

# Solar Pick!

태양열 발전 시설 설치 지역 추천 앱

<개발 완료 보고서>

팀명	COLORBORATION
작성자	권택상, 신준호, 성현석, 정권호
최종 작성일	2020.12.17

# 목 차

---

1. [제안](#)
  - 1.1 [팀 구성 및 역할 선정](#)
  - 1.2 [회의 시간 및 방식 선정](#)
  - 1.3 [개요 및 목적](#)
2. [구상 & 분석](#)
  - 2.1 [프로젝트 목표 인식](#)
  - 2.2 [콘텐츠 개요 작성](#)
  - 2.3 [pandas를 이용한 필요 데이터 수집 자료](#)
3. [설계 및 구현](#)
  - 3.1 [개발 환경](#)
  - 3.2 [개발 툴 학습 - 안드로이드 스튜디오](#)
  - 3.3 [데이터베이스 사용법 학습 - 파이어베이스](#)
  - 3.4 [디자인](#)
  - 3.5 [구현내용](#)
5. [테스트](#)
  - [단위 테스트 및 통합 테스트](#)
6. [완료](#)
  - 6.1 [구글 플레이 스토어 등록과정](#)
  - 6.2 [프로젝트 완료 소감](#)
7. [부록 - 코드](#)
  - 7.1 [클래스 및 데이터베이스](#)
    - 7.1.1 [SQLITE](#)
    - 7.1.2 [FIREBASE](#)
  - 7.2 [메인](#)
  - 7.3 [순위 추천](#)
  - 7.4 [그래프](#)
  - 7.5 [피드백](#)
  - 7.6 [인트로](#)
  - 7.7 [로딩](#)

## 1. 제안

### 가. 팀 구성 및 역할 선정

권택상(팀장)	메인화면 프론트엔드 구성요소 이미지화, 구글맵 API로 구글맵 연동, 등급선택 콤보박스(1~8등급 중 선택 시 등급별로 서로 다른 색의 마커<푸른색 좋음, 붉은색 나쁨>를 표시한다), 지역선택 콤보박스(도, 광역시, 특별시 선택할 경우 그 지역에 관련된 마커만 표시한다), 지도 설정 초기화 버튼(지도 마커 지우기, 콤보박스 설정 초기화하기), 파이썬으로 데이터 수집 및 판다스로 가공한 데이터를 기반으로 생성된 등급 식으로 지도 버튼 클릭 시 지도 상 마커 표시 구현
성현석(팀원)	지역 추천 main, sub 페이지, 로딩 페이지 프론트 및 백엔드 구현, 인트로 페이지 프론트 및 백엔드 구현, 등급 계산식 설계, 파이썬의 판다스로 데이터 수집 및 가공 가공
정권호(팀원)	플로팅 버튼 및 피드백 페이지 프론트 및 백엔드 개발 및 연동, 메인화면 검색 창 구현, 파이썬으로 데이터 수집 및 판다스로 데이터 가공, PPT 제작, 동영상 편집
신준호(팀원)	지역 그래프 convert, 합친 후 지속적 오류 테스트 및 그래프 페이지 이외의 페이지도 오류 시험력해서 지프 페이지 및 그래프 페이지 내 그래프 뷰 프론트 엔드 및 백엔드 개발, 연동, SQLite 및 파이어베이스 설계 및 연동, 파이썬으로 api, excel, csv 사용하여 데이터 수집, 지오코딩 및 판다스로 데이터 가공 후 Json 형식으로 코드 수정, 전체 페이지 테스트 및 전체 통합, aab 파일 변환 후 구글 플레이 스토어에 등록, 피드백 페이지 insert 부분

### 나. 회의 시간 및 방식 선정

마이크로소프트 discode(온라인), 대면회의(오프라인)

1차 회의	2020.09.12	주제선정, 요구사항 분석
2차 회의	2020.09.26	데이터 수집 및 필요 데이터 논의
3차 회의	2020.10.17	파이어베이스에 데이터 저장 및 개발 툴 설정
4차 회의	2020.10.31	안드로이드 스튜디오를 이용한 프론트엔드, 백 엔드 작성
5차 회의	2020.11.14	각종 오류 수정 및 추가 페이지 논의
6차 회의	2020.11.28	최종 수정 작업 및 완전한 구현
7차 회의	2020.12.17	최종 결과 보고서 작성 및 발표 진행 회의

### 다. 개요 및 목적

개요	신 재생 에너지로써 가능성이 무궁무진한 태양광 발전소의 설치를 위한 도우미 앱
목적	<div> <p> <b>&lt;그래프&gt; 산지 태양광발전시설 신규 증축 면적</b>              ■ 신규 증축 면적(ha)              2011: 21, 2012: 22, 2013: 44, 2014: 176, 2015: 522, 2016: 529, 2017: 1,435, 2018: 2,443, 2019: 1,024              자료 : 산림청           </p> <p> <b>최근 10년 국내 태양광 발전 설비 도입 현황</b> (단위:GW)              2010: 0.1, 2011: 0.1, 2012: 0.3, 2013: 0.5, 2014: 0.9, 2015: 1.1, 2016: 0.9, 2017: 1.3, 2018: 2.4, 2019: 3.1              출처: 에너지관리공단              CAGR +51%           </p> </div> <p>산지 태양광 발전시설이 증가하다 감소했지만 그럼에도 불구하고 국내 발전 설비 도입현황은 계속 증가하므로 태양광 발전시설을 설치하기 좋은 곳을 추천해주는 앱을 만들게 되었습니다.</p>

## 2. 구상 & 분석

### 가.프로젝트 목표 인식

태양광 발전소를 설치하기 좋은 지역을 등급을 매겨서 보여줌으로 적은 면적으로 높은 효율을 가지는 지역을 추천해준다.

### 나.콘텐츠 개요 작성

메인화면	
검색창	지역을 검색하면 그 지역에 대한 이동 및 마커 표시
콤보박스	도,광역시 선택, 등급 선택
플로팅 버튼	지역추천 페이지 이동 버튼, 피드백 페이지 이동 버튼
지역 추천 페이지	
도,광역시 추천	그 도에 대한 지역들 추천 페이지 이동 버튼
지역 추천	그래프 페이지 이동 버튼
그래프 페이지	
그래프	지역별 발전소 수,지역별 발전량,연별 산사태 발생 증감률
슬라이드	슬라이드를 통한 그래프간의 이동
피드백 페이지	
별점	별점을 데이터베이스에 전송
피드백	피드백 텍스트를 입력
전송 버튼	피드백 텍스트와 별점을 데이터베이스로 전송

### 다. pandas를 이용한 필요 데이터 수집 자료

가공 전 데이터	가공 후 데이터
<ul style="list-style-type: none"><li>- 공공데이터 포털 (산사태 취약지역,도별 태양광 발전소 설치현황)</li><li>- 통계청 데이터 (도별 탄소 대기오염도, 지역별 에너지 생산량)</li><li>- 카카오 맵 API</li><li>- V WORLD MAP API (지오코딩 위경도로부터의 29만여개의 지역이름 가져오기)</li><li>- 한국에너지기술연구원 (수평면 전 일사량)</li><li>- 전력통계정보시스템 (발전소당 발전량 및 세부주소)</li></ul>	<ul style="list-style-type: none"><li>-연별 발전소 설치 수</li><li>-수평면 전 일사량</li><li>-발전소 수 총량</li><li>-연별 허가된 태양광 발전소 수량</li><li>-지역별 총 산사태 횟수</li><li>-연별 산사태 증감률</li><li>-지역별 건축 허가 현황</li><li>-지역별 발전량</li></ul>

### 3. 설계 및 구현

---

#### 가. 개발 환경



#### 다. 개발 툴 학습 - 안드로이드 스튜디오

파이어 베이스의 데이터를 끌어오기 위해서 SQLite를 동영상또는 웹에서 검색해서 찾아보았다. 그리고 각각 xml과 안드로이드 스튜디오의 사용법을 책을 통해서 학습하였고 모르는 부분을 인터넷 검색을 통해서 찾아보았다.

#### 라. 데이터베이스 사용법 학습 - 파이어베이스

파이어베이스에 데이터를 집어넣고 안드로이드 스튜디오와 연동하는 방법을 공부하였고, 파이어베이스의 데이터를 안드로이드 스튜디오로 끌고오는 법을 공부하였다.

#### 나. 디자인

← 평가 페이지

별점을 매겨주세요!

★★★★★

별점!

실시간 FeedBack

제목

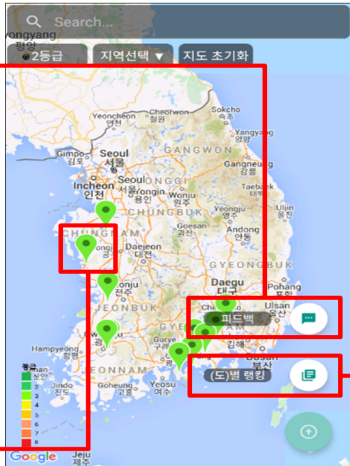
Hello

내용

World!

확인

시군구 순위 페이지  
도, 광역시 선택 후 표시되  
는 시/군/구 태양광 발전  
시설 순위를 표시



← 지역 선택

전라북도

전라남도

충청북도

충청남도

경기도

경상남도

경상북도

← 충청남도

2위 충청남도 청양군

3위 충청남도 아산시

4위 충청남도 보령시

5위 충청남도 당진시

6위 충청남도 홍성군

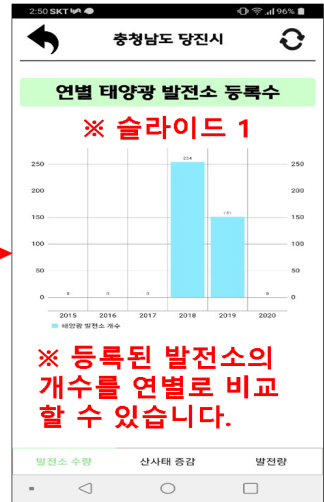
7위 충청남도 천안시

8위 충청남도 예산군

9위 충청남도 금산군

피드백 페이지  
별 점, 피드백 입력 시  
실시간 데이터 베이스  
에 저장

메인 페이지(구글 맵 API)  
지역 검색 창, 등급/지역선택 콤보 박스, 지도 설정  
초기화 버튼, 등급 표, 플로팅 액션 버튼 위치





그래프 페이지: 각 주제 별로 막대그래프와 꺾은 선 그래프로 표시

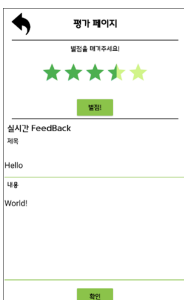
다. 구현 내용

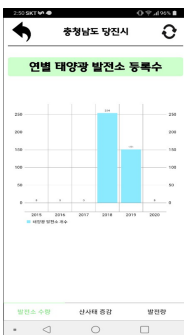
<p>메인화면</p>	검색창	검색한 지역으로 확대되어 지도 상 가운데에 보여집니다.
	등급선택 콤보 박스	1~8등급을 선택할 시 등급에 맞는 마커가 지도에 찍히게 됩니다.
	지역 선택 콤보 박스	원하는 지역(도, 광역시)을 선택할 시 그 지역에만 마커가 생성됩니다.
	지도 설정 초기화 버튼	버튼을 클릭할 시 콤보 박스 설정과 지도 확대 위치 마커가 모두 초기화됩니다.
	플로팅 버튼	메인화면 우측 하단의 버튼을 누를 시 (도)별 랭킹, 피드백 버튼이 생성됩니다.

	(도) 별 랭킹 버튼	버튼을 클릭할 시 (도)별 순위 페이지로 이동합니다.
	피드백 버튼	버튼을 클릭할 시 평가 페이지로 이동합니다.

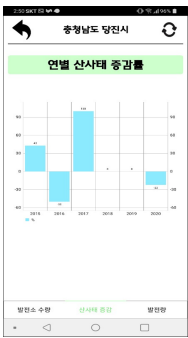
 <p>지역선택화면 (도,광역시)</p>	지역선택(도,광역시) 페이지	도와 광역시, 특별시가 세로로 나열되어 보여지고 사용자가 지역을 클릭할 시 시/군/구로 더 자세하게 알아볼 수 있도록 시/군/구 태양광 발전 시설 설치 유리 지역 순위를 보여줍니다.
---	-----------------	---

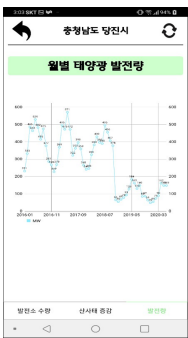
 <p>지역선택화면 (시, 군, 구)</p>	지역선택(시/군/구) 페이지	수집한 데이터를 기반으로 한 등급식으로 시/군/구의 태양광 발전 시설 설치가 유리한 지역을 순위별로 보여주어 사용자가 설치하기 유리한 지역을 쉽게 확인하여 고를 수 있도록 합니다.
--	-----------------	--

 <p>피드백화면</p>	별점 창	사용자가 앱에 관한 평가를 별점으로 표현할 수 있고 별점은 숫자로 변환되어 실시간 데이터베이스(Firebase)에 저장됩니다.
	피드백 창	사용자가 앱에 관한 자세한 평가를 문장으로 표현할 수 있고 제목과 내용을 입력하고 확인을 누를 시 텍스트 그대로 실시간 데이터베이스(Firebase)에 저장됩니다.

 <p>그래프화면1</p>	연별 태양광 발전소 등록수(막대 그래프)	연별 태양광 발전소 등록 수는 연 별로 등록한 태양광 발전소의 개수를 연 별로 비교하기 위해서 막대 그래프로 보여줍니다.
---	------------------------	---

(연별 태양광 발전소 등록 수)		
-------------------	--	--

 <p>그래프화면2 (연별 산사태 증감률)</p>	연별 산사태 증감률 (막대 그래프)	작년과 비교한 현재 년도의 산사태 발생의 증감률을 연별로 비교할 수 있도록 퍼센트 단위인 막대그래프로 보여줍니다.
--	------------------------	---

 <p>그래프화면3 (월별 태양광 발전량)</p>	월별 태양광 발전량 (꺾은 선 그래프)	선택한 시/군/구의 월별 태양광 발전량의 변화를 알 수 있도록 꺾은 선 그래프로 표시합니다.
---	--------------------------	---

#### 4. 테스트

단위 테스트 및 통합 테스트

권택상:

**2020-09-25** 프론트엔드 구성 요소 이미지화, 주제 선정 회의, 주제 관련된 데이터 유무 확인 조사



**2020-10-09** 피드백을 반영하여 프로젝트 주제 변경 및 데이터 유무 조사와 분석. 프론트엔드 구성 페이지(메인, 그래프, 순위 피드백 페이지) 결정

**2020-10-23** 구글 맵 API를 활용한 메인 페이지의 전체화면 구성, 그 외의 다른 구성요소(페이지 전환 플로팅 버튼, 마커, 콤보 박스) 배치 아이디어 도출

**2020-11-06** 구글 맵의 초기값을 설정하고 맵 위에 마커와, 검색창, 콤보박스, 플로팅 박스 위치 결정, 구글 맵 또는 콤보박스에 등급 표시 방법 아이디어 도출

**2020-11-20** FireBase에서 모아 놓은 데이터를 안드로이드 SQLite에서 다시 정리.

**2020-11-27** 정리한 데이터를 기반으로 생성한 등급식을 사용하여 1~8등급을 나타내는 8가지 색의 마커와 등급선택 콤보박스 이벤트 구현. 지역선택 버튼 이벤트 구현. 지도 설정 초기화 버튼 구현. 검색, 페이지 이동, 구성요소 선택 중 생기는 오류 수정

**2020-11-30** 피드백을 반영하여 사용자가 쉽게 알아볼 수 있도록 프론트엔드 디자인 최종 수정

성현석:

**2020-09-25** 프론트엔드 구성 요소 이미지화, 주제 선정 회의, 주제 관련된 데이터 유무 확인 조사

**2020-10-17** - 등급 판별식을 도출해 내기 위한 데이터들을 수집 및 순위 페이지의 명세서 작성

**2020-10-31** - 명세서에 따른 순위 페이지 프론트 엔드 작성 및 추가적으로 발전소 수량, 연별 발전량 현황 등의 데이터 수집

**2020-11-06** - 순위 페이지 백 엔드 작성 및 파이어 데이터베이스와의 연동 후 모은 데이터 들을 기반으로 등급 계산식 작성

**2020-11-13** - 파이어베이스의 문제점 발견(데이터 값 반환이 2번째 실행될 때에서야 반환이 됨) 및 SQLite와 연결을 통한 해결, 등급계산식을 통해 각 도,광역시의 시군구들을 등급기준으로 순위 판별 후 순위 페이지에 입력

