

Covid-19 시국 정부의 재난지원금 제공에 따른 경기도 지역 소상공인, 전통시장 실적과 소비 패턴의 변화

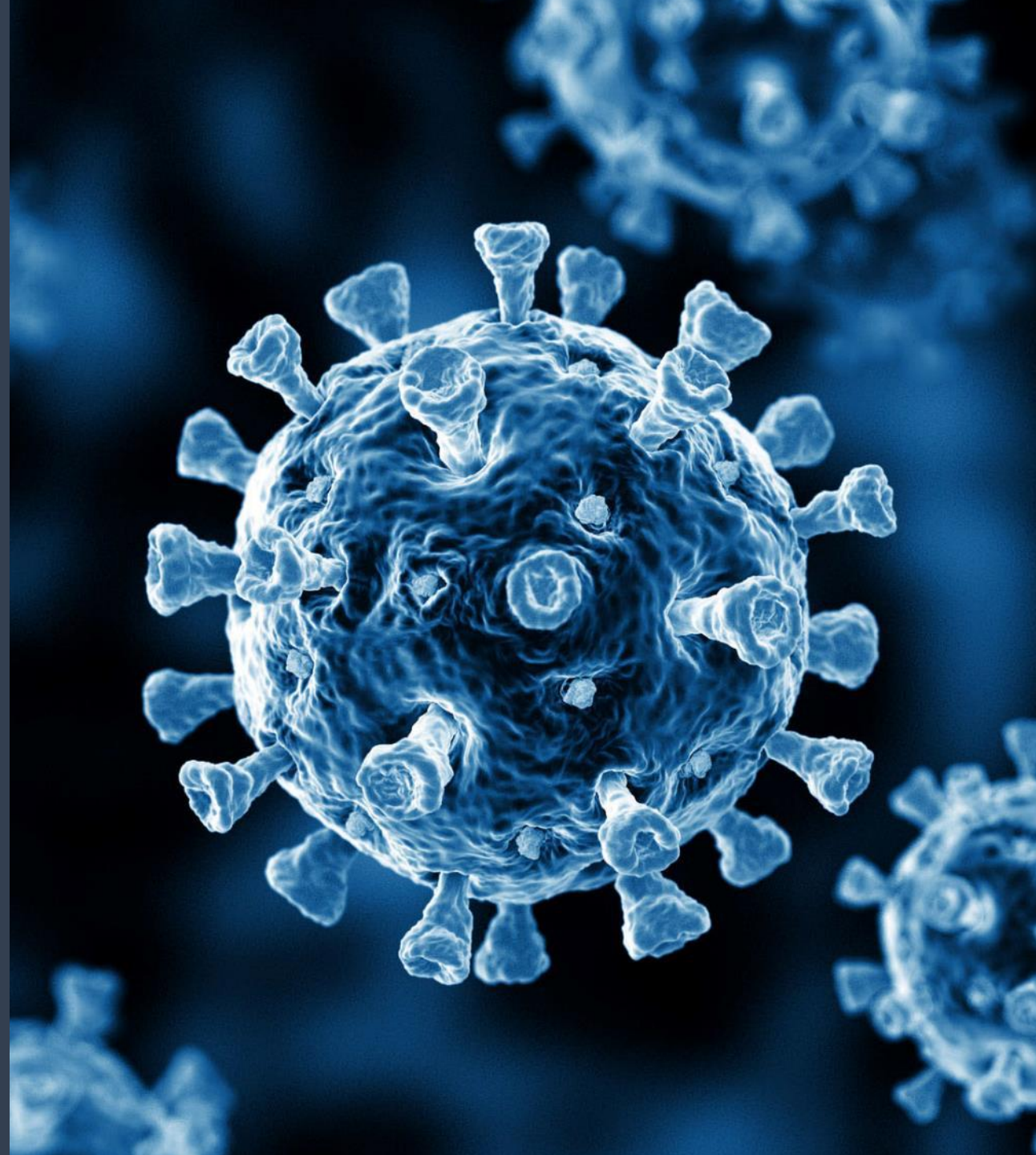
1조

조장 : 권지현

조원 : 강예진, 권용현

CONTENTS

- 1 프로젝트 수행방향
 - 연구배경 및 필요성
 - 연구목적
- 2 프로젝트 구축 메뉴얼
 - 가상환경 구축
 - streamlit
- 3 프로젝트 diagram
- 4 프로젝트 내용(데이터 전처리)
 - 월별 코로나 19 발생 현황
 - 월별 가구당 평균 소비액
 - 지역화폐 발행 및 소비 추이
 - 소상공인 소비추이
 - 전통시장 소비추이
- 5 프로젝트 결과



Part 1 프로젝트 수행 방향



1-1. 연구배경

COVID-19 팬데믹의 글로벌 확산은 경기 침체와 소비 위축 현상을 발생했습니다.
이에 따라 각국은 경제적 어려움을 겪는 국민들을 지원하고
경기 부양을 위해 다양한 정책을 도입했습니다.

그 중에서도 대한민국 정부는 **재난지원금을 지급**하는 방식으로
경제 활성화에 기여하려는 시도를 했습니다.
특히 재난지원금 정책은 경기도 내 소상공인과 전통시장에게 큰 영향을 미치게 되었습니다.
이들은 지역사회에서 중요한 역할을 하며 경제 활동의 중심입니다.
그러나 코로나 팬데믹으로 인해 소비 심리의 위축으로
매출이 급격하게 감소한 상황에서 정부의 재난지원금은
그들에게 큰 의미를 갖을 것으로 예상됩니다.

이러한 배경 아래, 이 연구는 정부의 **재난지원금 정책**이
경기도 소상공인과 전통시장의 경제적 상황과 소비 패턴에
어떤 영향을 미쳤는지를 알아보고자 합니다.

1-2. 연구필요성



재난지원금 및 지역화폐 효과 분석

Covid-19 대응으로 제공된 재난지원금이 경기도 지역 소비에 미친 영향을 분석하여 지역 경제 회복과 소비 증진 효과를 살펴본다.



소상공인 및 전통시장 실적 변화 파악

재난지원금 발행과 관련하여 경기도 지역 소상공인 및 전통시장의 매출 및 이익 변화를 추적하고 분석하여 어떤 영향을 받았는지 확인한다.

1-3. 연구목적

Covid-19 시기에 정부의 재난지원금이 경기도 지역 소비 추이에 미친 영향을
데이터 분석을 통해 탐구하고,
이에 따른 소상공인 및 전통시장의 실적과 패턴의 변화를 분석한다.

Part 2 프로젝트 구축 메뉴얼



2-1. 가상환경 구축하기

가상환경이란?

시스템에 설치된 파이썬과 별도로 다른 디렉토리에 파이썬 인터프리터와 파이썬 라이브러리를 설치한 것을 의미합니다.

Powershell prompt를 들어가면 지금 어떤 환경을 쓰고 있는지 보입니다.

2-1. 가상환경 구축하기

Install 1

```
(base) PS C:\Users\MAIN> conda create -n semi_project python=3.9 ipython  
numpy pandas matplotlib scipy|
```

가상환경을 구축하기 위해서는 위와 같은 코드를 입력해줍니다.
밑줄에는 가상환경 이름을 입력해주시면 됩니다.

Install 2

```
wcwidth      pkgs/main/noarch::wcwidth-0.2.5-pyhd3eb1b0_0  
wheel        pkgs/main/win-64::wheel-0.38.4-py39haa95532_0  
xz           pkgs/main/win-64::xz-5.4.2-h8cc25b3_0  
zip          pkgs/main/win-64::zip-3.11.0-py39haa95532_0  
zlib         pkgs/main/win-64::zlib-1.2.13-h8cc25b3_0  
zstd         pkgs/main/win-64::zstd-1.5.5-hd43e919_0  
  
Proceed ([y]/n)? y|
```

패키지가 다운로드 되고,
Proceed가 나오면 y를 치고 Enter키를 눌러줍니다.

2-1. 가상환경 구축하기

Install 3

```
done
#
# To activate this environment, use
#
# $ conda activate semi_project
#
# To deactivate an active environment, use
#
# $ conda deactivate
```

위와 같은 말이 나오면 정상적으로 환경구축이 완료된 것입니다.

Check

```
(base) PS C:\Users\MAIN> conda env list
# conda environments:
#
base          * C:\ProgramData\anaconda3
prjsmi        C:\Users\MAIN\.conda\envs\prjsmi
semi_project  C:\Users\MAIN\.conda\envs\semi_project
```

가상환경이 잘 구축되었는지 확인하는 또 다른 방법은 가상환경리스트를 보는 것입니다.

2-1. 가상환경 구축하기

Check

```
(base) PS C:\Users\MAIN> conda activate semi_project  
(semi_project) PS C:\Users\MAIN> |
```

가상환경으로 들어가면 위와 같은 코드를 입력하면 되고
맨 앞이 바뀐다면 가상환경이 변경된 것입니다.

End

```
(semi_project) PS C:\Users\MAIN> conda deactivate  
(base) PS C:\Users\MAIN> |
```

가상환경을 나오고 싶다면 위와같은 코드를 입력하면 됩니다.

2-2. streamlit 메뉴얼

Streamlit 이란?

Streamlit은 Machine learning과 Data science project에서 사용자가 Web 애플리케이션을 쉽게 생성하고 공유할 수 있는 Python 오픈 소스 라이브러리입니다.

2-2. streamlit 메뉴얼

Install 1

```
Anaconda Powershell Prompt  
(prjsmi) PS D:\spyder-py3\python_basic> pip install streamlit
```

아나콘다 파워셸 프롬프트를 기준으로 설명하겠습니다.
프롬프트에 **pip install streamlit** 입력하고 실행하세요

Install 2

```
Requirement already satisfied: mdurl~=0.1 in d:\anaconda\envs\prjsmi\lib\site-packages (0.1.2)  
<14,>=10.14.0->streamlit) (0.1.2)  
Using cached streamlit-1.25.0-py2.py3-none-any.whl (8.1 MB)  
Installing collected packages: streamlit  
Successfully installed streamlit-1.25.0  
(prjsmi) PS C:\Windows\system32>
```

위처럼 나오면 설치가 끝난 것입니다.

