



뽕뽕콩

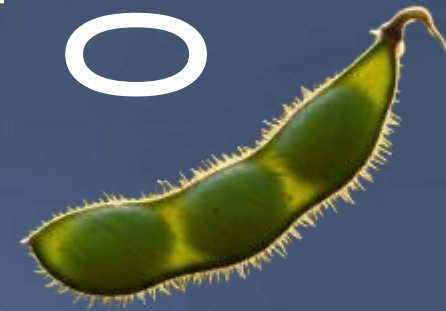


Table of contents

Part 1 프로젝트 개요

Part 2 동영상 - 코드

Part 3 Q & A



서울 생활 도우미

1-1. 서울지도와 여행지 표시
- 맛집, 관광명소, 교통정보 소개

1-2. 챗봇 인터페이스
- 서울에 대한 다양한 답변 제공

1-3. 네이버 블로그 바탕 추가 설명
- 검색한 키워드와 관련하여
경험담 요약본 제공



Google Map API

1

- 업로드한 csv 데이터를 Geocoding 하여 주소 -> 위도, 경도 변환

- 서울시 근방 지도 표시 및 선택한 장소에 대하여 마커 표시와 함께 팝업창에 추가 설명 표시



Hyper CLOVA

2

- Hyper CLOVA 챗봇은 서울의 관광, 맛집, 교통, 문화에 관한 정보를 친절하고 직관적으로 제공하도록 프롬프트 설정

- 사용자가 입력한 질문에 서울에 관련된 유용한 팁이나 장소 추천을 제공.



Naver 검색 API

3

- Naver 검색 API를 활용하여 검색 키워드와 관련도가 높은 블로그 데이터 크롤링

- 크롤링한 데이터를 Clova studio 스킴을 이용하여 3줄 요약

1. 지도 표시



서울시 음식점
Data.csv



서울시 관광지
Data.csv



Google Map API

```
1 # Import the Google Maps API module from the Google Cloud SDK
2 from google.cloud.maps import MapsClient
3
4 # Create a MapsClient object
5 client = MapsClient()
6
7 # Define the location to search for
8 location = "Seoul, South Korea"
9
10 # Search for the location
11 results = client.search(location)
12
13 # Print the results
14 for result in results:
15     print(result.location_name)
```



Google Map

2. 챗봇 & 블로그 크롤링 요약



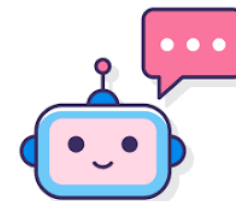
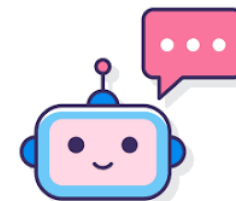
Hyper CLOVA



Naver검색 API



요약



서울특별시 공공데이터 (관광 음식, 관광 명소)

파일데이터명	서울특별시_관광 음식_20210904		
분류체계	문화체육관광 - 관광	제공기관	서울특별시

미리보기

닫힘 -

Sheet

Open API

필드명

필터선택

검색명

내려받기(CSV)

내려받기(JSON)

조회

고유번호	언어	상호명	콘텐츠URL	주소	신
38046	ja	リッチモンド製菓店:ザ・ビュ...	https://japanese.visitseoul.net/...	ソウル市ヨンドンポ(永登浦)...	07335 서울
1692	zh-TW	Apple House		137-811 首爾市瑞草區盤浦洞978	06548 首爾市
7140	en	Hanu Hanmari	https://english.visitseoul.net/re...	07779 Hwagok 1(il)-dong, Gang...	07779 65, Gan
7142	zh-TW	一隻雞牛	https://tchinese.visitseoul.net/r...	07779 首爾市江西區禾谷1洞 91...	07779 首爾市
47732	en	Seoul Rabbit	https://english.visitseoul.net/re...	서울 종로구 체부동 37-1	03041 1F, 19-1
47734	zh-CN	Seoul Rabbit	https://chinese.visitseoul.net/r...	서울 종로구 체부동 37-1	03041 首尔市
47735	zh-TW	Seoul Rabbit	https://tchinese.visitseoul.net/r...	서울 종로구 체부동 37-1	03041 首爾市
32418	zh-CN	晚上也能吃到的韩国代表性夜宵...			
47733	ja	Seoul Rabbit	https://japanese.visitseoul.net/...	서울 종로구 체부동 37-1	03041 ソウル
47731	ko	서울래빗	https://korean.visitseoul.net/re...	서울 종로구 체부동 37-1	03041 서울 종
1070	zh-TW	Dublin Terrace	https://tchinese.visitseoul.net/r...	135-908 首爾市江南區驛三洞63...	06130 首爾市
283	ja	トニン(遇仁)市場	https://japanese.visitseoul.net/...	110-043 ソウル市鍾路区通仁洞...	03036 ソウル
1039	zh-CN	Salon de The	https://chinese.visitseoul.net/r...	100-721 서울 중구 소공동 1	04533 首尔中
1098	en	Gaemijip	https://english.visitseoul.net/re...	135-888 528-4, Sinsa-dong, Gan...	06028 110, Ap
1099	ja	ケミジブ	https://japanese.visitseoul.net/...	135-888 ソウル市江南区新沙洞...	06028 ソウル
1048	ja	Queens Park 清潭店	https://japanese.visitseoul.net/...	135-954 ソウル市江南区清潭洞...	06014 ソウル
1102	ja	コービーバンアッカン	https://japanese.visitseoul.net/...	110-210 ソウル市鍾路区花洞 10...	03053 ソウル