**2020-11-20** - 쓰레드를 통한 다양한 애니메이션 표현으로 인트로 화면을 디자인, 순위 페이지의 디자인을 각 페이지들과 비교하며 의견수렴 후 확립

**2020-11-27** - 애니메이션을 이용하고 이 어플에 대한 팁을 주는 텍스트뷰를 제공하는 로딩 페이지를 작성, 순위 페이지와 메인 페이지를 연결

**2020-11-30** - 최종 인트로, 로딩 페이지 디자인 수정,그래프 페이지와 순위 페이지 사이를 로딩 페이지로 연결

신준호:

**2020-09-25** 프론트엔드 구성 요소 이미지화, 주제 선정 회의, 주제 관련된 데이터 유무 확인 조사

**2020-11-03** 파이어베이스 데이터 구조 생성 및 파이어 베이스 안드로이드 crud 연동 테스트

**2020-11-05** 슬라이드 및 슬라이드 내부 그래프 구성 및 반복 테스트 및 오류 수정

**2020-11-09** 그래프 출력 오류 수정 및 디자인 수정해서 테스트 반복

**2020-11-13** 파이어 베이스와 SharedPreferences 에서 2번째 실행에서 값이 반환되어서 페이지를 재호출하는것으로 수정, 순위 페이지 SharedPreferences 및 파이어베이스 동일 문제 수정 및 테스트

**2020-11-16** 디자인을 통일하여 수정 및 테스트

**2020-11-20** 전체페이지 통합 및 테스트

**2020-11-25** 메인페이지수정 본 받은후 추가해서 페이지 통합

**2020-11-26** 피드백 페이지 협력해서 파이어베이스로 데이터 insert 구현

**2020-11-27** 다른 페이지 수정하는 대로 전체 페이지 통합,수정 및 그래프 페이지 오류 발견

**2020-12-02** 그래프 페이지 11/13일 문제 재발생 후 SharedPreferences로 된 모든 데이터를 FIREBASE에서 SQLITE로 인트로에서 한번에 받아오는 방식으로 변경

**2020-12-03** 전체 페이지 인트로에서 파이어 베이스로 부터 받아온 데이터가 정리된 SQLITE를 사용하는 것으로 전체 수정

**2020-12-05** 그래프 오류 발견으로 HelloChart 차트 모듈을 MPAndroidChart로 변경하고 교수님의 피드백에 맞게 LineChart를 BarChart로 변경 및 반복 테스트

**2020-12-09** 각 페이지 오류 수집 및 전체 재수정

**2020-12-10** .aab 파일 생성 전체 기기별 테스트 및 오류 수정

**2020-12-14** 구글 플레이 스토등록 수차례 전체 테스트

정권호:

**2020-09-25** 프론트엔드 구성 요소 이미지화, 주제 선정 회의, 주제 관련된 데이터 유무 확인 조사

**2020-11-02** SearchView와 Floating Action Button 구현

**2020-11-03** Floating Action Button 연결 화면 구현 및 Floating Action Button 구현 및 개선

**2020-11-09** SearchView, GoogleMap API와 연동하여 테스트 및 개선

**2020-11-13** SearchView 디자인 개선 및 데이터베이스 연동, 테스트

**2020-11-22** Floating Action Button 및 SearchView파일 팀원에게 전달 및 Main Project에 연결, 피드백 페이지 구현 및 DB와 연동

**2020-11-23** Feedback 페이지 구현 및 개선 (DB와 연결)

**2020-11-26** 팀원들의 의견 반영하여 Main 화면 등급 이미지 창 구현을 통한 색상 별 등급의 구별의 가독성을 높임

**2020-11-27** Feedback 페이지 Main Project에 연결

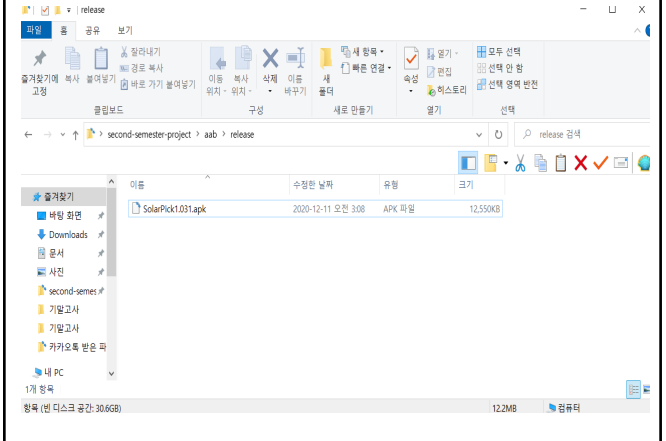
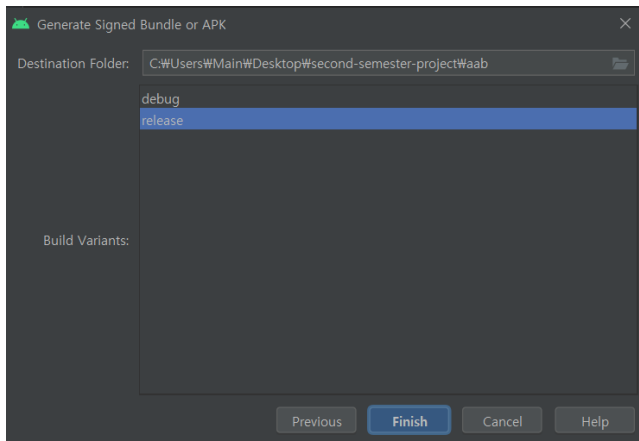
**2020-11-28** Feedback 페이지 문제점 발견 및 개선 (전체적인 Layout 구조 변경)

## 5. 완료

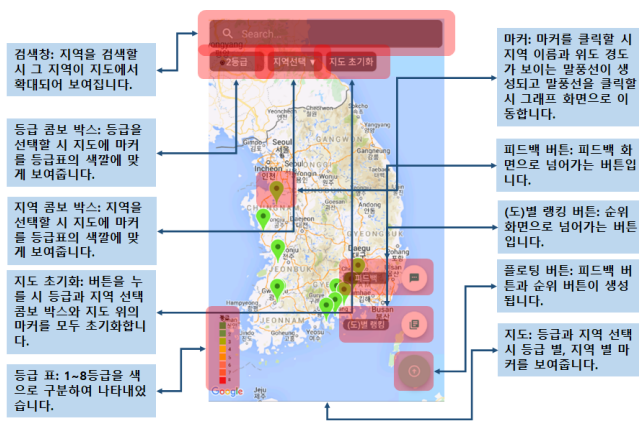
---

가. 구글 플레이 스토어 등록과정

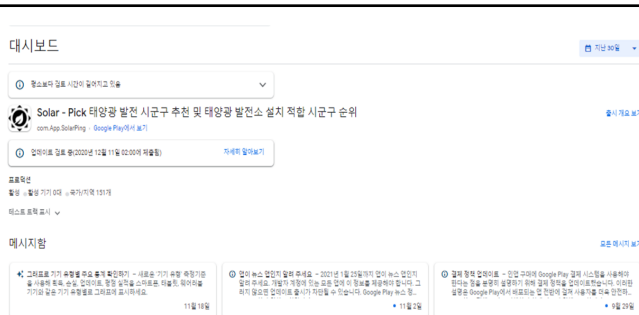
1. .aab파일로 변환
---------------



## 2. 사용 동영상, 이미지 제작 및 편집



## 3. 구글 플레이스토어 등록



플레이 스토어에서 사전 테스트 실시 및 필요 이미지 및 동영상 입력 후 등록

## 나. 프로젝트 완료 소감

인원	소감내용
권택상	<p><b>2학년 2학기</b>는 태양광 발전 시설 설치 유리 지역 추천 앱을 만들게 되었습니다. 앱 자체를 코딩하는 것 보다 주제 선정 및 데이터 조사 및 분석 과정이 쉽지 않았습니다. 그 이유 중 첫 번째는 주제를 먼저 정하고 데이터를 분석할 경우 주제에 맞는 데이터가 충분하지 않는 경우이고 두 번째는 데이터를 먼저 조사하고 주제에 맞추려는 경우에 주제가 흐지부지 되는 경우가 있었기 때문입니다. 마지막 이유로는 데이터를 조사하였음에도 불구하고 데이터 간 연관성을 확실하게 찾기가 힘든 경우가 있었습니다. 데이터 분석 후 시각화하는 프로젝트를 다음에 또 진행하게 된다면 새로운 주제도 좋지만 충분한 데이터가 존재할만한 주제로 고르는 것도 중요하고 조사한 데이터를 분석하여 연관성을 찾는 것이 중요함을 알게 되었습니다.</p> <p>프론트엔드를 구현하면서 새로웠던 점을 3가지 정도만 적어보겠습니다. 첫째로 구글 맵 API를 사용한 것인데 구글 맵에서 지원하는 옵션 기능(위도/경도, 마커, 나침반 등)을 사용하면서 맵 API라는 새로운 지식의 경로를 개척한 것 같아 뿌듯했습니다. 그 다음으로는 Firebase나 SQLite의 데이터를 가져와서 수식화하여 등급으로 나타내는 부분인데 데이터를 가져오는 과정과 수식을 사용하여 표현하는 과정이 참신했습니다. 마지막으로 이번 프로젝트는 저번 프로젝트에 비해 작업 처리량이 많아진 것 같아서 대견했습니다.</p> <p>프로젝트에서 저는 주로 프론트엔드 부분을 맡게 되었습니다. 전에 했던 프로젝트보다 프론트엔드 부분이 발전된 것 같아서 기분이 좋지만 다른 조의 프론트엔드를 보면서 배워야 할 부분도 많다고 느꼈습니다. 프로젝트 전의 저는 디자인에 자신이 있어서 프론트엔드를 하겠다고 지원했지만 프로젝트가 끝나고 스스로를 평가해 보았을 때 아직 서툰 모습이 많은 것 같아 분발의 계기가 되는 좋은 프로젝트였습니다.</p>
성현석	<p>이번 프로젝트에서는 주제선정이 가장 어려웠던 것 같습니다. 이미 있는 것들은 안되고 특정 누군가에게 도움이 되어야하며 왜 이 앱을 사용해야지 등 여러 사항을 고민 해야 했고, 그렇게 하루정도는 기본으로 밤을 새며 생각해보았었다. 그렇게 조원이 내준 아이디어가 이 태양광 발전소 설치 위치 추천 어플이었습니다. 조원 전원은 이 아이디어에 수긍하였고, 각자가 맡은 데이터들을 점차 모아갔습니다. 데이터 수집에만 거의 보름 정도가 걸렸고 이것을 파이어베이스에 집어넣어서 이제 기본 베이스를 깔고 안드로이드 스튜디오와 파이어베이스를 연동하여 프로그램을 작성하였습니다. 이 과정에서 꽤나 좋은 경험을 한 것 같습니다. 안드로이드 스튜디오와 데이터베이스 이 두가지의 사용법을 실제로 사용해보면서 습득한다는 것이, 조원들과 함께 하는 것이라서 더욱 책임감이 생기고 혼자였다면 하기 힘들었을 주제를 같이 함으로써 신선한 경험들을 쌓을 수 있었고, 이런 경험들이 계속 저를 강하게 이끌어주는 것 같았습니다. 이런 감각이 저를 프로젝트로 하여금 도전할 수 있게하는 계기가 되는 것 같습니다.</p>
정권호	<p>이렇게 <b>2학년 2학기</b>의 설계 및 프로젝트도 끝이 났습니다. 다같이 주제를 고민할 때에도 많은 시행착오가 있었고, 이후에 사용자 설정 및 구체적인 틀 작성을 하는데에도 많은 고민들을 해왔던 것 같습니다. 저희 팀은 데이터베이스를 위한 작업에만 해도 많은 시간을 쏟았었습니다. 왜냐하면 저희가 정했던 주제의 데이터들이 있는 것을 확인하고 주제를 확정하였지만 저희 생각보다도 너무 광범위한 범위에 걸쳐서 데이터가 퍼져있었기 때문입니다. 이에 저희는 보름에 걸쳐 공공데이터 포털이나 각종 사이트에 분포되어 있는 데이터들을 정제하고 분석하여 의미 있는 한 개의 데이터로 만들 수 있었습니다. 다만, 한 개의 DB로 프로그램을 만드는 것은 위험 할 수 있다는 교수님의 말씀을 듣고, 다음부터는</p>

	<p>DB를 구축할 때에 여러개의 DB로 나누고 각 DB를 외래키로 받아들이어 사용하면 좋겠다는 생각을 하였습니다. 이러한 DB를 바탕으로 안드로이드 프로그램을 구현하였고, 처음 사용해보는 툴이었지만 언어와 구조가 복잡하지 않았기 때문에 금방 적응할 수 있었던 거 같습니다. 안드로이드 스튜디오 뿐만 아니라 다양한 툴이 굉장히 많은데 이렇게 설계에서 프로젝트를 함으로서 알게되는 툴 혹은 다양한 자료들이 많은것 같습니다. 이러한 많은 경험들을 통해서 할 수 있는 범위가 넓어지고 많은 학견을 가질 수 있는 것 같습니다. 이번 학기 유능한 팀원들과 같이 할 수 있어서 좋은 프로젝트가 될 수 있었던 것 같습니다.</p>
신준호	<p>데이터 수집이 처음에 저에게는 난관이었던게 자료가 시/군/군 별로 다른 사이트에 따로따로 분리되어져있고 엑셀 줄 수가 30만개가 넘어가고 파일 개수가 수십개에 달할 정도로 방대해서 실질적으로 우리가 원하는 쓰임새로 정리하고 가공하는 데에 엄청난 시간을 쏟아 부었습니다. 그러나 앱구현에 관련해서 이번에는 방학 때 멋쟁이사자에서 했던 앱프로젝트 알바계산기 때 보다 프론트엔드에 대한 이해도가 높아져서 프론트 엔드를 좀더 심플하고 더 낮게 만들 었다고 느꼈습니다. 여러번의 프로젝트를 거치면서 늘어가고 거기서 제가 할 수 있는게 더 늘어나서 밤을 새면서하고 정말 프로젝트에 신경쓴 시간이 길었지만 프로그래밍이 재밌다 라고 느꼈던 프로젝트 였습니다. 어떻게 제가 팀원들과 상의해서 프로젝트의 방향을 정해나갈지 비록 팀원이지만 팀장과 다른 팀원들과 함께 고민 할 수 있어서 좋았습니다. 가장 좋았던 점은 제가 백엔드를 하고 싶었는데 데이터베이스 설계부터 실현까지 할 수 있어서 좋았습니다. 교수님의 좋은 피드백들로 제가 부족한 부분을 팀원들과 함께 채워나가고 제가 오래 걸리는부분들을 팀원들과 협력하여 할 수 있어서 정말 좋았습니다. 앞으로의 프로젝트에서 이번에 부족했던 부분 잘했던 부분을 참고하여 좀 더 좋은 소프트웨어를 만들고 사용자 입장에서 생각하고 제 자신이 더 발전해가는 느낌이 들었습니다. 프로젝트를 하는 한명한명 모두의 책임도 작지 않은 부분이라고 느꼈습니다.</p>

## 6. 부록 - code

### 7.1 클래스 및 데이터베이스

#### 7.1.1 SQLITE

## city

```
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.SQLException;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

import androidx.annotation.Nullable;

public class citiesDatabase {
    private static final String DATABASE_NAME = "citiesDataSet.db";
    private static final int DATABASE_VERSION = 1;
    public static SQLiteDatabase DB;
    Database DBHelper;
    private final Context dbCtx;

    private static class Database extends SQLiteOpenHelper{

        public Database(@Nullable Context context, @Nullable String name,
            @Nullable SQLiteDatabase.CursorFactory factory, int version) {
            super(context, name, factory, version);
        }

        @Override
        public void onCreate(SQLiteDatabase db) {
            db.execSQL(citiesNameDatabaseStructure.CreateDB._Create0);
        }

        @Override
        public void onUpgrade(SQLiteDatabase db, int i, int i1) {
            db.execSQL("DROP TABLE IF EXISTS "+
citiesNameDatabaseStructure.CreateDB.tablename);
            onCreate(db);
        }
    }

    public citiesDatabase(Context context){
        this.dbCtx = context;
    }

    public citiesDatabase open() throws SQLException{
        DBHelper = new Database(dbCtx,DATABASE_NAME,null,DATABASE_VERSION);
        DB = DBHelper.getWritableDatabase();
        return this;
    }

    public void create(){
        DBHelper.onCreate(DB);
    }

    public void close(){
        DB.close();
    }

    public long insertColumn(String province, String city,String anotherInfo){
        ContentValues values = new ContentValues();
        values.put(citiesNameDatabaseStructure.CreateDB.province,province);
    }
}
```

```

        values.put(citiesNameDatabaseStructure.CreateDB.city,city);
        values.put(citiesNameDatabaseStructure.CreateDB.anotherInfo,anotherInfo);

        return DB.insert(citiesNameDatabaseStructure.CreateDB.tablename,
null,values);
    }

    public Cursor selectColumns(){
        return
DB.query(citiesNameDatabaseStructure.CreateDB.tablename,null,null,null,null,null
,null);
    }

    public Cursor selectWhatIWant(String province,String city){
        String sql = String.format("SELECT * FROM %s WHERE %s = '%s' AND %s =
'%s'",citiesNameDatabaseStructure.CreateDB.tablename,citiesNameDatabaseStructure
.CreateDB
.province,province,citiesNameDatabaseStructure.CreateDB.city,city);
        return DB.rawQuery(sql,null);
    }

    public void deleteAll(){
        DBHelper.onUpgrade(DB,1,2);
    }

    public void UpdateAll(String province,String city,String anotherInfo){
        String a = String.format("UPDATE %s SET %s = '%s' , %s = '%s',%s = '%s'
WHERE %s = '%s' AND %s = '%s' ";
citiesNameDatabaseStructure.CreateDB.tablename,citiesNameDatabaseStructure.Creat
eDB.province,province,citiesNameDatabaseStructure.CreateDB.city,city,citiesNameD
atabaseStructure.CreateDB.anotherInfo,anotherInfo,citiesNameDatabaseStructure.Cr
eateDB.province,province,citiesNameDatabaseStructure.CreateDB.city,city);
        DB.execSQL(a);
    }
}

import android.provider.BaseColumns;

public class citiesNameDatabaseStructure {
    public static class CreateDB implements BaseColumns {
        final static String tablename = "citytable";
        final static String province = "province";
        final static String city = "city";
        final static String anotherInfo = "anotherInfo";
        final static String _Create0 = "CREATE TABLE IF NOT EXISTS "+tablename
        + "(" + _ID + " INTEGER PRIMARY KEY AUTOINCREMENT, "
        +province + " TEXT NOT NULL, "
        +city + " TEXT NOT NULL, "
        +anotherInfo + " TEXT NOT NULL );";
    }
}

