

# 코로나 기간(2020~2022년) 서울시 아파트 매매 동향

(데이터 분석 **with Python**)

- 권병철

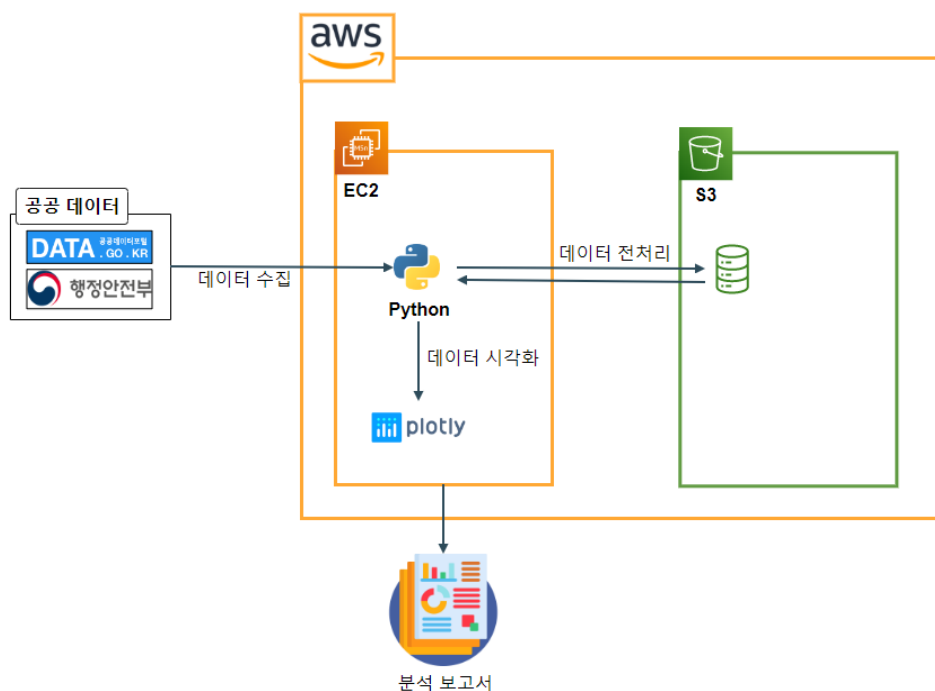
코로나 기간 3년 동안의 서울시 아파트 매매 실거래 동향을 알아보고자 한다. 아파트의 매매 가격은 공공데이터 포털에서 구하였으며 자료는 2020년 ~ 2022년으로 선정하였다. 해당 데이터 분석에 필요한 Tool은 다음과 같다.

- 분석 환경: AWS의 EC2
- 분석 언어: Python
- 데이터 적재 공간: S3
- 시각화 툴: Plotly
- 데이터 출처
  - 아파트매매자료: 공공데이터포털
  - 법정동코드: 행정안전부

## 분석 계획

서울시 아파트 매매 가격 동향에 대한 전반적인 시각화를 진행 해보려 한다. 서울시, 시군구별, 읍면동별로 나누어서 지역에 따른 차이, 날짜에 따른 차이에 대한 상황과 그에 따른 해석을 진행할 것이다. 또한 각 카테고리 데이터 수집에서 사용한 파이썬 코드도 작성해보자 .

아키텍처는 다음과 같다.



## 1. 라이브러리 로드

```
import pandas as pd
import os
import numpy as np
import plotly.express as px
import plotly.graph_objects as go
from plotly.subplots import make_subplots

os.chdir("/home/ubuntu/realestate")
print(os.getcwd())
```

## 2. 데이터 로드 및 전처리

데이터는 공공데이터 포털에서 수집한 데이터를 **Raw**데이터로 **S3**에 적재 한 후 간단한 전처리를 한 다음 **Ec2**로 가져왔다. 그리고 법정동 코드와 맵핑하여 주소, 지역명을 넣어 주었다.

```
AptTrade_Total = pd.read_csv("data/AptTrade_Total.csv")
AptTrade_Total_3 = pd.read_csv("data/AptTrade_Total_3.csv")
AptTrade_Total_4 = AptTrade_Total_3[AptTrade_Total_3['년'] == 2022]
AptTrade_Total_4 = AptTrade_Total_4[AptTrade_Total_4['월'] == 12]
AptTrade_Total_4 = AptTrade_Total_4.reset_index(drop = True)

AptTrade_Total =
AptTrade_Total.drop(['거래유형', '중개사소재지', '해제사유발생일', '해제여부'],
, axis=1)

regal_code = pd.read_csv('data/regal_code.csv')
regal_code = regal_code.rename(columns={"법정동시군구코드": "지역코드"})
regal_code = regal_code.where(pd.notnull(regal_code), " ")

regal_code['시도명'] = regal_code['시도명'].str.strip()
regal_code['시군구명'] = regal_code['시군구명'].str.strip()
regal_code['읍면동명'] = regal_code['읍면동명'].str.strip()
regal_code['동리명'] = regal_code['동리명'].str.strip()

regal_code['주소'] = regal_code['시도명'] + " " + \
                    regal_code['시군구명'] + " " + \
                    regal_code['읍면동명'] + " " + \
                    regal_code['동리명']

regal_code['주소'] = regal_code['주소'].str.replace(' ', ' ')
regal_code['주소'] = regal_code['주소'].str.strip()
```

```

AptRent_Total = pd.merge(AptRent_Total, regal_code[['지역코드',
'시군구명','주소']],
    on = '지역코드',
    how = 'left')

AptTrade_Total = pd.merge(AptTrade_Total, regal_code[['지역코드',
'시군구명','주소']],
    on = '지역코드',
    how = 'left')

```

### 3. 주택 매매 동향 (2020~2022년 3년간)

데이터 로드 및 전처리가 완료 되었으므로 코로나 기간 3년간의 서울시 아파트의 주택 매매 동향에 대해 알아보려 한다. 우선 서울시 전체 지도를 plotly를 사용하여 시각화하여 나타내보았다. 지도 각 동그라미 크기는 3년간 평균 거래금액의 크기를 의미한다.

#### 3.1 서울시 전체 지도

```

fig = px.scatter_mapbox(AptTrade_Total_4,
    lat="y",
    lon="x",
    hover_data={
        "x" : False,
        "y" : False,
        "주소" : True,
        "건축년도": True,
        "아파트": True,
    },
    color = '시군구명',
    size = '거래금액',
    height = 600,
    zoom=10)

fig.update_layout(
    mapbox_style="carto-positron",
    coloraxis_showscale=False,
    # showlegend=False,
    margin={"r":0,"t":0,"l":0,"b":0},
    hoverlabel=dict(
        bgcolor='white',
        font_size=15,
    ),
)

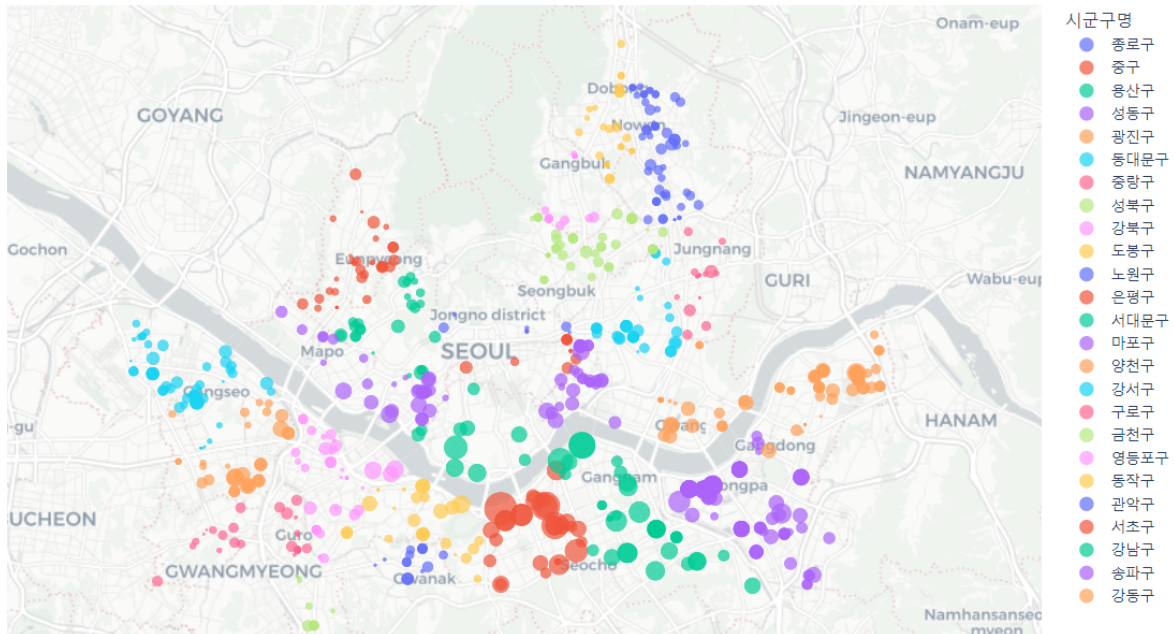
```

```

    template='plotly_white'
)

fig.show()

```



- 강남 3구(서초구, 강남구, 송파구)에서 거래금액의 크기가 높게 나타남.
- 마포구, 용산구, 성동구에서도 큰 거래금액이 분포한다는 것을 알 수 있음.
- 대체적으로 아파트의 높은 시세가 형성된 지역에서 3년간 평균 거래금액이 높게 나타남.

