

## Elastic Stack 을 활용한 Data Dashboard 만들기

Week 3 - 데이터를 검색/필터링 해보자



Fast Campus

내용	페이지
Kibana 기타	
Scripted Field	4
Data Format	27
Object Import/Export	52
Dashboard	63
데이터 검색	
Filter	80
Lucene Query Syntax	99
Discover	119

Scripted Field  

**String Concatenation 연산은 안되나?**

**Field 간 연산은 안되나?**

**Date Field에서 연/월/일/시/분/초 등 접근 안되나?**

**기존 Field에 조건을 적용해서 새로운 Field를 만들 수 없나?**

**String Concatenation 연산은 안되나?**

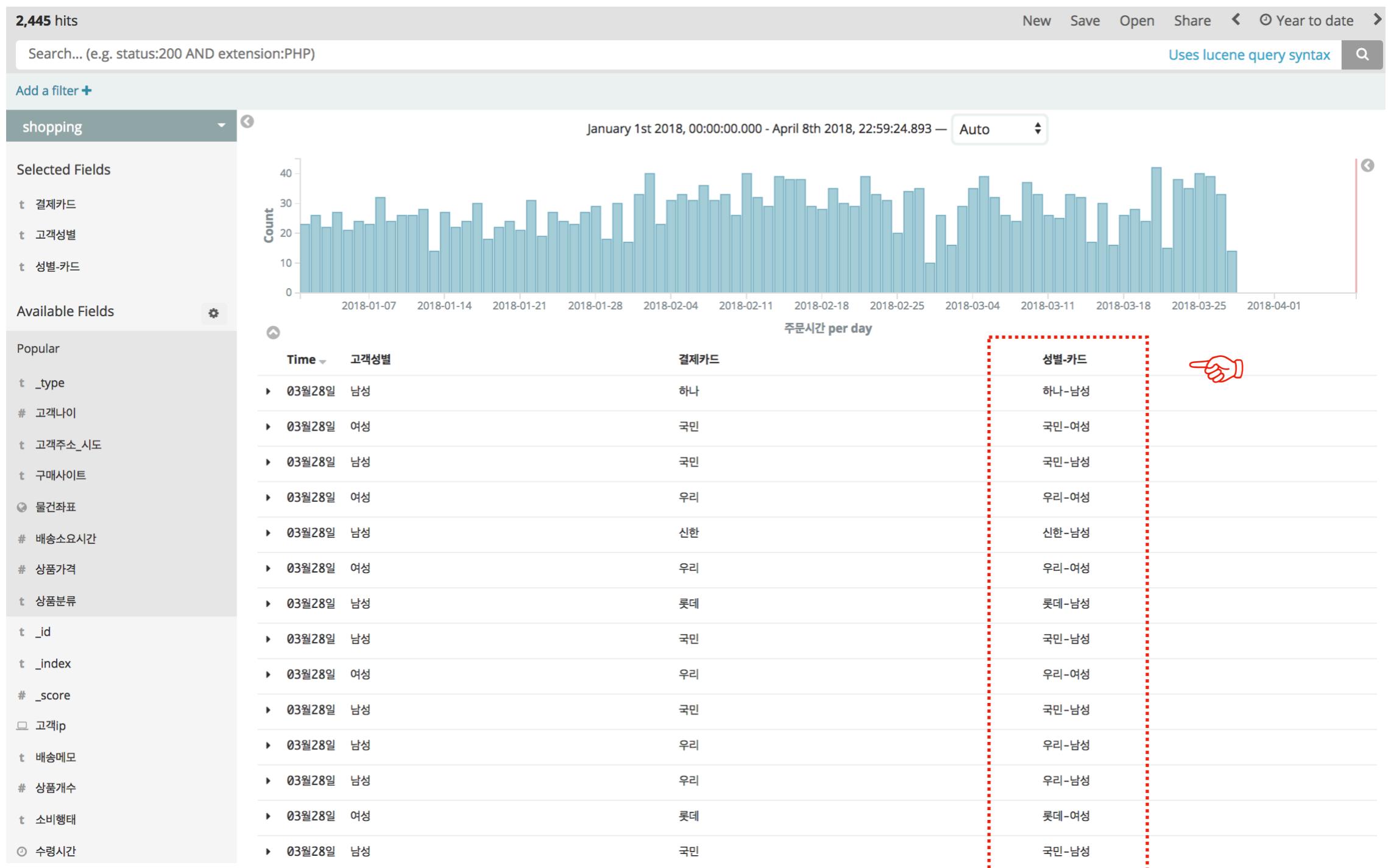


**Field 간 연산은 안되나?**

**Date Field에서 연/월/일/시/분/초 등 접근 안되나?**

**기존 Field에 조건을 적용해서 새로운 Field를 만들 수 없나?**

# 특정한 두 개 혹은 그 이상의 Field를 합쳐서 하나의 Field를 만들고 싶으면 어떻게 해야할까?



# scripted field를 추가하자

The screenshot shows the Kibana Management interface for the 'shopping' index pattern. A red dashed box highlights the 'Management' tab in the sidebar. A red hand icon labeled '1. 선택' points to the 'Management' tab. Another red hand icon labeled '2. 선택' points to the 'Index Patterns' tab, which is also highlighted with a red dashed box. A third red hand icon labeled '3. {id}\_2018.03.01' points to the 'scripted fields (7)' tab, which is also highlighted with a red dashed box. A fourth red hand icon labeled '4. 선택' points to the 'Add Scripted Field' button. A fifth red hand icon labeled '5. 선택' points to the 'scripted fields (7)' tab again.

Management / Kibana

Index Patterns Saved Objects Advanced Settings

+ Create Index Pattern

★ shopping

Time Filter field name: 주문시간

fields (24) scripted fields (7) source filters (0)

Filter All languages ▾

Scripted fields

These scripted fields are computed on the fly from your data. They can be used in visualizations and displayed in your documents, however they can not be searched. You can manage them here and add new ones as you see fit, but be careful, scripts can be tricky!

+ Add Scripted Field

name lang script

format controls

Scroll to top Page Size 25

Collapse

# script를 작성하자

Management / Kibana / Indices / shopping

Index Patterns Saved Objects Advanced Settings

+ Create Index Pattern

★ shopping

Create Scripted Field

실습시에는 {id}\_2018.03.01

Name   1. Field 이름 임의로 생성

Language

Type   2. String 선택

Format (Default: String)   3. String 선택

Transform

Samples

Input	Formatted
A Quick Brown Fox.	A Quick Brown Fox.
STAY CALM!	STAY CALM!
com.organizations.project.ClassName	com.organizations.project.ClassName
hostname.net	hostname.net
SGVsbG8gd29ybGQ=	SGVsbG8gd29ybGQ=

Popularity  + -

Script   4. 다음과 같이 script 작성

doc['고객성별'].value + '-' + doc['결제카드'].value

## Discover에 가서 확인하자

2,445 hits

New Save Open Share < ⏪ ⏩ Year to date >

Search... (e.g. status:200 AND extension:PHP)

Add a filter +

Selected Fields

- t 결제카드
- t 고객성별
- t 성별-카드

Available Fields

- Popular
- t \_type
- # 고객나이
- t 고객주소\_시도
- t 구매사이트
- ⌚ 물건좌표
- # 배송소요시간
- # 상품가격
- t 상품분류
- t \_id
- t \_index
- # \_score
- ⌚ 고객ip
- t 배송메모
- # 상품개수
- t 소비행태
- ⌚ 수령시간

January 1st 2018, 00:00:00.000 - April 8th 2018, 22:59:24.893 — Auto

Count

주문시간 per day

Date	Count
2018-01-01	25
2018-01-03	23
2018-01-05	28
2018-01-07	32
2018-01-09	24
2018-01-11	26
2018-01-13	25
2018-01-15	30
2018-01-17	18
2018-01-19	23
2018-01-21	20
2018-01-23	28
2018-01-25	25
2018-01-27	18
2018-01-29	22
2018-01-31	35
2018-02-02	32
2018-02-04	34
2018-02-06	38
2018-02-08	35
2018-02-10	40
2018-02-12	32
2018-02-14	38
2018-02-16	35
2018-02-18	33
2018-02-20	30
2018-02-22	28
2018-02-24	35
2018-02-26	33
2018-02-28	20
2018-03-01	28
2018-03-03	38
2018-03-05	25
2018-03-07	35
2018-03-09	30
2018-03-11	33
2018-03-13	18
2018-03-15	28
2018-03-17	35
2018-03-19	38
2018-03-21	35
2018-03-23	15
2018-03-25	35
2018-03-27	30
2018-03-29	38
2018-03-31	35
2018-04-01	30

Time ▾ 고객성별 결제카드

Date	Gender	Card Type
2018-03-28	남성	하나
2018-03-28	여성	국민
2018-03-28	남성	국민
2018-03-28	여성	우리
2018-03-28	남성	신한
2018-03-28	여성	우리
2018-03-28	남성	롯데
2018-03-28	남성	국민
2018-03-28	여성	우리
2018-03-28	남성	국민
2018-03-28	여성	우리
2018-03-28	남성	우리
2018-03-28	남성	우리
2018-03-28	여성	롯데
2018-03-28	남성	국민

성별-카드

- 하나-남성
- 국민-여성
- 국민-남성
- 우리-여성
- 신한-남성
- 우리-여성
- 롯데-남성
- 국민-남성
- 우리-여성
- 국민-남성
- 우리-남성
- 우리-남성
- 롯데-여성
- 국민-남성

⌚ 손

# 이번에는 아래와 같은 Field를 생성해보자 ✎ 🤴

2,445 hits      New Save Open Share < ⏪ Year to date ⏩

Search... (e.g. status:200 AND extension:PHP)      Uses lucene query syntax

Add a filter +

**shopping** **{id}\_2018.03.01 선택**

January 1st 2018, 00:00:00.000 - April 8th 2018, 23:40:13.681 — Auto

Selected Fields

- t 결제카드
- t 결제카드이름

Available Fields

Popular

- Time ▾ 결제카드
  - ▶ 03월28일 하나
  - ▶ 03월28일 국민
  - ▶ 03월28일 국민
  - ▶ 03월28일 우리
  - ▶ 03월28일 신한
  - ▶ 03월28일 우리
  - ▶ 03월28일 롯데
  - ▶ 03월28일 국민
  - ▶ 03월28일 우리
  - ▶ 03월28일 국민
  - ▶ 03월28일 우리
  - ▶ 03월28일 우리
  - ▶ 03월28일 롯데
  - ▶ 03월28일 국민
- t \_id
- t \_index
- # \_score
- 고객ip
- t 배송메모
- # 상품개수
- t 성별-카드
- t 소비행태

Count

주문시간 per day

- 결제카드이름
- 하나카드
- 국민카드
- 국민카드
- 우리카드
- 신한카드
- 우리카드
- 롯데카드
- 국민카드
- 우리카드
- 국민카드
- 우리카드
- 우리카드
- 롯데카드
- 국민카드

String Concatenation 연산은 안되나?

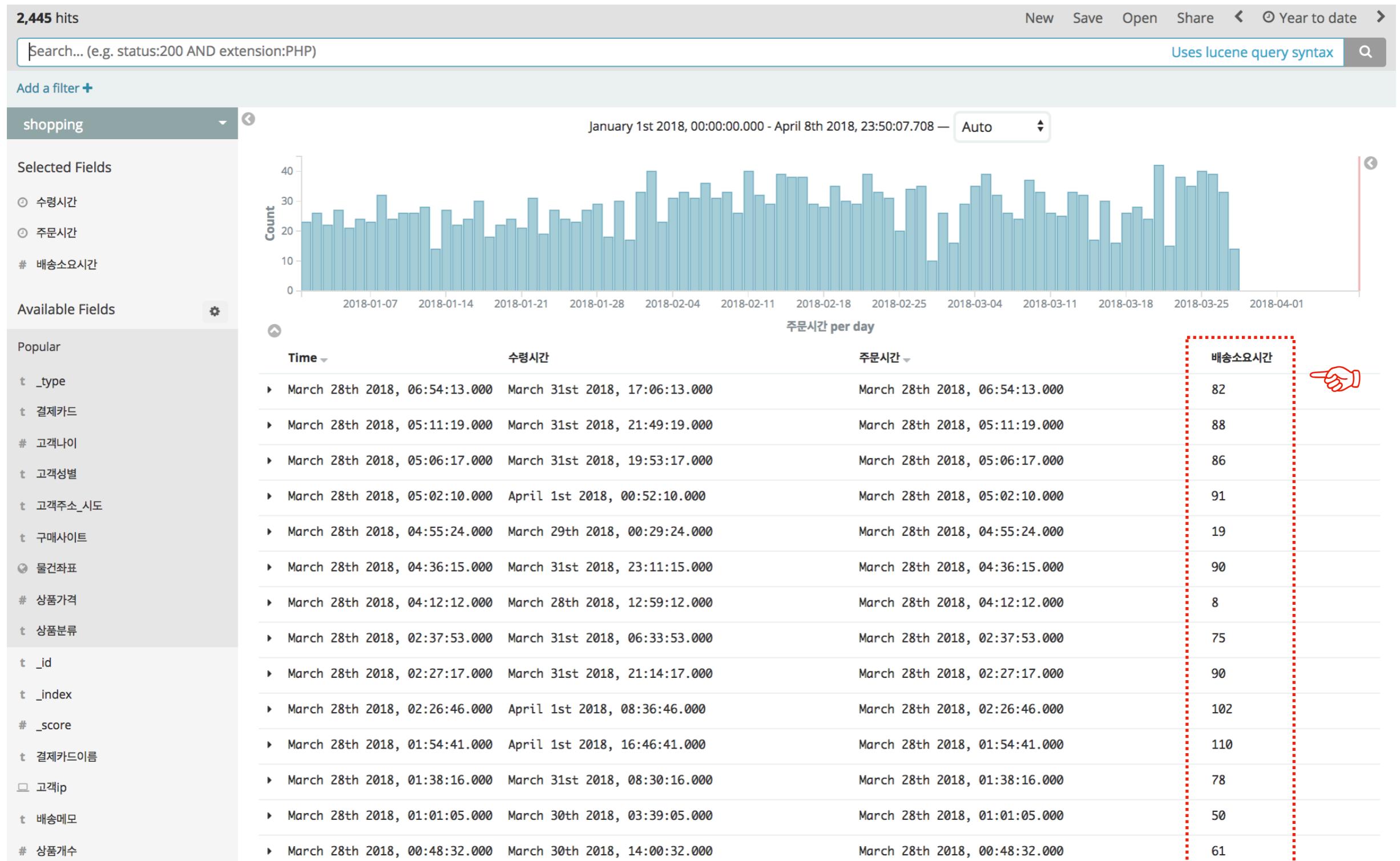
Field 간 연산은 안되나?



Date Field에서 연/월/일/시/분/초 등 접근 안되나?

기존 Field에 조건을 적용해서 새로운 Field를 만들 수 없나?

## 2개의 Date Field의 차이를 통해 특정한 event의 소요시간을 구하고 싶다면?



# scripted field를 추가하자

The screenshot shows the Kibana Management interface for the 'Index Patterns' section. A pink hand icon points to the 'Management' button in the sidebar, labeled '1. 선택'. Another pink hand icon points to the 'shopping' index pattern in the list, labeled '2. 선택'. A third pink hand icon points to the 'scripted fields (7)' tab, labeled '3. {id}\_2018.03.01'. A fourth pink hand icon points to the '+ Add Scripted Field' button, labeled '4. 선택'. A fifth pink hand icon points to the 'Add Scripted Field' form, labeled '5. 선택'.

Management / Kibana

Index Patterns Saved Objects Advanced Settings

+ Create Index Pattern

★ shopping

Time Filter field name: 주문시간

fields (24) scripted fields (7) source filters (0)

Filter All languages ▾

Scripted fields

These scripted fields are computed on the fly from your data. They can be used in visualizations and displayed in your documents, however they can not be searched. You can manage them here and add new ones as you see fit, but be careful, scripts can be tricky!

+ Add Scripted Field

name lang script

format controls

Scroll to top Page Size 25

Collapse

# script를 작성하자

Management / Kibana / Indices / shopping

Index Patterns Saved Objects Advanced Settings

+ Create Index Pattern

★ shopping

실습시에는 {id}\_2018.03.01

★ shopping

Create Scripted Field

Name

배송소요시간-



1. Field 이름 임의로 생성

Language

painless

Type

number



2. Number 선택

Format (Default: Number)

Number



3. Number 선택

Numeral.js format pattern (Default: "0,0.[0]"")

0,0.[0]



4. 적당한 Format 입력

Samples

Input

10000

Formatted

10,000

12.345678

12.3

-1

-1

-999

-999

0.52

0.5

Popularity

0



Script

(doc['수령시간'].value-doc['주문시간'].value)/1000/60/60

5. 다음과 같이 script 작성

(doc['수령시간'].value-doc['주문시간'].value)/1000/60/60

## Discover에 가서 확인하자

2,445 hits

New Save Open Share < ⏪ Year to date ⏩

Search... (e.g. status:200 AND extension:PHP) Uses lucene query syntax

Add a filter +

**shopping**  **{id}\_2018.03.01**

January 1st 2018, 00:00:00.000 - April 8th 2018, 23:50:07.708 — Auto

Selected Fields

- 수령시간
- 주문시간
- # 배송소요시간

Available Fields 

Popular

Time	수령시간	주문시간	배송소요시간
▶ March 28th 2018, 06:54:13.000	March 31st 2018, 17:06:13.000	March 28th 2018, 06:54:13.000	82
▶ March 28th 2018, 05:11:19.000	March 31st 2018, 21:49:19.000	March 28th 2018, 05:11:19.000	88
▶ March 28th 2018, 05:06:17.000	March 31st 2018, 19:53:17.000	March 28th 2018, 05:06:17.000	86
▶ March 28th 2018, 05:02:10.000	April 1st 2018, 00:52:10.000	March 28th 2018, 05:02:10.000	91
▶ March 28th 2018, 04:55:24.000	March 29th 2018, 00:29:24.000	March 28th 2018, 04:55:24.000	19
▶ March 28th 2018, 04:36:15.000	March 31st 2018, 23:11:15.000	March 28th 2018, 04:36:15.000	90
▶ March 28th 2018, 04:12:12.000	March 28th 2018, 12:59:12.000	March 28th 2018, 04:12:12.000	8
▶ March 28th 2018, 02:37:53.000	March 31st 2018, 06:33:53.000	March 28th 2018, 02:37:53.000	75
▶ March 28th 2018, 02:27:17.000	March 31st 2018, 21:14:17.000	March 28th 2018, 02:27:17.000	90
▶ March 28th 2018, 02:26:46.000	April 1st 2018, 08:36:46.000	March 28th 2018, 02:26:46.000	102
▶ March 28th 2018, 01:54:41.000	April 1st 2018, 16:46:41.000	March 28th 2018, 01:54:41.000	110
▶ March 28th 2018, 01:38:16.000	March 31st 2018, 08:30:16.000	March 28th 2018, 01:38:16.000	78
▶ March 28th 2018, 01:01:05.000	March 30th 2018, 03:39:05.000	March 28th 2018, 01:01:05.000	50
▶ March 28th 2018, 00:48:32.000	March 30th 2018, 14:00:32.000	March 28th 2018, 00:48:32.000	61



# 이번에는 아래와 같은 Field를 생성해보자 ✎ 🤴

2,445 hits      New Save Open Share < ⏪ Year to date ⏩

Search... (e.g. status:200 AND extension:PHP)      Uses lucene query syntax

Add a filter +

shopping  {id}\_2018.03.01 선택

January 1st 2018, 00:00:00.000 - April 9th 2018, 00:25:56.680 — Auto

Selected Fields

- # 상품가격
- # 상품개수
- # 총액

Available Fields

Popular

Time	상품가격	상품개수	총액
▶ March 28th 2018, 06:54:13.000	20,000	1	20,000
▶ March 28th 2018, 05:11:19.000	28,000	7	196,000
▶ March 28th 2018, 05:06:17.000	23,000	7	161,000
▶ March 28th 2018, 05:02:10.000	17,000	1	17,000
▶ March 28th 2018, 04:55:24.000	13,000	7	91,000
▶ March 28th 2018, 04:36:15.000	7,000	1	7,000
▶ March 28th 2018, 04:12:12.000	23,000	7	161,000
▶ March 28th 2018, 02:37:53.000	8,000	7	56,000
▶ March 28th 2018, 02:27:17.000	6,000	7	42,000
▶ March 28th 2018, 02:26:46.000	19,000	7	133,000
▶ March 28th 2018, 01:54:41.000	26,000	1	26,000
▶ March 28th 2018, 01:38:16.000	12,000	1	12,000
▶ March 28th 2018, 01:01:05.000	9,000	7	63,000
▶ March 28th 2018, 00:48:32.000	17,000	1	17,000

String Concatenation 연산은 안되나?

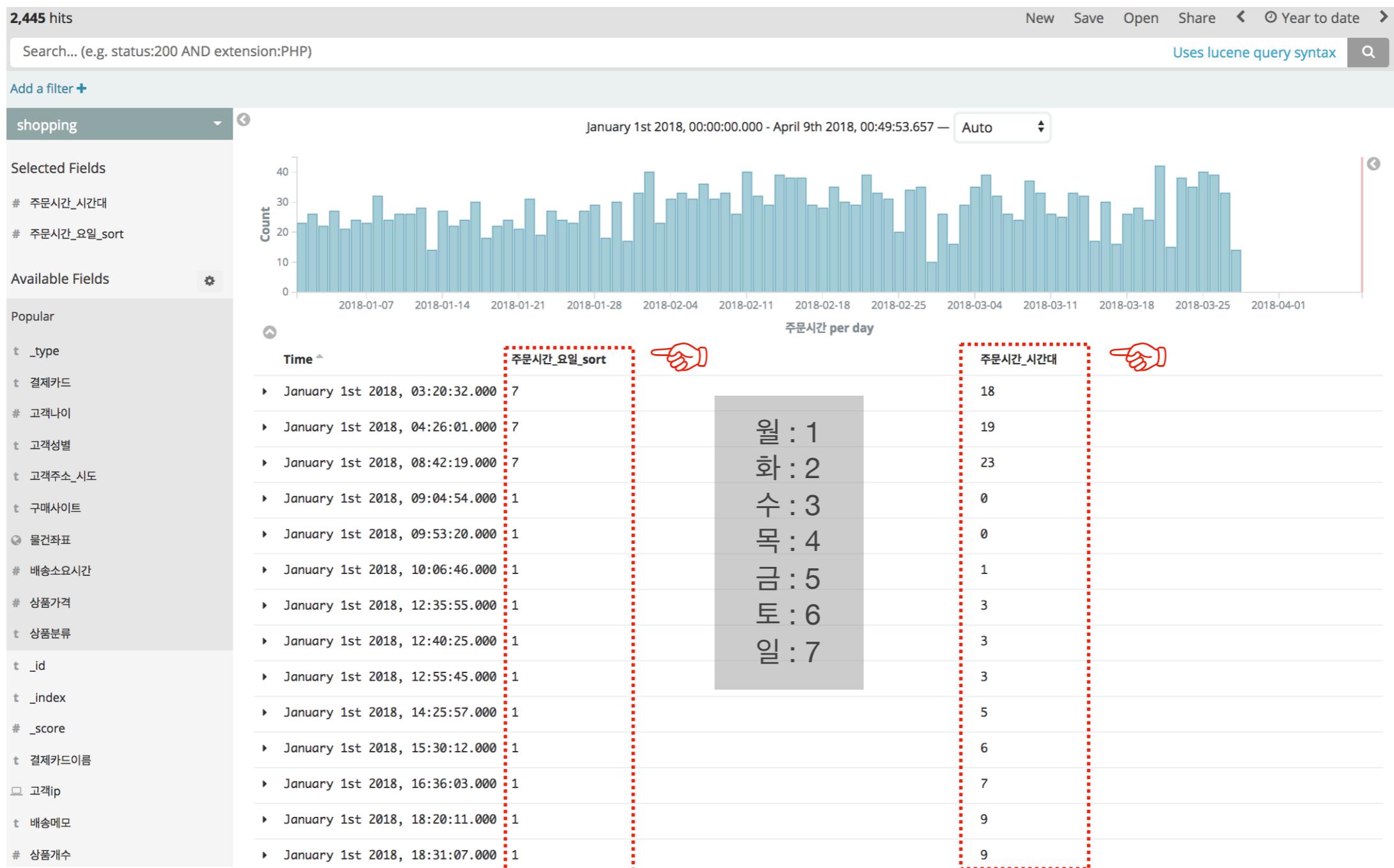
Field 간 연산은 안되나?

Date Field에서 연/월/일/시/분/초 등 접근 안되나?



기존 Field에 조건을 적용해서 새로운 Field를 만들 수 없나?

# Date Field에서 요일, 시간대 등을 뽑고 싶으면?



"Time"에 보이는 시간과 "주문시간\_요일\_sort" 및 "주문시간\_시간대" field의 값이 일치하지 않아 보이는 것은 설정문제다.  
현재 설정은 "elasticsearch에 저장된 값은 UTC 기준이며 Kibana에 표시할 때는 한국 시간"으로 보여주게 해놨기에 그렇다.  
예를 들어, "January 1st 2018, 03:20:32.000"은 실제로 "2017-12-31T18:20:32"로 저장되어 있다.

## 아래를 참고해서 “주문시간”의 요일과 시간대를 추출하자 ✎ 🎉

Expression	Description
<code>doc['field_name'].date.centuryOfEra</code>	Century (1-2920000)
<code>doc['field_name'].date.dayOfMonth</code>	Day (1-31), e.g. 1 for the first of the month.
<code>doc['field_name'].date.dayOfWeek</code>	Day of the week (1-7), e.g. 1 for Monday.
<code>doc['field_name'].date.dayOfYear</code>	Day of the year, e.g. 1 for January 1.
<code>doc['field_name'].date.era</code>	Era: 0 for BC, 1 for AD.
<code>doc['field_name'].date.hourOfDay</code>	Hour (0-23).
<code>doc['field_name'].date.millisOfDay</code>	Milliseconds within the day (0-86399999).
<code>doc['field_name'].date.millisOfSecond</code>	Milliseconds within the second (0-999).
<code>doc['field_name'].date.minuteOfDay</code>	Minute within the day (0-1439).
<code>doc['field_name'].date.minuteOfHour</code>	Minute within the hour (0-59).
<code>doc['field_name'].date.monthOfYear</code>	Month within the year (1-12), e.g. 1 for January.
<code>doc['field_name'].date.secondOfDay</code>	Second within the day (0-86399).
<code>doc['field_name'].date.secondOfMinute</code>	Second within the minute (0-59).
<code>doc['field_name'].date.year</code>	Year (-292000000 - 292000000).
<code>doc['field_name'].date.yearOfCentury</code>	Year within the century (1-100).
<code>doc['field_name'].date.yearOfEra</code>	Year within the era (1-292000000).

String Concatenation 연산은 안되나?

Field 간 연산은 안되나?

Date Field에서 연/월/일/시/분/초 등 접근 안되나?

기존 Field에 조건을 적용해서 새로운 Field를 만들 수 없나?



## 특정 조건에 따라 Value를 설정하고 싶다면?

2,445 hits

New Save Open Share < ⏪ ⏩ Year to date >

Search... (e.g. status:200 AND extension:PHP)

Add a filter +

shopping {id}\_2018.03.01 선택 January 1st 2018, 00:00:00.000 - April 9th 2018, 00:49:53.657 — Auto

Selected Fields

- # 고객나이
- t 연령대

Available Fields

Popular

- t \_type
- t 결제카드
- t 고객성별
- t 고객주소\_시도
- t 구매사이트
- 물건좌표
- # 배송소요시간
- # 상품가격
- t 상품분류
- t \_id
- t \_index
- # \_score
- t 결제카드이름
- 고객ip
- t 배송메모
- # 상품개수
- t 성별-카드

Count

주문시간 per day

Time ↑ 고객나이

- ▶ January 1st 2018, 03:20:32.000 49
- ▶ January 1st 2018, 04:26:01.000 50
- ▶ January 1st 2018, 08:42:19.000 36
- ▶ January 1st 2018, 09:04:54.000 20
- ▶ January 1st 2018, 09:53:20.000 57
- ▶ January 1st 2018, 10:06:46.000 63
- ▶ January 1st 2018, 12:35:55 36
- ▶ January 1st 2018, 12:40:25.000 35
- ▶ January 1st 2018, 12:55:45.000 30
- ▶ January 1st 2018, 14:25:57.000 22
- ▶ January 1st 2018, 15:30:12.000 42
- ▶ January 1st 2018, 16:36:03.000 39
- ▶ January 1st 2018, 18:20:11.000 61
- ▶ January 1st 2018, 18:31:07.000 41

연령대

- 40대
- 50대 이상
- 30대
- 20대
- 50대 이상
- 30대
- 30대
- 20대
- 40대
- 30대
- 50대 이상
- 40대

# scripted field를 추가하자

The screenshot shows the Kibana Management interface for the 'shopping' index pattern. A pink hand icon points to the 'Management' button in the sidebar, labeled '1. 선택'. Another pink hand icon points to the 'scripted fields (7)' tab in the main navigation bar, labeled '2. 선택'. A third pink hand icon points to the '+ Add Scripted Field' button, labeled '3. {id}\_2018.03.01'. A fourth pink hand icon points to the 'scripted fields' section, labeled '4. 선택'. A fifth pink hand icon points to the '+ Add Scripted Field' button again, labeled '5. 선택'.

