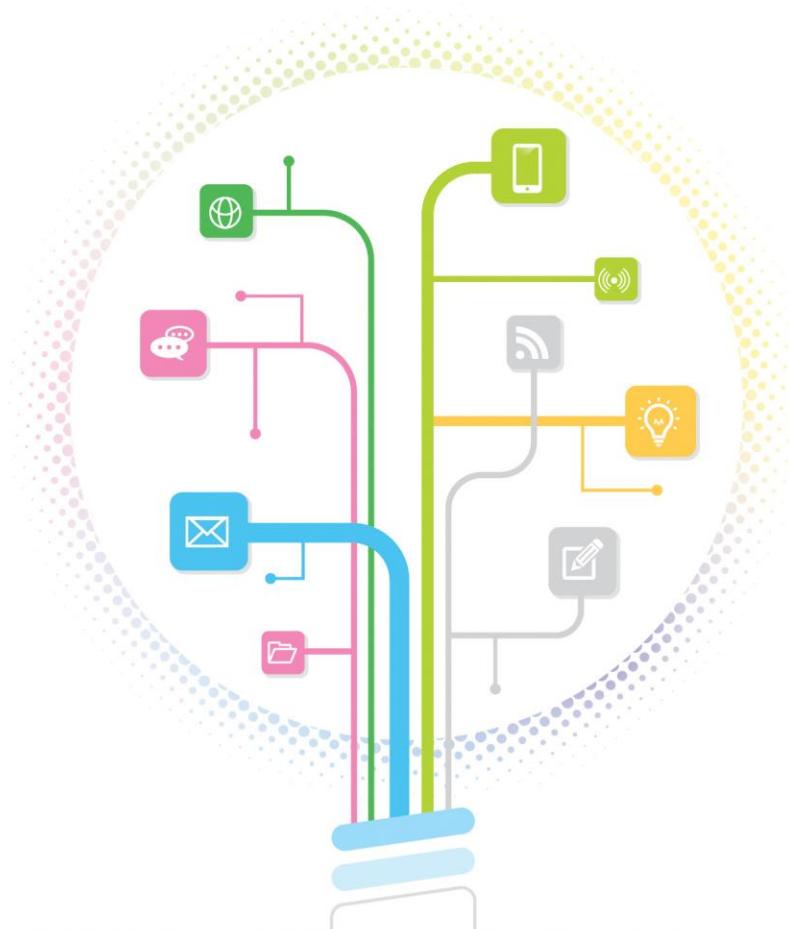


# [ 6 유통 데이터 분석 콘텐츠 활용 매뉴얼 ]



미래창조과학부



한국정보화진흥원



Korea Big Data Center  
전략센터

# CONTENTS

Beginning Level 초급과정

## I 개요

개요	9
----	---

## II 수집

개요	13
교육용 데이터 샘플	14
데이터 수집	15
데이터 작업 영역 이동 스크립트	18

## III 가공

개요	23
데이터 가공 과정	24

## IV 저장

개요	27
가공 데이터 저장	28



## V 분석

---

개요	33
데이터 분석 과정	34

## VI 시각화

---

개요	41
시각화 과정	42
시각화 데이터 분석	45

## VII 예제 문제

---

예제 문제1. 온라인 쇼핑몰의 연도별 월간 매출 변동을 분석하라.	49
예제 문제2. 분기별 최고 공급량 아이템을 파악하라.	50

# CONTENTS

Intermediate Level 

## I 개요

개요	55
----	----

## II 수집

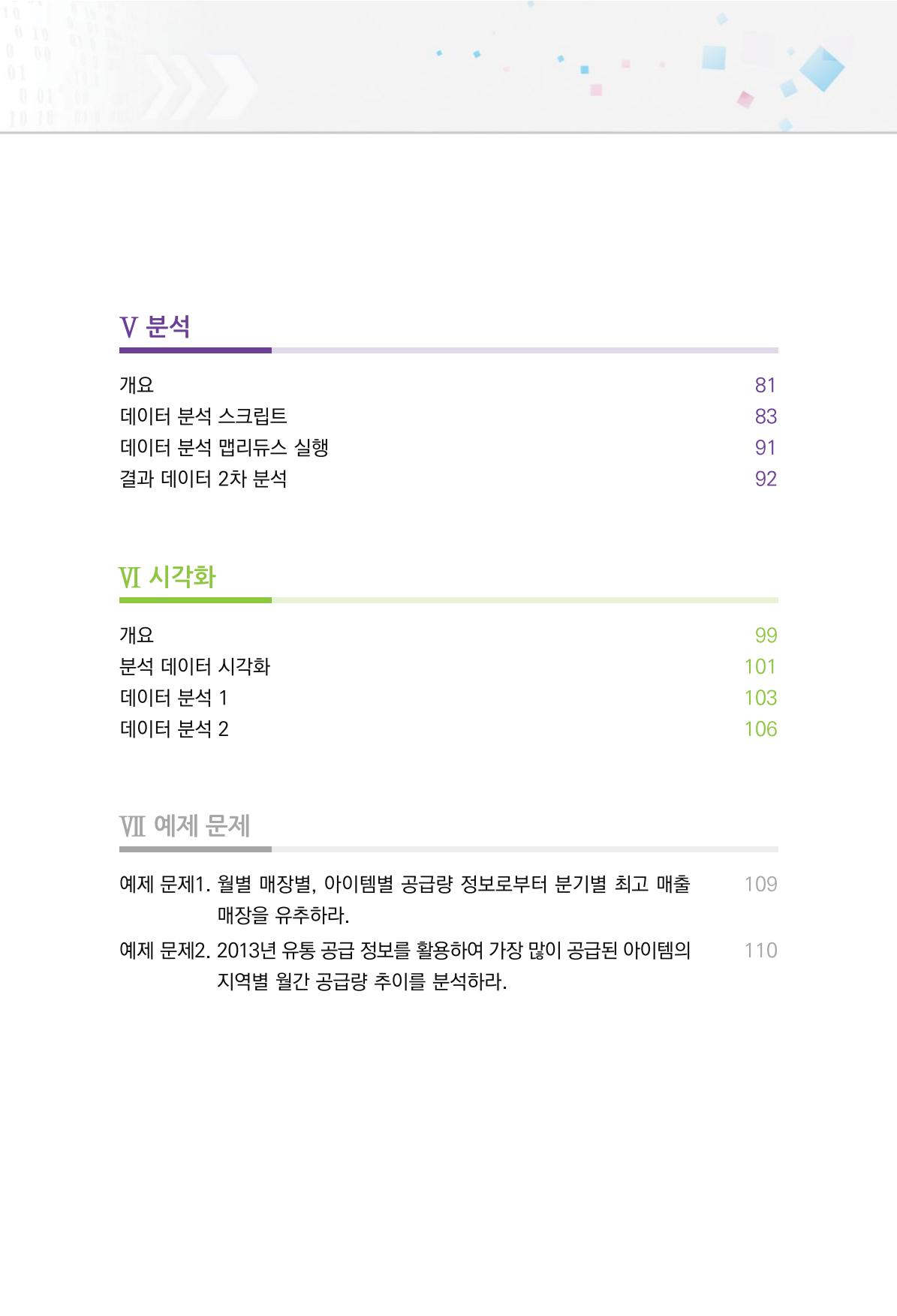
개요	59
교육용 데이터 샘플	60
데이터 수집	62
데이터 작업 영역 이동 스크립트	66

## III 가공

개요	71
데이터 가공 스크립트	72

## IV 저장

개요	75
가공 데이터 하둡 파일시스템 업로드	76



## V 분석

---

개요	81
데이터 분석 스크립트	83
데이터 분석 맵리듀스 실행	91
결과 데이터 2차 분석	92

## VI 시각화

---

개요	99
분석 데이터 시각화	101
데이터 분석 1	103
데이터 분석 2	106

## VII 예제 문제

---

예제 문제1. 월별 매장별, 아이템별 공급량 정보로부터 분기별 최고 매출 매장을 유추하라.	109
예제 문제2. 2013년 유통 공급 정보를 활용하여 가장 많이 공급된 아이템의 지역별 월간 공급량 추이를 분석하라.	110



유통 

Beginning Level

초급과정







## I 개요

개요

9

8

# I

## 개요

### > 개요

대한상공회의소에서 제공받은 2013년 서울 주요 소매점(대형 마트, 편의점, 슈퍼 등) 유통 공급 정보 데이터를 바탕으로 오픈오피스 스프레드 시트의 피벗테이블 기능을 활용한 매장별 유통 공급량에 대한 통계 분석 과정을 통해 원하는 항목별로 통계를 산출하여 분석하는 방법을 학습한다. 이러한 방법으로 소매점 별(대형 마트)의 유통 공급량을 보고 매장 규모 및 해당 지역의 소비수준 및 인구 등을 유추할 수 있는 기초 자료가 될 수 있다.

### > 활용 데이터

- **pds\_seoul\_201301.csv :**  
2013년 1월 서울 주요 소매점 유통 공급 정보

### > 선행학습

- **리눅스** – 파일시스템 구조, 쉘 명령어, 쉘 스크립트 실행 방법
- **오픈오피스** – 피벗테이블 기능, 차트 사용 방법

## > 요구사항

- 서울 지역 2013년 1월 소매점 유통 공급 정보로부터 주요 대형 마트들의 2013년 1월 매출 합계를 추출하고, 스프레드 시트 차트로 시각화하라.

## > 분석 절차

- 2013년 1월 소매점 데이터를 서버로부터 수집한다.
- 수집한 데이터를 오픈오피스 스프레드시트에서 로드한다.
- 스프레드시트의 피벗 테이블 기능을 이용하여, 매장별, 품목별로 제공된 유통 공급 금액에 대하여 대형마트들의 총 공급금액 통계를 계산한다.
- 서울 주요 소매점 별(대형 마트) 2013년 1월 매출 분석과 유통량 분석한다.
- 취합한 2013년 1월 서울 지역 대형 마트 별 공급금액 통계 데이터를 시계열 분석을 위한 막대그래프를 활용하여 시각화해 본다.
- 소매점 별(대형 마트)의 유통 공급량을 보고 매장 규모 및 해당 지역의 소비 수준 및 인구 등을 유추할 수 있는 단순 예측 방법의 적합성을 검증해 본다.



1

2

## II 수집

개요	13
교육용 데이터 샘플	14
데이터 수집	15
데이터 작업 영역 이동 스크립트	18



## 수집

### ▶ 개요

유통 데이터는 대한상공회의소에서 제공받은 2013년 서울 주요 소매점 (대형 마트, 편의점, 슈퍼 등) 유통 공급 정보 데이터를 수집하여 분석 목적을 달성할 수 있는 한도 내에서 개인 정보 비식별화 및 특정 상품 비식별화 처리를 통해 분석에 용이하게 편집하여 제공한다.

### ▶ 수집 방법

- **데이터 제공** : 유통 공급 정보 데이터는 대한상공회의소에서 제공하는 데이터를 OpenAPI, 자료수집기(Crawler)를 통하여 데이터를 수집하였고, 실습용 자료는 빅데이터 분석 활용센터에 접속하여 유통 데이터 셋을 다운로드할 수 있도록 원시데이터를 제공하고 있다.



용 어 정 리

- **비식별화** : 데이터 값 삭제, 가명처리, 총계처리, 범주화, 데이터 마스킹 등을 통해 개인정보의 일부 또는 전부를 삭제하거나 대체함으로써 다른 정보와 쉽게 결합하여도 특정 개인을 식별할 수 없도록 하는 조치를 말한다.

- \*출처: 방송통신위원회, “빅데이터 개인정보보호 가이드라인”, 작성일 2014.12.23

## ▶ 교육용 데이터 샘플

### ▶ 2013년 서울 주요 매장 물품 공급 정보(pds\_seoul\_201301.csv)

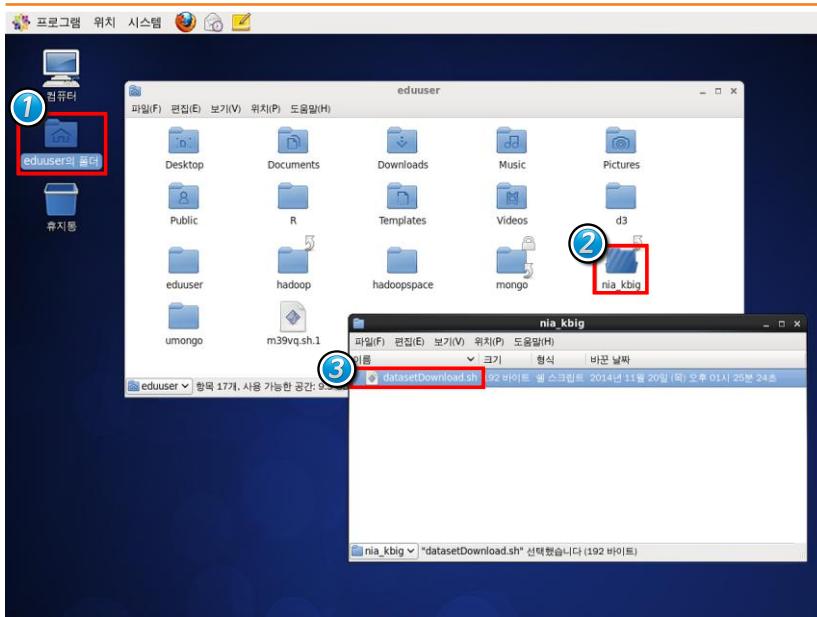
매장	지역	매장형태	품목	공급금액
2030BC	서울	대형마트	930890	27720
2030BC	서울	대형마트	3435418	114150
2030BC	서울	대형마트	940810	176300
2030BC	서울	대형마트	3456118	516750
2030BC	서울	대형마트	10557093	267020
2030BC	서울	대형마트	50390427	21600
2030BC	서울	대형마트	50430048	5400
2030BC	서울	대형마트	80050278	1550
2030BC	서울	대형마트	88002026	18960
2030BC	서울	대형마트	88003467	2500

## II. 수집

### > 데이터 수집(datasetDownload.sh)

- 데이터 저장소에서 서버 로컬로 유통 데이터 셋을 복사해 온다.
  - `pds_seoul_201301.csv` : 2013년 서울 주요 매장 물품 공급 데이터

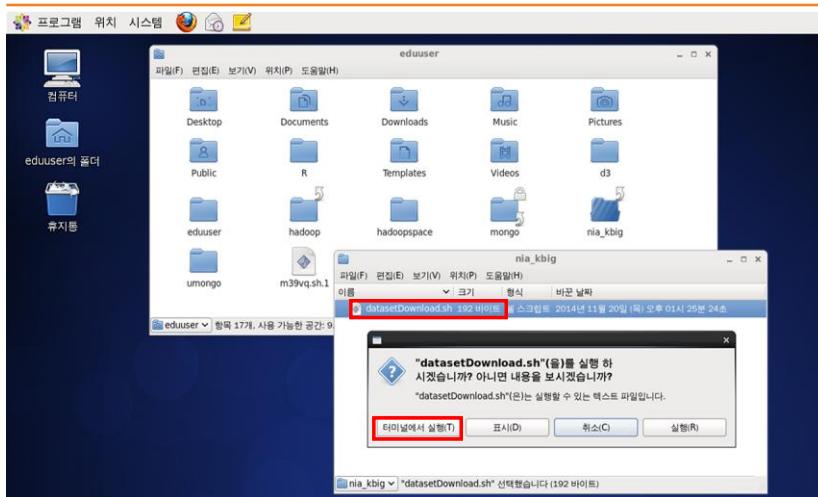
### > 실습코드 디렉토리로 이동



- ① 로그인 후 바탕화면에서 eduuser 폴더를 오픈한다.
- ② nia\_kbig 폴더를 오픈한다.
- ③ datasetDownload.sh를 더블클릭하여 실행한다.

