

COUPON_DETAIL

- [COUPON_DETAIL_TABLE](#)
- [코드진행](#)
 - 라이브러리 목록
 - 데이터 불러오기
 - ITEM_COUNT별 구매량
 - 월별 쿠폰발급 분석(구매에 사용된)
 - 구매량이 많은 상위 20개 지역
 - 상위 20명의 유저별 총 구매수량
 - 많이 사용된 상위 20개의 쿠폰

COUPON_DETAIL_TABLE [🔗](#)

| Column Name | Description | Type | Length | Decimal | Note |
|-----------------|----------------------|----------|--------|---------|----------------------------------|
| ITEM_COUNT | Purchased item count | NUMBER | 10 | 0 | |
| I_DATE | Purchase date | DATE | | | |
| SMALL_AREA_NAME | Small area name | VARCHAR2 | 30 | | [JPN] User residential area name |
| PURCHASEID_hash | Purchase ID | VARCHAR2 | 32 | | |
| USER_ID_hash | User ID | VARCHAR2 | 32 | | |
| COUPON_ID_hash | Coupon ID | VARCHAR2 | 32 | | 1. |

1. ITEM_COUNT : 구매한 아이템 수 로 유형은 NUMBER이며 길이는 10입니다.
2. I_DATE : 쿠폰을 발급한 날짜로서 DATE 유형입니다.
3. SMALL_AREA_NAME : VARCHAR2유형이며 길이는 30 입니다. (특정 도시, 마을, 동네, 혹은 구역 등과 같은 상대적으로 작은 규모의 지리적 지역을 말합니다)
4. PURCHASEID_hash : 구매ID로 VARCHAR2 유형이며 길이는 32 입니다. - 주문시 적용되는 ID
5. USER_ID_hash : 사용자 ID로 VARCHAR2 유형이며 길이는 32 입니다.
6. COUPON_ID_hash : 쿠폰 ID로 VARCHAR2 유형이며 길이는 32 입니다.

코드진행 [🔗](#)

라이브러리 목록 [🔗](#)

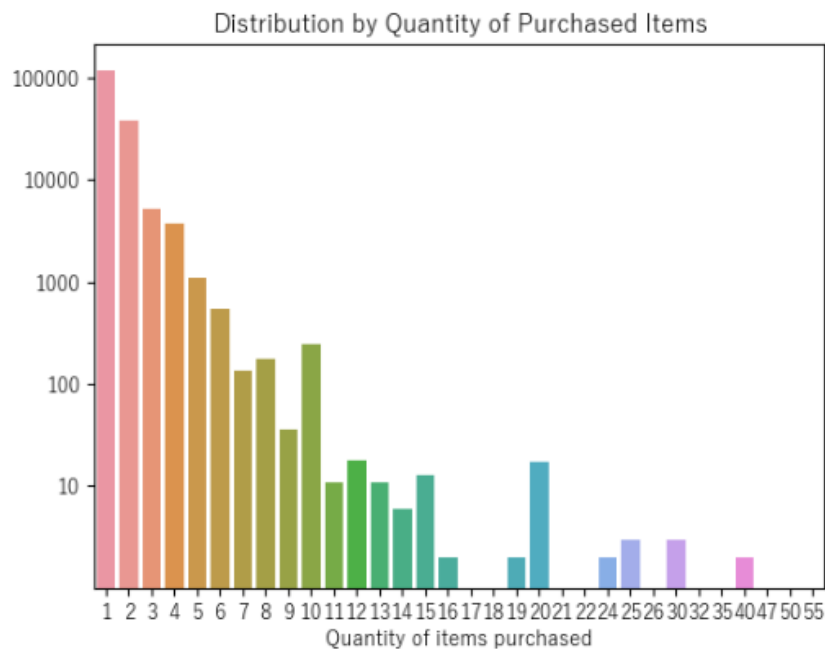
```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3 import seaborn as sns
4 import matplotlib as mpl
5 plt.rc('font', family = 'NanumGothic')
6 import numpy as np
7 import japanize_matplotlib
```

데이터 불러오기 [↗](#)

```
1 df = pd.read_csv('min_project_tables - coupon_detail_train.csv')
2 df.head()
```