Management / Kibana

Index Patterns Saved Objects Advanced Settings

+ Create Index Pattern

★ shopping

Time Filter field name: 주문시간

fields (24) scripted fields (7) source filters (0)

Filter All languages ▾

Scripted fields

These scripted fields are computed on the fly from your data. They can be used in visualizations and displayed in your documents, however they can not be searched. You can manage them here and add new ones as you see fit, but be careful, scripts can be tricky!

+ Add Scripted Field

name lang script

format controls

Scroll to top Page Size 25

Collapse

# script를 작성하자 🤖

## ★ shopping

Create Scripted Field

### Name

연령대- 

1. Field 이름 임의로 생성

### Language

painless

### Type

string 

2. String 선택

### Format (Default: String)

String 

3. String 선택

### Transform

### Popularity

0

+ -

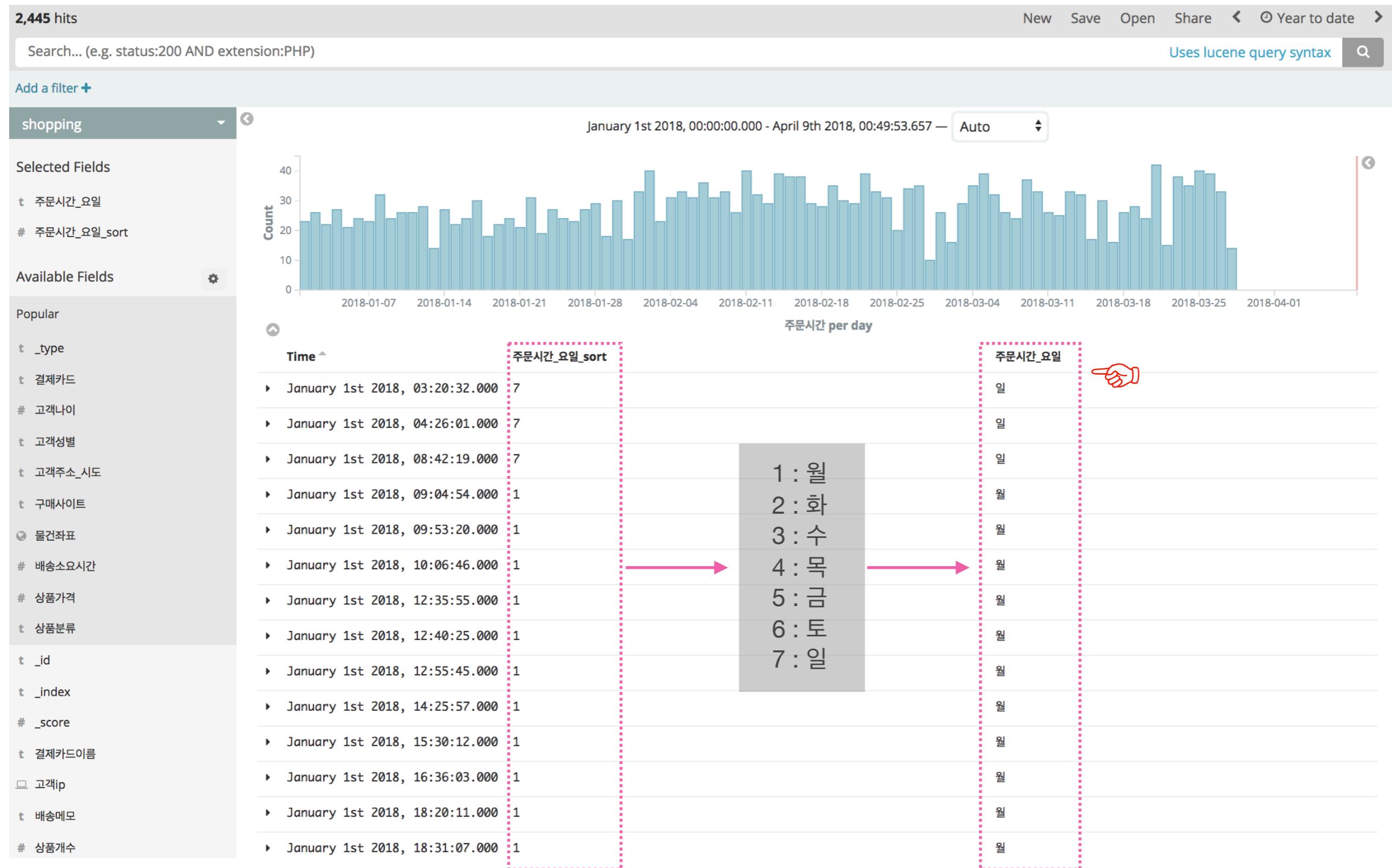
### Script

```
if (doc['고객나이'].value < 20) {  
    return "10대"  
}  
else if (doc['고객나이'].value < 30) {  
    return "20대"  
}  
else if (doc['고객나이'].value < 40) {  
    return "30대"  
}  
else if (doc['고객나이'].value < 50) {  
    return "40대" }  
else {  
    return "50대 이상"  
}
```



4. Script 작성

## 아래와 같은 Field를 생성해보자 ✎ 🤴



# 단, Scripted Field를 사용할 때 다음과 같은 사항에 주의하자!

## 1. Kibana 상에서 Lucene Query Syntax로 검색이 안된다

- ☞ 6.X에서는 kuery를 통해서는 가능
- ☞ 5.X에서는 Filter를 이용해서 검색해야 한다

## 2. 한 번에 한 개의 Document만 조회할 수 있다.

- ☞ 즉, 여러 Documents를 동시에 접근해서 계산하는 시계열 수식은 가능하지 않다

## 3. Scripted Field를 데이터 색인 시에 application에서 생성하자

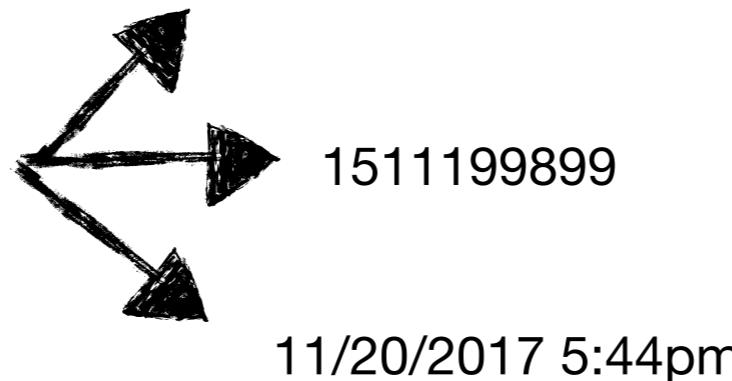
- ☞ Scripted Field는 elasticsearch에 저장되지 않고 쿼리 시점에 Elasticsearch에 전달된다 (연산 작업 필요)
- ☞ 그러므로 Kibana에서 사용자 응답시간 단축을 위해 데이터 색인 시에 scripted field를 생성하자

Managing Field 

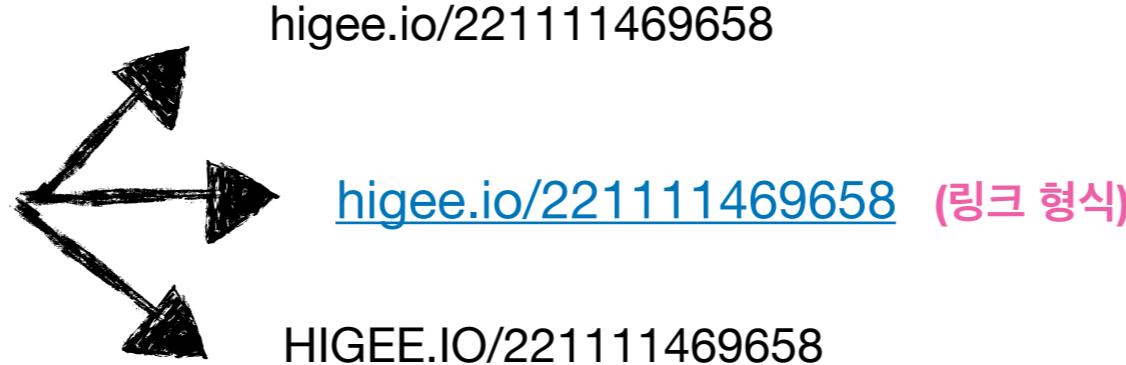
Elasticsearch에 저장된 데이터를 Kibana에서 **형식만 살짝 변경**해서 보여줄 수 없을까?

2017년 11월 20일 17시 44분

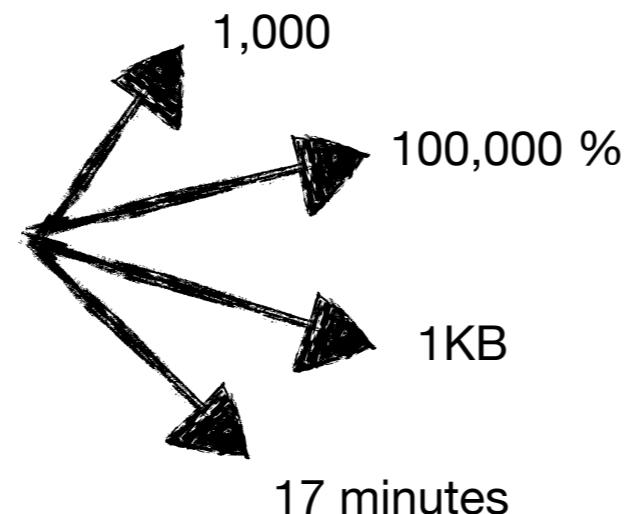
Date를 다르게 표현할 수 없나?



String를 다르게 보여줄 수 없나?



Number를 다르게 보여줄 수 없나?



# Discover에 가서 Index 선택 후 “주문시간”을 표시해보자

2,445 hits      New Save Open Share < ⏪ Year to date ⏩

Search... (e.g. status:200 AND extension:PHP)      Uses lucene query syntax ⏽

Add a filter +

shopping      1. {id}\_2018.03.01      January 1st 2018, 00:00:00.000 - April 9th 2018, 00:49:53.657 — Auto

Selected Fields      2. “주문시간” 선택      2018-01-01 ~ 2018-04-01

Available Fields

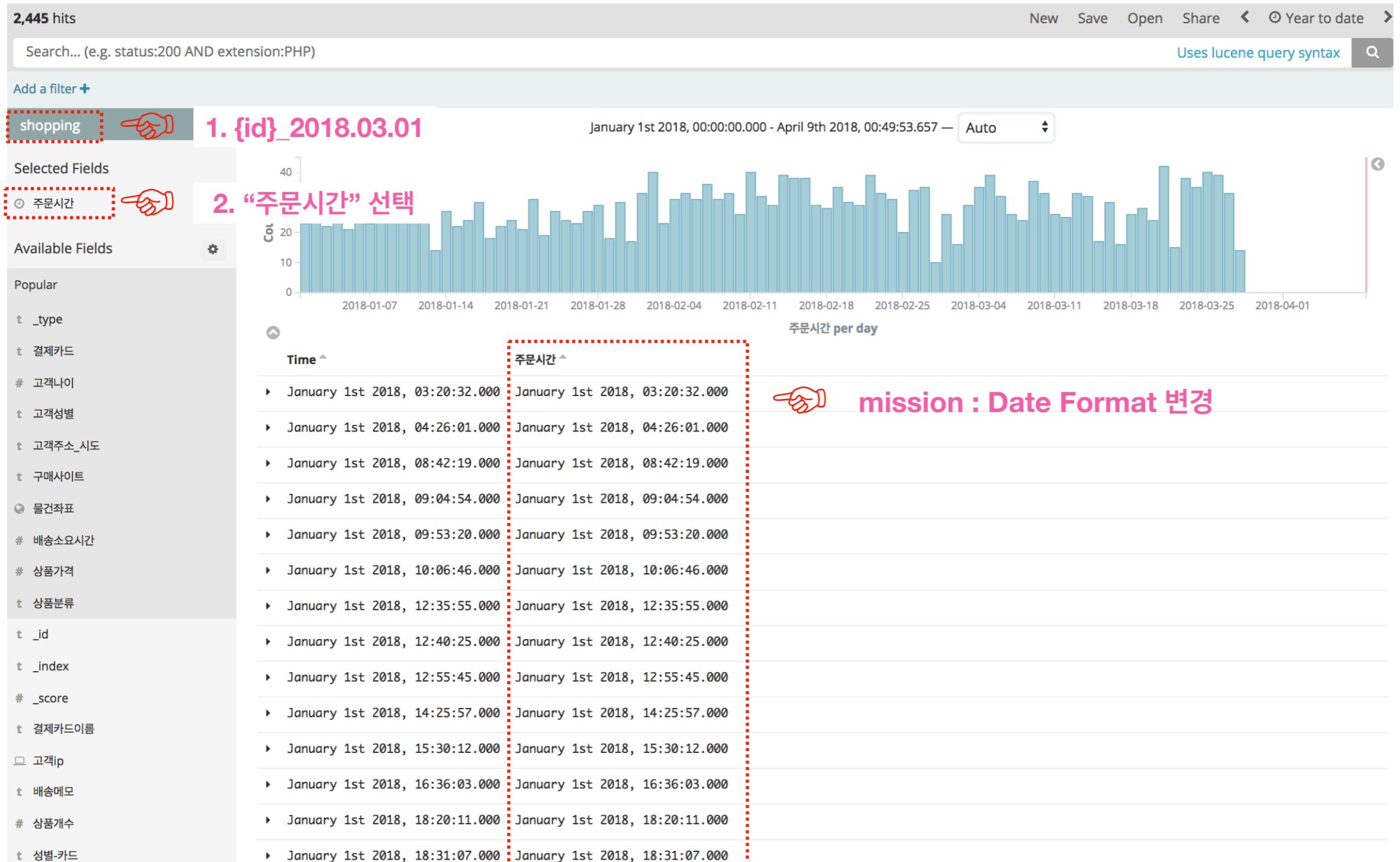
Popular

t \_type  
t 결제카드  
# 고객나이  
t 고객성별  
t 고객주소\_시도  
t 구매사이트  
t 물건좌표  
# 배송소요시간  
# 상품가격  
t 상품분류  
t \_id  
t \_index  
# \_score  
t 결제카드이름  
□ 고객ip  
t 배송메모  
# 상품개수  
t 성별-카드

Time ↑      주문시간 ↑

▶ January 1st 2018, 03:20:32.000	January 1st 2018, 03:20:32.000
▶ January 1st 2018, 04:26:01.000	January 1st 2018, 04:26:01.000
▶ January 1st 2018, 08:42:19.000	January 1st 2018, 08:42:19.000
▶ January 1st 2018, 09:04:54.000	January 1st 2018, 09:04:54.000
▶ January 1st 2018, 09:53:20.000	January 1st 2018, 09:53:20.000
▶ January 1st 2018, 10:06:46.000	January 1st 2018, 10:06:46.000
▶ January 1st 2018, 12:35:55.000	January 1st 2018, 12:35:55.000
▶ January 1st 2018, 12:40:25.000	January 1st 2018, 12:40:25.000
▶ January 1st 2018, 12:55:45.000	January 1st 2018, 12:55:45.000
▶ January 1st 2018, 14:25:57.000	January 1st 2018, 14:25:57.000
▶ January 1st 2018, 15:30:12.000	January 1st 2018, 15:30:12.000
▶ January 1st 2018, 16:36:03.000	January 1st 2018, 16:36:03.000
▶ January 1st 2018, 18:20:11.000	January 1st 2018, 18:20:11.000
▶ January 1st 2018, 18:31:07.000	January 1st 2018, 18:31:07.000

mission : Date Format 변경



# Date Format 변경하려는 Field의 Controls 선택

## 2. Index Patterns 선택

The screenshot shows the Kibana Management interface. On the left sidebar, the 'Management' tab is selected, indicated by a red dashed border. The main area displays the 'Index Patterns' page for the 'shopping' index. A green callout '3. {id}\_2018.03.01' points to the 'Time Filter field name: 주문시간' message at the top. The table lists fields with their types, formats, and searchability. A red dashed box highlights the '주문시간' field, which is of type 'date'. A green callout '4. 선택' points to the 'controls' column for this field.

name	type	format	searchable	aggregatable	excluded	controls
판매자평점	number		✓	✓		
주문시간	date		✓	✓		
접수시간	date		✓	✓		
접수번호	string		✓	✓		
예약여부	string		✓	✓		
수령시간	date		✓	✓		

## 적절한 Format Pattern으로 수정하자

Management / Kibana / Indices / shopping / Field

Index Patterns Saved Objects Advanced Settings

+ Create Index Pattern

★ shopping

주문시간

Type

수정 안됨 date

Format (Default: Date)

1. Date 선택 Date

moment.js format pattern (Default: "MMMM Do YYYY, HH:mm:ss.SSS")

2. 아래와 같이 입력하자 MM월DD일

MM월DD일

Samples

Input	Formatted
1516536304175	January 21st 2018, 21:05:04.175
1514732400000	January 1st 2018, 00:00:00.000
1546268399999	December 31st 2018, 23:59:59.999

Popularity

1

3. 선택하자 Cancel

**Discover**에 돌아가서 확인하자

아래를 참고해서 “수령시간”을 Unix Timestamp (Millisecond)로 표시해보자 ↗

날짜 단위	문법	예시	설명
Year	YYYY	2014	4자리 표시
Year	YY	14	2자리 표시
Month	M	1	1~2자리 표시
Month	MM	01	2자리 표시
Day	D	1	1~2자리 표시
Day	DD	01	2자리 표시
Day	Do	1st	며칠째인지 표시
Hour	H	1	1자리 표시 (24시)
Hour	HH	01	1~2자리 표시 (24시)
Hour	h	1	1자리 표시 (12시)
Hour	hh	01	1~2자리 표시 (12시)
a	h	am/pm	소문자 표시
A	hh	AM/PM	대문자 표시
Minute	m	1	1자리 표시
Minute	mm	01	1~2자리 표시
Second	s	1	1자리 표시
Second	ss	01	1~2자리 표시
Second	X	1410715640.579	Unix Timestamp 초
Millisecond	x	1410715640579	Unix Timestamp 밀리초

## **String Type의 Format 전환 실습을 위해 Index를 등록하자**

Index : url\_{id}

Time Filter Field : 없음

## Discover에 가서 “full\_url”과 “partial\_url”을 표시해보자

3 hits      New    Save    Open    Share

Search... (e.g. status:200 AND extension:PHP)      Uses lucene query syntax     

Add a filter +

**url\_higee** **Selected Fields** **1. url\_{id} 선택**

**full\_url** **2. “full\_url”, “partial\_url” 선택**

full_url	partial_url
▶ higee.io/221111469658	221111469658
▶ higee.io/221098816524	221098816524
▶ higee.io/221094794994	221094794994

**mission : 두 Field 모두 링크로 변경**

**Available Fields**

- t \_id
- t \_index
- # \_score
- t \_type

# Data Format 변경하려는 Field의 Control 선택

(=우선은 “full\_url” 부터)

Management / Kibana

Index Patterns Saved Objects Advanced Settings

+ Create Index Pattern

★ shopping

url\_higee  url\_{id} 선택

This page lists every field in the **url\_higee** index and the field's associated core type as recorded by Elasticsearch. While this list allows you to view the core type of each field, changing field types must be done using Elasticsearch's [Mapping API](#).

fields (7) scripted fields (0) source filters (0)

Filter All field types ▾

name	type	format	searchable	aggregatable	excluded	controls
_id	string		✓			
_index	string		✓	✓		
_score	number					
_source	_source					
_type	string		✓	✓		
full_url	string		✓	✓		
partial_url	string		✓	✓		

Scroll to top Page Size 25 ▾

# Url Templates에 적절한 값을 입력하자

url\_higee

full\_url

Type

string



수정 안됨

Format (Default: String)

⚠ Warning

Url



1. URL 선택

Type

Link

Url Template

ℹ Url Template Help

{{rawValue}}



2. 다음과 같이 입력하자

{{rawValue}}

Label Template

ℹ Label Template Help

전체 url



3. 다음과 같이 입력하자

전체 url

## Samples

### Input

john

/some pathname/asset.png

1234

### Formatted

[john](#)

[/some pathname/asset.png](#)

[1234](#)

## Popularity

0



Update Field

Cancel



4. Update Field 선택

## Discover에 돌아가서 확인하자

3 hits

Search... (e.g. status:200 AND extension:PHP)

Add a filter +

url_higee	partial_url
▶ full_url ▶ 전체 url ▶ 전체 url ▶ 전체 url	221111469658 221098816524 221094794994

Selected Fields

t full\_url  
t partial\_url

Available Fields

t \_id  
t \_index  
# \_score  
t \_type

full\_url  
전체 url  
전체 url  
전체 url

▶ 클릭해보자

그렇다면 이번에는 “partial\_url” field를 url data format으로 변형해보자   
단, url\_template과 label template는 아래와 같이 입력하자.

url\_template -> higee.io/{{value}}

label template -> 부분 url : {{value}}

## Discover에 돌아가서 확인하자

3 hits

Search... (e.g. status:200 AND extension:PHP)

Add a filter +

url\_higee

Selected Fields

t full\_url

t partial\_url

Available Fields

t \_id

t \_index

# \_score

t \_type

full\_url

▶ 전체 url

▶ 전체 url

▶ 전체 url

partial\_url

▶ 부분 url : 221111469658

▶ 부분 url : 221098816524

▶ 부분 url : 221094794994



클릭해보자

**Number Type의 Format 전환 실습을 위해 Index를 등록하자** 

Index : number\_{id}  
Time Filter Field : 없음

## Discover에 가서 “percent”, “bytes”, “duration”을 표시해보자

5 hits

Search... (e.g. status:200 AND extension:PHP)

Add a filter +

number\_higee



1. number\_{id} 선택

Selected Fields

# byte

# duration

# percent



2. Field 선택

Available Fields



t \_id

t \_index

# \_score

t \_type

byte	duration	percent
▶ 8,191	10,005	0.6
▶ 135,351	335,513	0.2
▶ 31	5,013	0.2
▶ 133	33,535	0.3
▶ 13,535,139	33	0



mission : 적절한 Format으로 변경

# Data Format 변경하려는 Field의 Control 선택

(=우선은 “duration” 부터)

Management / Kibana

Index Patterns Saved Objects Advanced Settings

+ Create Index Pattern

★ shopping

number\_higee

☞ number\_{id} 선택

This page lists every field in the **number\_higee** index and the field's associated core type as recorded by Elasticsearch. While this list allows you to view the core type of each field, changing field types must be done using Elasticsearch's [Mapping API](#).

fields (8)    scripted fields (0)    source filters (0)

Filter    All field types

name	type	format	searchable	aggregatable	excluded	controls
_id	string		✓			
_index	string		✓	✓		
_score	number					
_source	_source					
_type	string		✓	✓		
byte	number		✓	✓		
duration	number	Duration	✓	✓		
percent	number		✓	✓		

Scroll to top    Page Size 25

# duration Field Format을 변경하자

Management / Kibana / Indices / number\_higee / Field

Index Patterns Saved Objects Advanced Settings

+ Create Index Pattern

★ shopping

number\_higee

duration

Type: number

Format (Default: Number) ▲ Warning

Duration (1. 선택)

Input Format:Milliseconds (2. 원본 시간 단위)

Output Format: Seconds (3. 변환하려는 시간 단위)

Decimal Places: 2 (4. 소수점 자리수)

Samples

Input	Formatted
-123	-0.12
1	0.00
12	0.01
123	0.12
658	0.66
1988	1.99
3857	3.86
123292	123.29
923528271	923528.27

Popularity: 0 + -

Update Field (5. 선택) Cancel

## Discover에 돌아가서 확인하자

5 hits

Search... (e.g. status:200 AND extension:PHP)

Add a filter +

number_higee	byte	duration	percent
# byte	▶ 8,191	10.01	0.6
# duration	▶ 135,351	335.51	0.2
# percent	▶ 31	5.01	0.2
Available Fields	▶ 133	33.53	0.3
	▶ 13,535,139	0.03	0

Selected Fields

# \_id  
# \_index  
# \_score  
# \_type

Available Fields

t \_id  
t \_index  
# \_score  
t \_type

add

duration

seconds 단위로 변환됐다!

# Data Format 변경하려는 Field의 Control 선택

(=“percent”)

Management / Kibana

Index Patterns Saved Objects Advanced Settings

+ Create Index Pattern

★ shopping

number\_higee  **number\_{id} 선택**

This page lists every field in the **number\_higee** index and the field's associated core type as recorded by Elasticsearch. While this list allows you to view the core type of each field, changing field types must be done using Elasticsearch's [Mapping API](#).

name	type	format	searchable	aggregatable	excluded	controls
_id	string		✓			
_index	string		✓	✓		
_score	number					
_source	_source					
_type	string		✓	✓		
byte	number		✓	✓		
duration	number	Duration	✓	✓		
percent	number		✓	✓		

Scroll to top  Page Size 25

# percent Field Format을 변경하자

Management / Kibana / Indices / number\_higee / Field

Index Patterns Saved Objects Advanced Settings

+ Create Index Pattern

★ shopping

number\_higee  
percent

Type  
number

Format (Default: Number)  
Percentage      1. 선택  
Numeral.js format pattern (Default: "0,0.[000]%)

0,0.[000]%      2. percentage 표시 입력

Samples

Input	Formatted
0.1	10%
0.99999	99.999%
1	100%
100	10,000%
1000	100,000%

Popularity  
1      +      -  
Update Field      Cancel

3. 선택

## Discover에 돌아가서 확인하자

5 hits

Search... (e.g. status:200 AND extension:PHP)

Add a filter +

number_higee	byte	duration	percent
# byte	▶ 8,191	10.01	61%
# duration	▶ 135,351	335.51	15%
# percent	▶ 31	5.01	21%
	▶ 133	33.53	30%
	▶ 13,535,139	0.03	1%

Selected Fields

# byte  
# duration  
# percent

Available Fields

t \_id  
t \_index  
# \_score  
t \_type

 % 형태로 표시됐다

## “bytes” Field를 아래와 같이 표시해보자 ✎ 🤴

5 hits

Search... (e.g. status:200 AND extension:PHP)

Add a filter +

number_higee	byte	duration	percent
# byte	▶ 7.999KB	10.01	61%
# duration	▶ 132.179KB	335.51	15%
# percent	▶ 31B	5.01	21%
	▶ 133B	33.53	30%
	▶ 12.908MB	0.03	1%



# 주의!!

**Data Format**이 변하는 것이지 **Data Type**이 변하는 것이 아니다.

그러므로 Elasticsearch에 저장된 데이터 자체는 변하지 않는다!

