Elastic Stack을 활용한 Data Dashboard 만들기

Week 6 - Dashboard 만들기 최종실습

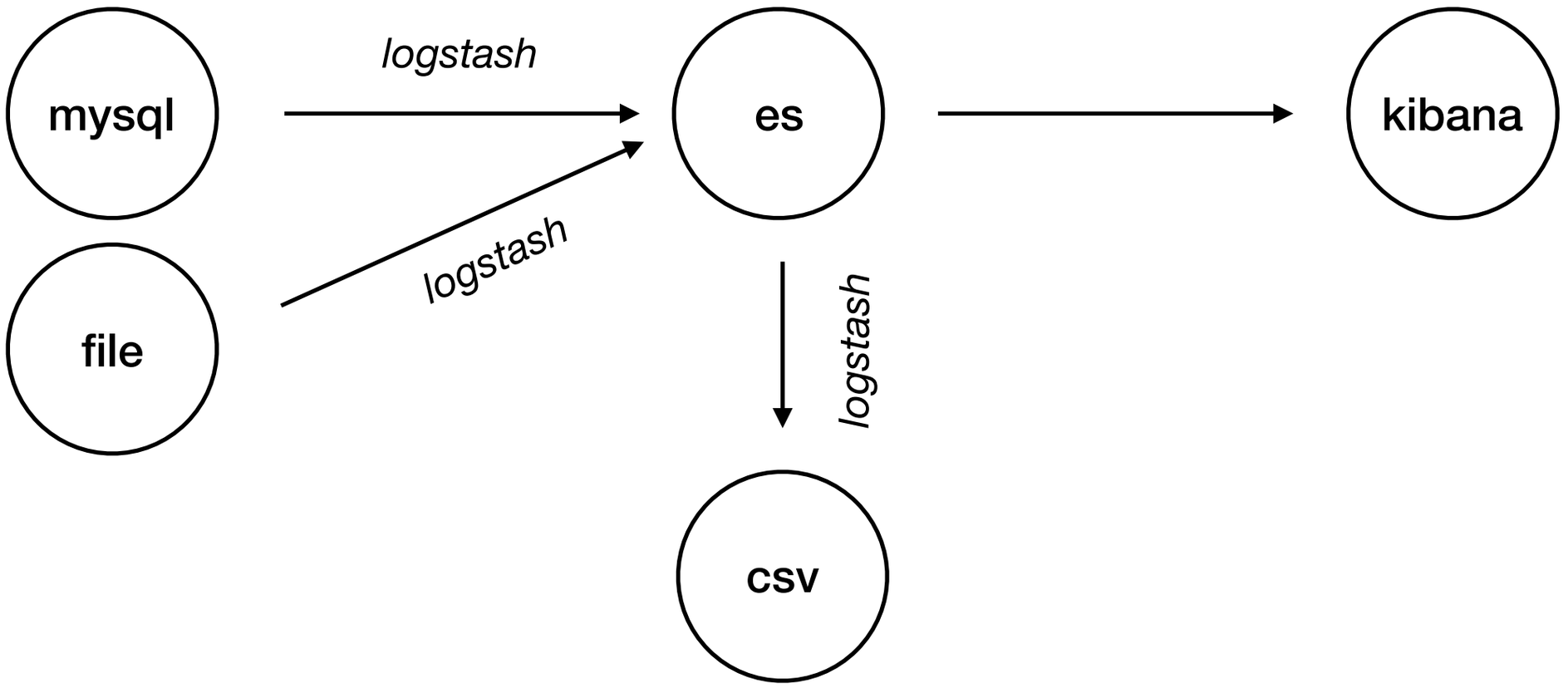
Fast Campus

Guide

- 실제로 대시보드 구축 프로젝트를 가정하고 실습 진행
- 자세한 안내보다는 **요구사항 위주의 문서**
- 최소한의 정보는 제공하지만 **직접 데이터를 조회하면서 작업**해야 함
- 지금까지 배웠던 내용을 사용하기를 권장
- 특히 **시각화의 경우 정해진 답이 없으므로**, 사용자의 니즈를 충족하는 범위 내에서 최대한 **자유롭게** 제작

