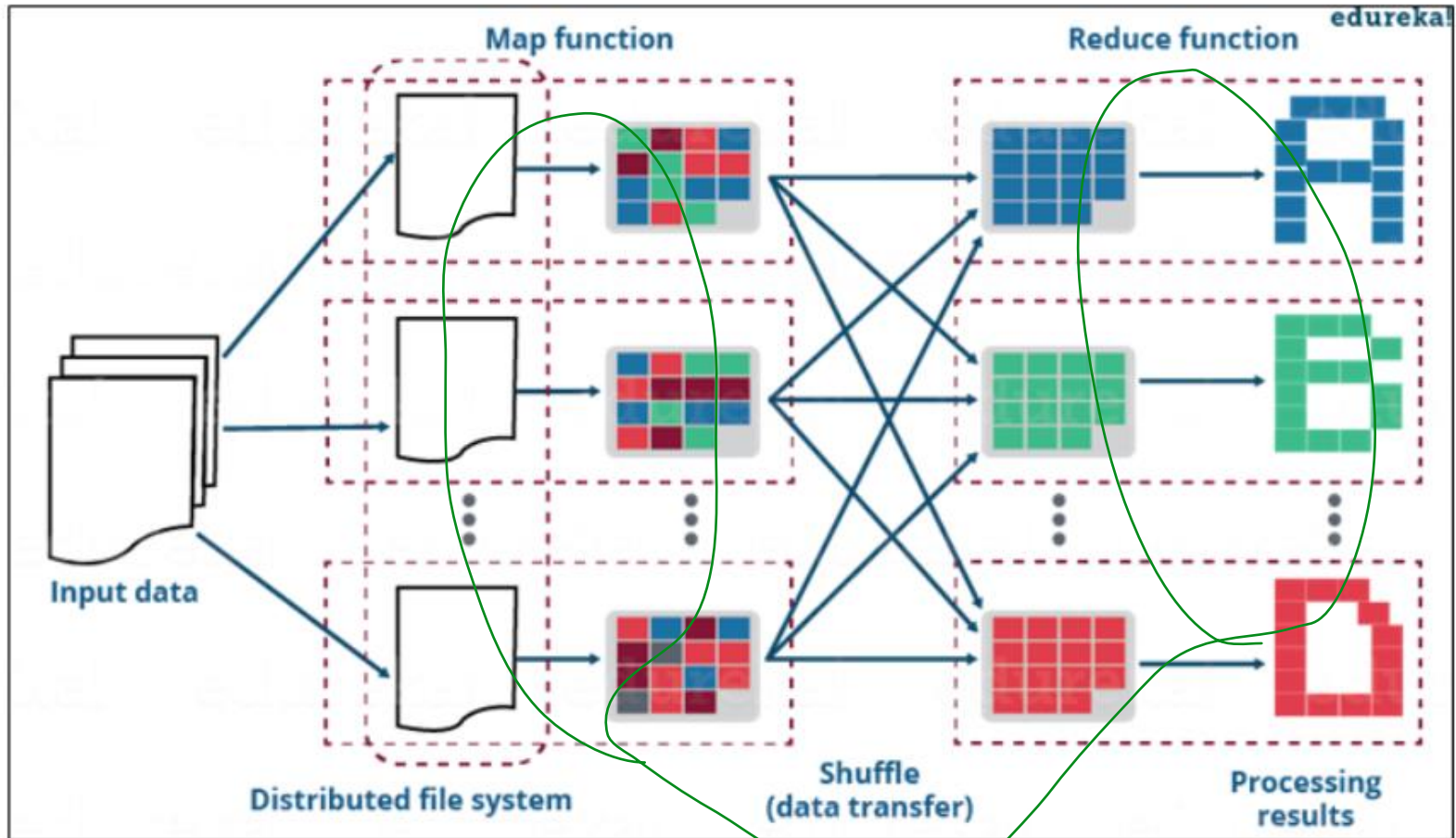
- 효과적인 분산 컴퓨팅을 위한 프로그래밍 모델
- Unix Pipeline 과 유사한 동작 방식



MapReduce 데이터 흐름



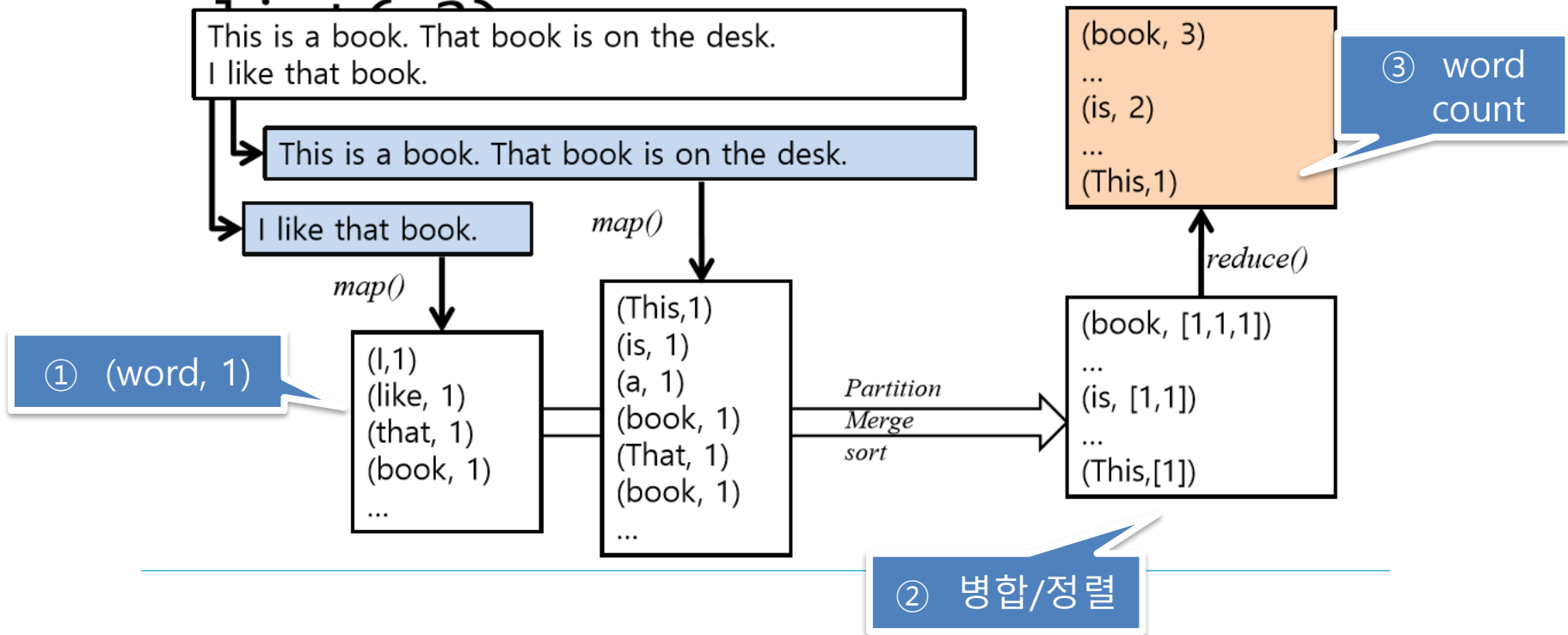
■ 맵리듀스(MapReduce)