Management / Kibana / Indices / number\_higee / Field

Index Patterns Saved Objects Advanced Settings

+ Create Index Pattern

★ shopping

number\_higee  
percent

Type  
  **변경 불가능**

Format (Default: Number)  
  **변경 가능**  
Numeral.js format pattern (Default: "0,0,[000]%)

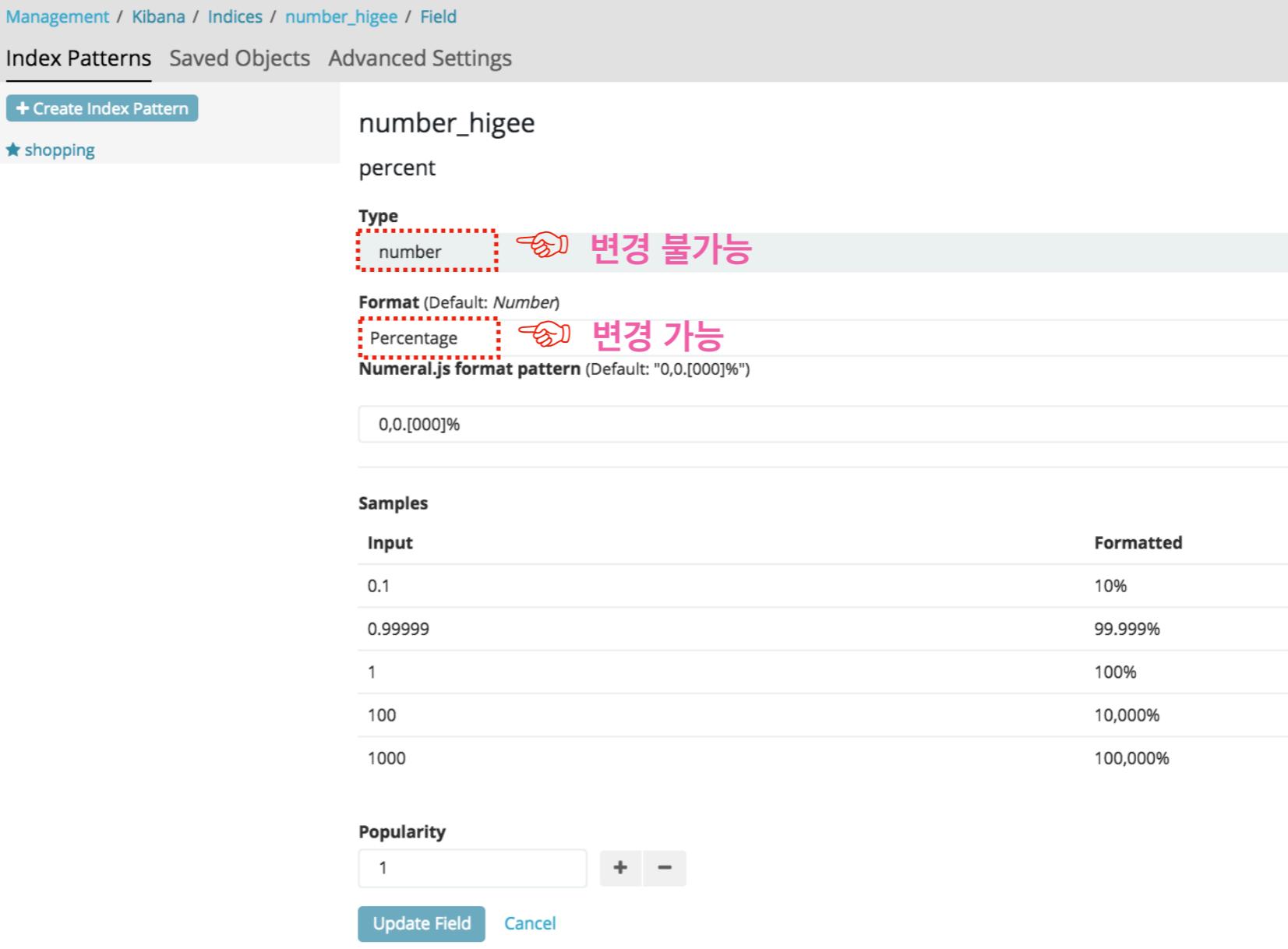
0,0,[000]%

Samples

Input	Formatted
0.1	10%
0.99999	99.999%
1	100%
100	10,000%
1000	100,000%

Popularity   

Update Field Cancel



Dashboard/Visualization export

Kibana Dashboard 및 Visualize Object를 Import/Export 할 수 없나?  
(~~데이터~~ 백업이 아닌 UI 백업)

# Dashboard export

The screenshot shows the Kibana Management interface with the following steps highlighted:

- 1. Management 선택**: A hand icon points to the "Management" button in the sidebar.
- 2. Saved Objects 선택**: A hand icon points to the "Saved Objects" tab in the top navigation bar.
- 3. Dashboards 선택**: A hand icon points to the "Dashboards (1)" tab in the sub-navigation bar.
- 4. {id}\_dashboard 선택**: A hand icon points to the "shopping" dashboard entry in the list.
- 5. Export 선택**: A hand icon points to the "Export" button in the top right corner.
- 6. {id}\_dashboard.json 저장**: A hand icon points to the "Export" button in the top right corner, with a note indicating the file will be saved as {id}\_dashboard.json.

The interface includes a sidebar with "Discover", "Visualize", "Dashboard", "Timelion", "Dev Tools", and "Management" options. The main area shows an "Edit Saved Objects" section with a search bar, a "Delete" button, and an "Export" button. The "Dashboards" tab is selected, showing one item named "shopping".

## Dashboard export

```
◀ ▶ dashboard.json X
1 [
2 {
3   "_id": "AV-a4D_kMbjMXoGxG0m2",
4   "_type": "dashboard",
5   "_source": {
6     "title": "shopping",
7     "hits": 0,
8     "description": "",
9     "panelsJSON": "[{"col":7,"id":"AWJdhp4tzMQVnr-9MxxP","panelIndex":19,"row":8,"size_x":6,"size_y":4,"type":"visualization"}, {"col":7,"id":"AWJdii5xzMQVnr-9MxxQ","panelIndex":20,"row":8,"size_x":6,"size_y":4,"type":"visualization"}]",
10    "optionsJSON": "{\"darkTheme\":false}",
11    "uiStateJSON": "{\"P-19\":{\"vis\":{\"defaultColors\":[\"0 - 7\":\"rgb(255,255,204)\",\"7 - 13\":\"rgb(254,217,118)\",\"13 - 20\":\"rgb(253,141,60)\",\"20 - 26\":\"rgb(227,27,28)\"}],\"legendOpen\":false}}, \"P-20\":{\"vis\":{\"defaultColors\":[\"0 - 7\":\"rgb(255,255,204)\",\"7 - 13\":\"rgb(254,217,118)\",\"13 - 20\":\"rgb(253,141,60)\",\"20 - 26\":\"rgb(227,27,28)\"],\"legendOpen\":true}}}",
12    "version": 1,
13    "timeRestore": false,
14    "kibanaSavedObjectMeta": {
15      "searchSourceJSON": "{\"filter\": [{\"query\": {\"match_all\": {}}}], \"highlightAll\": true, \"version\": true}"
16    }
17  }
18 }
19 }
```

대시보드를 백업 받으면 위와 같이 {id}\_dashboard.json으로 저장한 json 파일이 생기는데, 대시보드에 어떤 visualization이 어느 위치에 어떤 크기로 생성되었는지에 관한 정보를 담고 있다.

개별 visualization 들이 어떤 aggregation으로 만들어졌는지 등에 관한 정보는 없다.

## Visualization 백업 필요!

# Visualization export

Management / Kibana

Index Patterns Saved Objects Advanced Settings

**1. Management 선택**

**2. Saved Objects 선택**

Edit Saved Objects

From here you can delete saved objects, such as saved searches. You can also edit the raw data of saved objects. Typically objects are only modified via their associated application, which is probably what you should use instead of this screen. Each tab is limited to 100 results. You can use the filter to find objects not in the default list.

Export Everything Import

Dashboards (1) Searches (0) Visualizations (14)

**3. Visualizations 선택**

Delete Export

4. {id}로 검색

5. {id}로 시작하는 모든 visualization 선택  
= {id}\_dashboard에서 사용한 모든 visualization

6. Export 선택

7. {id}\_visualization.json 저장

shoping\_tag  
shopping\_area  
shopping\_coordinates  
shopping\_data\_table  
shopping\_goal  
shopping\_heatmap  
shopping\_horizontal\_bar  
shopping\_line  
shopping\_markdown  
shopping\_metrics  
shopping\_pie  
shopping\_region  
shopping\_timelion  
shopping\_timelion\_2

14 selected

Collapse

手指 icon 1: Management tab

手指 icon 2: Saved Objects tab

手指 icon 3: Visualizations tab

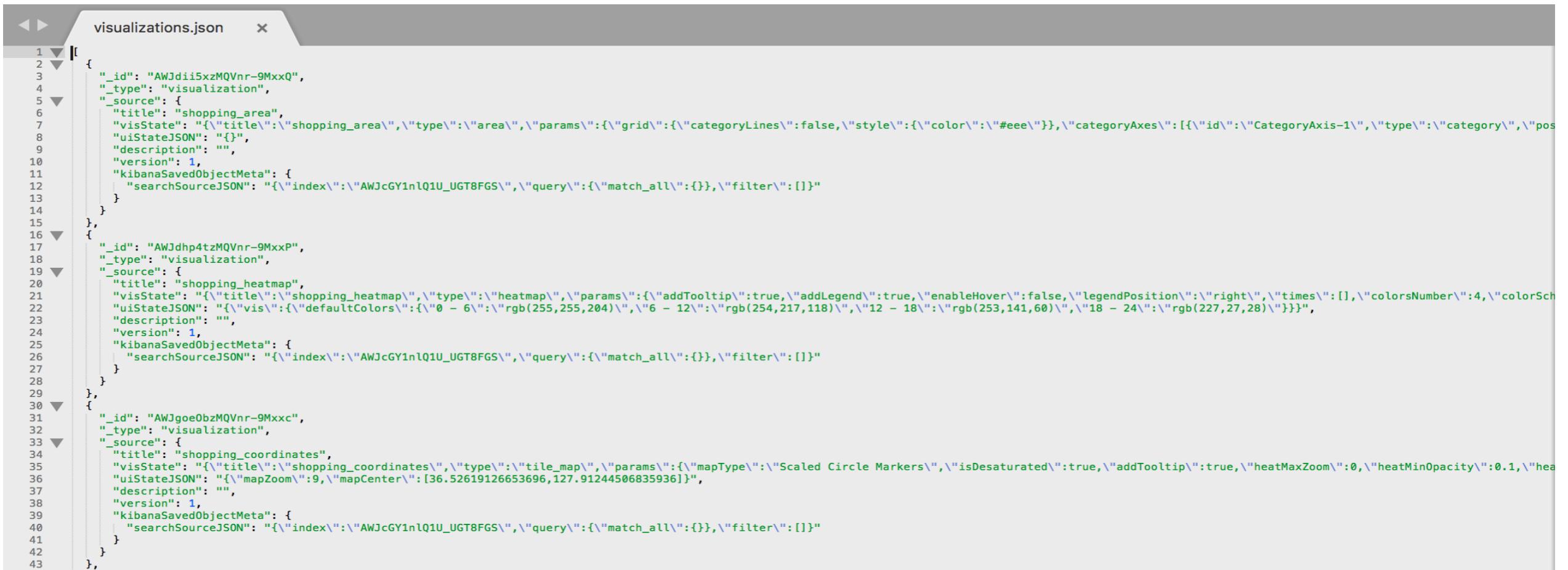
手指 icon 4: Search bar

手指 icon 5: Visualization list

手指 icon 6: Export button

手指 icon 7: Destination note

# Visualization export



```
1  {
2    "_id": "AWJdi5xzMQVnr-9MxxQ",
3    "_type": "visualization",
4    "_source": {
5      "title": "shopping_area",
6      "visState": "{\"title\":\"shopping_area\",\"type\":\"area\",\"params\":{\"grid\":{\"categoryLines\":false,\"style\":{\"color\":\"#eee\"}},\"categoryAxes\":[{\"id\":\"CategoryAxis-1\",\"type\":\"category\"},\"pos
7      "uiStateJSON": "{}",
8      "description": "",
9      "version": 1,
10     "kibanaSavedObjectMeta": {
11       "searchSourceJSON": "{\"index\":\"AWJcGY1nlQ1U_UGT8FGS\",\"query\":{\"match_all\":{}},\"filter\":[]}"
12     }
13   },
14   {
15     "_id": "AWJdhp4tzMQVnr-9MxxP",
16     "_type": "visualization",
17     "_source": {
18       "title": "shopping_heatmap",
19       "visState": "{\"title\":\"shopping_heatmap\",\"type\":\"heatmap\",\"params\":{\"addTooltip\":true,\"addLegend\":true,\"enableHover\":false,\"legendPosition\":\"right\",\"times\":[],\"colorsNumber\":4,\"colorSch
20       "uiStateJSON": "{\"vis\":{\"defaultColors\":[\"0 - 6\":\"rgb(255,255,204)\",\"6 - 12\":\"rgb(254,217,118)\",\"12 - 18\":\"rgb(253,141,60)\",\"18 - 24\":\"rgb(227,27,28)\"}}}",
21       "description": "",
22       "version": 1,
23       "kibanaSavedObjectMeta": {
24         "searchSourceJSON": "{\"index\":\"AWJcGY1nlQ1U_UGT8FGS\",\"query\":{\"match_all\":{}},\"filter\":[]}"
25       }
26     }
27   },
28   {
29     "_id": "AWJgoeObzMQVnr-9Mxxc",
30     "_type": "visualization",
31     "_source": {
32       "title": "shopping_coordinates",
33       "visState": "{\"title\":\"shopping_coordinates\",\"type\":\"tile_map\",\"params\":{\"mapType\":\"Scaled Circle Markers\",\"isDesaturated\":true,\"addTooltip\":true,\"heatMaxZoom\":0,\"heatMinOpacity\":0.1,\"heat
34       "uiStateJSON": "{\"mapZoom\":9,\"mapCenter\":[36.52619126653696,127.91244506835936]}","
35       "description": "",
36       "version": 1,
37       "kibanaSavedObjectMeta": {
38         "searchSourceJSON": "{\"index\":\"AWJcGY1nlQ1U_UGT8FGS\",\"query\":{\"match_all\":{}},\"filter\":[]}"
39       }
40     }
41   }
42 },
43 }
```

visualization을 백업 받으면 위와 같은 json 파일이 생기는데, 각 visualization이 어떻게 구성되었는지에 대한 정보를 담고 있다



즉, dashboard UI를 온전히 백업 받으려면 dashboard와 visualization을 모두 백업 받아야 한다

## Dashboard/Visualization import

(시연을 위해 dashboard 및 visualization 삭제)

# Dashboard import

1. Management 선택

2. Saved Objects 선택

3. Dashboards 선택

4. Import 선택

5. {id}\_dashboard.json 선택

The screenshot shows the Kibana Management interface. On the left, there's a sidebar with icons for Discover, Visualize, Dashboard, Timelion, Dev Tools, and Management. A pink callout points to the 'Management' icon with the text '1. Management 선택'. The main area has tabs for Index Patterns, Saved Objects (which is selected), and Advanced Settings. A pink callout points to the 'Saved Objects' tab with the text '2. Saved Objects 선택'. Below this, there's a section titled 'Edit Saved Objects' with a note about deleting saved objects. It includes tabs for Dashboards (0), Searches (1), and Visualizations (100 of 104). A pink callout points to the 'Dashboards' tab with the text '3. Dashboards 선택'. At the top right, there are 'Export Everything' and 'Import' buttons; the 'Import' button is highlighted with a pink border. A pink callout points to the 'Import' button with the text '4. Import 선택'. At the bottom right of the main area, there's a 'Delete' and 'Export' button, with a pink callout pointing to the 'Delete' button labeled '5. {id}\_dashboard.json 선택'.

# Import한 Dashboard에 가보면

The screenshot shows a Kibana dashboard titled "shopping". The left sidebar includes links for Discover, Visualize, Dashboard (selected), Timelion, Dev Tools, and Management. The main area displays six error messages, each consisting of a red exclamation mark icon and the text "Could not locate that visualization (id: <visualization\_id>)". The visualization IDs listed are AWJgIBU6zMQVnr-9Mxxa, AWJguWqpzMQVnr-9Mxx, AWJgoeObzMqVnr-9Mxxc, AWJgEGqCzMqVnr-9MxxW, AWJgsNh9zMqVnr-9Mxxe, and AWJdhP4tzMQVnr-9MxxP.

개별 Visualization들을 Import 하지 않았기에 위와 같이 에러가 뜬다.

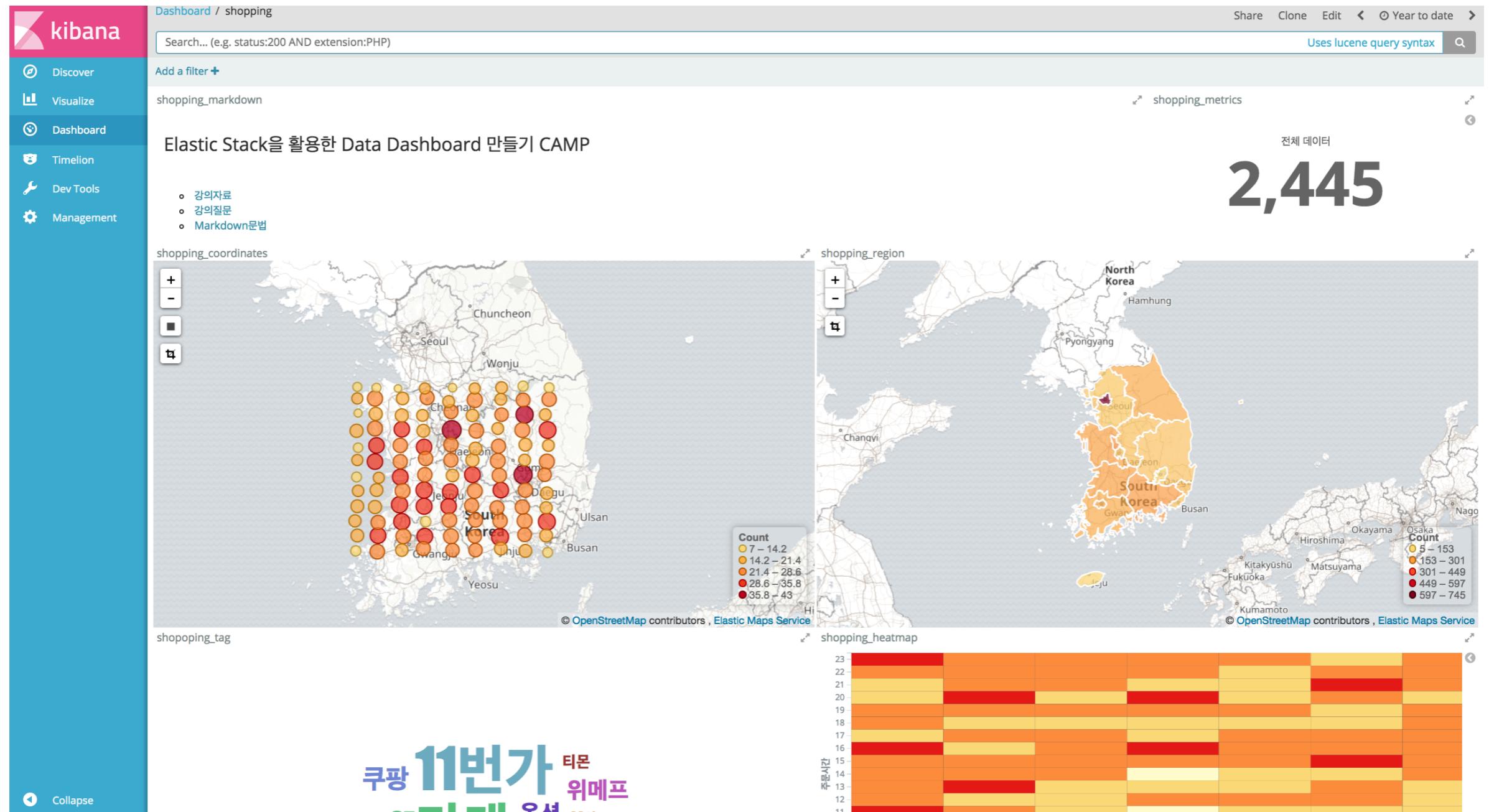
# Visualization import

The screenshot shows the Kibana Management interface with the following steps highlighted:

- 1. Management 선택**: A pink callout points to the "Management" icon in the left sidebar.
- 2. Saved Objects 선택**: A pink callout points to the "Saved Objects" tab in the top navigation bar.
- 3. Visualizations 선택**: A pink callout points to the "Visualizations" tab in the "Edit Saved Objects" header.
- 4. Import 선택**: A pink callout points to the "Import" button in the top right corner.
- 5. {id}\_visualization.json 선택**: A pink callout points to the "Import" button again, indicating where to select the JSON file.

The interface includes a sidebar with "Discover", "Visualize", "Dashboard", "Timelion", "Dev Tools", and "Management" options. The main area shows tabs for "Dashboards (1)", "Searches (1)", and "Visualizations (100 of 118)". A search bar and buttons for "Delete" and "Export" are also present.

# Import한 Dashboard에 다시 가보면



Dashboard 기능

# 동일한 환경에서 작업하기 위해 shopping dashboard를 clone 하자

1. Clone 선택

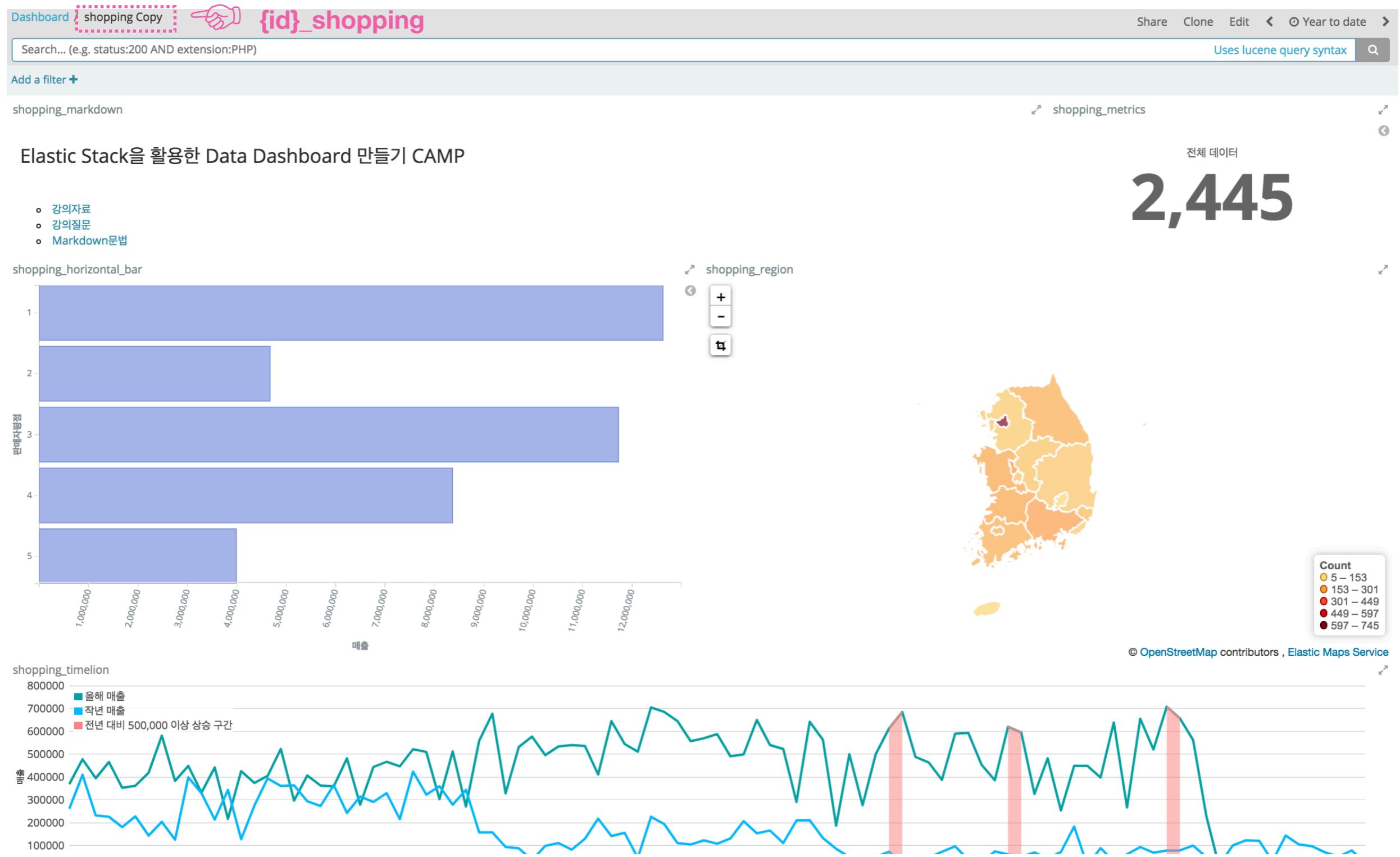
The screenshot shows a Kibana dashboard titled "shopping". The top navigation bar includes "Dashboard / shopping", "Share", "Clone" (which is highlighted with a pink dashed box), "Edit", and "Year to date". A search bar says "Search... (e.g. status:200 AND extension:PHP)". The left sidebar has links for "Discover", "Visualize", "Dashboard", "Timelion", "Dev Tools", and "Management".

The dashboard contains several visualizations:

- "shopping\_markdown": A text area containing "Elastic Stack을 활용한 Data Dashboard 만들기 CAMP" and a bulleted list: "○ 강의자료", "○ 강의질문", "○ Markdown문법".
- "shopping\_horizontal\_bar": A horizontal bar chart with categories from 1 to 5 on the y-axis and values in millions on the x-axis (1,000,000 to 12,000,000). The bars are blue.
- "shopping\_region": A map of South Korea and surrounding regions, color-coded by region ID. A legend on the right shows "Count" ranges: 5 - 153 (yellow), 153 - 301 (orange), 301 - 449 (red), 449 - 597 (dark red), and 597 - 745 (dark orange).
- "shopping\_timelion": A time series line chart showing "올해 매출" (green line) and "작년 매출" (blue line) over time. It also highlights "전년 대비 500,000 이상 상승 구간" (pink shaded areas).

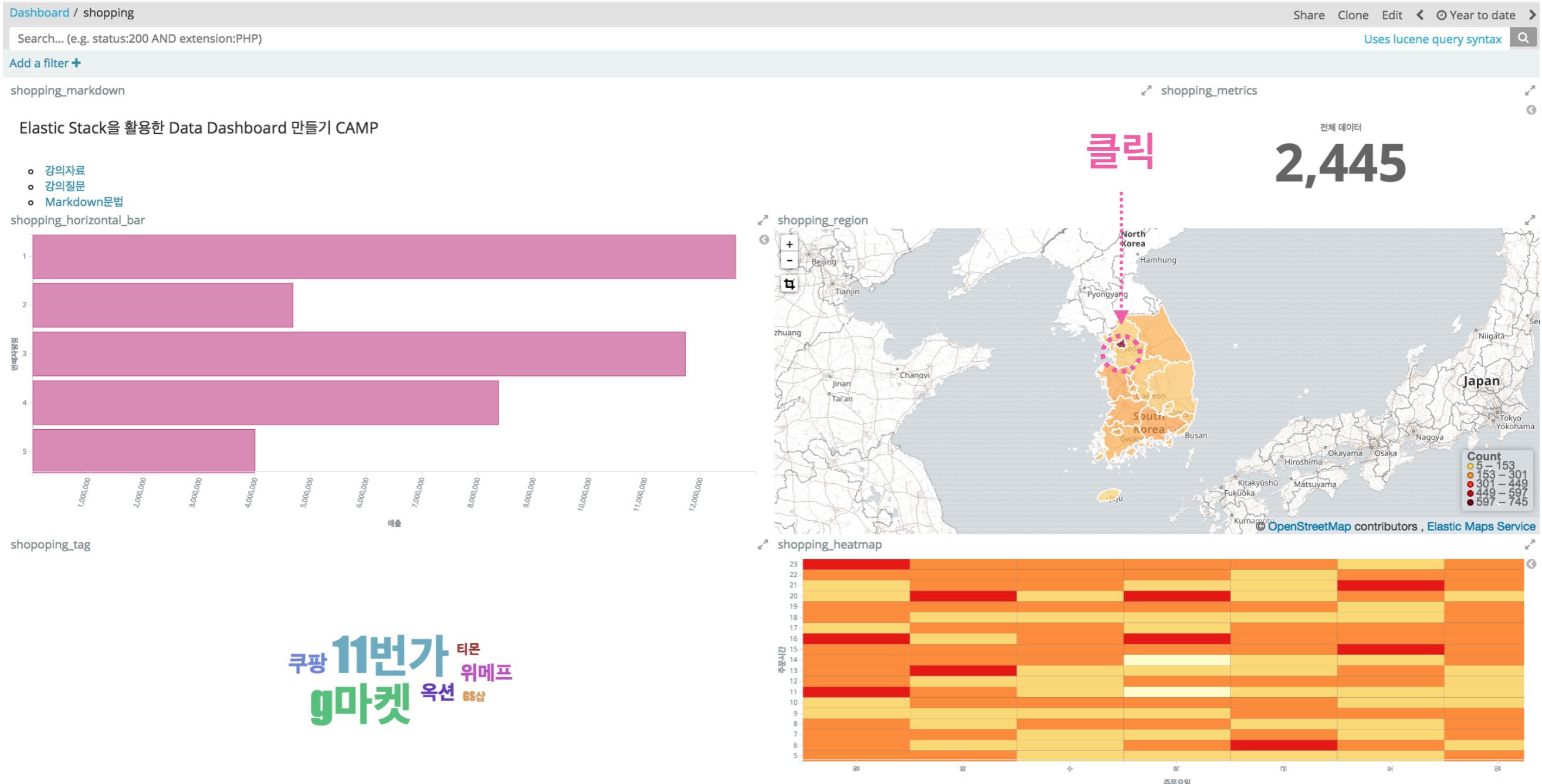
A central modal window is open, titled "Clone Dashboard". It contains the text "2. 이름 : {id}\_shopping" and "Please enter a new name for your dashboard." Below this is a text input field with "shopping Copy" and a pink hand cursor icon pointing at it. The modal has "Cancel" and "Confirm Clone" buttons.