2-2. streamlit 메뉴얼

Check

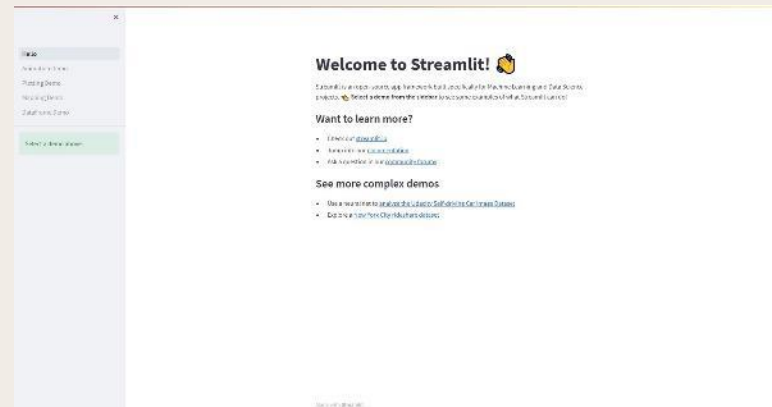
```
Anaconda Powershell Prompt
(prismi) PS D:\spyder-py3\python_basic\project_seven\test_webserver> streamlit hello

Welcome to Streamlit. Check out our demo in your browser.

Local URL: http://localhost:8501
Network URL: http://119.202.170.60:8501

Ready to create your own Python apps super quickly?
Head over to https://docs.streamlit.io

May you create awesome apps!
```



설치가 완료된 후 **streamlit hello**를 입력하고
실행하여 웹 페이지가 나오는지 확인하세요.

Check

```
(prismi) PS D:\spyder-py3\python_basic> python
Python 3.9.17 (main, Jul 5 2023, 21:22:06) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> import streamlit
```

혹은 **python**으로 들어가서 **import streamlit**이 실행되는지 확인해드립니다.
확인 작업이 잘 끝났다면 streamlit 설치는 완료입니다.

2-2. streamlit 메뉴얼

Deletion

```
Anaconda Powershell Prompt
(prjsmi) PS D:\download\멀티캠퍼스 세미프로젝트\streamlit 메뉴얼> pip uninstall streamlit
```

만약, streamlit 삭제를 원하신다면 **pip uninstall streamlit** 실행하면 됩니다.

Execute

```
Anaconda Powershell Prompt
(prjsmi) PS D:\download\멀티캠퍼스 세미프로젝트\streamlit 메뉴얼> streamlit run .\app.py

You can now view your Streamlit app in your browser.

Local URL: http://localhost:8501
Network URL: http://119.202.170.60:8501
```

streamlit run '파이썬파일' 을통해 실행할 수 있습니다.

2-2. streamlit 메뉴얼

Example 1

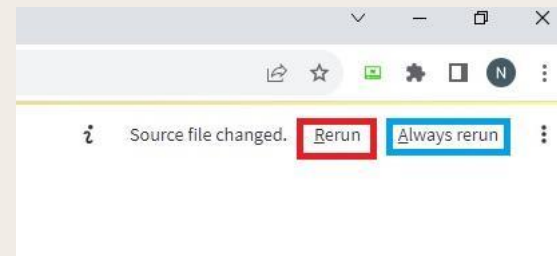
```
8 import streamlit as st
9
10 st.title('Hello World!')
11
```



예를들어 st.title()을 이용하여 Hello World!
라는 문구가 들어간 파이썬 파일을 실행하면 오른쪽과 같은 화면이 나옵니다.

Example 2

```
15 import streamlit as st
16 from datetime import datetime as dt
17
18 st.title('Hi!')
19
```

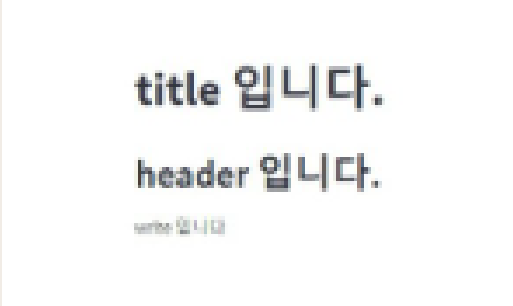


코드 수정시 rerun과 always rerun이 나오는데
수정사항이 적용된 상태에서 재실행을 의미합니다.
rerun은 직접 눌러줘야하고, always rerun은 실시간으로 수정사항이 적용됩니다.

2-2. streamlit 메뉴얼

Example 3

```
8 import streamlit as st
9
10
11 st.title('title 입니다.')
12 st.header('header 입니다.')
13 st.write('write 입니다')
14
```



title 입니다.

header 입니다.

write 입니다

st.title() : 큰 제목을 표시하는 데 사용됩니다.

화면 상단에 제목을 보여주며 주로 앱이나 섹션의 주요 주제를 강조하는 데 사용됩니다.

st.header() : 더 작은 크기의 제목을 표시하는 데 사용됩니다.

주로 섹션 또는 부분적인 내용을 구분하는 데 활용됩니다.

st.write() : 일반 텍스트를 표시하는 데 사용됩니다.

텍스트 외에도 숫자, 리스트, 테이블, 이미지 등 다양한 유형의 내용을 표시할 수 있습니다.

2-2. streamlit 메뉴얼

Example 4

```
10 st.title('st.dataframe()')
11
12 data = {'이름': ['Alice', 'Bob', 'Charlie'],
13         '나이': [25, 30, 22],
14         '직업': ['학생', '개발자', '디자이너']}
15 df = pd.DataFrame(data)
16
17 st.header('st.dataframe(df)')
18 st.dataframe(df)
19
20 st.header('가로 300, 세로 150, 인덱스 숨기고 나이, 이름을순으로 두개의 열만 표시')
21 st.dataframe(df, width=300, height=150, hide_index=True, column_order=['나이', '이름'])
22
23 st.header('각 열마다 다르게 설정')
24 column_config = {'이름': {'width': 100, 'justify': 'left'},
25                  '나이': {'width': 50, 'justify': 'right'},
26                  '직업': {'truncate': True}}
27 st.dataframe(df, column_config=column_config)
28
29
```



st.dataframe() : 데이터프레임을 시각화하여 웹에서 표시할 수 있게 해주는 기능입니다.

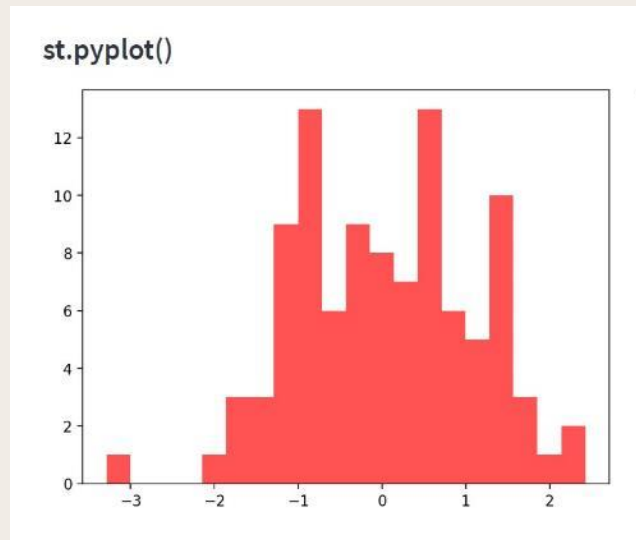
width:가로길이, height:세로길이,

hide_index:인덱스 숨김 유무, column_order:열의 정렬과 특정 열만 표시

2-2. streamlit 메뉴얼

Example 5

```
7 import streamlit as st
8 import matplotlib.pyplot as plt
9 import numpy as np
10
11 st.header('st.pyplot()')
12
13 fig = plt.figure()
14 ax = fig.add_subplot(1,1,1)
15 ax.hist(np.random.randn(100), bins = 20, color = 'red', alpha = 0.68)
16 st.pyplot(fig)
17
```



st.pyplot() : 그래프와 차트를 웹에서 표시할 수 있게 해주는 기능입니다.

2-2. streamlit 메뉴얼

Example 6

```
8 import streamlit as st
9 import pandas as pd
10 import matplotlib.pyplot as plt
11 import numpy as np
12
13 st.header('st.button()')
14 if st.button("클릭하세요"):
15     st.write("버튼이 클릭되었습니다!")
16     fig = plt.figure()
17     ax = fig.add_subplot(1,1,1)
18     ax.hist(np.random.randn(100), bins = 20, color = 'red', alpha = 0.68)
19     st.pyplot(fig)
20
21 st.header('st.checkbox()')
22 chk_button = st.checkbox('체크박스')
23
24 if chk_button:
25     st.write('체크박스 입니다.')
26
27 st.header('st.selectbox()')
28 s_box = st.selectbox('번호로 고르세요',
29                     ('1번', '2번', '3번'))
30
31 st.write('번호:', s_box)
32
33
```

st.button() : 웹에서 버튼을 생성하고 이를 통해 사용자와의 상호 작용을 가능하게 해주는 기능입니다.

st.checkbox() : 웹에서 체크박스를 생성하고 사용자의 선택 여부에 따라 다양한 동작을 수행할 수 있게 해주는 기능입니다.

st.selectbox() : 웹에서 선택 상자(드롭다운 메뉴)를 생성하고 사용자가 항목을 선택할 수 있게 해주는 기능입니다.

2-2. streamlit 메뉴얼

Example

7

```
8 import streamlit as st
9 import pandas as pd
10 import matplotlib.pyplot as plt
11 import numpy as np
12
13 st.header('st.text_box()')
14
15 st.text_input('텍스트를 입력하세요')
16 paswd = st.text_input('암호를 입력하세요. 입력되는 값은 보여지지 않습니다.', type="password")
17 st.write('암호는 :', paswd, ' 입니다')
18
19 st.header('st.number_box()')
20 num1 = st.number_input('첫번째 숫자를 입력하세요')
21 num2 = st.number_input('두번째 숫자를 입력하세요')
22 st.write('합은 ', num1+num2, ' 입니다')
23
24 st.header('st.text_area()')
25 sen = st.text_area('여러줄 입력하세요')
26 st.write(sen)
27
28
29 st.header('st.date_input()')
30 date = st.date_input('날짜를 선택해주세요')
31 st.write('선택하신 날짜는 ', date, ' 입니다')
32
33
34 st.header('st.time_input()')
35 times = st.time_input('시간을 선택해주세요. (15분 간격)')
36 st.write('선택하신 시간은 ', times, ' 입니다')
37
38
```

st.text_input() : 웹에서 텍스트를 입력받을 수 있는 입력 위젯(Widget)을 생성하는 함수입니다.

st.number_input() : 숫자 입력 위젯(Widget)을 생성하는 함수입니다.

st.date_input() : 날짜 입력 위젯(Widget)을 생성하는 함수입니다.

st.timer_input() : 시간 입력 위젯(Widget)을 생성하는 함수입니다.

st.text_area() : 텍스트 영역 입력 위젯(Widget)을 생성하는 함수입니다.

