

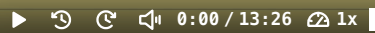
10-3. 데이터 분석

03

AIFTEL

데이터 분석

1. 데이터 분석
2. 전략 수립 및 실행



앞서가는 차세대 데이터 교육 AIFTEL

Copyright©2022. AIFTEL corp. All Rights Reserved.

3. 데이터 분석

3-1 데이터 불러오기



```
1 # 필요 라이브러리 호출하기
2 import pandas as pd
3 import numpy as np
4 import matplotlib.pyplot as plt
5 import seaborn as sns
6
7 plt.style.use('default')
```

실행 완료



```
1 # 데이터 호출하기
2 entry_data = pd.read_csv('/aiffel/data/entry_data.csv')
3 sales_data = pd.read_csv('/aiffel/data/sales_data.csv')
4 item_data = pd.read_csv('/aiffel/data/item_data.csv')
5 item_entry_data = pd.read_csv('/aiffel/data/item_entry_data.csv')
```

실행 완료



```
1 # entry_data 데이터 정보 확인하기
2 entry_data.head()
```

실행 완료

[60]:

	요금제명	입실	퇴실	이용	위치	금액	구분	고객
0	시간요금제(1시간)	2022-04-01 23:39	2022-04-02 0:39	오전 1:00:00	선불기2	3000	퇴실	customer263717
1	시간요금제(1시간)	2022-04-01 23:39	2022-04-02 0:39	오전 1:00:00	선불기2	3000	퇴실	customer263717
2	무제한 입장권	2022-04-01 20:55	2022-04-01 21:41	오전 12:46:00	선불기2	7400	퇴실	customer247263





```
1 # sales_data 데이터 정보 확인하기
2 sales_data.head()
```

실행 완료

[61]:

	결제일시	판매위치	상품명	결제방식	비고	판매수량	결제금액	고객
0	2022-04-01 19:41	선불기1	치킨카라아케 6pc	신용카드	결제완료	1	4,500	customer275507
1	2022-04-01 19:41	선불기1	짜장라면(계란추가)	신용카드	결제완료	1	5,500	customer275507
2	2022-04-01 19:41	선불기1	라면	신용카드	결제완료	1	4,000	customer275507
3	2022-04-01 19:41	선불기1	비엔나소세지(토핑)	신용카드	결제완료	1	1,000	customer275507
4	2022-04-01 19:41	선불기1	라면(계란/치즈추가)	신용카드	결제완료	1	5,000	customer275507



```
1 # item_data 데이터 정보 확인하기
2 item_data.head()
```

실행 완료

[62]:

	구분	분류	상품명	판매금액
0	1	게임&TV	이어폰(커널형)	3,000원
1	2	디저트	초코머핀	3,500원
2	3	디저트	쿠키(초코칩)	1,500원
3	4	디저트	쿠키(마카다미아)	1,500원
4	5	디저트	미니마카롱(3개)	2,000원



```
1 # item_entry_data 데이터 정보 확인하기
2 item_entry_data.head()
```

실행 완료

[63]:

	순위	분류	상품명	판매금액
0	1	한국대생	한국대생 요금제	2,000원
1	2	심야-조조	심야-조조 요금제	5,000원
2	3	일반인	일반요금제	2,500원
3	4	한국대생	한국대생 요금제	2,500원
4	5	일반인	일반요금제	2,000원

3-2 데이터 분석하기

필수 문제1 : 월별 매출액 추세(sales_data를 이용합니다.)

월별 매출액 추세를 집계하고 그 결과를 시각화해봅시다.