# Clone 받은 환경에서 Dashboard를 사용할 수 있다

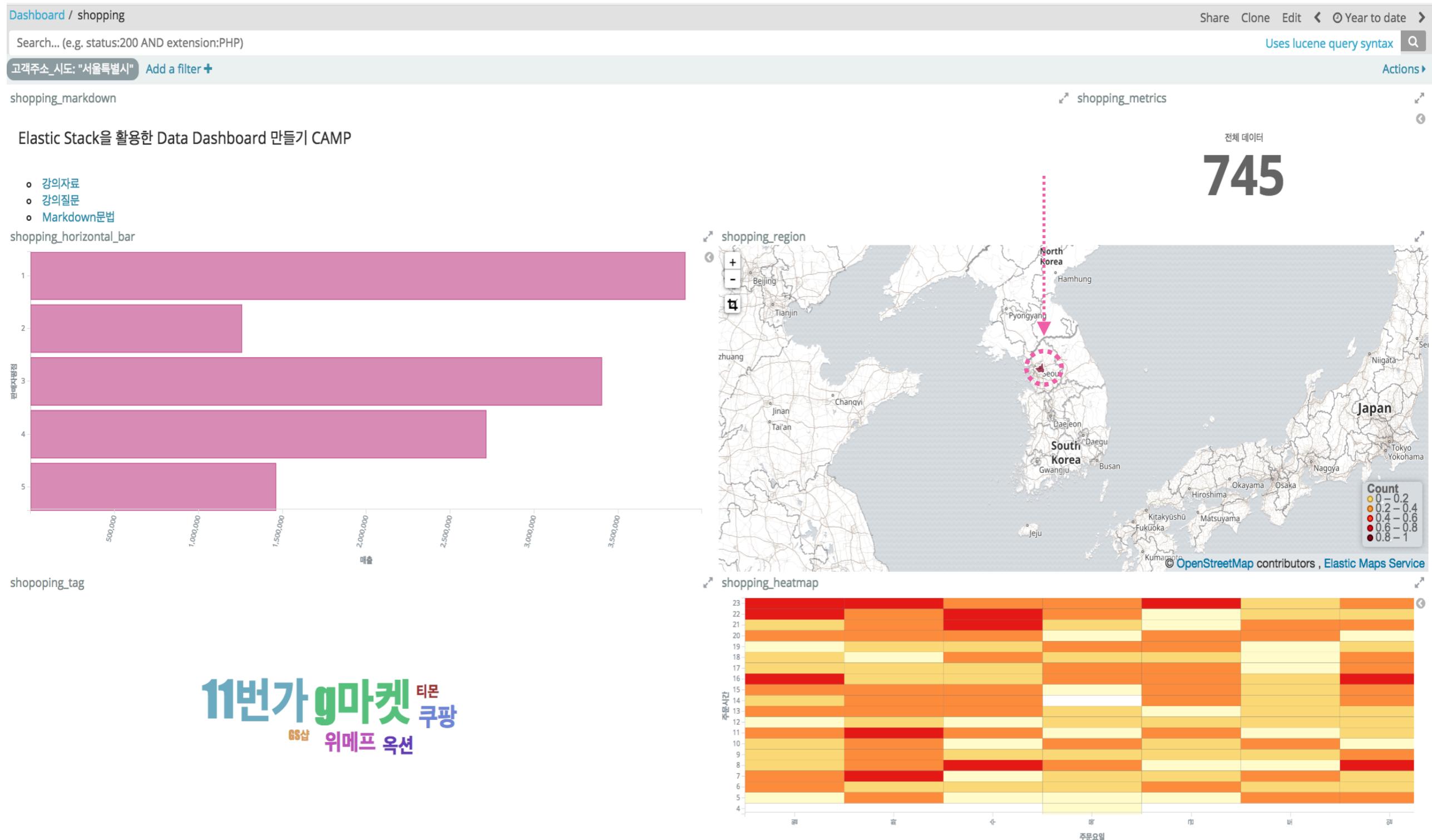


무엇보다 Dashboard가 **Interactive**하다!

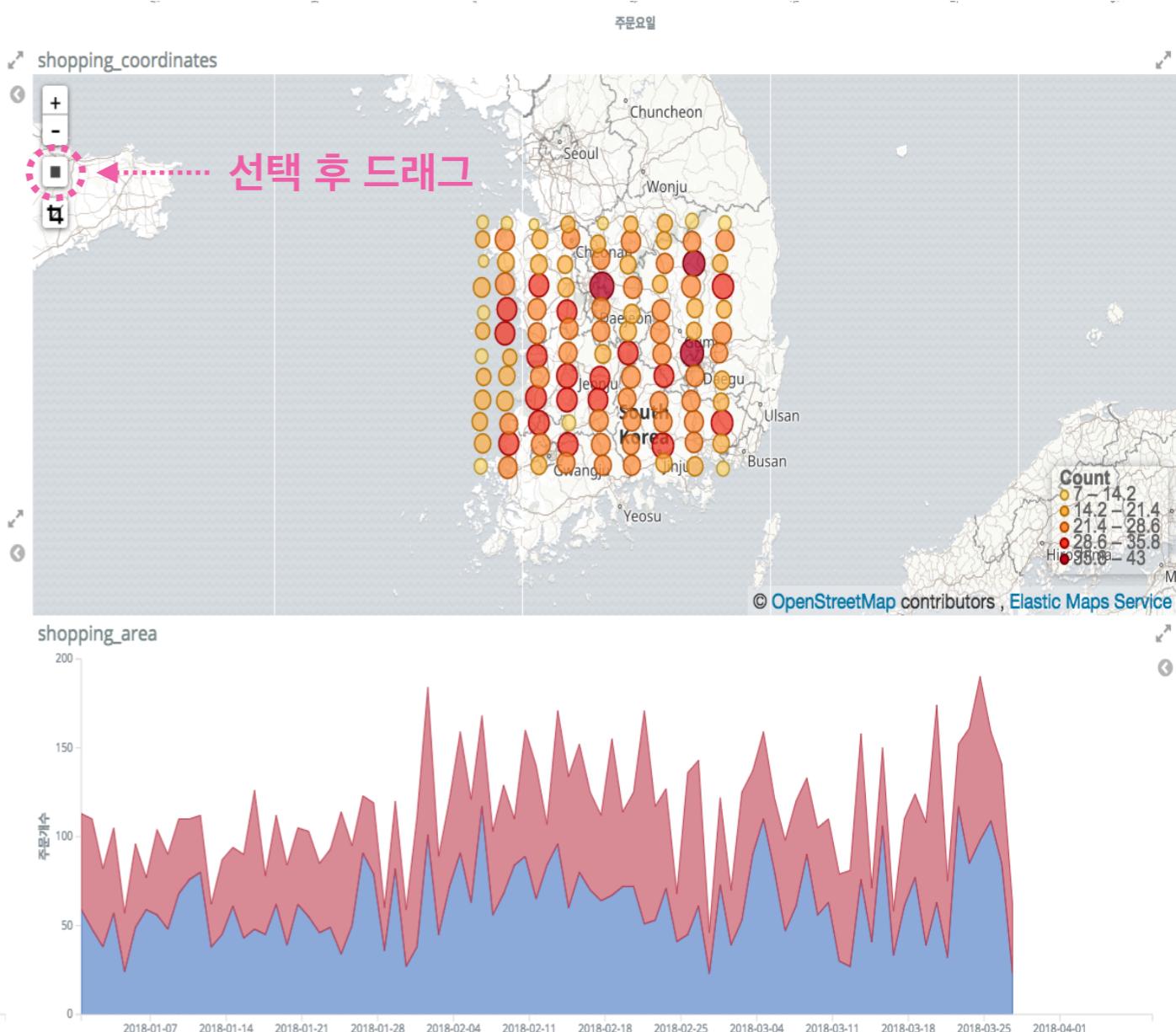
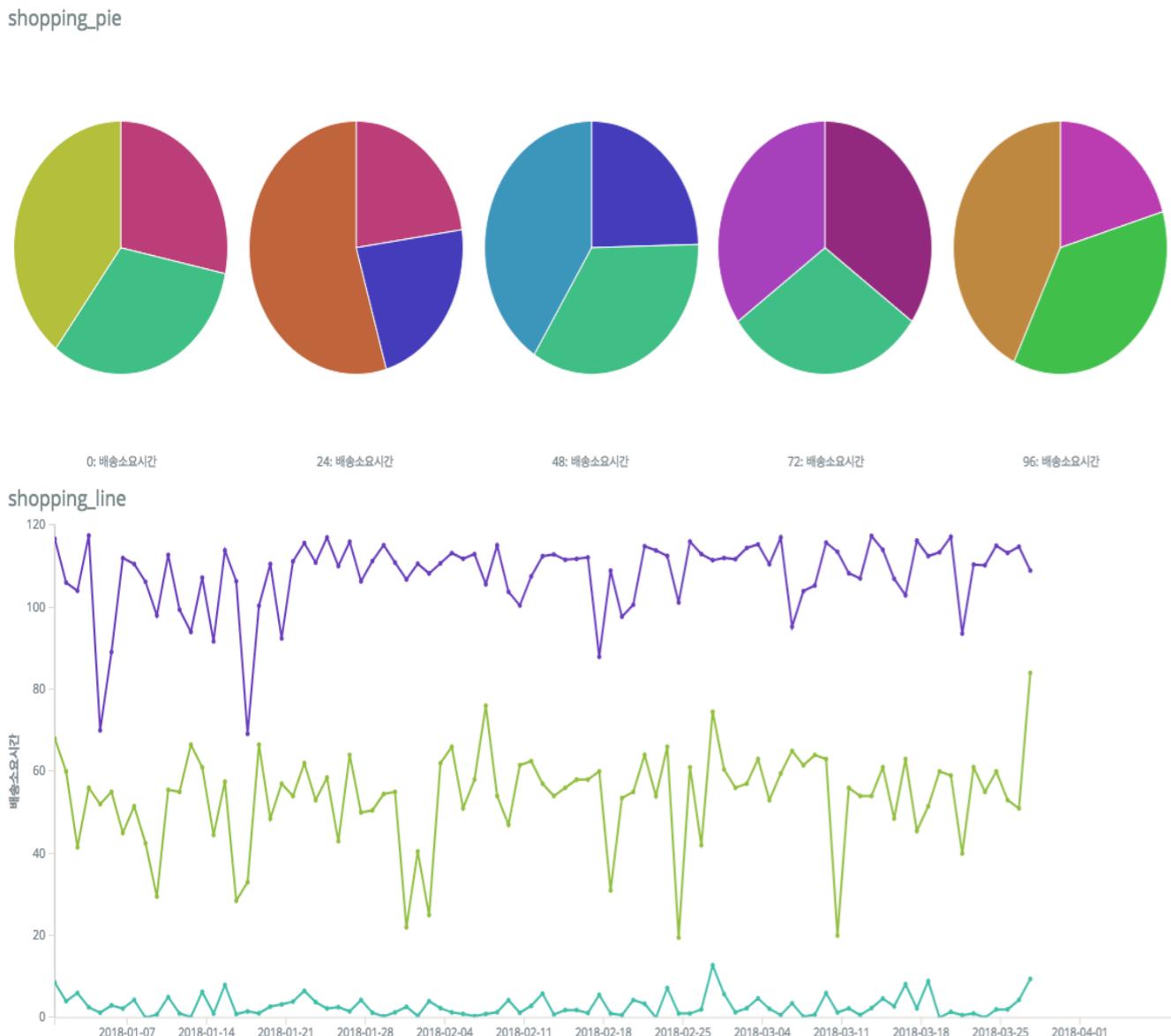
# 아래의 Dashboard에서 “서울특별시”에 해당하는 데이터만 보고 싶다고 하자



# 전체적인 UI는 유지한채 “서울특별시”에 해당하는 값만 필터링되어 보여진다



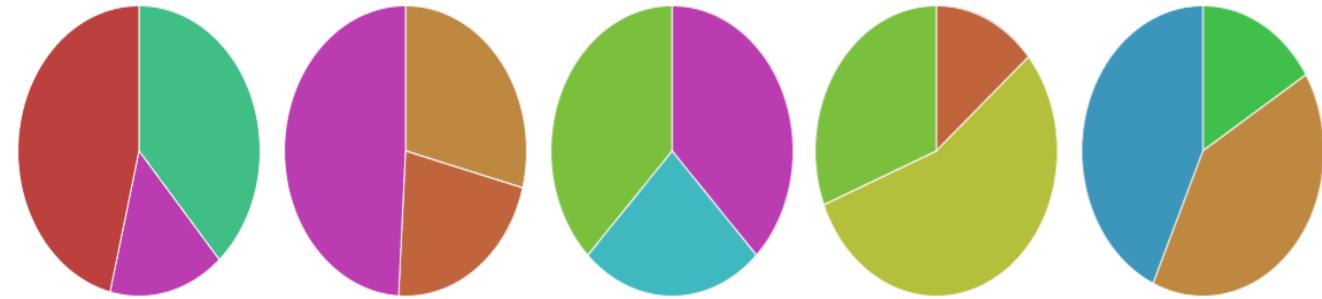
# Coordinate Map의 경우, 특정 boundary 내의 데이터만 필터링 할 수 있다.



# 지도 상에서 드래그한 부분에 해당하는 데이터만 필터링 되어 보인다

선택된 데이터

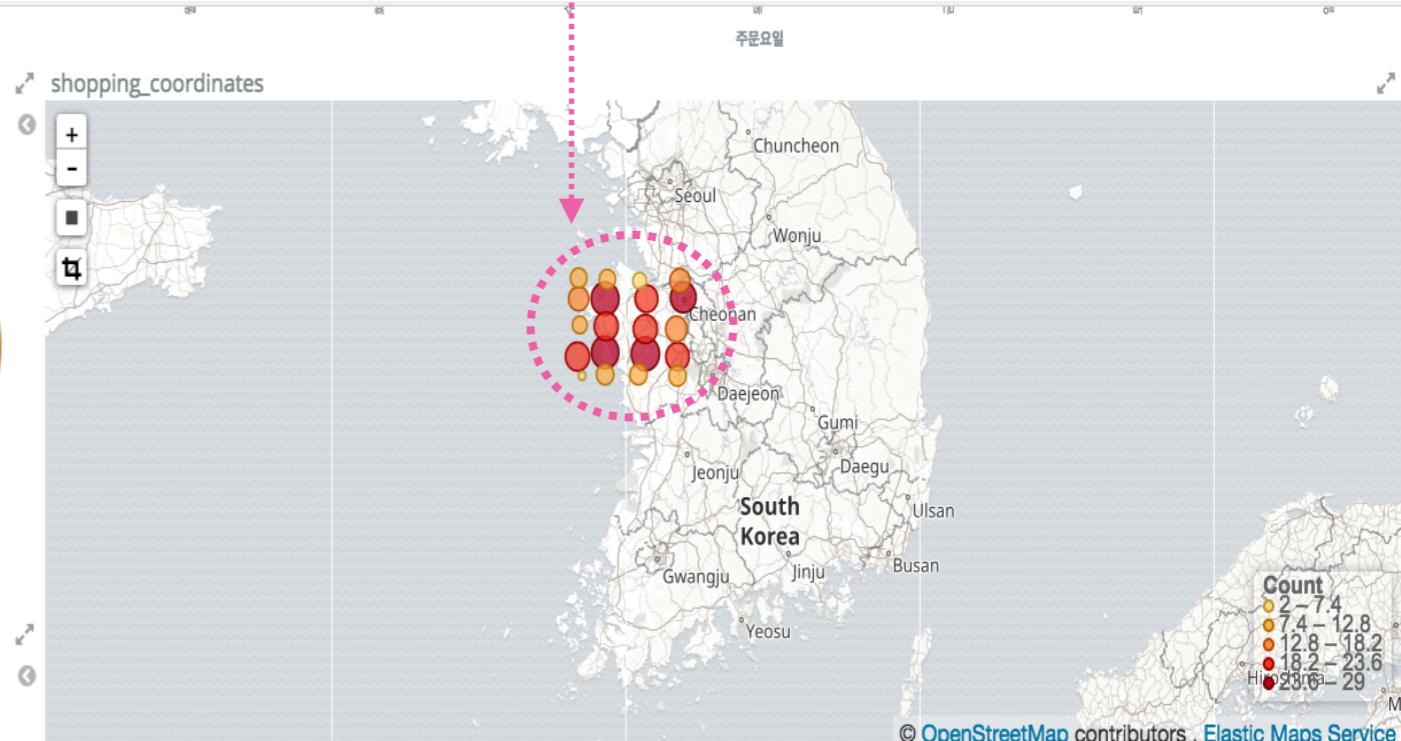
shopping\_pie



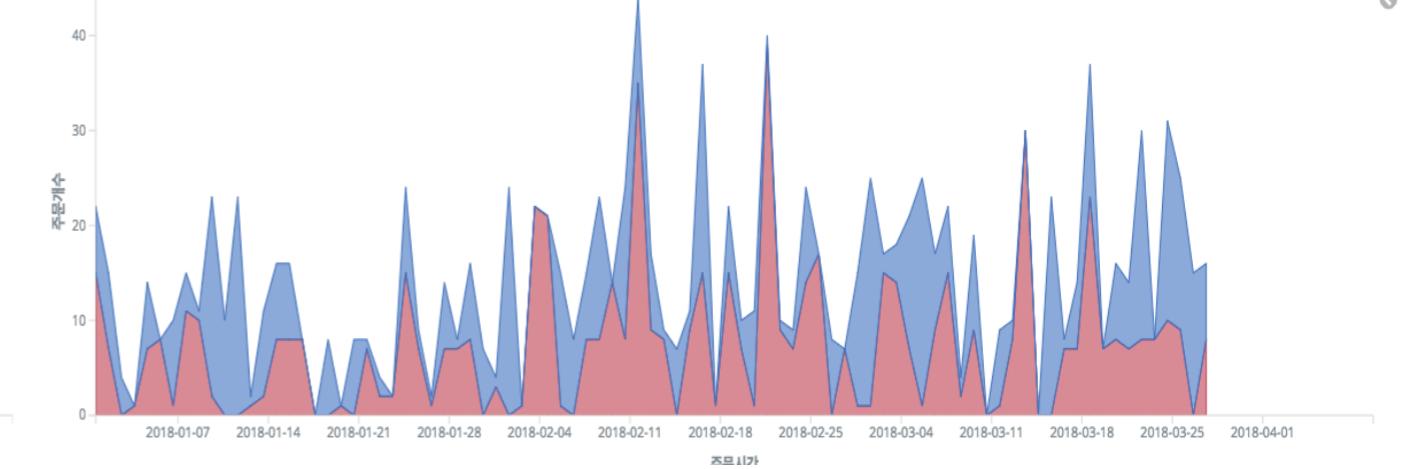
shopping\_line



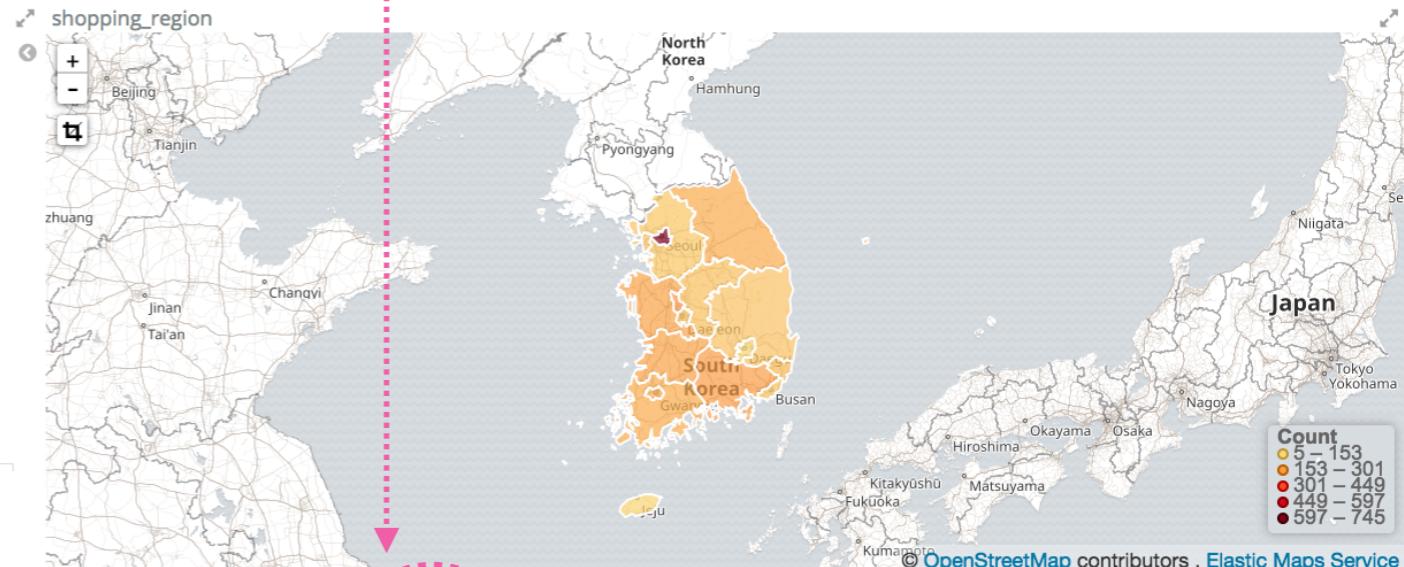
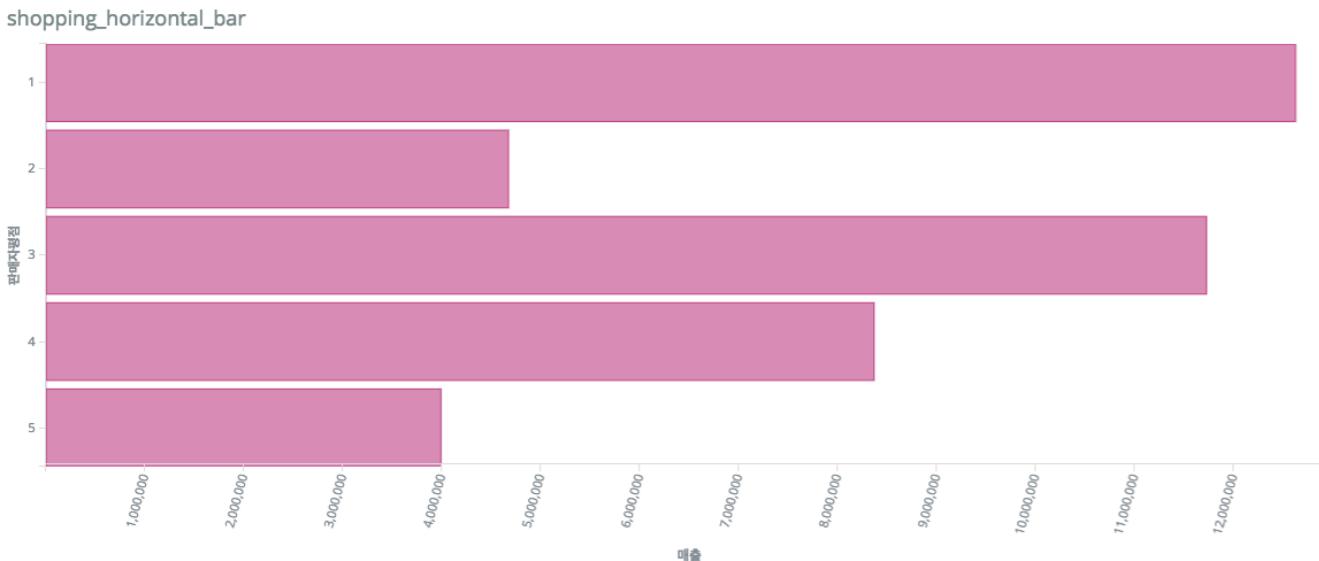
shopping\_coordinates