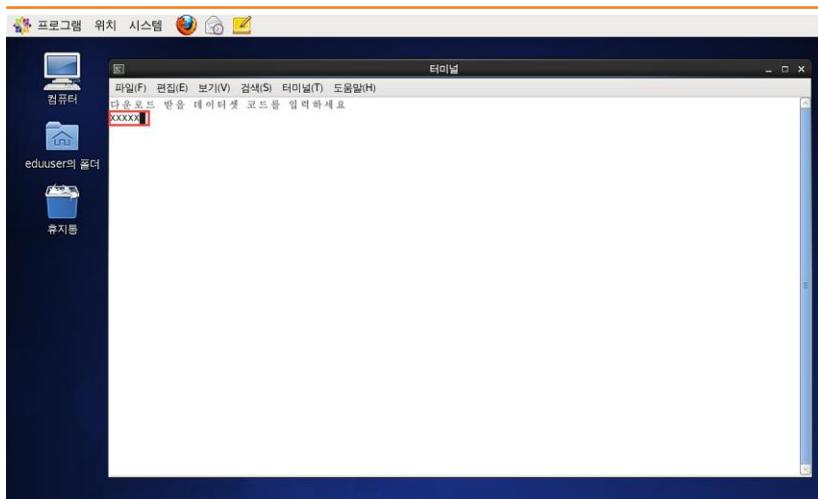
## ▶ 레파지토리에서 데이터 수집

### datasetDownload.sh (원시데이터로 컬서버로 복사)



- '터미널에서 실행' 버튼을 클릭한다.

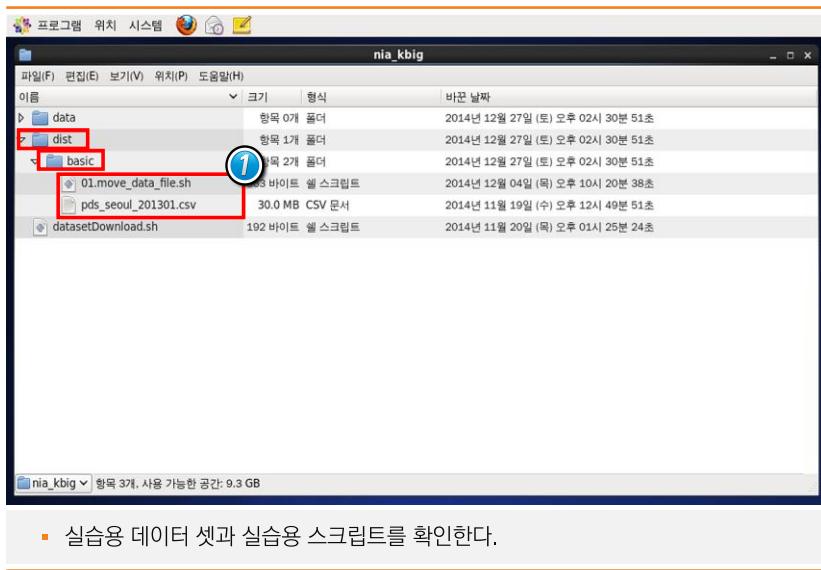
## ▶ 데이터셋 코드 입력



- 다운로드 받을 데이터셋 코드를 입력 후 엔터

## II. 수집

### ▶ 데이터셋과 실습용 쉘 스크립트



### ▶ ① 데이터 및 스크립트

#### ▪ 01.move\_data\_file.sh :

로컬로 수집해온 데이터를 작업 영역 Data 폴더로 자료를 이동하는 스크립트

#### ▪ pds\_seoul\_201301.csv :

2013년 1월 서울 지역 주요 소매점 유통 공급 정보

## > 데이터 작업 영역 이동 스크립트(01.move\_data\_file.sh)

### > 데이터 작업 공간으로 이동

- 로컬로 수집해온 데이터를 작업 영역 Data 폴더로 자료를 이동하는 스크립트

#### 01.move\_data\_file.sh (작업영역 폴더로 원시데이터 이동)

```

01.#!/bin/bash
02. # 2013년 1월 서울지역 유통 공급내역 파일
03. TARGET_DIST_BASIC=/home/eduuser/nia_kbig/dist/basic/pds_seoul_201301.
    ↗ csv
04. # 작업 디렉토리 정의
05. LOCAL_DIR=/home/eduuser/nia_kbig/data/
06. mv $TARGET_DIST_BASIC $LOCAL_DIR
07.

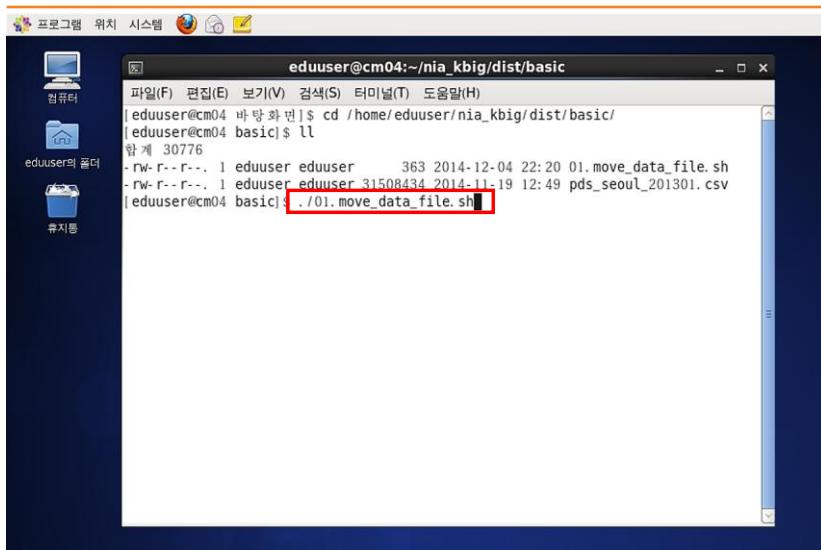
```



- 데이터 작업 영역 이동 스크립트 소스(01.move\_data\_file.sh)
- 라인 03 : 다운로드 받은 원시데이터 파일의 위치(path)를 변수(TARGET\_DIST\_BASIC)로 지정하는 라인이다.
- 라인 05 : 작업영역 디렉토리의 위치(path)를 변수(LOCAL\_DIR)로 지정하는 라인이다.
- 라인 06 : mv 명령어를 사용하여 다운로드 받은 원시데이터 파일들을 작업영역 디렉토리로 이동시키는 라인이다.

## II. 수집

### ▶ 수집 데이터 셋 작업 영역 폴더 이동



- 로컬에 원시데이터를 작업 영역 폴더로 이동(/home/eduuser/nia\_kbig/data/) 시킨다.  
`./01.move_data_file.sh` 입력 후 엔터

I. 개요

II. 수집

III. 가공

IV. 저장

V. 분석

VI. 시각화



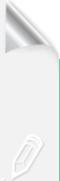




## 가공

### > 개요

작업 영역 폴더에 복사한 유통 데이터는 오픈 오피스 스프레드 시트에서 분석 과정에서 바로 사용할 수 있는 형태로 제공되고 있기 때문에, 별도의 전처리 가공 과정은 생략한다.



### > 가공 방법

- 대한상공회의소 유통 정보 데이터는 2013년 데이터가 저장되어 있다.  
2013년 1월 서울 유통 정보 데이터를 사용한다.
- 오픈 오피스에서 데이터를 읽어 들인다.
- 바로 가공 및 분석이 가능하도록 필요한 정보만 추출되어 있는 데이터이므로 별도의 추가 가공 과정은 생략한다.

### > 데이터셋

매장	지역	매장형태	품목	공급금액
2030BC	서울	대형마트	930890	27720
2030BC	서울	대형마트	3435418	114150
2030BC	서울	대형마트	940810	176300
2030BC	서울	대형마트	3456118	516750
2030BC	서울	대형마트	10557093	267020
2030BC	서울	대형마트	50390427	21600
2030BC	서울	대형마트	50430048	5400
2030BC	서울	대형마트	80050278	1550
2030BC	서울	대형마트	88002026	18960
2030BC	서울	대형마트	88003467	2500

## ▶ 데이터 가공 과정

### ▶ 데이터 로드

- 오픈오피스를 사용하여 연도별로 저장된 원시데이터를 읽어 들여 분석을 위한 기초 가공을 한다.

The screenshot shows a spreadsheet application window titled "pds\_seoul\_201301.csv - OpenOffice Calc". The data is organized into columns:

	A	B	C	D	E	F	G	H	I	J	K
1	대 청	서 울	구 풍	아이 냉	공급 종별						
2	2030BC	서 울	대 청 마트	3417812	78300						
3	2030BC	서 울	대 청 마트	3432112	27720						
4	2030BC	서 울	대 청 마트	3435418	114150						
5	2030BC	서 울	대 청 마트	3455412	176300						
6	2030BC	서 울	대 청 마트	3465118	516750						
7	2030BC	서 울	대 청 마트	50213245	267020						
8	2030BC	서 울	대 청 마트	50390427	21600						
9	2030BC	서 울	대 청 마트	50430044	5400						
10	2030BC	서 울	대 청 마트	80050278	1500						
11	2030BC	서 울	대 청 마트	80050365	18860						
12	2030BC	서 울	대 청 마트	88003467	2500						
13	2030BC	서 울	대 청 마트	88003696	58000						
14	2030BC	서 울	대 청 마트	88004259	447440						
15	2030BC	서 울	대 청 마트	88004266	293580						
16	2030BC	서 울	대 청 마트	88004471	481390						
17	2030BC	서 울	대 청 마트	88004488	596750						
18	2030BC	서 울	대 청 마트	88005317	783430						
19	2030BC	서 울	대 청 마트	88005324	859320						
20	2030BC	서 울	대 청 마트	88005509	92650						
21	2030BC	서 울	대 청 마트	88006609	575400						
22	2030BC	서 울	대 청 마트	88006611	432000						
23	2030BC	서 울	대 청 마트	88007519	882800						
24	2030BC	서 울	대 청 마트	88007748	263120						
25	2030BC	서 울	대 청 마트	88008809	455300						
26	2030BC	서 울	대 청 마트	88009505	743750						
27	2030BC	서 울	대 청 마트	88009537	80000						
28	2030BC	서 울	대 청 마트	88009861	132000						
29	2030BC	서 울	대 청 마트	88010261	32200						
30	2030BC	서 울	대 청 마트	88010268	3880						
31	2030BC	서 울	대 청 마트	88010700	123000						
32	2030BC	서 울	대 청 마트	98010717	672750						

- 데이터 로드 – 작업 폴더에서 pds\_seoul\_2013.csv를 더블클릭하여 오픈오피스 스프레드시트를 실행한다.



## IV 저 장

개요

27

가공 데이터 저장

28

### > 개요

오픈오피스 스프레드시트를 활용하여 데이터 로드 > 가공 > 분석 > 시각화 단계를 한번에 실행할 수 있으나, 기초 데이터를 유지하기 위하여 가공 단계에서 가공한 데이터를 별도로 저장한다.



### > 저장 방법

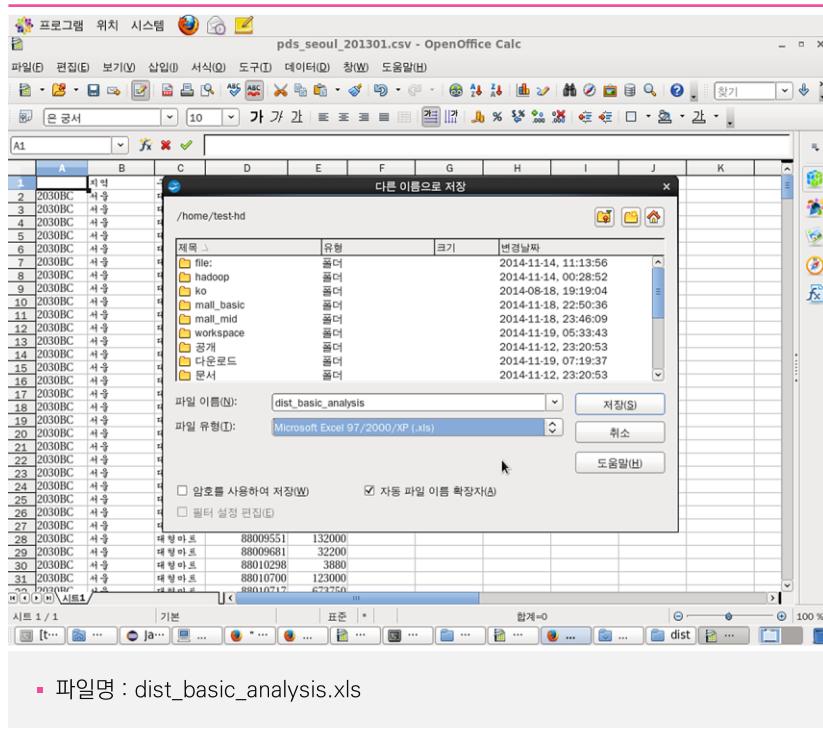
- 오픈 오피스 툴을 활용하여 읽어들인 csv 파일을 다양하게 분석하기 위해 ‘다른 이름으로 저장’하여 문서를 저장한다.

## > 가공 데이터 저장

### > 오픈오피스 스프레드 시트 데이터 저장

▪ 오픈오피스 스프레드시트에서 메뉴/파일/다른이름으로 저장을 선택하여 별도 파일을 저장한다.

#### IV. 저장



I. 개요

II. 수집

III. 가공

IV. 저장

V. 분석

VI. 시각화

# W





## V 분석

개요

33

데이터 분석 과정

34

# V 분석

## > 개요

유통 데이터 분석은 오픈오피스 스프레드 시트의 피봇 테이블 기능과 차트 기능을 활용한다. 저장 단계에서 저장한 가공 데이터(2013년 1월 서울 유통 정보) 파일을 읽어들여, 지역과 매장 형태 컬럼을 기준으로 피봇 테이블을 실행하고, 필터 기능을 활용하여 대형마트의 월간 공급 금액 통계를 계산한다.

## > 분석 내용

- 매장별, 품목별 공급금액 데이터를 읽어들여 매장별 매출 규모를 비교 분석한다.

## > 분석 방법

- [저장]과정에서 저장한 dist\_basic\_analysis.xls 파일을 읽어들인다.  
(열어 놓은 상태라면 바로 사용한다.)
- 스프레드 시트의 피봇테이블 기능을 활용하여 전체 매장별 공급금액 합계를 산출한다.
- 피봇테이블의 필터 기능을 사용하여 '서울'지역 '대형마트' 공급금액을 추출한다.
- **통계 분석 기술** : 매장별, 품목별 판매 데이터로부터 매장별, 매장 형태별 공급금액 합계를 계산하기 위해 스프레드 시트의 피봇테이블(Pivot Table) 기능을 활용한다.

## > 데이터 분석 과정

### > 데이터 로드

■ 1차 가공, 저장한 데이터 파일((dist\_basic\_analysis.ods)을 오픈오피스에서 읽어들인다.

## ▶ 피봇 테이블을 활용한 매출 통계 분석

- 오픈오피스의 피봇테이블 기능을 활용하여 연월(YYYYMM)별 매출(판매 금액 합계)를 산출한다.

The screenshot shows a Microsoft Windows desktop environment with the OpenOffice Calc application running. The window title is "pds\_seoul 201301.csv - OpenOffice Calc". The menu bar at the top includes "프로그램", "위치", "시스템", "파일(F)", "편집(E)", "보기(V)", "삽입(I)", "서식(O)", "도구(U)", "데이터(D)", "창(W)", and "도움말(H)". The "데이터(D)" menu is currently open, displaying options such as "범위 정의(I)...", "범위 선택(R)...", "정렬(S)...", "필터(F)...", "부분합(P)...", and "유형설(Y)...". A sub-menu "피벗 테이블(P)" is also open under "데이터(D)", with "만들기(M)..." highlighted. The main spreadsheet area contains a table with columns A, B, C, and D, and rows numbered 1 to 33. Column A contains dates from 201301 to 201309. Column B contains city names like "서울" and "대구". Column C contains store names like "아이 링" and "대형 마트". Column D contains numerical values representing sales amounts. The status bar at the bottom shows "시트 1 / 1", "기본", "표준", "합계=0", and "100 %".

- 1. 메뉴/데이터/피벗테이블/만들기를 선택하여 피벗테이블 마법사를 실행한다.

