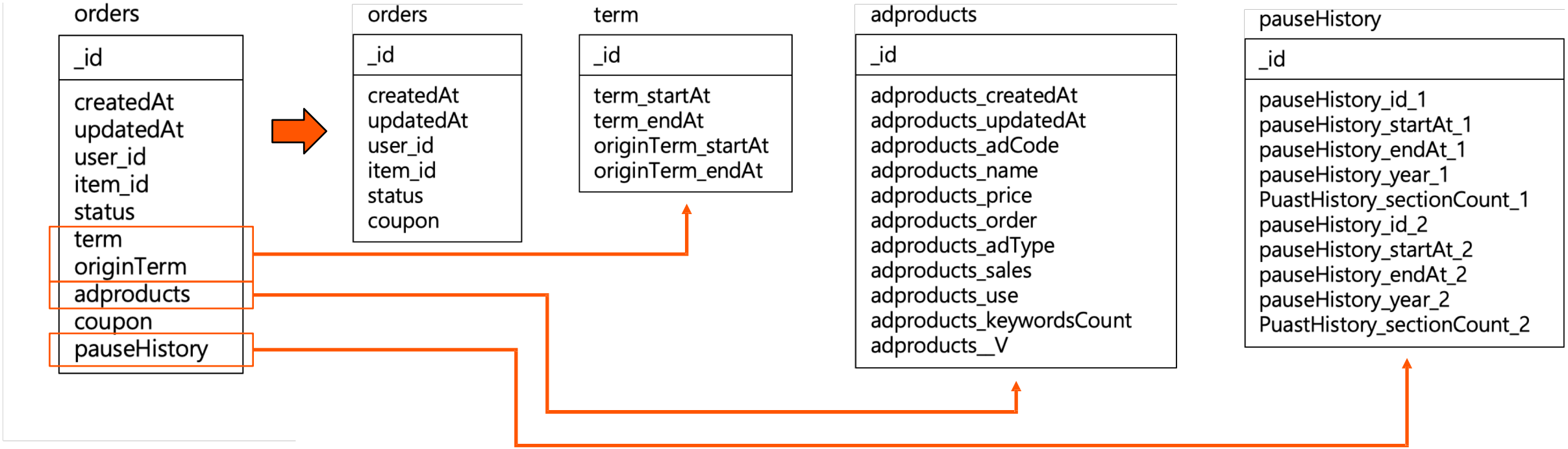
개인 프로젝트

-고객 로그 데이터를 통해 유저의 상품별 주문 리텐션 분석

1. Data Table Column 정리 및 분리 : Python 활용



Githus Gist: https://gist.github.com/jaehwan2/dfe44fbbe0ea917c3fb7bafa93eecc4e

Data Source: http://shorturl.at/qvRW0

지원자 **송재환**

개인 프로젝트

-고객 로그 데이터를 통해 유저의 상품별 주문 리텐션 분석

1. Data Table Column 정리 및 분리 : Python 활용

term 테이블 만들기

term
_id
term_startAt term_endAt originTerm_startAt originTerm_endAt

```
In [1]:
import pandas as pd
import re
import json

orders = pd.read_excel("/Users/jaehwan/Downloads/orders.xlsx", encoding='utf-8')
```

```
In [2]:
def columnSplit (columnName, column1, column2):

    new_id = []
    column1 = []
    column2 = []

    new_columns = [new_id, column1, column2]

    for i in orders._id.index:

        column_temp = columnName[i].replace(": ",": '").replace(")","'")
        column_temp = column_temp.replace("datetime.datetime(","")
        column_temp = column_temp.replace("ObjectId('","").replace("'","\"")

        column_dict = json.loads(column_temp)

        column_dict.keys()
        column_dict.values()

        key_list = list(column_dict.keys())

        for j in key_list:

            column_dict[j] = column_dict[j].split(", ")

            column_dict[j][1] = column_dict[j][1].zfill(2)
            column_dict[j][2] = column_dict[j][2].zfill(2)

            column_dict[j] = ','.join(column_dict[j]).replace(",","",2).split(",")[0]

        new_id.append(orders._id[i])
        column1.append(column_dict["startAt"])
        column2.append(column_dict["endAt"])

    df=pd.DataFrame(new_columns).transpose()

    return df
```

```
In [3]:
termSplit = columnSplit (orders.term, "term_startAt", "term_endAt")
termSplit.columns = ["_id", "term_startAt", "term_endAt"]

originTermSplit = columnSplit (orders.originTerm, "originTerm_startAt", "originTerm_endAt")
originTermSplit.columns = ["_id", "originTerm_startAt", "originTerm_endAt"]

result_join = pd.merge(left=termSplit, right=originTermSplit, how='left', on="_id")
result_join.to_excel("/Users/jaehwan/Downloads/orders_term_originTerm.xlsx")
```