shopping\_area

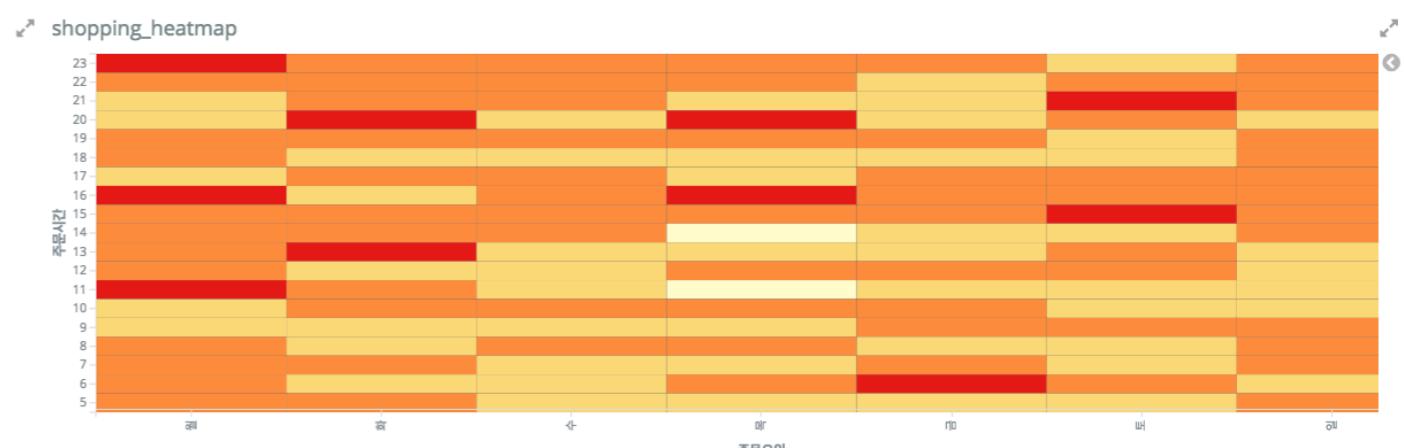


## 시계열 데이터에서 자세히 보고 싶은 특정 구간을 선택해보자

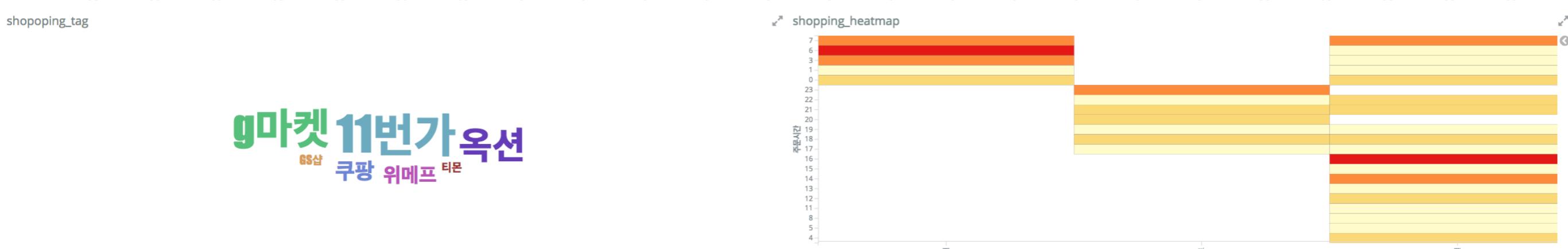
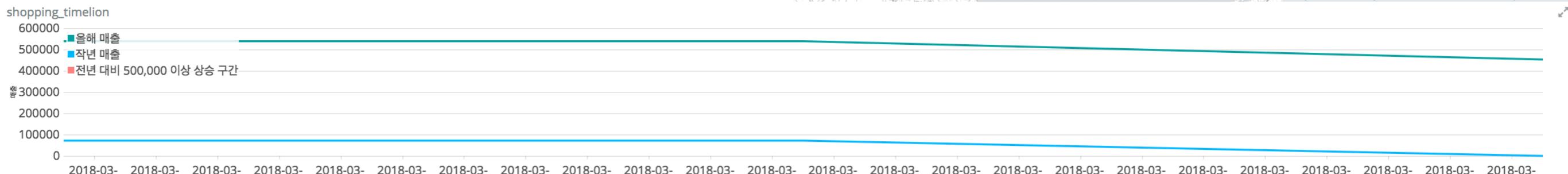
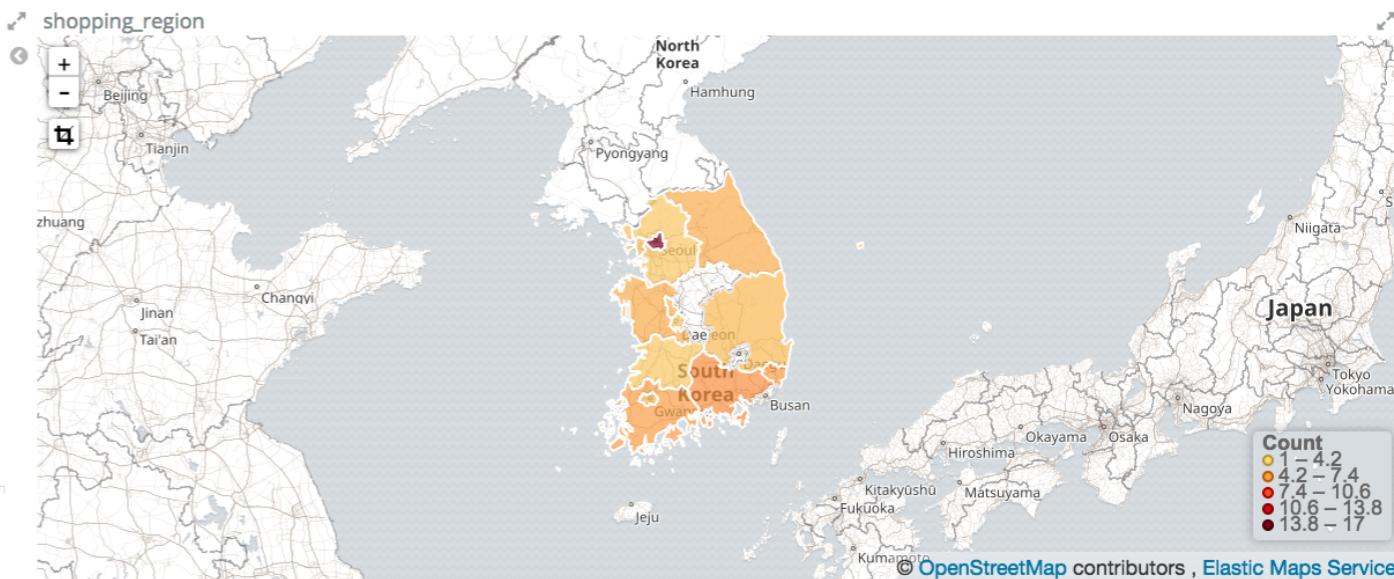
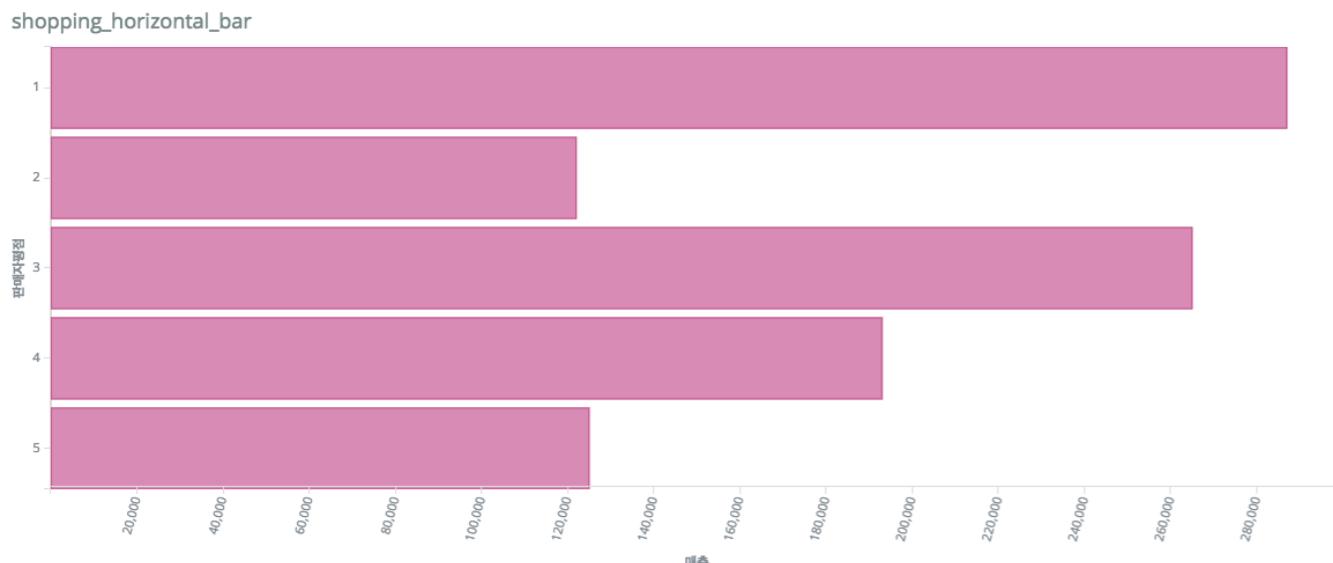


shopoping\_tag

g마켓 11번가 쿠팡  
GS 삼성 옥션 위메프 티몬

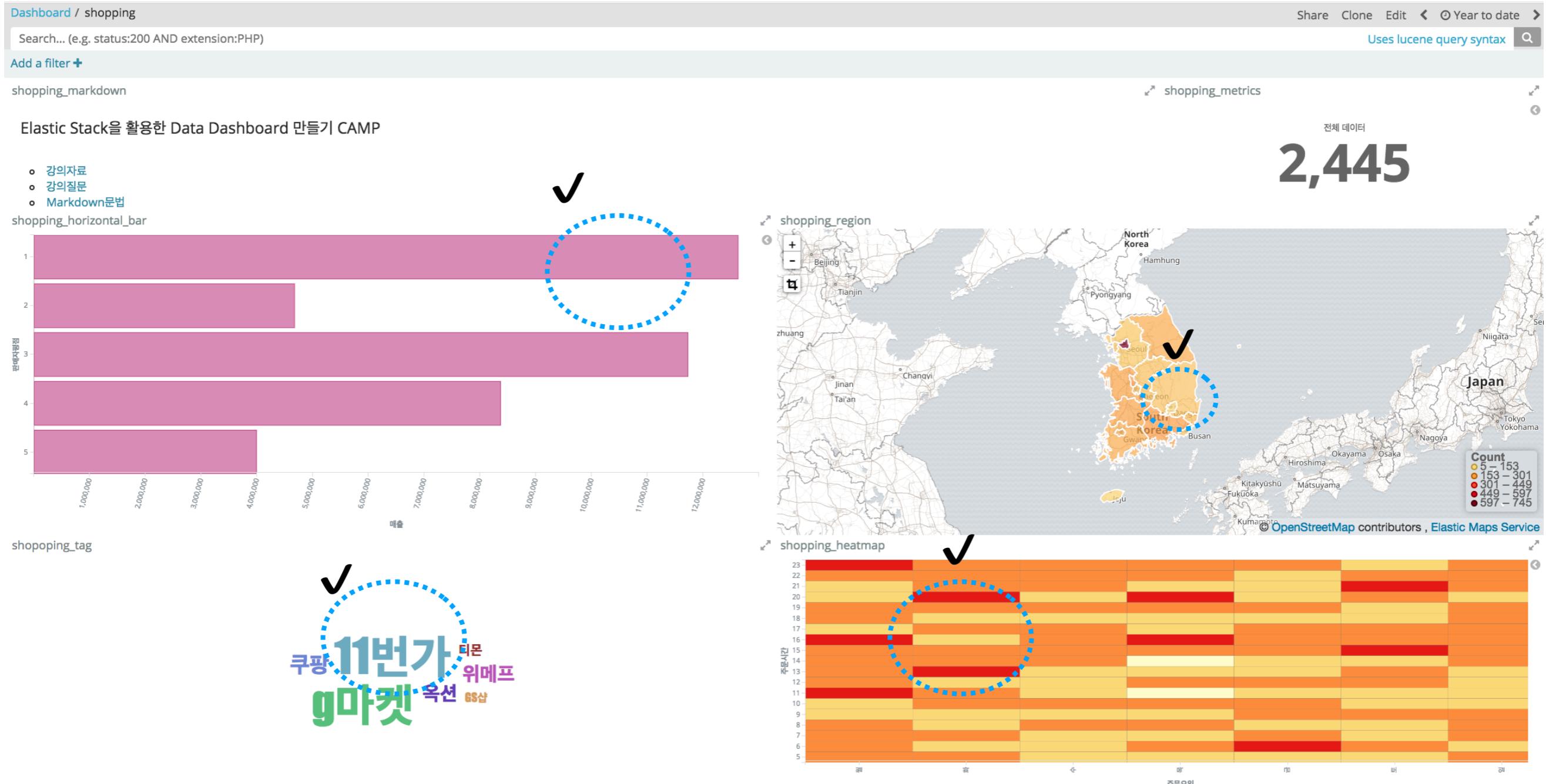


# 선택된 기간 동안의 데이터가 선별되어 보여진다



g마켓 11번가 옵션  
GS샵 쿠팡 위메프 티몬

# Dashboard 상의 여러 visualization을 클릭하면서 Interactive한 Dashboard를 경험해보자 ✎



실시간으로 데이터가 들어오는 경우 **Auto Refresh**를 이용하자

# Auto Refresh 기능을 이용하자

(실제 사연)

Dashboard / shopping

Refresh Interval

Off    5 seconds    1 minute    1 hour  
10 seconds    5 minutes    2 hour  
30 seconds    15 minutes    12 hour  
45 seconds    30 minutes    1 day

Search... (e.g. status:200 AND extension:PHP)

Add a filter +

shopping\_markdown

shopping\_metrics

Elastic Stack을 활용한 Data Dashboard 만들기 CAMP

전체 데이터

2,445

1. Auto-refresh 선택

2. Refresh 간격 설정

① 강의자료  
② 강의질문  
③ Markdown문법

shopping\_horizontal\_bar

판매자별 순위

매출

① 1,000,000 ~ 12,000,000  
② 1,000,000 ~ 12,000,000  
③ 1,000,000 ~ 12,000,000  
④ 1,000,000 ~ 12,000,000  
⑤ 1,000,000 ~ 12,000,000

shopping\_region

Count

© OpenStreetMap contributors, Elastic Maps Service

The screenshot shows a dashboard titled 'Dashboard / shopping'. At the top right, there are several buttons: 'Share', 'Clone', 'Edit', and 'Auto-refresh' (which is highlighted with a red dashed box). Below these are buttons for 'Year to date' and navigation arrows. The main area contains a search bar, a 'Refresh Interval' dropdown menu (with 'Off' selected), and a large number '2,445' representing total data. On the left, there's a horizontal bar chart titled 'shopping\_horizontal\_bar' showing sales by seller rank. On the right, there's a map titled 'shopping\_region' showing data distribution across regions, with a legend for 'Count'.

## dashboard를 공유하자

	dashboard 공유하기 전 변경 사항	dashboard 공유 후 변경 사항
saved dashboard	반영 o	반영 o
Snapshot	반영 o	반영 x (url이 변경됨)

# dashboard를 공유하자

## 1. Share 선택

Dashboard / shopping

### Saved Dashboard

You can share this URL with people to let them load the most recent saved version of this dashboard.

**Embedded iframe**

```
<iframe src="http://13.125.153.139:5601/app/kibana#/dashboard/AV-a4D_kMbJMXoGxGOM2?embed=true&_g=(refreshInterval:(display:Off,pause:If,viewMode:Normal))">
```

**Link**

[http://13.125.153.139:5601/app/kibana#/dashboard/AV-a4D\\_kMbJMXoGxGOM2?\\_g=\(refreshInterval%3A\(display%3AOFF%2CviewMode%3ANormal\)\)](http://13.125.153.139:5601/app/kibana#/dashboard/AV-a4D_kMbJMXoGxGOM2?_g=(refreshInterval%3A(display%3AOFF%2CviewMode%3ANormal)))

**Snapshot**

Share Snapshot

Snapshot URLs encode the current state of the dashboard in the URL itself. Edits to the saved dashboard won't be visible via this URL.

**Embedded iframe**

```
<iframe src="http://13.125.153.139:5601/app/kibana#/dashboard/AV-a4D_kMbJMXoGxGOM2?embed=true&_g=(refreshInterval:(display:Off,pause:If,viewMode:Normal))">
```

**Link**

[http://13.125.153.139:5601/app/kibana#/dashboard/AV-a4D\\_kMbJMXoGxGOM2?\\_g=\(refreshInterval:\(display:Off,pause:If,viewMode:Normal\)\)](http://13.125.153.139:5601/app/kibana#/dashboard/AV-a4D_kMbJMXoGxGOM2?_g=(refreshInterval:(display:Off,pause:If,viewMode:Normal)))

Search... (e.g. status:200 AND extension:PHP)

Add a filter +

shopping\_markdown

shopping\_metrics

Elastic Stack을 활용한 Data Dashboard 만들기 CAMP

전체 데이터

2,445

shopping\_horizontal\_bar

판매자평점	판매량
1	100
2	20
3	100
4	30
5	10

shopping\_region

North Korea  
Hamhung  
Pyongyang  
Changyi  
South Korea  
Seoul  
Daegu  
Busan  
Gwangju  
Japan  
Tokyo  
Osaka  
Nagoya  
Okayama  
Hiroshima

Collapse

# Embedded iFrame을 공유하자

Dashboard / shopping

Share saved dashboard  아래 url를 활용한 간단한 html 파일을 만들자

You can share this URL with people to let them load the most recent saved version of this dashboard.

Embedded iframe   
`<iframe src="http://13.125.153.139:5601/app/kibana#/dashboard/AV-a4D_kMbjMXoGxGOM2?embed=true&_g=(refreshInterval:(display:Off,pause:If,va`

Add to your HTML source. Note that all clients must be able to access Kibana.

Link   
`http://13.125.153.139:5601/app/kibana#/dashboard/AV-a4D_kMbjMXoGxGOM2?_g=(refreshInterval:(display:Off,pause:If,va`

Share Snapshot   
Snapshot URLs encode the current state of the dashboard in the URL itself. Edits to the saved dashboard won't be visible via this URL.

Embedded iframe   
`<iframe src="http://13.125.153.139:5601/app/kibana#/dashboard/AV-a4D_kMbjMXoGxGOM2?embed=true&_g=(refreshInterval:(display:Off,pause:If,va`

Add to your HTML source. Note that all clients must be able to access Kibana.

Link   
`http://13.125.153.139:5601/app/kibana#/dashboard/AV-a4D_kMbjMXoGxGOM2?_g=(refreshInterval:(display:Off,pause:If,va`

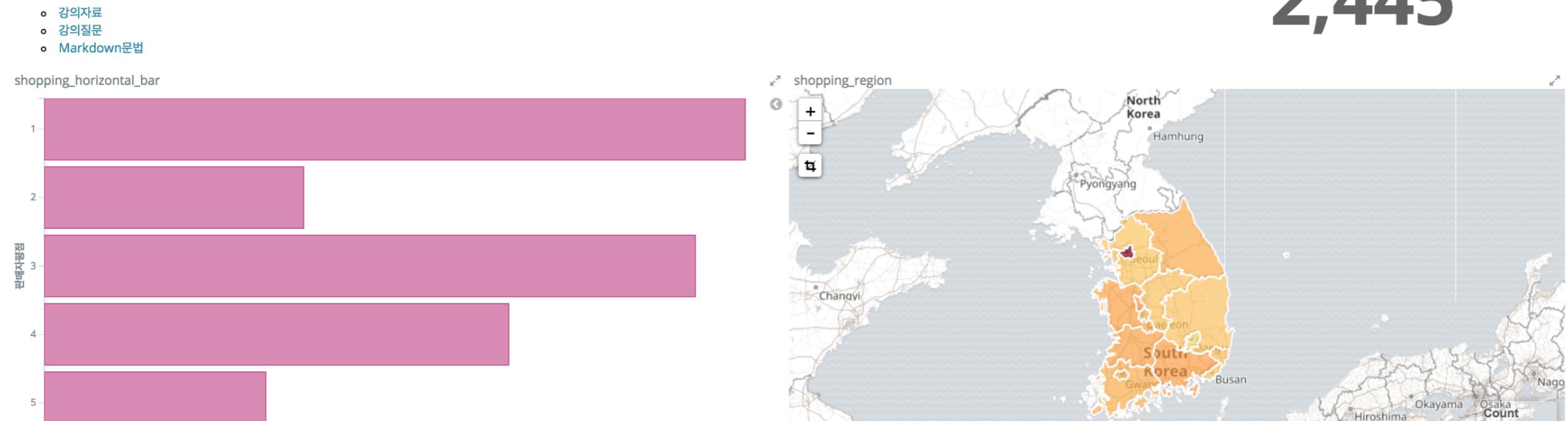
We recommend sharing shortened snapshot URLs for maximum compatibility. Internet Explorer has URL length restrictions, and some wiki and markup parsers don't do well with the full-length version of the snapshot URL, but the short URL should work great.

Search... (e.g. status:200 AND extension:PHP)  Uses lucene query syntax

Add a filter 

shopping\_markdown  shopping\_metrics 

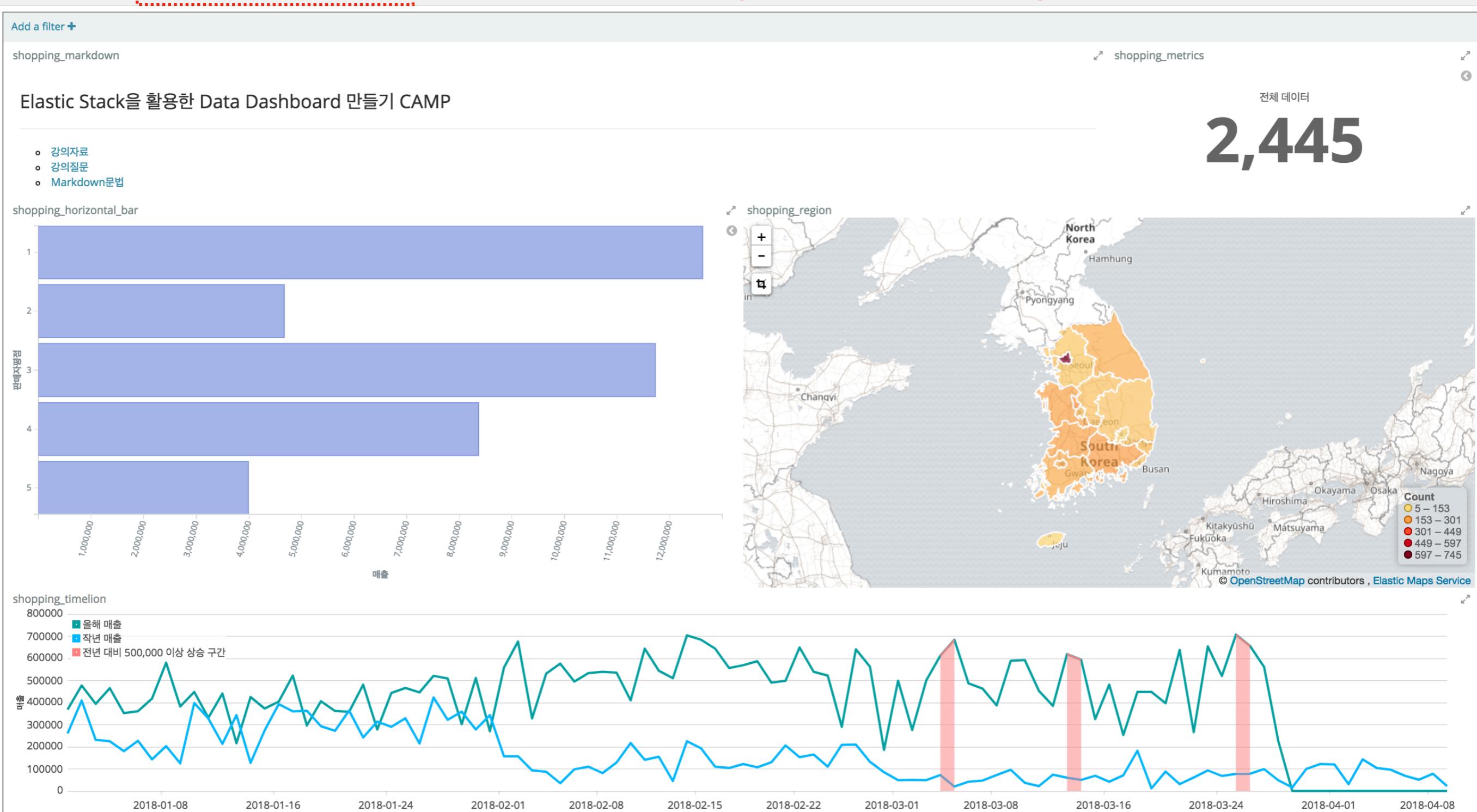
## Elastic Stack을 활용한 Data Dashboard 만들기 CAMP



## Embedded iFrame을 공유하자

① 13.125.153.139:8000/test.html

 html 파일을 실행해보자 (local 또는 web-server 선택)



Filtering by Field 

**Dashboard는 만들었는데  
원하는 조건의 데이터만 보고 싶으면?**

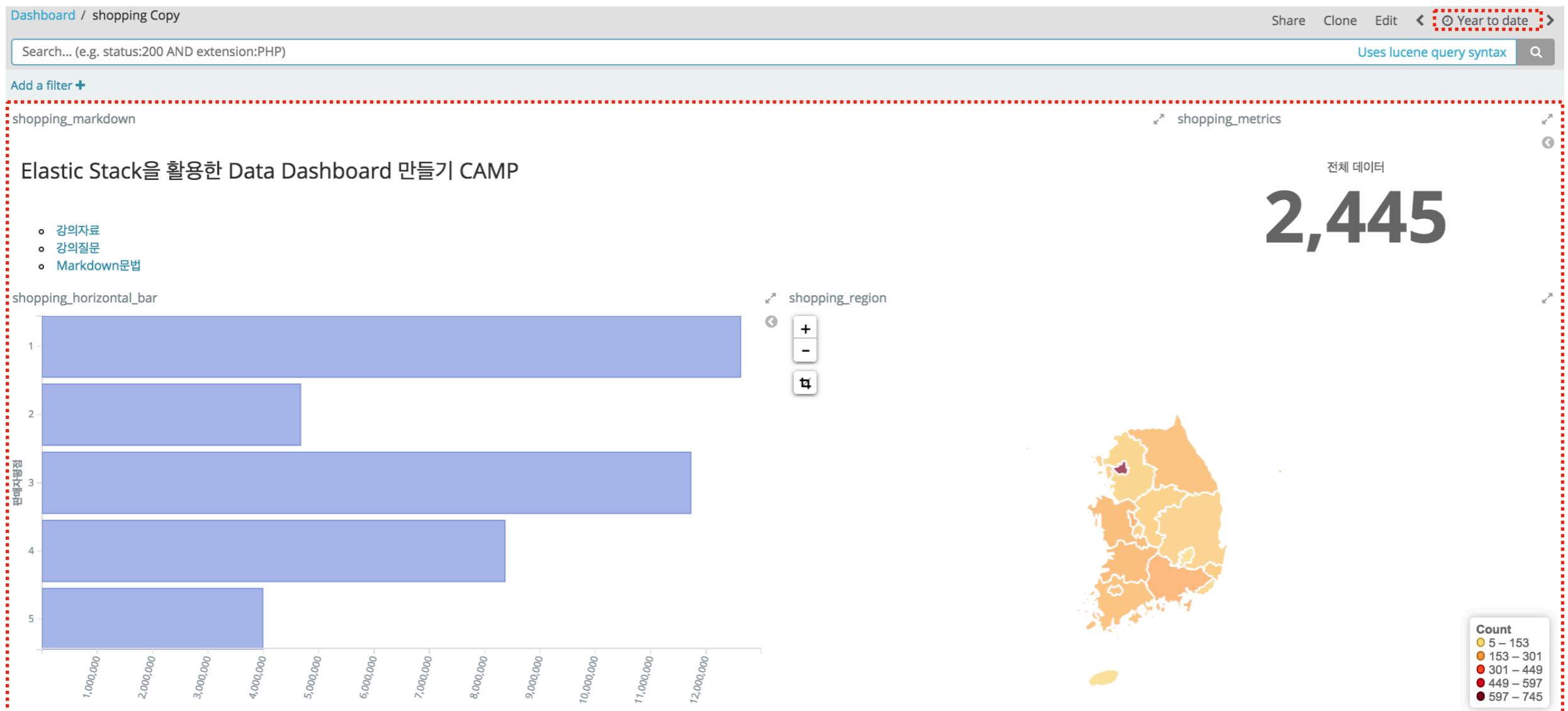
# 우선 Dashboard를 열자

The screenshot shows the Kibana interface with the title "Dashboard" at the top. On the left, there is a sidebar with the Kibana logo and several navigation options: Discover, Visualize, Dashboard (which is selected), Timelion, Dev Tools, and Management. At the bottom of the sidebar is a "Collapse" button. The main area displays a list of dashboards. At the top of this list is a search bar with the placeholder "Search..." and a blue "+" button. To the right of the search bar are navigation arrows for "1-2 of 2". The list itself has two items:

Name	Description
shopping	
{id}_shopping 선택	

The third item in the list, "{id}\_shopping 선택", is highlighted with a red dashed box around its name. To the left of this item is a red hand icon pointing towards it. The text "shopping Copy" is also visible near the highlighted item.

# Dashboard를 확인하자



전체 Documents 중에서 Time Picker 구간에 속한 Documents만 보여준다.

만약에 다른 조건을 추가하고 싶다면? 예를 들어, “경상도” 데이터만 보고 싶으면 어떻게 할까?