개요

- mysql에서 elasticsearch로 분석하려는 데이터 전송
- kibana를 통해 대시보드 구축
- Filter, Query DSL, Lucene Query Syntax 등을 이용해 원하는 정보 검색
- elasticsearch 데이터 csv 추출



1. AWS EC2 Instance에 접속하자 👑

2. Elastic Stack (5.6.4) 을 설치하자 👑

3. Elasticsearch와 Kibana를 실행하자 👑

4. exercise라는 Index를 생성하고 아래와 같이 mapping을 설정하자 👑

Field	Data Type
customer_age	integer
customer_card	keyword
customer_location	keyword
customer_sex	keyword
date_delivery	date
date_order	date
product_gps	geo_point
product_item	keyword
product_price	integer
product_quantity	integer
seller_rating	integer
seller_site	keyword

5. Logstash를 이용해서 mysql 데이터를 elasticsearch로 전송하자 👑

customer_age	customer_sex	customer_location	customer_card	product_item	product_gps	product_price	product_quantity	date_order	date_delivery	seller_rating	seller_site
52	여성	광주광역시	삼성	셔츠	36.288199573715545, 128.13057513572724	25000	3	2018-01-12 15:02:34	2018-01-16 00:41:34	3	옥션
29	남성	광주광역시	삼성	티셔츠	36.0869495071031, 127.10481983759166	14000	4	2018-01-09 11:57:47	2018-01-11 16:46:47	5	GS샵
36	여성	전라남도	신한	자켓	35.90825189035691, 128.0671312045079	20000	2	2018-01-02 09:24:16	2018-01-05 01:20:16	5	GS샵
38	여성	충청북도	삼성	코트	35.68831118876786, 128.133195592593	25000	1	2018-02-04 14:35:27	2018-02-07 00:57:27	3	티몬
49	남성	전라남도	신한	셔츠	36.513374488580546, 127.30635747850515	23000	3	2018-01-11 22:14:48	2018-01-14 11:31:48	3	티몬

- mysql 정보
  - host : 13.125.153.139:3306
  - database : week5
  - user/password : week5/week5
  - table : week5_test
- logstash 조건
  - @timestamp와 @version field는 삭제할 것

6. Logstash를 이용해서 file 데이터를 elasticsearch로 전송하자 👑

```
customer_card,seller_site,seller_rating,customer_sex,product_price,product_quantity,customer_age,customer_location,date_order,date_delivery,product_item,product_gps_lat,product_gps_lon
시티,옥션,3,남성,29000,4,46,경기도,2018-01-09T03:08:32,2018-01-12T19:23:32,가디건,37.5656494,126.868678
국민,옥션,1,남성,0,2,26,경기도,2018-01-19T03:08:32,2018-01-23T19:23:32,가디건,37.5556494,126.898678
시티,쿠팡,2,여성,10,4,46,서울특별시,2017-01-10T03:08:32,2017-11-12T19:23:32,가디건,37.50256494,127.038678
하나,11번가,5,남성,29000,4,46,경기도,2018-01-09T03:08:32,2018-01-12T19:23:32,셔츠,37.5656494,126.868678
하나,11번가,1,남성,0,2,26,경기도,2018-01-19T03:08:32,2018-01-23T19:23:32,니트,37.5556494,126.898678
우리,쿠팡,2,여성,1000,4,46,서울특별시,2017-01-10T03:08:32,2017-11-12T19:23:32,청바지,37.50256494,127.038678
시티,위메프,3,남성,3000,4,46,경기도,2018-01-09T03:08:32,2018-01-12T19:23:32,가디건,37.5656494,126.868678
국민,위메프,3,남성,5000,2,26,경기도,2018-01-19T03:08:32,2018-01-23T19:23:32,셔츠,37.5556494,126.898678
우리,쿠팡,3,여성,10000,4,46,서울특별시,2017-01-10T03:08:32,2017-11-12T19:23:32,셔츠,37.50256494,127.038678
```