2-2. streamlit 메뉴얼

Example 8

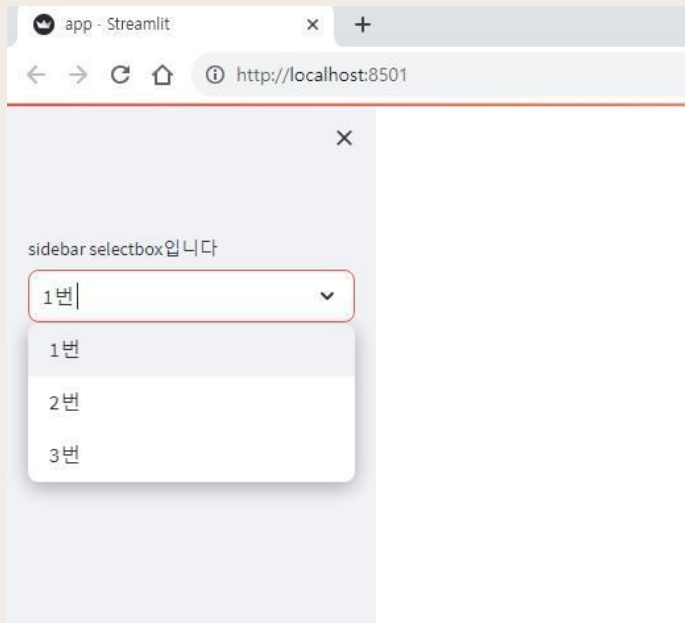
```
8 import streamlit as st
9 from streamlit_folium import folium_static
10 import folium
11
12 st.header('전국 주요 도시')
13 m = folium.Map(location=[36.5, 127.8], zoom_start=7)
14
15 cities = {
16     '서울': [37.5665, 126.9780],
17     '부산': [35.1796, 129.0756],
18     '대구': [35.8714, 128.6014],
19     '인천': [37.4536, 126.7317],
20     '광주': [35.1595, 126.8526],
21     '대전': [36.3504, 127.3845],
22     '울산': [35.5467, 129.3170],
23     '수원': [37.2636, 127.0286]}
24
25
26 for city in cities:
27     tooltip=city
28     folium.Marker(cities[city], popup=city, tooltip=tooltip).add_to(m)
29 folium_static(m)
30
```

folium_static() : Folium 라이브러리를 사용하여 생성한 지도를 Streamlit 웹 애플리케이션에 쉽게 표시할 수 있게 해주는 함수입니다.

2-2. streamlit 메뉴얼

Example 9

```
8 import streamlit as st
9
10
11 st.title('Hello World!')
12 item = st.sidebar.selectbox('sidebar selectbox입니다 ', ['1번', '2번', '3번'])
13
```



st.sidebar() : 웹의 사이드바에 컨트롤 요소를 추가하여
사용자와 상호 작용할 수 있는 기능을 제공하는 역할을 합니다.

2-2. streamlit 메뉴얼

End

```
(prjismi) PS D:\download\멀티캠퍼스 세미프로젝트\streamlit 메뉴얼> streamlit run .\app.py  
You can now view your Streamlit app in your browser.  
  
Local URL: http://localhost:8501  
Network URL: http://119.202.170.60:8501  
  
Stopping... ← Ctrl + c  
(prjismi) PS D:\download\멀티캠퍼스 세미프로젝트\streamlit 메뉴얼>
```

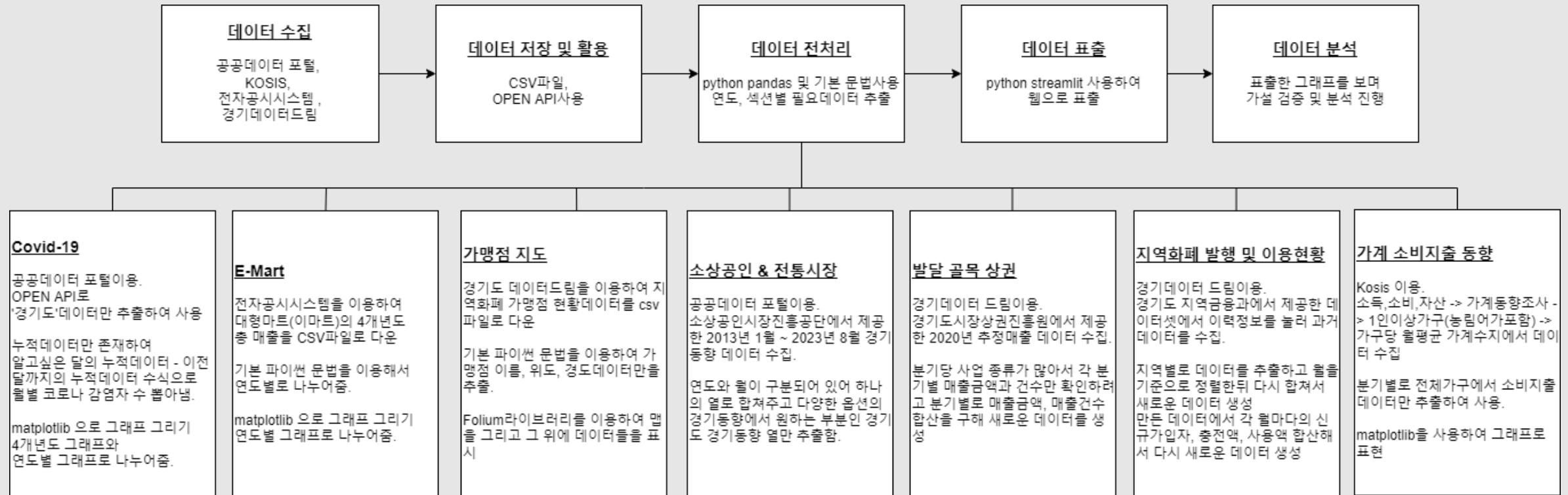
웹 페이지 표출을 멈추고 싶으면 프롬프트에서 Ctrl+c 를 눌러주면 됩니다.

이 때, 웹을 먼저 종료하면 Ctrl+c 가 먹이지 않아 프롬프트를 재실행 해야 합니다.

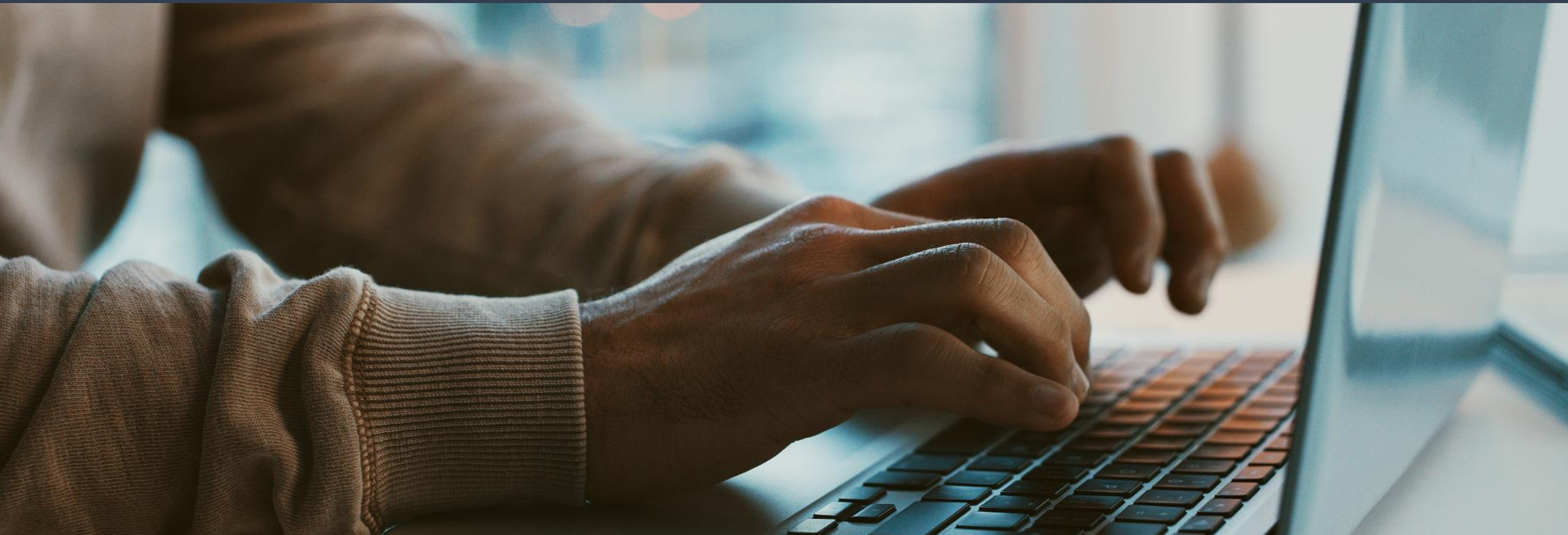
Part 3 프로젝트 Diagram



3. 프로젝트 Diagram



Part 4 프로젝트 내용(데이터 전처리)



4. 프로젝트 내용(데이터 전처리)

1	월별 경기도 코로나 현황
2	월별 이마트 매출액
3	분기별 가계 소비지출 현황
4	지역화폐이용현황(가맹점 지도)
5	소상공인 & 전통시장 경기동향
6	발달 골목 상권 매출 현황