Githus Gist: https://gist.github.com/jaehwan2/dfe44fbbe0ea917c3fb7bafa93eecc4e

Data Source: http://shorturl.at/qvRW0

지원자 **송재환**

개인 프로젝트

-고객 로그 데이터를 통해 유저의 상품별 주문 리텐션 분석

1. Data Table Column 정리 및 분리 : Python 활용

adproducts 테이블 만들기-1

adproducts

_id
adproducts_createdAt
adproducts_updatedAt
adproducts_adCode
adproducts_name
adproducts_price
adproducts_order
adproducts_adType
adproducts_sales
adproducts_use
adproducts_keywordsCount
adproducts__V

```
In [1]: import pandas as pd
        import re
        import json

        orders=pd.read_excel("/Users/jaehwan/Downloads/orders.xlsx",encoding='utf-8')
```

```
In [5]: new_id = []
        adproduct_id = []
        adproduct_createdAt = []
        adproduct_updatedAt = []
        adproduct_adCode = []
        adproduct_name = []
        adproduct_price = []
        adproduct_order = []
        adproduct_adType = []
        adproduct_sales = []
        adproduct_use = []
        adproduct_keywordsCount = []
        adproduct___v = []

        new_columns = [new_id,
                       adproduct_id,
                       adproduct_createdAt,
                       adproduct_updatedAt,
                       adproduct_adCode,
                       adproduct_name,
                       adproduct_price,
                       adproduct_order,
                       adproduct_adType,
                       adproduct_sales,
                       adproduct_use,
                       adproduct_keywordsCount,
                       adproduct___v]

        for i in orders._id.index:

            column_temp = orders.adproducts[i].replace("ObjectId(","").replace(")","'").replace(".0","")
            column_temp = column_temp.replace("datetime.datetime(","'").replace("True","'True'").replace("False","'False'")
            column_temp = column_temp.replace("'price': ","'price': '").replace("00, ","00', ")
            column_temp = column_temp.replace(": 0,",": '0',").replace(": 1,",": '1',").replace("0}","'0'}").replace("''","'")
            column_temp = column_temp.replace("'","\"").replace("[","").replace("]","")

            column_dict = json.loads(column_temp)

            column_dict.keys()
```

Githus Gist: https://gist.github.com/jaehwan2/dfe44fbbe0ea917c3fb7bafa93eecc4e

Data Source: http://shorturl.at/qvRW0

지원자 **송재환**

개인 프로젝트

-고객 로그 데이터를 통해 유저의 상품별 주문 리텐션 분석

1. Data Table Column 정리 및 분리 : Python 활용

adproducts 테이블 만들기-2

adproducts

_id
adproducts_createdAt
adproducts_updatedAt
adproducts_adCode
adproducts_name
adproducts_price
adproducts_order
adproducts_adType
adproducts_sales
adproducts_use
adproducts_keywordsCount
adproducts__V

```
column_dict = json.loads(column_temp)

column_dict.keys()
column_dict.values()

key_list = list(column_dict)

new_id.append(orders._id[i])
adproduct_id.append(None)
adproduct_createdAt.append(None)
adproduct_updatedAt.append(None)
adproduct_adCode.append(None)
adproduct_name.append(None)
adproduct_price.append(None)
adproduct_order.append(None)
adproduct_adType.append(None)
adproduct_sales.append(None)
adproduct_use.append(None)
adproduct_keywordsCount.append(None)
adproduct___v.append(None)

for j in key_list:

    if j == '_id' :
        column_dict[j] = "".join(column_dict[j]).replace("[","").replace("]","")
        adproduct_id[i] = column_dict[j]

    elif j == 'createdAt':
        column_dict[j] = column_dict[j].split(", ")
        column_dict[j][1] = column_dict[j][1].zfill(2)
        column_dict[j][2] = column_dict[j][2].zfill(2)
        column_dict[j] = ','.join(column_dict[j]).replace(",","",2).split(",")[0]
        adproduct_createdAt[i] = column_dict[j]

    elif j =='updatedAt':
        column_dict[j] = column_dict[j].split(", ")
        column_dict[j][1] = column_dict[j][1].zfill(2)
        column_dict[j][2] = column_dict[j][2].zfill(2)
        column_dict[j] = ','.join(column_dict[j]).replace(",","",2).split(",")[0]
        adproduct_updatedAt[i] = column_dict[j]

    elif j == 'adCode':
        adproduct_adCode[i] = column_dict[j]

    elif j == 'name':
        adproduct_name[i] = column_dict[j]
```