```

## CityGenerate

```

import android.content.ContentValues;
import android.content.Context;

```

```

import android.database.Cursor;
import android.database.SQLException;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

import androidx.annotation.Nullable;

public class generatingDatabase {
    private static final String DATABASE_NAME = "citiesDataSet.db";
    private static final int DATABASE_VERSION = 1;
    public static SQLiteDatabase DB;
    generatingDatabase.Database DBHelper;
    private final Context dbCtx;

    private static class Database extends SQLiteOpenHelper {

        public Database(@Nullable Context context, @Nullable String name,
            @Nullable SQLiteDatabase.CursorFactory factory, int version) {
            super(context, name, factory, version);
        }

        @Override
        public void onCreate(SQLiteDatabase db) {
            db.execSQL(generatingAmountDatabase.CreateDB._Create0);
        }

        @Override
        public void onUpgrade(SQLiteDatabase db, int i, int i1) {
            db.execSQL("DROP TABLE IF EXISTS "+
                generatingAmountDatabase.CreateDB.tablename);
            onCreate(db);
        }
    }

    public generatingDatabase(Context context){
        this.dbCtx = context;
    }

    public void open() throws SQLException {
        DBHelper = new
            generatingDatabase.Database(dbCtx,DATABASE_NAME,null,DATABASE_VERSION);
        DB = DBHelper.getWritableDatabase();
    }

    public void create(){
        DBHelper.onCreate(DB);
    }

    public void close(){
        DB.close();
    }

    public void insertColumn(String region,String value){
        ContentValues values = new ContentValues();
        values.put(generatingAmountDatabase.CreateDB.region,region);
        values.put(generatingAmountDatabase.CreateDB.amount,value);
        DB.insert(generatingAmountDatabase.CreateDB.tablename, null, values);
    }
}

```



```

    public Cursor selectColumns(){
        return DB.query(generatingAmountDatabase.CreateDB.tablename, null, null,
null, null, null, null);
    }

    public Cursor selectWhatIWant(String region){
        String sql = String.format("SELECT * FROM %s WHERE %s =
'%s'",generatingAmountDatabase.CreateDB.tablename,generatingAmountDatabase.Creat
eDB.region,region);
        return DB.rawQuery(sql,null);
    }

    public void UpdateColumn(String region,String values){
        String sql = String.format("UPDATE %s SET %s = '%s',%s = '%s' WHERE %s =
'%s'",generatingAmountDatabase.CreateDB.tablename,generatingAmountDatabase.Creat
eDB.region,region,generatingAmountDatabase.CreateDB.amount,values,generatingAmou
ntDatabase.CreateDB.region,region);
        DB.execSQL(sql);
    }

    public void deleteAll(){
        DBHelper.onUpgrade(DB,1,2);
    }

}

import android.provider.BaseColumns;

public class generatingAmountDatabase {
    public static class CreateDB implements BaseColumns {
        final static String tablename = "generatingTable";
        final static String region = "region";
        final static String amount = "amountOfGenerating";
        final static String _Create0 = "CREATE TABLE IF NOT EXISTS "+tablename
        + "(" + _ID + " INTEGER PRIMARY KEY AUTOINCREMENT, "
        +region + " TEXT NOT NULL, "
        +amount + " TEXT NOT NULL );";
    }
}

```

## Graph

```

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.SQLException;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

import androidx.annotation.Nullable;

public class GraphDatabase {
    private static final String DATABASE_NAME = "GDataset.db";
    private static final int DATABASE_VERSION = 1;

```

```

public static SQLiteDatabase DB;
Database DBHelper;
private final Context dbCtx;

private static class Database extends SQLiteOpenHelper{

    public Database(@Nullable Context context, @Nullable String name,
@Nullable SQLiteDatabase.CursorFactory factory, int version) {
        super(context, name, factory, version);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL(GraphDatabaseStructure.CreateDB._CREATE0);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int i, int i1) {
        db.execSQL("DROP TABLE IF EXISTS "+
GraphDatabaseStructure.CreateDB.TABLE_NAME);
        onCreate(db);
    }
}

public GraphDatabase(Context context){
    this.dbCtx = context;
}

public GraphDatabase open() throws SQLException{
    DBHelper = new Database(dbCtx,DATABASE_NAME,null,DATABASE_VERSION);
    DB = DBHelper.getWritableDatabase();
    return this;
}

public void create(){
    DBHelper.onCreate(DB);
}

public void close(){
    DB.close();
}

    public void insertColumn(String CITY_NAME, String YEARS_OF_SOLAR_POWER,
String NUMBERS_OF_SOLAR_POWER, String YEARS_OF_AVALANCHE, String
VALUES_OF_AVALANCHE){
        ContentValues values = new ContentValues();
        values.put(GraphDatabaseStructure.CreateDB.CITY_NAME,CITY_NAME);

values.put(GraphDatabaseStructure.CreateDB.YEARS_OF_SOLAR_POWER,YEARS_OF_SOLAR_
POWER);

values.put(GraphDatabaseStructure.CreateDB.NUMBERS_OF_SOLAR_POWER,NUMBERS_OF_SO
LAR_POWER);

values.put(GraphDatabaseStructure.CreateDB.YEARS_OF_AVALANCHE,YEARS_OF_AVALANCH
E);

values.put(GraphDatabaseStructure.CreateDB.VALUES_OF_AVALANCHE,VALUES_OF_AVALAN
CHE);

```

```

        DB.insert(GraphDatabaseStructure.CreatedDB.TABLE_NAME, null, values);
    }

    public Cursor selectColumns(){
        return
        DB.query(GraphDatabaseStructure.CreatedDB.TABLE_NAME,null,null,null,null,null,Gra
        phDatabaseStructure.CreatedDB.YEARS_OF_SOLAR_POWER);
    }

    public Cursor selectedWhatIWant(String province_city){
        String sql = String.format("SELECT * FROM %s WHERE %s =
        '%s'",GraphDatabaseStructure.CreatedDB.TABLE_NAME,GraphDatabaseStructure.CreatedDB
        .CITY_NAME,province_city);
        return DB.rawQuery(sql,null);
    }

    public void deleteAll(){
        DBHelper.onUpgrade(DB,1,2);
    }

    public void UpdateAll(String CITY_NAME, String YEARS_OF_SOLAR_POWER, String
    NUMBERS_OF_SOLAR_POWER, String YEARS_OF_AVALANCHE, String VALUES_OF_AVALANCHE){
        String sql = String.format("UPDATE %s SET %s = '%s' , %s = '%s' WHERE %s
        = '%s' AND %s = '%s' AND %s = '%s'";
        GraphDatabaseStructure.CreatedDB.TABLE_NAME,
        GraphDatabaseStructure.CreatedDB.NUMBERS_OF_SOLAR_POWER,NUMBERS_OF_SOLAR_POWER,G
        raphDatabaseStructure.CreatedDB.VALUES_OF_AVALANCHE,VALUES_OF_AVALANCHE,
        GraphDatabaseStructure.CreatedDB.YEARS_OF_SOLAR_POWER,YEARS_OF_SOLAR_POWER,Graph
        DatabaseStructure.CreatedDB.YEARS_OF_AVALANCHE,YEARS_OF_AVALANCHE,GraphDatabaseS
        tructure.CreatedDB.CITY_NAME,CITY_NAME);
        DB.execSQL(sql);
    }

}

import android.provider.BaseColumns;

public class GraphDatabaseStructure {
    public static final class CreatedDB implements BaseColumns{
        public static final String TABLE_NAME = "twoGraphsTmp";
        public static final String CITY_NAME = "cityName";
        public static final String YEARS_OF_SOLAR_POWER = "yearsOfSolarPower";
        public static final String NUMBERS_OF_SOLAR_POWER =
        "numberOfSolarPower";
        public static final String YEARS_OF_AVALANCHE = "yearsOfAvalanche";
        public static final String VALUES_OF_AVALANCHE = "ValuesOfAvalanche";

        public static final String _CREATE0 = "CREATE TABLE IF NOT EXISTS " +
        TABLE_NAME
            + "("
            + _ID + " INTEGER PRIMARY KEY AUTOINCREMENT, "
            + CITY_NAME + " TEXT NOT NULL, "
            + YEARS_OF_SOLAR_POWER + " TEXT NOT NULL, "
            + NUMBERS_OF_SOLAR_POWER + " TEXT NOT NULL, "

```

```

    +YEARS_OF_AVALANCHE + " TEXT NOT NULL, "
    +VALUES_OF_AVALANCHE + " TEXT NOT NULL );";
}
}

```

## 7.1.2 FIREBASE

```

{
  "1" : {
    "2015" : 4,
    "2016" : 2,
    "2017" : 0,
    "2018" : 3,
    "2019" : 0,
    "2020" : 1,
    "201501(전년누계대비 증감률)" : 0,
    "201601(전년누계대비 증감률)" : 0,
    "201701(전년누계대비 증감률)" : 0,
    "201801(전년누계대비 증감률)" : 0,
    "201901(전년누계대비 증감률)" : 0,
    "202001(전년누계대비 증감률)" : 0,
    "grade" : 0.62406586,
    "경도" : 128.5968,
    "도*광역시" : "강원도",
    "발전소수량" : 10,
    "산사태취약지역(개수)" : 0,
    "산사태취약지역(고속도로 근처)" : 0,
    "시군구" : "속초시",
    "시군구별 건축허가현황" : 48595,
    "시군구일사량평균" : 3.759341405,
    "위도" : 38.2124
  },
  "2" : {
    "2015" : 3,
    "2016" : 3,
    "2017" : 14,
    "2018" : 9,
    "2019" : 3,
    "2020" : 2,
    "201501(전년누계대비 증감률)" : 0,
    "201601(전년누계대비 증감률)" : -14.3,
    "201701(전년누계대비 증감률)" : -16.7,
    "201801(전년누계대비 증감률)" : 0,
    "201901(전년누계대비 증감률)" : -20,
    "202001(전년누계대비 증감률)" : 25,
    "grade" : 2.116152135,
    "경도" : 129.1143,
    "도*광역시" : "강원도",
    "발전소수량" : 34,
    "산사태취약지역(개수)" : 0,
    "산사태취약지역(고속도로 근처)" : 0,
    "시군구" : "동해시",
    "시군구별 건축허가현황" : 5063,
    "시군구일사량평균" : 3.776023134,
    "위도" : 37.5587
  },
  "3" : {
    "2015" : 18,
    "2016" : 59,
    "2017" : 93,
    "2018" : 50,
    "2019" : 7,
    "2020" : 0,
    "201501(전년누계대비 증감률)" : -22.2,
    "201601(전년누계대비 증감률)" : -7.1,
    "201701(전년누계대비 증감률)" : -23.1,
    "201801(전년누계대비 증감률)" : -10,

```

```

"201901(전년누계대비 증감률)" : -11.1,
"202001(전년누계대비 증감률)" : -37.5,
"grade" : 27.25862152,
"경도" : 127.8848,
"도*광역시" : "강원도",
"발전소수량" : 227,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 4,
"시군구" : "원주시",
"시군구별 건축허가현황" : 63469,
"시군구일사량평균" : 3.855315675,
"위도" : 37.17
},
"4" : {
"2015" : 17,
"2016" : 15,
"2017" : 171,
"2018" : 222,
"2019" : 65,
"2020" : 20,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : -100,
"202001(전년누계대비 증감률)" : 0,
"grade" : 31.97839055,
"경도" : 128.4662,
"도*광역시" : "강원도",
"발전소수량" : 510,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "고성군",
"시군구별 건축허가현황" : 9688,
"시군구일사량평균" : 3.729727344,
"위도" : 38.3801
},
"5" : {
"2015" : 4,
"2016" : 35,
"2017" : 94,
"2018" : 179,
"2019" : 181,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : -44.4,
"202001(전년누계대비 증감률)" : -40,
"grade" : 51.66525085,
"경도" : 128.7061,
"도*광역시" : "강원도",
"발전소수량" : 493,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 2,
"시군구" : "평창군",
"시군구별 건축허가현황" : 9319,
"시군구일사량평균" : 3.83169398,
"위도" : 37.6736
},
"6" : {
"2015" : 14,
"2016" : 30,
"2017" : 108,
"2018" : 194,
"2019" : 101,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : -12.5,
"201701(전년누계대비 증감률)" : -28.6,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : -20,

```

```

"202001(전년누계대비 증감률)" : 0,
"grade" : 58.79799305,
"경도" : 129.2052,
"도*광역시" : "강원도",
"발전소수량" : 447,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 6,
"시군구" : "삼척시",
"시군구별 건축허가현황" : 35302,
"시군구일사량평균" : 3.777546867,
"위도" : 37.1498
},
"7" : {
"2015" : 5,
"2016" : 10,
"2017" : 86,
"2018" : 151,
"2019" : 22,
"2020" : 6,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : -16.7,
"201701(전년누계대비 증감률)" : 20,
"201801(전년누계대비 증감률)" : -16.7,
"201901(전년누계대비 증감률)" : 40,
"202001(전년누계대비 증감률)" : -14.3,
"grade" : 25.36350411,
"경도" : 127.5165,
"도*광역시" : "경기도",
"발전소수량" : 280,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 1,
"시군구" : "가평군",
"시군구별 건축허가현황" : 16161,
"시군구일사량평균" : 3.867535046,
"위도" : 37.8254
},
"8" : {
"2015" : 29,
"2016" : 12,
"2017" : 31,
"2018" : 42,
"2019" : 45,
"2020" : 0,
"201501(전년누계대비 증감률)" : 16.7,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : -16.7,
"201801(전년누계대비 증감률)" : 20,
"201901(전년누계대비 증감률)" : 33.3,
"202001(전년누계대비 증감률)" : -12.5,
"grade" : 9.598713093,
"경도" : 126.8993,
"도*광역시" : "경기도",
"발전소수량" : 159,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "고양시",
"시군구별 건축허가현황" : 83322,
"시군구일사량평균" : 3.963073527,
"위도" : 37.6466
},
"9" : {
"2015" : 0,
"2016" : 0,
"2017" : 1,
"2018" : 2,
"2019" : 0,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : -12.5,
"201701(전년누계대비 증감률)" : 12.5,
"201801(전년누계대비 증감률)" : -11.1,
"201901(전년누계대비 증감률)" : 25,
"202001(전년누계대비 증감률)" : -20,

```

```

"grade" : 0.179745889,
"경도" : 126.9954,
"도*광역시" : "경기도",
"발전소수량" : 3,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "과천시",
"시군구별 건축허가현황" : 631192,
"시군구일사량평균" : 4.00847038,
"위도" : 37.4484
},
"10" : {
"2015" : 0,
"2016" : 2,
"2017" : 0,
"2018" : 3,
"2019" : 2,
"2020" : 0,
"201501(전년누계대비 증감률)" : 28.6,
"201601(전년누계대비 증감률)" : -22.2,
"201701(전년누계대비 증감률)" : 16.7,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : -12.5,
"grade" : 0.418665861,
"경도" : 126.8482,
"도*광역시" : "경기도",
"발전소수량" : 7,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "광명시",
"시군구별 건축허가현황" : 15461,
"시군구일사량평균" : 4.019059135,
"위도" : 37.4698
},
"11" : {
"2015" : 9,
"2016" : 5,
"2017" : 21,
"2018" : 60,
"2019" : 61,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : -11.1,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : -12.5,
"grade" : 13.88228745,
"경도" : 127.3524,
"도*광역시" : "경기도",
"발전소수량" : 156,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 1,
"시군구" : "광주시",
"시군구별 건축허가현황" : 82698,
"시군구일사량평균" : 3.975510012,
"위도" : 37.3509
},
"12" : {
"2015" : 0,
"2016" : 2,
"2017" : 0,
"2018" : 2,
"2019" : 2,
"2020" : 0,
"201501(전년누계대비 증감률)" : -20,
"201601(전년누계대비 증감률)" : -25,
"201701(전년누계대비 증감률)" : -16.7,
"201801(전년누계대비 증감률)" : 20,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : -16.7,
"grade" : 0.361452872,