4-1. 월별 경기도 코로나 현황

```
# 라이브러리 import
import requests

# url 입력
url = 'http://apis.data.go.kr/1352000/ODMS_COWID_04/callCovid04Api? #
serviceKey= &apiType=xml&gubun=경기'

# url 불러오기
response = requests.get(url)

#데이터 값 출력해보기
contents = response.text
```

2-1. 누적 확진자 수

	월	월별 확진자수
0	2020-01	2
1	2020-02	128
2	2020-03	3,070
3	2020-04	4,096
4	2020-06	8,061
5	2020-05	9,657
6	2020-08	13,087
7	2020-07	20,878
8	2020-09	39,149
9	2020-10	59,076

2-2. 월별 확진자 수

	월	월별 확진자수
0	2020-01	2
1	2020-02	53
2	2020-03	421
3	2020-04	192
4	2020-05	172
5	2020-06	290
6	2020-07	390
7	2020-08	1,740
8	2020-09	1,073
9	2020-10	1,043

1

공공 데이터 포털에서 OPENAPI 사용.
OPENAPI 인증키 발급 및 필요한 경기도 데이터만 추출

2

데이터 형태 확인.
일자별 누적데이터만 제공되어 데이터 가공 필요.
월별 누적데이터를 먼저 구한 후 파이썬 기본 문법과
수학 수식 사용하여 월별 확진자 수 계산

3

월별 확진자 수 데이터를 matplotlib을 이용하여
그래프로 표시
4개년도 통합 그래프와 연도별 그래프로 분리

-사용데이터출처

보건복지부_코로나19 시도 발생현황 : <https://www.data.go.kr/data/15098776/openapi.do>

4-2. 월별 이마트 매출액

	A	B	C	D	E	F	G
1	기준년월	매출액(단위:억원)					
2	20년 1월	13406					
3	20년 2월	10438					
4	20년 3월	10817					
5	20년 4월	10687					
6	20년 5월	10863					
7	20년 6월	10822					
8	20년 7월	11330					
9	20년 8월	12851					
10	20년 9월	14418					
11	20년 10월	11750					
12	20년 11월	11526					
13	20년 12월	13230					
14	21년 1월	13778					
15	21년 2월	12910					
16	21년 3월	11692					
17	21년 4월	11422					
18	21년 5월	12641					
19	21년 6월	11556					
20	21년 7월	13143					
21	21년 8월	13355					
22	21년 9월	13269					
23	21년 10월	12954					
24	21년 11월	10945					

2. 데이터 전처리

	기준년월	매출액(단위:억원)
0	20년 1월	13,406
1	20년 2월	10,438
2	20년 3월	10,817
3	20년 4월	10,687
4	20년 5월	10,863
5	20년 6월	10,822
6	20년 7월	11,330
7	20년 8월	12,851
8	20년 9월	14,418
9	20년 10월	11,750

1

전자공시시스템 포털에서 이마트 월별 매출액
xlsx파일 다운

2

월별로 파일이 나누어져 있어 하나의 파일로
통합 후 csv파일로 저장

3

데이터 형태 확인.
연도와 월로 분리하고, Matplotlib을 이용하여
매출액 그래프로 표시

4-3. 분기별 가계 소비지출 현황

#	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
1	가계수지항목별	2019.1/4	2019.1/4	2019.1/4	2019.2/4	2019.2/4	2019.2/4	2019.3/4	2019.3/4	2019.3/4	2019.4/4	2019.4/4	2019.4/4	2019.4/4	2020.1/4	2020.1/4	2020.2/4	2020.2/4	2020.2/4	2020.3/4	2020.3/4	2020.3/4
2	가계수지항목별	현재가	현재가	현재가	현재가	현재가	현재가	현재가	현재가	현재가	현재가	현재가	현재가	현재가	현재가	현재가	현재가	현재가	현재가	현재가	현재가	현재가
3	가구원수 (명)	2.42	2.61	2.18	2.41	2.56	2.19	2.42	2.56	2.22	2.42	2.55	2.22	2.4	2.53	2.22	2.4	2.52	2.25	2.4	2.52	2.24
4	가구주원명 (세)	52.2	46.71	59.57	52.15	47.6	58.9	52.03	47.17	58.99	52.08	47.3	59.08	52.43	47.13	59.55	52.42	47.4	59.17	52.46	47.37	59.62
5	가구주원명 (%)	100	57.28	42.72	100	59.72	40.28	100	58.87	41.13	100	59.41	40.59	100	57.32	42.68	100	57.32	42.68	100	58.44	41.56
6	소득	4310499	5266897	3027993	4188089	4913392	3312592	4324945	5036494	4295641	4846492	3489926	4363452	3273328	3141261	4334892	5055560	3367189	4369000	5072602	3379704	4379704
7	경상소득	4225444	5189122	2933209	4130260	4867934	3036420	4270077	4976894	3258332	4225632	4771790	3426280	4250646	5165885	3021250	4263406	4985741	3293465	4299377	5003445	3309427
8	근로소득	2792104	4488477	517350	2728106	4215804	522112	2805548	4353349	590010	2733312	4197524	590339	2813890	4470126	589149	2587923	4107007	548118	2776200	4345676	569446
9	재산소득	781338	193684	1569354	837158	229840	1737702	857309	195673	1804382	977609	223739	2080949	778950	200776	1555582	781957	175158	1596759	851800	183309	1791728
10	재산소득	34403	30706	39361	45418	50240	38268	27709	16340	43983	25209	15704	39121	37990	30191	48465	26631	16548	40170	31000	24431	40236
11	이전소득	617603	476254	807145	519579	372050	738337	579511	411533	819958	489502	334832	715871	619817	464792	828053	866895	687028	1108418	640378	450029	908017
12	비밀상소득	8504	77775	94784	57828	45456	76172	54867	59600	48093	70209	74693	63646	112886	107443	120011	71486	69819	73724	69823	69157	70278
13	가계지출	3489987	4063951	2720331	3206094	3670565	2517365	3404750	3890609	2709286	3296072	3747998	2634647	3262172	3763462	2588814	3195810	3636881	2603546	3278032	3711115	2669098
14	소비지출	2567902	2902648	2119024	2367963	2627497	1983119	2480951	2765597	2073506	2470792	2745422	2068852	2378150	2645608	2088886	2395785	2642353	2064696	2421813	2653109	2096660
15	01.식료품·비주류음료	344983	360889	323654	322964	337479	301442	365012	369764	358211	356348	357261	355012	368004	389973	338682	370417	383985	352199	399302	405819	390139
16	02.주류·담배	36833	40519	31890	35692	38571	31422	37372	39620	34154	33788	35822	30812	34412	37165	30713	38924	42230	34486	40226	44089	34794
17	03.의류·신발	135784	162040	100576	145812	170837	108704	112384	130118	86998	163792	187478	129127	97753	116271	72879	134899	159473	101900	96056	112488	72952
18	04.주거·수도·광열	323237	332242	311162	266303	274940	253484	253635	254958	251743	272809	285616	254065	316302	322362	308163	282712	2971618	270754	266032	281350	244495
19	05.가정용품·가사서비스	118131	135381	95418	118924	124763	110267	109911	123632	90270	115932	12										
20	06.보건	210647	207894	214338	200877	198579	204284	198620	189929	212492	211192	21										
21	07.교통	288300	334865	225858	259720	304109	193898	299674	366962	203358	312594	36										
22	08.통신	114190	126438	97766	119537	128211	106676	122083	134180	104767	126201	13										
23	09.오락·문화	193763	227422	148628	176997	198031	145808	181637	207761	144244	162754	15										
24	10.교육	252079	317229	164717	160725	205734	93984	238115	294839	156920	151323	16										
25	11.음식·숙박	339944	415616	237359	346866	410804	252058	356567	422869	261662	352387	47										
26	12.기타상품·서비스	209608	239322	169762	215057	234991	185499	203673	225507	172420	212764	23										
27	비소비지출	922086	1161303	601307	838131	1043069	534246	923799	1125012	635780	825279	106										
28	저분가농소득	3388404	4105594	2426686	3349957	3870323	2578346	3401145	3911482	2670645	3470561	384										
29	소득	820502	1202947	307663	981995	1242827	595227	920194	1145885	597139	999769	105										

2. 데이터 전처리

	Year/Quarte	Consumption Expenditure
0	2019.1/4	2,567,902
1	2019.2/4	2,367,963
2	2019.3/4	2,480,951
3	2019.4/4	2,470,792
4	2020.1/4	2,378,150
5	2020.2/4	2,395,785
6	2020.3/4	2,421,813
7	2020.4/4	2,402,804
8	2021.1/4	2,383,194
9	2021.2/4	2,425,835

-사용데이터출처

Kosis 가구당 월평균 가계수지:

https://kosis.kr/statHtml/statHtml.do?orgId=101&tblId=DT_1L9U001&vw_cd=MT_ZTITLE&list_id=G_A_10_003_001&seqNo=&lang_mode=ko&language=kor&obj_var_id=&itm_id=&conn_path=MT_ZTITLE

1

Kosis 포털에서 CSV 파일 다운로드.

데이터형태확인.

행 데이터는 분기별/가구별로 나뉘어져 있고
열 데이터는 항목별로 나뉘어져 있다.