- 맵(Map) : 입력파일을 한 줄 씩 읽어서 데이터를 변형시키는 역할
- 리듀스(Reduce) : 맵의 결과 데이터를 집계/처리 하는 역할

MapReduce Sample

- `map (k1,v1) → list(k2,v2)`
- `reduce (k2, list (v2)) →`

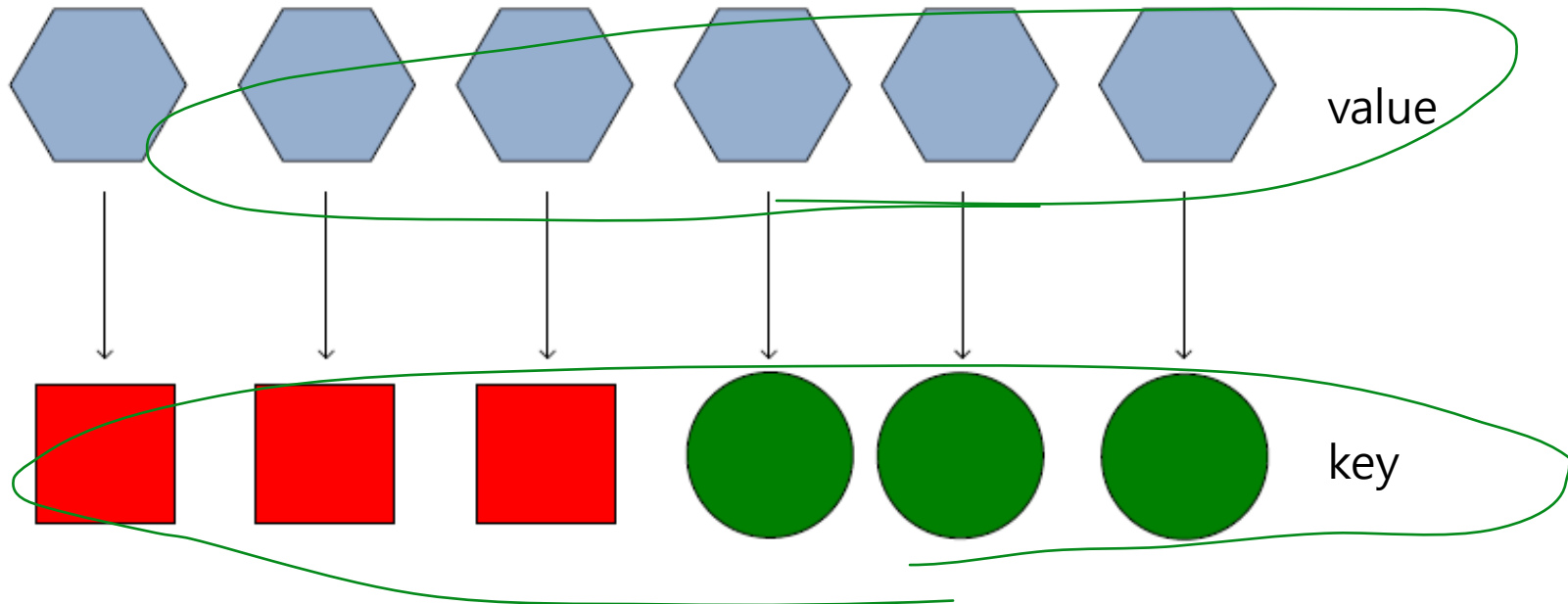


1) map

- 데이터 소스로부터 레코드(파일의 라인이나 DB의 Row 등)들을 읽어서 (key, value) 쌍으로 map 함수에게 보냄
- 맵 함수는 입력 레코드를 받아서 하나 이상의 (Key, Value) 형식의 중간데이터를 만들어내서 로컬 파일시스템에 저장(local temp file)