※ sheet는 최대 1,000건까지 노출됩니다. 전체 데이터는 CSV파일을 내려받아 확인해 주시기 바랍니다.

언어 항목을 필터링하여 csv 다운로드 / 한국어만 사용
한국어(ko) 영어(en) 중국어(zh-CN) 대만(zh-TW)일본어(jn)

파일데이터명	서울특별시_관광 명소_20210904		
분류체계	문화체육관광 - 관광	제공기관	서울특별시

미리보기

닫힘 -

Sheet

Open API

필드명

언어

검색명

ko

내려받기(CSV)

내려받기(JSON)

조회

고유번호	언어	상호명	콘텐츠URL	주소	신
15338	ko	1898 명동성당	https://korean.visitseoul.net/at...	100-809 서울 중구 명동2가 1-1	04537 서울 중
252	ko	63스퀘어	https://korean.visitseoul.net/at...	150-763 서울 영등포구 여의도동 60	07345 서울 영
23532	ko	SeMA 병커	https://korean.visitseoul.net/at...	150-010 서울 영등포구 여의도동 2-6	07327 서울 영
46812	ko	THEO 락코	https://korean.visitseoul.net/at...	서울 강남구 삼성동 124-16	06156 서울 강
43458	ko	JS슈즈디자인연구소	https://korean.visitseoul.net/at...	서울 성동구 성수동2가 350	04775 서울 성
28338	ko	ECC(Ewha Campus Complex)	https://korean.visitseoul.net/at...	120-750 서울 서대문구 대현동 11-1	03760 서울 서
29561	ko	aT센터	https://korean.visitseoul.net/at...	서울 서초구 양재동 232	06774 서울 서
37879	ko	HYBE INSIGHT (하이프 뮤지엄)	https://korean.visitseoul.net/at...	서울 용산구 한강로3가 65-9	04389 서울 용
37044	ko	KCDF갤러리	https://korean.visitseoul.net/at...	서울 종로구 관훈동 182-2	03145 서울 종
2885	ko	PKM갤러리	https://korean.visitseoul.net/at...	110-230 서울 종로구 삼청동 157-83	03049 서울 종
24732	ko	SJ콘서트홀레	https://korean.visitseoul.net/at...	135-819 서울 강남구 논현동 97-22	06054 서울 강
44032	ko	만나분식	https://korean.visitseoul.net/at...		06284 서울 강
1005	ko	가무	https://korean.visitseoul.net/at...		04536 서울 중
29238	ko	서울메트로미술관	https://korean.visitseoul.net/at...	서울 종로구 직선동 81-1	03170 서울 종
24706	ko	만나분식	https://korean.visitseoul.net/at...		03039 서울 종
1961	ko	검재장선미술관	https://korean.visitseoul.net/at...	157-801 서울 강서구 가양1동 243-1	07522 서울 강
1151	ko	경교장	https://korean.visitseoul.net/at...	110-746 서울 종로구 평동 108-1	03181 서울 종

※ sheet는 최대 1,000건까지 노출됩니다. 전체 데이터는 CSV파일을 내려받아 확인해 주시기 바랍니다.