Githus Gist: https://gist.github.com/jaehwan2/dfe44fbbe0ea917c3fb7bafa93eecc4e

Data Source: http://shorturl.at/qvRW0

지원자 **송재환**

개인 프로젝트

-고객 로그 데이터를 통해 유저의 상품별 주문 리텐션 분석

1. Data Table Column 정리 및 분리 : Python 활용

adproducts 테이블 만들기-3

adproducts

_id
adproducts_createdAt
adproducts_updatedAt
adproducts_adCode
adproducts_name
adproducts_price
adproducts_order
adproducts_adType
adproducts_sales
adproducts_use
adproducts_keywordsCount
adproducts__V

```
            adproduct_updatedAt[i] = column_dict[j]

        elif j == 'adCode':
            adproduct_adCode[i] = column_dict[j]

        elif j == 'name':
            adproduct_name[i] = column_dict[j]

        elif j == 'price':
            adproduct_price[i] = column_dict[j]

        elif j == 'order':
            adproduct_order[i] = column_dict[j]

        elif j == 'adType':
            adproduct_adType[i] = column_dict[j]

        elif j == 'sales':
            adproduct_sales[i] = column_dict[j]

        elif j == 'use':
            adproduct_use[i] = column_dict[j]

        elif j == 'keywordsCount':
            adproduct_keywordsCount[i] = column_dict[j]

        elif j == '__v':
            adproduct___v[i] = column_dict[j]

df = pd.DataFrame(new_columns).transpose()
df.columns=["_id",
            "adproduct_id",
            "adproduct_createdAt",
            "adproduct_updatedAt",
            "adproduct_adCode",
            "adproduct_name",
            "adproduct_price",
            "adproduct_order",
            "adproduct_adType",
            "adproduct_sales",
            "adproduct_use",
            "adproduct_keywordsCount",
            "adproduct___v"]

df.to_excel("/Users/jaehwan/Downloads/orders_adproducts.xlsx")
```

Githus Gist: https://gist.github.com/jaehwan2/dfe44fbbe0ea917c3fb7bafa93eecc4e

Data Source: http://shorturl.at/qvRW0

지원자 **송재환**

개인 프로젝트

-고객 로그 데이터를 통해 유저의 상품별 주문 리텐션 분석

1. Data Table Column 정리 및 분리 : Python 활용

pauseHistory 테이블 만들기-1

pauseHistory
_id
pauseHistory_id_1 pauseHistory_startAt_1 pauseHistory_endAt_1 pauseHistory_year_1 PuastHistory_sectionCount_1 pauseHistory_id_2 pauseHistory_startAt_2 pauseHistory_endAt_2 pauseHistory_year_2 PuastHistory_sectionCount_2

In [1]:
```
import pandas as pd
import re
import json

orders=pd.read_excel("/Users/jaehwan/Downloads/orders.xlsx",encoding='utf-8')
```

In [2]:
```
j=0

new_id = []
pauseHistory_id_1 = []
pauseHistory_startAt_1 = []
pauseHistory_endAt_1 = []
pauseHistory_year_1 = []
pauseHistory_sectionCount_1 = []

new_columns1 = [new_id,
                pauseHistory_id_1,
                pauseHistory_startAt_1,
                pauseHistory_endAt_1,
                pauseHistory_year_1,
                pauseHistory_sectionCount_1]

for i in orders._id.index:

    new_id.append(orders._id[i])
    pauseHistory_id_1.append(None)
    pauseHistory_startAt_1.append(None)
    pauseHistory_endAt_1.append(None)
    pauseHistory_year_1.append(None)
    pauseHistory_sectionCount_1.append(None)

    if len(str(orders.pauseHistory[i])) > 10:

        column_temp = orders.pauseHistory[i].split(', {')[0].replace("ObjectId(","").replace(")","'")
        column_temp = column_temp.replace("datetime.datetime(","'").replace("}","'}").replace("'year': 2020","'year': '
        column_temp = column_temp.replace("'sectionCount': ","'sectionCount': '").replace("0}","'0'}").replace("''","'"
        column_temp = column_temp.replace("'","\"").replace("[","").replace("]","")

        column_dict = json.loads(column_temp)

        column_dict.keys()
        column_dict.values()
```