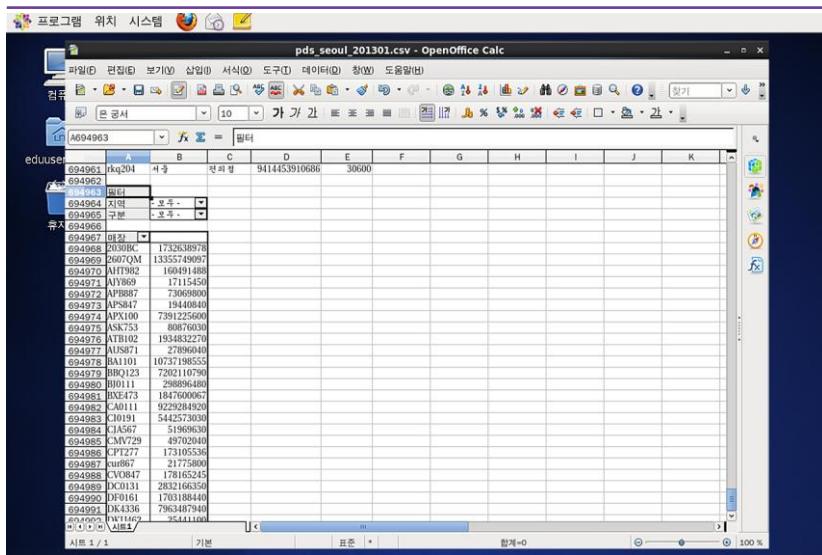
A screenshot of the OpenOffice Calc interface. A context menu is open over a cell containing the value '50213245'. The menu is titled '선택' (Select) and includes three options: '현재 선택' (Current Selection) (selected), 'OpenOffice 등록된 데이터 원본' (OpenOffice Registered Data Source), and '외부 웹본/인터넷페이지' (External Webpage/Internet Page). There are also '확인' (OK), '취소' (Cancel), and '도움말' (Help) buttons.

- 2. 자동으로 범위가 선택되어져 있으므로 '현재선택' 상태에서 '확인'버튼을 클릭한다.

A screenshot of the OpenOffice Calc interface showing the '피벗 테이블' (Pivot Table) dialog box. The '필드' (Fields) section is expanded, showing '매장' (Branch), '지역' (Region), '구분' (Category), and '데이터 필드' (Data Field). The '데이터 필드' section contains '합계 - 공급금액' (Sum - Supply Amount). Buttons for '확인' (OK), '취소' (Cancel), '도움말' (Help), '제거' (Delete), and '옵션' (Options) are visible.

- 3. 우측 '필드'영역에서 아이템들(매장, 지역, 구분, 공급금액)을 각각 드래그하여 위 화면과 같이 배치한다.

## V. 분석



- 4. 데이터의 최하단까지 스크롤하면 피벗테이블의 결과값이 출력된 것을 확인할 수 있다.

I. 개요

II. 수집

III. 가공

IV. 저장

V. 분석

VI. 시각화



1

2



## VI 시각화

개요	41
시각화 과정	42
시각화 데이터 분석	45

# VI

## 시각화

### > 개요

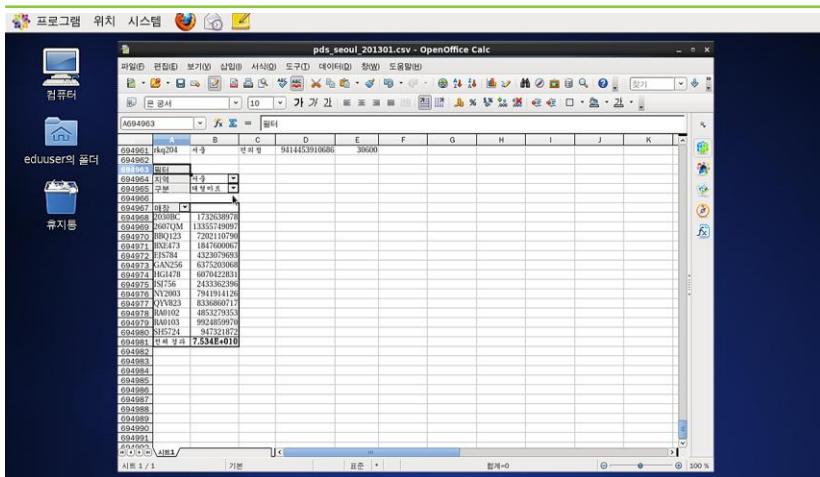
유통 데이터 분석 과정에서 오픈오피스 스프레드시트의 피벗테이블 기능을 활용하여 2010년 서울 지역 대형마트들의 매출 통계를 계산하였다. 이 분석 결과를 읽어 들여 스프레드 시트의 차트 기능을 활용하여 막대그래프 형태로 시각화하여, 주요 대형마트들의 연간 매출을 비교 분석한다.

### > 시각화 방법

- 분석 결과 데이터로부터 시각화 대상 데이터를 필터링한다.
- 스프레드 시트의 차트를 사용하여 시각화한다.
- **활용 기술** : 스프레드 시트의 필터 기능 및 차트 사용법

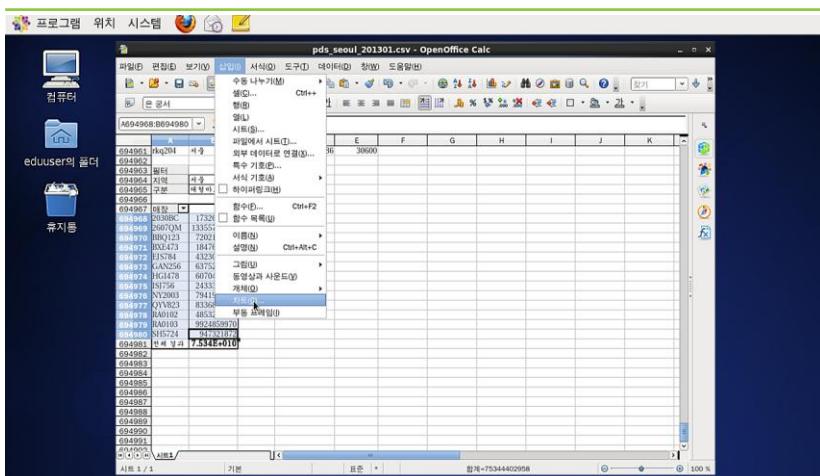
## > 시각화 과정

#### > 데이터 필터링



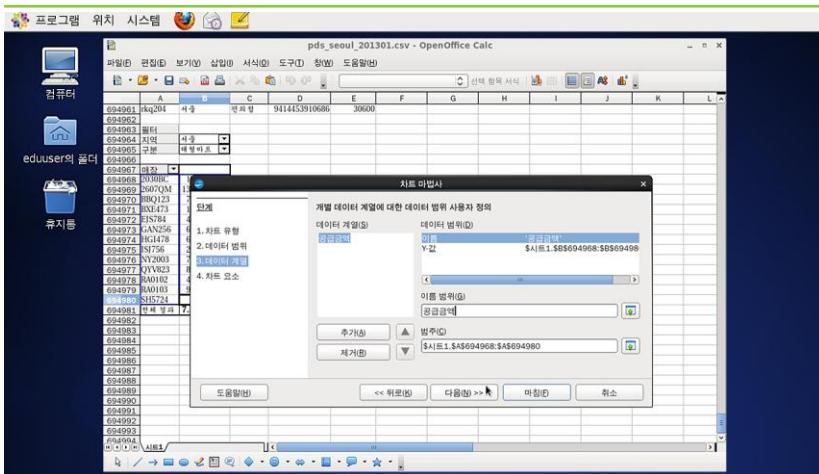
- 지역과 구분 필터를 '서울'과 '대형마트'를 선택하여 서울지역 대형마트 데이터만 필터링한다.

## > 차트 생성

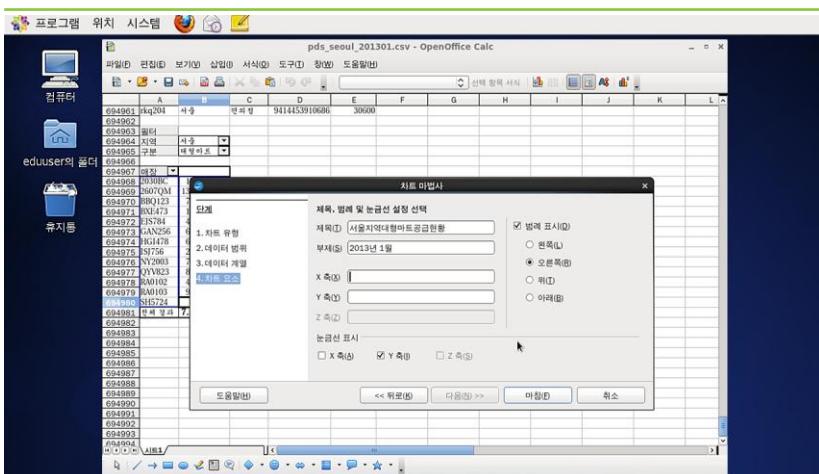


- 시각화 할 데이터 범위를 선택하여 지정한 후, 메뉴/삽입/차트(C)를 선택하여 차트 마법사를 실행한다

## ▶ 차트 설정

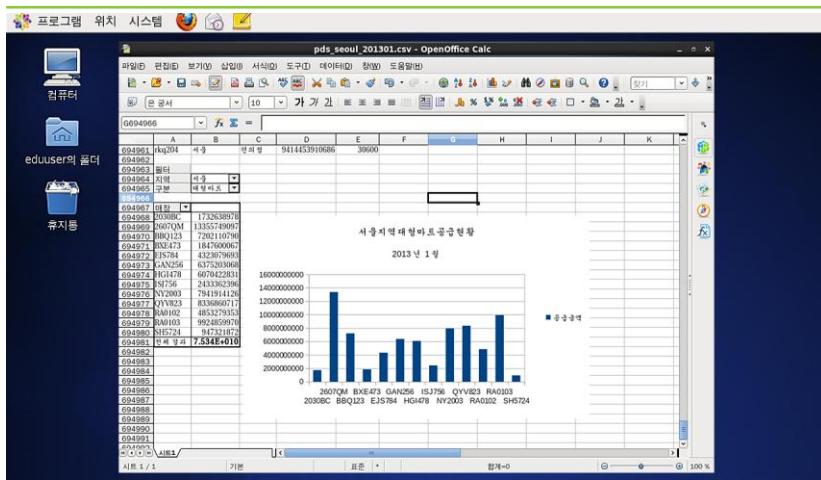


- 다른 것은 그대로 두고 '다음'을 두번 눌러, '데이터 계열' 단계에서 위 화면과 같이 이름 범위에 '공급금액'을 입력한 후 다음을 클릭한다.



- 위 화면과 같이 제목과 부제를 입력한 후 '마침'을 눌러 차트를 생성한다.

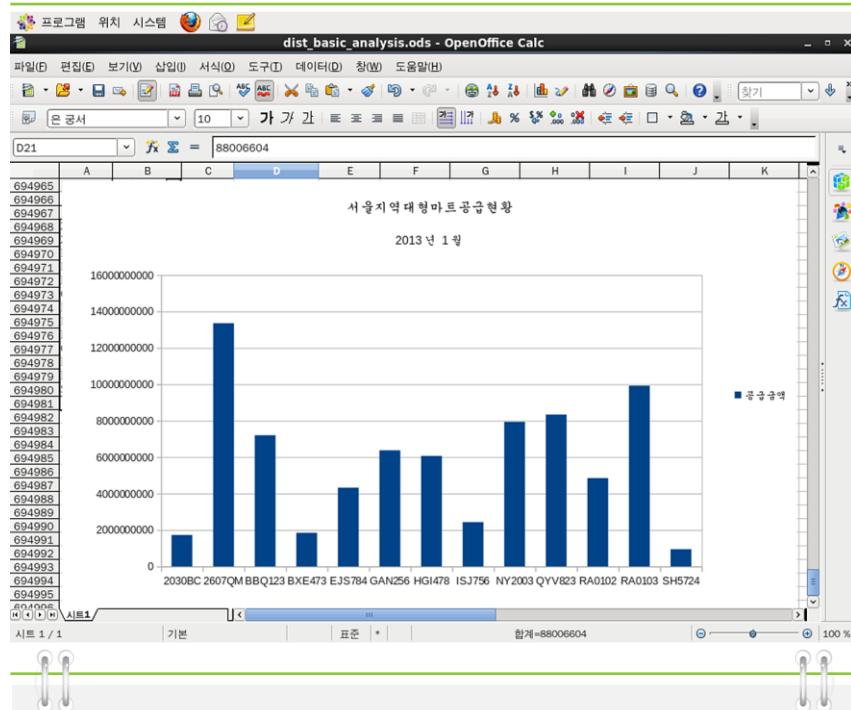
## > 차트 확인



- 막대그래프 형태로 각 서울지역 매장별 2013년 1월의 총 물품 공급 금액이 시각화된 것을 볼 수 있다.

## ▶ 시각화 데이터 분석

### ▶ 서울지역 대형마트 공급 현황



- 2013년 1월 물품 공급량을 보면 2607QM 매장의 유통량이 가장 많으며, 가장 적은 SH5724 매장과의 차이는 10배가 넘는다는 것을 알 수 있다.
- 실습용 데이터이기 때문에, 자세한 매장명을 알 수 없지만, 실제에서는 대형 마트의 유통 공급량을 보고 매장 규모 및 해당 지역의 소비수준 및 인구 등을 유추할 수 있는 기초 자료가 된다.
- 연도별, 매장별, 지역별로 확장하여 여러 가지 조건으로 분석하면 좀 더 의미 있는 정보를 획득할 수 있다.

I. 개요

II. 수집

III. 가공

IV. 저장

V. 분석

VI. 시각화



## VII 예제문제

예제 문제1

49

예제 문제2

50

# 예 / 제 / 문 / 제

## 예제 1

온라인 쇼핑몰의 연도별 월간 매출 변동을 분석하라.

- 2010~2012년 3년간의 월간 쇼핑몰 매출 추이를 분석하고, 꺾은선 그래프를 활용하여 시각화하라.

- 원시 데이터를 가공하여 필요한 필드만 추출한다.
- 연별로 월간 매출 추이를 구한다.
- 3년간의 월간 매출 추이를 취합한다.

## 예제 2

### 분기별 최고 공급량 아이템을 파악하라.

- 3년간 분기별 최고 매출 아이템 및 해당 아이템의 매출액을 파악하고, 최고 매출 아이템이 전체 매출액에서 차지하는 비율을 분석하라.

- 쇼핑몰 매출 원시데이터로부터 분기별, 아이템별 매출액 합계를 구한다.
- 분기별로 매출액 내림차순으로 데이터를 정렬한다.
- 분기별 최고 매출 아이템과 전체 매출액 대비 비중을 계산한다.



유통 

Intermediate Level

중급과정







## I 개요

개요

55

54

# I

## 개요

### > 개요

월별로 취합하여 제공된 대한상공회의소의 2013년 전국 소매점 유통 정보 데이터를 별도의 통합 가공 과정 없이 하둡에서 한꺼번에 읽어들여 지역별, 월별로 맵리듀싱 처리하고, 유통 공급량 통계를 계산하여 시계열 패턴 분석하는 과정을 통해 하둡과 자바를 이용한 대용량 분산 데이터 통계 분석 기법을 학습한다.

### > 활용 데이터

- **201312pdsfile.csv** : 2013년 12월 전국 주요 소매점 유통 공급 정보
- **201311pdsfile.csv** : 2013년 11월 전국 주요 소매점 유통 공급 정보
- **201310pdsfile.csv** : 2013년 10월 전국 주요 소매점 유통 공급 정보
- **201309pdsfile.csv** : 2013년 9월 전국 주요 소매점 유통 공급 정보
- **201308pdsfile.csv** : 2013년 8월 전국 주요 소매점 유통 공급 정보
- **201307pdsfile.csv** : 2013년 7월 전국 주요 소매점 유통 공급 정보
- **201306pdsfile.csv** : 2013년 6월 전국 주요 소매점 유통 공급 정보
- **201305pdsfile.csv** : 2013년 5월 전국 주요 소매점 유통 공급 정보
- **201304pdsfile.csv** : 2013년 3월 전국 주요 소매점 유통 공급 정보
- **201303pdsfile.csv** : 2013년 3월 전국 주요 소매점 유통 공급 정보
- **201302pdsfile.csv** : 2013년 2월 전국 주요 소매점 유통 공급 정보
- **201301pdsfile.csv** : 2013년 1월 전국 주요 소매점 유통 공급 정보

## > 선행학습

- **리눅스** – 파일시스템 구조, 쉘 명령어, 쉘 스크립트 실행 방법
- **하둡 에코시스템** – 하둡 시작, 종료, 하둡 파일 시스템 명령어, 맵리듀스 실행 방법
- **자바** – 자바코딩, 자바컴파일, JDK 설치, jar 파일 만드는 방법
- **오픈오피스** – 피벗테이블 기능, 차트 사용 방법

## > 요구사항

- 2013년 전국 소매점 월별, 매장별, 품목별 공급 정보 데이터를 분석하여 주요 매장의 지역별 연간 유통량 추이를 비교 분석하라.