힌트: `groupby()` 함수를 사용하세요.



```
1 # 문제 1: 월별 매출액 추세
2 # 활용할 데이터(sales_data)를 temp_var로 정의하기
3 temp_var = sales_data.copy()
4
5 # temp_var의 date열을 만들고, sales_data의 결제일시 중 앞에서 7번째 값까지 할당
6 temp_var['date'] = pd.to_datetime(temp_var['결제일시'])
7 temp_var['date'] = temp_var['date'].dt.strftime('%Y-%m')
8 temp_var.head(3)
```

실행 완료

[64]:

	결제일시	판매위치	상품명	결제방식	비고	판매수량	결제금액	고객	date
0	2022-04-01 19:41	선불기1	치킨카라아케 6pc	신용카드	결제완료	1	4,500	customer275507	2022-04
1	2022-04-01 19:41	선불기1	짜장라면(계란추가)	신용카드	결제완료	1	5,500	customer275507	2022-04

sales_data의 결제금액에는 콤마(,)가 들어있어서 먼저 문자열로 인식되고 있었요. 따라서 콤표를 시우고, 문자열을 숫자로 바꿔줘야 합니다.

```
1 # sales_data의 결제금액에 포함된 콤표(,) 지우기
2 temp_var['결제금액'] = temp_var['결제금액'].str.replace(',','')
3
4 # sales_data의 결제금액을 숫자형태로 바꾸기
5 temp_var['결제금액'] = temp_var['결제금액'].astype(int)
6
7 # temp_var의 payment열을 date 기준으로 집계하기
8 temp_var.groupby('date')['결제금액'].sum().reset_index()
```

실행 완료 ✓

[65]:

	date	결제금액
0	2021-02	7000
1	2021-03	2104500
2	2021-04	3116600
3	2021-05	2537000
4	2021-06	3621500
5	2021-07	1215000
6	2021-08	1229000
7	2021-09	1197500
8	2021-10	2138500
9	2021-11	2748500
10	2021-12	3290000
11	2022-01	1430500
12	2022-02	1181500
13	2022-03	1870900
14	2022-04	49900

```
1 payment_data = temp_var.groupby('date')['결제금액'].sum().reset_index()
2 payment_data
```

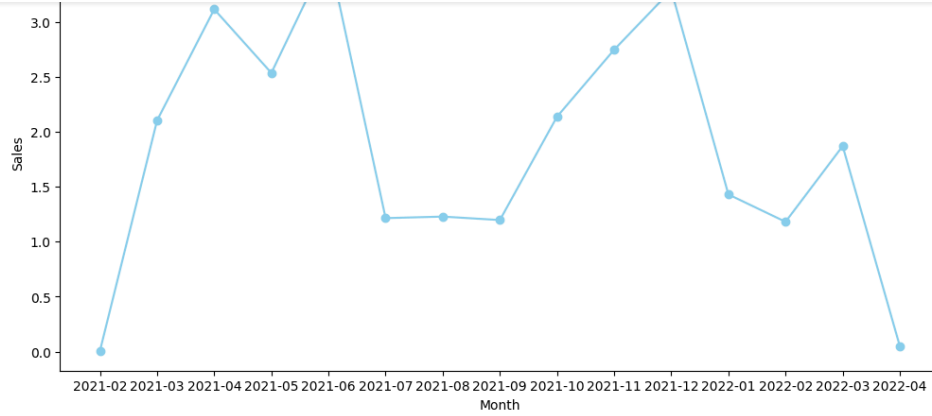
실행 완료 ✓

[67]:

	date	결제금액
0	2021-02	7000
1	2021-03	2104500
2	2021-04	3116600
3	2021-05	2537000
4	2021-06	3621500
5	2021-07	1215000
6	2021-08	1229000
7	2021-09	1197500
8	2021-10	2138500
9	2021-11	2748500
10	2021-12	3290000
11	2022-01	1430500
12	2022-02	1181500
13	2022-03	1870900
14	2022-04	49900

```
1 # 집계 결과를 시각화하기
2 plt.figure(figsize=(12, 6))
3 plt.plot(payment_data['date'], payment_data['결제금액'], marker='o', linestyle='--', color='skyblue')
4 plt.title('Sales per Month')
5 plt.xlabel('Month')
6 plt.ylabel('Sales')
7 plt.show()
```

실행 완료 ✓



• 필수 문제2: 가장 인기 있는 음식/음료(sales_data를 이용합니다.)

여러 메뉴들(음식,음료)을 판매 개수 순으로 정렬하고 상위 10개를 뽑아 결과를 시각화해봅니다.