음식 : 고유번호/언어/상호명/URL/주소/신주소/전화번호/운영시간/교통정보/대표메뉴

관광 명소 : 고유번호/언어/상호명/URL/주소/신주소/전화번호/운영시간/교통정보/휴무일/장애인 편의시설/태그

서울특별시 공공데이터 (관광 음식, 관광 명소)

구유번호	구유번호	언어	상호명	콘텐츠URL	주소	1.2 Latitude	1.2 Longitude	구유번호	구유번호	언어	상호명	콘텐츠URL	주소	신주소	1.2 Latitude	1.2 Longitude
1	47731	ko	서울라빗	https://korean.vis...	서울 종로구 체부동 37-1	37.5778073	126.9717504	1	15338	ko	1898 영등성당	https://korean.v...	100-809 서울 중구 명동2가 1-1	04537 서울 중구 명동길 74 (명동2가, 영등성당)	37.5633245	126.9872338
2	45792	ko	업스탠딩 Upstanding coff...	https://korean.vis...	서울 용산구 용산동2가 1-90	37.5449265	126.9849717	2	252	ko	63스퀘어	https://korean.v...	150-763 서울 영등포구 여의도동 60	07345 서울 영등포구 63로 50 (여의도동, 63한화생명빌)	37.5197898	126.9400579
3	1095	ko	개미집	https://korean.vis...	135-888 서울 강남구 신사동 528-4	37.5229663	127.0209482	3	23532	ko	SeMA 벙커	https://korean.v...	150-010 서울 영등포구 여의도동 2-6	07327 서울 영등포구 의사당대로 지하 101 (여의도역)	37.5235991	126.9222137
4	5780	ko	과미19	https://korean.vis...	서울 강남구 신사동 564-12 524-1	37.5209387	127.0272846	4	46812	ko	THEO 티오	https://korean.v...	서울 강남구 삼성동 124-16	06156 서울 강남구 봉은사로68길 55-3 (삼성동, BANG2)	37.5091073	127.0528147
5	47549	ko	하우스오브바미날 연회점	https://korean.vis...	서울 서대문구 연희동 190-17	37.5667961	126.9286826	5	43458	ko	JS슈즈디자인연구소	https://korean.v...	서울 성동구 성수동2가 350	04775 서울 성동구 성덕정길 72-4 (성수동2가) 2층	37.5377494	127.0516872
6	46950	ko	전미식당	https://korean.vis...	서울 마포구 공덕동 105-127	37.5505895	126.9557918	6	28338	ko	ECC(Ewha Campus Complex)	https://korean.v...	120-750 서울 서대문구 대현동 11-1	03760 서울 서대문구 이화여대길 52 (대현동, 이화여자)	37.5618588	126.9468339
7	5515	ko	두꺼비정육점식당	https://korean.vis...	135-841 서울 강남구 대치동 906-21	37.5015119	127.0530636	7	29561	ko	aT센터	https://korean.v...	서울 서초구 양재동 232	06774 서울 서초구 강남대로 27 (양재동, aT센터)	37.4683508	127.0390213
8	30871	ko	발우공양	https://korean.vis...	110-170 서울 종로구 견지동 71	37.5738873	126.9831949	8	37879	ko	HYBE INSIGHT (하이브 뮤지...	https://korean.v...	서울 용산구 한강로3가 65-9	04389 서울 용산구 한강대로 42 (한강로3가, 용산트레...	37.5245625	126.9641875
9	6948	ko	산골막국수	https://korean.vis...	110-826 서울 종로구 숭인동 278	37.5728996	127.0169835	9	37044	ko	KCDF갤러리	https://korean.v...	서울 종로구 관훈동 182-2	03145 서울 종로구 인사동11길 8 (관훈동, 공예종합유...	37.5740155	126.9841199
10	46871	ko	중로 은행나무집	https://korean.vis...	서울 종로구 관철동 33-3 2~3층	37.5686981	126.9880727	10	2885	ko	PKM갤러리	https://korean.v...	110-230 서울 종로구 삼청동 157-83	03049 서울 종로구 삼청로7길 40 (삼청동, 피케미갤 갤...	37.5840668	126.9800992
11	5226	ko	사루비아	https://korean.vis...	135-889 서울 강남구 신사동 542-3	37.5200384	127.0236138	11	24732	ko	SJ콘서트홀레	https://korean.v...	135-819 서울 강남구 논현동 97-22	06054 서울 강남구 언주로148길 5 (논현동)	37.5207671	127.0360264
12	12291	ko	서강잡데기	https://korean.vis...	마포구 서교동 383-1	37.5526895	126.9124804	12	44032	ko	만나분석	https://korean.v...		06284 서울 강남구 삼성로 212 (대치동, 은마아파트)	37.4974576	127.0653265
13	47245	ko	취득루	https://korean.vis...	서울 성동구 성수동1가 656-1661	37.5426033	127.0485903	13	1005	ko	가무	https://korean.v...		04536 서울 중구 명동4길 16	37.5629911	126.9836165
14	46955	ko	농동미나리	https://korean.vis...	서울 용산구 한강로2가 148-6	37.530083	126.9707874	14	29238	ko	서울메트로미술관	https://korean.v...	서울 종로구 적선동 81-1	03170 서울 종로구 사직로 지하 130 (적선동, 3호선 경...	37.574115	126.9743671