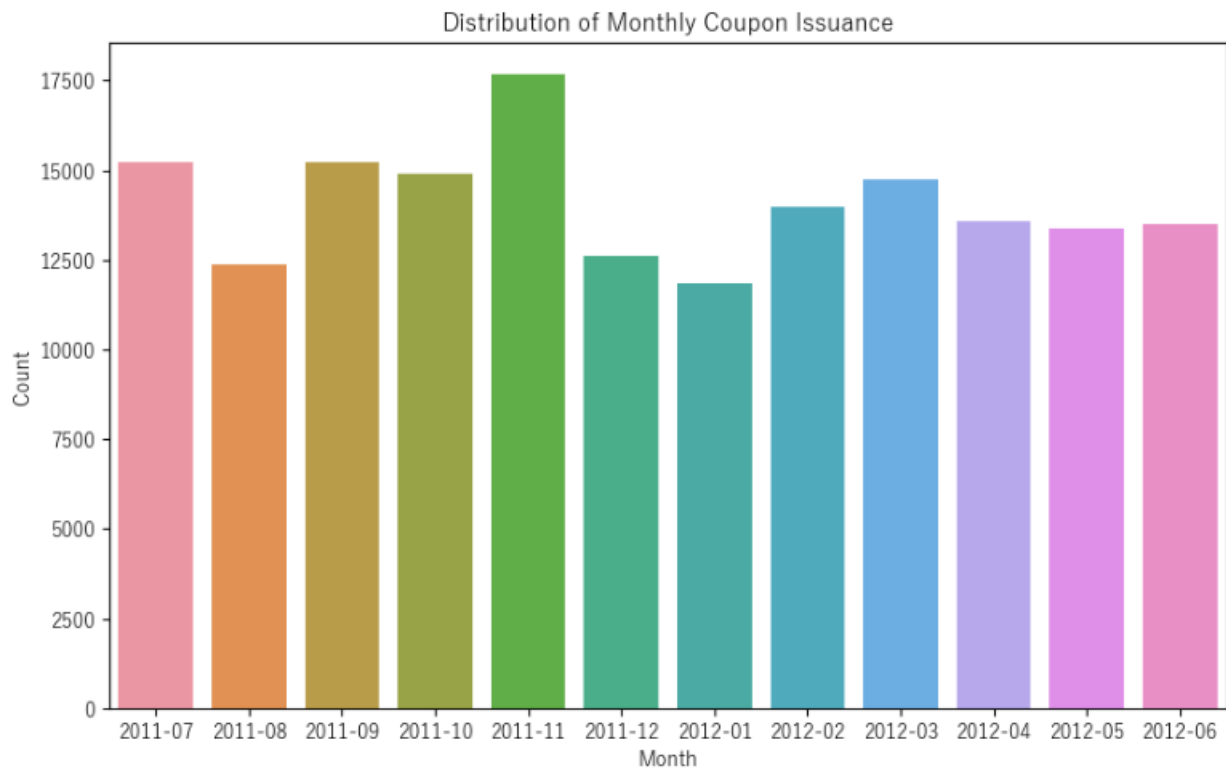
ITEM_COUNT별 구매량 [↗](#)

```
1 item_counts_log = np.log10(df['ITEM_COUNT'].value_counts())
2 sns.barplot(x=item_counts_log.index, y = item_counts_log.values)
3 plt.yticks(np.log10([10, 100, 1000, 10000, 100000]), [10, 100, 1000, 10000, 100000])
4 plt.xlabel('Quantity of items purchased')
5 plt.title('Distribution by Quantity of Purchased Items')
6 plt.show()
7
```