```

```

"경도" : 127.1492,
"도*광역시" : "경기도",
"발전소수량" : 6,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "구리시",
"시군구별 건축허가현황" : 21056,
"시군구일사량평균" : 3.975785462,
"위도" : 37.5964
},
"13" : {
"2015" : 0,
"2016" : 0,
"2017" : 3,
"2018" : 3,
"2019" : 4,
"2020" : 0,
"201501(전년누계대비 증감률)" : -28.6,
"201601(전년누계대비 증감률)" : 20,
"201701(전년누계대비 증감률)" : 16.7,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 14.3,
"202001(전년누계대비 증감률)" : -25,
"grade" : 0.884417332,
"경도" : 126.9431,
"도*광역시" : "경기도",
"발전소수량" : 10,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 1,
"시군구" : "군포시",
"시군구별 건축허가현황" : 18019,
"시군구일사량평균" : 4.012561058,
"위도" : 37.3464
},
"14" : {
"2015" : 24,
"2016" : 32,
"2017" : 58,
"2018" : 140,
"2019" : 84,
"2020" : 0,
"201501(전년누계대비 증감률)" : -22.2,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : -14.3,
"201801(전년누계대비 증감률)" : 16.7,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : -28.6,
"grade" : 30.14743603,
"경도" : 126.771,
"도*광역시" : "경기도",
"발전소수량" : 338,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 1,
"시군구" : "김포시",
"시군구별 건축허가현황" : 79304,
"시군구일사량평균" : 3.961660396,
"위도" : 37.6033
},
"15" : {
"2015" : 8,
"2016" : 3,
"2017" : 20,
"2018" : 22,
"2019" : 5,
"2020" : 0,
"201501(전년누계대비 증감률)" : -12.5,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : -14.3,
"201901(전년누계대비 증감률)" : 16.7,
"202001(전년누계대비 증감률)" : -28.6,
"grade" : 5.182736441,
"경도" : 127.2165,

```



```

"도*광역시" : "경기도",
"발전소수량" : 58,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 1,
"시군구" : "남양주시",
"시군구별 건축허가현황" : 358248,
"시군구일사량평균" : 3.950562653,
"위도" : 37.636
},
"16" : {
"2015" : 5,
"2016" : 3,
"2017" : 11,
"2018" : 10,
"2019" : 7,
"2020" : 4,
"201501(전년누계대비 증감률)" : -12.5,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : -14.3,
"201901(전년누계대비 증감률)" : 16.7,
"202001(전년누계대비 증감률)" : -42.9,
"grade" : 2.471717404,
"경도" : 127.0544,
"도*광역시" : "경기도",
"발전소수량" : 40,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "동두천시",
"시군구별 건축허가현황" : 12564,
"시군구일사량평균" : 3.820706489,
"위도" : 37.8903
},
"17" : {
"2015" : 3,
"2016" : 0,
"2017" : 2,
"2018" : 1,
"2019" : 4,
"2020" : 0,
"201501(전년누계대비 증감률)" : -11.1,
"201601(전년누계대비 증감률)" : -12.5,
"201701(전년누계대비 증감률)" : -14.3,
"201801(전년누계대비 증감률)" : -16.7,
"201901(전년누계대비 증감률)" : 40,
"202001(전년누계대비 증감률)" : -14.3,
"grade" : 0.59843508,
"경도" : 126.769,
"도*광역시" : "경기도",
"발전소수량" : 10,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "부천시",
"시군구별 건축허가현황" : 45088,
"시군구일사량평균" : 4.015649199,
"위도" : 37.4845
},
"18" : {
"2015" : 1,
"2016" : 1,
"2017" : 2,
"2018" : 3,
"2019" : 2,
"2020" : 0,
"201501(전년누계대비 증감률)" : 14.3,
"201601(전년누계대비 증감률)" : -12.5,
"201701(전년누계대비 증감률)" : 14.3,
"201801(전년누계대비 증감률)" : -25,
"201901(전년누계대비 증감률)" : 33.3,
"202001(전년누계대비 증감률)" : -12.5,
"grade" : 0.540718717,
"경도" : 127.1226,
"도*광역시" : "경기도",

```

```

"발전소수량" : 9,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "성남시",
"시군구별 건축허가현황" : 90804,
"시군구일사량평균" : 3.992014258,
"위도" : 37.3402
},
"19" : {
  "2015" : 8,
  "2016" : 5,
  "2017" : 13,
  "2018" : 12,
  "2019" : 12,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -11.1,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : -12.5,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 28.6,
  "202001(전년누계대비 증감률)" : -22.2,
  "grade" : 2.99850075,
  "경도" : 127.0246,
  "도*광역시" : "경기도",
  "발전소수량" : 50,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "수원시",
  "시군구별 건축허가현황" : 75402,
  "시군구일사량평균" : 4.002998501,
  "위도" : 37.2577
},
"20" : {
  "2015" : 5,
  "2016" : 9,
  "2017" : 18,
  "2018" : 27,
  "2019" : 19,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -11.1,
  "201601(전년누계대비 증감률)" : -12.5,
  "201701(전년누계대비 증감률)" : -12.5,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : -12.5,
  "grade" : 4.649161475,
  "경도" : 126.7229,
  "도*광역시" : "경기도",
  "발전소수량" : 78,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "시흥시",
  "시군구별 건축허가현황" : 139206,
  "시군구일사량평균" : 4.03953657,
  "위도" : 37.324
},
"21" : {
  "2015" : 6,
  "2016" : 17,
  "2017" : 20,
  "2018" : 31,
  "2019" : 18,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -12.5,
  "201601(전년누계대비 증감률)" : -14.3,
  "201701(전년누계대비 증감률)" : 16.7,
  "201801(전년누계대비 증감률)" : -14.3,
  "201901(전년누계대비 증감률)" : -16.7,
  "202001(전년누계대비 증감률)" : -100,
  "grade" : 5.471801939,
  "경도" : 126.8244,
  "도*광역시" : "경기도",
  "발전소수량" : 92,

```

```

"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "안산시",
"시군구별 건축허가현황" : 403815,
"시군구일사량평균" : 4.052389197,
"위도" : 37.3103
},
"22" : {
  "2015" : 15,
  "2016" : 14,
  "2017" : 86,
  "2018" : 246,
  "2019" : 134,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : -16.7,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 20,
  "202001(전년누계대비 증감률)" : 0,
  "grade" : 45.13386205,
  "경도" : 127.2699,
  "도*광역시" : "경기도",
  "발전소수량" : 495,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 1,
  "시군구" : "안성시",
  "시군구별 건축허가현황" : 243072,
  "시군구일사량평균" : 3.827214997,
  "위도" : 37.0006
},
"23" : {
  "2015" : 0,
  "2016" : 0,
  "2017" : 0,
  "2018" : 0,
  "2019" : 2,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -14.3,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 16.7,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : -33.3,
  "202001(전년누계대비 증감률)" : 50,
  "grade" : 0.119656107,
  "경도" : 126.9583,
  "도*광역시" : "경기도",
  "발전소수량" : 2,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "안양시",
  "시군구별 건축허가현황" : 133571,
  "시군구일사량평균" : 4.017194631,
  "위도" : 37.371
},
"24" : {
  "2015" : 6,
  "2016" : 9,
  "2017" : 30,
  "2018" : 47,
  "2019" : 57,
  "2020" : 21,
  "201501(전년누계대비 증감률)" : -12.5,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : -12.5,
  "201801(전년누계대비 증감률)" : -14.3,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : -28.6,
  "grade" : 17.6626898,
  "경도" : 127.0804,
  "도*광역시" : "경기도",
  "발전소수량" : 170,
  "산사태취약지역(개수)" : 0,

```

```

"산사태취약지역(고속도로 근처)" : 2,
"시군구" : "양주시",
"시군구별 건축허가현황" : 62920,
"시군구일사량평균" : 3.884637455,
"위도" : 37.7969
},
"25" : {
  "2015" : 17,
  "2016" : 23,
  "2017" : 79,
  "2018" : 104,
  "2019" : 65,
  "2020" : 18,
  "201501(전년누계대비 증감률)" : -12.5,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : -14.3,
  "grade" : 18.55099733,
  "경도" : 127.4855,
  "도*광역시" : "경기도",
  "발전소수량" : 306,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "양평군",
  "시군구별 건축허가현황" : 29563,
  "시군구일사량평균" : 3.937582571,
  "위도" : 37.4772
},
"26" : {
  "2015" : 53,
  "2016" : 88,
  "2017" : 236,
  "2018" : 518,
  "2019" : 103,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -12.5,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 14.3,
  "201801(전년누계대비 증감률)" : -12.5,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : 0,
  "grade" : 61.31099508,
  "경도" : 127.5451,
  "도*광역시" : "경기도",
  "발전소수량" : 998,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "여주시",
  "시군구별 건축허가현황" : 99098,
  "시군구일사량평균" : 3.856613719,
  "위도" : 37.2018
},
"27" : {
  "2015" : 11,
  "2016" : 37,
  "2017" : 88,
  "2018" : 210,
  "2019" : 113,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 16.7,
  "202001(전년누계대비 증감률)" : -14.3,
  "grade" : 42.45771092,
  "경도" : 127.0191,
  "도*광역시" : "경기도",
  "발전소수량" : 459,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 1,

```

```

"시군구": "연천군",
"시군구별 건축허가현황" : 19885,
"시군구일사량평균" : 3.737788335,
"위도" : 38.0852
},
"28" : {
  "2015" : 0,
  "2016" : 0,
  "2017" : 0,
  "2018" : 0,
  "2019" : 0,
  "2020" : 4,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : -14.3,
  "201701(전년누계대비 증감률)" : 16.7,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : -14.3,
  "grade" : 0.241837355,
  "경도" : 127.0408,
  "도*광역시" : "경기도",
  "발전소수량" : 4,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "오산시",
  "시군구별 건축허가현황" : 59336,
  "시군구일사량평균" : 3.954066127,
  "위도" : 37.1883
},
"29" : {
  "2015" : 20,
  "2016" : 19,
  "2017" : 40,
  "2018" : 63,
  "2019" : 39,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -11.1,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : -12.5,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : -14.3,
  "202001(전년누계대비 증감률)" : 16.7,
  "grade" : 10.9751137,
  "경도" : 127.1275,
  "도*광역시" : "경기도",
  "발전소수량" : 181,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "용인시",
  "시군구별 건축허가현황" : 47860,
  "시군구일사량평균" : 3.936401271,
  "위도" : 37.2724
},
"30" : {
  "2015" : 0,
  "2016" : 0,
  "2017" : 4,
  "2018" : 1,
  "2019" : 3,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : -20,
  "201701(전년누계대비 증감률)" : 12.5,
  "201801(전년누계대비 증감률)" : -11.1,
  "201901(전년누계대비 증감률)" : -12.5,
  "202001(전년누계대비 증감률)" : -14.3,
  "grade" : 0.479624389,
  "경도" : 126.9756,
  "도*광역시" : "경기도",
  "발전소수량" : 8,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "의왕시",

```

```

"시군구별 건축허가현황" : 33406,
"시군구일사량평균" : 4.004695139,
"위도" : 37.3783
},
"31" : {
  "2015" : 4,
  "2016" : 1,
  "2017" : 2,
  "2018" : 1,
  "2019" : 2,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -30,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : -14.3,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : -12.5,
  "202001(전년누계대비 증감률)" : 0,
  "grade" : 0.60771186,
  "경도" : 127.0469,
  "도*광역시" : "경기도",
  "발전소수량" : 10,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "의정부시",
  "시군구별 건축허가현황" : 39468,
  "시군구일사량평균" : 3.922881396,
  "위도" : 37.7459
},
"32" : {
  "2015" : 22,
  "2016" : 26,
  "2017" : 77,
  "2018" : 389,
  "2019" : 72,
  "2020" : 29,
  "201501(전년누계대비 증감률)" : -25,
  "201601(전년누계대비 증감률)" : 16.7,
  "201701(전년누계대비 증감률)" : -16.7,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 33.3,
  "202001(전년누계대비 증감률)" : -25,
  "grade" : 37.78530053,
  "경도" : 127.4908,
  "도*광역시" : "경기도",
  "발전소수량" : 615,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "이천시",
  "시군구별 건축허가현황" : 276849,
  "시군구일사량평균" : 3.856048695,
  "위도" : 37.2261
},
"33" : {
  "2015" : 3,
  "2016" : 0,
  "2017" : 69,
  "2018" : 131,
  "2019" : 124,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -10,
  "201601(전년누계대비 증감률)" : -11.1,
  "201701(전년누계대비 증감률)" : 12.5,
  "201801(전년누계대비 증감률)" : -11.1,
  "201901(전년누계대비 증감률)" : 25,
  "202001(전년누계대비 증감률)" : -20,
  "grade" : 20.04796096,
  "경도" : 126.7469,
  "도*광역시" : "경기도",
  "발전소수량" : 327,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "파주시",
  "시군구별 건축허가현황" : 395610,

```

```

"시군구일사량평균" : 3.869125088,
"위도" : 37.753
},
"34" : {
  "2015" : 50,
  "2016" : 21,
  "2017" : 47,
  "2018" : 108,
  "2019" : 70,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -14.3,
  "201601(전년누계대비 증감률)" : -16.7,
  "201701(전년누계대비 증감률)" : -16.7,
  "201801(전년누계대비 증감률)" : 40,
  "201901(전년누계대비 증감률)" : -16.7,
  "202001(전년누계대비 증감률)" : 0,
  "grade" : 18.2812928,
  "경도" : 127.0604,
  "도*광역시" : "경기도",
  "발전소수량" : 296,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "평택시",
  "시군구별 건축허가현황" : 123950,
  "시군구일사량평균" : 3.823887569,
  "위도" : 37.0533
},
"35" : {
  "2015" : 32,
  "2016" : 40,
  "2017" : 80,
  "2018" : 200,
  "2019" : 178,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -11.1,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : -12.5,
  "201901(전년누계대비 증감률)" : 16.7,
  "202001(전년누계대비 증감률)" : -14.3,
  "grade" : 32.95912052,
  "경도" : 127.1864,
  "도*광역시" : "경기도",
  "발전소수량" : 530,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "포천시",
  "시군구별 건축허가현황" : 47842,
  "시군구일사량평균" : 3.781298015,
  "위도" : 37.8487
},
"36" : {
  "2015" : 3,
  "2016" : 0,
  "2017" : 1,
  "2018" : 7,
  "2019" : 12,
  "2020" : 2,
  "201501(전년누계대비 증감률)" : -12.5,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 14.3,
  "202001(전년누계대비 증감률)" : -25,
  "grade" : 1.502766788,
  "경도" : 127.1617,
  "도*광역시" : "경기도",
  "발전소수량" : 25,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "하남시",
  "시군구별 건축허가현황" : 43745,
  "시군구일사량평균" : 3.988932846,

```

```

"위도" : 37.5162
},
"37" : {
  "2015" : 65,
  "2016" : 55,
  "2017" : 161,
  "2018" : 243,
  "2019" : 253,
  "2020" : 94,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : -16.7,
  "201701(전년누계대비 증감률)" : 16.7,
  "201801(전년누계대비 증감률)" : -14.3,
  "201901(전년누계대비 증감률)" : -16.7,
  "202001(전년누계대비 증감률)" : 20,
  "grade" : 77.08304461,
  "경도" : 127.0373,
  "도*광역시" : "경기도",
  "발전소수량" : 871,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 1,
  "시군구" : "화성시",
  "시군구별 건축허가현황" : 469420,
  "시군구일사량평균" : 4.008651836,
  "위도" : 37.2069
},
"38" : {
  "2015" : 6,
  "2016" : 0,
  "2017" : 0,
  "2018" : 0,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -20,
  "201601(전년누계대비 증감률)" : 50,
  "201701(전년누계대비 증감률)" : -25,
  "201801(전년누계대비 증감률)" : 66.7,
  "201901(전년누계대비 증감률)" : 40,
  "202001(전년누계대비 증감률)" : -57.1,
  "grade" : 0.543263463,
  "경도" : 128.5904,
  "도*광역시" : "경상남도",
  "발전소수량" : 6,
  "산사태취약지역(개수)" : 1,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "거제시",
  "시군구별 건축허가현황" : 192879,
  "시군구일사량평균" : 3.87024524,
  "위도" : 34.851
},
"39" : {
  "2015" : 53,
  "2016" : 1,
  "2017" : 0,
  "2018" : 0,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : -16.7,
  "grade" : 8.169893314,
  "경도" : 128.259,
  "도*광역시" : "경상남도",
  "발전소수량" : 54,
  "산사태취약지역(개수)" : 14,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "고성군",
  "시군구별 건축허가현황" : 13569,
  "시군구일사량평균" : 3.855813692,
  "위도" : 35.0987
}