# 원하는 조건의 데이터만 조회하기 위해 필터를 적용했다

Dashboard / shopping Copy Share Clone Edit < ⏪ Year to date ⏩

Search... (e.g. status:200 AND extension:PHP) Uses lucene query syntax

경상도 ~~As filter + “경상도”라는 필터가 적용됐다~~ Actions ▾

shopping\_markdown shopping\_metrics

Elastic Stack을 활용한 Data Dashboard 만들기 CAMP

전체 데이터 263

○ 강의자료  
○ 강의질문  
○ Markdown문법

shopping\_horizontal\_bar

판매지역	매출
1	1,200,000
2	500,000
3	1,400,000
4	600,000
5	200,000

shopping\_region

Count

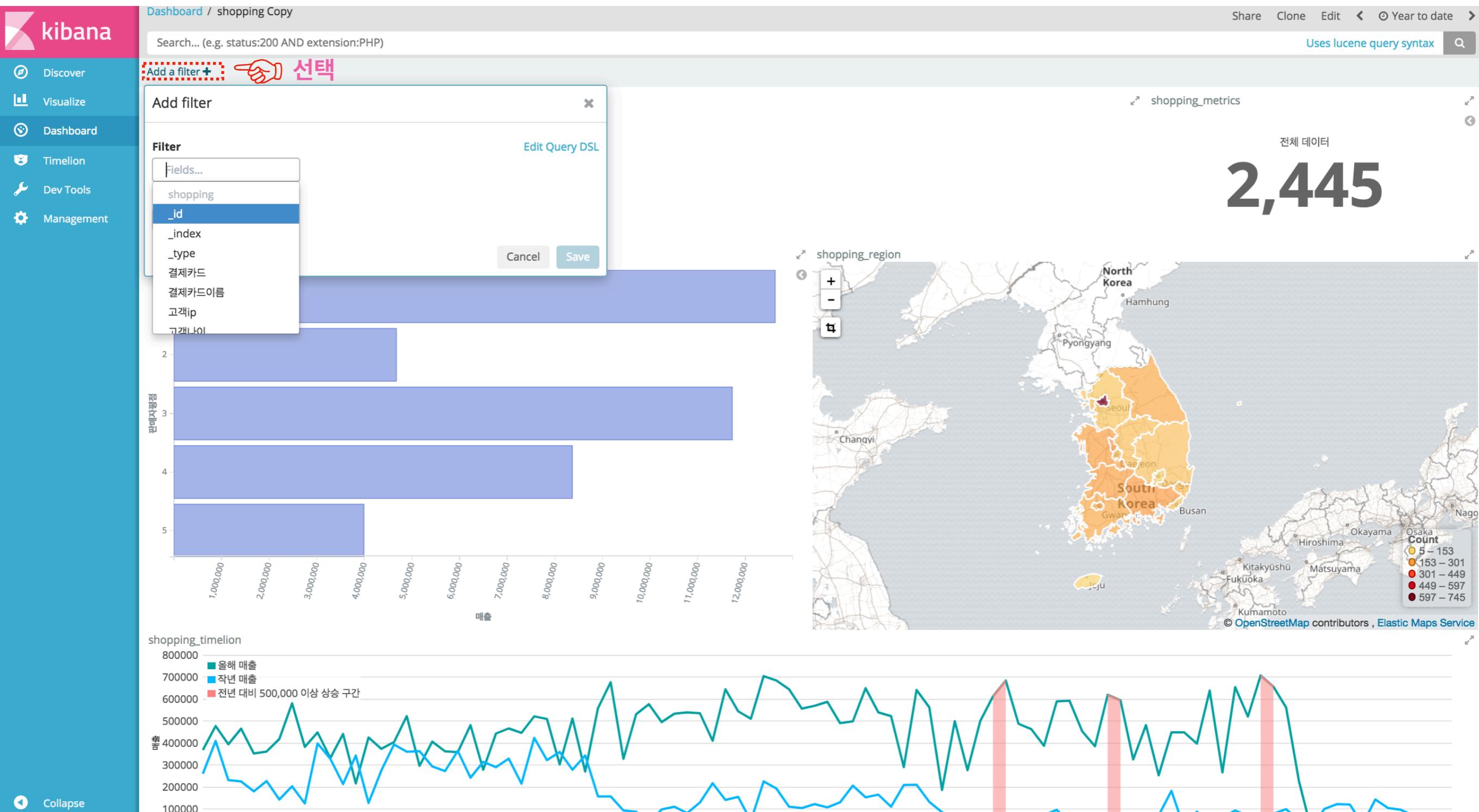
Count Range
71 - 95.2
95.2 - 119.4
119.4 - 143.6
143.6 - 167.8
167.8 - 192

경상남도와 경상북도만 표시됐다

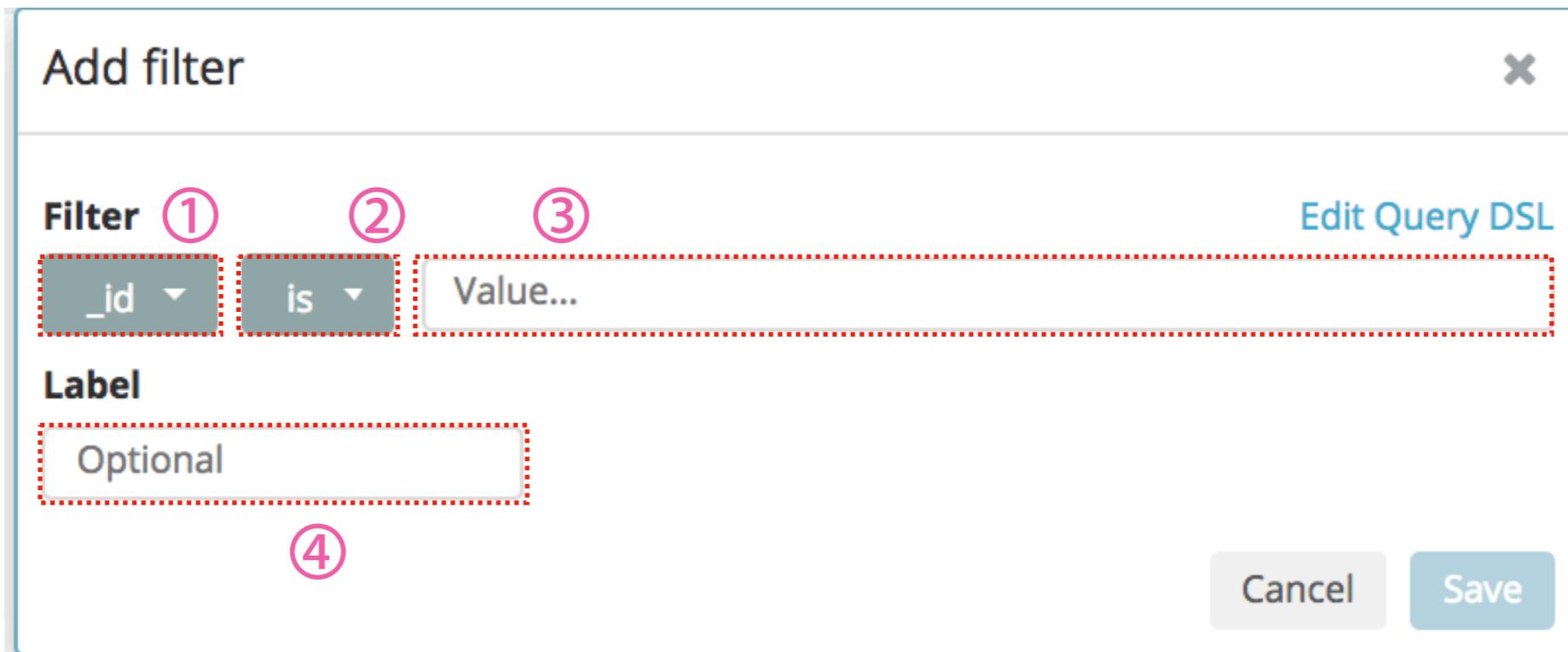
© OpenStreetMap contributors, Elastic Maps Service

**Filter**를 이용하면 **특정 조건을 만족하는**  
데이터만 선별하여 Dashboard에 시각화할 수 있다.  
그렇다면 어떻게 사용할까?

# Filter를 실행하자



## Filter의 사용법을 익히자



- ① Filter 적용할 Field 선택
- ② 적용할 Operator 선택 (다음 페이지 참조)
- ③ Filter에 적용하려는 Value 입력
- ④ (여러 Filter 구분하기 위한) 이름 입력

## Operator 설명

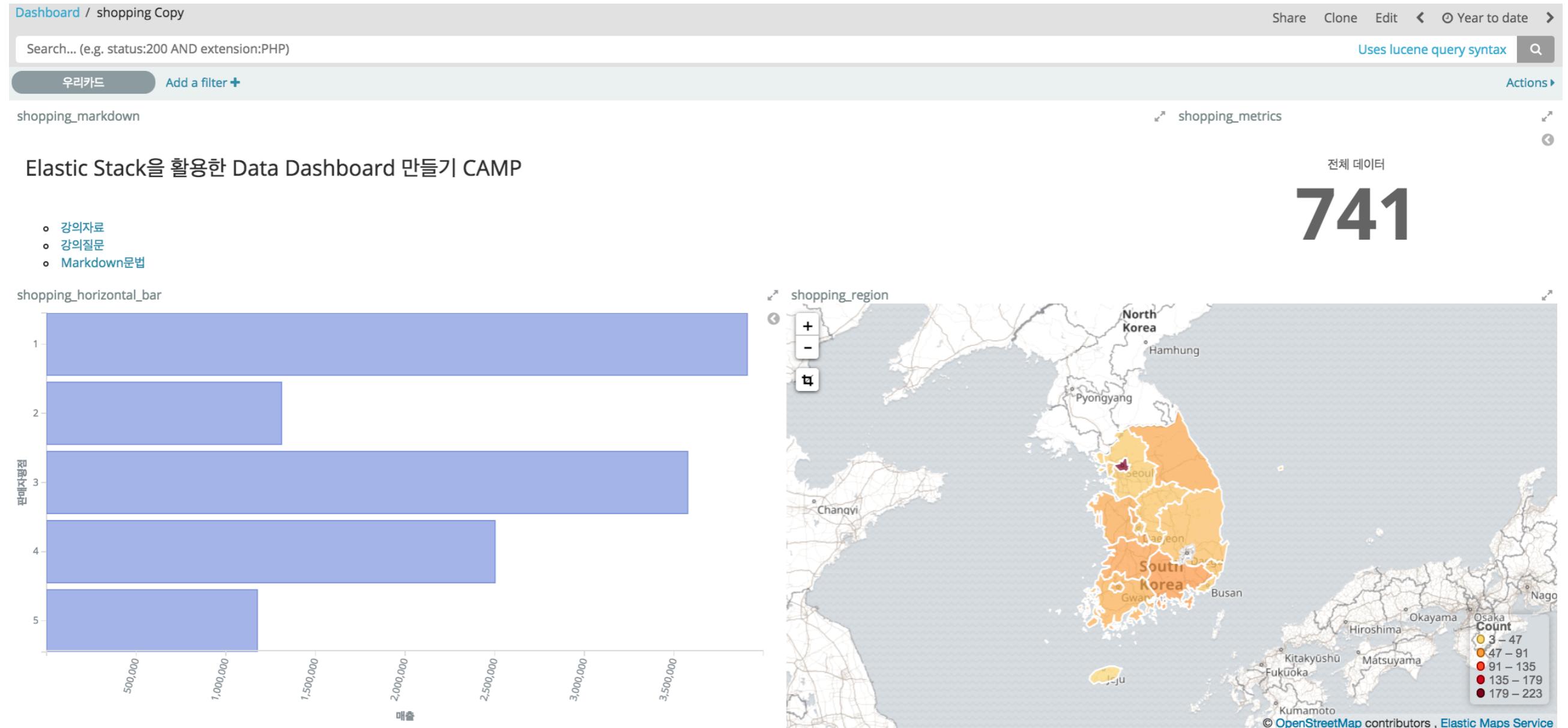
Operator	역할
is	Field의 Value가 입력한 값과 일치하는 Documents 선택
is not	Field의 Value가 입력한 값과 일치하지 않는 Documents 선택
is one of	Field의 Value가 입력한 값 중에 존재하는 Documents 선택
is not one of	Field의 Value가 입력한 값 중에 존재하지 않는 Documents 선택
exists	Field가 적어도 한 개의 non-null 값을 가지는 Documents 선택
does not exist	Field가 존재하지 않거나 null 값만 가지는 Documents 선택
is between	Field의 Value가 입력한 값 사이에 존재하는 Documents 검색
is not between	Field의 Value가 입력한 값 사이에 존재하지 않는 Documents 검색

## Operator 설명 - **is**

결제카드가 “우리” 인 데이터만 조회



# Operator 설명 - is



## Operator 설명 - **is one of**

“구매사이트”가 11번가, 옥션, 쿠팡 중 하나인 데이터



## Operator 설명 - is one of

Dashboard / shopping Copy

Share Clone Edit < ⏪ Year to date ⏩

Search... (e.g. status:200 AND extension:PHP)

Uses lucene query syntax

Actions ▾

즐겨찾기 Add a filter +

shopping\_markdown

shopping\_metrics

Elastic Stack을 활용한 Data Dashboard 만들기 CAMP

전체 데이터

1,289

- 강의자료
- 강의질문
- Markdown문법

shopping\_horizontal\_bar

A horizontal bar chart titled "shopping\_horizontal\_bar". The y-axis is labeled "판매지역별" and has values 1, 2, 3, 4, and 5. The x-axis is labeled "매출" and ranges from 1,000,000 to 6,000,000. There are five blue bars representing different regions.

판매지역별	매출
1	~2,500,000
2	~3,500,000
3	~5,500,000
4	~4,500,000
5	~1,500,000

shopping\_region

A map of South Korea titled "shopping\_region". The map uses a color-coded legend to represent different ranges of data. The legend includes:

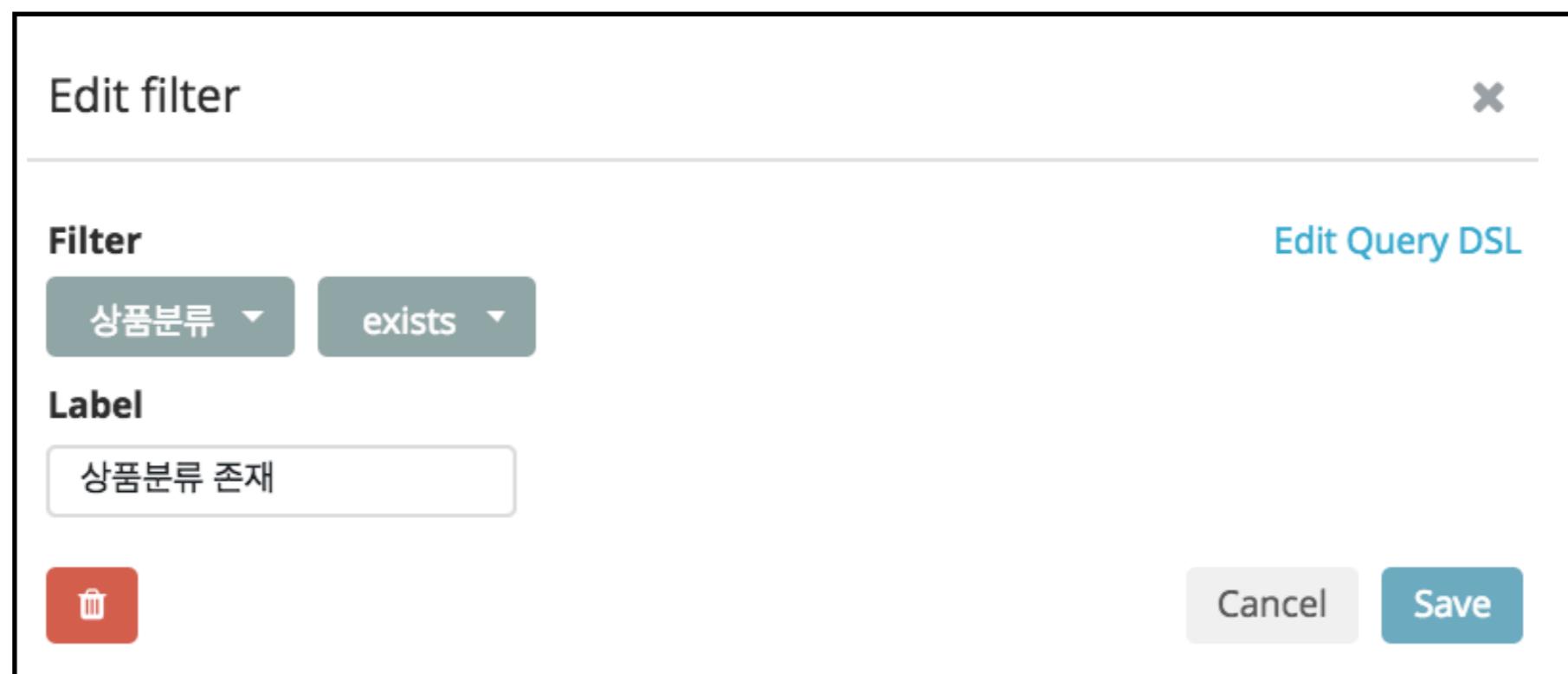
- Count: 2 - 78.6 (light yellow)
- 78.6 - 155.2 (orange)
- 155.2 - 231.8 (red)
- 231.8 - 308.4 (dark red)
- 308.4 - 385 (purple)

Labels on the map include North Korea, Hamhung, Pyongyang, Seoul, Gwangju, Busan, and various cities in the South Korean regions of Gyeonggi-do, Gyeongsangnam-do, and Gyeongsangbuk-do.

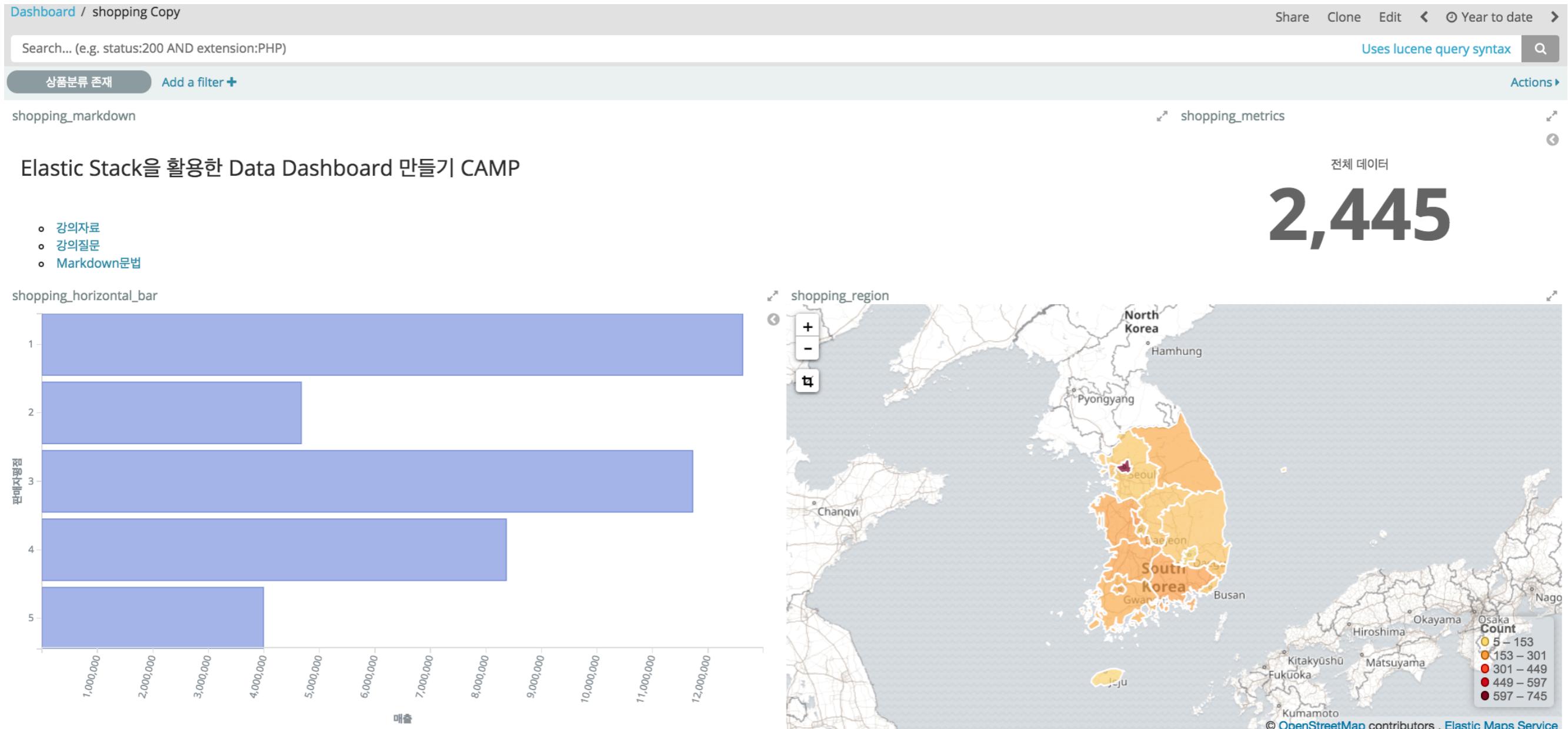
OpenStreetMap contributors Elastic Maps Service

## Operator 설명 - exists

“상품분류”필드에 적어도 한 개의 null이 아닌 값이 있는 데이터 조회



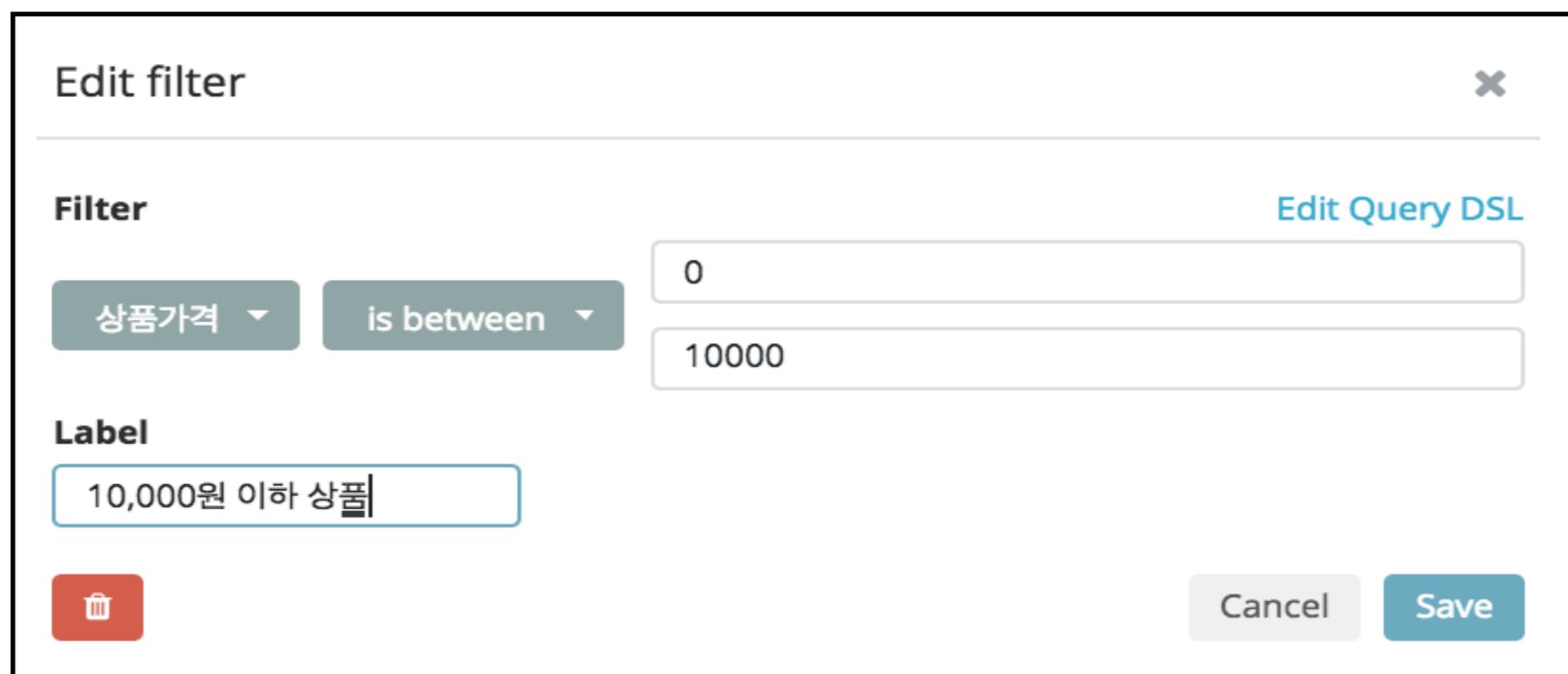
# Operator 설명 - exists



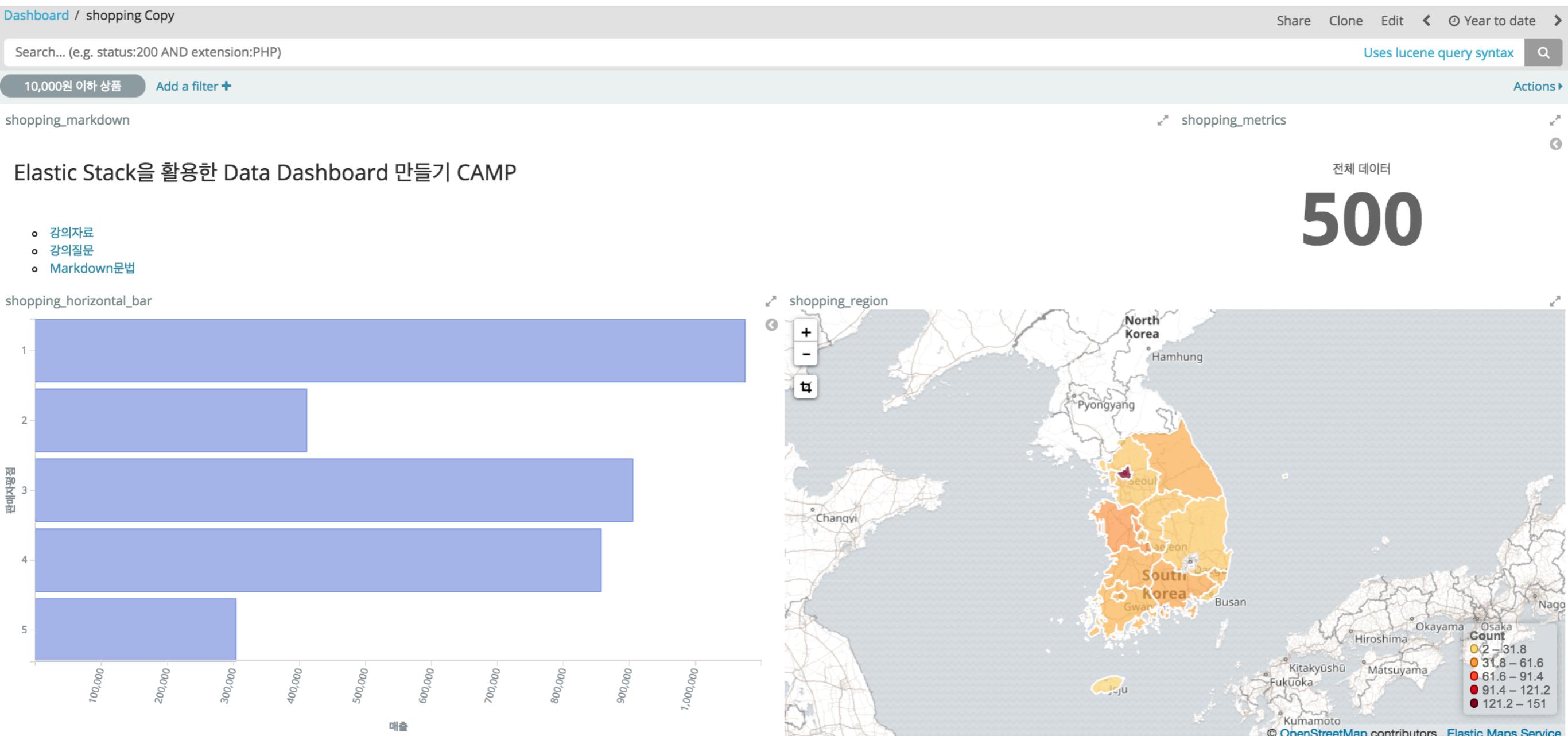
\* shopping index의 “상품분류” Field는 null 값이 없기에 전체 Documents가 표시된다

## Operator 설명 - **is between**

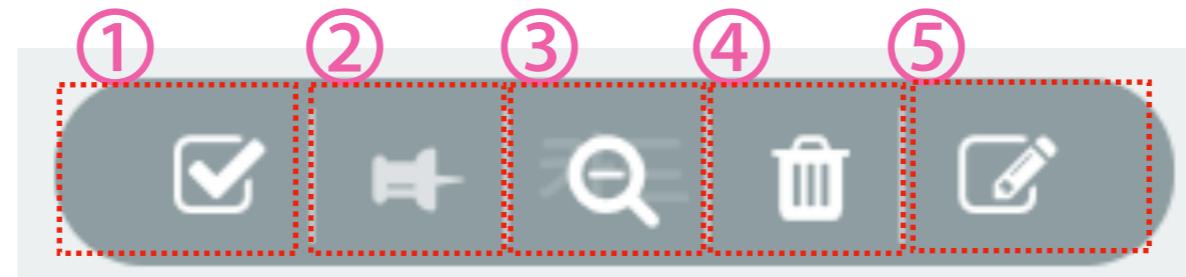
“상품가격”이 10,000 이하인 데이터 조회



# Operator 설명 - is between



## 필터에 마우스오버하면...



- ① 필터 적용 <=> 필터 적용 해제
- ② 필터 고정 (Discover, Visualize, Dashboard)
- ③ 필터 효과 적용 <=> 필터 효과 반대 적용
- ④ 필터 삭제
- ⑤ 필터 수정