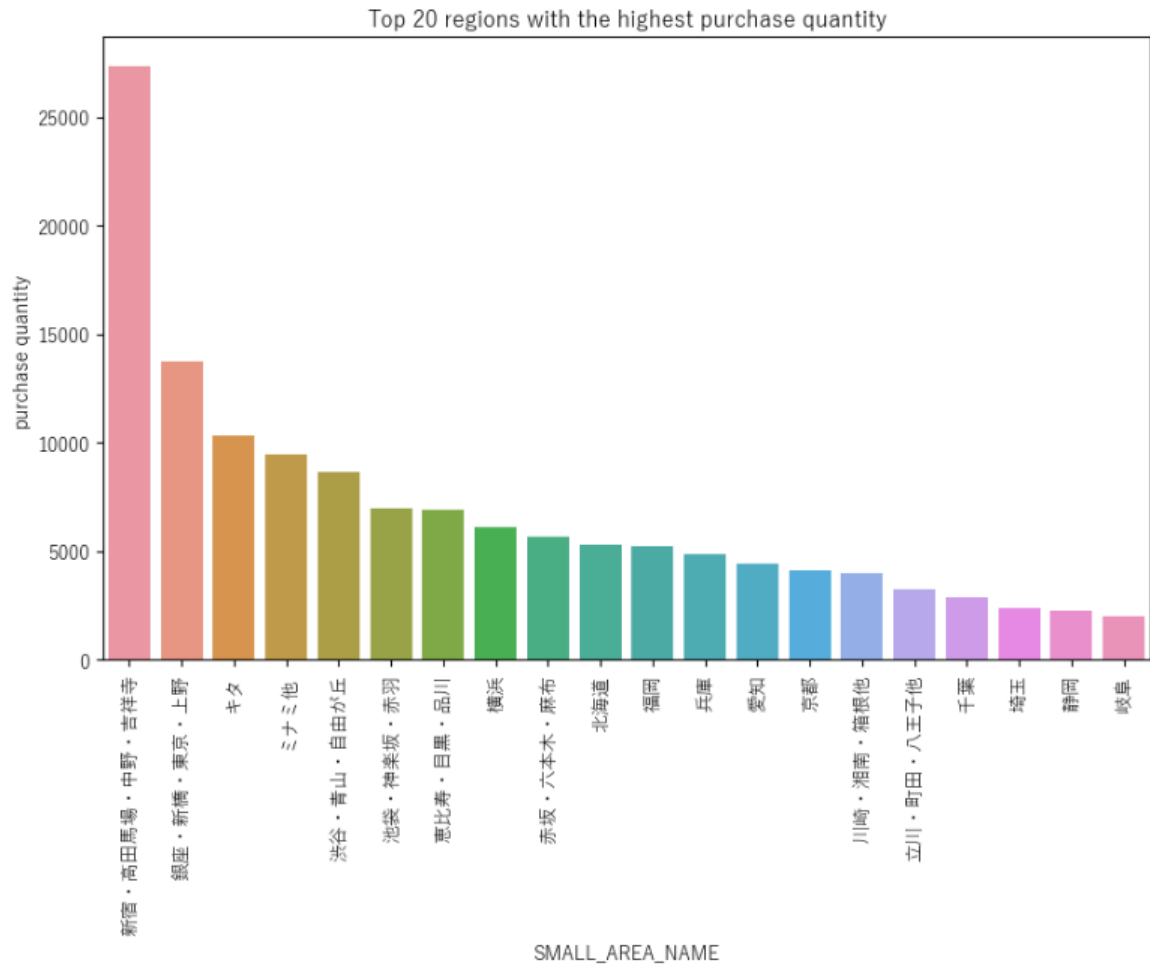
월별 쿠폰발급 분석(구매에 사용된) [↗](#)

```
1 df['I_DATE'] = pd.to_datetime(df['I_DATE'])
2
3 monthly_counts = df.resample('M', on='I_DATE').size()
4
5 monthly_data = pd.DataFrame({'Month': monthly_counts.index.strftime('%Y-%m'), 'Count': monthly_counts.values})
6
7 plt.figure(figsize=(10, 6))
8 sns.barplot(x='Month', y='Count', data=monthly_data)
9 plt.xlabel('Month')
10 plt.ylabel('Count')
11 plt.title('Distribution of Monthly Coupon Issuance')
12 plt.show()
```



구매량이 많은 상위 20개 지역 [↗](#)

```
1 area_counts = df['SMALL_AREA_NAME'].value_counts()
2
3 top_20_areas = area_counts[:20]
4
5 plt.figure(figsize=(10, 6))
6 sns.barpplot(x=top_20_areas.index, y=top_20_areas.values)
7 plt.xlabel('SMALL_AREA_NAME')
8 plt.ylabel('purchase quantity')
9 plt.title('Top 20 regions with the highest purchase quantity')
10 plt.xticks(rotation=90)
11 plt.show()
```