- file path : /usr/share/logstash/data/test.csv
- filter 조건
  - **csv** filter
    1) 데이터를 “,”로 구분해서 field를 생성하자
    2) field 이름은 test.csv 파일 최상단의 데이터를 이용하자
    3) customer_age, product_price, product_quantity, seller_rating field는 integer로 convert 하자
  - **drop** filter
    4) product_price 값이 0인 event는 drop하자
  - **mutate** filter
    5) product_gps_lat field와 product_gps_lon field를 합쳐서 product_gps field를 생성하자
    6) @timestamp, @version, host, path, message, product_gps_lat, product_gps_lon field는 삭제하자
  - **date** filter
    7) date_order field와 date_delivery field는 local time이므로 이 점이 반영될 수 있도록 작성하자

7. Kibana에서 Index Patterns를 등록하자 👑

Time filter field : date_order

8. Scripted Field를 생성하자 👑

Field	Value	type	format	detail
dayofweek	“평일” 또는 “주말”	string	string	*date_order*를 기준으로 “평일” 또는 “주말” 중 하나의 값을 갖도록 설정
hourofday	“새벽” 또는 “아침” 또는 “점심” 또는 “오후” 또는 “저녁” 또는 “밤”	string	string	*date_order*의 시간대를 기준으로 다음과 같은 기준으로 설정 - 0 ~ 5 : “새벽” - 5 ~ 10 : “아침” - 10 ~ 13 : “점심” - 13 ~ 17 : “오후” - 17 ~ 20 : “저녁" - 20 ~ 23 : “밤”
hourofday_sort	*0, 1, 2, 3, 4, 5*	integer	integer	*date_order*의 시간대를 기준으로 다음과 같은 기준으로 설정 - 0 ~ 5 : 0 - 5 ~ 10 : 1 - 10 ~ 13 : 2 - 13 ~ 17 : 3 - 17 ~ 20 : 4 - 20 ~ 23 : 5

9. Discover에서 다음과 같은 질문에 답해보자 👑

Time Range : Year-to-Date

1) 전체 건 수는?

2) 주말 동안 신한 카드 또는 국민 카드 결제 건수는?

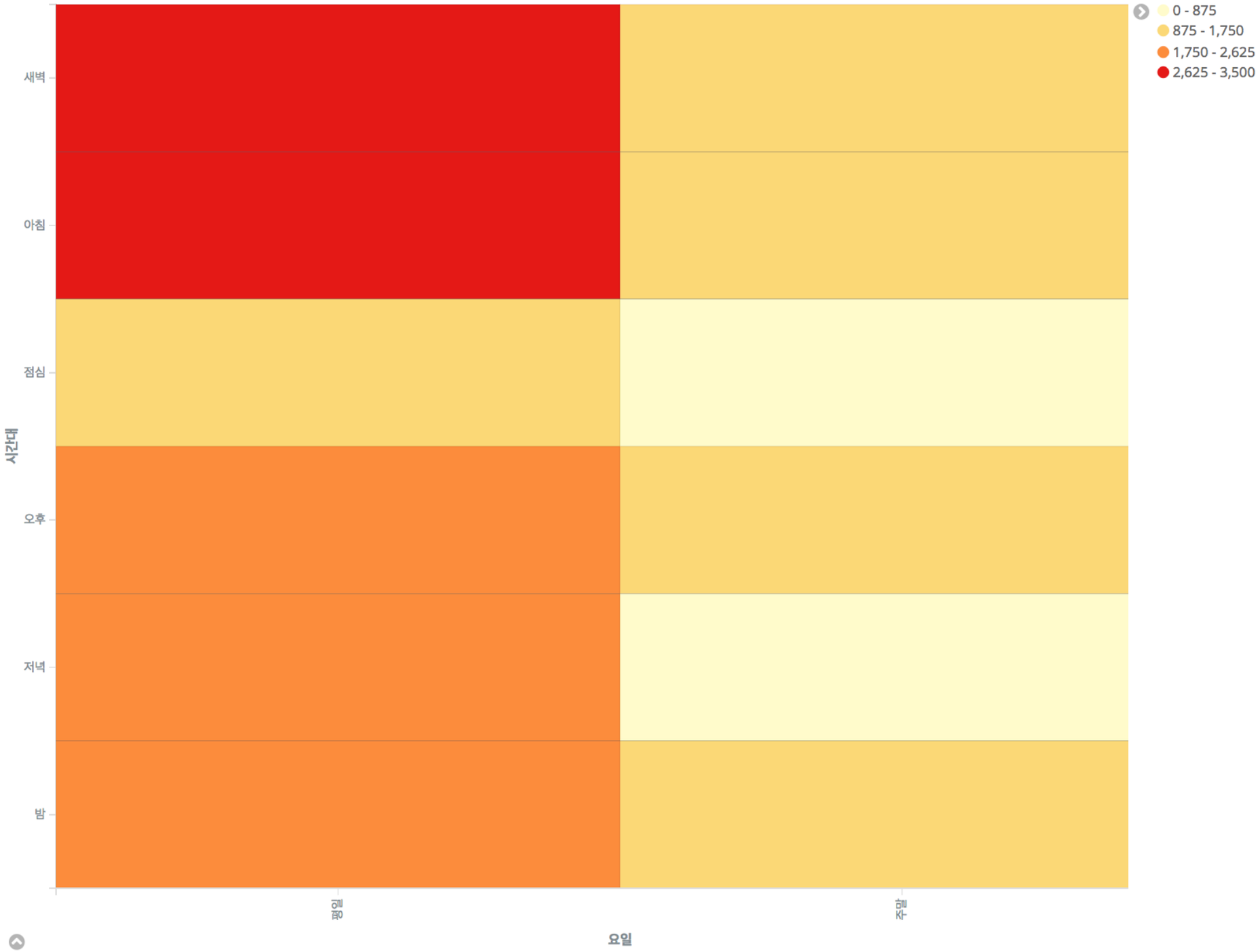
3) 주별로 봤을 때 건 수가 가장 많았던 주는 언제인가? 그 주의 건 수는?