## 아래와 같은 Filter를 Dashboard에 적용해보자 ↗

Dashboard : {id}\_shopping  
Time Range : year\_to\_date

1. “고객성별”이 여성인 Data
2. “결제카드”가 우리 또는 국민인 Data
3. “고객성별”이 남성이면서 “연령대”가 20대
4. “구매사이트”가 쿠팡이거나 “상품개수”가 1~3인 Data
5. “결제카드”가 “우”로 시작하는 모든 Data
6. “구매사이트”가 22번가(오타 아니에요)와 유사한 Data

Lucene Query 

**Filter는 사용하기 간단하나 기능이 제한적이다**

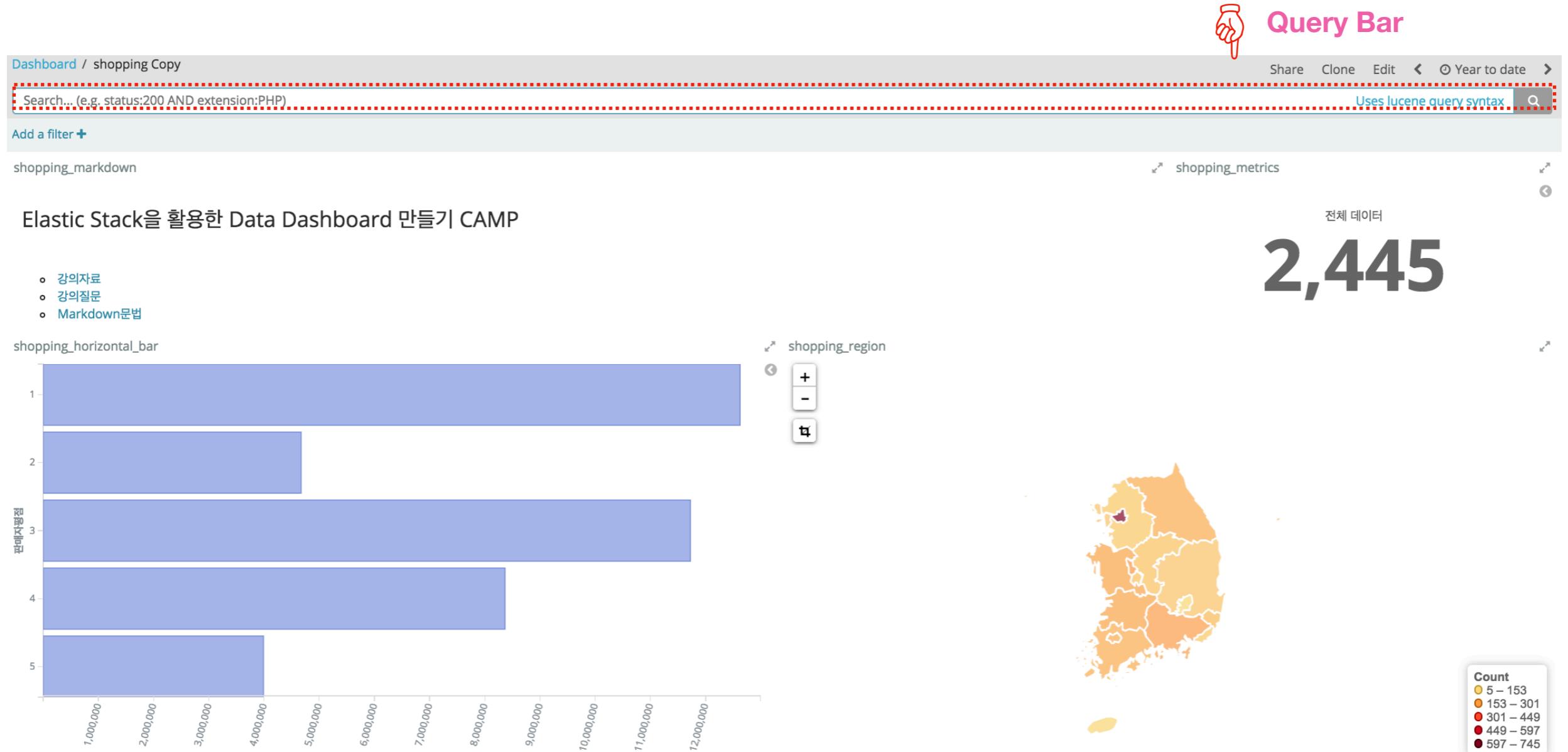
+

**더 스마트한 검색!!**

# 우선 Dashboard를 열자

The screenshot shows the Kibana interface with the title "Dashboard" at the top. On the left, there is a sidebar with the Kibana logo and navigation links: Discover, Visualize, Dashboard (which is selected), Timelion, Dev Tools, and Management. At the bottom of the sidebar is a "Collapse" button. The main area displays a list of dashboards. A search bar at the top of the list has the placeholder "Search...". Below it is a table with two columns: "Name" and "Description". The first row shows "Name" as "shopping" and "Description" as " ". The second row shows "Name" as "[ shopping Copy ]" and "Description" as " {id}\_shopping 선택". A red dashed box highlights the "Name" column of the second row. To the right of the table, there is a blue "+" button and page navigation controls labeled "1-2 of 2" and arrows.

# Query Bar를 확인하자



# **Query Bar에 뭐라고 검색을 해야될까?**

## Lucene Query의 사용법을 익히자 ↗

종류	기능	Query 예시
Keyword 검색	Field에 상관없이 Value 일치하는 Documents 검색	여성
Field Match 검색	특정 Field의 Value가 일치하는 Documents 검색	고객성별:여성
Exact Field Match 검색	특정 Value가 정확히 모두 일치하는 Documents 검색	배송메모:"상품 이상"
Must be 검색	특정 Field가 존재하는 Documents 검색	_exists_:구매사이트
Must not be present 검색	특정 Field가 존재하지 않는 Documents 검색	_missing_:구매사이트
AND 검색	특정 조건들을 모두 만족하는 Documents 검색	고객성별:여성 AND 상품분류:셔츠
OR 검색	특정 조건들 중 적어도 1개를 만족하는 Documents 검색	고객성별:남성 OR 상품분류:셔츠
NOT 검색	특정 조건을 만족하지 않는 Documents 검색	NOT 구매사이트:옵션
Term 검색	조건 중 적어도 하나라도 만족하는 Documents 검색	상품분류: (니트 코트)
Fuzzy 검색	검색어와 유사한 Documents 검색	경상복도~
Proximity 검색	검색어의 순서를 변경해서 찾을 수 있는 Documents 검색	배송메모:"내에 시간 배송 못함"~2
Numeric Value 검색	Numeric Field Value로 Documents 검색	상품가격:>5000
Range 검색	Field의 Value가 입력한 값 사이에 존재하는 Documents 검색	고객나이 : [10 TO 30]
Wildcard ? 검색	Wildcard ?(한글자)를 활용해서 Documents 검색	서?특별시
Wildcard * 검색	Wildcard *(모든글자)를 활용해서 Documents 검색	쿠*

## Search Type - Keyword

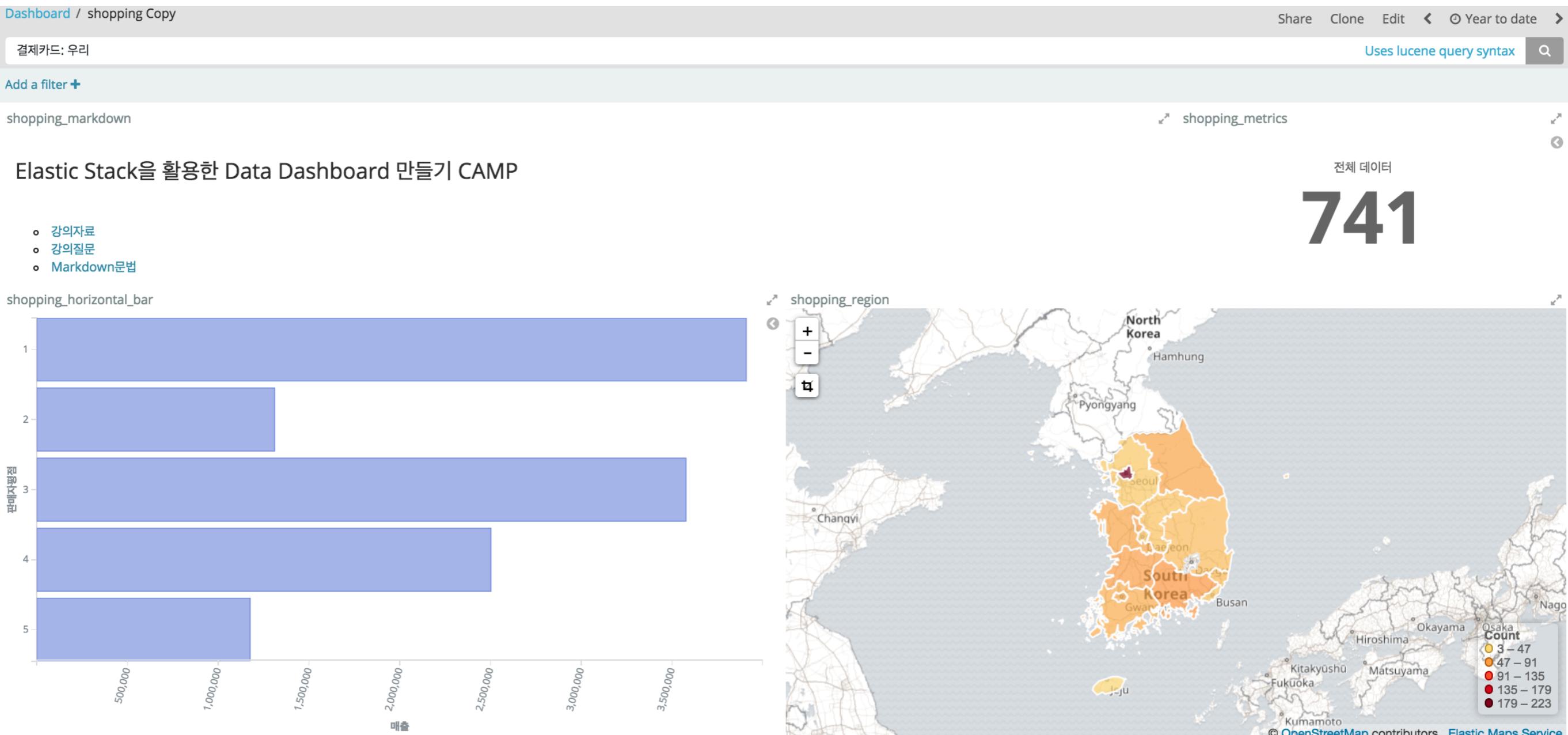
결제카드가 “우리” 인 데이터만 조회



The screenshot shows a search interface with the following elements:

- Top bar: Dashboard / shopping Copy, Share, Clone, Edit, < Year to date >
- Search bar: 결제카드: 우리, Uses lucene query syntax, magnifying glass icon
- Bottom bar: Add a filter +

# Search Type - Keyword



## Search Type - Term

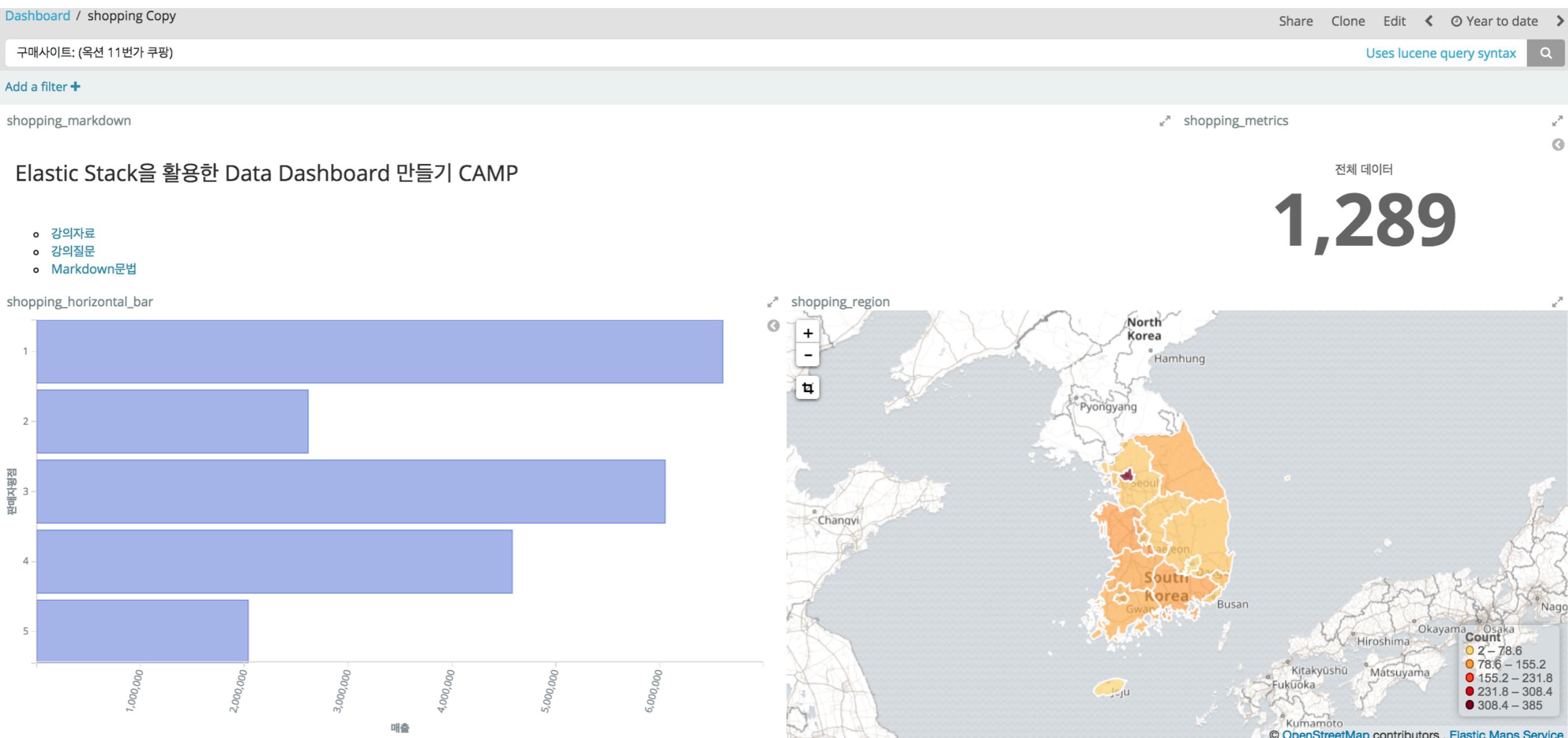
“구매사이트”가 쿠팡, 옥션, 11번가 중 하나인 데이터



The screenshot shows a search interface with the following elements:

- Top navigation: Dashboard / shopping Copy, Share, Clone, Edit, Year to date controls.
- Search bar: 구매사이트: (옥션 11번가 쿠팡) with a note: Uses lucene query syntax and a magnifying glass icon.
- Filter button: Add a filter +

# Search Type - Term



## Search Type - OR

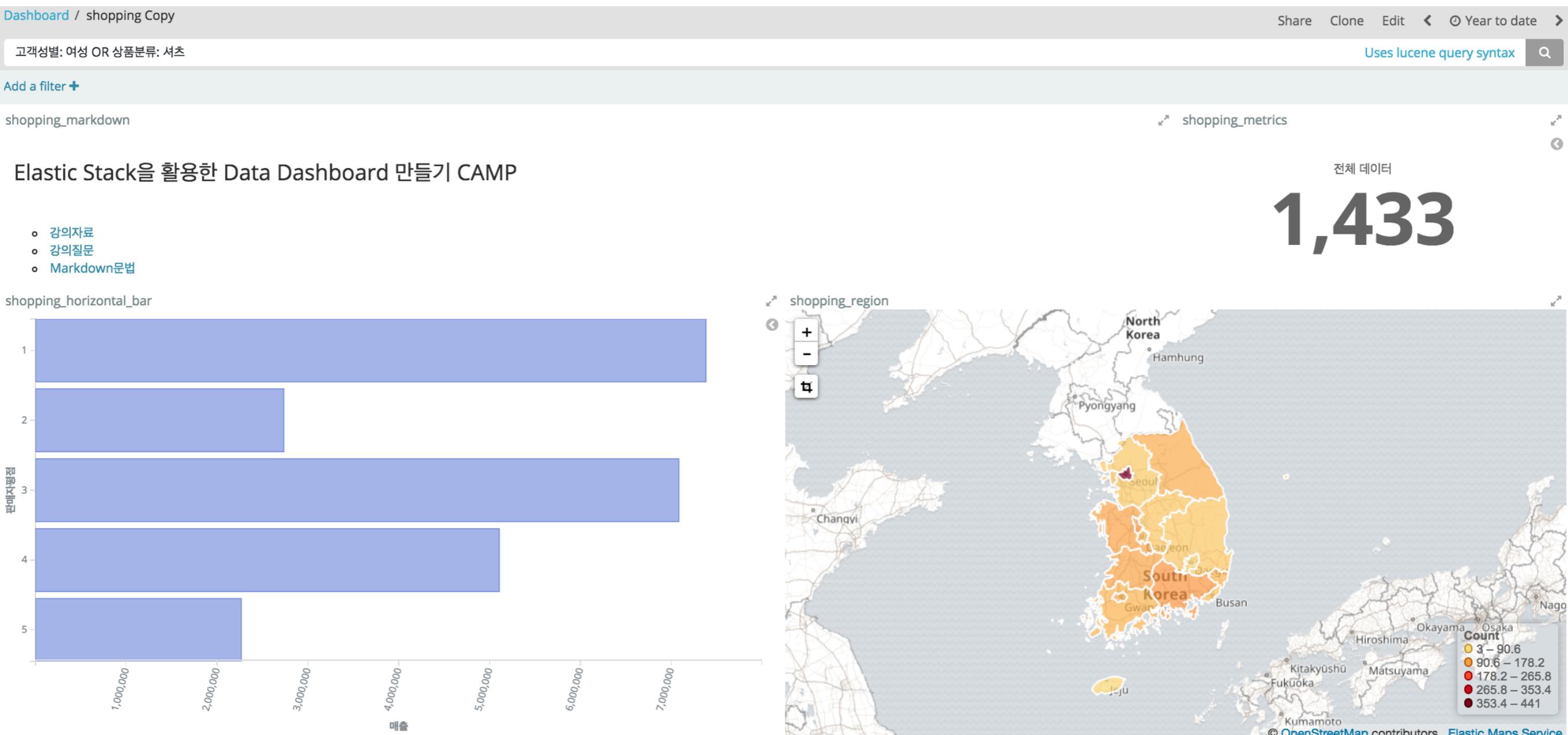
“고객성별”이 “여성”이거나 “상품분류”가 “셔츠”인 데이터



The screenshot shows a search interface with the following elements:

- Top navigation: Dashboard / shopping Copy, Share, Clone, Edit, Year to date.
- Search bar: 고객성별: 여성 OR 상품분류: 셔츠. This text is highlighted in blue, indicating it uses Lucene query syntax.
- Buttons: Add a filter +, Uses lucene query syntax, and a magnifying glass icon.

# Search Type - OR



## Search Type - Fuzzy

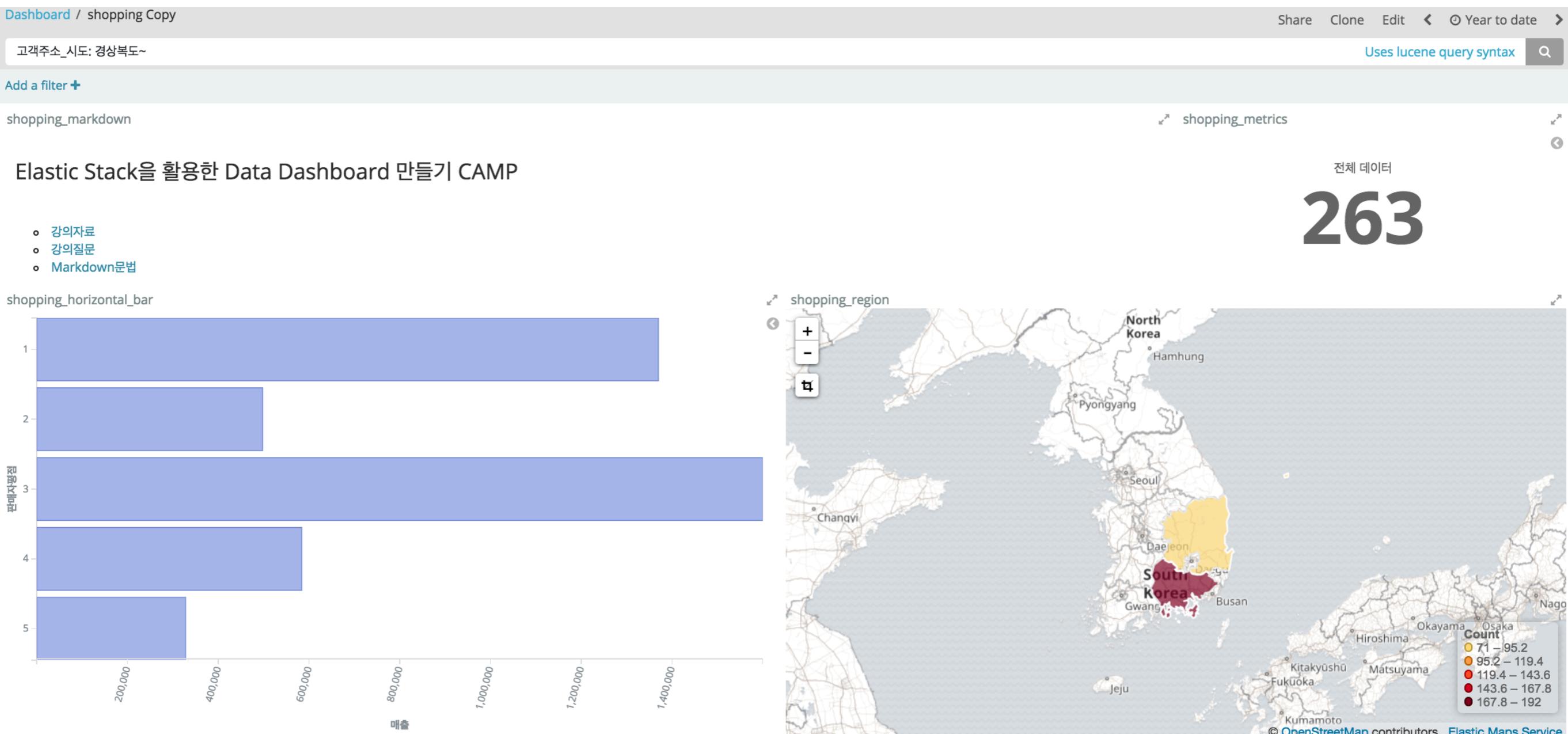
“고객주소\_시도”가 “경상북도”와 유사한 데이터



The screenshot shows a search interface with the following elements:

- Top navigation: Dashboard / shopping Copy
- Action buttons: Share, Clone, Edit, Year to date (with arrows for navigation).
- Search bar: 고객주소\_시도: 경상북도~
- Feedback message: Uses lucene query syntax
- Search icon: A magnifying glass icon.
- Filter button: Add a filter +

# Search Type - Fuzzy



## Search Type - Range

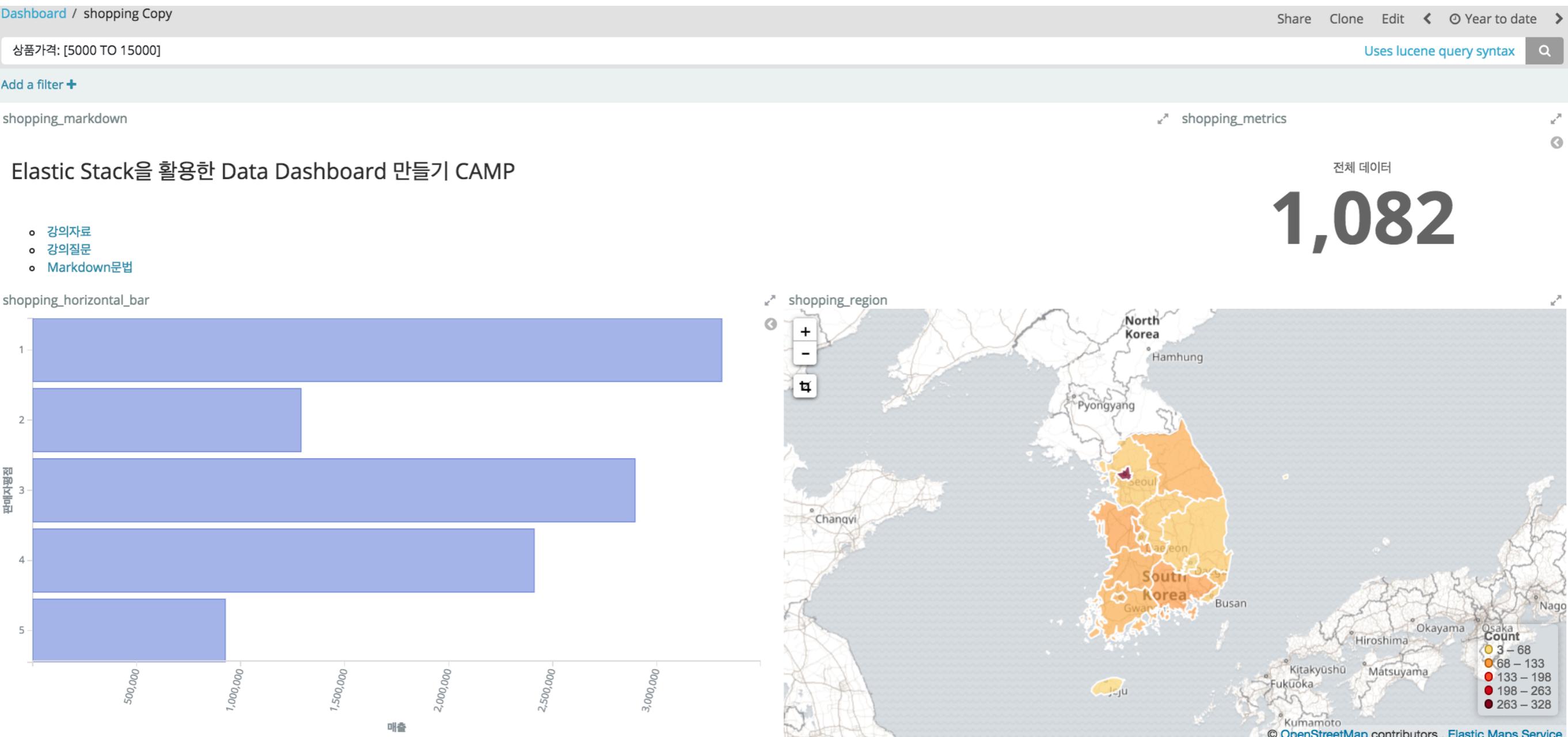
“상품가격”이 5000에서 15000 사이인 데이터



The screenshot shows a search interface with the following elements:

- Top navigation: Dashboard / shopping Copy, Share, Clone, Edit, Year to date controls.
- Search bar: 상품가격: [5000 TO 15000].
- Information text: Uses lucene query syntax.
- Buttons: Add a filter +.