```



```

},
"40" : {
  "2015" : 21,
  "2016" : 1,
  "2017" : 2,
  "2018" : 1,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -16.7,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : -20,
  "grade" : 3.145416197,
  "경도" : 128.8783,
  "도*광역시" : "경상남도",
  "발전소수량" : 25,
  "산사태취약지역(개수)" : 5,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "김해시",
  "시군구별 건축허가현황" : 94211,
  "시군구일사량평균" : 3.83672486,
  "위도" : 35.2315
},
"41" : {
  "2015" : 0,
  "2016" : 0,
  "2017" : 24,
  "2018" : 172,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : -16.7,
  "grade" : 24.93741554,
  "경도" : 127.8927,
  "도*광역시" : "경상남도",
  "발전소수량" : 196,
  "산사태취약지역(개수)" : 5,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "남해군",
  "시군구별 건축허가현황" : 6966,
  "시군구일사량평균" : 3.767406299,
  "위도" : 34.8377
},
"42" : {
  "2015" : 25,
  "2016" : 4,
  "2017" : 33,
  "2018" : 236,
  "2019" : 52,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -14.3,
  "201601(전년누계대비 증감률)" : -16.7,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 20,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : -16.7,
  "grade" : 48.06074164,
  "경도" : 128.0721,
  "도*광역시" : "경상남도",
  "발전소수량" : 350,
  "산사태취약지역(개수)" : 7,
  "산사태취약지역(고속도로 근처)" : 1,
  "시군구" : "사천시",
  "시군구별 건축허가현황" : 17480,
  "시군구일사량평균" : 3.843553046,
  "위도" : 34.9297
},
},

```

```

"43" : {
  "2015" : 3,
  "2016" : 0,
  "2017" : 12,
  "2018" : 2,
  "2019" : 1,
  "2020" : 1,
  "201501(전년누계대비 증감률)" : -20,
  "201601(전년누계대비 증감률)" : -25,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 25,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : 0,
  "grade" : 2.723887665,
  "경도" : 129.0301,
  "도*광역시" : "경상남도",
  "발전소수량" : 19,
  "산사태취약지역(개수)" : 9,
  "산사태취약지역(고속도로 근처)" : 1,
  "시군구" : "양산시",
  "시군구별 건축허가현황" : 362471,
  "시군구일사량평균" : 3.826487058,
  "위도" : 35.3249
},
"44" : {
  "2015" : 42,
  "2016" : 0,
  "2017" : 0,
  "2018" : 9,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : 0,
  "grade" : 5.795432867,
  "경도" : 128.2453,
  "도*광역시" : "경상남도",
  "발전소수량" : 51,
  "산사태취약지역(개수)" : 2,
  "산사태취약지역(고속도로 근처)" : 1,
  "시군구" : "의령군",
  "시군구별 건축허가현황" : 3130,
  "시군구일사량평균" : 3.841197806,
  "위도" : 35.3267
},
"45" : {
  "2015" : 11,
  "2016" : 12,
  "2017" : 19,
  "2018" : 42,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -33.3,
  "201601(전년누계대비 증감률)" : 16.7,
  "201701(전년누계대비 증감률)" : 60,
  "201801(전년누계대비 증감률)" : -12.5,
  "201901(전년누계대비 증감률)" : 16.7,
  "202001(전년누계대비 증감률)" : -28.6,
  "grade" : 12.36339846,
  "경도" : 128.1067,
  "도*광역시" : "경상남도",
  "발전소수량" : 84,
  "산사태취약지역(개수)" : 12,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "진주시",
  "시군구별 건축허가현황" : 27251,
  "시군구일사량평균" : 3.862093528,
  "위도" : 35.16
},
"46" : {

```

```

"2015" : 22,
"2016" : 0,
"2017" : 23,
"2018" : 88,
"2019" : 2,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 18.97641848,
"경도" : 128.5438,
"도*광역시" : "경상남도",
"발전소수량" : 135,
"산사태취약지역(개수)" : 5,
"산사태취약지역(고속도로 근처)" : 4,
"시군구" : "창녕군",
"시군구별 건축허가현황" : 18341,
"시군구일사량평균" : 3.831449667,
"위도" : 35.4981
},
"47" : {
"2015" : 1,
"2016" : 0,
"2017" : 74,
"2018" : 53,
"2019" : 0,
"2020" : 0,
"201501(전년누계대비 증감률)" : -50,
"201601(전년누계대비 증감률)" : 40,
"201701(전년누계대비 증감률)" : 25,
"201801(전년누계대비 증감률)" : 20,
"201901(전년누계대비 증감률)" : 16.7,
"202001(전년누계대비 증감률)" : -14.3,
"grade" : 20.03679239,
"경도" : 128.5624,
"도*광역시" : "경상남도",
"발전소수량" : 128,
"산사태취약지역(개수)" : 15,
"산사태취약지역(고속도로 근처)" : 2,
"시군구" : "창원시",
"시군구별 건축허가현황" : 49492,
"시군구일사량평균" : 3.846963295,
"위도" : 35.1904
},
"48" : {
"2015" : 0,
"2016" : 0,
"2017" : 0,
"2018" : 0,
"2019" : 22,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : -20,
"grade" : 2.632334881,
"경도" : 128.4068,
"도*광역시" : "경상남도",
"발전소수량" : 22,
"산사태취약지역(개수)" : 4,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "통영시",
"시군구별 건축허가현황" : 5870,
"시군구일사량평균" : 3.877341993,
"위도" : 34.8998
},
"49" : {
"2015" : 1,

```

```

"2016" : 10,
"2017" : 0,
"2018" : 0,
"2019" : 0,
"2020" : 0,
"201501(전년누계대비 증감률)" : -20,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 25,
"201801(전년누계대비 증감률)" : -20,
"201901(전년누계대비 증감률)" : 25,
"202001(전년누계대비 증감률)" : -20,
"grade" : 1.924266394,
"경도" : 127.8173,
"도*광역시" : "경상남도",
"발전소수량" : 11,
"산사태취약지역(개수)" : 31,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "하동군",
"시군구별 건축허가현황" : 6723,
"시군구일사량평균" : 3.750910065,
"위도" : 35.0221
},
"50" : {
"2015" : 4,
"2016" : 19,
"2017" : 35,
"2018" : 59,
"2019" : 88,
"2020" : 36,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : -16.7,
"grade" : 29.00633971,
"경도" : 128.4065,
"도*광역시" : "경상남도",
"발전소수량" : 241,
"산사태취약지역(개수)" : 3,
"산사태취약지역(고속도로 근처)" : 1,
"시군구" : "함안군",
"시군구별 건축허가현황" : 14232,
"시군구일사량평균" : 3.84118115,
"위도" : 35.2726
},
"51" : {
"2015" : 34,
"2016" : 7,
"2017" : 0,
"2018" : 4,
"2019" : 0,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 25,
"grade" : 7.4500601,
"경도" : 127.663,
"도*광역시" : "경상남도",
"발전소수량" : 45,
"산사태취약지역(개수)" : 23,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "함양군",
"시군구별 건축허가현황" : 7518,
"시군구일사량평균" : 3.804235477,
"위도" : 35.3969
},
"52" : {
"2015" : 23,
"2016" : 36,

```

```

"2017" : 32,
"2018" : 159,
"2019" : 102,
"2020" : 0,
"201501(전년누계대비 증감률)" : -14.3,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 21.80273236,
"경도" : 128.7279,
"도*광역시" : "경상북도",
"발전소수량" : 352,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "경산시",
"시군구별 건축허가현황" : 32486,
"시군구일사량평균" : 3.806041943,
"위도" : 35.8205
},
"53" : {
"2015" : 5,
"2016" : 16,
"2017" : 51,
"2018" : 104,
"2019" : 77,
"2020" : 11,
"201501(전년누계대비 증감률)" : -14.3,
"201601(전년누계대비 증감률)" : -16.7,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : -40,
"202001(전년누계대비 증감률)" : 33.3,
"grade" : 16.32526026,
"경도" : 129.5016,
"도*광역시" : "경상북도",
"발전소수량" : 264,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "경주시",
"시군구별 건축허가현황" : 53637,
"시군구일사량평균" : 3.816189295,
"위도" : 35.8074
},
"54" : {
"2015" : 19,
"2016" : 79,
"2017" : 192,
"2018" : 247,
"2019" : 224,
"2020" : 111,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 79.26431328,
"경도" : 128.4543,
"도*광역시" : "경상북도",
"발전소수량" : 872,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 1,
"시군구" : "고령군",
"시군구별 건축허가현황" : 6463,
"시군구일사량평균" : 3.84617603,
"위도" : 35.8258
},
"55" : {
"2015" : 27,
"2016" : 84,
"2017" : 189,

```

```

"2018" : 348,
"2019" : 0,
"2020" : 0,
"201501(전년누계대비 증감률)" : 42.9,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 20,
"201901(전년누계대비 증감률)" : -25,
"202001(전년누계대비 증감률)" : 16.7,
"grade" : 40.11100182,
"경도" : 128.3845,
"도*광역시" : "경상북도",
"발전소수량" : 648,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "구미시",
"시군구별 건축허가현황" : 27664,
"시군구일사량평균" : 3.810030583,
"위도" : 36.1014
},
"56" : {
"2015" : 53,
"2016" : 103,
"2017" : 185,
"2018" : 250,
"2019" : 145,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 45.61853056,
"경도" : 128.573,
"도*광역시" : "경상북도",
"발전소수량" : 736,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "군위군",
"시군구별 건축허가현황" : 4833,
"시군구일사량평균" : 3.801830087,
"위도" : 36.243
},
"57" : {
"2015" : 24,
"2016" : 30,
"2017" : 76,
"2018" : 1111,
"2019" : 241,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 92.2484962,
"경도" : 128.9001,
"도*광역시" : "경상북도",
"발전소수량" : 1482,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "봉화군",
"시군구별 건축허가현황" : 5439,
"시군구일사량평균" : 3.775405115,
"위도" : 36.8588
},
"58" : {
"2015" : 80,
"2016" : 90,
"2017" : 185,
"2018" : 410,

```

```

"2019" : 305,
"2020" : 71,
"201501(전년누계대비 증감률)" : -30,
"201601(전년누계대비 증감률)" : -14.3,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 33.3,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 70.76035021,
"경도" : 128.7041,
"도*광역시" : "경상북도",
"발전소수량" : 1141,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "안동시",
"시군구별 건축허가현황" : 40967,
"시군구일사량평균" : 3.798391743,
"위도" : 36.571
},
"59" : {
"2015" : 2,
"2016" : 10,
"2017" : 74,
"2018" : 559,
"2019" : 123,
"2020" : 0,
"201501(전년누계대비 증감률)" : 33.3,
"201601(전년누계대비 증감률)" : -100,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 47.62294075,
"경도" : 129.3786,
"도*광역시" : "경상북도",
"발전소수량" : 768,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "영덕군",
"시군구별 건축허가현황" : 10887,
"시군구일사량평균" : 3.799096256,
"위도" : 36.3618
},
"60" : {
"2015" : 47,
"2016" : 68,
"2017" : 121,
"2018" : 67,
"2019" : 34,
"2020" : 3,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 31.26645083,
"경도" : 128.5264,
"도*광역시" : "경상북도",
"발전소수량" : 340,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 1,
"시군구" : "예천군",
"시군구별 건축허가현황" : 30311,
"시군구일사량평균" : 3.774366918,
"위도" : 36.7212
},
"61" : {
"2015" : 4,
"2016" : 9,
"2017" : 9,
"2018" : 42,
"2019" : 52,

```

```

"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 7.204936834,
"경도" : 129.3945,
"도*광역시" : "경상북도",
"발전소수량" : 116,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "울진군",
"시군구별 건축허가현황" : 14820,
"시군구일사량평균" : 3.788847557,
"위도" : 36.9628
},
"62" : {
"2015" : 10,
"2016" : 9,
"2017" : 13,
"2018" : 39,
"2019" : 32,
"2020" : 8,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 6.865630464,
"경도" : 128.5893,
"도*광역시" : "경상북도",
"발전소수량" : 111,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "청도군",
"시군구별 건축허가현황" : 10571,
"시군구일사량평균" : 3.81474733,
"위도" : 35.6873
},
"63" : {
"2015" : 8,
"2016" : 2,
"2017" : 9,
"2018" : 37,
"2019" : 89,
"2020" : 50,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 12.09506844,
"경도" : 129.0634,
"도*광역시" : "경상북도",
"발전소수량" : 195,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "청송군",
"시군구별 건축허가현황" : 6316,
"시군구일사량평균" : 3.797400799,
"위도" : 36.3446
},
"64" : {
"2015" : 23,
"2016" : 25,
"2017" : 53,
"2018" : 48,
"2019" : 38,
"2020" : 18,

```



```

"201501(전년누계대비 증감률)" : -14.3,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 66.7,
"201901(전년누계대비 증감률)" : -16.7,
"202001(전년누계대비 증감률)" : 0,
"grade" : 33.20086635,
"경도" : 126.7448,
"도*광역시" : "광주광역시",
"발전소수량" : 205,
"산사태취약지역(개수)" : 18,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "광산구",
"시군구별 건축허가현황" : 205769,
"시군구일사량평균" : 3.693819139,
"위도" : 35.2196
},
"65" : {
  "2015" : 1,
  "2016" : 6,
  "2017" : 7,
  "2018" : 7,
  "2019" : 3,
  "2020" : 2,
  "201501(전년누계대비 증감률)" : -30,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : -14.3,
  "201801(전년누계대비 증감률)" : 16.7,
  "201901(전년누계대비 증감률)" : -14.3,
  "202001(전년누계대비 증감률)" : -16.7,
  "grade" : 3.87386848,
  "경도" : 126.9157,
  "도*광역시" : "광주광역시",
  "발전소수량" : 26,
  "산사태취약지역(개수)" : 11,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "남구",
  "시군구별 건축허가현황" : 10171,
  "시군구일사량평균" : 3.691273851,
  "위도" : 35.1369
},
"66" : {
  "2015" : 0,
  "2016" : 2,
  "2017" : 4,
  "2018" : 6,
  "2019" : 2,
  "2020" : 1,
  "201501(전년누계대비 증감률)" : -12.5,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : 0,
  "grade" : 2.457843771,
  "경도" : 126.9196,
  "도*광역시" : "광주광역시",
  "발전소수량" : 15,
  "산사태취약지역(개수)" : 19,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "동구",
  "시군구별 건축허가현황" : 326491,
  "시군구일사량평균" : 3.676103214,
  "위도" : 35.1585
},
"67" : {
  "2015" : 20,
  "2016" : 15,
  "2017" : 48,
  "2018" : 51,
  "2019" : 54,
  "2020" : 5,
  "201501(전년누계대비 증감률)" : -16.7,

```

```

"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : -25,
"201801(전년누계대비 증감률)" : 66.7,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 32.7492385,
"경도" : 126.9369,
"도*광역시" : "광주광역시",
"발전소수량" : 193,
"산사태취약지역(개수)" : 23,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "북구",
"시군구별 건축허가현황" : 29542,
"시군구일사량평균" : 3.649739545,
"위도" : 35.1815
},
"68" : {
"2015" : 12,
"2016" : 10,
"2017" : 11,
"2018" : 21,
"2019" : 26,
"2020" : 3,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 11.37155109,
"경도" : 126.8903,
"도*광역시" : "광주광역시",
"발전소수량" : 83,
"산사태취약지역(개수)" : 7,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "서구",
"시군구별 건축허가현황" : 12005,
"시군구일사량평균" : 3.704003036,
"위도" : 35.152
},
"69" : {
"2015" : 5,
"2016" : 5,
"2017" : 7,
"2018" : 14,
"2019" : 20,
"2020" : 16,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : -30,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : -14.3,
"201901(전년누계대비 증감률)" : 33.3,
"202001(전년누계대비 증감률)" : -25,
"grade" : 4.269541994,
"경도" : 127.4276,
"도*광역시" : "대전광역시",
"발전소수량" : 67,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "대덕구",
"시군구별 건축허가현황" : 21291,
"시군구일사량평균" : 3.627549262,
"위도" : 36.3719
},
"70" : {
"2015" : 2,
"2016" : 3,
"2017" : 13,
"2018" : 6,
"2019" : 13,
"2020" : 8,
"201501(전년누계대비 증감률)" : -10,
"201601(전년누계대비 증감률)" : -22.2,