4) date_order를 ascending으로 Documents를 정렬했을 때, 가장 인기 있었던 product_item과 그 비율은?

10. Heatmap Visualization을 만들자

Time Range : Year-to-Date

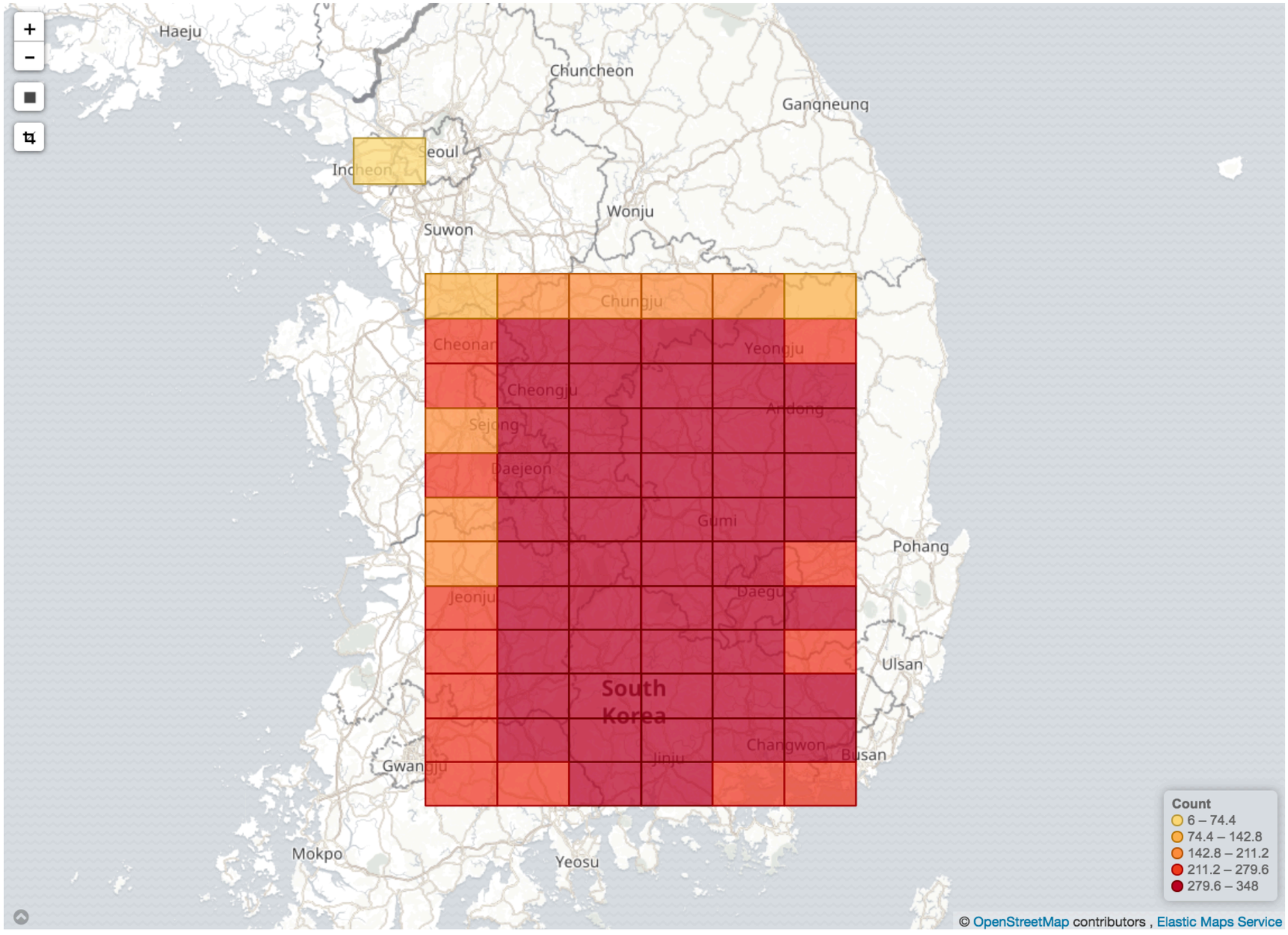
- 목적 : date_order를 기준으로 요일별 시간대별 건 수 시각화해서 요일/시간에 따른 판매 현황 파악
- 데이터 : 요일별, 시간대별 정보는 11페이지에서 생성한 Scripted Field 이용