## > 분석 절차

- 연도별로 제공된 판매 데이터를 하둡 파일 시스템으로 로드한다.
- 전국 매장 정보 데이터와 아이템(품목) 정보 데이터를 하둡 파일 시스템으로 로드한다.
- 자바와 하둡을 이용하여 지역별, 월별로 맵리듀싱을 실행하여 1년간의 지역별 월별 유통 금액 합계를 계산한다.
- 계산 결과를 꺾은선 멀티 그래프를 활용하여 지역별로 시각화해 본다.
- 시각화 한 데이터를 보고, 지역별 월별 유통 공급량의 추이를 분석한다.
- 지역별 유통 공급량의 규모(레벨) 차이를 확인하고, 지역별로 계절별 공급량 변화 패턴이 일치하는지를 분석한다.



## II 수집

개요	59
교육용 데이터 샘플	60
데이터 수집	62
데이터 작업 영역 이동 스크립트	66



## 수집

### ▶ 개요

유통 데이터는 대한상공회의소에서 제공 받은 2013년 서울 주요 소매점 (대형 마트, 편의점, 슈퍼 등) 유통 공급 정보 데이터를 수집하여 분석 목적을 달성할 수 있는 한도 내에서 개인정보 비식별화 및 특정 상품 비식별화 처리를 통해 분석에 용이하게 편집하여 제공한다.

### ▶ 수집 방법

- **데이터 제공** : 유통 공급 정보 데이터는 대한상공회의소에서 제공하는 데이터를 OpenAPI, 자료수집기(Crawler)를 통하여 데이터를 수집하였고, 실습용 자료는 빅데이터 분석 활용센터에 접속하여 유통 데이터 셋을 다운로드할 수 있도록 원시데이터를 제공하고 있다.



용 어 정 리

- **비식별화** : 데이터 값 삭제, 가명처리, 총계처리, 범주화, 데이터 마스킹 등을 통해 개인정보의 일부 또는 전부를 삭제하거나 대체함으로써 다른 정보와 쉽게 결합하여도 특정 개인을 식별할 수 없도록 하는 조치를 말한다.

- \*출처: 방송통신위원회, “빅데이터 개인정보보호 가이드라인”, 작성일 2014.12.23

## > 교육용 데이터 샘플

### > 2013년 1월 소매점 유통 정보 데이터 샘플(201301pdsfile.csv)

매장	지역	매장형태	품목	공급금액
2030BC	서울	대형마트	930890	27720
2030BC	서울	대형마트	3435418	114150
2030BC	서울	대형마트	940810	176300
2030BC	서울	대형마트	3456118	516750
2030BC	서울	대형마트	10557093	267020
2030BC	서울	대형마트	50390427	21600
2030BC	서울	대형마트	50430048	5400
2030BC	서울	대형마트	80050278	1550
2030BC	서울	대형마트	88002026	18960
2030BC	서울	대형마트	88003467	2500

- 위와 같은 형태로 2013년 1월~ 2013년 12월 유통 정보 데이터가 제공된다.
  - 2013년 1월(201301pdsfile.csv)
  - 2013년 2월(201302pdsfile.csv)
  - 2013년 3월(201303pdsfile.csv)
  - 2013년 4월(201304pdsfile.csv)
  - 2013년 5월(201305pdsfile.csv)
  - 2013년 6월(201306pdsfile.csv)
  - 2013년 7월(201307pdsfile.csv)
  - 2013년 8월(201308pdsfile.csv)
  - 2013년 9월(201309pdsfile.csv)
  - 2013년 10월(201310pdsfile.csv)
  - 2013년 11월(201311pdsfile.csv)
  - 2013년 12월(201312pdsfile.csv)

## II. 수집

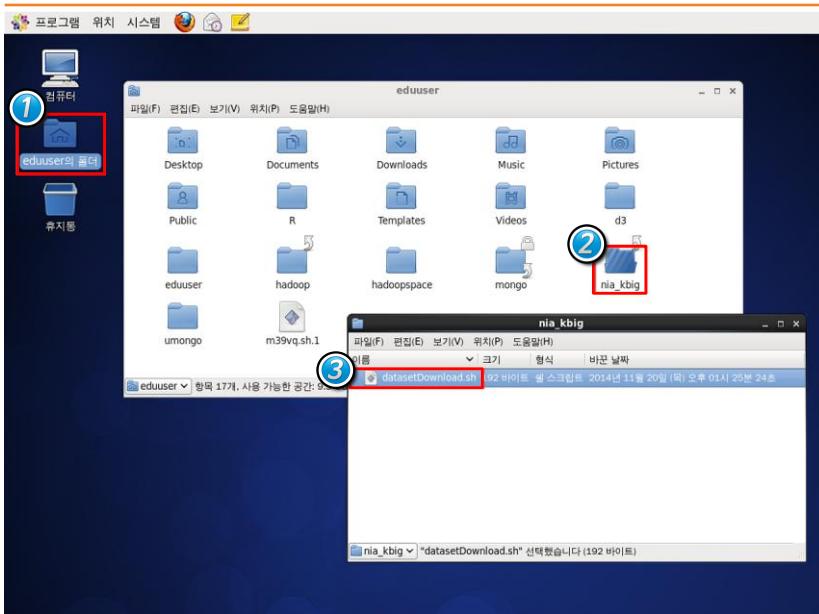
### ➤ 매장 마스터 정보(shop\_master.csv)

매장	지역	매장형태	품목	공급금액
2030BC	서울	대형마트	930890	27720
2030BC	서울	대형마트	3435418	114150
2030BC	서울	대형마트	940810	176300
2030BC	서울	대형마트	3456118	516750
2030BC	서울	대형마트	10557093	267020
2030BC	서울	대형마트	50390427	21600
2030BC	서울	대형마트	50430048	5400
2030BC	서울	대형마트	80050278	1550
2030BC	서울	대형마트	88002026	18960
2030BC	서울	대형마트	88003467	2500

## > 데이터 수집(datasetDownload.sh)

- 데이터 저장소에서 서버 로컬로 유통 데이터 셋을 복사해 온다.
- **2013\*pds.csv** : 2013년 01월~2013년 12월 쇼핑몰 거래 데이터

### > 실습코드 디렉토리로 이동

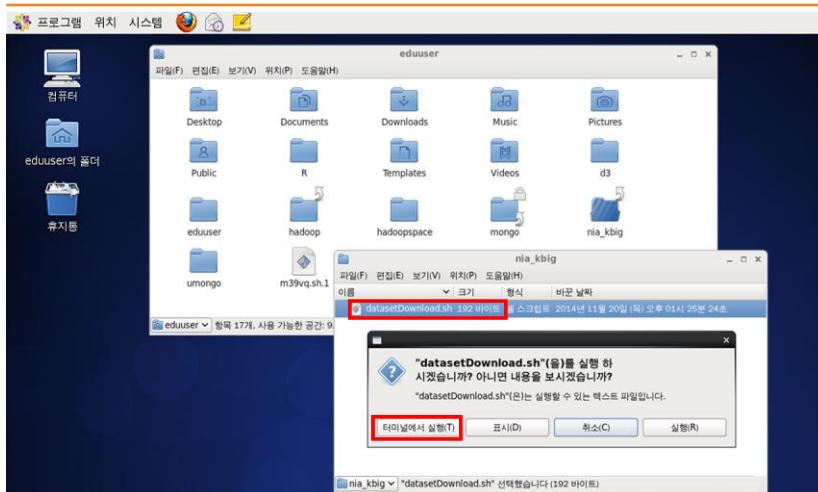


- ① 로그인 후 바탕화면에서 eduuser 폴더를 오픈한다.
- ② nia\_kbig 폴더를 오픈한다.
- ③ datasetDownload.sh를 더블클릭하여 실행한다.

## II. 수집

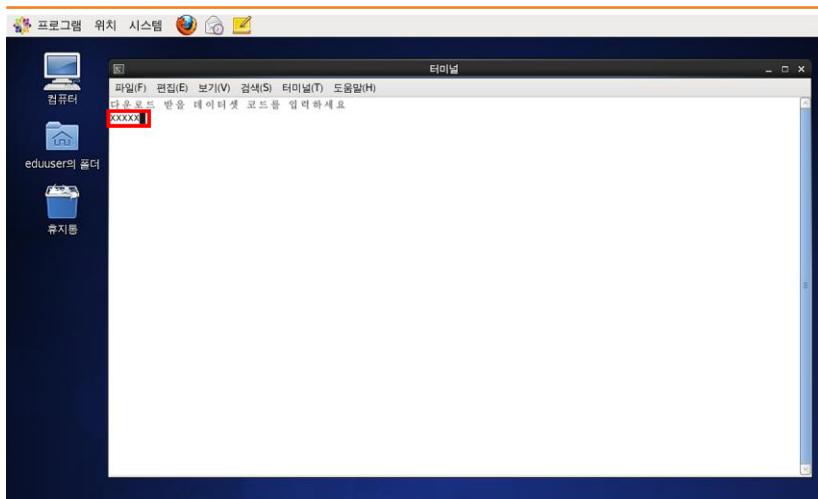
### ▶ 레파지토리에서 데이터 수집

datasetDownload.sh (원시데이터로 컬서버로 복사)



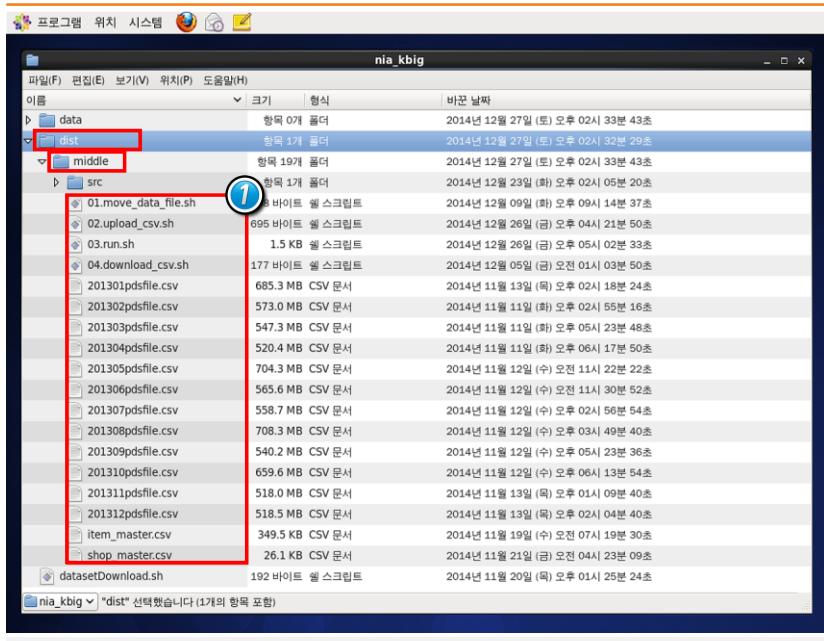
- '터미널에서 실행' 버튼을 클릭한다.

### ▶ 데이터셋 코드 입력



- 다운로드 받을 데이터셋 코드를 입력 후 엔터

## ▶ 데이터셋과 실습용 쉘 스크립트



- 실습용 데이터셋과 실습용 스크립트를 확인한다.

## ▶ ① 데이터 및 스크립트

### ▪ 01.move\_data\_file.sh :

로컬로 수집해온 데이터를 작업 영역 Data 폴더로 자료를 이동하는 스크립트

### ▪ 02.upload\_csv.sh :

유통 데이터를 하둡 파일시스템으로 업로드하는 스크립트

### ▪ 03.run.sh:

유통 분석용 하둡 맵리듀스 프로그램 실행 스크립트

### ▪ 04.download\_csv.sh :

하둡 맵리듀스 분석 결과 파일을 다운로드하는 스크립트.

### ▪ 201312pdsfile.csv : 2013년 12월 전국 주요 소매점 유통 공급 정보

### ▪ 201311pdsfile.csv : 2013년 11월 전국 주요 소매점 유통 공급 정보

## II. 수집

- **201310pdsfile.csv** : 2013년 10월 전국 주요 소매점 유통 공급 정보
- **201309pdsfile.csv** : 2013년 9월 전국 주요 소매점 유통 공급 정보
- **201308pdsfile.csv** : 2013년 8월 전국 주요 소매점 유통 공급 정보
- **201307pdsfile.csv** : 2013년 7월 전국 주요 소매점 유통 공급 정보
- **201306pdsfile.csv** : 2013년 6월 전국 주요 소매점 유통 공급 정보
- **201305pdsfile.csv** : 2013년 5월 전국 주요 소매점 유통 공급 정보
- **201304pdsfile.csv** : 2013년 3월 전국 주요 소매점 유통 공급 정보
- **201303pdsfile.csv** : 2013년 3월 전국 주요 소매점 유통 공급 정보
- **201302pdsfile.csv** : 2013년 2월 전국 주요 소매점 유통 공급 정보
- **201301pdsfile.csv** : 2013년 1월 전국 주요 소매점 유통 공급 정보

## > 데이터 작업 영역 이동 스크립트(01.move\_data\_file.sh)

### > 데이터 이동 스크립트

- 로컬로 수집해온 데이터를 작업 영역 Data 폴더로 자료를 이동하는 스크립트

#### 01.move\_data\_file.sh (작업영역 폴더로 원시데이터 이동)

```

01. 
02. #!/bin/bash
03. # 복사 대상 파일 정의
04. # 전국 소매점 월별 유통 내역 파일
05. TARGET_DIST_2013=/home/eduuser/nia_kbig/dist/middle/2013*pdsfile.csv
06. # 매장정보 파일
07. TARGET_DIST_SHOPMASTER=/home/eduuser/nia_kbig/dist/middle/shop_ma
08. ster.csv
09. # 품목정보 파일
10. TARGET_DIST_ITEMMASTER=/home/eduuser/nia_kbig/dist/middle/item_m
11. aster.csv
12. # 작업영역 디렉토리 정의
13. LOCAL_DIR=/home/eduuser/nia_kbig/data/
14. mv $TARGET_DIST_2013 $LOCAL_DIR
15. mv $TARGET_DIST_SHOPMASTER $LOCAL_DIR
16. mv $TARGET_DIST_ITEMMASTER $LOCAL_DIR

```

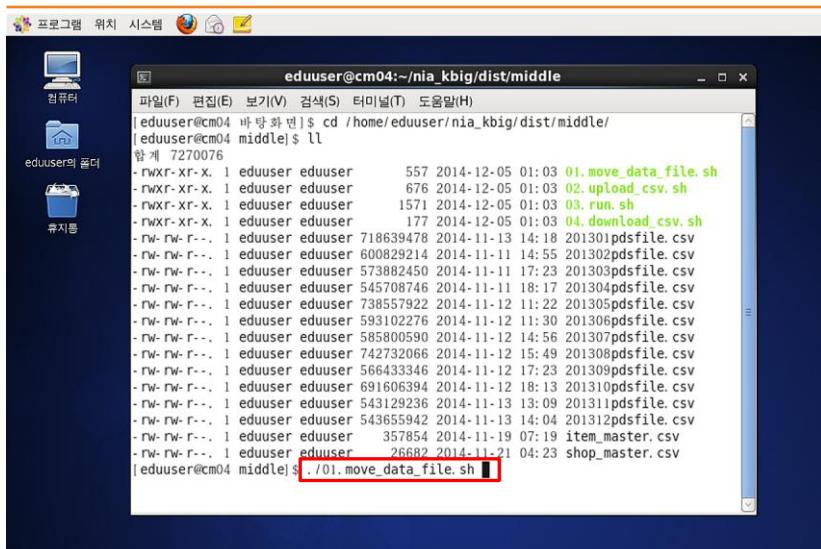


#### 부연설명

- 데이터 작업 영역 이동 스크립트 소스(01.move\_data\_file.sh)
- 라인 05~09 : 다운로드 받은 원시데이터 파일들의 위치(path)를 변수(TARGET\_DIST\_2013, TARGET\_DIST\_SHOPMASTER, TARGET\_DIST\_ITEMMASTER)로 지정하는 라인이다.
- 라인 11 : 작업영역 디렉토리의 위치(path)를 변수(LOCAL\_DIR)로 지정하는 라인이다.
- 라인 12~14 : mv 명령어를 사용하여 다운로드 받은 원시데이터 파일들을 작업영역 디렉토리로 이동시키는 라인이다.