map

```
map (in_key, in_value) ->  
    (out_key, intermediate_value) list
```

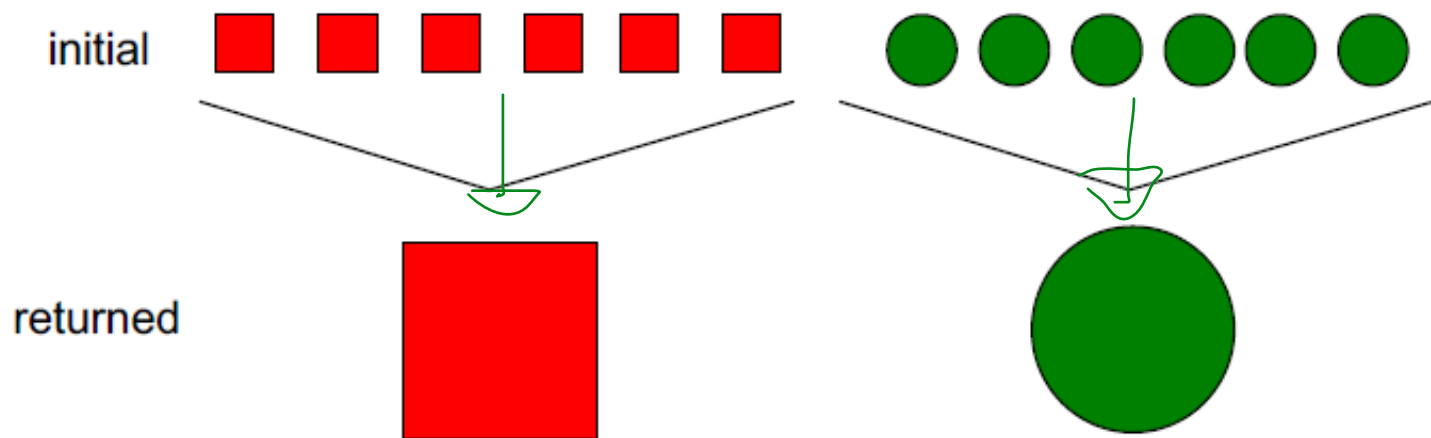


2) reduce

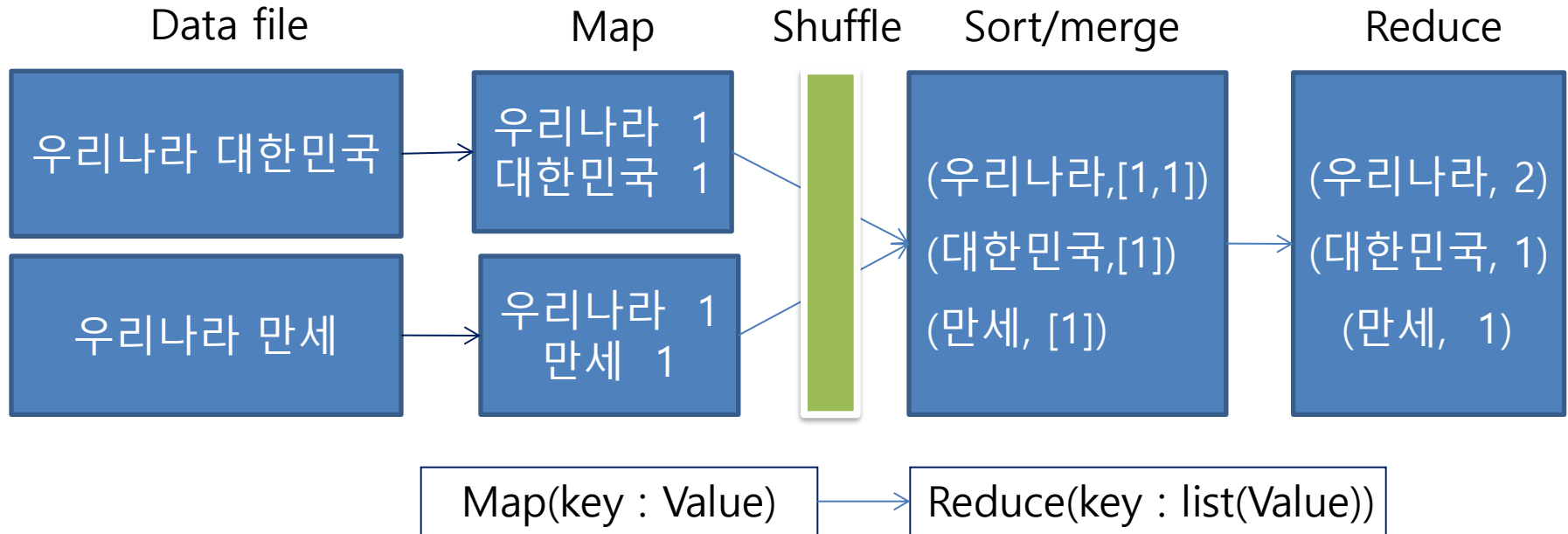
- map 단계가 끝난 후에는 동일한 output key 를 가진 모든 중간 값들은 리스트로 결합됨
- reduce 함수에는 전달된 output key와 중간값들의 리스트를 가지고 최종 결과 값으로 축소