## 3.2 서울시 전체 트리맵

서울시 전체 거래 금액과 거래량을 트리맵을 통하여 시각화하려고 한다.

### 3.2.1 거래금액

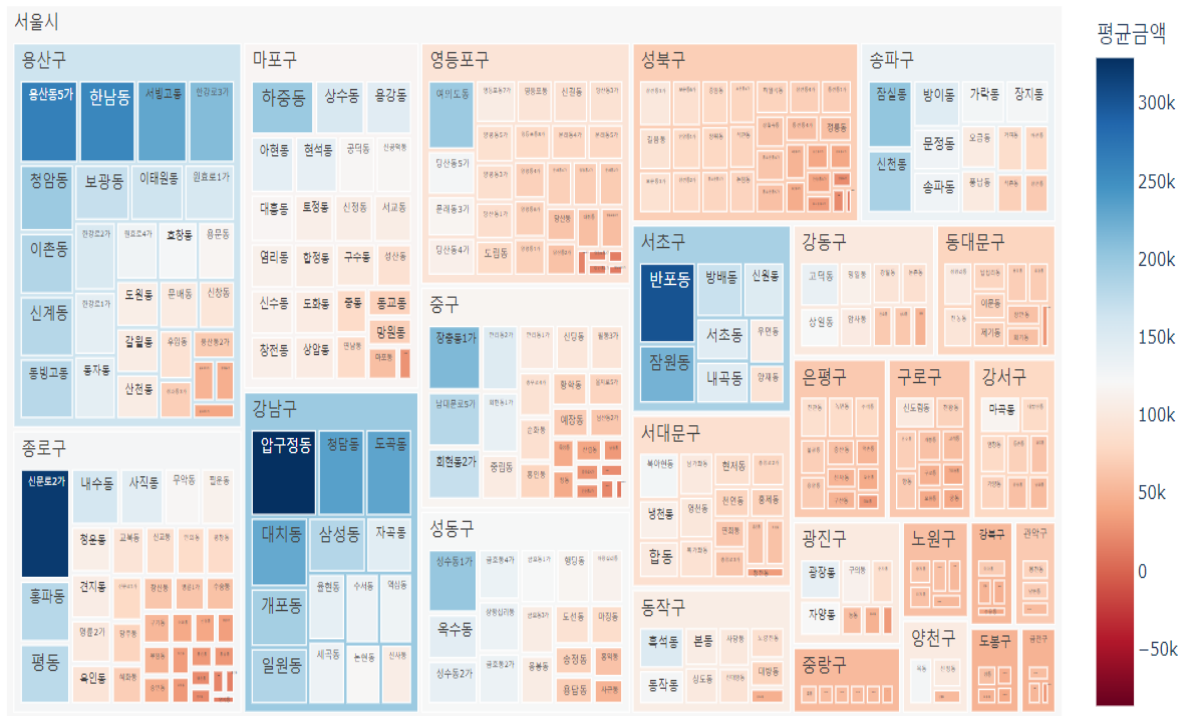
```

fig = px.treemap(AptTrade_Total_1_total, path=[px.Constant("서울시"),
'시군구명', '법정동'], values='평균금액',
                color='평균금액', hover_data=['법정동'],
                color_continuous_scale='RdBu',

                color_continuous_midpoint=np.average(AptTrade_Total_1_total['평균금액'],
                weights=AptTrade_Total_1_total['평균금액']))
fig.update_layout(title= f'서울시 거래금액 비교',
                margin = dict(t=50, l=25, r=25, b=25))
fig.show()

```

## 서울시 거래금액 비교



- 용산구, 서초구, 강남구가 평균 거래 금액이 가장 높은것으로 나타남.
- 용산구의 용산동,한남동 / 강남구의 압구정동 / 서초구의 반포동 / 종로구의 신문로2가 에서 뚜렷하게 높은 평균 거래 금액을 보임.
- 송파구의 경우 강남3구 중 가장 낮은 평균 거래 금액을 나타냄.

### 3.2.2 거래량

```
fig = px.treemap(AptTrade_Total_1_total, path=[px.Constant("서울시"),
'시군구명', '법정동'], values='거래량',
color='거래량', hover_data=['법정동'],
color_continuous_scale='RdBu',

color_continuous_midpoint=np.average(AptTrade_Total_1_total['거래량'],
weights=AptTrade_Total_1_total['거래량']))
fig.update_layout(title= f'서울시 거래량 비교',
margin = dict(t=50, l=25, r=25, b=25))
```

## 서울시 거래량 비교



- 2020~2022년 3년 동안 성북구, 중구가 다른 구들에 비해 확연하게 높은 거래량을 나타냄.
- 성북구의 정릉동, 중구의 신당동의 거래량이 뚜렷하게 높게 나타남.

## 3.3 서울시 각 구별 거래량 및 거래금액

이제 서울시 각 구별 거래량과 거래금액을 시각화하여 알아보려고 한다.

```
AptTrade_Total_2 =
AptTrade_Total[['거래금액', '시군구명']].groupby(['시군구명']).describe()
AptTrade_Total_2 = AptTrade_Total_2.reset_index()
AptTrade_Total_2_total = pd.concat([AptTrade_Total_2[['시군구명']],
                                   AptTrade_Total_2[['거래금액']],
                                   axis = 1) # 컬럼별로 붙힐때 axis = 1

AptTrade_Total_2_total.columns =
['시군구명', '거래량', '평균금액', '표준편차', '최소값', '1분위수', '중앙값', '3분
위수', '최대값']

AptTrade_Total_2_total = AptTrade_Total_2_total.sort_values('중앙값')

# Create figure with secondary y-axis
fig = make_subplots(specs=[[{"secondary_y": True}]])
```

```

fig.add_trace(
    go.Bar(
        name = '거래량',
        x = AptTrade_Total_2_total['시군구명'],
        y = AptTrade_Total_2_total['거래량']
    )
)

fig.add_trace(
    go.Scatter(
        name = '평균금액',
        x = AptTrade_Total_2_total['시군구명'],
        y = AptTrade_Total_2_total['평균금액'],
        yaxis="y2"
    )
)

fig.update_layout(
    title= f'서울시 거래량 및 거래금액 (단위: 만원)',
    title_font_family="맑은고딕",
    title_font_size = 18,
    hoverlabel=dict(
        bgcolor='white',
        font_size=15,
    ),
    hovermode="x unified",
    template='plotly_white',
    xaxis_tickangle=90,
    yaxis_tickformat = ',',
    legend = dict(orientation = 'h', xanchor = "center", x = 0.85, y=
1.1), #Adjust legend position
    barmode='group'
)

```

## 단위(만원)



- 성북구의 경우 길음뉴타운 등의 많은 아파트 단지들이 분포하고 있고, 영등포의 경우 신길뉴타운 조성으로 인한 아파트 공급의 원인으로 가장 많은 거래량을 나타냄.
- 거래금액의 평균은 전통적으로 고가 아파트 단지들이 많이 분포하고 있는 강남구, 서초구, 용산구에서 높게 나타남.
- 금천구의 경우 아파트 단지 수가 타 구에 비해 상대적으로 적어 낮은 거래량을 나타낸 것으로 보임.
- 도봉구의 경우 서울 도심과의 접근성 문제로 매매가가 타 구에 비해 낮게 형성되어 있어 평균 거래금액이 낮게 나타나고 있음.