Githus Gist: https://gist.github.com/jaehwan2/dfe44fbbe0ea917c3fb7bafa93eecc4e

Data Source: http://shorturl.at/qvRW0

지원자 **송재환**

개인 프로젝트

-고객 로그 데이터를 통해 유저의 상품별 주문 리텐션 분석

1. Data Table Column 정리 및 분리 : Python 활용

pauseHistory 테이블 만들기-2

pauseHistory
_id
pauseHistory_id_1
pauseHistory_startAt_1
pauseHistory_endAt_1
pauseHistory_year_1
PuastHistory_sectionCount_1
pauseHistory_id_2
pauseHistory_startAt_2
pauseHistory_endAt_2
pauseHistory_year_2
PuastHistory_sectionCount_2

```
        key_list = list(column_dict)

        for j in key_list:

            if j == '_id' :
                column_dict[j] = "".join(column_dict[j]).replace("[","").replace("]","")
                pauseHistory_id_1[i] = column_dict[j]

            elif j == 'startAt':
                column_dict[j] = column_dict[j].split(", ")
                column_dict[j][1] = column_dict[j][1].zfill(2)
                column_dict[j][2] = column_dict[j][2].zfill(2)
                column_dict[j] = ','.join(column_dict[j]).replace(",","",2).split(",")[0]
                pauseHistory_startAt_1[i] = column_dict[j]

            elif j =='endAt':
                column_dict[j] = column_dict[j].split(", ")
                column_dict[j][1] = column_dict[j][1].zfill(2)
                column_dict[j][2] = column_dict[j][2].zfill(2)
                column_dict[j] = ','.join(column_dict[j]).replace(",","",2).split(",")[0]
                pauseHistory_endAt_1[i] = column_dict[j]

            elif j =='year':
                pauseHistory_year_1[i] = column_dict[j]

            elif j =='sectionCount':
                pauseHistory_sectionCount_1[i] = column_dict[j]
```

In [3]:

```
j=0

new_id = []
pauseHistory_id_2 = []
pauseHistory_startAt_2 = []
pauseHistory_endAt_2 = []
pauseHistory_year_2 = []
pauseHistory_sectionCount_2 = []

new_columns2 = [new_id,
                pauseHistory_id_2,
                pauseHistory_startAt_2,
                pauseHistory_endAt_2,
                pauseHistory_year_2,
                pauseHistory_sectionCount_2]
```