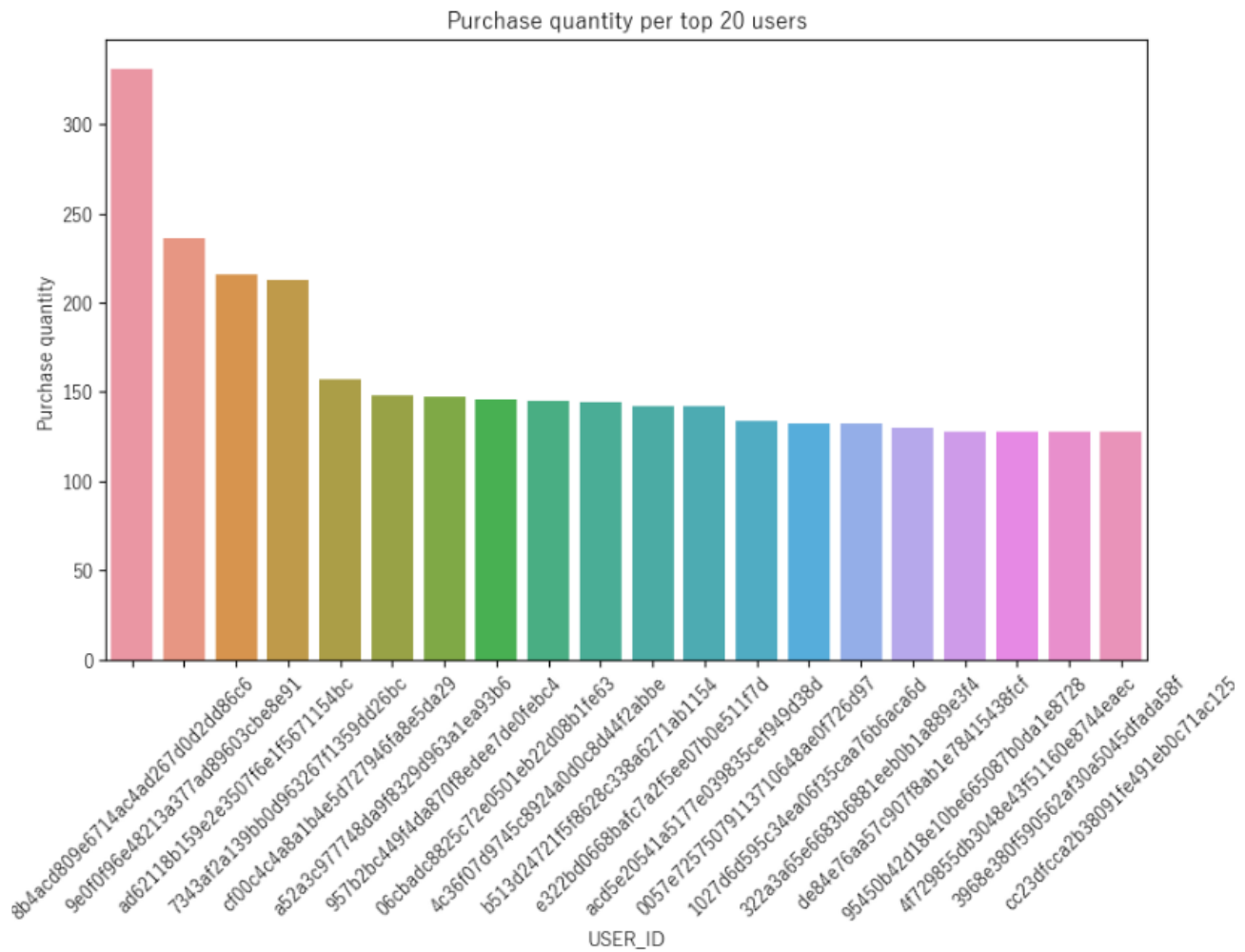


상위 20명의 유저별 총 구매수량 🔗

```

1 user_id_groupby = df.groupby('USER_ID_hash')['ITEM_COUNT'].sum()
2 sorted_user_id_groupby = user_id_groupby.sort_values(ascending=False)
3
4 top_20_users = sorted_user_id_groupby[:20]
5
6 plt.figure(figsize=(10, 6))
7 sns.barplot(x=top_20_users.index, y=top_20_users.values)
8 plt.xlabel('USER_ID')
9 plt.ylabel('Purchase quantity')
10 plt.title('Purchase quantity per top 20 users')
11 plt.xticks(rotation=45)
12 plt.show()