15	47062	ko	통영굴밥	https://korean.vis...	서울 종로구 적선동 80	37.5753941	126.9732554	15	24706	ko	만나분석	https://korean.v...		03039 서울 종로구 필운동14길 4 1층	37.5773634	126.9690974
16	47041	ko	무예	https://korean.vis...	서울 종로구 계동 76	37.5808332	126.9866653	16	1961	ko	경제정선미술관	https://korean.v...	157-801 서울 강서구 가양1동 243-1	07522 서울 강서구 양천로47길 36 (가양동, 경제정선...	37.5722129	126.8384499
17	4549	ko	형일집	https://korean.vis...	122-860 서울 은평구 불광동 310-11 연서	37.6192823	126.9214788	17	1151	ko	경교장	https://korean.v...	110-746 서울 종로구 평동 108-1	03181 서울 종로구 새문안로 29 (경교장)	37.5682936	126.9680453
18	6805	ko	자하손만두	https://korean.vis...	110-817 서울 종로구 부암동 245-2	37.5931183	126.9660427	18	2012	ko	경인미술관	https://korean.v...		03146 서울 종로구 인사동10길 11-4	37.574506	126.9856172
19	6599	ko	갯마을낙지	https://korean.vis...	138-862 서울 송파구 잠실동 197-7	37.5089625	127.0798171	19	18897	ko	금박연	https://korean.v...		03056 서울 종로구 북촌로12길 24-12	37.5826313	126.9861842
20	2570	ko	예당	https://korean.vis...	서울 강남구 신사동 628-16 628-20	37.522619	127.0325255	20	43607	ko	상진다방	https://korean.v...	서울 영등포구 문래동2가 14-84	07290 서울 영등포구 도림로133길 9 (문래동2가, 상진...	37.5133808	126.8945146
21	13056	ko	최탁	https://korean.vis...	100-804 서울특별시 중구 남창동 34-34	37.5599063	126.9770883	21	46608	ko	경국사	https://korean.v...	서울 성북구 정릉동 751	02705 서울 성북구 보국로113-10 (정릉동, 경국사)	37.6146726	127.0049176
22	4826	ko	이태원숯불구이	https://korean.vis...	140-863 서울 용산구 이태원동 44-50	37.5328486	126.9907927	22	2084	ko	가나아트센터	https://korean.v...	110-846 서울 종로구 평창30길 28 (가나아트센터)	03004 서울 종로구 평창30길 28 (가나아트센터)	37.6121861	126.9750972
23	47560	ko	연희동 라이브이 본점	https://korean.vis...	서울 서대문구 연희동 132-27	37.5685534	126.9304667	23	40770	ko	국립가상박물관	https://korean.v...	서울 종로구 숭실동 1-1	03178 서울 종로구 숭실길 52 (숭실동, 국립가상박물관)	37.5714092	126.9662955
24	47036	ko	법원	https://korean.vis...	서울 종로구 계동 140-51	37.5792594	126.987371	24	1647	ko	국립민속박물관	https://korean.v...	110-820 서울 종로구 세종로 1-1 1-1	03045 서울 종로구 삼청로 37 (세종로, 국립민속박물관)	37.5816456	126.9789948
25	11581	ko	종도	https://korean.vis...	100-120 서울 중구 정동 1-54	37.5677077	126.9740319	25	10611	ko	가미분석	https://korean.v...		03767 서울 서대문구 이화여대8길 2 무궁화상가아파트	37.5586501	126.9459568
26	12120	ko	열차집	https://korean.vis...	110-160 서울 종로구 공평동 130-1	37.5717032	126.9827896	26	46873	ko	가봉루	https://korean.v...	서울 종로구 세종로 166-1	03183 서울 종로구 세종대로23길 3 (세종로)	37.5715064	126.9762278
27	27461	ko	시루케이크	https://korean.vis...	서울 마포구 합정동 355-8	37.5467925	126.9182748	27	27812	ko	가원공방	https://korean.v...		03138 서울 종로구 돈화문로6나길 45 (봉익동)	37.5723512	126.9923053
28	12702	ko	차크라	https://korean.vis...	140-884 서울특별시 용산구 한남동 28-9	37.5344972	127.0096897	28	38858	ko	가을단풍길(월드컵로)	https://korean.v...	서울 마포구 상암동 1730	03906 서울 마포구 월드컵로42길 9-1 (가로판매대07)	37.579489	126.8809456
29	4073	ko	시실리	https://korean.vis...	137-805 서울 서초구 반포동 90-11 90-11	37.4973269	126.998492	29	18912	ko	가회민화박물관	https://korean.v...	110-260 서울 종로구 가회동 28	03056 서울 종로구 북촌로 52 (가회동) 가회민화박물관	37.5814789	126.9852404
30	45741	ko	클럽 에스프레소	https://korean.vis...	서울 종로구 부암동 257-1	37.5926553	126.9659796	30	29715	ko	개미슈퍼	https://korean.v...	서울 용산구 서계동 39	04303 서울 용산구 청파로85길 31 (서계동)	37.5525658	126.966442