0 - 875
875 - 1,750
1,750 - 2,625
2,625 - 3,500
새벽
아침
점심
오후
저녁
밤
시간
평일
주말
요일

11. Coordinate Map Visualization을 만들자

Time Range : Year-to-Date

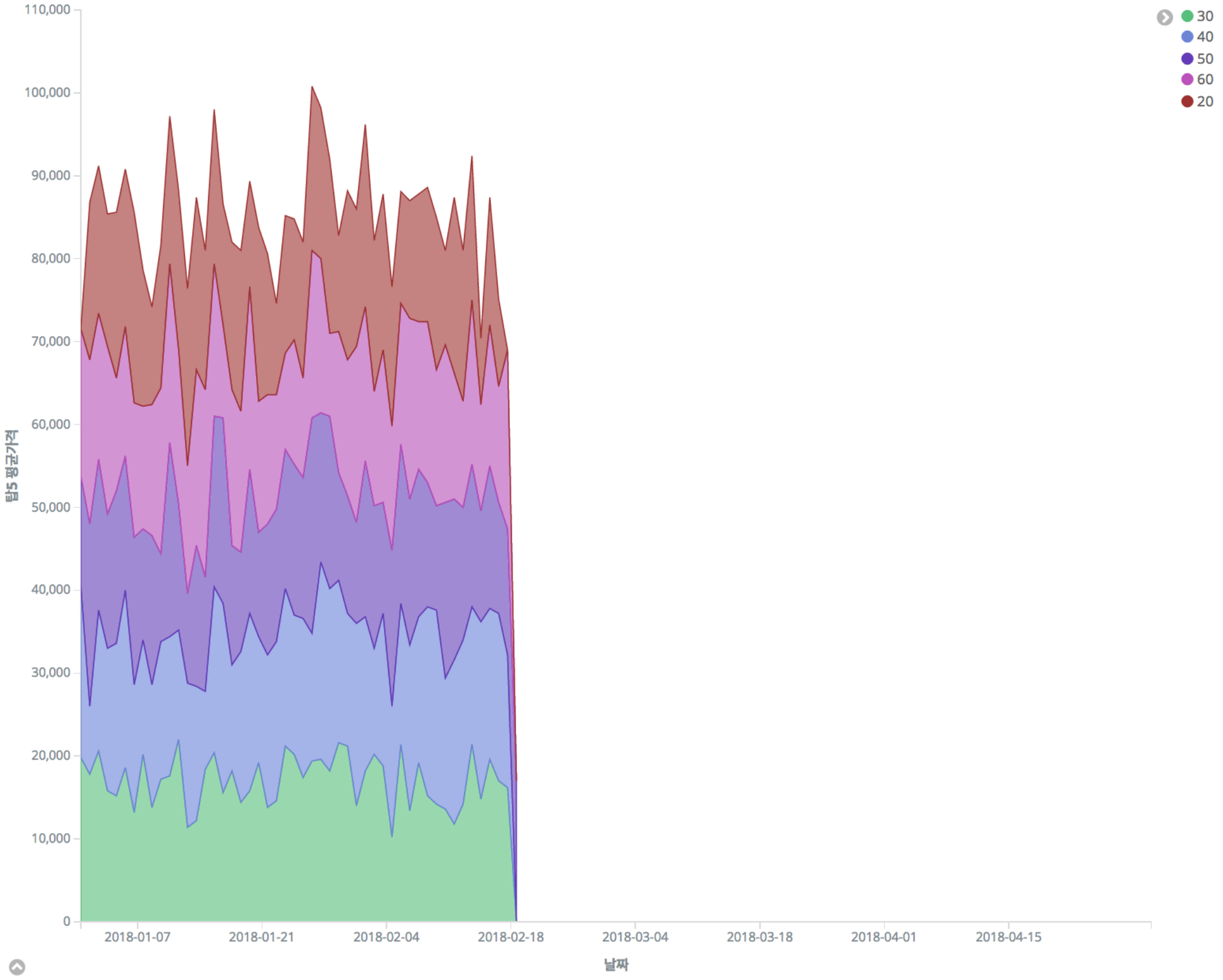
- 목적 : 지역별 판매 데이터 현황을 시각적으로 파악
- 데이터 : product_gps field로 gps point를 생성해서 documents count aggregation 이용