### 3.4 각 구의 동별 거래량 및 거래금액 (대표 3개구)

서울시 **25개** 자치구를 모두 제시하기에는 분량이 과다해질 우려가 있어 임의로 대표 **3개구** (강남구, 영등포구, 성북구)를 선정하였다. 해당 **3개구** 안에 포함되어 있는 동별 거래량 및 거래금액을 그래프로 나타내보자.

```
AptTrade_Total_1 = AptTrade_Total[['거래금액', '시군구명',
'법정동']].groupby(['시군구명', '법정동']).describe()
AptTrade_Total_1 = AptTrade_Total_1.reset_index()
AptTrade_Total_1_total =
pd.concat([AptTrade_Total_1[['시군구명', '법정동']],
AptTrade_Total_1['거래금액']],
```



```

axis = 1) # 컬럼별로 볼힐때 axis = 1
AptTrade_Total_1_total.columns =
['시군구명', '법정동', '거래량', '평균금액', '표준편차', '최소값', '1분위수', '중앙
값', '3분위수', '최대값']
AptTrade_Total_1_total = AptTrade_Total_1_total.sort_values('거래량')

```

### 3.4.1 강남구

```

emd_area = '강남구'
df1 = AptTrade_Total_1_total[AptTrade_Total_1_total['시군구명'] ==
emd_area]

# Create figure with secondary y-axis
fig = make_subplots(specs=[[{"secondary_y": True}]])

fig.add_trace(
    go.Bar(
        name = '거래량',
        x = df1['법정동'],
        y = df1['거래량']
    )
)

fig.add_trace(
    go.Scatter(
        name = '평균금액',
        x = df1['법정동'],
        y = df1['평균금액'],
        yaxis="y2"
    )
)

fig.update_layout(
    title= f'{emd_area} 거래량 및 거래금액 <br><sup>단위(만원)</sup>',
    title_font_family="맑은고딕",
    title_font_size = 18,
    hoverlabel=dict(
        bgcolor='white',
        font_size=15,
    ),
    hovermode="x unified",
    template='plotly_white',

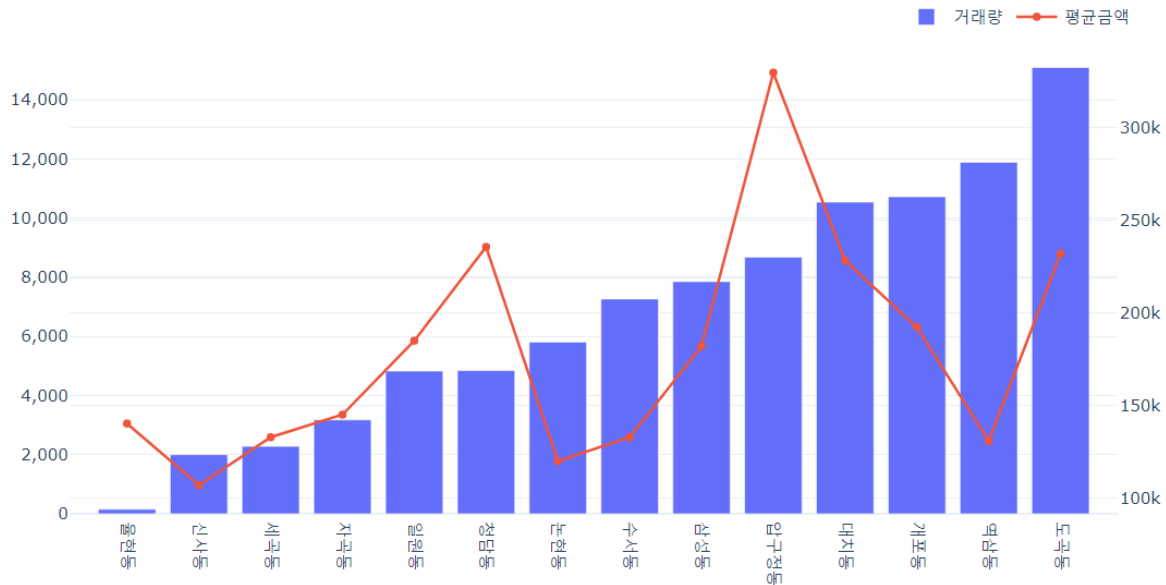
```

```

axis_tickangle=90,
axis_tickformat = ',',
legend = dict(orientation = 'h', xanchor = "center", x = 0.85, y=
1.1), #Adjust legend position
barmode='group'
)

```

강남구 거래량 및 거래금액  
단위(만원)



- 거래량 높은 지역: 도곡동 > 역삼동 > 개포동
- 거래금액 평균 높은 지역: 압구정동 > 청담동 > 도곡동
- 거래량 낮은 지역: 율현동 > 신사동 > 세곡동
- 거래금액 평균 낮은 지역: 신사동 > 세곡동 > 수서동
- 도곡동과 역삼동은 도곡렉슬, 역삼 푸르지오 등의 대단지 아파트가 많이 분포되어 있으며 학군으로 인한 많은 수요로 인해 높은 거래량을 나타내는것으로 보임.
- 거래금액의 평균은 전통적으로 고가 아파트 단지들이 많이 분포하고 있는 압구정동에서 가장 높게 나타남.
- 율현동의 경우 주택 및 빌라가 상대적으로 많고 아파트 매물은 적어 낮은 거래량을 보이고 있음.
- 신사동에는 아파트 매물이 희소하여 거래량이 적으며, 상권으로 인한 주거환경의 한계로 인해 타 동에 비해 낮은 시세를 형성하고 있으므로 낮은 평균 거래금액을 나타냄.

### 3.4.2 영등포구

```

emd_area = '영등포구'
df1 = AptTrade_Total_1_total[AptTrade_Total_1_total['시군구명'] ==
emd_area]

# Create figure with secondary y-axis
fig = make_subplots(specs=[[{"secondary_y": True}]])

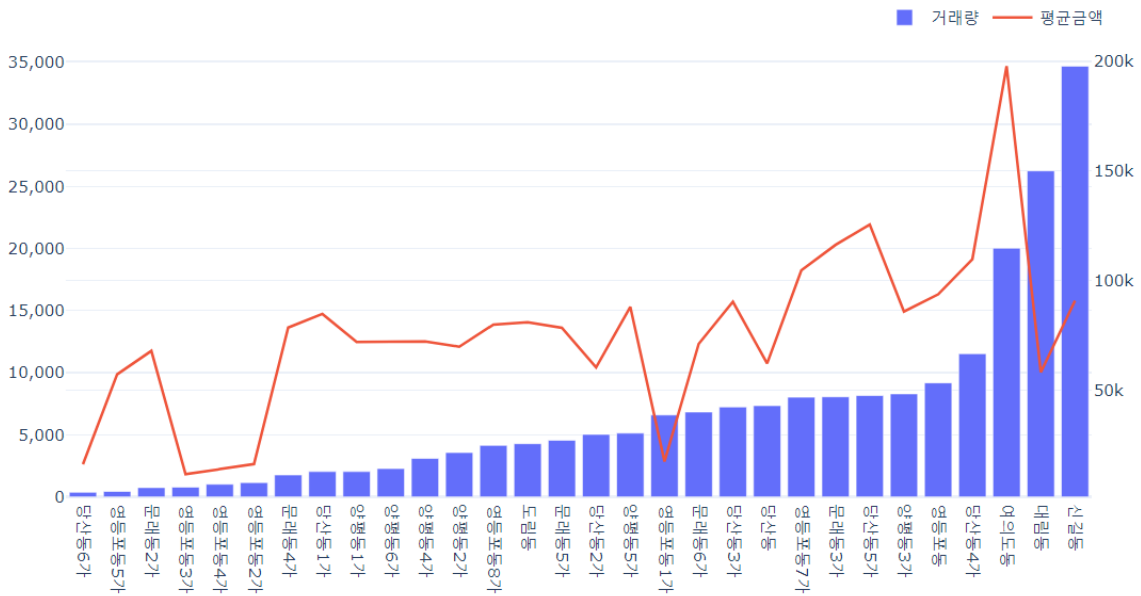
fig.add_trace(
    go.Bar(
        name = '거래량',
        x = df1['법정동'],
        y = df1['거래량']
    )
)

fig.add_trace(
    go.Scatter(
        name = '평균금액',
        x = df1['법정동'],
        y = df1['평균금액'],
        yaxis="y2"
    )
)

fig.update_layout(
    title= f'{emd_area} 거래량 및 거래금액 <br><sup>단위(만원)</sup>',
    title_font_family="맑은고딕",
    title_font_size = 18,
    hoverlabel=dict(
        bgcolor='white',
        font_size=15,
    ),
    hovermode="x unified",
    template='plotly_white',
    xaxis_tickangle=90,
    yaxis_tickformat = ',',
    legend = dict(orientation = 'h', xanchor = "center", x = 0.85, y=
1.1), #Adjust legend position
    barmode='group'
)

```

영등포구 거래량 및 거래금액  
단위(만원)



- 거래량 높은 지역: 신길동 > 대림동 > 여의도동
- 거래금액 평균 높은 지역: 여의도동 > 당산동5가 > 문래동3가
- 거래량 낮은 지역: 당산동6가 > 영등포동5가 > 영등포동3가
- 거래금액 평균 낮은 지역: 영등포동3가 > 영등포동4가 > 영등포동2가
- 신길뉴타운의 공급으로 인해 신길동에서 높은 거래량을 나타내는것으로 보임.
- 높은 시세가 형성되어 있는 여의도동에서 가장 높은 거래금액을 나타냄.
- 당산동6가는 아파트 단지보다 빌라가 상대적으로 훨씬 더 많이 분포한 지역으로 낮은 아파트 거래량을 형성한 것으로 보임.
- 주로 상권이 형성되어 있는 지역으로 아파트 매물은 희소하며 타 동에 비해 낮은 시세가 형성되어 있어 영등포동2~4가에서 낮은 거래량과 낮은 거래금액을 형성하고 있음.

