3장

목표

1. 워드클라우드를 실행해보자

개략도

1. UNIX(wordcount-data.txt) 의 데이터를 hdfs(하둡 분산 파일 시스템)의 wordcount_test/로 이동
2. mapreduce를 실행해서 처리함!
3. 처리결과가 wordcount_test_out에 결과가 출력됨!

dirver.java

- vi Driver.java로 에디터를 킬 수 있음

```
pgd.addClass("wordcount", Wordcount.class, "A map/reduce program that performs
word counting.");
```

- compile을 위해서 반드시 필요한 코드, 모든 새로운 코드마다 필요
- 가장 앞에 "wordcount" 는 이 ""안의 java파일을 통해서 실행하겠다고 하는 것
- 수정 후 반드시 ant를 다시 수행해야만 함

ant

- java 표준 빌드 도구로 여러 dependency를 고려하여 소스파일을 컴파일한다.
- src 디렉토리를 다 모아서 컴파일 하여 ssafy.jar를 생성
- build.xml 파일에 정의한대로 수행됨

HDFS 실행하기

```
hdfs dfs -mkdir wordcount_test
```

- 해당 디렉토리를 hdfs에 생성

```
hdfs dfs -put data/wordcount-data.txt wordcount_test
```

- 리눅스의 데이터 디렉토리 내부 txt파일을 hdfs의 wordcount디렉토리로 이동

```
hdfs dfs -rm -r wordcount_test_out
```

- 같은 파일이 존재할 시, 에러가 나므로 반드시 사용전 지울 것!!

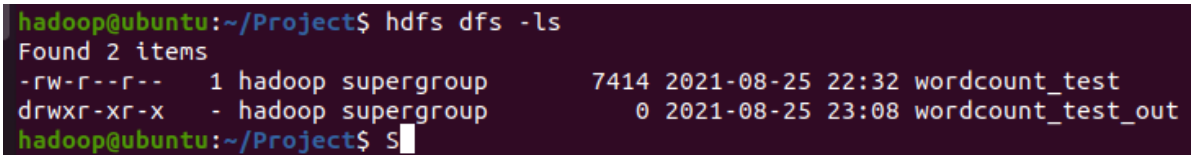
```
hadoop jar ssafy.jar wordcount wordcount_test wordcount_test_out
(1)hadoop jar (2)ssafy.jar (3) wordcount (4)wordcount_test (5)wordcount_test_out
```

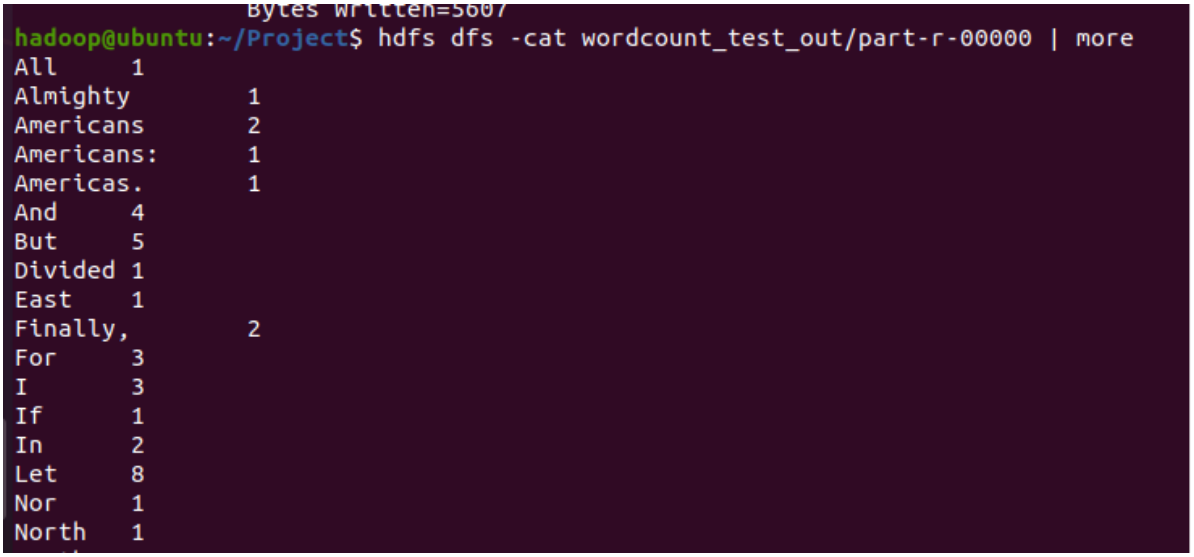
- (1). 하둡에서 jar 이라는 코드묶음 덩어리를 실행한다
- (2). 이때 이름은 ssafy.jar인 파일이다
- (3). wordcount 코드를 이용하여 실행한다
- (4). input datas는 (4)번에서 가져온다
- (5). outdata는 (5)번에 기록한다

```
hdfs dfs -cat wordcount_test_out/part-r-000000|more
```

- cat: 화면에 특정데이터를 찍는 명령어
- part-r-000000 에서 개수만큼 n개가 찍힌다
- |more : 화면크기만큼만 표시하고 더 볼지 선택권을 줌

참고그림





기본입출력 디폴트

- 이 툴들을 이용해서 자바에 입력 및 출력이 가능하다
- 자바 잘하면 개인적으로 만들어도 된다