Githus Gist: https://gist.github.com/jaehwan2/dfe44fbbe0ea917c3fb7bafa93eecc4e

Data Source: http://shorturl.at/qvRW0

지원자 **송재환**

개인 프로젝트

-고객 로그 데이터를 통해 유저의 상품별 주문 리텐션 분석

1. Data Table Column 정리 및 분리 : Python 활용

pauseHistory 테이블 만들기-3

pauseHistory

_id
pauseHistory_id_1
pauseHistory_startAt_1
pauseHistory_endAt_1
pauseHistory_year_1
PuastHistory_sectionCount_1
pauseHistory_id_2
pauseHistory_startAt_2
pauseHistory_endAt_2
pauseHistory_year_2
PuastHistory_sectionCount_2

```
for i in orders._id.index:

    new_id.append(orders._id[i])
    pauseHistory_id_2.append(None)
    pauseHistory_startAt_2.append(None)
    pauseHistory_endAt_2.append(None)
    pauseHistory_year_2.append(None)
    pauseHistory_sectionCount_2.append(None)

    if len(str(orders.pauseHistory[i])) > 191:

        column_temp = orders.pauseHistory[i].split('}, ')[1].replace("ObjectId(","").replace(")","'")
        column_temp = column_temp.replace("datetime.datetime(","'").replace("}","'}").replace("'year': 2020","'year': '
        column_temp = column_temp.replace("'sectionCount': ","'sectionCount': '").replace("0}","'0'}").replace("'","'"
        column_temp = column_temp.replace("'","\"").replace("[","").replace("]","")

        column_dict = json.loads(column_temp)

        column_dict.keys()
        column_dict.values()

        key_list = list(column_dict)

        for j in key_list:

            if j == '_id' :
                column_dict[j] = "".join(column_dict[j]).replace("[","").replace("]","")
                pauseHistory_id_2[i] = column_dict[j]

            elif j == 'startAt':
                column_dict[j] = column_dict[j].split(", ")
                column_dict[j][1] = column_dict[j][1].zfill(2)
                column_dict[j][2] = column_dict[j][2].zfill(2)
                column_dict[j] = ','.join(column_dict[j]).replace(",","",2).split(",")[0]
                pauseHistory_startAt_2[i] = column_dict[j]

            elif j =='endAt':
                column_dict[j] = column_dict[j].split(", ")
                column_dict[j][1] = column_dict[j][1].zfill(2)
                column_dict[j][2] = column_dict[j][2].zfill(2)
                column_dict[j] = ','.join(column_dict[j]).replace(",","",2).split(",")[0]
                pauseHistory_endAt_2[i] = column_dict[j]

            elif j =='year':
                pauseHistory_year_2[i] = column_dict[j]

            elif j =='sectionCount':
                pauseHistory_sectionCount_2[i] = column_dict[j]
```

Githus Gist: https://gist.github.com/jaehwan2/dfe44fbbe0ea917c3fb7bafa93eeee4e

Data Source: http://shorturl.at/qvRW0

지원자 **송재환**

개인 프로젝트

-고객 로그 데이터를 통해 유저의 상품별 주문 리텐션 분석

1. Data Table Column 정리 및 분리 : Python 활용

pauseHistory 테이블 만들기-4

pauseHistory
_id
pauseHistory_id_1
pauseHistory_startAt_1
pauseHistory_endAt_1
pauseHistory_year_1
PuastHistory_sectionCount_1
pauseHistory_id_2
pauseHistory_startAt_2
pauseHistory_endAt_2
pauseHistory_year_2
PuastHistory_sectionCount_2

In [4]:
```
df1 = pd.DataFrame(new_columns1).transpose()
df1.columns=["_id",
             "pauseHistory_id_1",
             "pauseHistory_startAt_1",
             "pauseHistory_endAt_1",
             "pauseHistory_year_1",
             "pauseHistory_sectionCount_1"]

df2 = pd.DataFrame(new_columns2).transpose()
df2.columns=["_id",
             "pauseHistory_id_2",
             "pauseHistory_startAt_2",
             "pauseHistory_endAt_2",
             "pauseHistory_year_2",
             "pauseHistory_sectionCount_2"]

df_join = pd.merge(left=df1, right=df2, how='left', on="_id")
df_join.to_excel("/Users/jaehwan/Downloads/orders_pauseHistory.xlsx")
```