다운로드 받은 파일(csv)의 주소(신주소)를 Geocoding을 이용하여 위도(Latitude), 경도(Longitude)데이터 추가

The screenshot displays a video player interface with a dark theme. The main content area shows a VS Code editor window with a file named `project05.py`. The script is a Python program that uses `pandas`, `streamlit`, and `requests` to fetch data from Naver Cloud. It defines API keys for Google Maps and Naver Cloud, reads CSV files for restaurants and attractions, and then iterates through them to extract location information. The terminal at the bottom shows the command `streamlit run project05.py` being executed. The video player controls at the bottom indicate the video is at 00:03 and has a total duration of 01:03. The video title is "경과원시 네이버 클라우드_독독공2조_동영상_2024-12-04".

```
1 #
2 # 라이브러리 불러오기
3 import pandas as pd
4 import streamlit as st
5 import streamlit.components.v1 as components
6 import requests
7 import json
8 import re
9
10 # Google Maps API Key
11 API_KEY = "AIzaSyC17u03UNBuaMgT2HwCTcFouCYfN0K6g"
12
13 # Naver Cloud API 키
14 CLOVA_HOST = "https://clovastudio.stream.ntruss.com"
15 CLOVA_API_KEY = "HT40PJU2H6G1ZTcND31Y7vV12pdx6f37dclHggQ82VhY8GAScWR4ysx75rc8z"
16 CLOVA_API_KEY_PRIMARY_VAL = "ucrt6v1af04vWtXf2g0Kc1ayy0AfeVdknauu"
17 CLOVA_RESULTS_ID = "5d5b55cd/cudasLab/swe221eff5d3"
18
19 # CSV 파일 불러오기
20 df_restaurants = pd.read_csv("restaurants1.csv", encoding="utf-8")
21 df_attractions = pd.read_csv("attraction1.csv", encoding="utf-8")
22
23 #-----
24 # 거리 및 좌표 정보 위치 정보를 추출
25 restaurants_locations = []
26 attractions_locations = []
27
28 for idx, row in df_restaurants.iterrows():
29     lat, lon = row["Latitude"], row["Longitude"]
30     restaurant_location = {
31         "name": row["상표명"],
32         "address": row["주소"],
33         "hours": row["영업시간"],
34         "transport": row["교통정보"],
35         "menu": row["대표메뉴"],
36         "lat": row["Latitude"],
37         "lon": row["Longitude"],
38     }
39     restaurants_locations.append(restaurant_location)
40
41 for idx, row in df_attractions.iterrows():
42     lat, lon = row["Latitude"], row["Longitude"]
43     attraction_location = {
44         "name": row["이름"],
45         "address": row["주소"],
46         "hours": row["영업시간"],
47         "transport": row["교통정보"],
48         "menu": row["대표메뉴"],
49         "lat": row["Latitude"],
50         "lon": row["Longitude"],
51     }
52     attractions_locations.append(attraction_location)
53
54 # Streamlit UI
55 st.title("Naver Cloud API Test")
56 st.subheader("Restaurant and Attraction Data")
57
58 # Restaurant Data
59 st.write("Restaurant Data")
60 st.json(restaurants_locations)
61
62 # Attraction Data
63 st.write("Attraction Data")
64 st.json(attractions_locations)
```