```

```

"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 2.876950594,
"경도" : 127.4414,
"도*광역시" : "대전광역시",
"발전소수량" : 45,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "동구",
"시군구별 건축허가현황" : 14004,
"시군구일사량평균" : 3.606776458,
"위도" : 36.347
},
"71" : {
  "2015" : 3,
  "2016" : 9,
  "2017" : 23,
  "2018" : 21,
  "2019" : 22,
  "2020" : 11,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : -25,
  "201701(전년누계대비 증감률)" : -16.7,
  "201801(전년누계대비 증감률)" : 20,
  "201901(전년누계대비 증감률)" : -16.7,
  "202001(전년누계대비 증감률)" : 20,
  "grade" : 5.658398071,
  "경도" : 127.3494,
  "도*광역시" : "대전광역시",
  "발전소수량" : 89,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "서구",
  "시군구별 건축허가현황" : 63949,
  "시군구일사량평균" : 3.642249359,
  "위도" : 36.3036
},
"72" : {
  "2015" : 4,
  "2016" : 5,
  "2017" : 16,
  "2018" : 20,
  "2019" : 19,
  "2020" : 9,
  "201501(전년누계대비 증감률)" : -16.7,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : -20,
  "grade" : 4.659994134,
  "경도" : 127.3596,
  "도*광역시" : "대전광역시",
  "발전소수량" : 73,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "유성구",
  "시군구별 건축허가현황" : 261509,
  "시군구일사량평균" : 3.616446392,
  "위도" : 36.3926
},
"73" : {
  "2015" : 3,
  "2016" : 8,
  "2017" : 11,
  "2018" : 13,
  "2019" : 10,
  "2020" : 6,
  "201501(전년누계대비 증감률)" : -11.1,
  "201601(전년누계대비 증감률)" : -25,
  "201701(전년누계대비 증감률)" : 33.3,

```

```

"201801(전년누계대비 증감률)" : -25,
"201901(전년누계대비 증감률)" : 16.7,
"202001(전년누계대비 증감률)" : -14.3,
"grade" : 3.248614049,
"경도" : 127.3953,
"도*광역시" : "대전광역시",
"발전소수량" : 51,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "충구",
"시군구별 건축허가현황" : 364791,
"시군구일사량평균" : 3.630168532,
"위도" : 36.315
},
"74" : {
  "2015" : 11,
  "2016" : 5,
  "2017" : 27,
  "2018" : 53,
  "2019" : 72,
  "2020" : 19,
  "201501(전년누계대비 증감률)" : -25,
  "201601(전년누계대비 증감률)" : 66.7,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : -20,
  "201901(전년누계대비 증감률)" : 25,
  "202001(전년누계대비 증감률)" : 0,
  "grade" : 16.99463558,
  "경도" : 128.9018,
  "도*광역시" : "부산광역시",
  "발전소수량" : 187,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 1,
  "시군구" : "강서구",
  "시군구별 건축허가현황" : 70463,
  "시군구일사량평균" : 3.847464864,
  "위도" : 35.1964
},
"75" : {
  "2015" : 0,
  "2016" : 0,
  "2017" : 3,
  "2018" : 5,
  "2019" : 3,
  "2020" : 2,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 25,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : -20,
  "grade" : 1.565415759,
  "경도" : 129.1115,
  "도*광역시" : "부산광역시",
  "발전소수량" : 13,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 4,
  "시군구" : "금정구",
  "시군구별 건축허가현황" : 9275,
  "시군구일사량평균" : 3.83819581,
  "위도" : 35.2162
},
"76" : {
  "2015" : 5,
  "2016" : 1,
  "2017" : 4,
  "2018" : 12,
  "2019" : 20,
  "2020" : 5,
  "201501(전년누계대비 증감률)" : -50,
  "201601(전년누계대비 증감률)" : 66.7,
  "201701(전년누계대비 증감률)" : -40,
  "201801(전년누계대비 증감률)" : 33.3,

```

```

"201901(전년누계대비 증감률)" : -25,
"202001(전년누계대비 증감률)" : 0,
"grade" : 4.281249214,
"경도" : 129.2222,
"도*광역시" : "부산광역시",
"발전소수량" : 47,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 1,
"시군구" : "기장군",
"시군구별 건축허가현황" : 43460,
"시군구일사량평균" : 3.83324771,
"위도" : 35.2444
},
"77" : {
  "2015" : 0,
  "2016" : 1,
  "2017" : 0,
  "2018" : 1,
  "2019" : 1,
  "2020" : 5,
  "201501(전년누계대비 증감률)" : -20,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : -25,
  "202001(전년누계대비 증감률)" : 0,
  "grade" : 0.49212138,
  "경도" : 129.0453,
  "도*광역시" : "부산광역시",
  "발전소수량" : 8,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "동구",
  "시군구별 건축허가현황" : 13261,
  "시군구일사량평균" : 3.848482755,
  "위도" : 35.1293
},
"78" : {
  "2015" : 1,
  "2016" : 1,
  "2017" : 3,
  "2018" : 2,
  "2019" : 0,
  "2020" : 1,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : 0,
  "grade" : 0.492624628,
  "경도" : 129.0801,
  "도*광역시" : "부산광역시",
  "발전소수량" : 8,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "동래구",
  "시군구별 건축허가현황" : 24614,
  "시군구일사량평균" : 3.842192147,
  "위도" : 35.2201
},
"79" : {
  "2015" : 1,
  "2016" : 0,
  "2017" : 1,
  "2018" : 1,
  "2019" : 1,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -25,
  "201601(전년누계대비 증감률)" : 33.3,
  "201701(전년누계대비 증감률)" : -25,
  "201801(전년누계대비 증감률)" : 33.3,
  "201901(전년누계대비 증감률)" : -25,

```

```

"202001(전년누계대비 증감률)" : 33.3,
"grade" : 0.246206218,
"경도" : 129.029,
"도*광역시" : "부산광역시",
"발전소수량" : 4,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "부산진구",
"시군구별 건축허가현황" : 60837,
"시군구일사량평균" : 3.844844544,
"위도" : 35.1497
},
"80" : {
  "2015" : 0,
  "2016" : 0,
  "2017" : 1,
  "2018" : 2,
  "2019" : 0,
  "2020" : 1,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : -16.7,
  "201701(전년누계대비 증감률)" : -20,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 25,
  "202001(전년누계대비 증감률)" : 20,
  "grade" : 0.246324397,
  "경도" : 129.0138,
  "도*광역시" : "부산광역시",
  "발전소수량" : 4,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "북구",
  "시군구별 건축허가현황" : 933,
  "시군구일사량평균" : 3.841890067,
  "위도" : 35.2328
},
"81" : {
  "2015" : 0,
  "2016" : 1,
  "2017" : 4,
  "2018" : 8,
  "2019" : 19,
  "2020" : 2,
  "201501(전년누계대비 증감률)" : -20,
  "201601(전년누계대비 증감률)" : -25,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 33.3,
  "201901(전년누계대비 증감률)" : -25,
  "202001(전년누계대비 증감률)" : 0,
  "grade" : 2.09207522,
  "경도" : 128.9872,
  "도*광역시" : "부산광역시",
  "발전소수량" : 34,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "사상구",
  "시군구별 건축허가현황" : 34949,
  "시군구일사량평균" : 3.846837587,
  "위도" : 35.1638
},
"82" : {
  "2015" : 0,
  "2016" : 0,
  "2017" : 4,
  "2018" : 10,
  "2019" : 16,
  "2020" : 1,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : -20,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 50,
  "202001(전년누계대비 증감률)" : -16.7,

```

```

"grade" : 2.811371564,
"경도" : 129.0099,
"도*광역시" : "부산광역시",
"발전소수량" : 31,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 1,
"시군구" : "사하구",
"시군구별 건축허가현황" : 35465,
"시군구일사량평균" : 3.860395346,
"위도" : 35.0984
},
"83" : {
"2015" : 0,
"2016" : 0,
"2017" : 1,
"2018" : 1,
"2019" : 1,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 0.184647885,
"경도" : 129.1147,
"도*광역시" : "부산광역시",
"발전소수량" : 3,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "수영구",
"시군구별 건축허가현황" : 31054,
"시군구일사량평균" : 3.845070486,
"위도" : 35.1438
},
"84" : {
"2015" : 0,
"2016" : 0,
"2017" : 2,
"2018" : 0,
"2019" : 1,
"2020" : 0,
"201501(전년누계대비 증감률)" : -20,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 25,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 25,
"grade" : 0.184678911,
"경도" : 129.0761,
"도*광역시" : "부산광역시",
"발전소수량" : 3,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "연제구",
"시군구별 건축허가현황" : 20745,
"시군구일사량평균" : 3.844036309,
"위도" : 35.1845
},
"85" : {
"2015" : 1,
"2016" : 0,
"2017" : 0,
"2018" : 3,
"2019" : 1,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 25,
"201801(전년누계대비 증감률)" : -40,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 33.3,
"grade" : 0.307359787,

```

```

"경도" : 129.0447,
"도*광역시" : "부산광역시",
"발전소수량" : 5,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "영도구",
"시군구별 건축허가현황" : 5533,
"시군구일사량평균" : 3.852804264,
"위도" : 35.0902
},
"86" : {
"2015" : 6,
"2016" : 0,
"2017" : 2,
"2018" : 2,
"2019" : 1,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : -42.9,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : -25,
"202001(전년누계대비 증감률)" : 0,
"grade" : 0.677380416,
"경도" : 129.1298,
"도*광역시" : "부산광역시",
"발전소수량" : 11,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "해운대구",
"시군구별 건축허가현황" : 16038,
"시군구일사량평균" : 3.841996221,
"위도" : 35.2197
},
"87" : {
"2015" : 2,
"2016" : 1,
"2017" : 0,
"2018" : 0,
"2019" : 0,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 16.7,
"202001(전년누계대비 증감률)" : 0,
"grade" : 0.180333529,
"경도" : 127.0737,
"도*광역시" : "서울특별시",
"발전소수량" : 3,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "강남구",
"시군구별 건축허가현황" : 176216,
"시군구일사량평균" : 3.988882372,
"위도" : 37.4921
},
"88" : {
"2015" : 1,
"2016" : 2,
"2017" : 3,
"2018" : 0,
"2019" : 0,
"2020" : 0,
"201501(전년누계대비 증감률)" : -12.5,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : -14.3,
"grade" : 0.360979445,
"경도" : 127.1329,

```



```

"도*광역시" : "서울특별시",
"발전소수량" : 6,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "강동구",
"시군구별 건축허가현황" : 191886,
"시군구일사량평균" : 3.983675912,
"위도" : 37.526
},
"89" : {
"2015" : 3,
"2016" : 0,
"2017" : 1,
"2018" : 0,
"2019" : 0,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : -14.3,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 16.7,
"202001(전년누계대비 증감률)" : -14.3,
"grade" : 0.241542838,
"경도" : 127.0232,
"도*광역시" : "서울특별시",
"발전소수량" : 4,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "강북구",
"시군구별 건축허가현황" : 16907,
"시군구일사량평균" : 3.961429044,
"위도" : 37.6387
},
"90" : {
"2015" : 1,
"2016" : 0,
"2017" : 4,
"2018" : 0,
"2019" : 0,
"2020" : 0,
"201501(전년누계대비 증감률)" : -37.5,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 20,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 16.7,
"202001(전년누계대비 증감률)" : 0,
"grade" : 0.300093409,
"경도" : 126.8384,
"도*광역시" : "서울특별시",
"발전소수량" : 5,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "강서구",
"시군구별 건축허가현황" : 50649,
"시군구일사량평균" : 3.998131811,
"위도" : 37.5425
},
"91" : {
"2015" : 1,
"2016" : 4,
"2017" : 2,
"2018" : 1,
"2019" : 0,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : -28.6,
"201901(전년누계대비 증감률)" : 40,
"202001(전년누계대비 증감률)" : 0,
"grade" : 0.479058892,
"경도" : 126.9515,
"도*광역시" : "서울특별시",

```

```

"발전소수량" : 8,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "관악구",
"시군구별 건축허가현황" : 31073,
"시군구일사량평균" : 4.011763854,
"위도" : 37.4782
},
"92" : {
"2015" : 3,
"2016" : 4,
"2017" : 3,
"2018" : 2,
"2019" : 0,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 16.7,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 42.9,
"202001(전년누계대비 증감률)" : -20,
"grade" : 0.72241525,
"경도" : 127.092,
"도*광역시" : "서울특별시",
"발전소수량" : 12,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "광진구",
"시군구별 건축허가현황" : 19378,
"시군구일사량평균" : 3.979872917,
"위도" : 37.538
},
"93" : {
"2015" : 2,
"2016" : 2,
"2017" : 1,
"2018" : 1,
"2019" : 0,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : -14.3,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : -16.7,
"202001(전년누계대비 증감률)" : 0,
"grade" : 0.359353745,
"경도" : 126.8875,
"도*광역시" : "서울특별시",
"발전소수량" : 6,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "구로구",
"시군구별 건축허가현황" : 80011,
"시군구일사량평균" : 4.010770914,
"위도" : 37.4955
},
"94" : {
"2015" : 2,
"2016" : 0,
"2017" : 2,
"2018" : 0,
"2019" : 0,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 0.239267083,
"경도" : 126.8918,
"도*광역시" : "서울특별시",
"발전소수량" : 4,

```

```

"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "금천구",
"시군구별 건축허가현황" : 43983,
"시군구일사량평균" : 4.018322917,
"위도" : 37.4769
},
"95" : {
  "2015" : 2,
  "2016" : 0,
  "2017" : 4,
  "2018" : 0,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -25,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 14.3,
  "201801(전년누계대비 증감률)" : -12.5,
  "201901(전년누계대비 증감률)" : 14.3,
  "202001(전년누계대비 증감률)" : -12.5,
  "grade" : 0.362243415,
  "경도" : 127.068,
  "도*광역시" : "서울특별시",
  "발전소수량" : 6,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "노원구",
  "시군구별 건축허가현황" : 10376,
  "시군구일사량평균" : 3.96260975,
  "위도" : 37.632
},
"96" : {
  "2015" : 5,
  "2016" : 2,
  "2017" : 1,
  "2018" : 0,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -12.5,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 16.7,
  "201901(전년누계대비 증감률)" : 28.6,
  "202001(전년누계대비 증감률)" : -22.2,
  "grade" : 0.48367819,
  "경도" : 127.0261,
  "도*광역시" : "서울특별시",
  "발전소수량" : 8,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "도봉구",
  "시군구별 건축허가현황" : 5864,
  "시군구일사량평균" : 3.954022619,
  "위도" : 37.648
},
"97" : {
  "2015" : 2,
  "2016" : 0,
  "2017" : 3,
  "2018" : 1,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -11.1,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : -11.1,
  "201801(전년누계대비 증감률)" : -25,
  "201901(전년누계대비 증감률)" : 33.3,
  "202001(전년누계대비 증감률)" : -25,
  "grade" : 0.362532312,
  "경도" : 127.0706,
  "도*광역시" : "서울특별시",
  "발전소수량" : 6,
  "산사태취약지역(개수)" : 0,

```

```

"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "동대문구",
"시군구별 건축허가현황" : 35773,
"시군구일사량평균" : 3.957794792,
"위도" : 37.5785
},
"98" : {
  "2015" : 0,
  "2016" : 1,
  "2017" : 0,
  "2018" : 0,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 16.7,
  "201801(전년누계대비 증감률)" : -14.3,
  "201901(전년누계대비 증감률)" : 16.7,
  "202001(전년누계대비 증감률)" : -14.3,
  "grade" : 0.060017951,
  "경도" : 126.9561,
  "도*광역시" : "서울특별시",
  "발전소수량" : 1,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "동작구",
  "시군구별 건축허가현황" : 99967,
  "시군구일사량평균" : 3.998204924,
  "위도" : 37.5131
},
"99" : {
  "2015" : 3,
  "2016" : 1,
  "2017" : 0,
  "2018" : 0,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -12.5,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 33.3,
  "201801(전년누계대비 증감률)" : -12.5,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : -28.6,
  "grade" : 0.240573533,
  "경도" : 126.9499,
  "도*광역시" : "서울특별시",
  "발전소수량" : 4,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "마포구",
  "시군구별 건축허가현황" : 26125,
  "시군구일사량평균" : 3.985661667,
  "위도" : 37.5415
},
"100" : {
  "2015" : 3,
  "2016" : 1,
  "2017" : 4,
  "2018" : 0,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -25,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : -14.3,
  "201801(전년누계대비 증감률)" : -100,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : -18.2,
  "grade" : 0.481898391,
  "경도" : 126.9342,
  "도*광역시" : "서울특별시",
  "발전소수량" : 8,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,