## II. 수집

### ▶ 수집 데이터 셋 작업영역 폴더 이동



```
eduuser@cm04:~/nia_kbig/dist/middle
[eduuser@cm04 middle]$ ll
합계 7270076
-rwxr--r-- 1 eduuser eduuser      557 2014-12-05 01:03 01.move_data_file.sh
-rwxr--r-- 1 eduuser eduuser     676 2014-12-05 01:03 02.upload_csv.sh
-rwxr--r-- 1 eduuser eduuser    1571 2014-12-05 01:03 03.run.sh
-rwxr--r-- 1 eduuser eduuser     177 2014-12-05 01:03 04.download_csv.sh
-rw-rw-r-- 1 eduuser eduuser 718639478 2014-11-13 14:18 201301pdfsfile.csv
-rw-rw-r-- 1 eduuser eduuser 600829214 2014-11-11 14:55 201302pdfsfile.csv
-rw-rw-r-- 1 eduuser eduuser 573882450 2014-11-11 17:23 201303pdfsfile.csv
-rw-rw-r-- 1 eduuser eduuser 545708746 2014-11-11 18:17 201304pdfsfile.csv
-rw-rw-r-- 1 eduuser eduuser 738557922 2014-11-12 11:22 201305pdfsfile.csv
-rw-rw-r-- 1 eduuser eduuser 593102276 2014-11-12 11:30 201306pdfsfile.csv
-rw-rw-r-- 1 eduuser eduuser 585800590 2014-11-12 14:56 201307pdfsfile.csv
-rw-rw-r-- 1 eduuser eduuser 742732066 2014-11-12 15:49 201308pdfsfile.csv
-rw-rw-r-- 1 eduuser eduuser 566433346 2014-11-12 17:23 201309pdfsfile.csv
-rw-rw-r-- 1 eduuser eduuser 691606394 2014-11-12 18:13 201310pdfsfile.csv
-rw-rw-r-- 1 eduuser eduuser 543129236 2014-11-13 13:09 201311pdfsfile.csv
-rw-rw-r-- 1 eduuser eduuser 543655592 2014-11-13 14:04 201312pdfsfile.csv
-rw-rw-r-- 1 eduuser eduuser 357854 2014-11-19 07:19 item_master.csv
-rw-rw-r-- 1 eduuser eduuser 26682 2014-11-21 04:23 shop_master.csv
[eduuser@cm04 middle]$ ./01.move_data_file.sh
```

- 로컬에 원시데이터를 작업 영역 폴더로 이동 (/home/eduuser/nia\_kbig/data/) 시킨다.
- ./01.move\_data\_file.sh 입력 후 엔터

I. 개요

II. 수집

III. 가공

IV. 저장

V. 분석

VI. 시각화





개요

71

데이터 가공 스크립트

72



## 가공

### > 개요

작업 영역 폴더에 복사한 유통 데이터의 가공은, 자바와 하둡을 이용한 분석 과정에서 동시에 진행되기 때문에, 별도의 가공 과정은 생략한다. 하둡에서 데이터를 일어들여 맵리듀스 분석하는 과정에서 필요한 데이터만 추출하고 계산하는 일련의 가공 과정이 메모리상에서 동시에 이루어 진다.

### > 가공 방법

- 대한상공회의소에서 제공된 2013년 매장별, 품목별 유통 정보 데이터는 월별로 분할된 텍스트 파일로 저장되어 있다.
- 하둡과 자바를 활용하면 맵 리듀싱을 진행하면서 대부분의 가공 작업을 할 수 있으므로, 자세한 내용은 분석 과정에서 설명한다.

### > 데이터셋

- 2013년 1월 소매점 유통 정보 데이터 샘플(201301pdsfile.csv)

매장	지역	매장형태	품목	공급금액
2030BC	서울	대형마트	930890	27720
2030BC	서울	대형마트	3435418	114150
2030BC	서울	대형마트	940810	176300
2030BC	서울	대형마트	3456118	516750
2030BC	서울	대형마트	10557093	267020
2030BC	서울	대형마트	50390427	21600
2030BC	서울	대형마트	50430048	5400
2030BC	서울	대형마트	80050278	1550
2030BC	서울	대형마트	88002026	18960
2030BC	서울	대형마트	88003467	2500

## ▶ 데이터 가공 스크립트

### ▶ 데이터 가공 스크립트

- 자바와 하둡을 이용하여 가공 및 분석 실행
- 자바를 활용하여 맵리듀스 분석을 할 경우에는 [분석] 단계에서 가공 및 분석을 동시에 진행할 수 있으므로 [분석] 단계에서 상세 설명을 하기로 한다.



## IV 저 장

개요

75

가공 데이터 하둡 파일시스템 업로드

76

## > 개요

자바와 하둡을 이용하여 맵리듀스를 실행하기 위해서는 하둡 파일 시스템에 데이터를 업로드하여야 한다. 따라서, 하둡 파일 시스템에서 제공하는 커맨드를 사용하여 2013년 월별 대한상공회의소 유통 정보 데이터 파일을 업로드한다.

## > 저장 방법

- 2013년 월별 대한상공회의소 데이터 파일(2013\*pdsfile.csv)과 마스터 정보 파일(shop\_master.csv, item\_master.csv)을 하둡에 업로드한다.
- 하둡 커맨드를 이용해서 업로드한다.

## > 가공 데이터 하둡 파일시스템 업로드(02.upload\_csv.sh)

### > 하둡 파일시스템에 업로드 스크립트

- 가공 데이터를 하둡 파일시스템으로 업로드(02.upload\_csv.sh)

#### 02.upload\_csv.sh (가공데이터를 하둡파일시스템으로 업로드)

```

01.#!/bin/bash
02. # 2013년 전국 소매점 유통 정보 데이터
03. TRANS_OUTPUT_FILE_2013= '/home/eduuser/nia_kbig/data/2013*pdsfile.
    ↵ csv'
04.
05. # 매장 정보 데이터
06. TRANS_OUTPUT_FILE_SHOPMASTER= '/home/eduuser/nia_kbig/data/sho
    ↵ p_master.csv'
07. # 품목 정보 데이터
08. TRANS_OUTPUT_FILE_ITEMMASTER= '/home/eduuser/nia_kbig/data/item
    ↵ _master.csv'
09.
10. # 하둡의 출력결과 저장 위치
11. HDFS_TRANSACTION=/user/bigdata/
12.
13. # 파일을 하둡의 파일 시스템에 업로드
14. hadoop fs -put $TRANS_OUTPUT_FILE_2013 $HDFS_TRANSACTION
15. hadoop fs -put $TRANS_OUTPUT_FILE_SHOPMASTER $HDFS_TRANSACTION
16. hadoop fs -put $TRANS_OUTPUT_FILE_ITEMMASTER $HDFS_TRANSACTION

```



- **하둡 파일시스템 업로드 스크립트(02.upload\_csv.sh)**
- **라인 02~08 :** 작업영역으로 이동한 원시데이터 파일들의 위치(path)를 변수(TRANS\_OUTPUT\_FILE\_2013, TRANS\_OUTPUT\_FILE\_2011\_1, TRANS\_OUTPUT\_FILE\_SHOPMASTER, TRANS\_OUTPUT\_FILE\_ITEMMASTER)로 지정하는 라인이다.
- **라인 11 :** 하둡 파일시스템에 업로드할 파일들의 위치(path)를 변수(HDFS\_TRANSACTION)로 지정하는 라인이다.
- **라인 14~16 :** hadoop fs -put 명령어를 사용하여 원시데이터 파일들을 하둡 파일시스템으로 업로드하는 라인이다.

#### IV. 저장

The screenshot shows a terminal window titled "eduuser@cm04:~/nia\_kbig/dist/middle". The window contains the following text:

```
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[ eduuser@cm04 nia_kbig]$ cd dis
bash: cd: 그런 파일이나 디렉터리가 없습니다
[ eduuser@cm04 nia_kbig]$ cd dist/middle/
[ eduuser@cm04 middle]$ ll
합계 7270076
-rwxr-Xr-X 1 eduuser eduuser      557 2014-12-05 01:03 01.move_data_file.sh
-rwxr-Xr-X 1 eduuser eduuser     676 2014-12-05 01:03 02.upload_csv.sh
-rwxr-Xr-X 1 eduuser eduuser    1571 2014-12-05 01:03 03.run.sh
-rwxr-Xr-X 1 eduuser eduuser     177 2014-12-05 01:03 04.download_csv.sh
-rw-rw-r-- 1 eduuser eduuser 718639478 2014-11-13 14:18 201301pdsfile.csv
-rw-rw-r-- 1 eduuser eduuser 600829214 2014-11-11 14:55 201302pdsfile.csv
-rw-rw-r-- 1 eduuser eduuser 573882450 2014-11-11 17:23 201303pdsfile.csv
-rw-rw-r-- 1 eduuser eduuser 545708746 2014-11-11 18:17 201304pdsfile.csv
-rw-rw-r-- 1 eduuser eduuser 738557922 2014-11-12 11:22 201305pdsfile.csv
-rw-rw-r-- 1 eduuser eduuser 593102276 2014-11-12 11:30 201306pdsfile.csv
-rw-rw-r-- 1 eduuser eduuser 585800590 2014-11-12 14:56 201307pdsfile.csv
-rw-rw-r-- 1 eduuser eduuser 742732066 2014-11-12 15:49 201308pdsfile.csv
-rw-rw-r-- 1 eduuser eduuser 566433346 2014-11-12 17:23 201309pdsfile.csv
-rw-rw-r-- 1 eduuser eduuser 691606394 2014-11-12 18:13 201310pdsfile.csv
-rw-rw-r-- 1 eduuser eduuser 543129236 2014-11-13 13:09 201311pdsfile.csv
-rw-rw-r-- 1 eduuser eduuser 543655942 2014-11-13 14:04 201312pdsfile.csv
-rw-rw-r-- 1 eduuser eduuser   357854 2014-11-19 07:19 item_master.csv
-rw-rw-r-- 1 eduuser eduuser   26682 2014-11-21 04:23 shop_master.csv
[ eduuser@cm04 middle]$ ./02.upload_csv.sh
```

A red box highlights the command `./02.upload_csv.sh`. Below the terminal window, a note indicates that the command was run after entering it.

▪ ./02.upload\_csv.sh 입력 후 엔터

I. 개요

II. 수집

III. 가공

IV. 저장

V. 분석

VI. 시각화

# W





## V 분석

개요	81
데이터 분석 스크립트	83
데이터 분석 맵리द스 실행	91
결과 데이터 2차 분석	92

# V 분석

## > 개요

유통 데이터의 분석은 자바 스크립트와 하둡 파일 시스템의 맵리듀스 기능을 활용하여 여러 개의 파일로 분산되어 저장된 매장별, 품목별 데이터로부터 지역별 월간 매출 통계를 계산하고, 이를 바탕으로 지역별, 계절별 매출 패턴을 분석한다.

## > 분석 예제

- 2013년 전국 소매점 월별, 매장별, 품목별 공급 정보 데이터를 분석하여 주요 지역별 대형마트의 연간 유통량 추이를 비교 분석한다.

## ▶ 분석 방법

1. 월별, 매장별, 품목별 유통 공급 데이터 파일들(2013\*\*.pdfsfile.csv)을 하둡 파일 시스템으로 업로드한다.
2. 매장 정보(shop\_master.csv)와 품목 정보(item\_master.csv) 데이터 파일을 하둡 파일 시스템으로 업로드한다.
3. 맵리듀스 자바프로그램을 작성한다.(Distribution.java)
  - 매장 정보 데이터파일(shop\_master.csv)을 읽어들여 매장의 유형과 지역 정보를 메모리에 로드한다.
  - 월별, 매장별, 품목별 유통 공급 데이터 파일들(2013\*\*.pdfsfile.csv)에 대하여 맵리듀스 작업을 통해 2013년 지역별, 월별 총 유통 금액 데이터를 산출한다.
  - 산출 결과는 하둡 파일 시스템의 '/user/bigdata/dist/out' 경로에 파일로 출력하도록 한다.
4. 작성한 맵리듀스 자바 프로그램을 distribution.jar(Runnable Jar 형태)로 컴파일한다.
5. 콘솔에 로그인해서 실행은 하둡의 yarn 커맨드로 실행한다.
6. 콘솔에서 하둡 명령어를 통해 산출한 결과 데이터를 하둡으로부터 다운로드 한다.

### Tip ↗

• 가공 데이터 분석 스크립트(03.run.sh) 실행시, 맵리듀스 분석 실행 중 멈춤 현상 해결 방법

1. Ctrl+C 를 눌러 스크립트 실행 종료.
2. 하둡 종료 : 터미널 입력창에 stop-all.sh 입력 후 엔터.
3. 하둡 재실행 : 터미널 입력창에 start-all.sh 입력 후 엔터.
4. 하둡 실행 상태 확인 : 터미널 입력창에 jps 입력 후 엔터.  
(목록 중에 NodeManager가 존재하는지 확인한다.)
5. 가공 데이터 분석 스크립트(03.run.sh) 재실행 : 터미널 입력창에 ./03.run.sh 입력 후 엔터

## ➤ 데이터 분석 스크립트

### ➤ 분석 프로그램 작성

- 맵리듀스를 처리하는 프로그램은 Distribution.java에 구현되어 있다.
- 자바 프로그램을 컴파일하여 distribution.jar 파일로 만든 후 yarn 커맨드를 이용해서 distribution.jar 파일로 맵리듀스 작업을 수행한다.
- 분석 결과는 하둡 파일시스템의 지정한 디렉토리에 저장한다.

#### Distribution.java 작성

```

01. import java.io.BufferedReader;
02. import java.io.IOException;
03. import java.io.InputStreamReader;
04. import java.util.ArrayList;
05. import java.util.Iterator;
06.
07. import org.apache.hadoop.fs.FileSystem;
08. import org.apache.hadoop.fs.Path;
09. import org.apache.hadoop.conf.*;
10. import org.apache.hadoop.io.*;
11. import org.apache.hadoop.mapred.*;
12. import org.apache.hadoop.util.*;
13.
14.
15. public class Distribution extends Configured implements Tool{
16.     public static int data = 0;
17.
18.     public int run(String[] args) throws Exception
19.     {
20.         //하둡의 Job 이름 및 맵리듀스 클래스를 지정한다.
21.         JobConf conf = new JobConf(getConf(), Distribution.class);
22.         conf.setJobName("DistributionJob");
23.
24.         initShopData(conf);
25.

```

```

26. //맵리듀스 출력의 키와 값의 클래스 타입을 지정한다.
27. conf.setOutputKeyClass(Text.class);
28. conf.setOutputValueClass(IntWritable.class);
29.
30. //매퍼 클래스와 리듀서 클래스를 지정한다.
31. conf.setMapperClass(DistributionMapper.class);
32. conf.setReducerClass(DistributionReducer.class);
33. //맵리듀스를 실행할 입력 데이터와 출력 데이터의 파일 위치를 지정한다.
34. Path inp = new Path(args[0]);
35. Path out = new Path(args[1]);
36.
37. FileInputFormat.addInputPath(conf, inp);
38. FileOutputFormat.setOutputPath(conf, out);
39.
40. JobClient.runJob(conf);
41. return 0;
42. }
43.
44. public static void main(String[] args) throws Exception
45. {
46.     // 실제 실행되는 메인함수.(맵리듀스 클래스(Distribution)를 생성한다.)
47.     int res = ToolRunner.run(new Configuration(), new Distribution(),args);
48.     System.exit(res);
49. }
50.
51.
52. public static void initShopData(Configuration conf)
53. {
54.     if (_listShops == null)
55.     {
56.         _listShops = new ArrayList<String[]>();
57.
58.         Path pt=new Path("/usr/local/dist/shop_master.csv");
59.