reduce

`reduce (out_key, intermediate_value list) ->`
`out_value list`

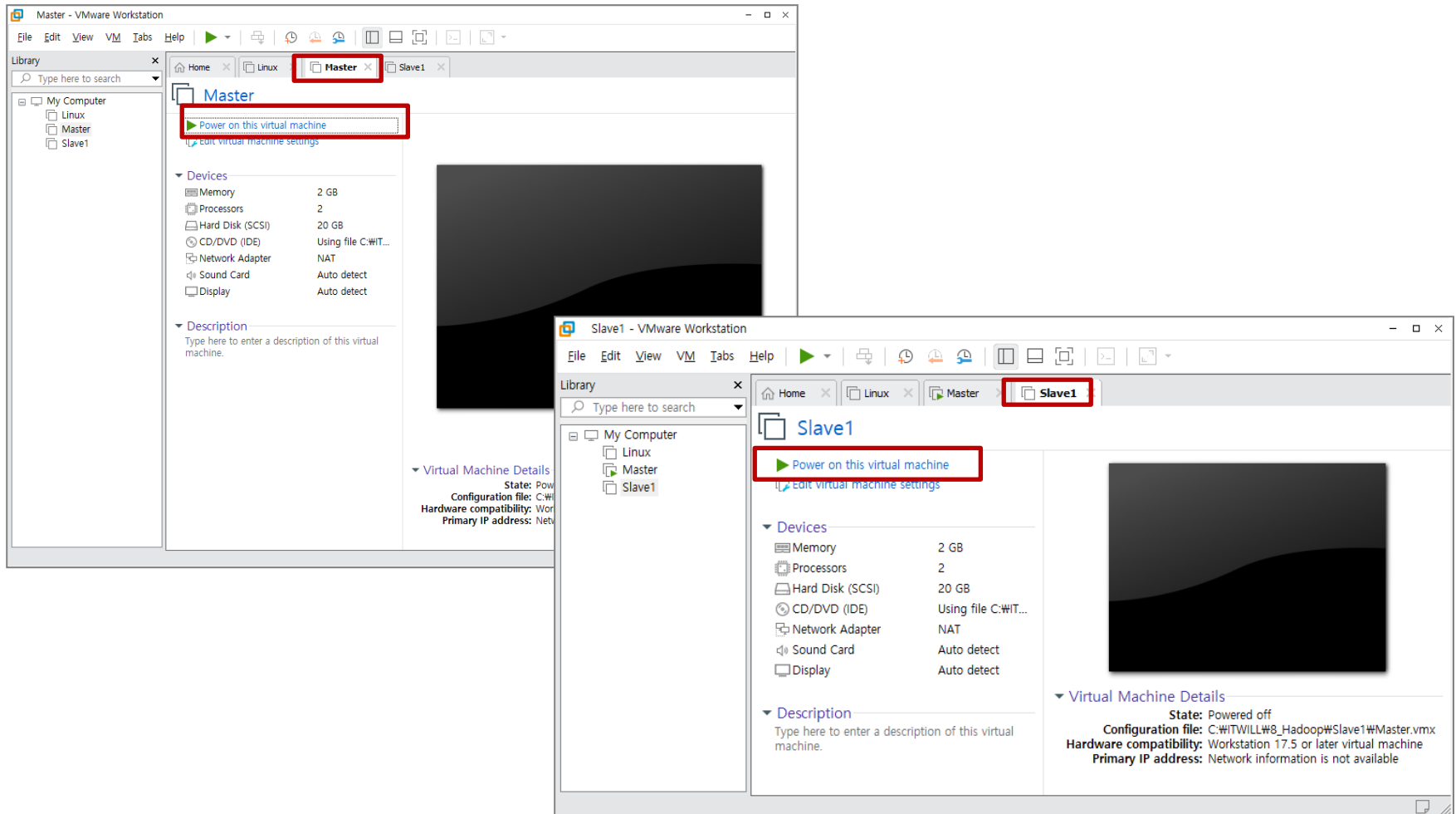


Word Counter 애플리케이션 예



- ✓ Map : 입력 파일 한 줄 읽기 → 데이터 변형
- ✓ Shuffle : Map의 중간 데이터를 Reduce 단계로 전달(파티셔닝, 병합, 정렬)
- ✓ Sort/merge : Shuffle 단계의 결과
- ✓ Reduce : Sort/merge의 집계 결과

2. Master/Slave1 서버 ON



3. Hadoop/Yarn/Historyserver 시작

✓ 맵리듀스, 하이브, 스파크 등의 애플리케이션은 양(YARN)에서 실행된다.

1) Hadoop/Yarn/Historyserver 시작

[hadoop@master ~]\$ start-all.sh # 하둡/양 시작

[hadoop@master ~]\$ mr-jobhistory-daemon.sh start historyserver # 데몬 실행

start-all.sh는 Hadoop 클러스터에서 모든 핵심 데몬(서비스)을 한꺼번에 시작해주는 스크립트 파일
-데몬 서비스: HDFS데몬, YARN데몬

데이터는 HDFS에 저장되어 있고, 그 위에서 작업을 처리하는 것은 YARN

Hadoop에서 MapReduce Job의 기록(히스토리)을 볼 수 있게 해주는 데몬, 즉 HistoryServer를 실행하는 명령어

MapReduce작업을 위해 필요

굳이 안켜도 되지만 문제를 추적하고 진단하고 싶으면 켜두는게 좋음

2) 각 서버 데몬 실행 상태 확인

The image shows two terminal windows. The top window is titled 'hadoop@master:~' and has tabs for 'Home', 'Linux', 'Master', and 'Slave1'. The 'Master' tab is selected and highlighted with a red box. The terminal shows the command `jps` being executed, with the output listing four processes: `3232 SecondaryNameNode`, `4534 Jps`, `2983 NameNode`, and `3464 ResourceManager`. A blue bracket groups these four lines. A red box highlights the `jps` command, and a green arrow points from it to the text '현재 실행 중인 Java 프로세스의 PID(프로세스 ID)와 프로세스 이름(class 이름)을 출력'.

The bottom window is titled 'hadoop@slave1:~' and has tabs for 'Home', 'Linux', 'Master', and 'Slave1'. The 'Slave1' tab is selected and highlighted with a red box. The terminal shows the command `jps` being executed, with the output listing three processes: `3881 Jps`, `2698 NodeManager`, and `2570 DataNode`. A blue bracket groups these three lines.

```
hadoop@master:~$ jps
3232 SecondaryNameNode
4534 Jps
2983 NameNode
3464 ResourceManager
hadoop@master:~$
```

```
hadoop@slave1:~$ jps
3881 Jps
2698 NodeManager
2570 DataNode
hadoop@slave1:~$
hadoop@slave1:~$
```