```

```

"시군구" : "서대문구",
"시군구별 건축허가현황" : 9921,
"시군구일사량평균" : 3.976270112,
"위도" : 37.5801
},
"101" : {
  "2015" : 1,
  "2016" : 0,
  "2017" : 0,
  "2018" : 0,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 40,
  "201801(전년누계대비 증감률)" : -14.3,
  "201901(전년누계대비 증감률)" : -16.7,
  "202001(전년누계대비 증감률)" : 0,
  "grade" : 0.060019987,
  "경도" : 127.0413,
  "도*광역시" : "서울특별시",
  "발전소수량" : 1,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "서초구",
  "시군구별 건축허가현황" : 27754,
  "시군구일사량평균" : 3.998001344,
  "위도" : 37.4706
},
"102" : {
  "2015" : 0,
  "2016" : 0,
  "2017" : 2,
  "2018" : 2,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 16.7,
  "202001(전년누계대비 증감률)" : -14.3,
  "grade" : 0.242933465,
  "경도" : 127.0369,
  "도*광역시" : "서울특별시",
  "발전소수량" : 4,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "성동구",
  "시군구별 건축허가현황" : 22336,
  "시군구일사량평균" : 3.926663368,
  "위도" : 37.5634
},
"103" : {
  "2015" : 3,
  "2016" : 6,
  "2017" : 1,
  "2018" : 0,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : -12.5,
  "201801(전년누계대비 증감률)" : -14.3,
  "201901(전년누계대비 증감률)" : 66.7,
  "202001(전년누계대비 증감률)" : -20,
  "grade" : 0.6036745,
  "경도" : 127.0214,
  "도*광역시" : "서울특별시",
  "발전소수량" : 10,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "성북구",

```

```

"시군구별 건축허가현황" : 38606,
"시군구일사량평균" : 3.963255,
"위도" : 37.5858
},
"104" : {
  "2015" : 3,
  "2016" : 5,
  "2017" : 3,
  "2018" : 0,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 12.5,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 33.3,
  "201801(전년누계대비 증감률)" : -25,
  "201901(전년누계대비 증감률)" : 33.3,
  "202001(전년누계대비 증감률)" : -12.5,
  "grade" : 0.661063077,
  "경도" : 127.1468,
  "도*광역시" : "서울특별시",
  "발전소수량" : 11,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "송파구",
  "시군구별 건축허가현황" : 86799,
  "시군구일사량평균" : 3.990335667,
  "위도" : 37.4936
},
"105" : {
  "2015" : 2,
  "2016" : 0,
  "2017" : 3,
  "2018" : 0,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -12.5,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 14.3,
  "202001(전년누계대비 증감률)" : -12.5,
  "grade" : 0.299739067,
  "경도" : 126.8315,
  "도*광역시" : "서울특별시",
  "발전소수량" : 5,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "양천구",
  "시군구별 건축허가현황" : 14989,
  "시군구일사량평균" : 4.005218663,
  "위도" : 37.5328
},
"106" : {
  "2015" : 0,
  "2016" : 0,
  "2017" : 1,
  "2018" : 1,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 12.5,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 28.6,
  "201801(전년누계대비 증감률)" : -22.2,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : 28.6,
  "grade" : 0.120182862,
  "경도" : 126.9346,
  "도*광역시" : "서울특별시",
  "발전소수량" : 2,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "영등포구",
  "시군구별 건축허가현황" : 175774,

```

```

"시군구일사량평균" : 3.990856924,
"위도" : 37.5177
},
"107" : {
  "2015" : 1,
  "2016" : 3,
  "2017" : 2,
  "2018" : 1,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 40,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 28.6,
  "202001(전년누계대비 증감률)" : -11.1,
  "grade" : 0.622900961,
  "경도" : 126.9167,
  "도*광역시" : "서울특별시",
  "발전소수량" : 7,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 1,
  "시군구" : "은평구",
  "시군구별 건축허가현황" : 20356,
  "시군구일사량평균" : 3.975724653,
  "위도" : 37.6237
},
"108" : {
  "2015" : 1,
  "2016" : 0,
  "2017" : 1,
  "2018" : 0,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -20,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 75,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 14.3,
  "202001(전년누계대비 증감률)" : -12.5,
  "grade" : 0.120985362,
  "경도" : 127.003,
  "도*광역시" : "서울특별시",
  "발전소수량" : 2,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "종로구",
  "시군구별 건축허가현황" : 187549,
  "시군구일사량평균" : 3.95073192,
  "위도" : 37.5718
},
"109" : {
  "2015" : 1,
  "2016" : 2,
  "2017" : 0,
  "2018" : 1,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 16.7,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 16.7,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : 0,
  "grade" : 0.244155125,
  "경도" : 127.0214,
  "도*광역시" : "서울특별시",
  "발전소수량" : 4,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "종구",
  "시군구별 건축허가현황" : 181651,
  "시군구일사량평균" : 3.896121875,

```

```

"위도" : 37.5674
},
"110" : {
  "2015" : 4,
  "2016" : 2,
  "2017" : 5,
  "2018" : 0,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : -14.3,
  "201801(전년누계대비 증감률)" : 16.7,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : 0,
  "grade" : 0.662877655,
  "경도" : 127.0785,
  "도*광역시" : "서울특별시",
  "발전소수량" : 11,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "중랑구",
  "시군구별 건축허가현황" : 24456,
  "시군구일사량평균" : 3.973839497,
  "위도" : 37.6124
},
"111" : {
  "2015" : 56,
  "2016" : 41,
  "2017" : 123,
  "2018" : 78,
  "2019" : 68,
  "2020" : 12,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : -25,
  "201801(전년누계대비 증감률)" : 16.7,
  "201901(전년누계대비 증감률)" : 14.3,
  "202001(전년누계대비 증감률)" : 12.5,
  "grade" : 24.09307618,
  "경도" : 127.2652,
  "도*광역시" : "충청남도",
  "발전소수량" : 378,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "세종시",
  "시군구별 건축허가현황" : 73640,
  "시군구일사량평균" : 3.626170322,
  "위도" : 36.4979
},
"112" : {
  "2015" : 2,
  "2016" : 1,
  "2017" : 4,
  "2018" : 2,
  "2019" : 2,
  "2020" : 11,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : -10,
  "201701(전년누계대비 증감률)" : 11.1,
  "201801(전년누계대비 증감률)" : -20,
  "201901(전년누계대비 증감률)" : 12.5,
  "202001(전년누계대비 증감률)" : 0,
  "grade" : 1.311613109,
  "경도" : 126.7134,
  "도*광역시" : "인천광역시",
  "발전소수량" : 22,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "남동구",
  "시군구별 건축허가현황" : 38971,
  "시군구일사량평균" : 4.038122231,
  "위도" : 37.4561
}

```



```

},
"113" : {
  "2015" : 2,
  "2016" : 1,
  "2017" : 4,
  "2018" : 2,
  "2019" : 2,
  "2020" : 10,
  "201501(전년누계대비 증감률)" : -10,
  "201601(전년누계대비 증감률)" : -11.1,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : 0,
  "grade" : 1.253320313,
  "경도" : 126.7401,
  "도*광역시" : "인천광역시",
  "발전소수량" : 21,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "동구",
  "시군구별 건축허가현황" : 19054,
  "시군구일사량평균" : 4.031808036,
  "위도" : 37.4485
},
"114" : {
  "2015" : 0,
  "2016" : 9,
  "2017" : 14,
  "2018" : 1,
  "2019" : 1,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 33.3,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : -25,
  "201801(전년누계대비 증감률)" : 166.7,
  "201901(전년누계대비 증감률)" : -37.5,
  "202001(전년누계대비 증감률)" : -20,
  "grade" : 1.500083913,
  "경도" : 126.1447,
  "도*광역시" : "인천광역시",
  "발전소수량" : 25,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "옹진군",
  "시군구별 건축허가현황" : 5070,
  "시군구일사량평균" : 3.99966435,
  "위도" : 37.2277
},
"115" : {
  "2015" : 3,
  "2016" : 1,
  "2017" : 1,
  "2018" : 2,
  "2019" : 1,
  "2020" : 1,
  "201501(전년누계대비 증감률)" : -14.3,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 16.7,
  "201901(전년누계대비 증감률)" : 14.3,
  "202001(전년누계대비 증감률)" : -12.5,
  "grade" : 0.536391662,
  "경도" : 126.4613,
  "도*광역시" : "인천광역시",
  "발전소수량" : 9,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "중구",
  "시군구별 건축허가현황" : 14739,
  "시군구일사량평균" : 4.040092641,
  "위도" : 37.4404
},

```

```

"116" : {
  "2015" : 4,
  "2016" : 6,
  "2017" : 24,
  "2018" : 2,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -33.3,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : -20,
  "201801(전년누계대비 증감률)" : -100,
  "201901(전년누계대비 증감률)" : 20,
  "202001(전년누계대비 증감률)" : -16.7,
  "grade" : 2.290581782,
  "경도" : 127.5811,
  "도*광역시" : "전라남도",
  "발전소수량" : 36,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "광양시",
  "시군구별 건축허가현황" : 10330,
  "시군구일사량평균" : 3.637272828,
  "위도" : 34.9733
},
"117" : {
  "2015" : 6,
  "2016" : 11,
  "2017" : 7,
  "2018" : 3,
  "2019" : 0,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : 0,
  "grade" : 2.908442237,
  "경도" : 127.5438,
  "도*광역시" : "전라남도",
  "발전소수량" : 27,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 2,
  "시군구" : "구례군",
  "시군구별 건축허가현황" : 6532,
  "시군구일사량평균" : 3.659683069,
  "위도" : 35.1782
},
"118" : {
  "2015" : 54,
  "2016" : 120,
  "2017" : 587,
  "2018" : 934,
  "2019" : 311,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : -50,
  "grade" : 127.8596045,
  "경도" : 126.6081,
  "도*광역시" : "전라남도",
  "발전소수량" : 2006,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "나주시",
  "시군구별 건축허가현황" : 79582,
  "시군구일사량평균" : 3.626141353,
  "위도" : 34.9433
},
"119" : {

```

```

"2015" : 8,
"2016" : 11,
"2017" : 847,
"2018" : 688,
"2019" : 88,
"2020" : 30,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 106.9905736,
"경도" : 126.3946,
"도*광역시" : "전라남도",
"발전소수량" : 1672,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "무안군",
"시군구별 건축허가현황" : 199137,
"시군구일사량평균" : 3.601042251,
"위도" : 35.0129
},
"120" : {
"2015" : 16,
"2016" : 18,
"2017" : 229,
"2018" : 676,
"2019" : 142,
"2020" : 1,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 145.7327544,
"경도" : 127.1518,
"도*광역시" : "전라남도",
"발전소수량" : 1082,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 6,
"시군구" : "보성군",
"시군구별 건축허가현황" : 9550,
"시군구일사량평균" : 3.628575117,
"위도" : 34.8295
},
"121" : {
"2015" : 2,
"2016" : 0,
"2017" : 45,
"2018" : 1641,
"2019" : 0,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 108.2438805,
"경도" : 125.9534,
"도*광역시" : "전라남도",
"발전소수량" : 1688,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "신안군",
"시군구별 건축허가현황" : 8456,
"시군구일사량평균" : 3.587447838,
"위도" : 34.7036
},
"122" : {
"2015" : 28,

```

```

"2016" : 41,
"2017" : 74,
"2018" : 465,
"2019" : 85,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : -20,
"grade" : 75.03974632,
"경도" : 126.9389,
"도*광역시" : "전라남도",
"발전소수량" : 693,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 2,
"시군구" : "장흥군",
"시군구별 건축허가현황" : 21875,
"시군구일사량평균" : 3.626581873,
"위도" : 34.5616
},
"123" : {
"2015" : 1,
"2016" : 7,
"2017" : 172,
"2018" : 255,
"2019" : 81,
"2020" : 0,
"201501(전년누계대비 증감률)" : 16.7,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 75,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : -42.9,
"202001(전년누계대비 증감률)" : 25,
"grade" : 67.26711771,
"경도" : 126.6786,
"도*광역시" : "전라북도",
"발전소수량" : 516,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 5,
"시군구" : "고창군",
"시군구별 건축허가현황" : 15551,
"시군구일사량평균" : 3.614034567,
"위도" : 35.4099
},
"124" : {
"2015" : 22,
"2016" : 15,
"2017" : 41,
"2018" : 92,
"2019" : 70,
"2020" : 0,
"201501(전년누계대비 증감률)" : 100,
"201601(전년누계대비 증감률)" : -10,
"201701(전년누계대비 증감률)" : 16.7,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 14.3,
"202001(전년누계대비 증감률)" : -25,
"grade" : 15.17636444,
"경도" : 126.7965,
"도*광역시" : "전라북도",
"발전소수량" : 240,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "군산시",
"시군구별 건축허가현황" : 23885,
"시군구일사량평균" : 3.676514816,
"위도" : 35.9584
},
"125" : {
"2015" : 7,
"2016" : 41,

```

```

"2017" : 291,
"2018" : 215,
"2019" : 65,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : -28.6,
"201901(전년누계대비 증감률)" : -20,
"202001(전년누계대비 증감률)" : 25,
"grade" : 104.1000779,
"경도" : 126.9037,
"도*광역시" : "전라북도",
"발전소수량" : 619,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 20,
"시군구" : "김제시",
"시군구별 건축허가현황" : 70269,
"시군구일사량평균" : 3.563394063,
"위도" : 35.8032
},
"126" : {
"2015" : 5,
"2016" : 33,
"2017" : 258,
"2018" : 429,
"2019" : 78,
"2020" : 0,
"201501(전년누계대비 증감률)" : -11.1,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : -12.5,
"201801(전년누계대비 증감률)" : 14.3,
"201901(전년누계대비 증감률)" : -25,
"202001(전년누계대비 증감률)" : -16.7,
"grade" : 50.5034035,
"경도" : 127.3754,
"도*광역시" : "전라북도",
"발전소수량" : 803,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "남원시",
"시군구별 건축허가현황" : 34682,
"시군구일사량평균" : 3.710659589,
"위도" : 35.4024
},
"127" : {
"2015" : 2,
"2016" : 1,
"2017" : 31,
"2018" : 40,
"2019" : 19,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : -16.7,
"grade" : 5.880405658,
"경도" : 127.6615,
"도*광역시" : "전라북도",
"발전소수량" : 93,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "무주군",
"시군구별 건축허가현황" : 5018,
"시군구일사량평균" : 3.676983164,
"위도" : 36.0101
},
"128" : {
"2015" : 8,
"2016" : 20,
"2017" : 213,

```

```

"2018" : 157,
"2019" : 327,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 40,
"201901(전년누계대비 증감률)" : -14.3,
"202001(전년누계대비 증감률)" : 0,
"grade" : 46.55635642,
"경도" : 126.6981,
"도*광역시" : "전라북도",
"발전소수량" : 725,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "부안군",
"시군구별 건축허가현황" : 15171,
"시군구일사량평균" : 3.578433597,
"위도" : 35.7617
},
"129" : {
"2015" : 0,
"2016" : 6,
"2017" : 43,
"2018" : 134,
"2019" : 16,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : -20,
"grade" : 12.63922331,
"경도" : 127.1023,
"도*광역시" : "전라북도",
"발전소수량" : 199,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "순창군",
"시군구별 건축허가현황" : 55768,
"시군구일사량평균" : 3.648631501,
"위도" : 35.453
},
"130" : {
"2015" : 13,
"2016" : 27,
"2017" : 65,
"2018" : 127,
"2019" : 31,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 20,
"grade" : 24.89859986,
"경도" : 127.2526,
"도*광역시" : "전라북도",
"발전소수량" : 263,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 1,
"시군구" : "완주군",
"시군구별 건축허가현황" : 12069,
"시군구일사량평균" : 3.590811718,
"위도" : 36.0226
},
"131" : {
"2015" : 15,
"2016" : 19,
"2017" : 113,
"2018" : 236,

```

```

"2019" : 71,
"2020" : 0,
"201501(전년누계대비 증감률)" : -50,
"201601(전년누계대비 증감률)" : 50,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : -16.7,
"201901(전년누계대비 증감률)" : -25,
"202001(전년누계대비 증감률)" : -16.7,
"grade" : 28.93688247,
"경도" : 126.9726,
"도*광역시" : "전라북도",
"발전소수량" : 454,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "익산시",
"시군구별 건축허가현황" : 94031,
"시군구일사량평균" : 3.626237341,
"위도" : 35.9523
},
"132" : {
"2015" : 2,
"2016" : 8,
"2017" : 191,
"2018" : 126,
"2019" : 20,
"2020" : 0,
"201501(전년누계대비 증감률)" : -20,
"201601(전년누계대비 증감률)" : -25,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : -100,
"201901(전년누계대비 증감률)" : 33.3,
"202001(전년누계대비 증감률)" : -100,
"grade" : 22.02448282,
"경도" : 127.1636,
"도*광역시" : "전라북도",
"발전소수량" : 347,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "임실군",
"시군구별 건축허가현황" : 18945,
"시군구일사량평균" : 3.652886794,
"위도" : 35.5303
},
"133" : {
"2015" : 4,
"2016" : 6,
"2017" : 128,
"2018" : 169,
"2019" : 7,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 19.81619748,
"경도" : 127.5777,
"도*광역시" : "전라북도",
"발전소수량" : 314,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "장수군",
"시군구별 건축허가현황" : 29338,
"시군구일사량평균" : 3.689109083,
"위도" : 35.7029
},
"134" : {
"2015" : 2,
"2016" : 5,
"2017" : 16,
"2018" : 21,
"2019" : 13,