2

파이썬 기본 문법(인덱싱)을 이용해서
필요한 행과 열을 선택 후 데이터 추출

3

분기별 가계 소비지출데이터를 matplotlib을
이용하여 그래프로 표시

4-4. 지역화폐이용현황

지역화폐발행및이용현황.csv - sFree Office for Docs and PDF Calc

파일(F) 편집(E) 보기(V) 삽입(I) 서식(O) 스타일(Y) 시트(S) 데이터(D) 도구(T) 창(W)

맑은 고딕 10 pt

	A	B	C	D	E
	시군명	기준년월	월별신규가입자수(명)	월별충전액(천원)	월별사용액(천원)
1	시군명	기준년월	월별신규가입자수(명)	월별충전액(천원)	월별사용액(천원)
2	가평군	2020-12	579	1177615	1121896
3	가평군	2020-11	524	1014259	991877
4	가평군	2020-10	550	921618	952087
5	가평군	2020-09	862	1052610	964030
6	가평군	2020-08	312	597319	661956
7	가평군	2020-07	533	676234	732454
8	가평군	2020-06	398	616944	518214
9	가평군	2020-05	3425	713833	642870
10	가평군	2020-04	6835	639698	477477
11	가평군	2020-03	655	389311	297771
12	가평군	2020-02	211	180458	
13	가평군	2020-01	338	288742	
14	고양시	2020-12	7511	8159264	
15	고양시	2020-11	10341	21892388	
16	고양시	2020-10	10941	18871423	
17	고양시	2020-09	20944	25220665	
18	고양시	2020-08	7545	22718749	
19	고양시	2020-07	16520	10801299	
20	고양시	2020-06	7044	7521775	
21	고양시	2020-05	60602	24391234	
22	고양시	2020-04	113877	10835285	
23	고양시	2020-03	10166	8183828	
24	고양시	2020-02	5041	6134958	
25	고양시	2020-01	6758	9669657	
26	과천시	2020-12	781	1651913	
27	과천시	2020-11	981	2409712	
28	과천시	2020-10	824	2222257	

	월별신규가입자수(명)	월별충전액(천원)	월별사용액(천원)
2020-01	2,514.6071	2,530,340.4483	2,091,412.2759
2020-02	2,310.2857	1,990,737.2759	1,880,574.1724
2020-03	5,057.3571	3,020,029.7931	2,292,041.4828
2020-04	52,050	7,230,893.0345	4,102,790
2020-05	33,283.7857	6,257,853.4828	5,458,055.4138
2020-06	5,685.2857	6,386,726.6552	5,862,454.2759
2020-07	7,427.1429	7,527,029.9655	7,508,314.4828
2020-08	2,932.75	6,157,532.4828	6,809,348.0345
2020-09	7,520.6429	9,472,084.4138	8,805,915.5172
2020-10	4,121.5714	8,253,279.5862	8,773,961.7586

1

경기도 데이터드림 포털에서 csv파일 다운로드.

2

데이터 확인.
시흥, 김포시 데이터와 성남시 신규가입자수 부분
결측값 확인

3

데이터 전처리과정
파이썬 pandas를 사용하여 신규가입자수, 월별충전액,
월별사용액 등 필요한 데이터 추출

-사용데이터출처

지역화폐 발행 및 이용 현황 : <https://data.gg.go.kr/portal/adjust/service/selectHisServicePage.do?infId=6FEDD6KGEJWYCY2G15OY29527318&infSeq=1&hisBsDt=20230630#none>

4-4. 지역화폐이용현황(가맹점 지도)

군명	상호명	업종명(종목명)	소재지도로명주소	소재지지번주소	우편번호	위도	경도	사업자등록번호	업종코드	가맹번호	휴업상태	휴업상태코드	제업일자	데이터기준일자
0 연시	0525기동현정형	편의점	경기도 용인시 기흥구 구갈로28번길 21-13	경기도 용인시 기흥구 구갈동 396번지 한양아파트 상가동 108호	16,973	37.2787	127.1134	3531500645	4030	794,816,640	폐업자	3	2023-04-30	2023-06-09
1 산시	장가네손잡국수	스넥	경기도 오산시 오산로272번길 12	경기도 오산시 오산동 855-15번지 1층	16,136	37.151	127.0698	1245153519	8021	794,076,414	계속사업자	1	None	2023-06-09
2 현군	상일공사	보일러·냉도·냉기	None	경기도 연천군 연천읍 차탄리 34-17번지	None	None	None	1272447247	9001	721,406,967	계속사업자	1	None	2023-06-09
3 산시	죽고갈래서갈래	일반한식	경기도 오산시 우청로 42	경기도 오산시 내삼미동 861-4번지 A동 1층	16,110	37.174	127.0562	3582100262	8001	798,722,054	계속사업자	1	None	2023-06-09
4 연시	백암푸러시집밥	일반한식	경기도 용인시 처인구 백암면 덕평로 1-1	경기도 용인시 처인구 백암면 근곡리 12-8번지 1층 108호	17,172	37.1708	127.3707	8262001127	8001	723,768,607	계속사업자	1	None	2023-06-09
5 산시	상성계탕용명업소	기타사우용물	None	경기도 오산시 오산동 123-2	None	None	None	1241721283	5299	746,327,229	계속사업자	1	None	2023-06-09
6 산시	산전떡볶이 율동점	스넥	경기도 오산시 갈리사로 51	경기도 오산시 율동 626-3번지 율동빌딩 1층	16,121	37.1598	127.0563	6244300313	8021	709,728,813	계속사업자	1	None	2023-06-09
7 불시	신가네 왕문어통태피게	일반한식	경기도 의왕시 오정동업길 19	경기도 의왕시 오정동 174번지 대원테크노빌드 108호	16,672	37.3552	126.9662	2904000671	8001	716,835,322	계속사업자	1	None	2023-06-09
8 불시	봉구버어 의왕오전통탕	주점	경기도 의왕시 분골로 24	경기도 의왕시 오정동 104-1번지 2층	16,050	37.359	126.9687	7192700931	8013	711,090,640	계속사업자	1	None	2023-06-09
9 불시	부전고추떡방앗간	계과점	경기도 의왕시 사그나1길 1	경기도 의왕시 고현동 274-2번지	16,061	37.348	126.9751	1380971934	8301	772,955,255	계속사업자	1	None	2023-06-09
10 불시	물푸름집	미용점	경기도 의왕시 부곡시장길 8	경기도 의왕시 상동 185-1번지 9호	16,096	37.3199	126.9499	1380691147	7102	762,540,131	계속사업자	1	None	2023-06-09
11 불시	장가네청과	농축수산품	None	경기도 의왕시 내손동영1길 37.202호 (내손동)	None	None	None	8759200689	8310	711,737,283	계속사업자	1	None	2023-06-09
12 형부A	0525신곡음용점	편의점	경기도 의정부시 발곡로 5	경기도 의정부시 신곡동 707-11번지	11,729	37.7314	127.0555	6822400832	4030	715,586,981	계속사업자	1	None	2023-06-09
13 형부A	불라논병태도점	일반한식	경기도 의정부시 부흥로95번길 10	경기도 의정부시 경운동 471-3번지 110,111호	11,757	37.7514	127.0686	4065500126	8001	799,309,590	계속사업자	1	None	2023-06-09

1

경기도 데이터드림 포털에서 csv파일 다운로드.
데이터형태, 휴업상태코드 확인

2

데이터 전처리과정
파이썬 pandas를 사용하여 가맹점의 상호명, 위도와 경도를 추출한다.
결측값이 있으면 행을 삭제

3

데이터가 너무 많아 가독성이 떨어져 MarkerCluster를 사용하여 비슷한 위치의 가맹점은 묶어서 지도에 노출

```
script_dir = os.path.dirname(os.path.abspath(__file__)) # 현재 스크립트 파일의 디렉토리 경로
path_dir = os.path.join(script_dir, 'row_data')

#데이터 불러오기, low=memory=False는 column에 NaN값이나 여러 type의 데이터가 섞여 있으면 경고가 뜨는데 경고 메시지가 출력되지 않게 하는 것
data = pd.read_csv(path_dir + f'\\가맹점.csv', low_memory=False)
#1.0은 지금 계속 영업 하는 곳
r_data = data[data['휴업상태코드']==1.0]
rr_data = r_data.loc[:,['상호명','위도','경도']]
df = rr_data.dropna(axis=0) #결측값 있는 행 제거
```

```
locations = list(zip(df.iloc[:,1], df.iloc[:,2]))

#markercluster : 비슷한 위치에 있는 곳들을 묶어주는 함수
map = folium.Map(location=[37.541, 126.986], zoom_start=12)
mc = MarkerCluster()

for _, i in df.iterrows():
    mc.add_child(
        Marker(location = [i['위도'],i['경도']],
            popup=i['상호명']))
map.add_child(mc)

map.save('가맹점.html')
```

-사용데이터출처

지역화폐 가맹점 데이터 :

<https://data.gg.go.kr/portal/data/service/selectServicePage.do?page=1&sortColumn=&sortDirection=&infId=3NPA52LBMO36CQEQ1GMY28894927&infSeq=1&searchWord=%EC%A7%80%EC%97%AD%ED%99%94%ED%8F%90+%EA%B0%80%EB%A7%B9%EC%A0%90>

4-5. 소상공인 & 전통시장 경기동향

소상공인

전통시장

연월	경기체감	경기전망
2020.6	78.7	96.7
2020.7	65.1	89
2020.8	67.4	92
2020.9	53.2	92.9
2020.10	80.4	95.8
2020.11	84.9	99
2020.12	51	97.3
2021.1	38.6	87.8
2021.2	48.5	67.4
2021.3	65.8	76

-사용데이터출처

소상공인 경기동향 : <https://www.data.go.kr/data/3060077/fileData.do>

전통시장 경기동향 : <https://www.data.go.kr/data/3060078/fileData.do>

1

공공 데이터포털에서 csv파일 다운로드.