Githus Gist: https://gist.github.com/jaehwan2/dfe44fbbe0ea917c3fb7bafa93eecc4e

Data Source: http://shorturl.at/qvRW0

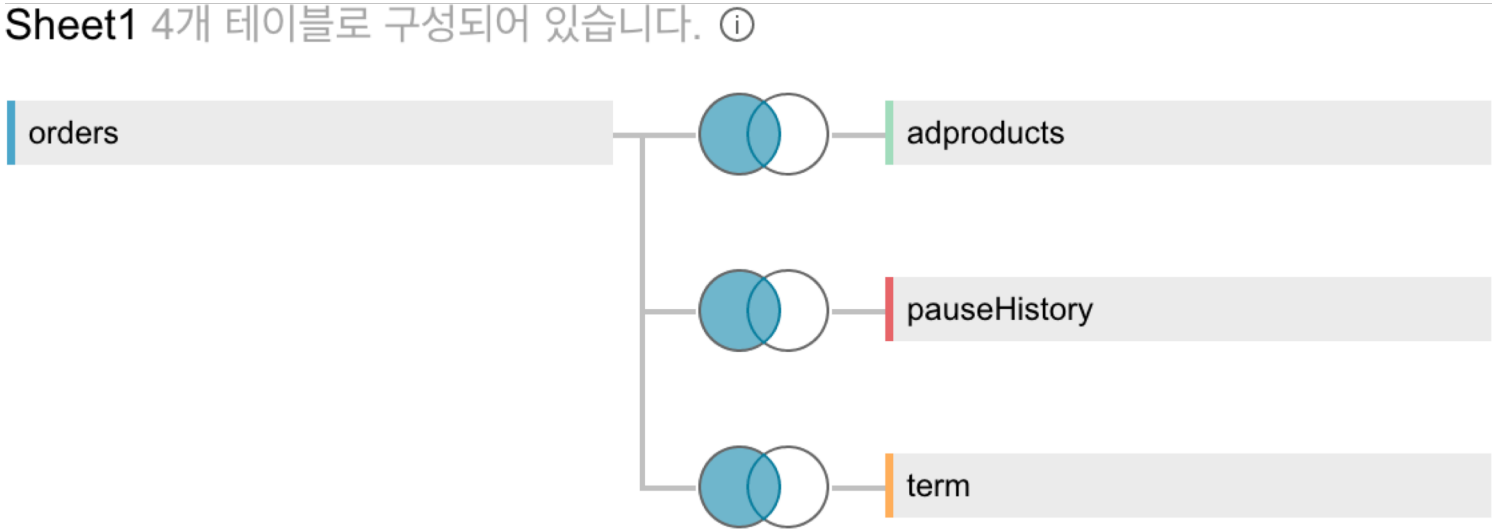
지원자 **송재환**

개인 프로젝트

-고객 로그 데이터를 통해 유저의 상품별 주문 리텐션 분석

2. Data Visualization : Tableau 활용

1) Data Load: [_id]를 기준으로 데이터 조인