4. HDFS 명령어

명령어 형식) \$ hdfs dfs -명령어 <인수>

명령어	기능
hdfs dfs -cat	HDFS의 특정 파일 내용 보기
hdfs dfs -put	로컬 시스템의 파일을 HDFS에 업로드
hdfs dfs -get	HDFS 파일을 로컬 시스템으로 다운로드
hdfs dfs -cp	HDFS 파일을 목적지로 복사
hdfs dfs -ls	파일과 디렉터리를 조회한다.
hdfs dfs -mkdir /test	디렉터리 생성 하둡의 분산 파일 시스템(HDFS) 내부에 /test라는 디렉토리를 만드는 명령
hdfs dfs -rmdir	디렉터리 삭제
hdfs dfs -rm -R	디렉터리+파일 동시 삭제
hdfs dfs -rm	파일 삭제
hdfs dfs -count	파일/디렉터리 이름, 파일 수, 디렉터리 수 출력
hdfs dfs -chmod	파일과 디렉터리에 대한 접근 권한 변경

👉 그냥 cat: 로컬 컴퓨터(리눅스 시스템)에 있는 파일을 읽는 명령

🌿 hdfs dfs -cat: HDFS(분산 파일 시스템)에 저장된 파일을 읽는 명령

👉 하둡 클러스터 내부에 저장된 파일이라, 일반 cat으로는 접근 불가

5. MapReduce Word Count 실습

1) Word Count 파일 준비

```
[hadoop@master ~]$  
[hadoop@master ~]$  
[hadoop@master ~]$  
[hadoop@master ~]$ hdfs dfs -ls /  
Found 2 items  
drwxr-xr-x - hadoop supergroup 0 2024-07-04 16:05 /test  
drwxrwx--- - hadoop supergroup 0 2024-07-04 17:34 /tmp  
[hadoop@master ~]$  
[hadoop@master ~]$ hdfs dfs -put ./hadoop-3.3.6/NOTICE.txt /test  
[hadoop@master ~]$  
[hadoop@master ~]$ hdfs dfs -cat /test/NOTICE.txt  
Apache Hadoop  
Copyright 2006 and onwards The Apache Software Foundation  
This product includes software developed at The Apache Software Foundation (http://www.apache.org/)
```

로컬에 있는 NOTICE.txt 파일을 HDFS의 /test 디렉토리에 업로드하는 명령어

HDFS안에 있는 /test/NOTICE.txt보기

2) Word Count 실행

순서대로 하둡에서 .jar파일 실행하는 명령어/ mapreduce프로그램 들어간 jar파일
/ 실행할 클래스이름/ 입력파일경로/ 출력디렉토리경로

```
hadoop@master:~  
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)  
drwxr-xr-x - hadoop supergroup 0 2021-12-30 14:42 /test  
drwxrwx--- - hadoop supergroup 0 2021-12-30 13:57 /tmp  
[hadoop@master ~]$ hdfs dfs -ls /test  
Found 1 items  
-rw-r--r-- 3 hadoop su  
[hadoop@master ~]$  
[hadoop@master ~]$  
[hadoop@master ~]$ hadoop jar $HADOOP_HOME/share/hadoop/mapreduce/hadoop-  
mapreduce-examples-3.3.6.jar wordcount /test/NOTICE.txt /output
```

hadoop jar /디렉터리/*.jar 파라미터 대상파일 출력디렉터리

Map & Reduce 작업 상태

```
21/12/30 15:06:07 INFO mapreduce.Job: The url to track the job is http://10.10.10.10:8080/jobreport/wordcount_1640843773438_0001/  
21/12/30 15:06:07 INFO mapreduce.Job: Running job: job_1640843773438_0001  
21/12/30 15:06:18 INFO mapreduce.Job: Job job_1640843773438_0001 running in uber mode : false  
21/12/30 15:06:18 INFO mapreduce.Job: map 0% reduce 0%  
21/12/30 15:06:25 INFO mapreduce.Job: map 100% reduce 0%  
21/12/30 15:06:31 INFO mapreduce.Job: map 100% reduce 100%  
21/12/30 15:06:33 INFO mapreduce.Job: Job job_1640843773438_0001 completed successfully  
21/12/30 15:06:33 INFO mapreduce.Job: Counters: 49  
File System Counters  
FILE: Number of bytes read=12054  
FILE: Number of bytes written=441367  
FILE: Number of read operations=0  
FILE: Number of large read operations=0  
FILE: Number of write operations=0
```

hadoop@master:~/hadoop-2.7.1

파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)

```
Combine output records=11
Reduce input groups=11
Reduce shuffle bytes=173
Reduce input records=11
Reduce output records=11
Spilled Records=22
Shuffled Maps =1
Failed Shuffles=0
Merged Map outputs=1
GC time elapsed (ms)=175
CPU time spent (ms)=1150
Physical memory (bytes) snapshot=277086208
Virtual memory (bytes) snapshot=4200316928
Total committed heap usage (bytes)=137498624

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=101
File Output Format Counters
  Bytes Written=123
```

[hadoop@master hadoop-2.7.1]\$

3) Word Count 결과보기

```
hadoop@master ~]$  
[hadoop@master ~]$ hdfs dfs -ls /output  
Found 2 items  
-rw-r--r-- 3 hadoop supergroup 0 2021-12-30 15:06 /output/_SUCCESS  
-rw-r--r-- 3 hadoop supergroup 9456 2021-12-30 15:06 /output/part-r-00000  
[hadoop@master ~]$  
[hadoop@master ~]$ hdfs dfs -cat /output/part-r-00000  
AS 1  
"GCC 1  
"License"); 1  
& 1  
'Aalto 1  
'Apache 4  
'ArrayDeque', 1  
'Bouncy 1  
'Caliper', 1  
'Compress-LZF', 1
```

워드 카운터 결과 디렉터리

아까 만든 출력 경로

워드 카운터 실행 결과 파일

❖ HDFS에서 로컬 파일 시스템으로 파일 복사

```
hdfs dfs -get /output/part-r-00000 ~/hfile/word_count.txt
```