```

I. 개요

II. 수집

III. 가공

IV. 저장

V. 분석

VI. 시각화

## V. 분석

```
60.         FileSystem fs;
61.         try {
62.             fs = FileSystem.get(conf);
63.
64.             BufferedReader br=new BufferedReader(new InputStreamReader(
65.                 ↪ fs.open(pt)));
66.             try {
67.                 String line;
68.                 line=br.readLine();
69.                 while (line != null){
70.                     line = br.readLine();
71.                     if (line == null)
72.                         continue;
73.                     String[] arr = line.split(",(?=(["""]*\\\""+")*[^"]*$)");
74.                     if (arr.length < 3)
75.                         continue;
76.
77.                     String shopCode = arr[0];
78.                     String type = arr[2];
79.                     String area = arr[1];
80.
81.                     System.out.println(shopCode + ":" + area + ":" + type);
82.                     _listShops.add(new String[]{shopCode, area});
83.                 }
84.             } finally {
85.                 br.close();
86.             }
87.         } catch (IOException e) {
88.             e.printStackTrace();
89.         }
90.     }
91.     return;
92. }
```

```

94. public static String findArea(String shopCode)
95. {
96.     for (int i=0; i<_listShops.size(); i++)
97.     {
98.         String[] arr = _listShops.get(i);
99.         if (arr[0].contains(shopCode))
100.         {
101.             return arr[1];
102.         }
103.     }
104.     return "";
105. }
106.
107. private static ArrayList<String[]> _listShops = null;
108.
109. public static class DistributionMapper extends MapReduceBase implements
110.     ↪ Mapper<LongWritable, Text, Text, IntWritable>
111. {
112.     public DistributionMapper()
113.     {
114.         super();
115.     }
116.     @Override
117.     public void configure(JobConf job) {
118.         initShopData(job);
119.         super.configure(job);
120.     }
121.     private final IntWritable number = new IntWritable(1);
122.     private Text word = new Text();
123.
124.     public void map(LongWritable key, Text value, OutputCollector<Text, IntW
125.         ↪ ritable> output, Reporter reporter) throws IOException
126.     {
127.         //맵 함수는 한번에 입력 데이터 한 줄에 해당하는 데이터가 키(Key)/값(Value) 한쌍이
128.         ↪ 파라미터로 들어온다.

```

I. 개요

II. 수집

III. 가공

IV. 저장

V. 분석

VI. 시각화

## V. 분석

```
128.         String line = value.toString();
129.         String shopCode = line.substring(0, 6).trim();
130.         String ym = line.substring(6, 10).trim();
131.         String itemCode = line.substring(12, 26).trim();
132.         String amount = line.substring(26, 32).trim();
133.
134.         String area = Distribution.findArea(shopCode);
135.         if (area.equals(""))
136.             return;
137.         // 키값 지정
138.         word.set(area+"\t"+ym);
139.         // 리듀서로 넘길 키값과 데이터를 정의한다.
140.         output.collect(word, new IntWritable(Integer.valueOf(amount)));
141.     }
142. }
143. public static class DistributionReducer extends MapReduceBase implements
144.     Reducer<text, intwritable,="" text,="" intwritable="">
145. {
146.     public DistributionReducer()
147.     {
148.         super();
149.         //reduce 함수는 매퍼로부터 매핑된 키에 해당하는 값의 리스트를 파라미터로 받아
150.         //들인 후, 키에 해당하는 결과 값 한쌍을 output 콜렉션에 저장한다.
151.         public void reduce(Text key, Iterator<intwritable> values, OutputCollecto
152.             r<text, intwritable=""> output, Reporter reporter) throws IOException
153.         {
154.             int sum = 0;
155.             //입력된 키에 해당하는 값들에 대하여 루프를 돌며 합계를 계산한다.
156.             while (values.hasNext())
157.             {
158.                 sum += values.next().get();
159.             }
160.         }
161.     }
```



- 83페이지 맵리듀스 분석 프로그램 소스(Distribution.java)
- **DistributionMapper** 클래스 (라인 143~142)
  - 맵리듀스 과정에서 매핑 기능을 정의한다.
  - map 함수(라인 125~141) : 판매년월을 기준으로 입력데이터를 매핑(묶음)한다.
  - map 함수를 수정하면 통계를 산출할 기준을 설정할 수 있다.
- **DistributionReducer** 클래스 (라인 143~160)
  - 맵리듀스 과정에서 리듀스 기능을 정의한다.
  - reduce 함수(143~160) : 매핑을 통해 기준별로 묶인 데이터들을 합산하는 기능을 수행한다.
  - reduce 함수를 수정하면 합산 이외에 평균, 표준편차 등 원하는 통계값을 산출할 수 있다.

I. 개요

II. 수집

III. 가공

IV. 저장

V. 분석

VI. 시각화

## ➤ 분석 프로그램 실행(03.run.sh)

### 03.run.sh (맵리듀스 실행)

```

01.#!/bin/bash
02.CURRENT_DIR=/home/eduuser/nia_kbig/dist/middle
03.# 컴파일하여 생성할 프로그램(jar) 경로를 지정한다.
04.TARGET_JAR=$CURRENT_DIR/distribution.jar
05.TARGET_SOURCE_DIR=$CURRENT_DIR/src
06.# 컴파일할 소스를 지정한다.
07.TARGET_SOURCE=com/nia/hadoop/*.java
08.TARGET_CLASSES=com/nia/hadoop/*.class
09.EXE_CLASS=com/nia/hadoop/Distribution
10.INPUT_DATA=/user/bigdata/2013*.csv
11.# 맵리듀스로 처리한 결과 데이터파일을 생성할 디렉토리를 지정한다.
12.OUTPUT_DIR=/user/bigdata/dist/out
13.hadoop fs -rm -r $OUTPUT_DIR
14.cd $TARGET_SOURCE_DIR
15.#컴파일에 필요한 하둡 라이브러리 패스와 함께 source를 컴파일한다.
javac -classpath /eduuser/hadoop/share/hadoop/common/lib/hadoop-anno
      ↛ tations-2.2.0.jar:/eduuser/hadoop/share/hadoop/common/lib/hadoop-an
      ↛ notations-2.0.0.jar:/eduuser/hadoop/share/hadoop/mapreduce/hadoop-
      ↛ mapreduce-client-core-2.2.0.jar:/eduuser/hadoop/share/hadoop/comm
      ↛ on/lib/commons-cli-1.2.jar:/eduuser/hadoop/share/hadoop/common/ha
      ↛ doop-common-2.2.0.jar:/eduuser/hadoop/share/hadoop/common/lib/h
      ↛ adoop-annotations-2.0.0.jar $TARGET_SOURCE
16.
17.jar cf $TARGET_JAR $TARGET_CLASSES
18.
19.# yarn 커맨드로 하둡에서 TARGET_JAR 프로그램을 돌려서 맵리듀스를 실행한다.
20.yarn jar $TARGET_JAR $EXE_CLASS $INPUT_DATA $OUTPUT_DIR

```



- 데이터 분석 스크립트 소스(03.run.sh)
- 라인 02 : 현재 작업폴더 위치를 변수(CURRENT\_DIR)로 지정하는 라인이다.
- 라인 04~08 : 하둡 맵리듀스 작업 수행 프로그램 파일(distribution.jar)을 컴파일하기 위한 환경을 지정하는 라인이다.
- 라인 09~10 : 맵리듀스 프로그램 실행 객체의 경로와 입력 데이터 경로를 변수(EXE\_CLASS, INPUT\_DATA)로 저장하는 라인이다.
- 라인12 : 맵리듀스 프로그램 실행 결과 파일을 저장할 경로를 변수(OUTPUT\_DIR)로 저장하는 라인이다.
- 라인 13 : 하둡 맵리듀스 프로그램을 재실행 할 때를 대비하여 이전 결과 데이터를 삭제하는 라인이다.
- 라인 16 : javac 명령을 사용하여 하둡 맵리듀스 작업 수행 프로그램 소스(Distribution.java)를 컴파일하여 작업 수행 프로그램(distribution.jar)을 컴파일하는 라인이다.
- 라인 17 : 하둡 맵리듀스 프로그램을 실행 가능한 작업 위치로 이동하는 라인이다.
- 라인 20 : yarn 명령을 사용하여 하둡 맵리듀스 프로그램을 실행하는 라인이다.

I. 개요

II. 수집

III. 가공

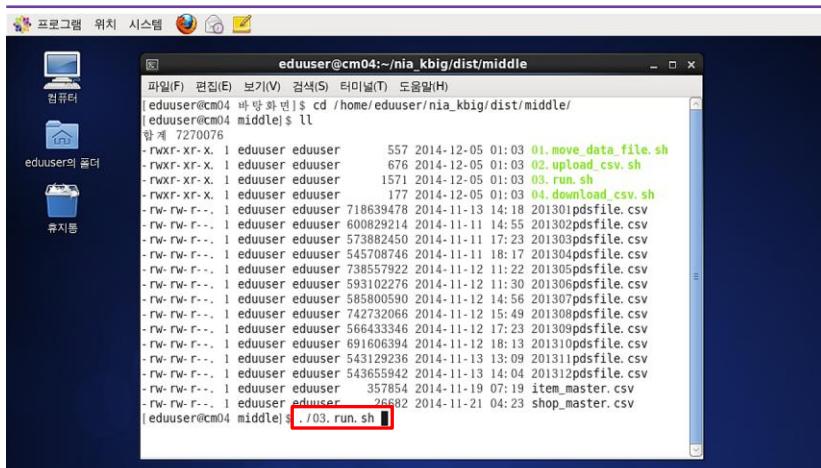
IV. 저장

V. 분석

VI. 시각화

## > 데이터 분석 맵리듀스 실행

### > 분석 맵리듀스 실행



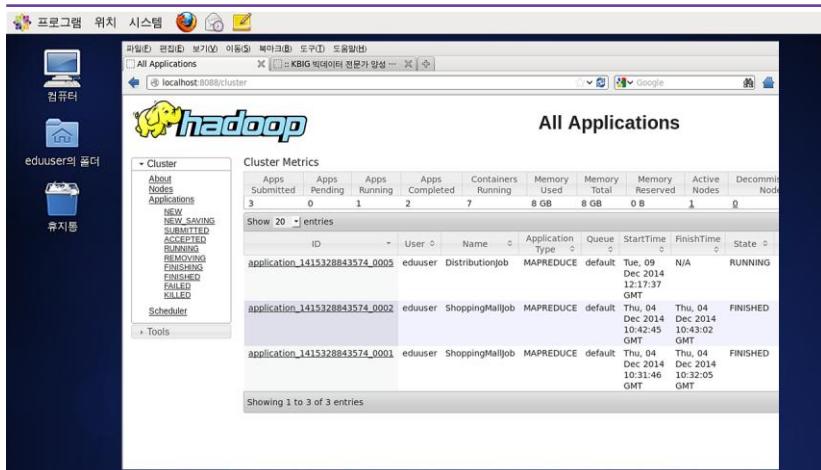
```

파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[eduuser@cm04 바탕 화면]$ cd /home/eduuser/nia_kbig/dist/middle/
[eduuser@cm04 middle]$ ll
합계 7270076
-rwxr-xr-x 1 eduuser eduuser 557 2014-12-05 01:03 01_move_data_file.sh
-rwxr-xr-x 1 eduuser eduuser 676 2014-12-05 01:03 02_upload.csv.sh
-rwxr-xr-x 1 eduuser eduuser 1571 2014-12-05 01:03 03_run.sh
-rwxr-xr-x 1 eduuser eduuser 177 2014-12-05 01:03 04_download_csv.sh
-rw-rw-r-- 1 eduuser eduuser 718639478 2014-11-13 14:18 201301pdffile.csv
-rw-rw-r-- 1 eduuser eduuser 600829214 2014-11-11 14:55 201302pdffile.csv
-rw-rw-r-- 1 eduuser eduuser 573882450 2014-11-11 17:23 201303pdffile.csv
-rw-rw-r-- 1 eduuser eduuser 545708744 2014-11-11 18:17 201304pdffile.csv
-rw-rw-r-- 1 eduuser eduuser 738557924 2014-11-12 11:22 201305pdffile.csv
-rw-rw-r-- 1 eduuser eduuser 593102274 2014-11-12 11:30 201306pdffile.csv
-rw-rw-r-- 1 eduuser eduuser 585800594 2014-11-12 14:56 201307pdffile.csv
-rw-rw-r-- 1 eduuser eduuser 742732066 2014-11-12 15:49 201308pdffile.csv
-rw-rw-r-- 1 eduuser eduuser 566433346 2014-11-12 17:23 201309pdffile.csv
-rw-rw-r-- 1 eduuser eduuser 691606394 2014-11-12 18:13 201310pdffile.csv
-rw-rw-r-- 1 eduuser eduuser 543129236 2014-11-13 13:09 201311pdffile.csv
-rw-rw-r-- 1 eduuser eduuser 543655842 2014-11-13 14:04 201312pdffile.csv
-rw-rw-r-- 1 eduuser eduuser 357854 2014-11-19 07:19 item_master.csv
-rw-rw-r-- 1 eduuser eduuser 26982 2014-11-21 04:23 shop_master.csv
[eduuser@cm04 middle]$ ./03.run.sh

```

■ 하둡의 맵리듀스를 실행하여 데이터를 분석하여 결과 파일을 하둡 파일시스템에 생성한다. ./03.run.sh 입력 후 엔터

### > 맵리듀스 실행 현황 조회



ID	User	Name	Application Type	Queue	StartTime	FinishTime	State
application_1415328843574_0005	eduuser	DistributionJob	MAPREDUCE	default	Tue, 09 Dec 2014 12:17:37 GMT	N/A	RUNNING
application_1415328843574_0002	eduuser	ShoppingMallJob	MAPREDUCE	default	Thu, 04 Dec 2014 10:42:45 GMT	10:43:02 GMT	FINISHED
application_1415328843574_0001	eduuser	ShoppingMallJob	MAPREDUCE	default	Thu, 04 Dec 2014 10:31:46 GMT	10:32:05 GMT	FINISHED

■ 파일어폭스 브라우저를 클릭한 후 주소 입력창에 <http://localhost:8088>을 입력 후 엔터를 치면 맵리듀스 진행과정을 볼 수 있다.

## ▶ 결과 데이터 2차 분석

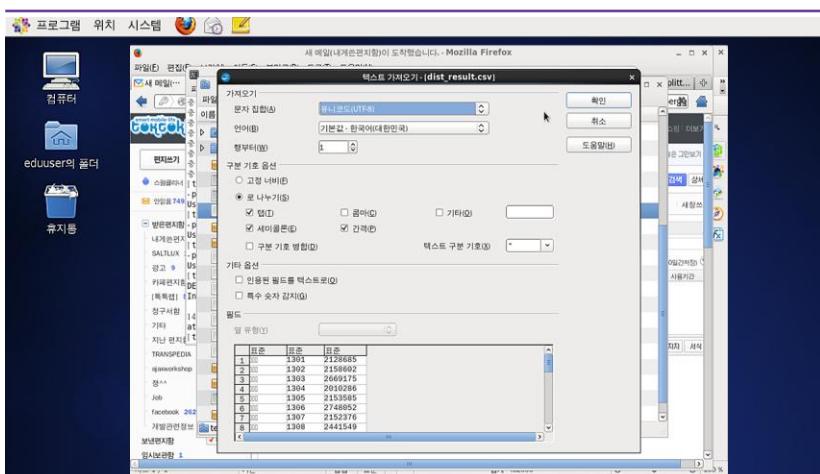
### ▶ 맵리듀스 분석 결과 파일 다운로드(04.download\_csv.sh)

- 터미널 상에서 다운로드 스크립트(04.download\_csv.sh)를 실행한다.