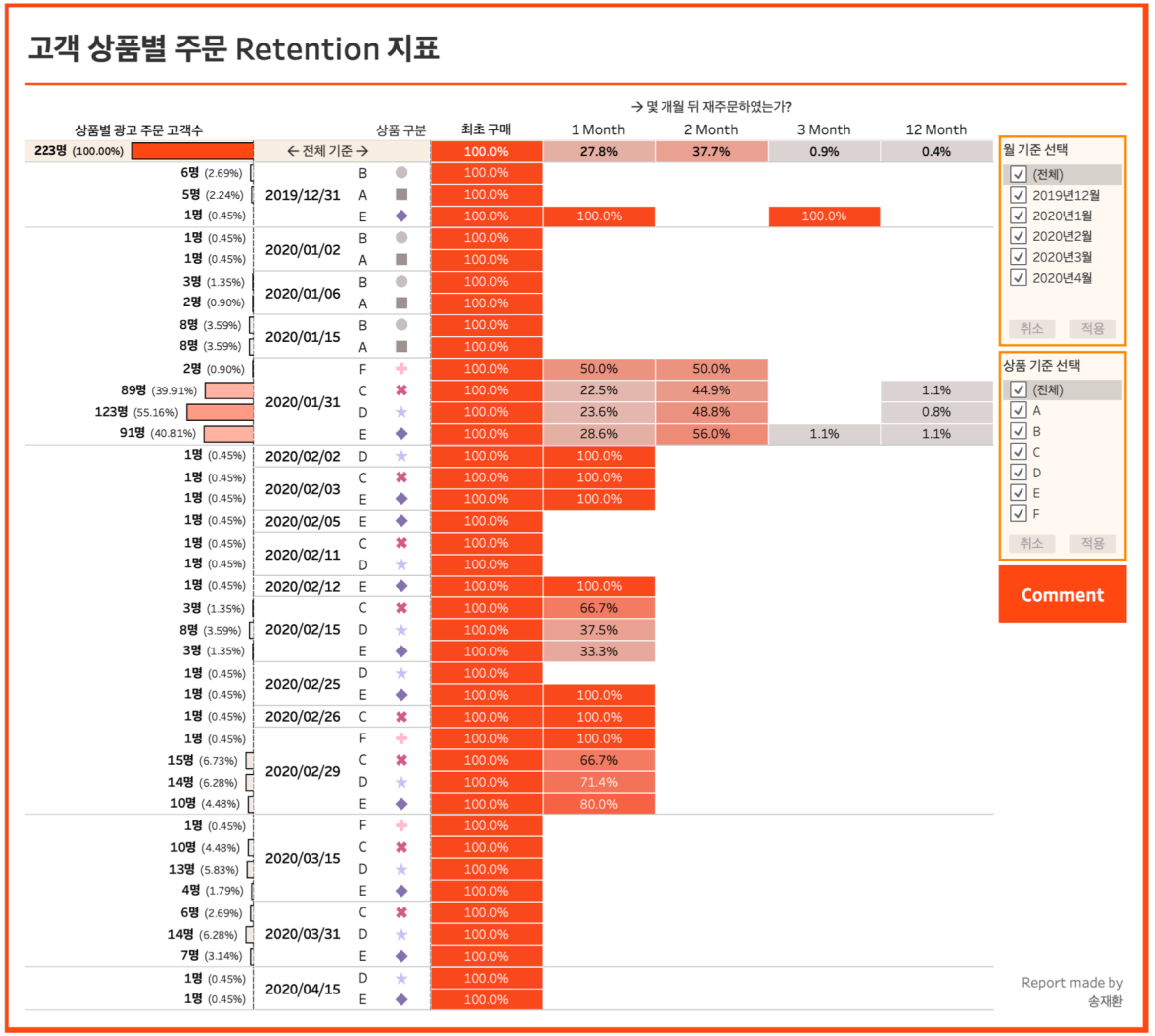
개인 프로젝트

-고객 로그 데이터를 통해 유저의 상품별 주문 리텐션 분석

2. Data Visualization : Tableau 활용

2) Dashboard 만들기

- 유저의 상품별 주문 리텐션 파악을 위한 Cohort 차트
  (같은 일자에 광고를 최초 주문한 집단의 재주문 여부 및 시기 비교)
- 일자별 최초 주문 건수 확인을 통해 Volumn 확인
- 월 기준, 상품 기준으로 필터링 가능
- 우측 하단 'Comment ' 버튼 클릭시 분석의 결과 텍스트로 공유