```

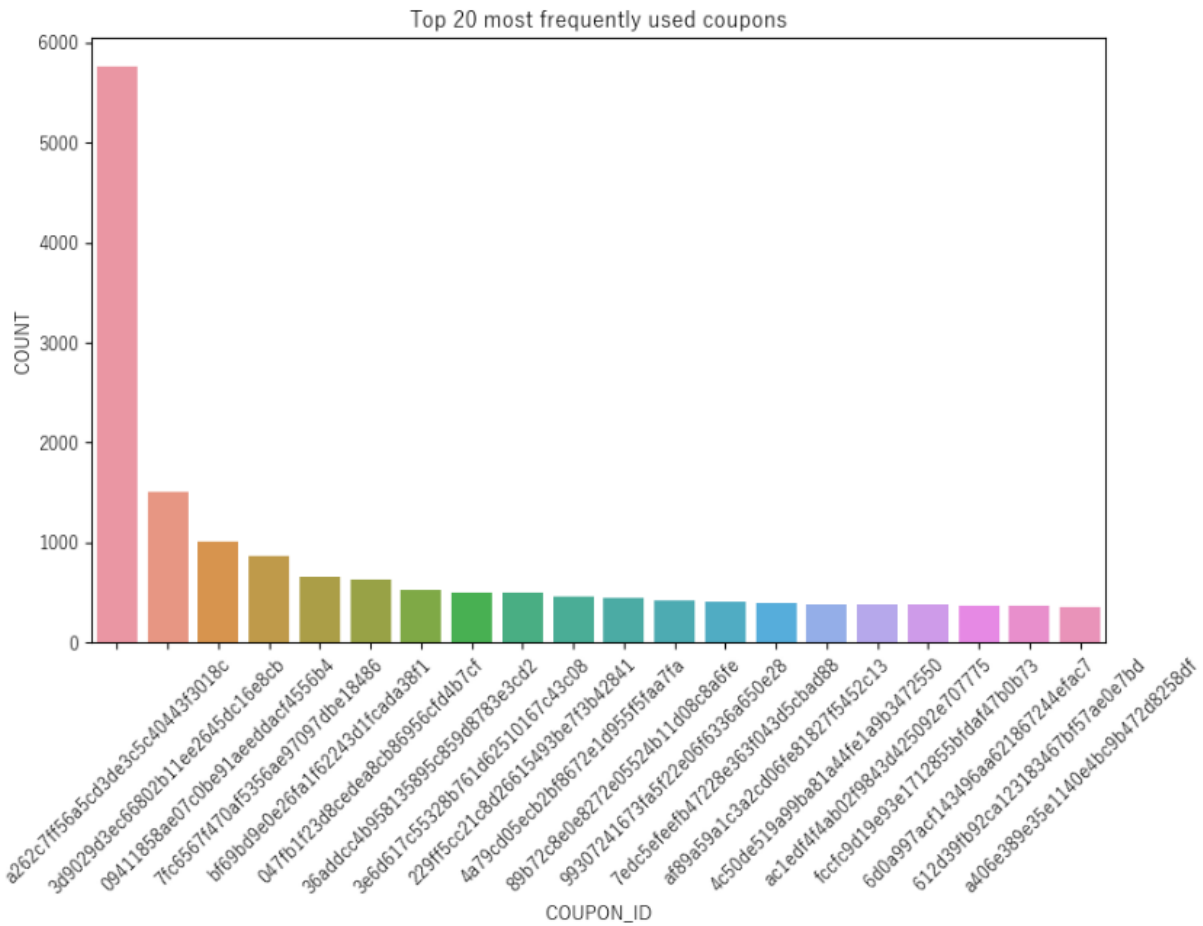


많이 사용된 상위 20개의 쿠폰 [↗](#)

```

1 coupon_counts = df['COUPON_ID_hash'].value_counts()
2
3 top_20_coupons = coupon_counts[0:20]
4
5 plt.figure(figsize=(10,6))
6 sns.barplot(x=top_20_coupons.index, y=top_20_coupons.values)
7 plt.xlabel('COUPON_ID')
8 plt.ylabel('COUNT')
9 plt.title('Top 20 most frequently used coupons')
10 plt.xticks(rotation=45)
11 plt.show()

```



- 쿠폰이 구매 수량 전체에 다 적용되는지 확인 할 수 없어서 일단 쿠폰 번호별로 카운트만 했습니다.