2

데이터형태확인.
연도와 월이 다른 열로 구분되어있음
빈값은 삭제하고 필요한 데이터만 추출

3

코로나 발생 이전과 이후를 비교하기 위해
2019년도의 데이터 부터 추출하여 그래프 작성

4-6. 발달 골목 상권 매출 현황

[illegible]

	기준연도	기준분기	매출금액	매출건수
0	2,020	1	4,309,150,807,872	69,095,884
1	2,020	2	4,532,611,502,682	72,748,329
2	2,020	3	4,308,603,124,854	71,642,647
3	2,020	4	4,180,471,766,909	66,494,762

-사용데이터출처

발달 골목 상권 :

<https://data.gg.go.kr/portal/data/service/selectServicePage.do?page=1&rows=10&sortColumn=&sortDirection=&infId=B7BIA8NM4VIXSPPQQV0132089283&infSeq=3&order=&loc=&searchWord=%EA%B3%A8%EB%A%A9>

1

경기데이터드림에서 2020년도 매출 현황 csv파일 다운

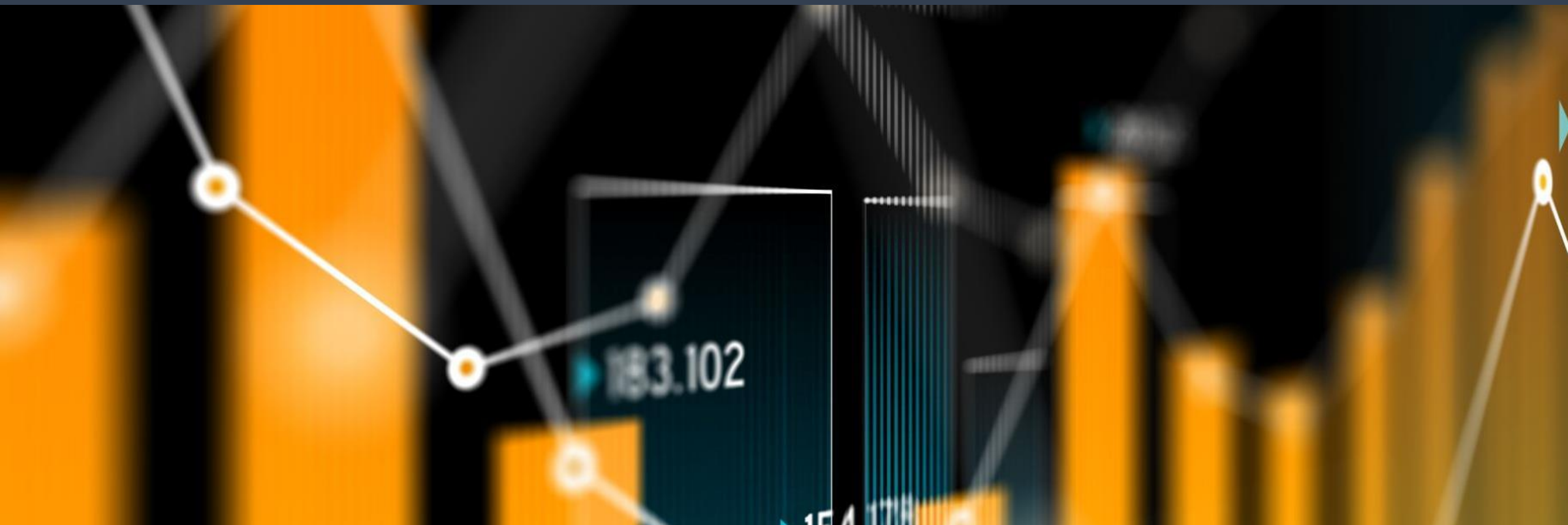
2

데이터확인
매출 금액과 매출 건수는 BC카드 데이터를 기반으로
추정된 값.
기준연도 및 분기로 데이터 합산

3

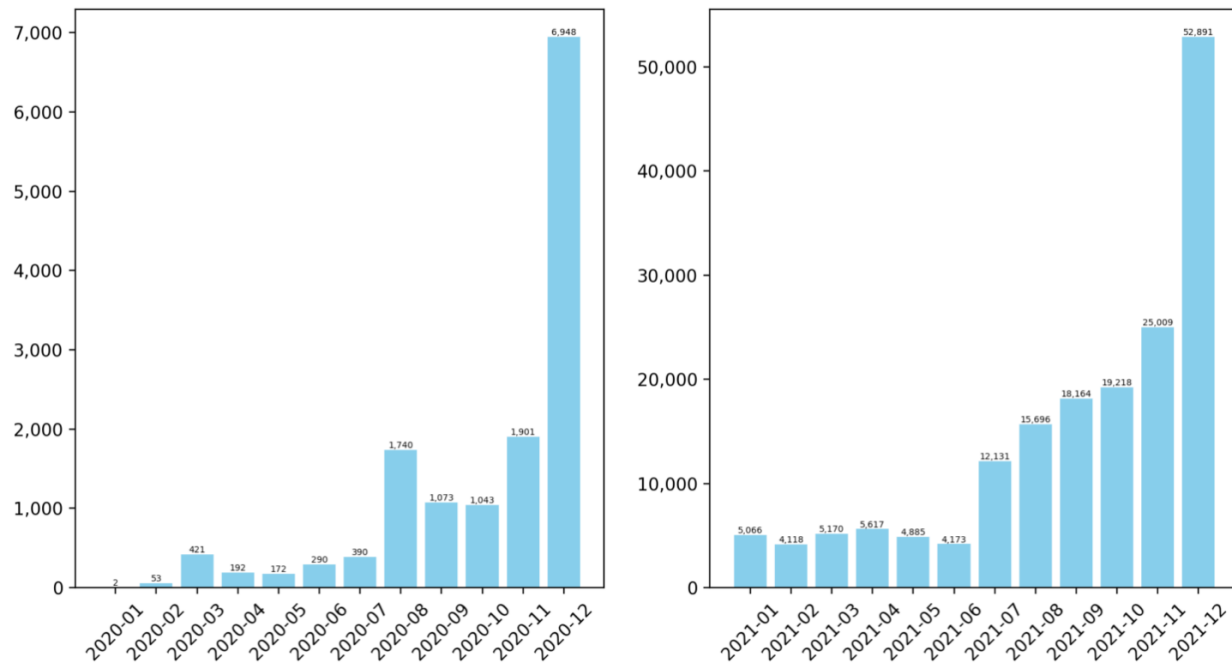
합산된 데이터를 사용하여 Matplotlib으로 그래프 그리기.

Part 5 프로젝트 결과



5. 프로젝트 결과

경기도 코로나 확진자 수



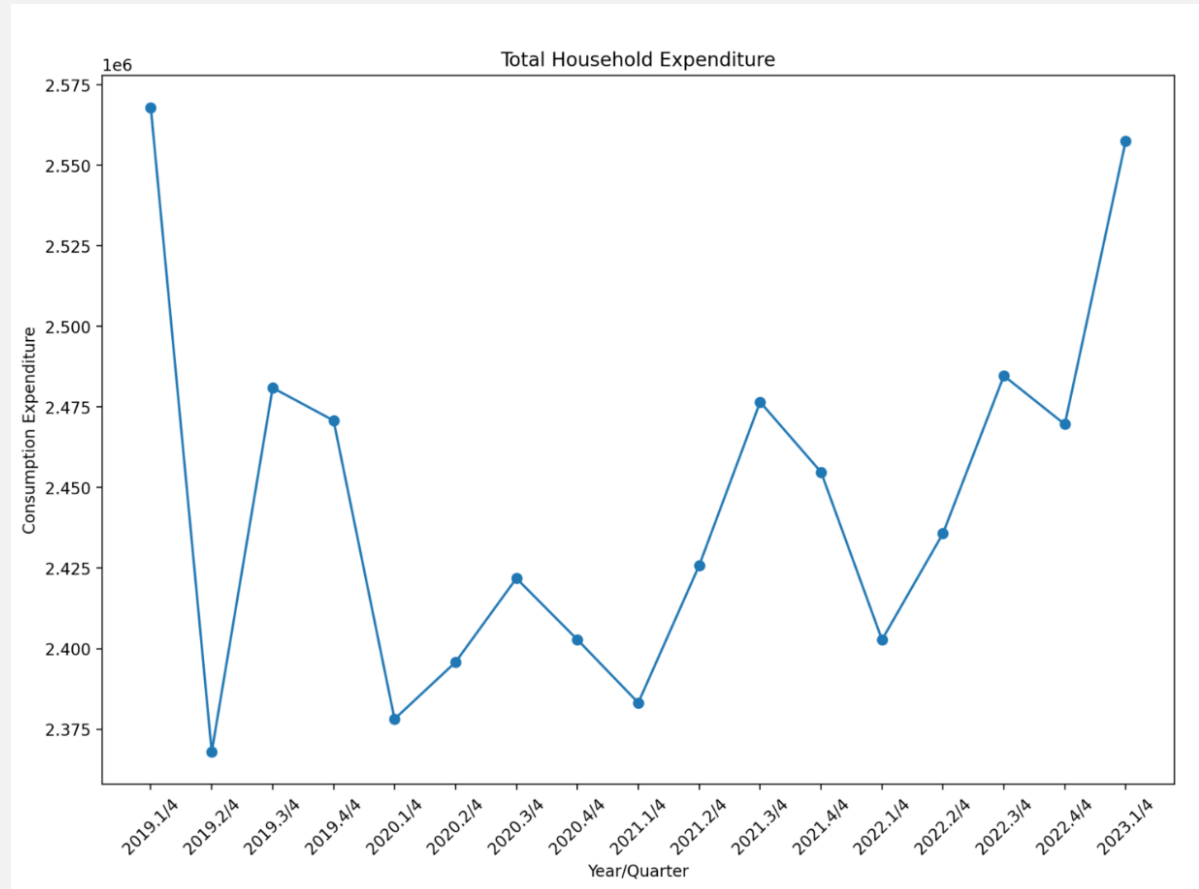
2020~2021년도 경기도 월별 코로나 현황

2020년 초반에는 코로나 바이러스의 확산이 시작되었습니다. 초기에는 느린 속도로 증가하다가 2020년 말에 급속도로 증가했습니다. 이후로도 확진자 수는 파도처럼 오르락 내리락하는 모습을 보였습니다. 2021년에도 여전히 높은 확진자 수를 유지했으며, 21년도 11월 달 오미크론 등의 변이 바이러스가 처음 확산됨에 따라 확진자 수가 급증하는 추이를 보였습니다.