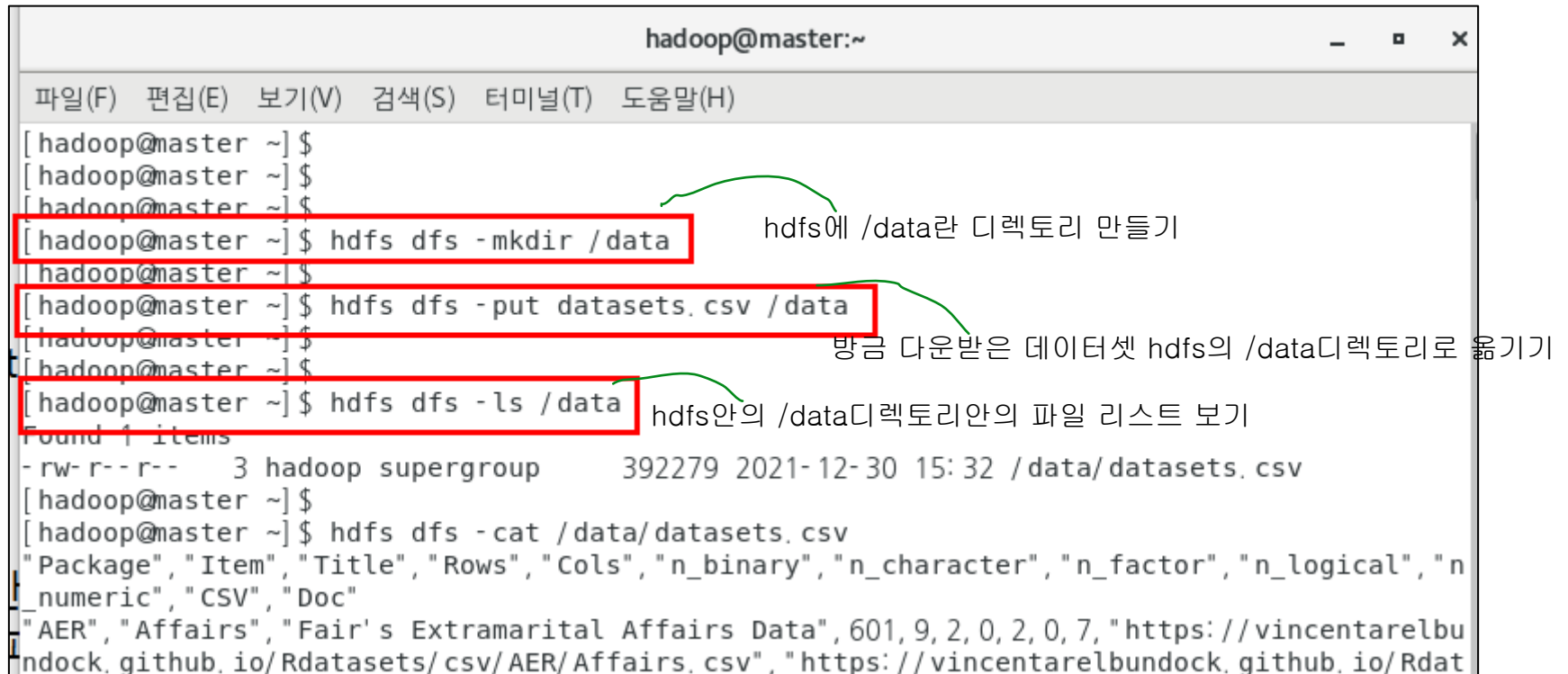
6. HDFS 명령어 실습

- csv 파일 다운로드 & HDFS에 업로드

wget은 웹에서 파일을 다운로드하는 명령어

```
hadoop@master ~$  
[hadoop@master ~]$  
[hadoop@master ~]$ wget http://vincentarelbundock.github.io/Rdatasets/datasets.csv  
--2021-12-30 15:31:09-- http://vincentarelbundock.github.io/Rdatasets/datasets.csv  
Resolving vincentarelbundock.github.io (vincentarelbundock.github.io)... 185.199.109.153, 185.199.111.153, 185.199.110.153, ...  
Connecting to vincentarelbundock.github.io (vincentarelbundock.github.io)|185.199.109.153|:80... connected.  
HTTP request sent, awaiting response... 301 Moved Permanently  
Location: https://vincentarelbundock.github.io/Rdatasets/datasets.csv [following]  
--2021-12-30 15:31:09-- https://vincentarelbundock.github.io/Rdatasets/datasets.csv  
Connecting to vincentarelbundock.github.io (vincentarelbundock.github.io)|185.199.109.153|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 392279 (383K) [text/csv]  
Saving to: 'datasets.csv'  
  
100%===== 3.02 MB/s in 0.1s  
2021-12-30 15:31:09 (3.02 MB/s) 'datasets.csv' saved [392279/392279]  
  
[hadoop@master ~]$ ls  
datasets.csv  hadoop-2.10.1.tar.gz  다운로드  바탕화면  사진  음악  
hadoop-2.10.1  공개  문서  비디오  서식  
[hadoop@master ~]$
```