# Search Type - Range



## Search Type - Wildcard

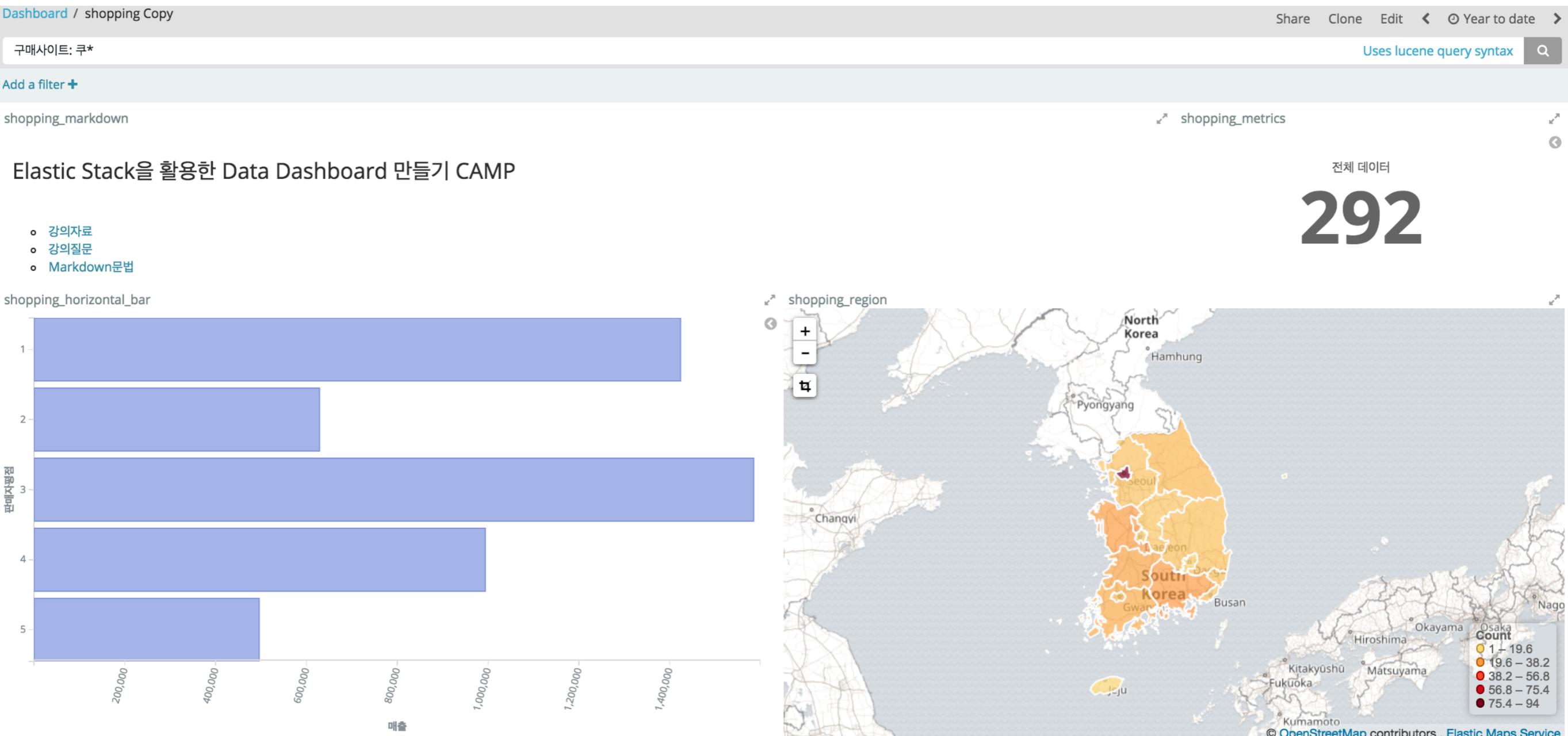
“구매사이트”가 “쿠”로 시작하는 데이터



The screenshot shows a search interface with the following elements:

- Top navigation: Dashboard / shopping Copy, Share, Clone, Edit, <, ⏴ Year to date, >
- Search bar: 구매사이트: 쿠\*
- Feedback message: Uses lucene query syntax
- Search icon: A magnifying glass icon.
- Filter button: Add a filter +

# Search Type - Wildcard



아래와 같은 Query를 Dashboard에서 검색해보자 ↗

Dashboard :{id}\_shopping  
Time Range : Last 6 months

1. “고객성별”이 여성인 Data
2. “결제카드”가 우리 또는 국민인 Data
3. “고객성별”이 남성이면서 “연령대”가 20대
4. “구매사이트”가 쿠팡이거나 “상품개수”가 1~3인 Data
5. “결제카드”가 “우”로 시작하는 모든 Data
6. “구매사이트”가 22번가(오타 아니에요)와 유사한 Data

## Filter와 Search를 비교해보자

		Filter	Search
“고객성별”이 여성인 Data		✓	✓
“결제카드”가 우리 또는 국민인 Data		✓	✓
“고객성별”이 남성이면서 “연령대”가 20대	SCRIPTED FIELD	✓	
“구매사이트”가 쿠팡이거나 “상품개수”가 1~3인 Data	OR 연산		✓
“결제카드”가 “우”로 시작하는 모든 Data	WILDCARD 검색		✓
“구매사이트”가 22번가(오타 아니에요)와 유사한 Data	FUZZY / PROXIMITY 검색		✓

**Discover**

## 데이터를 시각화 하기 전에 데이터를 탐색하는 과정

주요 기능

세부 기능

데이터 검색

(복잡한) 조건을 만족하는 데이터 선별적 탐색 가능

데이터 검색 저장

검색한 결과를 저장하여 Visualize에서 사용

데이터 필터링

(간단한) 특정 조건을 만족하는 데이터 선별적 탐색 가능

데이터 조화

- 특정 Document를 Table/JSON 형태 조화
- Histogram 특정 구간 내의 데이터 조화
- Histogram Bin 간격 설정
- Histogram 데이터를 csv 출력
- 특정 Field의 정보만 조화
- 특정 Field 값을 기준으로 정렬

데이터 통계

- (선택한 Time Range 내의) Documents 개수 확인
- 특정 Field Value의 분포 확인 (주로 Categorical Data, 상위 500개)
- 특정 Field에 non-null Value가 아닌 Documents 수 확인

# Discover에서 Filter를 적용해보자

263 hits

Search... (e.g. status:200 AND extension:PHP) Uses lucene query syntax

**Filter 적용** 경상도 Add filter +

Selected Fields: ? \_source

Available Fields: Popular, t \_type, t 결제카드, # 고객나이, t 고객성별, t 고객주소\_시도, # 배송소요시간, # 상품가격, # 주문시간\_시간대, t 주문시간\_요일, # 주문시간\_요일\_sort, t \_id, t \_index, # \_score, t 결제카드이름, ☐ 고객ip, t 구매사이트, ☐ 물건좌표, t 배송메모, # 상품개수, t 상품분류, t 성별-카드, t 소비행태, ☐ 수령시간

Time: January 1st 2018, 00:00:00.000 - April 10th 2018, 00:54:57.778 — Auto

Count: 주문시간 per day

Time: \_source

- 03월27일 고객주소\_시도: 경상남도 접수번호: 310 주문시간: 03월27일 수령시간: March 29th 2018, 00:56:03.000 예약여부: 일반 배송메모: 상품 이상 고객ip: 89.209.6.10 고객성별: 남성 고객나이: 64 물건좌표: 36.230824009871654, 126.01261457837082 구매사이트: 옥션 판매자평점: 3 상품분류: 가디건 상품가격: 7,000 상품개수: 7 결제카드: 우리 \_id: AWJcYuMGlQ1U\_UGT8VC4 \_type: shopping \_index: shopping \_score: - 주문시간\_시간대: 14 성별-카드: 우리-남성 주문시간\_요일: 화 결제카드이름: 우리카드 주문시간\_요일\_sort: 2 연령대: 50대 이상 소비행태: 과소비 총액: 49,000 배송소요시간: 25
- 03월27일 고객주소\_시도: 경상북도 접수번호: 514 주문시간: 03월27일 수령시간: March 29th 2018, 17:31:22.000 예약여부: 일반 배송메모: 부재중 고객ip: 216.183.176.36 고객성별: 남성 고객나이: 21 물건좌표: 35.039402766378124, 127.70465292384097 구매사이트: 11번가 판매자평점: 1 상품분류: 가디건 상품가격: 6,000 상품개수: 7 결제카드: 국민 \_id: AWJcYuxf1Q1U\_UGT8VGE \_type: shopping \_index: shopping \_score: - 주문시간\_시간대: 6 성별-카드: 국민-남성 주문시간\_요일: 화 결제카드이름: 국민카드 주문시간\_요일\_sort: 2 연령대: 20대 소비행태: 과소비 총액: 42,000 배송소요시간: 50
- 03월27일 고객주소\_시도: 경상북도 접수번호: 4,721 주문시간: 03월27일 수령시간: March 28th 2018, 17:46:45.000 예약여부: 일반 배송메모: 부재중 고객ip: 164.155.95.5 고객성별: 여성 고객나이: 23 물건좌표: 35.23682105495307, 127.02182196746992 구매사이트: 11번가 판매자평점: 1 상품분류: 블라우스 상품가격: 24,000 상품개수: 7 결제카드: 국민 \_id: AWJcY6hqlQ1U\_UGT8Whz \_type: shopping \_index: shopping \_score: - 주문시간\_시간대: 19 성별-카드: 국민-여성 주문시간\_요일: 월 결제카드이름: 국민카드 주문시간\_요일\_sort: 1 연령대: 20대 소비행태: 과소비 총액: 168,000 배송소요시간: 37
- 03월26일 고객주소\_시도: 경상남도 접수번호: 3,681 주문시간: 03월26일 수령시간: March 26th 2018, 21:17:42.000 예약여부: 예약 배송메모: 주소 오류 고객ip: 84.194.148.216 고객성별: 남성 고객나이: 19 물건좌표: 35.839051485380025, 127.52030527202295 구매사이트: g마켓 판매자평점: 3 상품분류: 남방 상품가격: 11,000 상품개수: 1 결제카드: 국민 \_id: AWJcY3oZlQ1U\_UGT8V3j \_type: shopping \_index: shopping \_score: - 주문시간\_시간대: 9 성별-카드: 국민-남성 주문시간\_요일: 월 결제카드이름: 국민카드 주문시간\_요일\_sort: 1 연령대: 10대 소비행태: 저소비 총액: 11,000 배송소요시간: 2
- 03월26일 고객주소\_시도: 경상북도 접수번호: 2,049 주문시간: 03월26일 수령시간: March 27th 2018, 04:38:10.000 예약여부: 예약 배송메모: 관리실에 맡김 고객ip: 10.30.28.47 고객성별: 남성 고객나이: 47 물건좌표: 35.76802911321951, 126.1712980434435 구매사이트: 옥션 판매자평점: 1 상품분류: 남방 상품가격: 9,000 상품개수: 7 결제카드: 우리 \_id: AWJcYzI1lQ1U\_UGT8VeD \_type: shopping \_index: shopping \_score: - 주문시간\_시간대: 7 성별-카드: 우리-남성 주문시간\_요일: 월 결제카드이름: 우리카드 주문시간\_요일\_sort: 1 연령대: 40대 소비행태: 과소비 총액: 63,000 배송소요시간: 12

Filter된 Documents만 조회할 수 있다

# Discover에서 Search를 해보자

292 hits

New Save Open Share ⏪ ⏴ Year to date ⏴

쿠팡

Uses lucene query syntax ⏴

Add a filter +

shopping

Selected Fields

? \_source

Available Fields

Popular

t \_type

t 결제카드

# 고객나이

t 고객성별

t 고객주소\_시도

# 배송소요시간

# 상품가격

# 주문시간\_시간대

t 주문시간\_요일

# 주문시간\_요일\_sort

t \_id

t \_index

# \_score

t 결제카드이름

□ 고객ip

t 구매사이트

● 물건좌표

t 배송메모

# 상품개수

t 상품분류

t 성별-카드

t 소비행태

○ 수령시간

January 1st 2018, 00:00:00.000 - April 10th 2018, 02:32:47.557 — Auto

Count

주문시간 per day

Time ▾ \_source

▶ 03월28일 구매사이트: 쿠팡 접수번호: 1,330 주문시간: 03월28일 수령시간: March 29th 2018, 00:29:24.000 예약여부: 일반 배송메모: 부재증 고객ip: 63.198.167.245 고객성별: 남성 고객나이: 58 물건좌표: 36.385316030586274, 126.662589840303 고객주소\_시도: 경기도 판매자평점: 1 상품분류: 스웨터 상품가격: 13,000 상품개수: 7 결제카드: 신한 \_id: AWJcYxF\_lQ1U\_UGT8VS0 \_type: shopping \_index: shopping \_score: - 주문시간\_시간대: 19 성별-카드: 신한-남성 주문시간\_요일: 화 결제카드이름: 신한카드 주문시간\_요일\_sort: 2 연령대: 50대 이상 소비행태: 과소비 총액: 91,000 배송소요시간: 19

▶ 03월28일 구매사이트: 쿠팡 접수번호: 1,536 주문시간: 03월28일 수령시간: March 31st 2018, 06:33:53.000 예약여부: 일반 배송메모: 부재증 고객ip: 200.193.149.98 고객성별: 남성 고객나이: 56 물건좌표: 36.1961447937324, 126.54471331089651 고객주소\_시도: 강원도 판매자평점: 3 상품분류: 펜초 상품가격: 8,000 상품개수: 7 결제카드: 국민 \_id: AWJcYxq3lQ1U\_UGT8VWC \_type: shopping \_index: shopping \_score: - 주문시간\_시간대: 17 성별-카드: 국민-남성 주문시간\_요일: 화 결제카드이름: 국민카드 주문시간\_요일\_sort: 2 연령대: 50대 이상 소비행태: 과소비 총액: 56,000 배송소요시간: 75

▶ 03월27일 구매사이트: 쿠팡 접수번호: 1,087 주문시간: 03월27일 수령시간: March 30th 2018, 10:12:53.000 예약여부: 일반 배송메모: 부재증 고객ip: 111.216.220.142 고객성별: 여성 고객나이: 31 물건좌표: 36.191517127131036, 126.84033762456899 고객주소\_시도: 전라남도 판매자평점: 3 상품분류: 코트 상품가격: 17,000 상품개수: 1 결제카드: 우리 \_id: AWJcYwbClQ1U\_UGT8VPB \_type: shopping \_index: shopping \_score: - 주문시간\_시간대: 10 성별-카드: 우리-여성 주문시간\_요일: 화 결제카드이름: 우리카드 주문시간\_요일\_sort: 2 연령대: 30대 소비행태: 저소비 총액: 17,000 배송소요시간: 62

▶ 03월27일 구매사이트: 쿠팡 접수번호: 4,299 주문시간: 03월27일 수령시간: March 27th 2018, 14:55:04.000 예약여부: 일반 배송메모: 부재증 고객ip: 167.221.4.198 고객성별: 여성 고객나이: 29 물건좌표: 36.01560629139833, 128.6145777654113 고객주소\_시도: 서울특별시 판매자평점: 3 상품분류: 셔츠 상품가격: 7,000 상품개수: 7 결제카드: 하나 \_id: AWJcY5Wk1Q1U\_UGT8WBN \_type: shopping \_index: shopping \_score: - 주문시간\_시간대: 16 성별-카드: 하나-여성 주문시간\_요일: 월 결제카드이름: 하나카드 주문시간\_요일\_sort: 1 연령대: 20대 소비행태: 과소비 총액: 49,000 배송소요시간: 13

▶ 03월27일 구매사이트: 쿠팡 접수번호: 2,423 주문시간: 03월27일 수령시간: March 29th 2018, 04:03:29.000 예약여부: 예약 배송메모: 주소 오류 고객ip: 0.210.135.130 고객성별: 여성 고객나이: 33 물건좌표: 35.383894825254686, 128.000473655377 고객주소\_시도: 전라남도 판매자평점: 4 상품분류: 니트 상품가격: 15,000 상품개수: 7 결제카드: 우리 \_id: AWJcY0Kt1Q1U\_UGT8Vj5 \_type: shopping \_index: shopping \_score: - 주문시간\_시간대: 16 성별-카드: 우리-여성 주문시간\_요일: 월 결제카드이름: 우리카드 주문시간\_요일\_sort: 1 연령대: 30대 소비행태: 과소비 총액: 105,000 배송소요시간: 50

검색에 match된 Documents만 조회할 수 있다

**이걸로 무얼할 수 있을까?**

**: 특정한 조건을 만족하는 데이터만 선별해서 시각화 할 수 있다.**

# 검색 결과 저장

292 hits

New Save Open Share < ⌂ Year to date >

Save Search

2. 검색 결과 저장

쿠팡검색

3. 검색 결과를 이름을 지정하여 저장

Save

쿠팡

1. Lucene Query 작성 (혹은 필터 적용)

Uses lucene query syntax



Add a filter +

shopping



January 1st 2018, 00:00:00.000 - April 10th 2018, 02:32:47.557 — Auto



Selected Fields

? \_source

Available Fields



Popular

t \_type

t 결제카드

# 고객나이

t 고객성별

t 고객주소\_시도

# 배송소요시간

# 상품가격

# 주문시간\_시간대

t 주문시간\_요일

# 주문시간\_요일\_sort

t \_id

t \_index

# \_score

t 결제카드이름

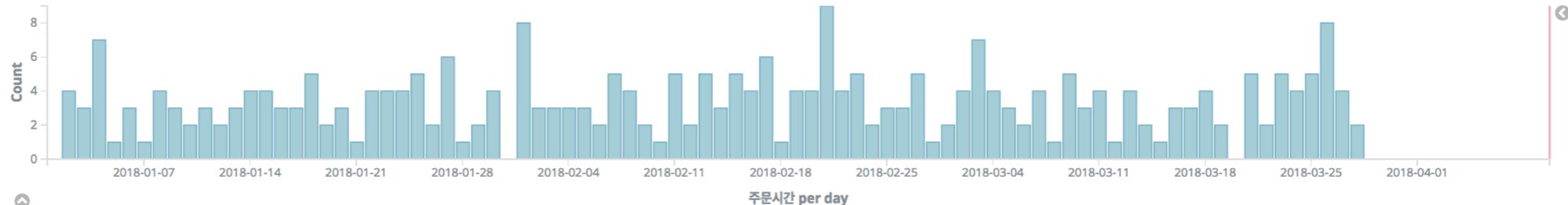
□ 고객ip

t 구매사이트

⌚ 물건좌표

t 배송메모

# 상품개수



Time	_source
03월28일	구매사이트: 쿠팡 접수번호: 1,330 주문시간: 03월28일 수령시간: March 29th 2018, 00:29:24.000 예약여부: 일반 배송메모: 부재중 고객ip: 63.198.167.245 고객성별: 남성 고객나이: 58 물건좌표: 36.385316030586274, 126.662589840303 고객주소_시도: 경기도 판매자평점: 1 상품분류: 스웨터 상품가격: 13,000 상품개수: 7 결제카드: 신한 _id: AWJcYxF_1Q1U_UGT8VS0 _type: shopping _index: shopping _score: - 주문시간_시간대: 19 성별-카드: 신한-남성 주문시간_요일: 화 결제카드이름: 신한카드 주문시간_요일_sort: 2 연령대: 50대 이상 소비행태: 과소비 총액: 91,000 배송소요시간: 19
03월28일	구매사이트: 쿠팡 접수번호: 1,536 주문시간: 03월28일 수령시간: March 31st 2018, 06:33:53.000 예약여부: 일반 배송메모: 부재중 고객ip: 200.193.149.98 고객성별: 남성 고객나이: 56 물건좌표: 36.1961447937324, 126.54471331089651 고객주소_시도: 강원도 판매자평점: 3 상품분류: 팬츠 상품가격: 8,000 상품개수: 7 결제카드: 국민 _id: AWJcYxq3lQ1U_UGT8WVC _type: shopping _index: shopping _score: - 주문시간_시간대: 17 성별-카드: 국민-남성 주문시간_요일: 화 결제카드이름: 국민카드 주문시간_요일_sort: 2 연령대: 50대 이상 소비행태: 과소비 총액: 56,000 배송소요시간: 75
03월27일	구매사이트: 쿠팡 접수번호: 1,087 주문시간: 03월27일 수령시간: March 30th 2018, 10:12:53.000 예약여부: 일반 배송메모: 부재중 고객ip: 111.216.220.142 고객성별: 여성 고객나이: 31 물건좌표: 36.191517127131036, 126.84033762456899 고객주소_시도: 전라남도 판매자평점: 3 상품분류: 코트 상품가격: 17,000 상품개수: 1 결제카드: 우리 _id: AWJcYwbC1Q1U_UGT8VPB _type: shopping _index: shopping _score: - 주문시간_시간대: 10 성별-카드: 우리-여성 주문시간_요일: 화 결제카드이름: 우리카드 주문시간_요일_sort: 2 연령대: 30대 소비행태: 저소비 총액: 17,000 배송소요시간: 62
03월27일	구매사이트: 쿠팡 접수번호: 4,299 주문시간: 03월27일 수령시간: March 27th 2018, 14:55:04.000 예약여부: 일반 배송메모: 부재중 고객ip: 167.221.4.198 고객성별: 여성 고객나이: 29 물건좌표: 36.01560629139833, 128.6145777654113 고객주소_시도: 서울특별시 판매자평점: 3 상품분류: 셔츠 상품가격: 7,000 상품개수: 7 결제카드: 하나 _id: AWJcY5Wk1Q1U_UGT8WBN _type: shopping _index: shopping _score: - 주문시간_시간대: 16 성별-카드: 하나-여성 주문시간_요일: 월 결제카드이름: 하나카드 주문시간_요일_sort: 1 연령대: 20대 소비행태: 과소비 총액: 49,000 배송소요시간: 13

## 저장된 검색 결과를 이용한 시각화 1단계

Visualize / New

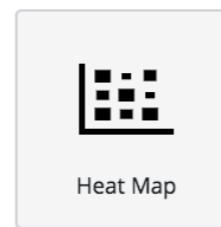
### Select visualization type

Search visualization types...

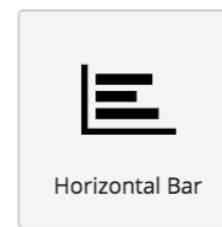
#### Basic Charts



Area



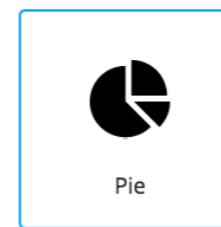
Heat Map



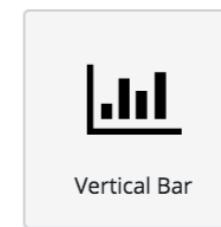
Horizontal Bar



Line



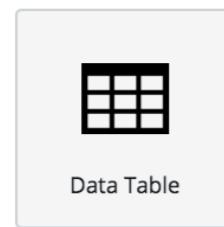
Pie



Vertical Bar

Compare parts of a whole

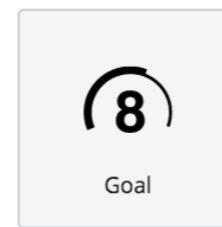
#### Data



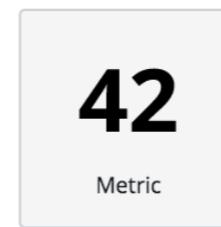
Data Table



Gauge

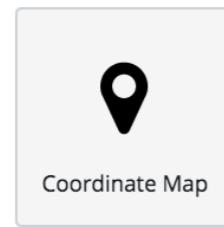


Goal

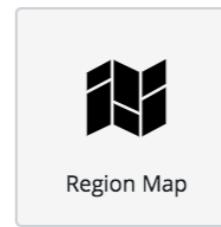


Metric

#### Maps

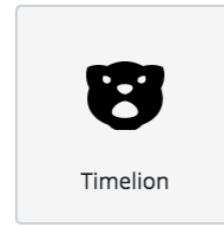


Coordinate Map



Region Map

#### Time Series



Timelion



Visual Builder

## 저장된 검색 결과를 이용한 시각화 2단계

Visualize / New / Choose search source

### From a New Search, Select Index

Filter... 67 of 67

Name ▲
nia.jeong_2018.03.01
acroama_2018.03.01
brilliantjay_2018.03.01
ainsally6575_2018.03.01
sbhong513_2018.03.01
jungsungwook_2018.03.01
jaycelynnkim1123_2018.03.01
brilliantjay_2018.03.02
selee_2018.03.01
young3771_2018.03.02
sm1835dh_2018.03.02
inchang.oh_2018.03.01
sofiejhlee_2018.03.01
jaycelynnkim1123_2018.03.02
nia.jeong_2018.03.02
robert_2018.03.02
young3771_2018*
sbhong513_2018*
jungsungwook_2018.03.02
colossus_2018.03.01

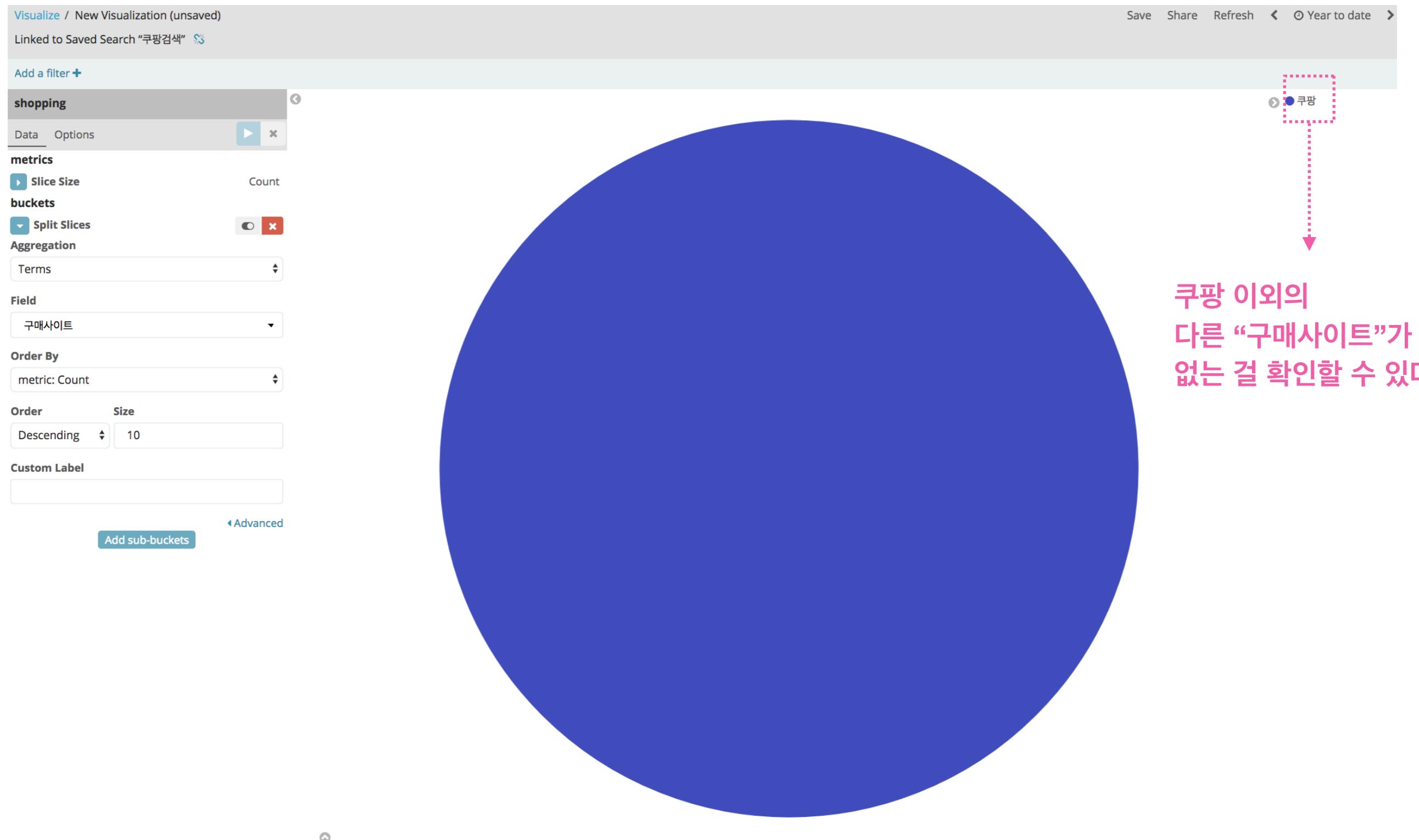
1 2 3 4 »

### Or, From a Saved Search

Saved Searches Filter... 1-3 of 3 Manage saved searches

Name ▲
쿠팡검색 “저장한 검색결과” 선택

### 저장된 검색 결과를 이용한 시각화 3단계



쿠팡 이외의  
다른 “구매사이트”가  
없는 걸 확인할 수 있다.

**질문 및 Feedback은**

**gshock94@gmail.com로 주세요**