Haeju
Chuncheon
Gangneung
Seoul
Incheon
Wonju
Suwon
Chungju
Cheonan
Yeongju
Cheongju
Andong
Sejong
Daejeon
Gumi
Pohang
Jeonju
Daegu
Ulsan
South Korea
Changwon
Busan
Jinju
Gwangju
Mokpo
Yeosu
Count
6 – 74.4
74.4 – 142.8
142.8 – 211.2
211.2 – 279.6
279.6 – 348
© OpenStreetMap contributors , Elastic Maps Service

12. Area Chart Visualization을 만들자

Time Range : Year-to-Date

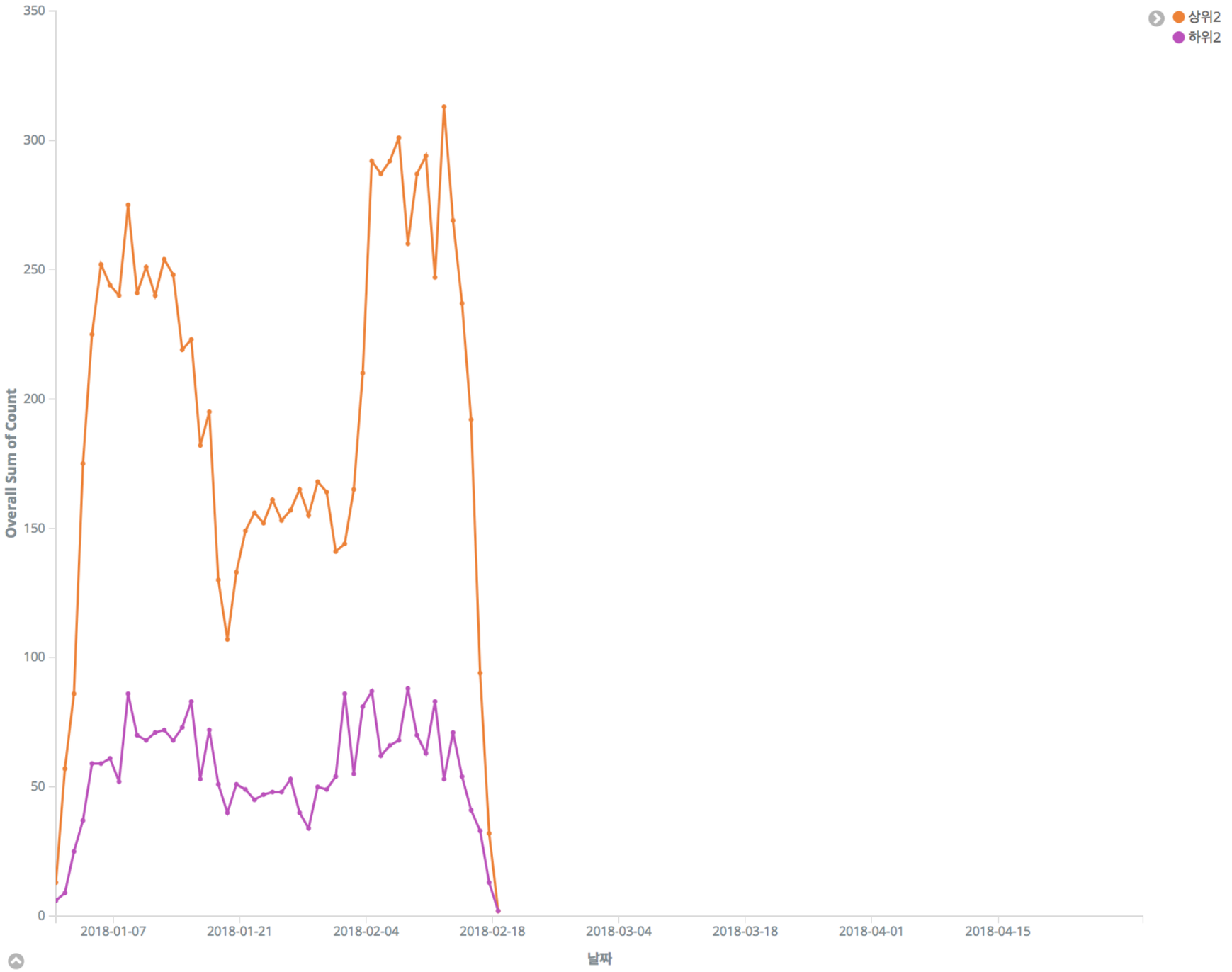
- 목적 : 일 별로 각 연령대가 구매한 상품중 seller_rating이 가장 높은 5개 상품의 product_price의 평균을 각각 추적
- 데이터
  - 일 별 : date_order를 기준으로 daily로 구분
  - 연령대 : customer_age를 10살 단위로 구분


30
40
50
60
20
110,000
100,000
90,000
80,000
70,000
60,000
50,000
40,000
30,000
20,000
10,000
0
탑5 평균가격
2018-01-07
2018-01-21
2018-02-04
2018-02-18
2018-03-04
2018-03-18
2018-04-01
2018-04-15
날짜

13. Line Chart Visualization을 만들자

Time Range : Year-to-Date