### 3.4.3 성북구

```
emd_area = '성북구'
df1 = AptTrade_Total_1_total[AptTrade_Total_1_total['시군구명'] ==
emd_area]

# Create figure with secondary y-axis
fig = make_subplots(specs=[[{"secondary_y": True}]])

fig.add_trace(
    go.Bar(
```

```

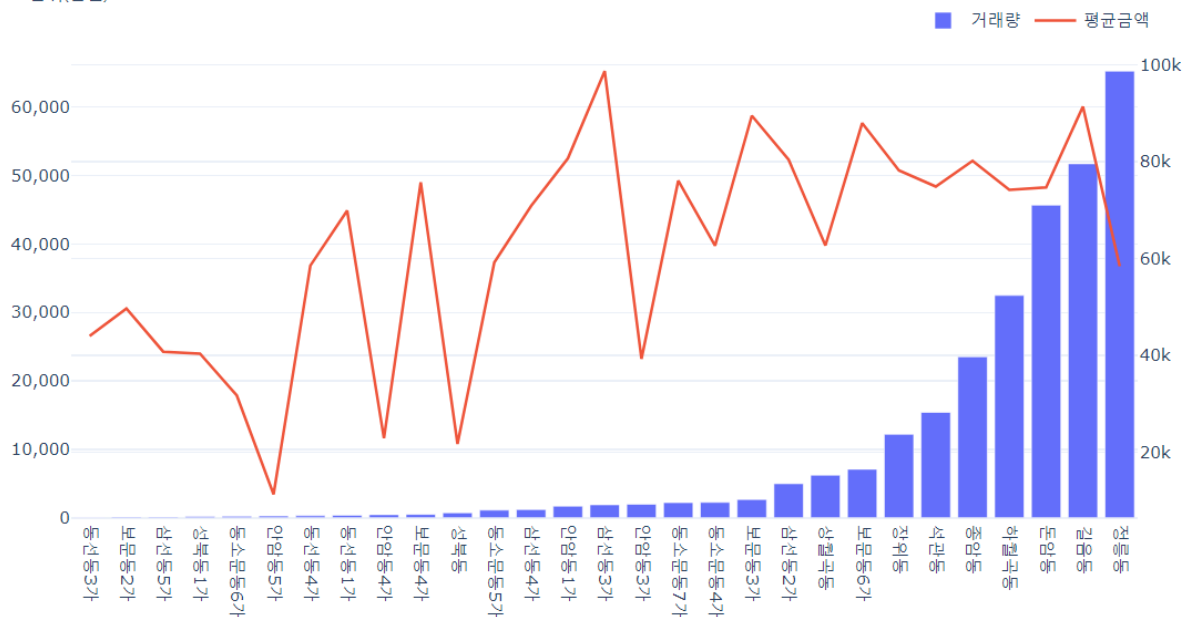
        name = '거래량',
        x = df1['법정동'],
        y = df1['거래량']
    )
)

fig.add_trace(
    go.Scatter(
        name = '평균금액',
        x = df1['법정동'],
        y = df1['평균금액'],
        yaxis="y2"
    )
)

fig.update_layout(
    title= f'{emd_area} 거래량 및 거래금액 <br><sup>단위(만원)</sup>',
    title_font_family="맑은고딕",
    title_font_size = 18,
    hoverlabel=dict(
        bgcolor='white',
        font_size=15,
    ),
    hovermode="x unified",
    template='plotly_white',
    xaxis_tickangle=90,
    yaxis_tickformat = ',',
    legend = dict(orientation = 'h', xanchor = "center", x = 0.85, y=
1.1), #Adjust legend position
    barmode='group'
)

```

성북구 거래량 및 거래금액  
단위(만원)



- 거래량 높은 지역: 정릉동 > 길음동 > 돈암동
- 거래금액 평균 높은 지역: 삼선동3가 > 길음동 > 보문동3가
- 거래량 낮은 지역: 동선동3가 > 보문동2가 > 삼선동5가
- 거래금액 평균 낮은 지역: 안암동5가 > 성북동 > 안암동4가
- 길음 뉴타운 등의 수 많은 아파트 단지들로 매물이 많이 길음동과 함께 매물이 가장 많이 형성되어 있는 정릉동에서 거래량이 가장 높게 나타남.
- 삼선동3가의 경우 '삼선 sk뷰' 아파트의 대형 평형 매물이 가장 높은 평균 금액을 형성하는데 영향을 끼친 것으로 보임.
- 동선동3가는 아파트 매물이 적고, 빌라나 단독주택들로 대부분 형성되어 있는 지역이므로 낮은 아파트 거래량을 보임.
- 안암동5가의 경우 지역의 대다수가 '고려대학교'의 부지이므로 아파트 매물이 희소하여 낮은 거래량과 함께 가장 낮은 평균 거래금액을 나타냄.

### 3.5 서울시 날짜별 거래량 및 거래금액

이제 날짜별(년,월) 거래량 및 평균 거래금액을 살펴보자. 우선 서울시 전체의 날짜별 거래량 및 평균 거래금액 데이터를 그래프로 나타내보았다.

```
AptTrade_Total_3 = AptTrade_Total[['거래금액', '년',
'월']].groupby(['년', '월']).describe()
AptTrade_Total_3 = AptTrade_Total_3.reset_index()
AptTrade_Total_3['월'] =
AptTrade_Total_3["월"].astype('str').str.zfill(2)
```

```

AptTrade_Total_3_total = pd.concat([AptTrade_Total_3[['년', '월']],
                                     AptTrade_Total_3['거래금액']], axis = 1) #
컬럼별로 볼힐때 axis = 1
AptTrade_Total_3_total.columns =
['년', '월', '거래량', '평균금액', '표준편차', '최소값', '1분위수', '중앙값', '3분위
수', '최대값']

AptTrade_Total_3_total['거래날짜'] =
AptTrade_Total_3_total['년'].astype(str) +
AptTrade_Total_3_total['월'].astype(str) + "01"
AptTrade_Total_3_total['거래날짜'] = AptTrade_Total_3_total.apply(lambda
x: pd.to_datetime(x['거래날짜'], format='%Y%m%d'), axis=1)
df1 = AptTrade_Total_3_total

# Create figure with secondary y-axis
fig = make_subplots(specs=[[{"secondary_y": True}]])

fig.add_trace(
    go.Bar(
        name = '거래량',
        x = df1['거래날짜'],
        y = df1['거래량']
    )
)

fig.add_trace(
    go.Scatter(
        name = '평균금액',
        x = df1['거래날짜'],
        y = df1['평균금액'],
        yaxis="y2"
    )
)

fig.update_layout(
    title= f'서울시 날짜별 거래량 및 거래금액 <br><sup>단위(만원)</sup> ',
    title_font_family="맑은고딕",
    title_font_size = 18,
    hoverlabel=dict(
        bgcolor='white',
        font_size=15,
    ),
    hovermode="x unified",
    template='plotly_white',
    xaxis_tickangle=90,

```

```

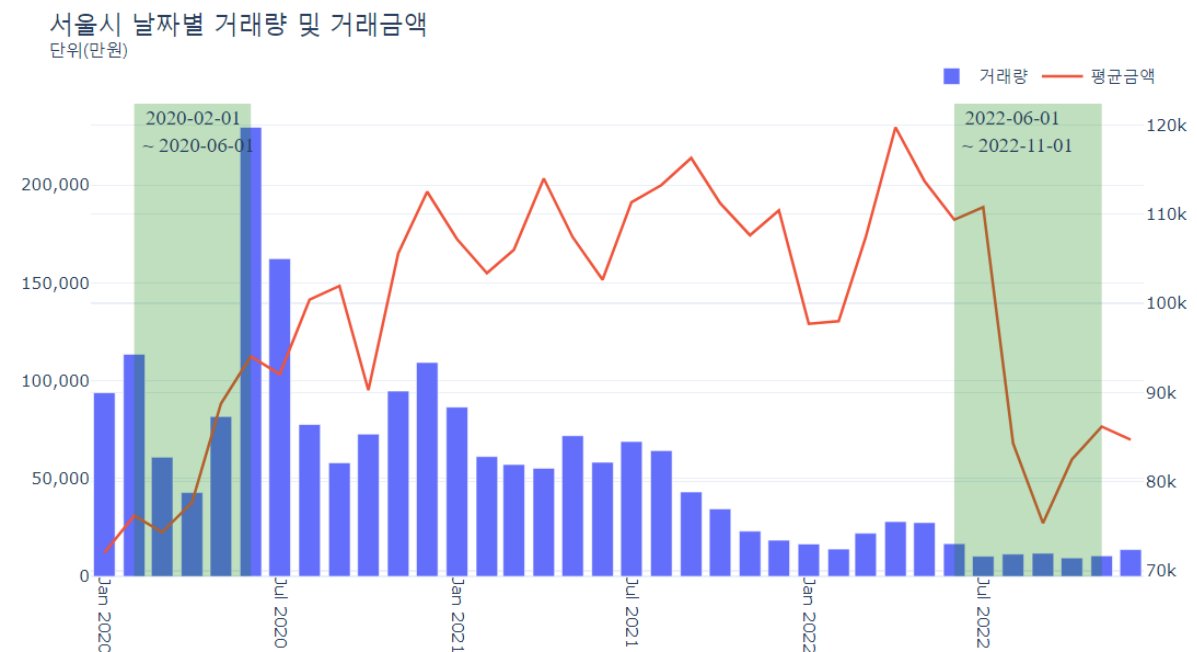
axis_tickformat = ',',
legend = dict(orientation = 'h', xanchor = "center", x = 0.85, y=
1.1), #Adjust legend position
barmode='group'
)

fig.add_vrect(x0="2022-06-01", x1="2022-10-01",
              annotation_text="2022-06 <br> ~ 2022-10",
              annotation_position="top left",
              annotation=dict(font_size=15, font_family="Times New
Roman"),
              fillcolor="green", opacity=0.25, line_width=0)

fig.add_vrect(x0="2020-02-01", x1="2020-06-01",
              annotation_text="2020-02 <br> ~ 2020-06",
              annotation_position="top left",
              annotation=dict(font_size=15, font_family="Times New
Roman"),
              fillcolor="green", opacity=0.25, line_width=0)

fig.show()