- csv 파일을 HDFS에 업로드하기



```
hadoop@master:~  
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)  
[hadoop@master ~]$  
[hadoop@master ~]$  
[hadoop@master ~]$  
[hadoop@master ~]$ hdfs dfs -mkdir /data  
[hadoop@master ~]$  
[hadoop@master ~]$ hdfs dfs -put datasets.csv /data  
[hadoop@master ~]$  
[hadoop@master ~]$  
[hadoop@master ~]$ hdfs dfs -ls /data  
Found 1 items  
-rw-r--r-- 3 hadoop supergroup 392279 2021-12-30 15:32 /data/datasets.csv  
[hadoop@master ~]$  
[hadoop@master ~]$ hdfs dfs -cat /data/datasets.csv  
"Package", "Item", "Title", "Rows", "Cols", "n_binary", "n_character", "n_factor", "n_logical", "n  
_numeric", "CSV", "Doc"  
"AER", "Affairs", "Fair's Extramarital Affairs Data", 601, 9, 2, 0, 2, 0, 7, "https://vincentarelbund  
dock.github.io/Rdatasets/csv/AER/Affairs.csv", "https://vincentarelbundock.github.io/Rdat
```

hdfs에 /data란 디렉토리 만들기

방금 다운받은 데이터셋 hdfs의 /data디렉토리로 옮기기

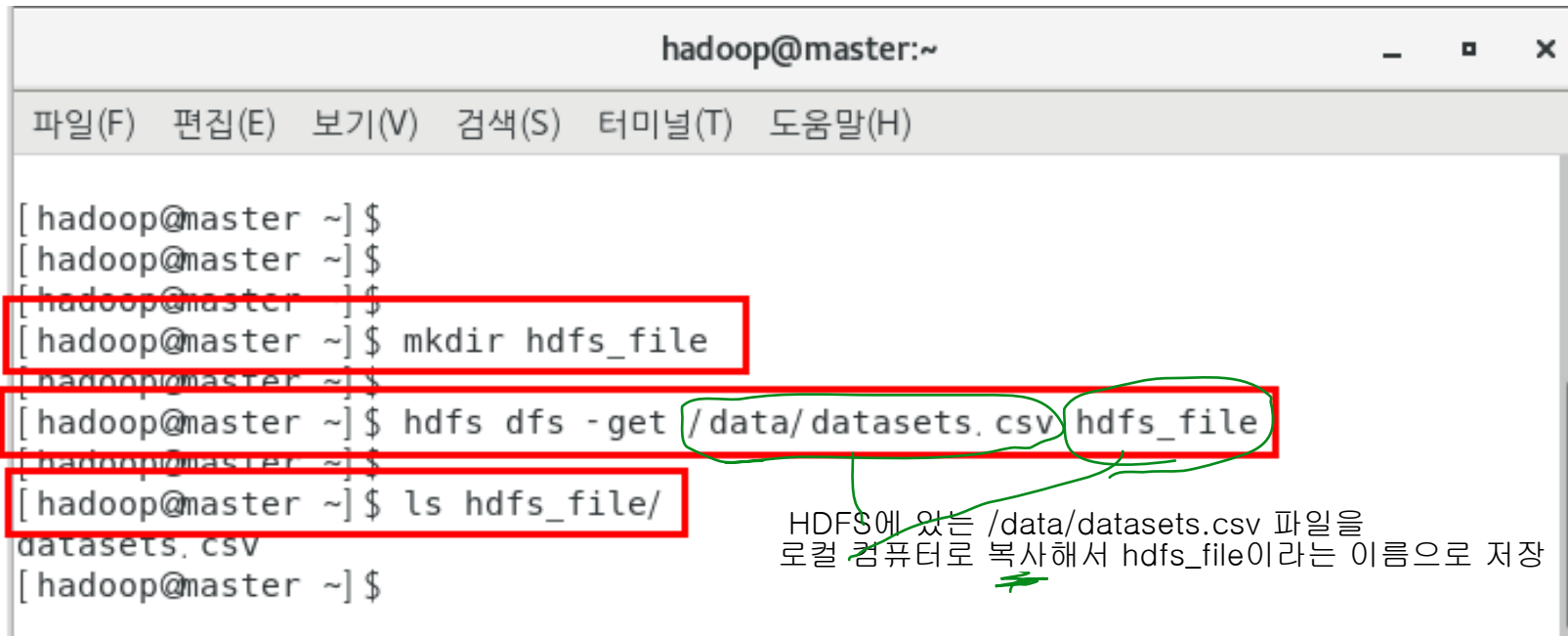
hdfs안의 /data디렉토리안의 파일 리스트 보기

- HDFS에서 파일 복제

```
hadoop@master:~  
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)  
[hadoop@master ~]$  
[hadoop@master ~]$  
[hadoop@master ~]$ copy hdfs dfs -cp /data/datasets.csv /data/datasets2.csv  
[hadoop@master ~]$  
[hadoop@master ~]$  
[hadoop@master ~]$ hdfs dfs -ls /data  
Found 2 items  
-rw-r--r-- 3 hadoop supergroup 392279 2021-12-30 15:32 /data/datasets.csv  
-rw-r--r-- 3 hadoop supergroup 392279 2021-12-30 15:35 /data/datasets2.csv  
[hadoop@master ~]$
```