#### 04.download\_csv.sh (결과 파일 다운로드)

```
01. #!/bin/bash
02. # 파일을 하둡의 파일 시스템으로부터 다운로드
03. hadoop fs -get /user/bigdata/dist/out/part-00000 /user/bigdata/dist_result.
    ↳ csv
04.
```

### ▶ 분석 결과 파일 로드



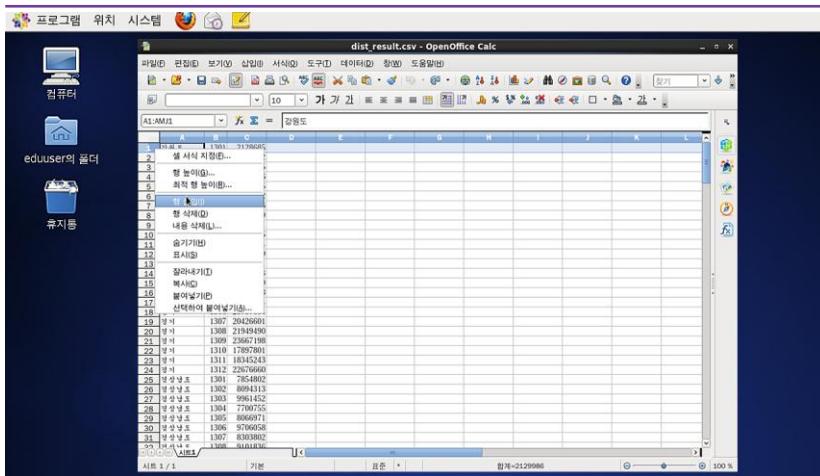
- 분석 결과 데이터(/user/bigdata/dist\_result.csv)를 더블클릭하여 오픈 오피스 스프레드시트로 로드한다.



- 결과 파일 다운로드 스크립트 (04.download\_csv.sh)
- 라인 03 : hadoop fs -get 명령을 이용하여 하둡 파일시스템으로부터 맵리듀스 분석 결과 파일을 다운로드하는 라인이다.

## V. 분석

### ▶ 가공 및 분석



The screenshot shows the OpenOffice Calc interface with a pivot table named '강원도' (Gangwon-do) currently selected. The data source is 'dist\_result.csv'. The pivot table has '지역' (Region) in the rows, '공급 연월' (Supply Month) in the columns, and '공급가액' (Supply Amount) as the value. The first row of data is being edited.

A1	B1	C1	D1	E1	F1	G1	H1	I1	J1	K1
1	1	2	3	4	5	6	7	8	9	10
18	1	19	20	21	22	23	24	25	26	27
28	29	30	31	32	33	34	35	36	37	38
39	40	41	42	43	44	45	46	47	48	49
50	51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70	71
72	73	74	75	76	77	78	79	80	81	82
83	84	85	86	87	88	89	90	91	92	93
94	95	96	97	98	99	100	101	102	103	104
105	106	107	108	109	110	111	112	113	114	115
116	117	118	119	120	121	122	123	124	125	126
127	128	129	130	131	132	133	134	135	136	137
138	139	140	141	142	143	144	145	146	147	148
149	150	151	152	153	154	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169	170
171	172	173	174	175	176	177	178	179	180	181
182	183	184	185	186	187	188	189	190	191	192
193	194	195	196	197	198	199	200	201	202	203
204	205	206	207	208	209	210	211	212	213	214
215	216	217	218	219	220	221	222	223	224	225
226	227	228	229	230	231	232	233	234	235	236
237	238	239	240	241	242	243	244	245	246	247
248	249	250	251	252	253	254	255	256	257	258
259	260	261	262	263	264	265	266	267	268	269
270	271	272	273	274	275	276	277	278	279	280
281	282	283	284	285	286	287	288	289	290	291
292	293	294	295	296	297	298	299	300	301	302
303	304	305	306	307	308	309	310	311	312	313
314	315	316	317	318	319	320	321	322	323	324
325	326	327	328	329	330	331	332	333	334	335
336	337	338	339	340	341	342	343	344	345	346
347	348	349	350	351	352	353	354	355	356	357
358	359	360	361	362	363	364	365	366	367	368
369	370	371	372	373	374	375	376	377	378	379
380	381	382	383	384	385	386	387	388	389	390
391	392	393	394	395	396	397	398	399	400	401
402	403	404	405	406	407	408	409	410	411	412
413	414	415	416	417	418	419	420	421	422	423
424	425	426	427	428	429	430	431	432	433	434
435	436	437	438	439	440	441	442	443	444	445
446	447	448	449	450	451	452	453	454	455	456
457	458	459	460	461	462	463	464	465	466	467
468	469	470	471	472	473	474	475	476	477	478
479	480	481	482	483	484	485	486	487	488	489
490	491	492	493	494	495	496	497	498	499	500
501	502	503	504	505	506	507	508	509	510	511
512	513	514	515	516	517	518	519	520	521	522
523	524	525	526	527	528	529	530	531	532	533
534	535	536	537	538	539	540	541	542	543	544
545	546	547	548	549	550	551	552	553	554	555
556	557	558	559	560	561	562	563	564	565	566
567	568	569	570	571	572	573	574	575	576	577
578	579	580	581	582	583	584	585	586	587	588
589	590	591	592	593	594	595	596	597	598	599
599	600	601	602	603	604	605	606	607	608	609
609	610	611	612	613	614	615	616	617	618	619
619	620	621	622	623	624	625	626	627	628	629
629	630	631	632	633	634	635	636	637	638	639
639	640	641	642	643	644	645	646	647	648	649
649	650	651	652	653	654	655	656	657	658	659
659	660	661	662	663	664	665	666	667	668	669
669	670	671	672	673	674	675	676	677	678	679
679	680	681	682	683	684	685	686	687	688	689
689	690	691	692	693	694	695	696	697	698	699
699	700	701	702	703	704	705	706	707	708	709
709	710	711	712	713	714	715	716	717	718	719
719	720	721	722	723	724	725	726	727	728	729
729	730	731	732	733	734	735	736	737	738	739
739	740	741	742	743	744	745	746	747	748	749
749	750	751	752	753	754	755	756	757	758	759
759	760	761	762	763	764	765	766	767	768	769
769	770	771	772	773	774	775	776	777	778	779
779	780	781	782	783	784	785	786	787	788	789
789	790	791	792	793	794	795	796	797	798	799
799	800	801	802	803	804	805	806	807	808	809
809	810	811	812	813	814	815	816	817	818	819
819	820	821	822	823	824	825	826	827	828	829
829	830	831	832	833	834	835	836	837	838	839
839	840	841	842	843	844	845	846	847	848	849
849	850	851	852	853	854	855	856	857	858	859
859	860	861	862	863	864	865	866	867	868	869
869	870	871	872	873	874	875	876	877	878	879
879	880	881	882	883	884	885	886	887	888	889
889	890	891	892	893	894	895	896	897	898	899
899	900	901	902	903	904	905	906	907	908	909
909	910	911	912	913	914	915	916	917	918	919
919	920	921	922	923	924	925	926	927	928	929
929	930	931	932	933	934	935	936	937	938	939
939	940	941	942	943	944	945	946	947	948	949
949	950	951	952	953	954	955	956	957	958	959
959	960	961	962	963	964	965	966	967	968	969
969	970	971	972	973	974	975	976	977	978	979
979	980	981	982	983	984	985	986	987	988	989
989	990	991	992	993	994	995	996	997	998	999
999	1000	1001	1002	1003	1004	1005	1006	1007	1008	1009
1009	1010	1011	1012	1013	1014	1015	1016	1017	1018	1019
1019	1020	1021	1022	1023	1024	1025	1026	1027	1028	1029
1029	1030	1031	1032	1033	1034	1035	1036	1037	1038	1039
1039	1040	1041	1042	1043	1044	1045	1046	1047	1048	1049
1049	1050	1051	1052	1053	1054	1055	1056	1057	1058	1059
1059	1060	1061	1062	1063	1064	1065	1066	1067	1068	1069
1069	1070	1071	1072	1073	1074	1075	1076	1077	1078	1079
1079	1080	1081	1082	1083	1084	1085	1086	1087	1088	1089
1089	1090	1091	1092	1093	1094	1095	1096	1097	1098	1099
1099	1100	1101	1102	1103	1104	1105	1106	1107	1108	1109
1109	1110	1111	1112	1113	1114	1115	1116	1117	1118	1119
1119	1120	1121	1122	1123	1124	1125	1126	1127	1128	1129
1129	1130	1131	1132	1133	1134	1135	1136	1137	1138	1139
1139	1140	1141	1142	1143	1144	1145	1146	1147	1148	1149
1149	1150	1151	1152	1153	1154	1155	1156	1157	1158	1159
1159	1160	1161	1162	1163	1164	1165	1166	1167	1168	1169
1169	1170	1171	1172	1173	1174	1175	1176	1177	1178	1179
1179	1180	1181	1182	1183	1184	1185	1186	1187	1188	1189
1189	1190	1191	1192	1193	1194	1195	1196	1197	1198	1199
1199	1200	1201	1202	1203	1204	1205	1206	1207	1208	1209
1209	1210	1211	1212	1213	1214	1215	1216	1217	1218	1219
1219	1220	1221	1222	1223	1224	1225	1226	1227	1228	1229
1229	1230	1231	1232	1233	1234	1235	1236	1237	1238	1239
1239	1240	1241	1242	1243	1244	1245	1246	1247	1248	1249
1249	1250	1251	1252	1253	1254	1255	1256	1257	1258	1259
1259	1260	1261	1262	1263	1264	1265	1266	1267	1268	1269
1269	1270	1271	1272	1273	1274	1275	1276	1277	1278	1279
1279	1280	1281	1282	1283	1284	1285	1286	1287	1288	1289
1289	1290	1291	1292	1293	1294	1295	1296	1297	1298	1299
1299	1300	1301	1302	1303	1304	1305	1306	1307	1308	1309
1309	1310	1311	1312	1313	1314	1315	1316			

## I. 개요

## II. 수집

## III. 기공

## IV. 저장

## V. 분석

## VI. 시각화

dist\_result.csv - OpenOffice Calc

파일(F) 편집(E) 보기(V) 삽입(I) 서식(S) 도구(U) 데이터(D) 창(W) 도움말(H)

C1 A B C D

G H I J K

선택 범위 정의(D)... 범위 선택(R)...  
필터(F)  
부분합(D)  
유형설정(U)...  
다중 연산(M)... 텍스트를 열로(O)... 통합(Q)...  
그룹과 개요(G)...  
피벗 테이블(P) 만들기(G)... 세션 관리(B)...  
영역 새로 고침(R)...  
삭제(D)

	A	B	C	D
1	지역	국_교보	국_가계부	
2	광_천_도	1301	2128685	
3	광_천_도	1302	2158602	
4	광_천_도	1303	2669175	
5	광_천_도	1304	2010286	
6	광_천_도	1305	2153585	
7	광_천_도	1306	2748052	
8	광_천_도	1307	2152376	
9	광_천_도	1308	2669149	
10	광_천_도	1309	2635091	
11	광_천_도	1310	1862395	
12	광_천_도	1311	2043254	
13	광_천_도	1312	2435609	
14	경_기	1301	19908771	
15	경_기	1302	19793303	
16	경_기	1303	24798339	
17	경_기	1304	19455178	
18	경_기	1305	19052600	
19	경_기	1306	25757684	
20	경_기	1307	20426601	
21	경_기	1308	21949490	
22	경_기	1309	23667198	
23	경_기	1310	17897801	
24	경_기	1311	18345243	
25	경_기	1312	22676660	
26	경_상_남_도	1301	7854802	
27	경_상_남_도	1302	8094313	
28	경_상_남_도	1303	9961452	
29	경_상_남_도	1304	7700755	
30	경_상_남_도	1305	8066971	
31	경_상_남_도	1306	9706058	
32	경_상_남_도	1307	9303802	

- 메뉴/데이터/피벗테이블/만들기를 클릭하여 피벗테이블 마법사를 실행한다.

dist\_result.csv - OpenOffice Calc

파일(F) 편집(E) 보기(V) 삽입(I) 서식(S) 도구(U) 데이터(D) 창(W) 도움말(H)

A1:C193 A B C D E F G H I J K

원본 선택

선택

현재 선택(C)

OpenOffice에 등록된 데이터 원본(D)

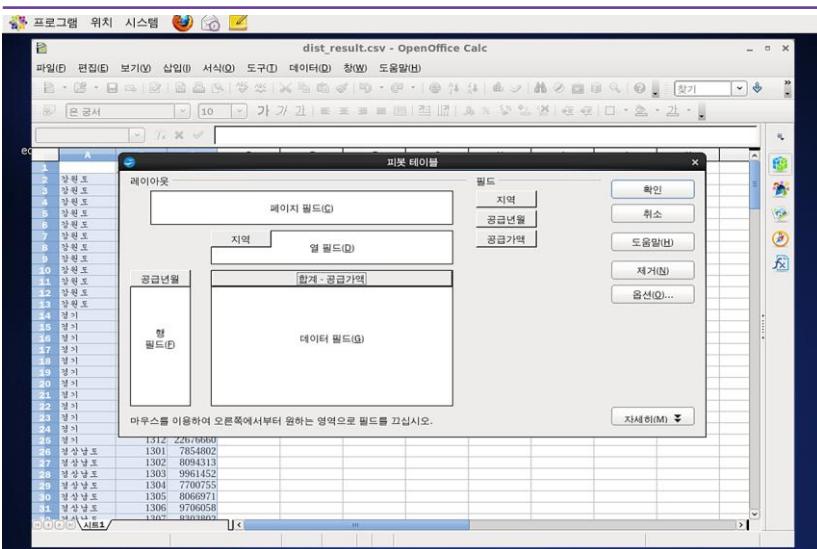
외부 원본/인터페이스(E)

확인 취소 도움말(H)

	A	B	C	D	E	F	G	H	I	J	K
1	지역	국_교보	국_가계부								
2	광_천_도	1301	2128685								
3	광_천_도	1302	2158602								
4	광_천_도	1303	2669175								
5	광_천_도	1304	2010286								
6	광_천_도	1305	2153585								
7	광_천_도	1306	2748052								
8	광_천_도	1307	2152376								
9	광_천_도	1308	244151								
10	광_천_도	1309	2635091								
11	광_천_도	1310	1862395								
12	광_천_도	1311	2043254								
13	광_천_도	1312	2435609								
14	경_기	1301	19908771								
15	경_기	1302	19793303								
16	경_기	1303	24798339								
17	경_기	1304	19455178								
18	경_기	1305	19052600								
19	경_기	1306	25757684								
20	경_기	1307	20426601								
21	경_기	1308	21949490								
22	경_기	1309	23667198								
23	경_기	1310	17897801								
24	경_기	1311	18345243								
25	경_기	1312	22676660								
26	경_상_남_도	1301	7854802								
27	경_상_남_도	1302	8094313								
28	경_상_남_도	1303	9961452								
29	경_상_남_도	1304	7700755								
30	경_상_남_도	1305	8066971								
31	경_상_남_도	1306	9706058								
32	경_상_남_도	1307	9303802								

- 자동적으로 범위가 선택되므로 '현재 선택' 상태에서 그대로 확인을 누른다.

## V. 분석



- 필드 영역의 아이템을 드래그하여 위 화면과 같이 구성한다.  
'데이터 필드'는 같은 '행필드값'과 '열필드값'에 대해 표현할 통계 데이터 수식을 나타낸다. (여기서는 공급금액에 대한 합계를 선택한다.)  
같은 '행 필드'와 '열 필드'를 가지는 '데이터 필드' 항목의 합계를 계산한다.

I. 개요

II. 수집

III. 가공

IV. 저장

V.분석

VI. γ | 각도

■ 화면의 최하단까지 스크롤을 하면 위와 같이 피봇테이블이 집계되어 있는 것을 볼 수 있다. (시각화를 위한 데이터 분석 애료)

■ 다른 이름으로 저장하여 2차 분석 데이터를 저장한다.