```



- 2020년 6월에 가장 많은 거래량을 나타냄.
- 2022년 4월에 평균 거래금액의 최고점을 찍음.
- 2022년 7월 이후 평균 거래금액이 급락함.
- 2022년 9월 이후 급매물 소화 후 일시적인 반등 양상을 나타냄.
- 2020년 중순 이후로 전반적인 거래량 감소세를 나타냄.



### 3.6 서울시 구별 날짜별 거래량 및 거래금액

앞서 살펴보았던 대표 3개구 (강남구, 영등포구, 성북구) 의 날짜별 거래량 및 평균 거래금액을 알아보자.

#### 3.6.1 강남구

```
sig_area = '강남구'
df1 = AptTrade_Total_3_total[AptTrade_Total_3_total['시군구명'] ==
sig_area]

# Create figure with secondary y-axis
fig = make_subplots(specs=[[{"secondary_y": True}]]))

fig.add_trace(
    go.Bar(
        name = '거래량',
        x = df1['거래날짜'],
        y = df1['거래량']
    )
)

fig.add_trace(
    go.Scatter(
        name = '평균금액',
        x = df1['거래날짜'],
        y = df1['평균금액'],
        yaxis="y2"
    )
)

fig.update_layout(
    title= f'{sig_area} 거래량 및 거래금액 (단위: 만원)',
    title_font_family="맑은고딕",
    title_font_size = 18,
    hoverlabel=dict(
        bgcolor='white',
        font_size=15,
    ),
    hovermode="x unified",
    template='plotly_white',
```

```

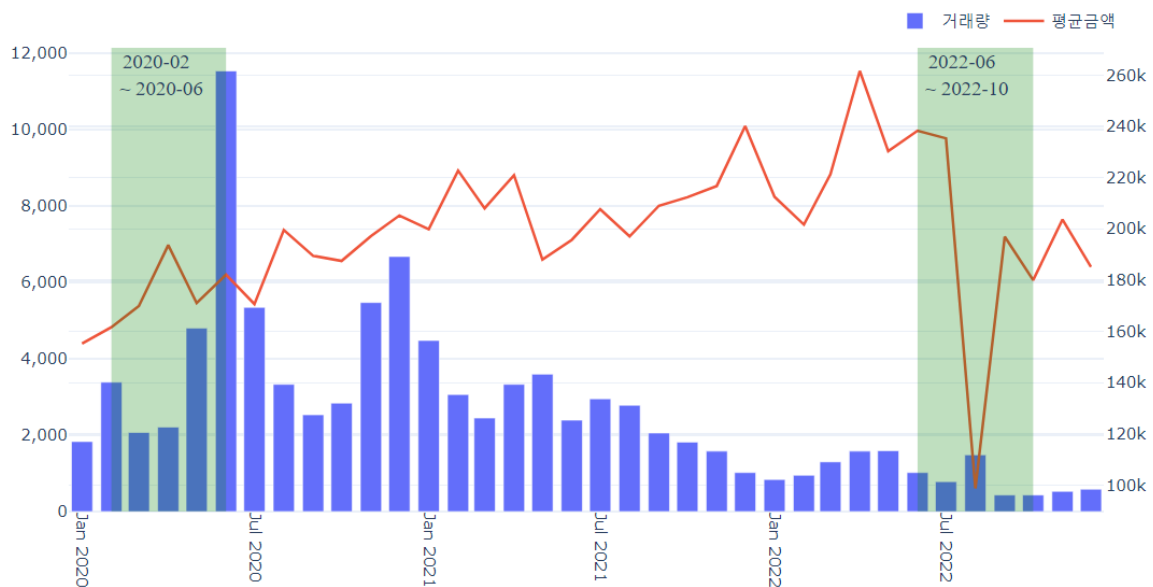
axis_tickangle=90,
axis_tickformat = ',',
legend = dict(orientation = 'h', xanchor = "center", x = 0.85, y=
1.1), #Adjust legend position
barmode='group'
)

fig.add_vrect(x0="2022-06-01", x1="2022-10-01",
              annotation_text="2022-06 <br> ~ 2022-10",
              annotation_position="top left",
              annotation=dict(font_size=15, font_family="Times New
Roman"),
              fillcolor="green", opacity=0.25, line_width=0)

fig.add_vrect(x0="2020-02-01", x1="2020-06-01",
              annotation_text="2020-02 <br> ~ 2020-06",
              annotation_position="top left",
              annotation=dict(font_size=15, font_family="Times New
Roman"),
              fillcolor="green", opacity=0.25, line_width=0)

```

강남구 거래량 및 거래금액 (단위: 만원)



- 2020년 6월에 가장 많은 거래량을 나타냄.
- 2022년 4월에 평균 거래금액의 최고점을 찍음.
- 2022년 7월 이후 평균 거래금액이 급락함.
- 2020년 중순 이후로 전반적인 거래량 감소세를 나타냄.
- 2022년 9월 이후 급매물 소화 후 일시적인 반등 양상을 나타냄.
- 서울시 전체와 상당히 유사한 양상을 나타내나 22년 9월 이후의 반등폭은 서울시 전체 보다 훨씬 높게 나타남.

### 3.6.2 영등포구

```
sig_area = '영등포구'
df1 = AptTrade_Total_3_total[AptTrade_Total_3_total['시군구명'] ==
sig_area]

# Create figure with secondary y-axis
fig = make_subplots(specs=[[{"secondary_y": True}]])

fig.add_trace(
    go.Bar(
        name = '거래량',
        x = df1['거래날짜'],
        y = df1['거래량']
    )
)

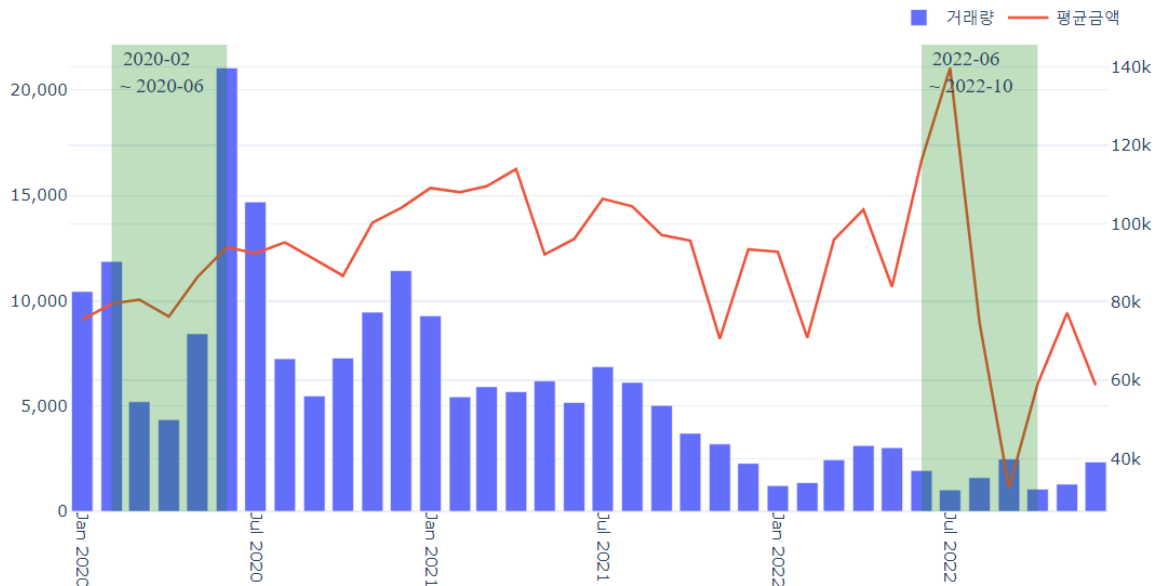
fig.add_trace(
    go.Scatter(
        name = '평균금액',
        x = df1['거래날짜'],
        y = df1['평균금액'],
        yaxis="y2"
    )
)

fig.update_layout(
    title= f'{sig_area} 거래량 및 거래금액 (단위: 만원)',
    title_font_family="맑은고딕",
    title_font_size = 18,
    hoverlabel=dict(
        bgcolor='white',
        font_size=15,
    ),
    hovermode="x unified",
    template='plotly_white',
    xaxis_tickangle=90,
    yaxis_tickformat = ',',
    legend = dict(orientation = 'h', xanchor = "center", x = 0.85, y=
1.1), #Adjust legend position
    barmode='group'
)
```

```
fig.add_vrect(x0="2022-06-01", x1="2022-10-01",
              annotation_text="2022-06 <br> ~ 2022-10",
              annotation_position="top left",
              annotation=dict(font_size=15, font_family="Times New
Roman"),
              fillcolor="green", opacity=0.25, line_width=0)

fig.add_vrect(x0="2020-02-01", x1="2020-06-01",
              annotation_text="2020-02 <br> ~ 2020-06",
              annotation_position="top left",
              annotation=dict(font_size=15, font_family="Times New
Roman"),
              fillcolor="green", opacity=0.25, line_width=0)
```