힌트: `value_counts()` 함수를 사용하세요.

```

1 # 문제 2: 가장 인기 있는 음식/음료
2 # temp_var의 상품명을 value_counts를 이용해 집계하기
3 # 주의! value_counts로 바로 집계하면 판매취소와 결제취소가 함께 집계됩니다.
4 # 따라서 '판매수량' 칼럼 값이 '-1'인 데이터를 제외하고 집계해야 합니다.
5 # 힌트 코드 : temp_var[temp_var['판매수량']==-1]
6 popular = temp_var[temp_var['판매수량']!=-1]
7 popular['상품명'].value_counts()
8
9 # 집계 결과를 데이터프레임으로 변환하고, 상위 10개 선정하기
10 popular_df = popular['상품명'].value_counts().reset_index()
11 popular_df.columns = ['상품명', '판매 개수']
12 popular_df.head(10)

```

실행 완료 ✓

[69]:

	상품명	판매 개수
0	김치볶음밥	1204
1	라면	749
2	짜장라면	455
3	치즈불고기볶음밥	420
4	계란추가	314
5	낙지볶음밥	312
6	새우볶음밥	309
7	아메리카노(ICE)	284
8	치즈감자튀김	246
9	감자튀김	242

```

1 items_count = popular_df.head(10)
2 items_count

```

실행 완료 ✓

[70]:

	상품명	판매 개수
0	김치볶음밥	1204
1	라면	749
2	짜장라면	455
3	치즈불고기볶음밥	420
4	계란추가	314
5	낙지볶음밥	312
6	새우볶음밥	309
7	아메리카노(ICE)	284
8	치즈감자튀김	246
9	감자튀김	242

```

3 import matplotlib.pyplot as plt
4 import matplotlib.font_manager as fm
5
6 fontpath = '/usr/share/fonts/truetype/nanum/NanumBarunGothic.ttf'
7 font = fm.FontProperties(fname=fontpath, size=9)
8 plt.rc('font', family='NanumBarunGothic')
9 mpl.font_manager.findfont(font)
10 print("완료!")

```

실행 완료 ✓

완료!



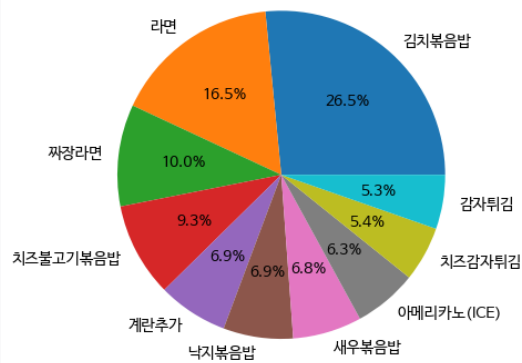
```

1 # 집계 결과를 시각화하기
2 plt.pie(items_count['판매 개수'], labels=items_count['상품명'], autopct = '%1.1f%%')
3 plt.title('가장 인기있는 음식 / 음료')
4 plt.show()

```

실행 완료 ✓

가장 인기있는 음식 / 음료



여러 메뉴들(음식,음료)을 판매 매출(판매 개수 x 결제 금액) 순으로 정렬하고 상위 10개를 뽑아 결과를 시각화해봅시다.

힌트: `value_counts()` 함수를 사용하세요.



```

1 # temp_var의 payment열을 상품명으로 집계하기
2 temp_var['payment'] = temp_var['판매수량'] * temp_var['결제금액']
3 item_payment = temp_var.groupby('상품명')['payment'].sum()
4
5 # item_payment의 값을 내림차순으로 정렬하고 상위 10개 도출하기
6 item_payment = item_payment.sort_values(ascending=False).head(10)
7 item_payment

```

실행 완료 ✓

[73]:

```

상품명
김치볶음밥      6036900
라면            2967000
치즈불고기볶음밥  2309100
짜장라면        2044500
낙지볶음밥      1721700
새우볶음밥      1549800
치즈감자튀김    1359700
아메리카노(ICE)  1180100
감자튀김        1083800
불닭볶음면      697500
Name: payment, dtype: int64

