

Colab_to_Streamlit_StepByStep_README

폐의약품 수거 약국 찾기 실습 가이드

Colab 데이터 가공 → VS Code Streamlit 연결



1

Colab 단계: 데이터 가공

1) Colab 노트북 열기

- <https://colab.research.google.com/> 에서 새 노트북 열기

2) CSV 파일 업로드

```
from google.colab import files
uploaded = files.upload()
```

3) pandas로 데이터 불러오기

```
import pandas as pd

file_path = "TOTAL_sp_ta_medical_ifo.csv"
df = pd.read_csv(file_path, encoding="euc-kr")
print(df.head())
print(df.columns)
print(df.shape)
print(df.info())
```

4) '약국' + '천안서북구'만 추출

```
df_pharmacy = df[df['종별코드명'].str.contains('약국', na=False)]
df_seobuk = df_pharmacy[df_pharmacy['시군구명'] == '천안서북구']
df_seobuk.to_csv("cheonan_seobuk_pharmacy.csv", index=False, encoding="utf-8-sig")
```

5) 위도/경도 빈칸 확인 & 저장

```
missing_coords = df_seobuk[df_seobuk['위도'].isnull() | df_seobuk['경도'].isnull()]
print(missing_coords)
filtered = df_seobuk.dropna(subset=['위도', '경도'])
filtered.to_csv("cheonan_seobuk_pharmacy_clean.csv", index=False, encoding="utf-8-sig")
```

📌 6) 수거약품목 컬럼 랜덤 추가

```
import random

df = pd.read_csv("cheonan_seobuk_pharmacy_clean.csv", encoding="utf-8-sig")

categories = [
    "해열제", "진통제", "소염제", "항생제", "소화제",
    "심혈관계", "호흡기계", "내분비계", "신경계",
    "안이비인후과용", "외용제", "건강기능식품"
]

def random_items():
    return ", ".join(random.sample(categories, random.randint(1, 3)))

df['수거약품목'] = df.apply(lambda row: random_items(), axis=1)
df.to_csv("cheonan_seobuk_pharmacy_with_items.csv", index=False, encoding="utf-8-sig")

from google.colab import files
files.download("cheonan_seobuk_pharmacy_with_items.csv")
```



2

VS Code + Streamlit 단계: 앱 실행

📌 1) 프로젝트 폴더 구조

```
my_streamlit_project
├─ cheonan_seobuk_pharmacy_with_items.csv
└─ app.py
```

📌 2) app.py 기본 예제

```
import streamlit as st
import pandas as pd
import folium
```

```

from streamlit_folium import folium_static

st.set_page_config(page_title="폐의약품 수거 약국 찾기", page_icon="📍",
layout="wide")
st.title("📍 폐의약품 수거 약국 찾기")
st.markdown("원하는 약품목을 선택하면 표와 지도에서 확인할 수 있어요!")

@st.cache_data
def load_data():
    df = pd.read_csv("cheonan_seobuk_pharmacy_with_items.csv",
encoding="utf-8-sig")
    return df

df = load_data()

all_items = []
df['수거약품목'].dropna().apply(lambda x: all_items.extend([i.strip() for i
in x.split(',')]))
categories = sorted(set(all_items))

st.subheader("🗑 수거 약품목 선택 (최대 3개)")
cols = st.columns(3)
selected = []
for i, cat in enumerate(categories):
    if cols[i % 3].checkbox(cat):
        selected.append(cat)

if len(selected) > 3:
    st.error(" ! 최대 3개까지만 선택할 수 있어요.")
    selected = selected[:3]

if selected:
    mask = df['수거약품목'].apply(lambda x: any(tag in str(x) for tag in
selected))
    result = df[mask]
    st.success(f"선택한 약품목: {selected} → 약국 {len(result)}곳")
    st.dataframe(result[['병원명', '주소', '전화번호', '수거약품목']],
use_container_width=True)

    st.subheader("📍 약국 위치 지도")
    coords = result.dropna(subset=['위도', '경도'])
    if not coords.empty:
        m = folium.Map()
        bounds = [
            [coords['위도'].min(), coords['경도'].min()],
            [coords['위도'].max(), coords['경도'].max()]
        ]
        m.fit_bounds(bounds)
        for _, row in coords.iterrows():
            folium.Marker(

```

```

        [row['위도'], row['경도']],
        popup=f"{row['병원명']}<br>{row['수거약품목']}",
        tooltip=row['병원명']
    ).add_to(m)
    folium_static(m, width=800, height=500)
else:
    st.info("선택한 조건에 맞는 약국 위치 정보가 없습니다!")
else:
    st.info("위쪽에서 수거 약품목을 선택해주세요!")

```

3) 실행

```
streamlit run app.py
```

흐름 요약

단계	역할
Colab	CSV 구조 확인, 수거약품목 추가, 새 CSV 저장
내 PC	Colab에서 CSV 다운로드
VS Code	Streamlit 앱 작성, 실행
Streamlit	필터링 & 지도 출력 완성!

팁

- Colab = 데이터 가공
- VS Code = Streamlit 앱화
- 둘로 나누면 실습 실패율 0%!