영등포구 거래량 및 거래금액 (단위: 만원)



- 2020년 6월에 가장 많은 거래량을 나타냄.
- 2022년 7월에 평균 거래금액의 최고점을 찍음.
- 2022년 7월 이후 평균 거래금액이 급락함.
- 거래량은 2020년 중순 이후로 전반적으로 감소세를 나타냄.
- 2022년 9월 이후 급매물 소화 후 일시적인 반등 양상을 나타냄.
- 영등포구의 경우 강남구와는 다르게 21년에 좀 더 깊은 가격 조정을 겪은 후 22년 중반에 급등하는 양상을 나타냄.
- 영등포구는 평균 거래금액의 최고점이 강남구나 서울시 전체보다 한두달 늦게 형성됨.
- 22년 7월 이후에는 서울시 전체와 강남구 보다 훨씬 더 큰 급락을 보임.

### 3.6.3 성북구

```
sig_area = '성북구'
df1 = AptTrade_Total_3_total[AptTrade_Total_3_total['시군구명'] ==
sig_area]

# Create figure with secondary y-axis
fig = make_subplots(specs=[[{"secondary_y": True}]])

fig.add_trace(
    go.Bar(
        name = '거래량',
        x = df1['거래날짜'],
        y = df1['거래량']
    )
)

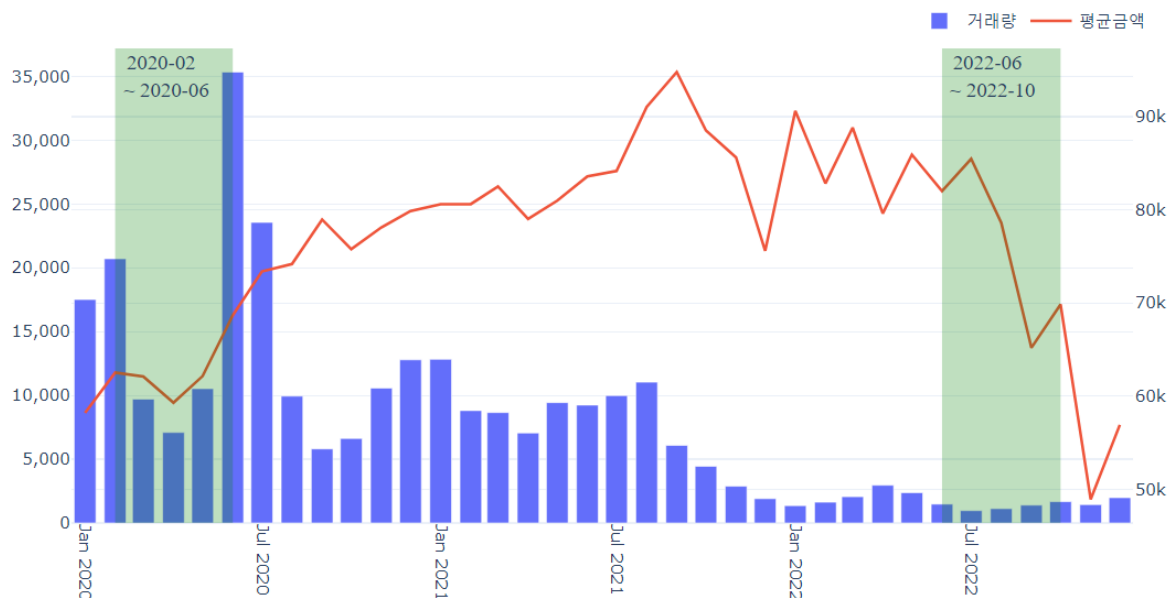
fig.add_trace(
    go.Scatter(
        name = '평균금액',
        x = df1['거래날짜'],
        y = df1['평균금액'],
        yaxis="y2"
    )
)

fig.update_layout(
    title= f'{sig_area} 거래량 및 거래금액 (단위: 만원)',
    title_font_family="맑은고딕",
    title_font_size = 18,
    hoverlabel=dict(
        bgcolor='white',
        font_size=15,
    ),
    hovermode="x unified",
    template='plotly_white',
    xaxis_tickangle=90,
    yaxis_tickformat = ',',
    legend = dict(orientation = 'h', xanchor = "center", x = 0.85, y=
1.1), #Adjust legend position
    barmode='group'
)
```

```
fig.add_vrect(x0="2022-06-01", x1="2022-10-01",
              annotation_text="2022-06 <br> ~ 2022-10",
              annotation_position="top left",
              annotation=dict(font_size=15, font_family="Times New Roman"),
              fillcolor="green", opacity=0.25, line_width=0)

fig.add_vrect(x0="2020-02-01", x1="2020-06-01",
              annotation_text="2020-02 <br> ~ 2020-06",
              annotation_position="top left",
              annotation=dict(font_size=15, font_family="Times New Roman"),
              fillcolor="green", opacity=0.25, line_width=0)
```

성북구 거래량 및 거래금액 (단위: 만원)



- 2020년 6월에 가장 많은 거래량을 나타냄.
- 2021년 9월에 평균 거래금액의 최고점을 찍음.
- 2022년 7월 이후 평균 거래금액이 급락함.
- 거래량은 2020년 중순 이후로 전반적으로 감소세를 나타냄.
- 2022년 9월 이후 소폭 반등하다 다시 하락 양상을 나타냄.
- 성북구의 경우 평균 거래금액의 최고점이 서울시 전체나 타 구보다 1년정도 앞서 형성됨.
- 22년 7월 이후 반등폭이 적은 강한 하락 추세를 보임.

### 3.7 각 구의 동별, 날짜별 거래량 및 거래금액

앞서 살펴본 3개 구에서 각 구에서 거래량이 가장 높았던 대표 1개동씩을 선정하여 각 동의 날짜별 거래량 및 평균 거래금액을 알아보려고 한다. 강남구에서는 도곡동, 영등포구에서는 신길동, 성북구에서는 정릉동을 선정하였다.