```



```

1 items_payment = item_payment.sort_values(ascending=False).head(10).reset_index()
2 items_payment.columns = ['상품명', '판매 금액']
3 items_payment

```

실행 완료 ✓

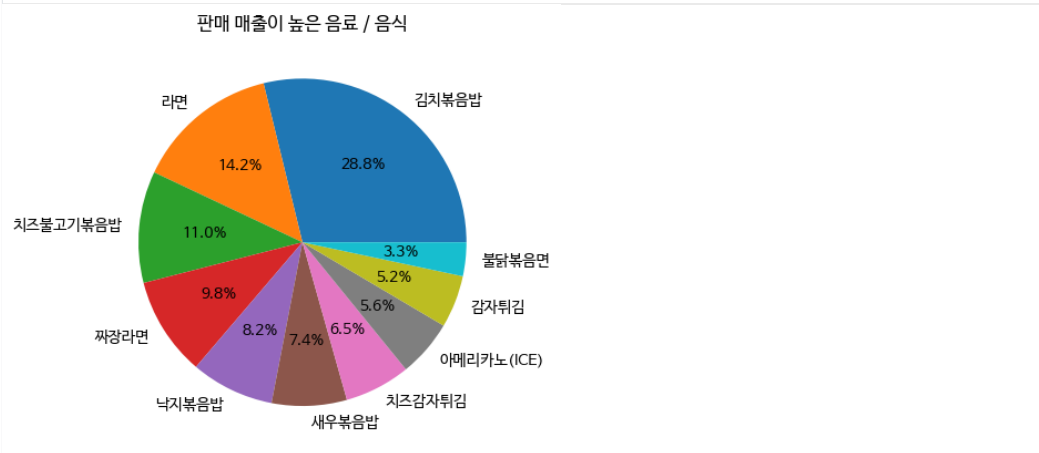
1	라면	2967000
2	치즈불고기볶음밥	2309100
3	짜장라면	2044500
4	낙지볶음밥	1721700
5	새우볶음밥	1549800
6	치즈감자튀김	1359700
7	아메리카노(ICE)	1180100
8	감자튀김	1083800
9	불닭볶음면	697500

```

1 # 집계 결과 시각화하기
2 plt.pie(items_payment['판매 금액'], labels=items_payment['상품명'], autopct = '%1.1f%%')
3 plt.title('판매 매출이 높은 음료 / 음식')
4 plt.show()

```

실행 완료 ☑



• 필수 문제3: 가장 많이 팔린 입장권 종류(entry_data를 이용합니다.)

여러 입장권 종류를 판매 매출(판매 개수 x 금액) 순으로 정렬하고 상위 10개를 뽑아 결과를 시각화해봅니다.

힌트: value_counts() 함수를 사용하세요.

```

1 # 문제 3: 가장 많이 팔린 입장권 종류
2 # 활용할 데이터(entry_data)를 temp_var로 정의하기
3 temp_var = entry_data.copy()
4 temp_var.head(3)
5
6 # temp_var의 금액을 숫자형태로 변환하기
7 temp_var['금액'] = temp_var['금액'].astype(int)
8 temp_var.head(3)
9
10 # temp_var의 금액을 요금제명을 기준으로 집계하기
11 #tickets = temp_var.groupby('요금제명')['금액'].sum()
12
13 # tickets의 값을 내림차순으로 정렬하고 상위 10개 도출하기
14 #tickets.sort_values(ascending=False).head(10)

```

실행 완료 ☑

[76]:

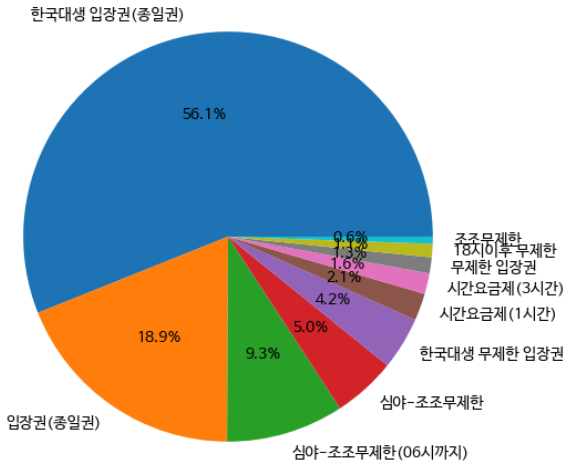
	요금제명	입실	퇴실	이용	위치	금액	구분	고객
0	시간요금제(1시간)	2022-04-01 23:39	2022-04-02 0:39	오전 1:00:00	선불기2	3000	퇴실	customer263717
1	시간요금제(1시간)	2022-04-01 23:39	2022-04-02 0:39	오전 1:00:00	선불기2	3000	퇴실	customer263717
2	무제한 입장권	2022-04-01 20:55	2022-04-01 21:41	오전 12:46:00	선불기2	7400	퇴실	customer247263

```

1 # 집계 결과 시각화하기
2 tickets_df = tickets.sort_values(ascending=False).head(10).reset_index()
3 tickets_df.columns = ['요금제', '판매 매출']
4 plt.figure(figsize=(10,6))
5 plt.pie(tickets_df['판매 매출'], labels=tickets_df['요금제'], autopct='%1.1f%%')
6 plt.title('가장 많이 팔린 입장권 종류')
7 plt.show()

```

가장 많이 팔린 입장권 종류



내가 정의한 문제 해결하기

■ □

1 # 내 Key Question을 중심으로 다양한 분석을 진행해봅시다.
2 temp_var = item_data.copy()
3 temp_var.head(3)

실행 완료

[78]:

	구분	분류	상품명	판매금액
0	1	게임&TV	이어폰(커널형)	3,000원
1	2	디저트	초코머핀	3,500원
2	3	디저트	쿠키(초코칩)	1,500원

■ □

1 # 판매금액 숫자로 변경하기
2 temp_var['판매금액'] = temp_var['판매금액'].str.replace(',','')
3 temp_var['판매금액'] = temp_var['판매금액'].str.replace('원','')
4 temp_var['판매금액'] = temp_var['판매금액'].astype(int)
5 temp_var.info()

실행 완료

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 174 entries, 0 to 173  
Data columns (total 4 columns):  
#   Column  Non-Null Count  Dtype  
---  ---  
0   구분    174 non-null     int64  
1   분류    174 non-null     object  
2   상품명  174 non-null     object  
3   판매금액 174 non-null     int64  
dtypes: int64(2), object(2)  
memory usage: 5.6+ KB
```

■ □

1 # 상품별 판매 금액 살펴보기
2 temp_var.groupby('상품명')['판매금액'].sum()

실행 완료

[80]:

상품명	
(면)계란추가-국물라면	500
(면)계란후라이 추가-불닭/짜장	1000
(면)치즈추가	500
(밥)계란후라이 추가	1000
(밥)슈레드치즈추가	1500

```
헤이즐넛라떼 (HOT)          4500
헤이즐넛라떼 (ICE)          7500
Name: 판매금액, Length: 98, dtype: int64
```



```
1 # 가장 많이 팔린 상품 상위 10개 살펴보기
2 popular_item = temp_var.groupby('상품명')['판매금액'].sum().sort_values(ascending=False).head(10)
3 popular_item
```

실행 완료 ✓

[81]:

```
상품명
김치볶음밥          25200
치즈불고기볶음밥    19900
낙지볶음밥          15400
달기스무디          14900
아메리카노 (ICE)    14400
새우볶음밥          14400
쿠앤크셰이크        14400
플레인요거트스무디  13900
레몬에이드          13400
치즈감자튀김        11400
Name: 판매금액, dtype: int64
```



