

# 빅데이터 분석 시스템 개발

김용선

# INDEX

01

주소의 좌표 구하기

02

좌표들을 지도에 표시

03

결과 확인

# 01. 주소의 좌표 구하기\_01

```
import storedb
import folium

# 모든 지점 정보를 가져와서 지도에 표시

datalist = storedb.select_data() # 모든 지점의 주소를 가져와서 저장
addrList = [] # 마커의 x, y좌표들을 담을 리스트
nameList = [] # 마커의 지점 이름을 담을 리스트

# datalist -> (지점 이름, 지점 주소)을 담을 튜플들을 담을 리스트
for idx, data in enumerate(datalist):
    # 지점의 주소(string)을 위도 경도로 변환
    addr = storedb.conversion(data[1])
    # 주소->좌표 변환로 변환시 좌표값이 나오지 않은 경우
    if addr is None:
        continue
    # 주소가 좌표로 변환되었다면 좌표를 담는 리스트와 지점 이름을 담는 리스트에 추가
    addrList.append(addr)
    nameList.append(data[0])
```

## 01. 주소의 좌표 구하기\_02

```
# db에 저장되어 있는 지점들의 이름과 주소 정보를 읽어 올
def select_data():
    conn = oci.connect(connContent)
    cursor = conn.cursor()
    sql = """
        select * from store
    """
    cursor.execute(sql)
    datas = cursor.fetchall()
    cursor.close()
    conn.commit()
    conn.close()
    return datas # (지점 이름, 지점 주소)을 담은 튜플들을 담은 리스트 형태로 반환

# 텍스트파일에 저장되어있는 apikey를 읽어와서 반환
def get_key():
    with open('./properties.txt') as f:
        key = str(f.readline())
    return key
```

# 01. 주소의 좌표 구하기\_03

```
# 주소를 받아서 위도, 경도를 포함한 해당 주소에 대한 정보를 받아옴(Kakao API 이용)
def kakao_conversion(addr):
    headers = {
        'Content-Type': 'application/json; charset=utf-8',
        'Authorization': 'KakaoAK {}'.format(get_key())
    }
    address = addr.encode("utf-8")
    p = urllib.parse.urlencode(
        {
            'query': address
        }
    )
    result = requests.get("https://dapi.kakao.com/v2/local/search/address.json", headers=headers, params=p)
    return result.json()

# Kakao API에서 뽑아온 정보들 중에서 필요한 데이터 추출(주소의 위도, 경도)
def conversion(addr):
    value = kakao_conversion(addr)
    if len(value['documents']) >= 1: # 해당 주소에 대한 정보가 있는 경우에만 추출
        return value['documents'][0]['address']
    else:
        return None
```

## 02. 좌표들을 지도에 표시

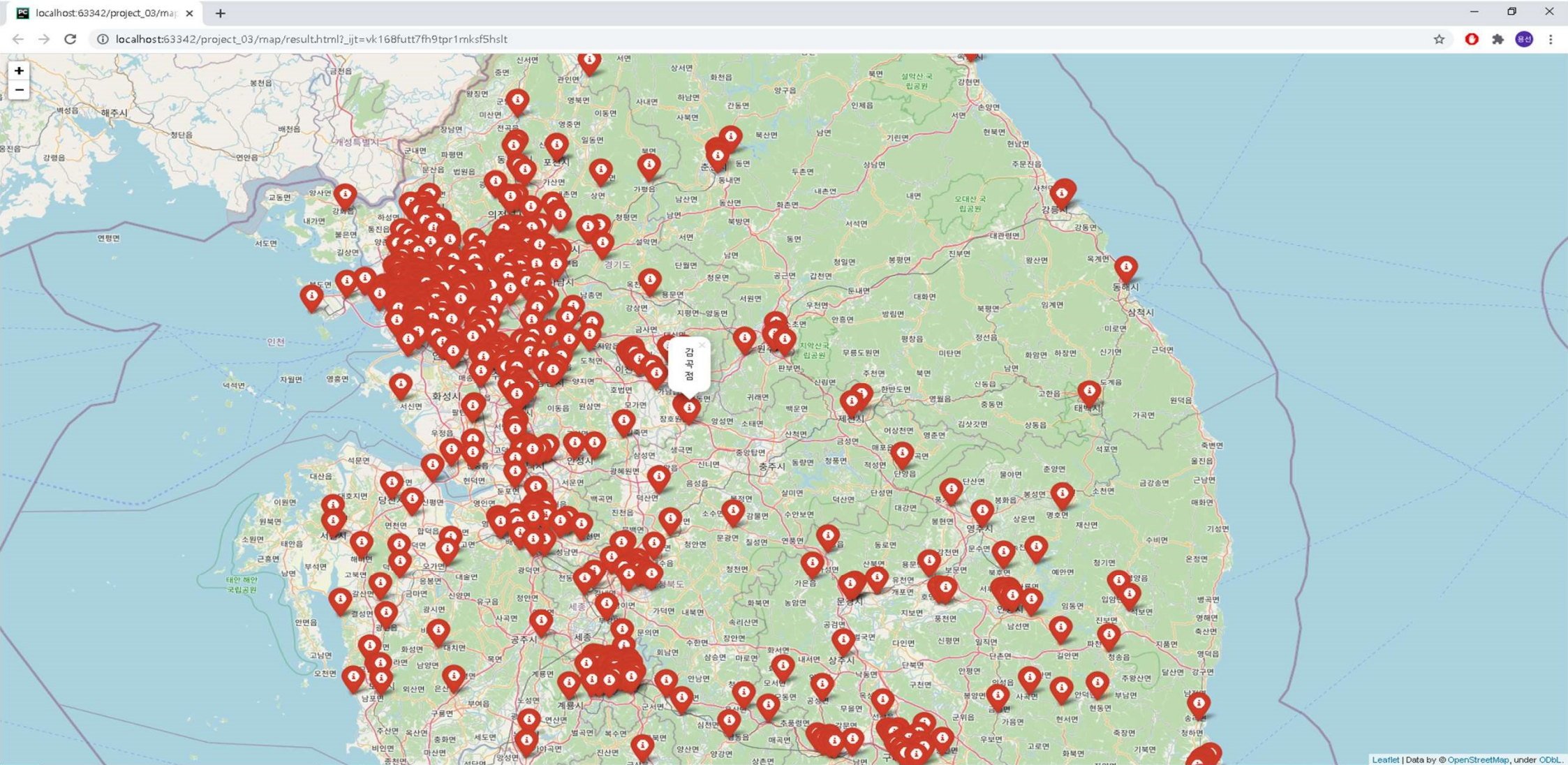
```
# 지도의 포커스를 맨 처음 리스트에 담긴 매장 위치를 기준으로 설정
map_osm = folium.Map(location=[addrList[0]['y'], addrList[0]['x']], zoom_start=10)

# 주소리스트에 담긴 x,y좌표에 마커를 지도에 표시 (popup시 지점 이름리스트에 담긴 이름들이 표기)
for idx, conversiondata in enumerate(addrList):
    folium.Marker(location=[addrList[idx]['y'], addrList[idx]['x']],
                  popup=nameList[idx], icon=folium.Icon(color="red", icon="info-sign")).add_to(map_osm)

# 지점들의 위치가 표시된 지도를 파일로 저장
map_osm.save("./map/result.html")
```



## 03. 결과 확인



THANK YOU!