00:03 01:03

경과원시 네이버 클라우드_독독공2조_동영상_2024-12-04

3. 코드 >  project09.py > ...

```
1  #-----
2  # 라이브러리 불러오기
3  import pandas as pd
4  import streamlit as st
5  import streamlit.components.v1 as components
6  import requests
7  import json
8  import re
9
10 # Google Maps API Key
11 API_KEY = "AIzaSyCI7uo3qNBuoUgT2WkeCTcFbdCYfWQK6zg"
12
13 # Naver Clova API 정보
14 CLOVA_HOST = 'https://clovastudio.stream.ntruss.com'
15 CLOVA_API_KEY = 'NTA0MjU2MWZlZTcxNDJiY7uVi2pdxbxf37dd1HgqQ82zYkY8GA5zNRHYsx7Src8z'
16 CLOVA_API_KEY_PRIMARY_VAL = 'uG6Ia31mjoAOMIxeFgp0sit1ayyOafnVWkcnaawL'
17 CLOVA_REQUEST_ID = '5d59553c47c64e31a5759e221efff533'
18
19 # CSV 파일 읽기
20 df_restaurants = pd.read_csv('restaurants1.csv', encoding='utf-8')
21 df_attractions = pd.read_csv('attraction1.csv', encoding='utf-8')
22
23 #-----
```

3. 코드 > project09.py > ...

```
22
23 #-----
24 # 맛집 및 관광 명소 위치 정보를 저장
25 restaurants_locations = []
26 attractions_locations = []
27
28 for idx, row in df_restaurants.iterrows():
29     lat, lon = row['Latitude'], row['Longitude']
30     restaurant_location = {
31         "name": row["상호명"],
32         "address": row["주소"],
33         "hours": row["운영시간"],
34         "transport": row["교통정보"],
35         "menu": row["대표메뉴"],
36         "lat": row["Latitude"],
37         "lon": row["Longitude"],
38     }
39     restaurants_locations.append(restaurant_location)
40
41 for idx, row in df_attractions.iterrows():
42     lat, lon = row['Latitude'], row['Longitude']
43     attraction_location = {
44         "name": row["상호명"],
45         "address": row["주소"],
46         "hours": row["운영시간"],
47         "transport": row["교통정보"],
48         "tag": row["태그"],
49         "lat": row["Latitude"],
50         "lon": row["Longitude"],
51     }
52     attractions_locations.append(attraction_location)
53
54 #-----
```

3. 코드 > project09.py > ...

```

54  #-----
55  # 서울시 챗봇 실행 클래스
56  class CompletionExecutor:
57      def __init__(self, host, api_key, api_key_primary_val, request_id):
58          self._host = host
59          self._api_key = api_key
60          self._api_key_primary_val = api_key_primary_val
61          self._request_id = request_id
62
63      def execute(self, completion_request):
64          headers = {
65              'X-NCP-CLOVASTUDIO-API-KEY': self._api_key,
66              'X-NCP-APIGW-API-KEY': self._api_key_primary_val,
67              'X-NCP-CLOVASTUDIO-REQUEST-ID': self._request_id,
68              'Content-Type': 'application/json; charset=utf-8',
69              'Accept': 'text/event-stream'
70          }
71          response = requests.post(
72              self._host + '/testapp/v1/chat-completions/HCX-003',
73              headers=headers,
74              json=completion_request,
75              stream=True
76          )
77          full_response = ""
78          for line in response.iter_lines():
79              if line:
80                  try:
81                      data = json.loads(line.decode("utf-8")[5:]) # `data:` 부분 제거
82                      if "message" in data and data["message"]["role"] == "assistant":
83                          content = data["message"]["content"]
84                          if full_response != content: # 중복된 내용인지 확인
85                              full_response += content # 중복이 아닌 경우만 추가
86
87                  except json.JSONDecodeError:
88                      continue
89          return full_response.strip() # 중복 방지를 위해 출력 문자열을 정리
90  #-----