```
1 # 데이터 프레임 형태로 변환하기
2 popular_item = popular_item.reset_index()
3 popular_item.columns = ['상품명', '판매 금액']
4 popular_item
```

실행 완료 ✓

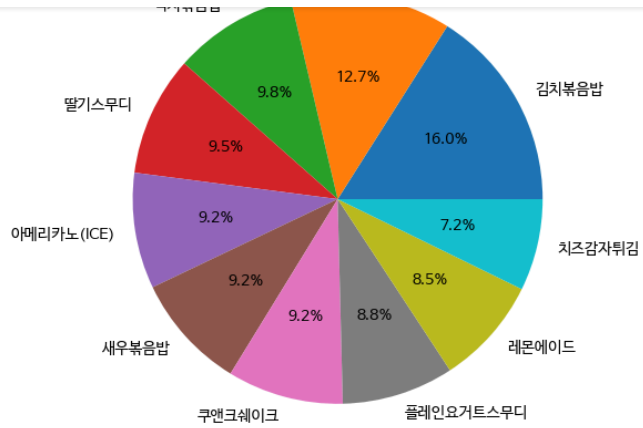
[82]:

	상품명	판매 금액
0	김치볶음밥	25200
1	치즈불고기볶음밥	19900
2	낙지볶음밥	15400
3	달기스무디	14900
4	아메리카노(ICE)	14400
5	새우볶음밥	14400
6	쿠앤크셰이크	14400
7	플레인요거트스무디	13900
8	레몬에이드	13400
9	치즈감자튀김	11400



```
1 # 파이 차트로 시각화 하기
2 plt.figure(figsize = (10, 6))
3 plt.pie(popular_item['판매 금액'], labels=popular_item['상품명'], autopct='%1.1f%%')
4 plt.title('가장 많이 팔리는 상품')
5 plt.show()
```

실행 완료 ✓



```

1 # 가장 많이 팔린 분류 상위 10개 살펴보기
2 popular_item = temp_var.groupby('분류')['판매금액'].sum().sort_values(ascending=False).head(10)
3
4 # 데이터 프레임 형태로 변환하기
5 popular_item = popular_item.reset_index()
6 popular_item.columns = ['분류', '판매 금액']
7 popular_item

```

실행 완료 ✓

[84]:

	분류	판매 금액
0	음식	212900
1	차가운음료(커피외)	126800
2	커피	97200
3	스무디&쉐이크	86200
4	따뜻한음료(커피외)	65200
5	디저트	28500
6	포장	25000
7	게임&TV	3000

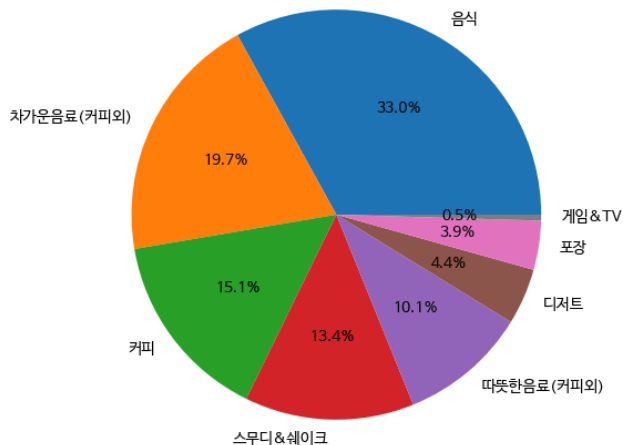
```

1 # 파이 차트로 시각화 하기
2 plt.figure(figsize = (10, 6))
3 plt.pie(popular_item['판매 금액'], labels=popular_item['분류'], autopct='%1.1f%%')
4 plt.title('가장 많이 팔리는 상품')
5 plt.show()

```

실행 완료 ✓

가장 많이 팔리는 상품





개인정보처리방침 이용약관

(주)모두의연구소 사업자 정보

대표 김승일 | 개인정보보호책임자 안우진 | 서울특별시 강남구 강남대로 324 역삼디오슈퍼리움 2층 | cs@aiffel.io | 사업자등록번호: 517-88-00184 | 통신판매업신고: 2017-서울강남-04920 | 고객센터: 070 - 7743 - 5882 | 주식회사 모두의연구소는 전자상거래 등에서의 소비자보호에 관한 법률에 따른 통신판매업을 영위하고 있습니다.

Copyright ©2025 AIFFEL. All Rights Reserved.