(파일명 : dist\_result\_analysis.ods)



1

2



## VI 시각화

개요	99
분석 데이터 시각화	101
데이터 분석 1	103
데이터 분석 2	106

# VI

## 시각화

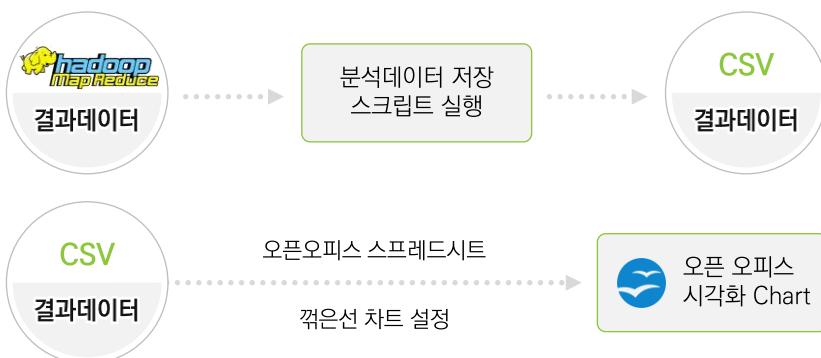
### > 개요

유통 데이터의 시각화 과정에서는, 분석 과정에서 하둡 맵리듀스를 통해 작성된 2013년 지역별 공급 금액 통계 데이터를 오픈 오피스 스프레드 시트의 차트 기능을 활용하여 꺾은선 그래프로 시각화하여, 지역별 매출 규모 및 월간 공급량 패턴 등을 파악한다. 이후, 유통 공급 규모별로 대표 지역을 선별하여 비교 분석한다.

### > 시각화 방법 및 활용기술

- 오픈 오피스 스프레드 시트의 차트 기능을 활용하여 멀티 컬럼 데이터를 하나의 차트로 표현한다.
- 3개 년도의 매출 패턴을 한눈에 확인하기 위하여 오픈오피스 스프레드시트의 꺾은선 차트를 활용하여 시각화한다.

### > 시각화 절차



## ▶ 시각화 과정

- 하둡 분석 결과 파일을 로컬의 작업폴더로 다운로드한다.
- 결과 파일을 오픈오피스 스프레드 시트에서 로드한다.
- 데이터의 헤더(제목) 부분을 입력한다.
- 시각화 할 데이터 영역을 지정한다.
- 파일/삽입/차트 를 선택하여 차트 설정을 연다.
- X축과 Y축에 표현 될 값의 범위를 지정한다.
- 완료하여 표현된 차트를 확인한다.

## VI. 시각화

## > 분석 데이터 시각화

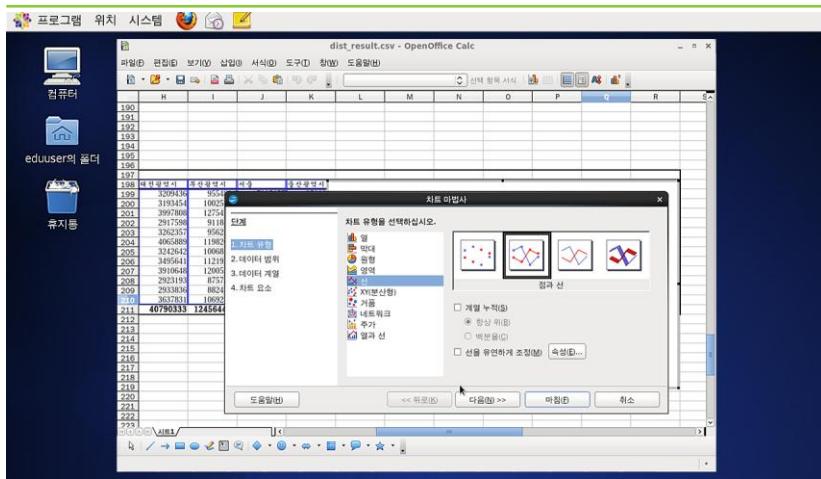
## > 결과 파일 로드

- 2차 분석한 결과 파일을 로드한다.(./user/bigdata/dist result analysis.ods)

## > 차트 생성

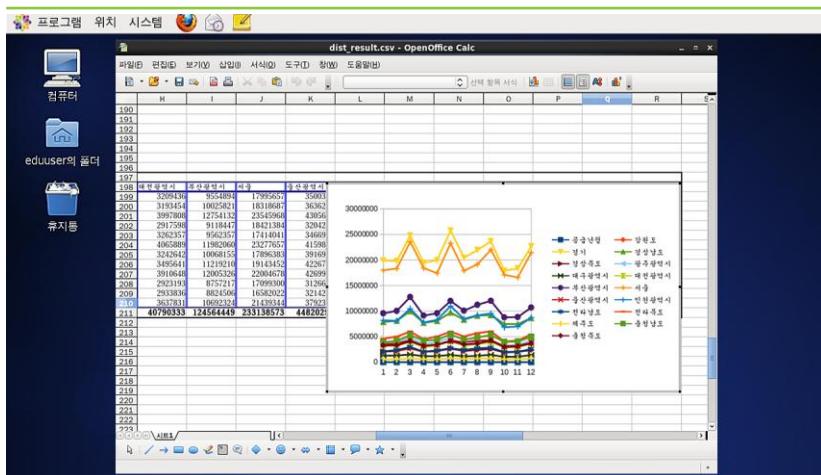
- 시각화할 데이터 범위를 지정하고 메뉴/삽입/차트를 클릭하여 차트 설정 화면을 오픈한다

## > 시각화 차트 설정



- 차트 유형에서 선을 선택하고, 우측에서 점과 선을 선택한 후 마침을 누른다.

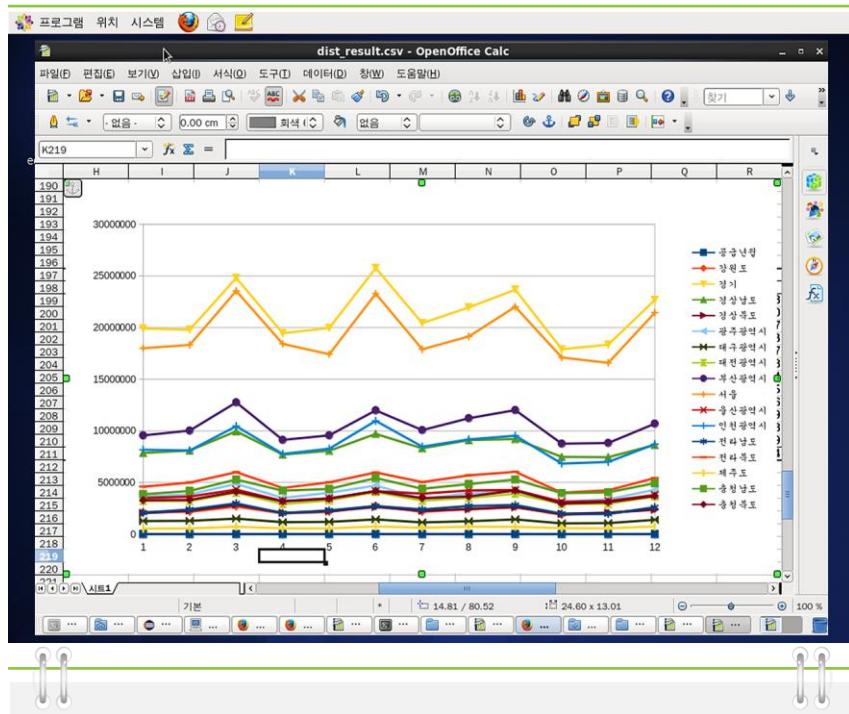
## > 차트 확인



- 꺾은선 차트로 시각화된 결과를 확인한다.

## ▶ 데이터 분석 1

### ▶ 결과 분석



- 1년간 지역별 유통 공급량 패턴을 보면, 지역에 상관없이 연간 소비 패턴의 변화가 비슷함을 알 수 있다.
- 지역별 소매점 물품 유통량 수준은 경기-서울-부산 순으로, 인구/경제 규모와 일치한다.
- 3, 6, 9, 12월 등 분기별로 3개월 주기로 공급량이 점프하는 패턴이 발생한다. (타 데이터와의 연관성 분석을 통해 원인을 파악해 볼 필요가 있다.)
- 추가적으로, 전체 지역의 매출 규모를 보면 크게 세 부류로 군집을 이루어 나타나고 있다. 각 군별로 각각 서울, 부산, 대구 지역을 대표로 따로 통계를 산출하여 시각화 하여 살펴보자 한다.

## ▶ 통계 산출 대상 지역 선별

선택한 항목만 표시

확인 취소

- 분석 결과로 나온 피봇 테이블에서 지역 필터를 클릭하여 서울, 부산, 대구광역시를 선택하고 확인을 누른다.

## ▶ 데이터 확인

선택한 항목만 표시

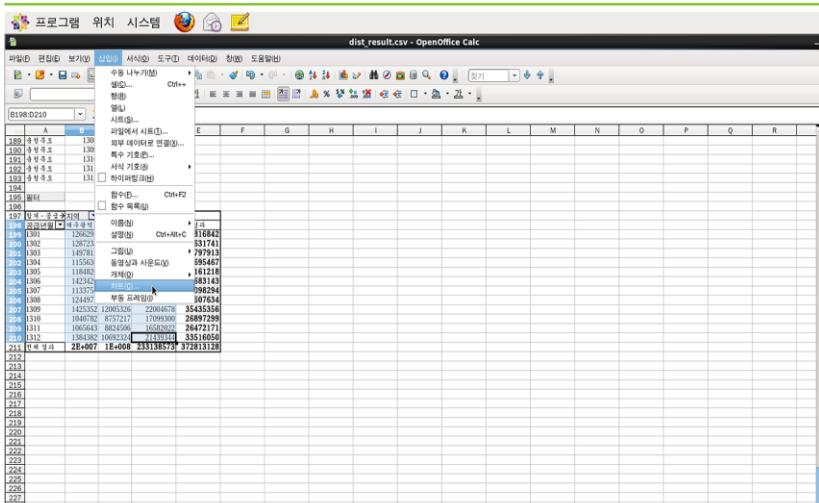
선택한 항목만 표시

선택한 항목만 표시

- 꺾은선 차트로 시각화된 결과를 확인한다.

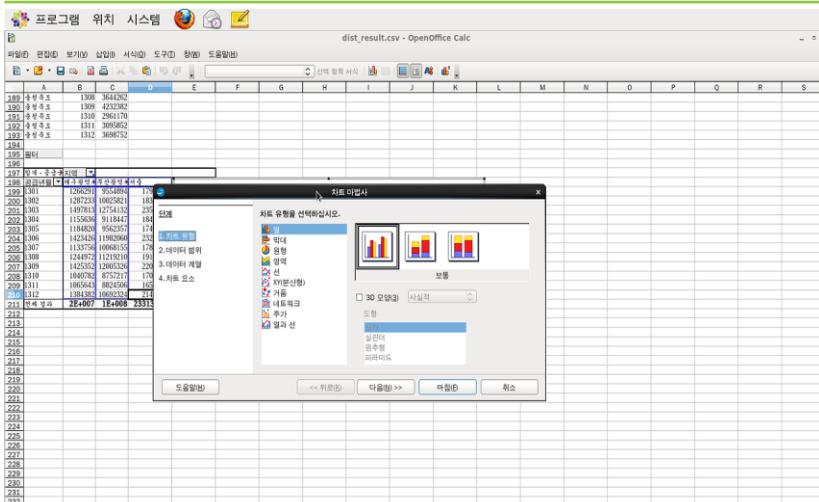
## VI. 시각화

### ▶ 차트 생성



▪ 시각화할 데이터 범위를 지정하고 메뉴/삽입/차트를 클릭하여 차트 설정 화면을  
오픈한다.

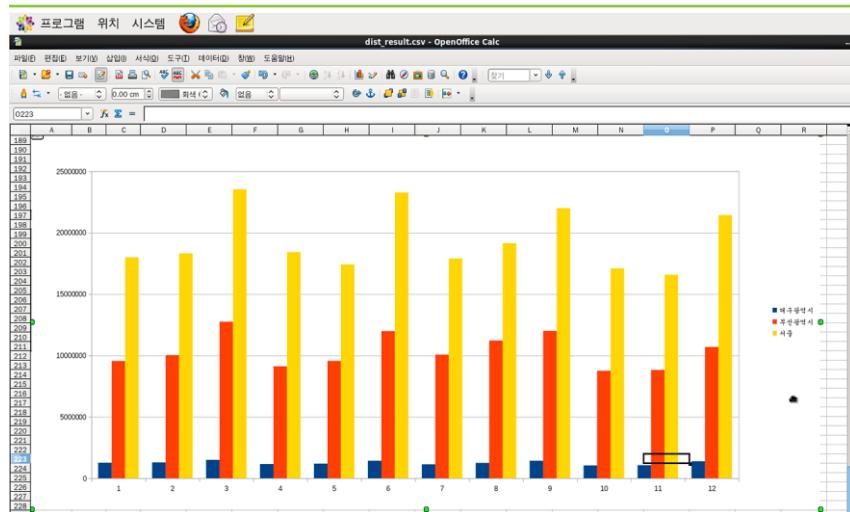
### ▶ 시각화 차트 설정



▪ 차트 유형에서 열을 선택하고, 마침을 누른다.

## > 데이터 분석 2

### > 결과 분석



- 서울과 부산 지역 데이터를 비교해 보면, 계절별 패턴도 거의 동일하며, 월별로 부산지역 공급금액이 서울지역 공급금액의 1/2 이상이 유지됨을 알 수 있다.
- 특이할 점은, 인구가 서울의 1/3 정도밖에 안되는 부산의 주요 공급처별 공급 금액이 1/2 이상으로 나타난다는 점인데, 본 데이터만으로 판단하자면 의외로 부산지역의 평균 소비 성향이 서울지역보다 높다는 것을 알 수 있다.
- 대구지역의 경우에도 인구비율에 비해서는 매우 적은 수치의 유통 공급 수준으로 나타난다.
- 종합하여 볼 때, 좀 더 의미있는 결과를 도출하기 위해서는 서울, 부산, 대구 지역의 인구, 가계 소득 등 소비에 영향을 주는 다양 한 데이터들과의 매쉬업을 통해 상세 분석을 진행할 필요가 있음을 알 수 있다.



## VII 예제문제

예제 문제1

109

예제 문제2

110

## 예 / 제 / 문 / 제

### 예제 1

월별 매장별, 아이템별 공급량 정보로부터 분기별 최고  
매출 매장을 유추하라.

- 월별 아이템별 공급량 정보를 취합하여 매장별 분기별 매출을 계산하여 최고 매출 매장을 유추하라.

- 월별로 저장된 아이템별 공급금액 정보를 취합한다.
- 맵리듀스를 통해 매장별, 분기별 공급금액을 합산한다.
- 분기별로 공급금액을 기준으로 내림차순 정렬한다.
- 분기별 최고 매출 매장을 선정한다.

## 예제 2

2013년 유통 공급 정보를 활용하여 가장 많이 공급된 아이템의 지역별 월간 공급량 추이를 분석하라.

- 2013년 유통 공급 정보를 모두 취합하여 가장 많이 공급된 아이템을 추출하고, 2차 분석을 통해 해당 아이템의 지역별 월간 공급량 추이를 분석하라.

- 2013년도 전체 데이터로부터 아이템별 총 공급량을 구한다.
- 공급량 최고인 아이템을 선정한다.
- 공급량 최고인 아이템에 대하여 지역별, 월간 공급량을 계산한다.
- 대표 공급 아이템의 지역별 월간 공급량 추이를 알아본다.

## **데이터 분석 콘텐츠 활용 매뉴얼**

---

2014년 12월 인쇄

2015년 1월 발행

**발 행 처** 한국정보화진흥원 빅데이터전략센터

**집    필** 신신애, 김성현, 박재원, 김현태, 김지홍, 정다운,  
이승하, 신은비

**주    소** 서울시 중구 청계천로 14

**연 락 처** (02) 2131-0114

**인    쇄** HNJ Printing

---

〈비매품〉



[ 데 이 터      분 석      콘 텐 츠 ]  
**활용 매뉴얼**

**NIA**  한국정보화진흥원

(100-775) 서울시 종구 청계천로 14 한국정보화진흥원  
TEL 02-2131-0114 FAX 02-2131-0109  
[www.nia.or.kr](http://www.nia.or.kr)