```

3. 코드 > project09.py > ...

```
90 # Streamlit 페이지 설정
91 st.set_page_config(layout="wide")
92
93 # 화면 좌우 분할
94 col1, col2 = st.columns([3, 2])
95
96 # <<- 좌측 지도 및 데이터프레임 표시
97 with col1:
98     # 지도 섹션
99     st.title("서울시 여행 지도")
100     st.write("서울시에서 놀러갈 맛집, 놀 곳, 핫플들을 지도에서 확인하세요.")
101
102     # 장소 선택 박스 (맛집)
103     selected_restaurant_name = st.selectbox("맛집을 선택하세요", df_restaurants['상호명'])
104     selected_restaurant = df_restaurants[df_restaurants['상호명'] == selected_restaurant_name].iloc[0]
105     selected_restaurant_address = selected_restaurant['주소']
106
107     # 장소 선택 박스 (관광 명소)
108     selected_attraction_name = st.selectbox("관광 명소를 선택하세요", df_attractions['상호명'])
109     selected_attraction = df_attractions[df_attractions['상호명'] == selected_attraction_name].iloc[0]
110     selected_attraction_address = selected_attraction['주소']
```



```

3. 코드 > project09.py > ...
111     selected_attraction_address = selected_attraction["주소"]
112
113     # Google Maps HTML 생성
114     html_code = f"""
115     <!DOCTYPE html>
116     <html>
117     <head>
118         <title>서울시 여행 지도</title>
119         <script src="https://maps.googleapis.com/maps/api/js?key={API_KEY}"></script>
120         <script>
121             function initMap() {{
122                 var mapCenter = {{lat: 37.5665, lng: 126.9780}}; // 서울 중심
123                 var map = new google.maps.Map(document.getElementById('map'), {{
124                     zoom: 12,
125                     center: mapCenter
126                 }});
127
128                 // 맛집 마커 추가
129                 var restaurantLocation = {{lat: {selected_restaurant["Latitude"]}, lng: {selected_restaurant["Longitude"]}};
130                 var restaurantMarker = new google.maps.Marker({{
131                     map: map,
132                     position: restaurantLocation,
133                     title: '{selected_restaurant_name}',
134                     icon: {{
135                         url: 'http://maps.google.com/mapfiles/ms/icons/red-dot.png',
136                         scaledSize: new google.maps.Size(60, 60),
137                     }}
138                 }});
139                 var restaurantInfowindow = new google.maps.InfoWindow({{
140                     content: "<b>" + '{selected_restaurant_name}' + "</b><br>" +
141                         "주소: " + '{selected_restaurant_address}' + "<br>" +
142                         "운영시간: " + '{selected_restaurant["운영시간"]}' + "<br>" +
143                         "교통정보: " + '{selected_restaurant["교통정보"]}' + "<br>" +
144                         "대표메뉴: " + '{selected_restaurant["대표메뉴"]}'
145                 }});

```

3. 코드 > project09.py > ...

```

146 restaurantMarker.addListener("click", function() {{
147     restaurantInfowindow.open(map, restaurantMarker);
148 }});
149 map.setCenter(restaurantLocation);
150
151 // 관광 명소 마커 추가
152 var attractionLocation = {{lat: {selected_attraction["Latitude"]}, lng: {selected_attraction["Longitude"]}}};
153 var attractionMarker = new google.maps.Marker({{
154     map: map,
155     position: attractionLocation,
156     title: '{selected_attraction_name}',
157     icon: {{
158         url: 'http://maps.google.com/mapfiles/ms/icons/blue-dot.png',
159         scaledSize: new google.maps.Size(60, 60),
160     }}
161 }});
162 var attractionInfowindow = new google.maps.InfoWindow({{
163     content: "<b>" + '{selected_attraction_name}' + "</b><br>" +
164         "주소: " + '{selected_attraction_address}' + "<br>" +
165         "운영시간: " + '{selected_attraction["운영시간"]}' + "<br>" +
166         "교통정보: " + '{selected_attraction["교통정보"]}' + "<br>" +
167         "태그: " + '{selected_attraction["태그"]}'
168 }});
169 attractionMarker.addListener("click", function() {{
170     attractionInfowindow.open(map, attractionMarker);
171 }});
172 map.setCenter(attractionLocation);
173 }}
174 </script>
175 </head>
176 <body onload="initMap()">
177     <div id="map" style="height: 500px; width: 100%;"></div>
178 </body>
179 </html>
180 ""
181 components.html(html_code, height=700)
182

```

3. 코드 > project09.py > ...

```
238
239 # 서울 챗봇 표시
240 st.title("서울 챗봇 🗺️")
241 st.write("안녕! 서울에 대해 무엇이든 물어보세요. 질문을 입력하면 친절하게 답변해줄게요!")
242 system_prompt = (
243     "- 나는 서울의 관광, 맛집, 교통, 문화에 대해 잘 아는 친절한 안내자야.\n"
244     "- 반말을 사용해서 질문에 쉽고 명확하게 답변해줄게.\n"
245     "- 사용자에게 서울에서 갈 만한 장소, 맛집 추천, 대중교통 정보 등을 직관적으로 알려줘.\n"
246     "- 답변은 서울에서 유용한 팁이나 실제 경험을 바탕으로 사례를 들어서 설명해.\n"
247     "- 처음 답변시 '안녕! 서울에 대해 뭐든 물어봐. 내가 친절히 알려줄게!'로 시작해."
248 )
249
```