파일 복사

- HDFS에서 로컬에 파일 다운로드

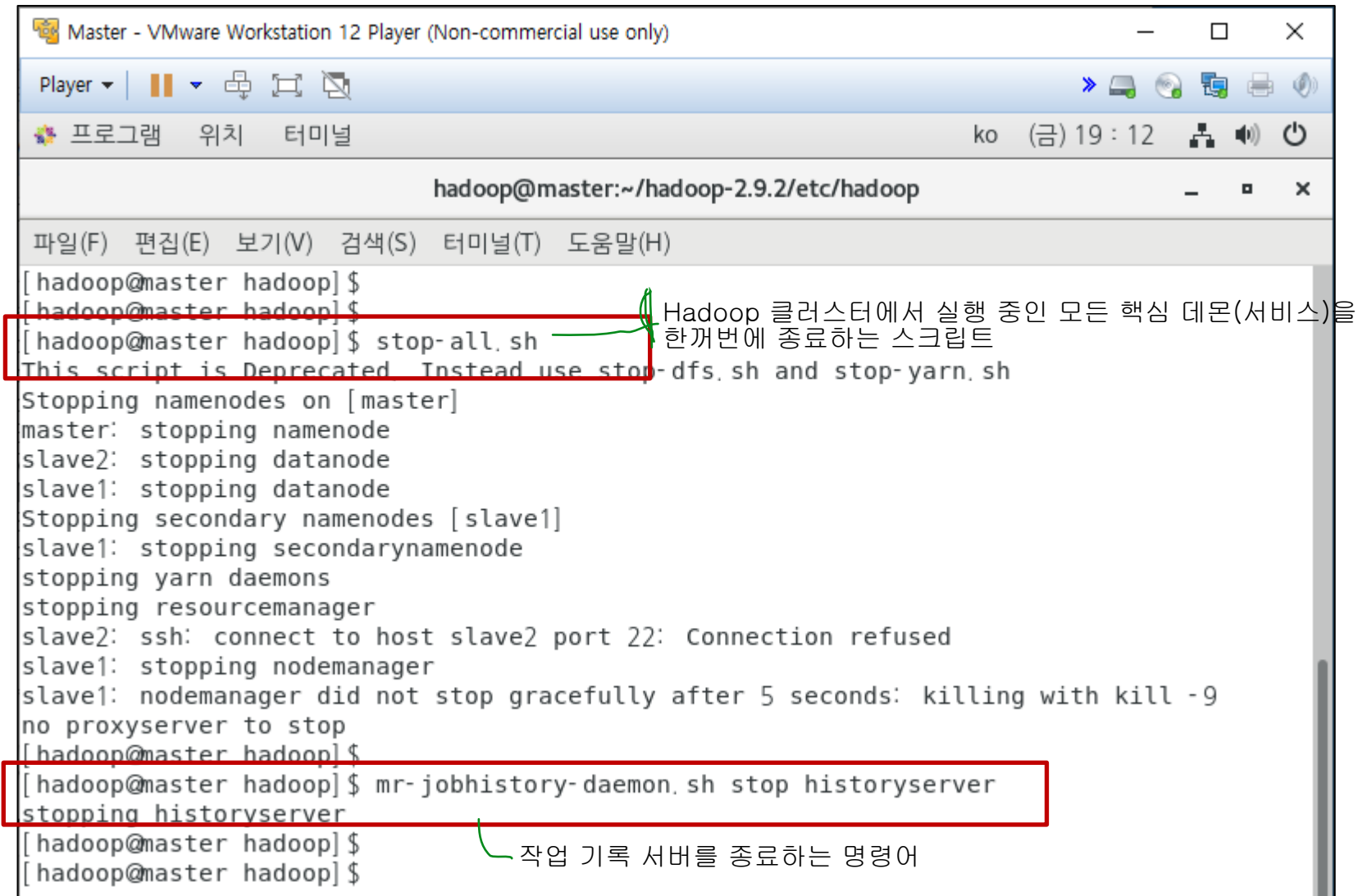


A terminal window titled 'hadoop@master:~' with a menu bar containing '파일(F)', '편집(E)', '보기(V)', '검색(S)', '터미널(T)', and '도움말(H)'. The terminal shows the following commands and their outputs:

```
[hadoop@master ~]$  
[hadoop@master ~]$  
[hadoop@master ~]$  
[hadoop@master ~]$ mkdir hdfs_file  
[hadoop@master ~]$  
[hadoop@master ~]$ hdfs dfs -get /data/datasets.csv hdfs_file  
[hadoop@master ~]$  
[hadoop@master ~]$ ls hdfs_file/  
datasets.csv  
[hadoop@master ~]$
```

Annotations in the image include red boxes around the commands `mkdir hdfs_file`, `hdfs dfs -get /data/datasets.csv hdfs_file`, and `ls hdfs_file/`. A green circle highlights the file path `/data/datasets.csv` in the `hdfs dfs -get` command. A green arrow points from this circle to a Korean text annotation: 'HDFS에 있는 /data/datasets.csv 파일을 로컬 컴퓨터로 복사해서 hdfs_file이라는 이름으로 저장'.

7. Hadoop/Yarn/Historyserver 종료



The screenshot shows a terminal window titled "Master - VMware Workstation 12 Player (Non-commercial use only)". The terminal prompt is `hadoop@master:~/hadoop-2.9.2/etc/hadoop`. The terminal output shows the execution of `stop-all.sh` and `mr-jobhistory-daemon.sh stop historyserver`. A red box highlights the `stop-all.sh` command and its output, and another red box highlights the `mr-jobhistory-daemon.sh stop historyserver` command. A green arrow points from the text "Hadoop 클러스터에서 실행 중인 모든 핵심 데몬(서비스)을 한꺼번에 종료하는 스크립트" to the `stop-all.sh` command. Another green arrow points from the text "작업 기록 서버를 종료하는 명령어" to the `mr-jobhistory-daemon.sh stop historyserver` command.

```
[hadoop@master hadoop]$  
[hadoop@master hadoop]$  
[hadoop@master hadoop]$ stop-all.sh  
This script is Deprecated. Instead use stop-dfs.sh and stop-yarn.sh  
Stopping namenodes on [master]  
master: stopping namenode  
slave2: stopping datanode  
slave1: stopping datanode  
Stopping secondary namenodes [slave1]  
slave1: stopping secondarynamenode  
stopping yarn daemons  
stopping resourcemanager  
slave2: ssh: connect to host slave2 port 22: Connection refused  
slave1: stopping nodemanager  
slave1: nodemanager did not stop gracefully after 5 seconds: killing with kill -9  
no proxyserver to stop  
[hadoop@master hadoop]$  
[hadoop@master hadoop]$ mr-jobhistory-daemon.sh stop historyserver  
stopping historyserver  
[hadoop@master hadoop]$  
[hadoop@master hadoop]$
```

Hadoop 클러스터에서 실행 중인 모든 핵심 데몬(서비스)을 한꺼번에 종료하는 스크립트

작업 기록 서버를 종료하는 명령어