지원자 **송재환**

개인 프로젝트

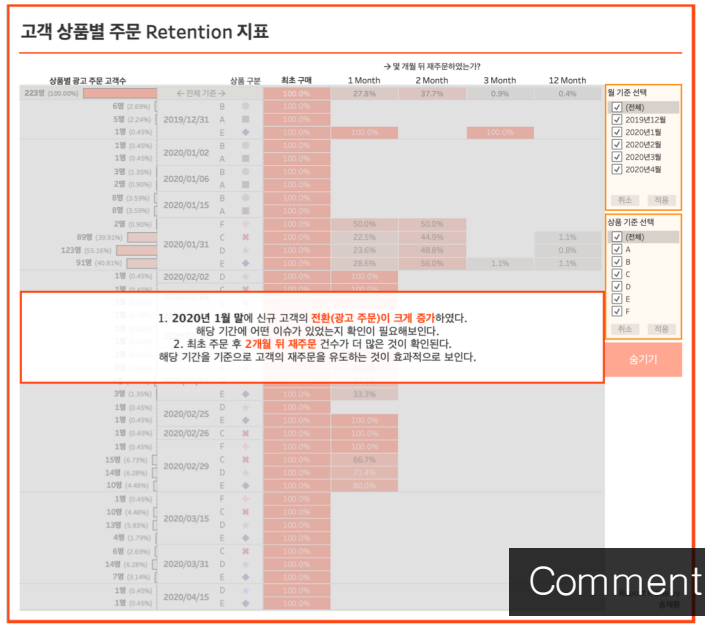
-고객 로그 데이터를 통해 유저의 상품별 주문 리텐션 분석

2. Data Visualization : Tableau 활용

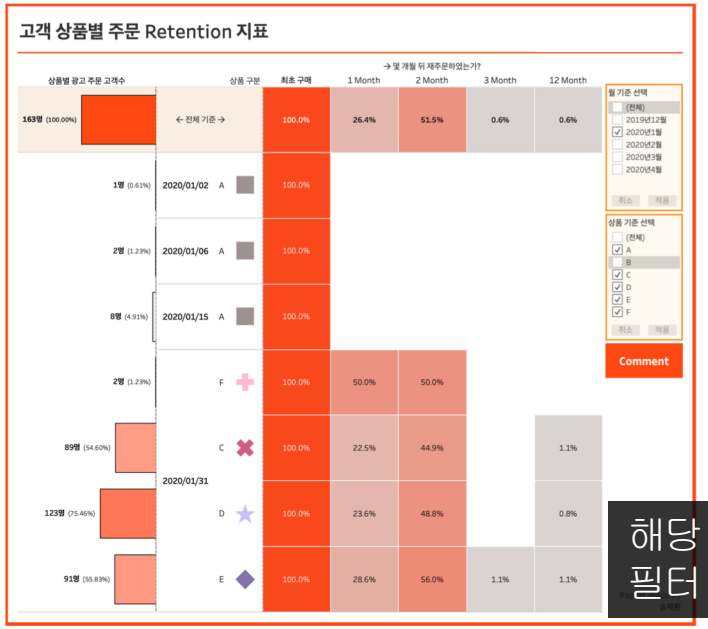
3) 분석 결과

1. 2020년 1월 말에 신규 고객의 전환(광고 주문)이 크게 증가하였다.
   해당 기간에 어떤 이슈가 있었는지 F/U 필요. (해당 시기 쿠폰 사용률 88%)

2. 최초 주문 후 2개월 뒤 재주문 건수가 더 많은 것이 확인된다.
   해당 기간을 기준으로 고객의 재주문을 유도하는 것이 효과적으로 보인다.



Comment 표시 예시



해당 기간
필터 적용 예시

지원자 **송재환**

고객 상품별 주문 Retention 지표

→ 몇 개월 뒤 재주문하였는가?

상품별 광고 주문 고객수		상품 구분	최초 구매	1 Month	2 Month	3 Month	12 Month
223명 (100.00%)	← 전체 기준 →		100.0%	27.8%	37.7%	0.9%	0.4%
6명 (2.69%)	2019/12/31	B	100.0%				
5명 (2.24%)		A	100.0%				
1명 (0.45%)		E	100.0%	100.0%		100.0%	
1명 (0.45%)	2020/01/02	B	100.0%				
1명 (0.45%)		A	100.0%				
3명 (1.35%)	2020/01/06	B	100.0%				
2명 (0.90%)		A	100.0%				
8명 (3.59%)	2020/01/15	B	100.0%				
8명 (3.59%)		A	100.0%				
2명 (0.90%)	2020/01/31	F	100.0%	50.0%	50.0%		
89명 (39.91%)		C	100.0%	22.5%	44.9%		1.1%
123명 (55.16%)		D	100.0%	23.6%	48.8%		0.8%
91명 (40.81%)		E	100.0%	28.6%	56.0%	1.1%	1.1%
1명 (0.45%)	2020/02/02	D	100.0%	100.0%			
1명 (0.45%)	2020/02/03	C	100.0%	100.0%			
1명 (0.45%)		E	100.0%	100.0%			
1명 (0.45%)	2020/02/05	E	100.0%				
1명 (0.45%)	2020/02/11	C	100.0%				
1명 (0.45%)		D	100.0%				
1명 (0.45%)	2020/02/12	E	100.0%	100.0%			
3명 (1.35%)	2020/02/15	C	100.0%	66.7%			
8명 (3.59%)		D	100.0%	37.5%			
3명 (1.35%)		E	100.0%	33.3%			
1명 (0.45%)	2020/02/25	D	100.0%				
1명 (0.45%)		E	100.0%	100.0%			
1명 (0.45%)	2020/02/26	C	100.0%	100.0%			
1명 (0.45%)	2020/02/29	F	100.0%	100.0%			
15명 (6.73%)		C	100.0%	66.7%			
14명 (6.28%)		D	100.0%	71.4%			
10명 (4.48%)		E	100.0%	80.0%			
1명 (0.45%)	2020/03/15	F	100.0%				
10명 (4.48%)		C	100.0%				
13명 (5.83%)		D	100.0%				
4명 (1.79%)		E	100.0%				
6명 (2.69%)	2020/03/31	C	100.0%				
14명 (6.28%)		D	100.0%				
7명 (3.14%)		E	100.0%				
1명 (0.45%)	2020/04/15	D	100.0%				
1명 (0.45%)		E	100.0%				

월 기준 선택

- [x] (전체)
- [x] 2019년12월
- [x] 2020년1월
- [x] 2020년2월
- [x] 2020년3월
- [x] 2020년4월

취소 적용

상품 기준 선택

- [x] (전체)
- [x] A
- [x] B
- [x] C
- [x] D
- [x] E
- [x] F

취소 적용

Comment

Report made by
송재환

2021. 3. 1 제출
지원자 **송재환**