5. 프로젝트 결과

월평균가계소비지출

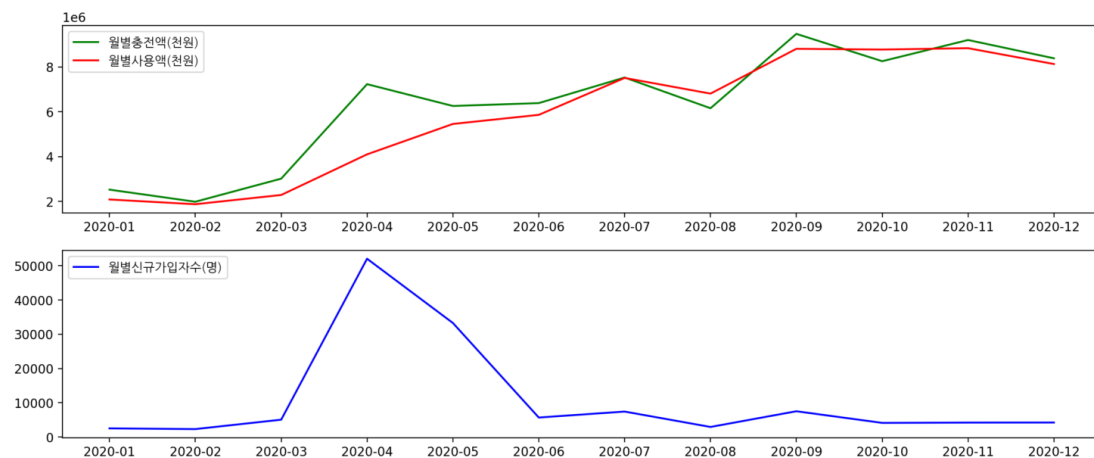
20년 초반에 코로나 바이러스의 확산으로 전분기 대비 소비지출이 급감했습니다. 20년 5월 1차재난지원금이 지급되면서 2분기, 3분기는 회복되는 추세를 보였지만 20년 말 다시 코로나 확진자 수가 늘어나면서 소비지출이 줄어든 것으로 보입니다. 21년 초에 재난지원금이 지급되면서 소비지출은 다시 늘어나는 것을 알 수 있습니다.



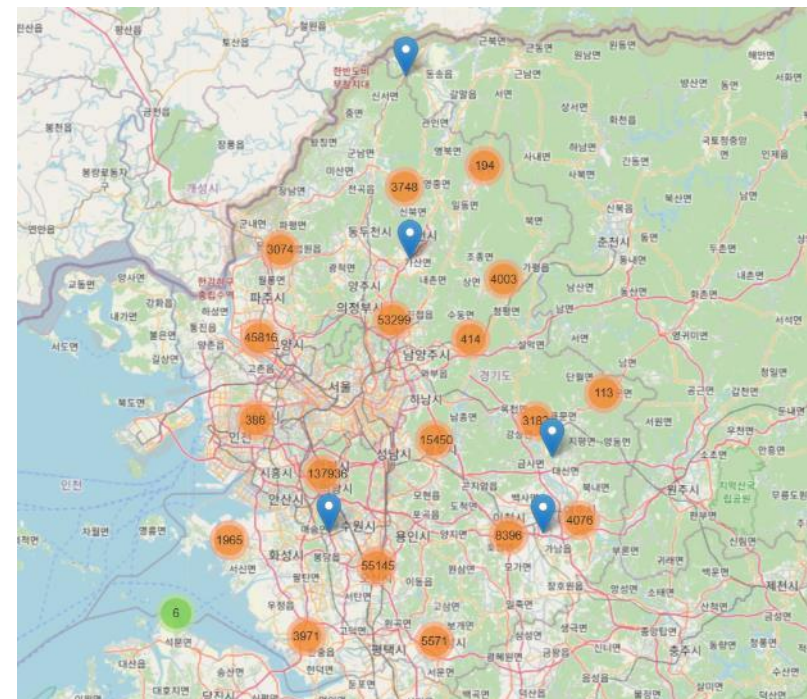
5. 프로젝트 결과

경기도지역화폐이용현황

경기도 지역화폐 발행 및 이용 현황

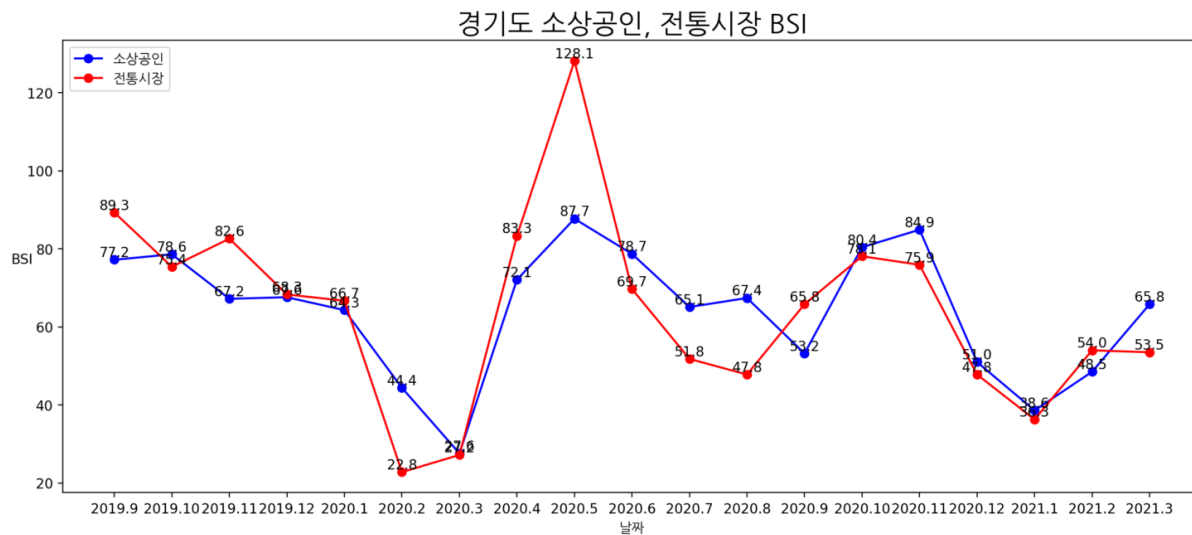


가맹점지도



20년 4월부터 지급되었던 1차 경기도재난기본소득 전후로 월별 신규 가입자수와 이용자수가 늘어난 것으로 보아 재난지원금을 지역화폐로도 제공하고, 홍보한 것이 효과가 있어 보입니다. 사용자가 늘어나는 만큼 경기도 곳곳에 가맹점이 위치하고 있습니다.

5. 프로젝트 결과



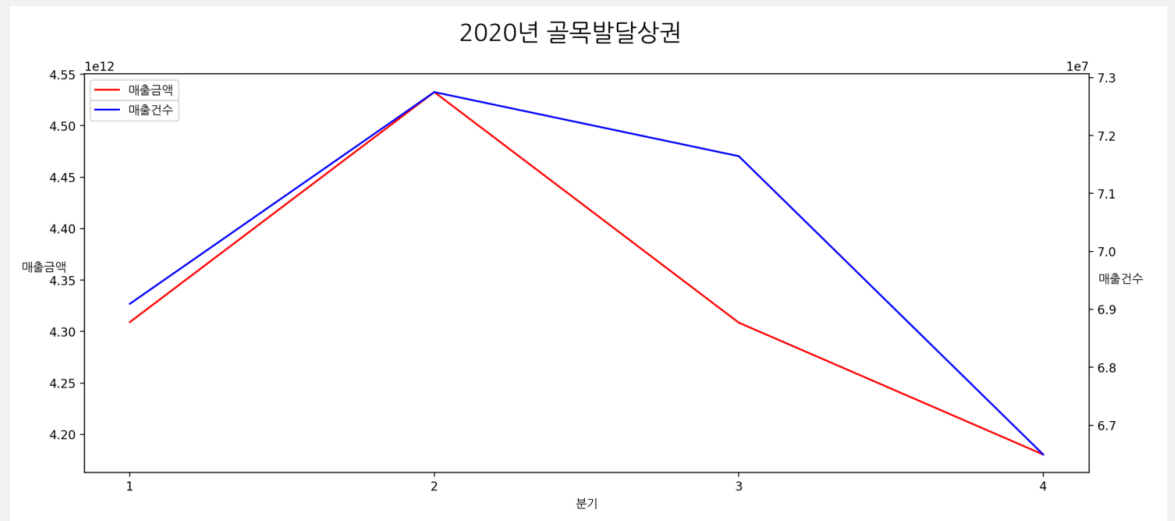
소상공인&전통시장 경기체감

- BSI 그래프에서 2019년 9월부터 2019년 12월까지의 추세를 살펴보면 하락하는 경향을 보이지만 급격한 하락은 나타나지 않았습니다. 그리고 코로나 발생 달인 12월 이후 급격한 하락이 관찰되어 코로나 발생이 소비패턴에 변화를 미친 것으로 판단됩니다.
- 2020년 5월에 지급된 1차 재난지원금으로 인해 소상공인과 전통시장 모두 체감 지수가 높아지고 다음 달부터 하락하는 경향을 보입니다. 이는 지원금으로 인해 일시적인 소비 활동이 증가했지만 이후 유지되지 않았거나, 전 달과의 비교에서 체감 지수가 낮아진 것일 수 있습니다.
- 2차 지원금과 3차 지원금은 지원 대상이 제한적이지만 각각 7조8천억 원과 9조3천억 원의 규모로 2020년 9월과 2021년 1월에 체감 지수가 상승한 것을 확인할 수 있습니다.

5. 프로젝트 결과

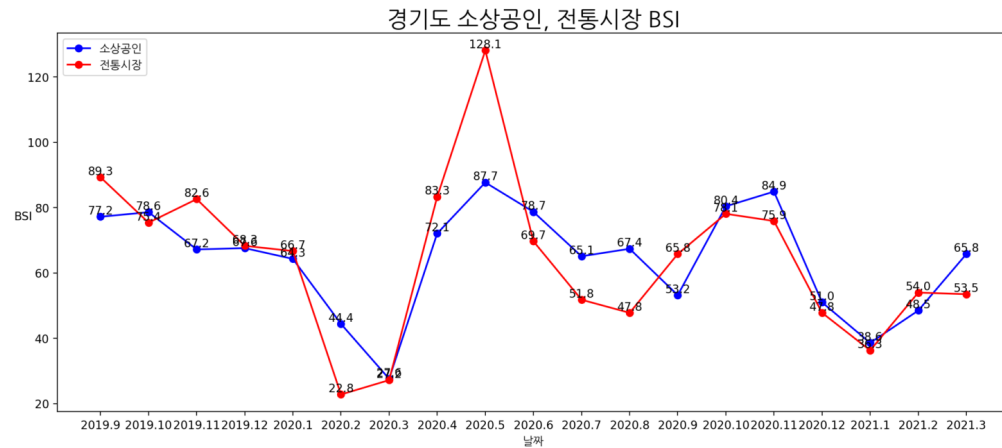
2020년 골목발달상권

2020년 발달골목상권 그래프를 살펴보면 매출 금액과 건수가 2분기에 고점을 찍고 서서히 감소하는 경향을 보입니다. 2분기에는 1차 지원금(20년 5월)으로 인한 잠깐의 상승이 관찰되며, 3분기(20년 9월)에는 매출 건수는 많은데 금액은 적어진 것으로 보아 편의점이나 동네 슈퍼와 같은 저가 상품의 구매가 늘어난 것으로 예상됩니다.