#### 3.7.1 강남구 도곡동

```
sig_area = '강남구'
emd_area = '도곡동'

df2 = AptTrade_Total_4_total[AptTrade_Total_4_total['시군구명'] ==
sig_area]

emd_list = df2['법정동'].unique()

df1 = df2[df2['법정동'] == emd_area]

# Create figure with secondary y-axis
fig = make_subplots(specs=[[{"secondary_y": True}]])

fig.add_trace(
    go.Bar(
        name = '거래량',
        x = df1['거래날짜'],
        y = df1['거래량']
    )
)

fig.add_trace(
    go.Scatter(
        name = '평균금액',
        x = df1['거래날짜'],
        y = df1['평균금액'],
        yaxis="y2"
    )
)

fig.update_layout(
    title= f'{sig_area} {emd_area} 거래량 및 거래금액  
<br><sup>단위 (만원)</sup>',
    title_font_family="맑은고딕",
```

```

title_font_size = 18,
hoverlabel=dict(
    bgcolor='white',
    font_size=15,
),
hovermode="x unified",
template='plotly_white',
xaxis_tickangle=90,
yaxis_tickformat = ',',
legend = dict(orientation = 'h', xanchor = "center", x = 0.85, y=
1.1), #Adjust legend position
barmode='group'
)

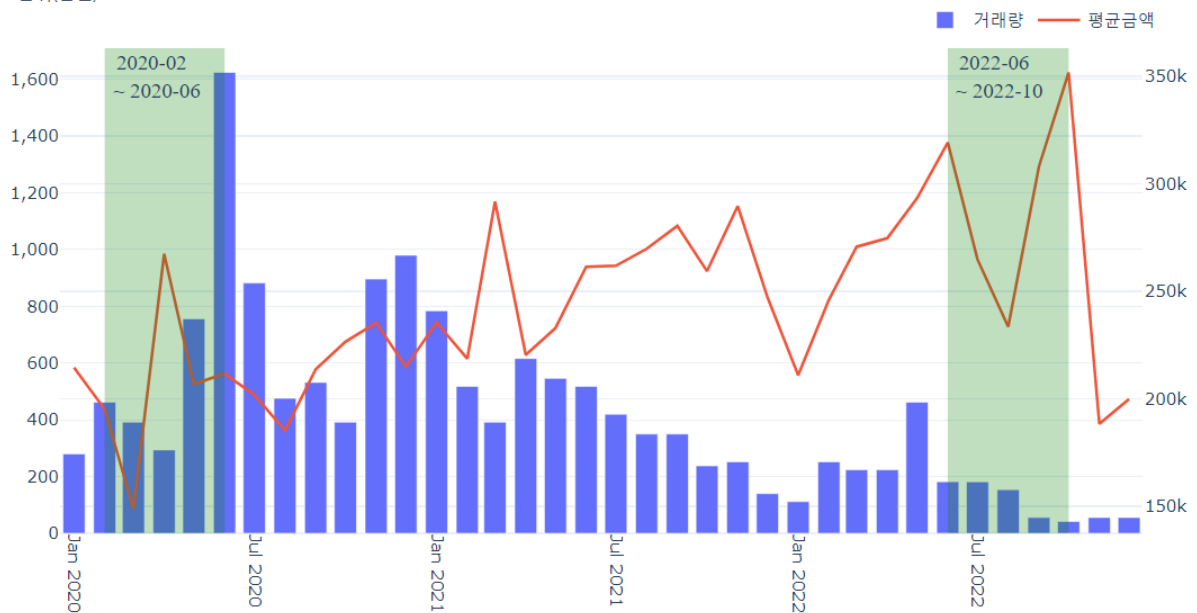
fig.add_vrect(x0="2022-06-01", x1="2022-10-01",
              annotation_text="2022-06 <br> ~ 2022-10",
              annotation_position="top left",
              annotation=dict(font_size=15, font_family="Times New
Roman"),
              fillcolor="green", opacity=0.25, line_width=0)

fig.add_vrect(x0="2020-02-01", x1="2020-06-01",
              annotation_text="2020-02 <br> ~ 2020-06",
              annotation_position="top left",
              annotation=dict(font_size=15, font_family="Times New
Roman"),
              fillcolor="green", opacity=0.25, line_width=0)

```



강남구 도곡동 거래량 및 거래금액  
단위(만원)



- 2020년 6월에 가장 많은 거래량을 나타냄.
- 2022년 10월에 평균 거래금액의 최고점을 찍음.
- 2022년 10월 이후 평균 거래금액이 급락함.
- 거래량은 2020년 말 이후로 전반적으로 감소세를 나타냄.
- 2022년 8월 이후 큰 반등 상승을 보이고 이후 급락함.
- 도곡동의 경우 평균 거래금액의 타 지역 대비 최고점이 낮은 시기에 형성됨.
- 타 지역의 하락 국면 전환 시기에 도곡동은 일시적으로 더 큰 상승을 보임.

### 3.7.2 영등포구 신길동

```
sig_area = '영등포구'
emd_area = '신길동'

df2 = AptTrade_Total_4_total[AptTrade_Total_4_total['시군구명'] ==
sig_area]

emd_list = df2['법정동'].unique()

df1 = df2[df2['법정동'] == emd_area]

# Create figure with secondary y-axis
```

```

fig = make_subplots(specs=[[{"secondary_y": True}]])

fig.add_trace(
    go.Bar(
        name = '거래량',
        x = df1['거래날짜'],
        y = df1['거래량']
    )
)

fig.add_trace(
    go.Scatter(
        name = '평균금액',
        x = df1['거래날짜'],
        y = df1['평균금액'],
        yaxis="y2"
    )
)

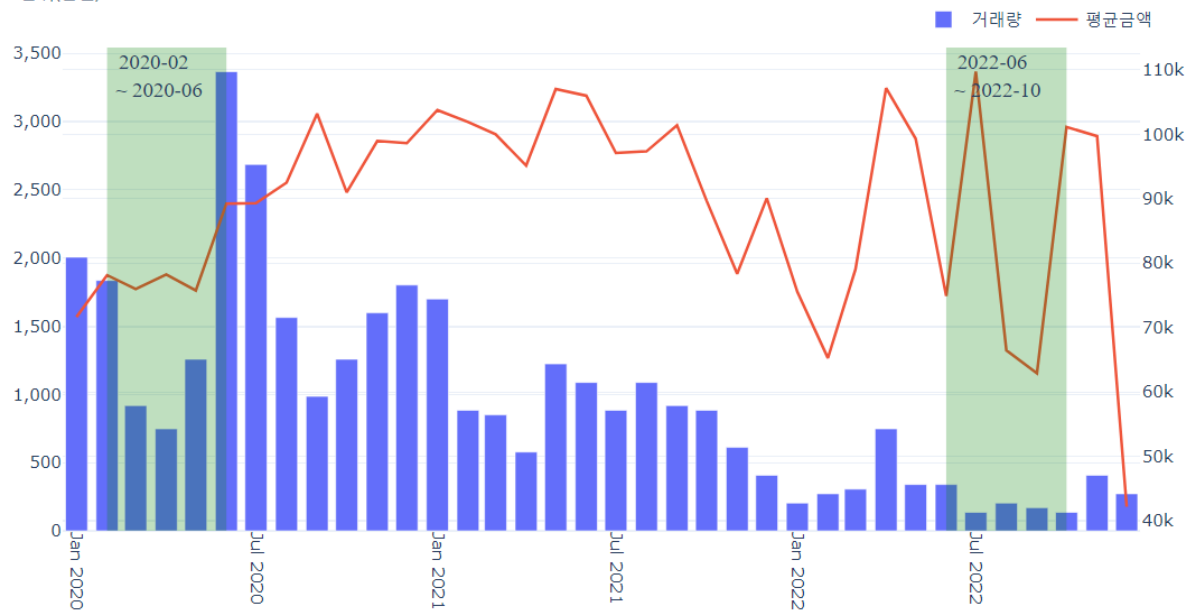
fig.update_layout(
    title= f'{{sig_area}} {{emd_area}} 거래량 및 거래금액  
<br><sup>단위(만원)</sup>',
    title_font_family="맑은고딕",
    title_font_size = 18,
    hoverlabel=dict(
        bgcolor='white',
        font_size=15,
    ),
    hovermode="x unified",
    template='plotly_white',
    xaxis_tickangle=90,
    yaxis_tickformat = ',',
    legend = dict(orientation = 'h', xanchor = "center", x = 0.85, y=
1.1), #Adjust legend position
    barmode='group'
)

fig.add_vrect(x0="2022-06-01", x1="2022-10-01",
              annotation_text="2022-06 <br> ~ 2022-10",
              annotation_position="top left",
              annotation=dict(font_size=15, font_family="Times New
Roman"),
              fillcolor="green", opacity=0.25, line_width=0)

```

```
fig.add_vrect(x0="2020-02-01", x1="2020-06-01",
              annotation_text="2020-02 <br> ~ 2020-06",
              annotation_position="top left",
              annotation=dict(font_size=15, font_family="Times New
                              Roman"),
              fillcolor="green", opacity=0.25, line_width=0)
```

영등포구 신길동 거래량 및 거래금액  
단위(만원)



- 2020년 6월에 가장 많은 거래량을 나타냄.
- 2022년 7월에 평균 거래금액의 최고점을 찍음.
- 2022년 10월 이후 평균 거래금액이 급락함.
- 거래량은 2020년 말 이후로 전반적으로 감소세를 나타냄.
- 신길동의 경우 2022년 7월 가격 하락 이후 전반적인 반등시기에 상당히 뚜렷한 반등을 나타냄.
- 2022년 10월의 하락 기점 이후로 반등을 보이지 못하고 있음.