- 목적 : 일 별로 "product_price" field의 합이 큰 상위 2개 그룹과 하위 2개 그룹의 판매 개수 (documents)를 비교하자
- 데이터
  - 일 별 : date_order를 기준으로 daily로 구분
  - 판매개수 : documents의 개수


상위2
하위2
350
300
250
200
150
100
50
0
Overall Sum of Count
2018-01-07
2018-01-21
2018-02-04
2018-02-18
2018-03-04
2018-03-18
2018-04-01
2018-04-15
날짜

14. 앞서 만든 Visualizations을 적절히 배치해서 Dashboard를 만들자 👑

15. Query DSL로 Dashboard에서 다음 조건에 맞는 데이터만 보여주자 👑

1) 25세 이상 경상도 사람 중에서 "신한"으로 결제했거나 “가디건, 셔츠, 스웨터”를 구매한 사람들의 데이터

2) 서울 출신 남성이거나, 35세 이하 부산시민이거나, 판매자평점이 2~4 사이인 충청도민인 사람들의 데이터

16. Logstash를 이용해서 elasticsearch 데이터를 csv로 추출하자 👑

1) 다음 모든 조건을 만족하는 Documents만 출력
   - product_item : 셔츠 또는 니트
   - 2 <= seller_rating <= 4
   - customer_location : "경상"으로 시작

2) 출력 Field
   - product_item
   - product_price
   - seller_rating
   - seller_site
   - customer_location

질문 및 Feedback은

gshock94@gmail.com로 주세요