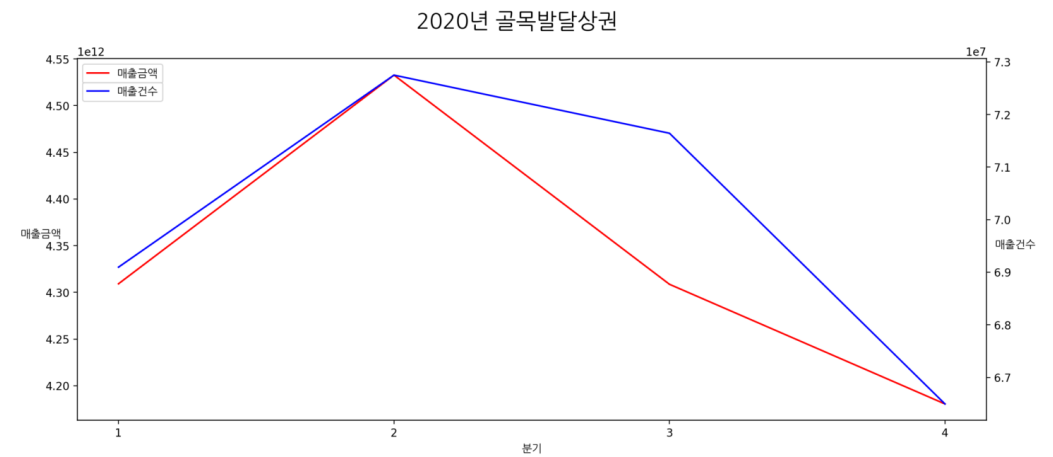


5. 프로젝트 결과

소상공인&전통시장 경기체감



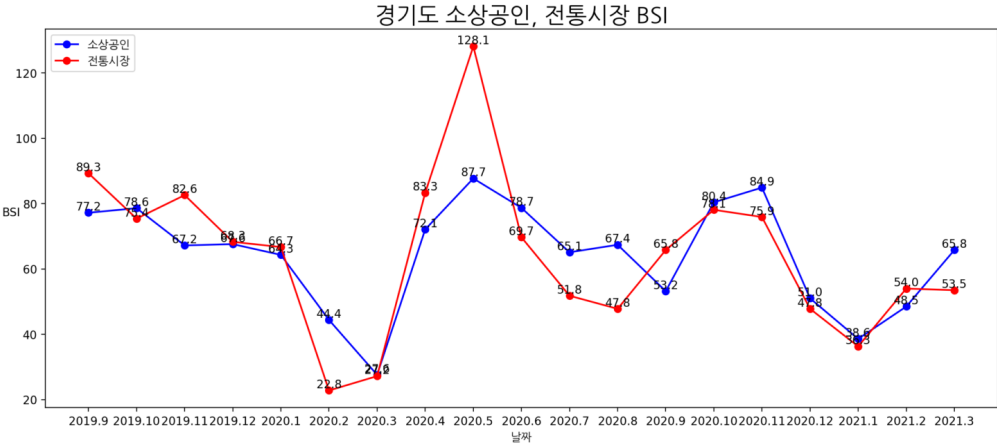
2020년 골목발달상권



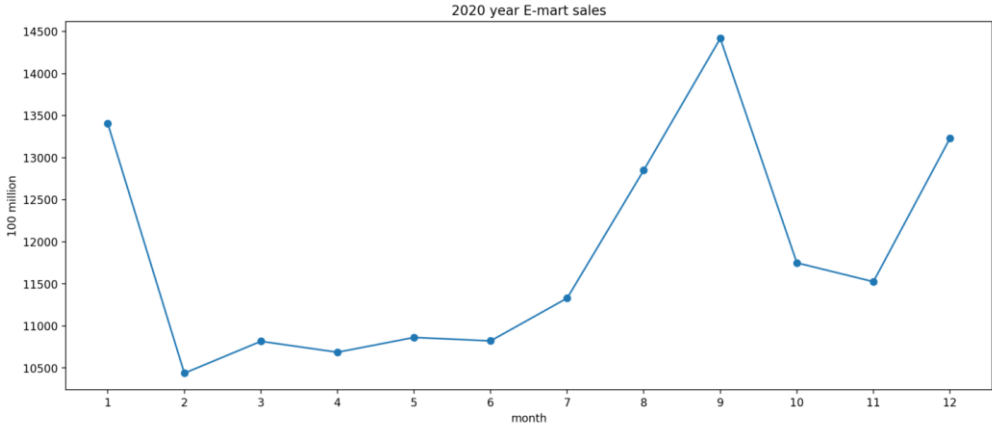
골목발달상권 매출 그래프가 3분기와 4분기에 하락하는 반면, BSI 지수가 오르락하며 변동을 보이는 것을 확인할 수 있습니다. 이를 통해 20년 9월 2차와 21년 1월 3차 지원금 대상자들의 소비 활동이 증가했지만 그렇지 않은 사람들은 더욱 감소한 것으로 해석할 수 있습니다. 추가적인 데이터가 없어 확인이 어려우나, 코로나로 인한 거리두기, 실업, 폐업 등의 영향으로 전체 소비는 줄었지만 지원금이 지급될 때마다 해당 대상자들의 소비만 일시적으로 증가한 것으로 예상됩니다.

5.프로젝트 결과

소상공인&전통시장 경기체감

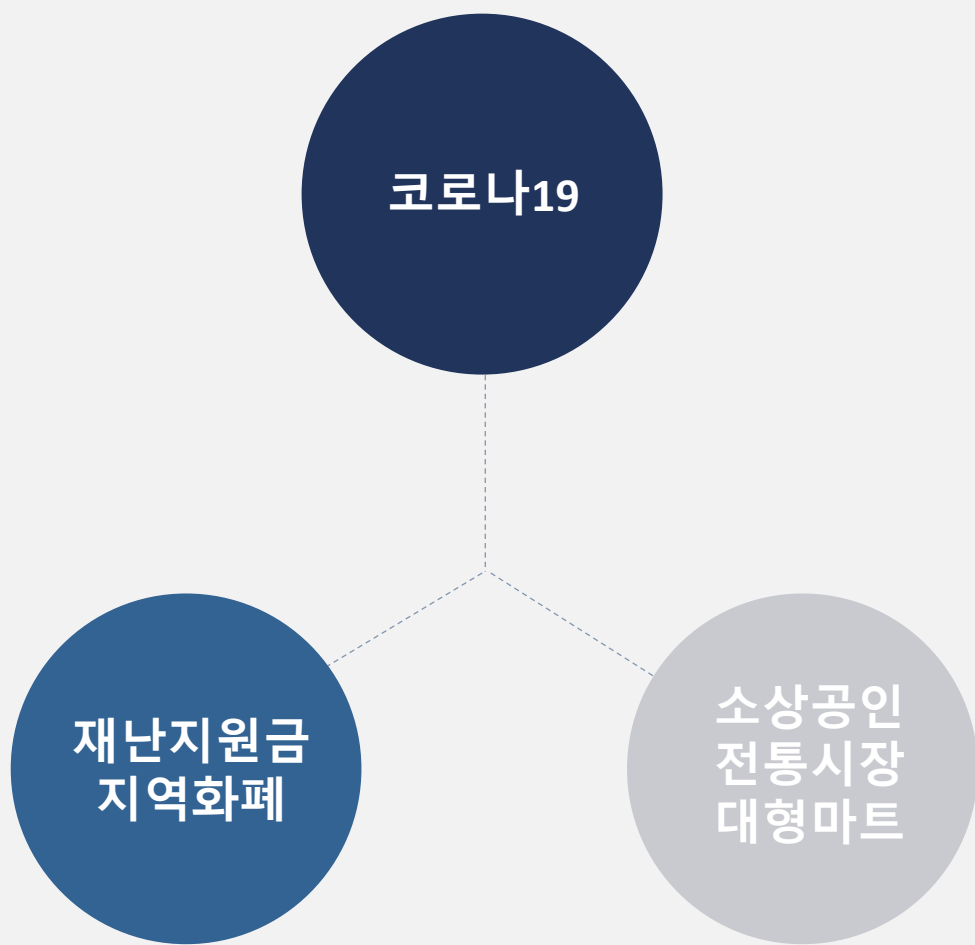


2020년도 대형마트(이마트)매출액



- 이마트 2020년 매출 그래프를 살펴보면, 코로나 발생 직후인 1월 매출이 급락했고 6월부터 매출이 상승하여 9월에 고점을 찍고 하락한 뒤 11월부터 다시 증가하는 경향을 보입니다.
- 20년도 5월 1차 재난지원금이 지급됨에 따라 소상공인 & 전통시장 BSI 그래프와 반대의 성향을 보이는데 이러한 변동이 발생한 배경에는 대형마트와 소상공인 & 전통시장 간의 재난지원금 사용 가능 여부와 관련된 요인이 있을 것으로 예상됩니다.

5. 프로젝트 결과



결론

재난지원금이 제공됨에 따라 소상공인, 전통시장 소비, 매출이 증가하였으나 지속적으로 영향을 미치지 는 못 하였다는 결론을 도출하였습니다.

프로젝트 역할분담

강예진

조원1

- 데이터 수집, 전처리, 분석, 시각화(경기도 지역화폐 이용현황, 가맹점 지도)
- 가상환경 구축 메뉴얼 작성
- Streamlit으로 프로젝트 웹 표출
- 발표

권용현

조원2

- 데이터 수집, 전처리, 분석, 시각화(소상공인 & 전통시장 경기체감, 골목발달상권)
- Streamlit 메뉴얼 작성
- Streamlit으로 프로젝트 웹 표출

권지현

조장

- 기획안 작성
- 데이터 수집, 전처리, 분석, 시각화(경기도 코로나 현황, Emart 매출액, 가계소비지출 현황)
- 프로젝트 diagram 작성
- Streamlit으로 프로젝트 웹 표출
- 포트폴리오 작성

감사합니다