### 3.7.3 성북구 정릉동

```
sig_area = '성북구'
emd_area = '정릉동'
```

```
df2 = AptTrade_Total_4_total[AptTrade_Total_4_total['시군구명'] ==
                              sig_area]
```

```

emd_list = df2['법정동'].unique()

df1 = df2[df2['법정동'] == emd_area]

# Create figure with secondary y-axis
fig = make_subplots(specs=[[{"secondary_y": True}]])

fig.add_trace(
    go.Bar(
        name = '거래량',
        x = df1['거래날짜'],
        y = df1['거래량']
    )
)

fig.add_trace(
    go.Scatter(
        name = '평균금액',
        x = df1['거래날짜'],
        y = df1['평균금액'],
        yaxis="y2"
    )
)

fig.update_layout(
    title= f'{sig_area} {emd_area} 거래량 및 거래금액  
<br><sup>단위(만원)</sup>',
    title_font_family="맑은고딕",
    title_font_size = 18,
    hoverlabel=dict(
        bgcolor='white',
        font_size=15,
    ),
    hovermode="x unified",
    template='plotly_white',
    xaxis_tickangle=90,
    yaxis_tickformat = ',',
    legend = dict(orientation = 'h', xanchor = "center", x = 0.85, y=
1.1), #Adjust legend position
    barmode='group'
)

fig.add_vrect(x0="2022-06-01", x1="2022-10-01",
              annotation_text="2022-06 <br> ~ 2022-10",

```

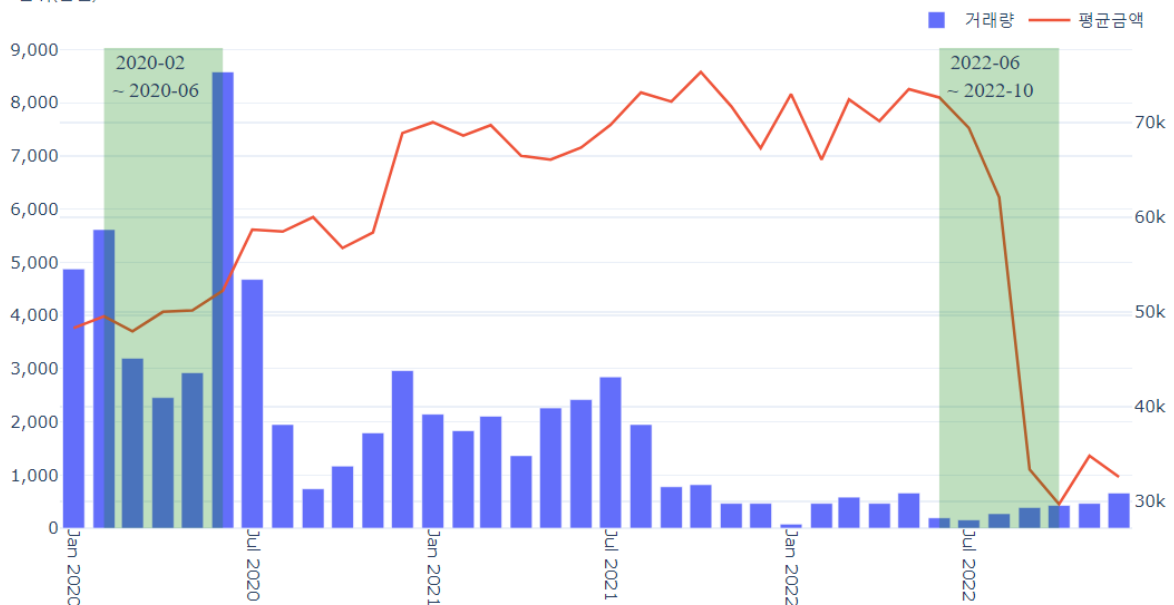
```

annotation_position="top left",
                annotation=dict(font_size=15, font_family="Times New
Roman"),
                fillcolor="green", opacity=0.25, line_width=0)

fig.add_vrect(x0="2020-02-01", x1="2020-06-01",
                annotation_text="2020-02 <br> ~ 2020-06",
                annotation_position="top left",
                annotation=dict(font_size=15, font_family="Times New
Roman"),
                fillcolor="green", opacity=0.25, line_width=0)

```

성북구 정릉동 거래량 및 거래금액  
단위(만원)



- 2020년 6월에 가장 많은 거래량을 나타냄.
- 2021년 10월에 평균 거래금액의 최고점을 찍음.
- 2022년 6월 이후 평균 거래금액이 급락함.
- 거래량은 2020년 중순 이후로 뚜렷한 감소세를 나타냄.
- 정릉동의 경우 타 지역에 비해 오랜 기간 동안 고점 부근에서 평균 금액이 형성되어 있다가, 2022년 6월 이후 급락을 나타냄.
- 2022년 10월 이후 거래 금액에 작은 반등을 보였으나 미미해 보임.
- 정릉동의 경우 2020년 중반 이후의 거래량이 감소가 상당히 뚜렷하며, 2022년의 거래량은 아주 미미한 수준.

## 4. 결과 분석

파이썬, AWS를 활용하여 2020~2022년 3년간의 서울시의 아파트 매매 동향을 알아 보았다. 지난 2020년 부터 2022년 까지 3년간의 장기간 코로나로 인해 대한민국 부동산 시장은ダイナミック한 변화 양상을 나타내었다. 데이터에서 나타나듯이 2020년 부터 시작된 코로나로 인한 저금리 및 양적 완화 정책으로 서울시 아파트의 매매가는 약 2년간 큰 상승을 이루었다. 그러나 과도한 인플레이션과 각종 이벤트에 따른 유동성 회수로 인해 대부분의 서울시 아파트들의 매매가는 22년 중반 이후 급락을 보였다. 그러나 이러한 대표적인 흐름에서 각 구별 또는 동별로 시기와 형태상 차이는 존재했다. 기존의 가격이 높다고 판단 되는 지역들(강남구 등)의 경우 전체적인 급락의 시기가 짧고, 반등이 강하게 온 반면, 타 지역(영등포구, 성북구 등)에서는 전체적인 급락시기 이후에 하락의 추세가 강하고 반등이 미미하게 나타났다. 2022년 7월 이후의 전체적인 급락은 아마 미국 연준의 가파른 금리 인상과 전반적인 금융시장의 하락에 기인한 아파트 매물 적재 및 금매물의 소화로 인한 것으로 추측된다.

그 밖에 지역별로 매물 분포와 상권의 형태가 지역별 아파트 매매 동향에 어떤 영향을 끼치는지 간접적으로 알 수 있었다. 서울 도심에서 멀어질 수록, 아파트 매물이 작을 수록, 중심 상권 붙어 있는 지역일 수록 (주거단지가 거의 없는) 아파트 거래 금액과 거래량이 작게 나타났다. (대표적인 예로 영등포동 2~4가나 도봉구의 아파트들).

데이터를 보면 2022년 초반에 전반적인 서울시 아파트 매매 시장의 큰 상승이 있었는데, 이는 2020년 이후 부터 지속된 부동산 규제와 막판의 대출규제로 인해 뚝뚝한 한채로 투자수요가 대거 몰리면서 서울시 아파트들에 대한 수요가 늘어난 것을 원인으로 추측한다. 또한 이러한 막판 상승 형태는 서울시 내에서도 지역별로 편차를 보인다. 서울시의 외곽지역보다 도심 지역이 조금 더 늦게 거래금액의 최고점을 찍는 양상을 보였으며 거래가 급락에 반발하는 반등폭도 더 높게 나타났다.

서울시 각 구와 각 동의 아파트 매매 동향이 서울시 전체 아파트 매매 동향과 유사하게 흘러가는 것이 전반적인 형태이지만 구 마다 그리고 동 마다 특이점을 갖는 형태를 나타내기도 하였다. 예를 들어 영등포구의 경우 타 거래금액 최고점이 22년 7월에 형성되었는데, 이는 타지역에 비해 늦은 사이클이었다. 아마 이것은 정책의 변화로 인한 영등포구 여의도의 재개발 기대감이 후반부 가격 상승에 반영된것이 아닐까 생각한다. 이 밖에 지역들마다 개별적인 상황을 지니고 있는 이유로 아파트 매매 동향의 경우 전반적인 사이클에서 벗어난 형태를 띄는 경우가 상당히 많이 나타나보인다.

부동산의 데이터를 수집하고 분석하는데 있어서, 단순 수치적인 자료로만으로는 상당히 빈약한 분석을 보일 수가 있다. 따라서 부동산 데이터는 각종 부동산 정책 및 지역별 개발 계획, 거리, 상권 등의 여러 요소가 고려되어야 조금 더 나은 인사이트를 도출해 낼 수 있겠다는 것을 필요성을 느낀다.

## 5. Reference

- KB 부동산 보고서
  - <https://www.kbfg.com/kbresearch/report/reportView.do?reportId=2000196>
- 서울특별시 부동산 월간 동향
  - <https://land.seoul.go.kr:444/land/>
- 부동산 통계정보 R-One
  - <https://www.reb.or.kr/r-one/na/ntt/selectNttList.do?mi=9509&bbsId=1106>
- 월간 데이콘 한국 부동산 데이터 시각화 경진대회
  - <https://dacon.io/competitions/official/235724/leaderboard>

- Plotly
  - <https://plotly.com/python/>