3. 코드 > project09.py > ...

```
248     )
249
250 # 사용자 입력 및 버튼
251 user_input = st.text_input("질문을 입력하세요:", placeholder="예: 명동 칼국수 맛집, 서울 산책길 명소")
252 submit_button = st.button("질문 보내기")
253
254 # 질문 처리 및 응답 출력
255 if submit_button and user_input.strip():
256     # 프롬프트 구성
257     preset_text = [
258         {"role": "system", "content": system_prompt},
259         {"role": "user", "content": user_input}
260     ]
261
262     request_data = {
263         'messages': preset_text,
264         'topP': 0.6,
265         'topK': 0,
266         'maxTokens': 512,
267         'temperature': 0.5,
268         'repeatPenalty': 5.0,
269         'stopBefore': [],
270         'includeAiFilters': True,
271         'seed': 0
272     }
273
274     # 서울 챗봇 실행
275     completion_executor = CompletionExecutor(
276         host=CLOVA_HOST,
277         api_key=CLOVA_API_KEY,
278         api_key_primary_val=CLOVA_API_KEY_PRIMARY_VAL,
279         request_id=CLOVA_REQUEST_ID
280     )
```


3. 코드 > project09.py > ...

```

183     # 맛집 리스트 표시
184     st.subheader("맛집 리스트")
185     st.write(df_restaurants[['상호명', '주소', '운영시간', '교통정보', '대표메뉴']])
186
187     # 관광 명소 리스트 표시
188     st.subheader("관광 명소 리스트")
189     st.write(df_attractions[['상호명', '주소', '운영시간', '교통정보', '태그']])
190
191     # -----
192     # ->> 무촉 챗봇 표시(크롤링 데이터)
193     with col2:
194         # 네이버 검색 API를 통한 데이터 크롤링
195         def search_blog(query, display=4):
196             encoded_query = query
197             url = f"https://openapi.naver.com/v1/search/blog.json?query={encoded_query}&display={display}"
198             headers = {
199                 "X-Naver-Client-Id": "o58Yv2p1v5JR4CrzUhWe",
200                 "X-Naver-Client-Secret": 'FoH00xQQ1N'
201             }
202             response = requests.get(url, headers=headers)
203             if response.status_code == 200:
204                 items = response.json().get('items', [])
205                 for item in items:
206                     # HTML 태그 제거 (<b>, </b>)
207                     item['title'] = re.sub(r'<\/?b>', '', item['title'])
208                     item['description'] = re.sub(r'<\/?b>', '', item['description'])
209                 return items
210             else:
211                 st.error(f"Failed to search blog. Status code: {response.status_code}")
212                 return []

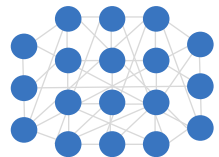
```

3. 코드 > project09.py > ...

```
211
212
213 # 크롤링 데이터 요약
214 def summarize_clova(text_to_summarize):
215     completion_executor = CompletionExecutor(
216         host='https://clovastudio.apigw.ntruss.com',
217         api_key=CLOVA_API_KEY,
218         api_key_primary_val=CLOVA_API_KEY_PRIMARY_VAL,
219         request_id=CLOVA_REQUEST_ID
220     )
221
222     # 크롤링 데이터 요약 프롬프트 및 파라미터
223     request_data = {
224         "messages": [
225             {"role": "user", "content": f"다음 글을 3줄로 요약해줘. 친절하게 대답해줘.: {text_to_summarize}"}
226         ],
227         "topP": 0.6,
228         "topK": 0,
229         "maxTokens": 150,
230         "temperature": 0.5,
231         "repeatPenalty": 5.0,
232         "includeAiFilters": True,
233         "seed": 0
234     }
235     # 텍스트 요약 요청 및 응답 처리
236     summarization_response = completion_executor.execute(request_data)
237     return summarization_response if summarization_response else 'Summarization Error'
238
```

3. 코드 > project09.py > ...

```
280     )
281     # 챗봇 답변 대기
282     with st.spinner("답변 생성 중..."):
283         response = completion_executor.execute(request_data)
284
285     # 챗봇 답변 출력
286     if response:
287         st.success("답변이 생성되었습니다!")
288         st.write(response)
289         st.markdown("<hr style='border: 1px solid gray; margin: 15px 0;'", unsafe_allow_html=True)
290         st.title("네이버 블로그 검색결과")
291         blog_items = search_blog(query=user_input)
292         for item in blog_items:
293             st.subheader(item['title'])
294
295     # 블로그 요약 생성 및 출력
296     summary = summarize_clova(item['description'])
297     st.write("요약:", summary)
298 else:
299     st.error("답변 생성에 실패했습니다. 다시 시도해주세요.")
300 elif submit_button:
301     st.warning("질문을 입력해주세요.")
302
```



류근우

조장 / PPT 제작 및 발표

장수진

Data 수집 / Google Map API

변소윤

Hyper Clova 챗봇

정용우

네이버 블로그 크롤링 및 요약

Q&A