```

```

"2020" : 0,
"201501(전년누계대비 증감률)" : -16.7,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : -16.7,
"201801(전년누계대비 증감률)" : 20,
"201901(전년누계대비 증감률)" : -11.1,
"202001(전년누계대비 증감률)" : -100,
"grade" : 7.138318788,
"경도" : 127.1431,
"도*광역시" : "전라북도",
"발전소수량" : 57,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 4,
"시군구" : "전주시",
"시군구별 건축허가현황" : 46203,
"시군구일사량평균" : 3.591703059,
"위도" : 35.8386
},
"135" : {
"2015" : 14,
"2016" : 61,
"2017" : 479,
"2018" : 319,
"2019" : 79,
"2020" : 0,
"201501(전년누계대비 증감률)" : -42.9,
"201601(전년누계대비 증감률)" : 25,
"201701(전년누계대비 증감률)" : 60,
"201801(전년누계대비 증감률)" : -12.5,
"201901(전년누계대비 증감률)" : -28.6,
"202001(전년누계대비 증감률)" : 20,
"grade" : 89.87244861,
"경도" : 126.9285,
"도*광역시" : "전라북도",
"발전소수량" : 952,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 1,
"시군구" : "정읍시",
"시군구별 건축허가현황" : 14777,
"시군구일사량평균" : 3.608931408,
"위도" : 35.7149
},
"136" : {
"2015" : 2,
"2016" : 16,
"2017" : 51,
"2018" : 78,
"2019" : 23,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : -33.3,
"grade" : 10.82633615,
"경도" : 127.5679,
"도*광역시" : "전라북도",
"발전소수량" : 170,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "진안군",
"시군구별 건축허가현황" : 11030,
"시군구일사량평균" : 3.631566971,
"위도" : 35.8382
},
"137" : {
"2015" : 81,
"2016" : 174,
"2017" : 307,
"2018" : 547,
"2019" : 11,
"2020" : 6,

```



```

"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 75,
"202001(전년누계대비 증감률)" : -14.3,
"grade" : 175.7308204,
"경도" : 127.5277,
"도*광역시" : "충청남도",
"발전소수량" : 1126,
"산사태취약지역(개수)" : 12,
"산사태취약지역(고속도로 근처)" : 1,
"시군구" : "금산군",
"시군구별 건축허가현황" : 16909,
"시군구일사량평균" : 3.581114837,
"위도" : 36.1689
},
"138" : {
"2015" : 170,
"2016" : 82,
"2017" : 375,
"2018" : 497,
"2019" : 318,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 28.6,
"202001(전년누계대비 증감률)" : -33.3,
"grade" : 251.6471184,
"경도" : 127.018,
"도*광역시" : "충청남도",
"발전소수량" : 1442,
"산사태취약지역(개수)" : 27,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "논산시",
"시군구별 건축허가현황" : 24020,
"시군구일사량평균" : 3.631771246,
"위도" : 36.1581
},
"139" : {
"2015" : 0,
"2016" : 0,
"2017" : 0,
"2018" : 254,
"2019" : 151,
"2020" : 0,
"201501(전년누계대비 증감률)" : 42.9,
"201601(전년누계대비 증감률)" : -50,
"201701(전년누계대비 증감률)" : 100,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : -22.2,
"grade" : 57.77072478,
"경도" : 126.5969,
"도*광역시" : "충청남도",
"발전소수량" : 405,
"산사태취약지역(개수)" : 11,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "당진시",
"시군구별 건축허가현황" : 24154,
"시군구일사량평균" : 3.96019462,
"위도" : 36.926
},
"140" : {
"2015" : 6,
"2016" : 16,
"2017" : 6,
"2018" : 46,
"2019" : 4,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,

```

```

"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : -28.6,
"grade" : 12.92329477,
"경도" : 126.605,
"도*광역시" : "충청남도",
"발전소수량" : 78,
"산사태취약지역(개수)" : 20,
"산사태취약지역(고속도로 근처)" : 1,
"시군구" : "보령시",
"시군구별 건축허가현황" : 14326,
"시군구일사량평균" : 3.708553224,
"위도" : 36.3046
},
"141" : {
"2015" : 26,
"2016" : 53,
"2017" : 93,
"2018" : 89,
"2019" : 173,
"2020" : 121,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : -22.2,
"201801(전년누계대비 증감률)" : -14.3,
"201901(전년누계대비 증감률)" : 33.3,
"202001(전년누계대비 증감률)" : -25,
"grade" : 70.30699732,
"경도" : 126.8774,
"도*광역시" : "충청남도",
"발전소수량" : 555,
"산사태취약지역(개수)" : 2,
"산사태취약지역(고속도로 근처)" : 3,
"시군구" : "아산시",
"시군구별 건축허가현황" : 36983,
"시군구일사량평균" : 3.794468092,
"위도" : 36.7575
},
"142" : {
"2015" : 10,
"2016" : 49,
"2017" : 139,
"2018" : 243,
"2019" : 284,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 60,
"202001(전년누계대비 증감률)" : -37.5,
"grade" : 113.5554582,
"경도" : 126.7732,
"도*광역시" : "충청남도",
"발전소수량" : 725,
"산사태취약지역(개수)" : 16,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "예산군",
"시군구별 건축허가현황" : 18414,
"시군구일사량평균" : 3.780927089,
"위도" : 36.5489
},
"143" : {
"2015" : 18,
"2016" : 17,
"2017" : 30,
"2018" : 71,
"2019" : 72,
"2020" : 21,
"201501(전년누계대비 증감률)" : 42.9,
"201601(전년누계대비 증감률)" : 0,

```

```

"201701(전년누계대비 증감률)" : -20,
"201801(전년누계대비 증감률)" : -25,
"201901(전년누계대비 증감률)" : 28.6,
"202001(전년누계대비 증감률)" : -33.3,
"grade" : 38.75052332,
"경도" : 127.1608,
"도*광역시" : "충청남도",
"발전소수량" : 229,
"산사태취약지역(개수)" : 24,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "천안시",
"시군구별 건축허가현황" : 30814,
"시군구일사량평균" : 3.70989125,
"위도" : 36.8014
},
"144" : {
  "2015" : 14,
  "2016" : 2,
  "2017" : 9,
  "2018" : 9,
  "2019" : 47,
  "2020" : 40,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 50,
  "202001(전년누계대비 증감률)" : -16.7,
  "grade" : 16.26943545,
  "경도" : 126.7761,
  "도*광역시" : "충청남도",
  "발전소수량" : 121,
  "산사태취약지역(개수)" : 6,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "청양군",
  "시군구별 건축허가현황" : 6010,
  "시군구일사량평균" : 3.639463229,
  "위도" : 36.4013
},
"145" : {
  "2015" : 173,
  "2016" : 152,
  "2017" : 200,
  "2018" : 618,
  "2019" : 357,
  "2020" : 93,
  "201501(전년누계대비 증감률)" : 0,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : 0,
  "201901(전년누계대비 증감률)" : 40,
  "202001(전년누계대비 증감률)" : -28.6,
  "grade" : 226.0259474,
  "경도" : 126.5501,
  "도*광역시" : "충청남도",
  "발전소수량" : 1593,
  "산사태취약지역(개수)" : 9,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "홍성군",
  "시군구별 건축허가현황" : 17979,
  "시군구일사량평균" : 3.77348334,
  "위도" : 36.6052
},
"146" : {
  "2015" : 25,
  "2016" : 46,
  "2017" : 10,
  "2018" : 60,
  "2019" : 24,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -20,
  "201601(전년누계대비 증감률)" : 0,
  "201701(전년누계대비 증감률)" : 25,

```

```

"201801(전년누계대비 증감률)" : -20,
"201901(전년누계대비 증감률)" : 25,
"202001(전년누계대비 증감률)" : 0,
"grade" : 15.51232354,
"경도" : 127.875,
"도*광역시" : "충청북도",
"발전소수량" : 165,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 1,
"시군구" : "괴산군",
"시군구별 건축허가현황" : 15698,
"시군구일사량평균" : 3.635317226,
"위도" : 36.8367
},
"147" : {
"2015" : 38,
"2016" : 34,
"2017" : 34,
"2018" : 50,
"2019" : 28,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 11.70450208,
"경도" : 127.6641,
"도*광역시" : "충청북도",
"발전소수량" : 184,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "보은군",
"시군구별 건축허가현황" : 4696,
"시군구일사량평균" : 3.638857564,
"위도" : 36.5779
},
"148" : {
"2015" : 0,
"2016" : 0,
"2017" : 0,
"2018" : 0,
"2019" : 82,
"2020" : 2,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 20,
"grade" : 5.343921008,
"경도" : 127.9321,
"도*광역시" : "충청북도",
"발전소수량" : 84,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "영동군",
"시군구별 건축허가현황" : 18635,
"시군구일사량평균" : 3.638189276,
"위도" : 36.1909
},
"149" : {
"2015" : 58,
"2016" : 60,
"2017" : 69,
"2018" : 104,
"2019" : 55,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,

```

```

"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : -28.6,
"grade" : 22.10421226,
"경도" : 127.5344,
"도*광역시" : "충청북도",
"발전소수량" : 346,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "옥천군",
"시군구별 건축허가현황" : 18736,
"시군구일사량평균" : 3.611499346,
"위도" : 36.3302
},
"150" : {
"2015" : 22,
"2016" : 20,
"2017" : 27,
"2018" : 37,
"2019" : 62,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : 0,
"grade" : 15.82289902,
"경도" : 127.6436,
"도*광역시" : "충청북도",
"발전소수량" : 168,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 1,
"시군구" : "음성군",
"시군구별 건축허가현황" : 62699,
"시군구일사량평균" : 3.623819138,
"위도" : 37.1123
},
"151" : {
"2015" : 5,
"2016" : 19,
"2017" : 14,
"2018" : 22,
"2019" : 11,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 0,
"201901(전년누계대비 증감률)" : 0,
"202001(전년누계대비 증감률)" : -28.6,
"grade" : 4.541885082,
"경도" : 127.613,
"도*광역시" : "충청북도",
"발전소수량" : 71,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "증평군",
"시군구별 건축허가현황" : 61173,
"시군구일사량평균" : 3.602978757,
"위도" : 36.8134
},
"152" : {
"2015" : 23,
"2016" : 22,
"2017" : 27,
"2018" : 75,
"2019" : 28,
"2020" : 0,
"201501(전년누계대비 증감률)" : 0,
"201601(전년누계대비 증감률)" : 0,
"201701(전년누계대비 증감률)" : 0,
"201801(전년누계대비 증감률)" : 33.3,
"201901(전년누계대비 증감률)" : 12.5,

```

```

"202001(전년누계대비 증감률)" : -22.2,
"grade" : 11.13565688,
"경도" : 127.4357,
"도*광역시" : "충청북도",
"발전소수량" : 175,
"산사태취약지역(개수)" : 0,
"산사태취약지역(고속도로 근처)" : 0,
"시군구" : "진천군",
"시군구별 건축허가현황" : 27805,
"시군구일사량평균" : 3.636767498,
"위도" : 36.9934
},
"153" : {
  "2015" : 1,
  "2016" : 2,
  "2017" : 6,
  "2018" : 1,
  "2019" : 3,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : 16.7,
  "201601(전년누계대비 증감률)" : -14.3,
  "201701(전년누계대비 증감률)" : 14.3,
  "201801(전년누계대비 증감률)" : -12.5,
  "201901(전년누계대비 증감률)" : 40,
  "202001(전년누계대비 증감률)" : -28.6,
  "grade" : 0.828811937,
  "경도" : 127.5488,
  "도*광역시" : "충청북도",
  "발전소수량" : 13,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 0,
  "시군구" : "청주시",
  "시군구별 건축허가현황" : 280481,
  "시군구일사량평균" : 3.624523561,
  "위도" : 36.5536
},
"154" : {
  "2015" : 43,
  "2016" : 15,
  "2017" : 112,
  "2018" : 163,
  "2019" : 46,
  "2020" : 0,
  "201501(전년누계대비 증감률)" : -18.2,
  "201601(전년누계대비 증감률)" : -11.1,
  "201701(전년누계대비 증감률)" : 0,
  "201801(전년누계대비 증감률)" : -12.5,
  "201901(전년누계대비 증감률)" : 0,
  "202001(전년누계대비 증감률)" : -25,
  "grade" : 49.15234148,
  "경도" : 127.9387,
  "도*광역시" : "충청북도",
  "발전소수량" : 379,
  "산사태취약지역(개수)" : 0,
  "산사태취약지역(고속도로 근처)" : 5,
  "시군구" : "충주시",
  "시군구별 건축허가현황" : 35010,
  "시군구일사량평균" : 3.647005973,
  "위도" : 36.9642
},
"StarNum" : {
  "2020-11-23 13:12:26" : {
    "starEdit" : 3
  },
  "2020-11-24 00:07:21" : {
    "starEdit" : 4
  },
  "2020-11-24 03:13:55" : {
    "starEdit" : 3
  },
  "2020-11-24 03:20:41" : {
    "starEdit" : 4
  }
},

```

```

"2020-11-24 04:47:12" : {
  "starEdit" : 3
},
"2020-11-25 21:02:40" : {
  "starEdit" : 2
},
"2020-11-26 17:42:30" : {
  "starEdit" : 2
},
"2020-11-26 22:06:01" : {
  "starEdit" : 4
},
"2020-12-03 13:17:23" : {
  "starEdit" : 3
},
"2020-12-04 04:02:48" : {
  "starEdit" : 2
},
"2020-12-04 04:17:20" : {
  "starEdit" : 4
},
"2020-12-13 15:14:17" : {
  "starEdit" : 2
}
},
"feedback" : {
  "2020-11-24 03:26:04" : {
    "contents" : "ㅋㅋㅋㅋㅋㅋ 아 젠장",
    "title" : "ㅋㅋㅋㅋ"
  },
  "2020-11-24 04:47:12" : {
    "contents" : "ㅋㅋㅋㅋ",
    "title" : "ㅎㅎ"
  },
  "2020-11-24 19:13:59" : {
    "contents" : "나야 권호",
    "title" : "안녕 준호여"
  },
  "2020-11-26 22:02:03" : {
    "contents" : "Marry X-Mas",
    "title" : "Yo ho ho"
  },
  "2020-11-26 22:06:01" : {
    "contents" : "Merry X-MAS",
    "title" : "Ah Yo hoho"
  },
  "2020-11-28 15:58:33" : {
    "contents" : "ㅠㅠ우우우에에에에에에에에",
    "title" : "우에에에에에에에에에에에에에"
  },
  "2020-12-03 13:17:23" : {
    "contents" : "World!",
    "title" : "Hello"
  },
  "2020-12-03 16:51:24" : {
    "contents" : "",
    "title" : ""
  },
  "2020-12-04 04:02:48" : {
    "contents" : "싫어!",
    "title" : "코로나"
  },
  "2020-12-04 04:17:20" : {
    "contents" : "싫어!",
    "title" : "코로나"
  },
  "2020-12-10 15:10:40" : {
    "contents" : "ㅋㅋㅋㅋㅋㅋ",
    "title" : "ㅋㅋ"
  },
  "아" : {
    "contents" : "코로나",
    "formateDate" : "2020-12-13 15:14:17",

```

[illegible]



[illegible]



[illegible]

## 메인

```
package com.App.SolarPing;

import android.content.Context;
import android.content.Intent;
import android.database.Cursor;
import android.location.Address;
import android.location.Geocoder;
```

```

import android.os.Bundle;
import android.view.View;

import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.Button;
import android.widget.ListPopupWindow;
import android.widget.SearchView;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.fragment.app.FragmentActivity;

import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.Marker;
import com.google.android.gms.maps.model.MarkerOptions;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.io.IOException;
import java.lang.reflect.Field;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Objects;

public class MainActivity extends FragmentActivity implements OnMapReadyCallback
{
    private GoogleMap mMap;          // 구글맵 뷰에 필요한 변수
    SupportMapFragment mapFragment;  // 구글맵 뷰에 필요한 변수
    private SearchView searchView;    // 구글맵 뷰 위의 검색창에 필요한 변수
    List<Marker> AllMarkers = new ArrayList<Marker>(); // 구글맵 마커 저장을
    위한 리스트 변수

    citiesDatabase cities_db;
    GraphDatabase graph_db;

    // 플로팅 버튼에 필요한 변수
    private Context mContext;
    private FloatingActionButton add_btn, ranking_btn, feedback_btn;
    private TextView ranking_text, feedback_text;
    private Animation rotate_open_anim, rotate_close_anim, from_bottom_anim,
    to_bottom_anim;
    private boolean isFabOpen = false;

```

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    searchView = findViewById(R.id.sv_location);

    /* 플로팅 버튼 관련 */
    mContext = getApplicationContext();
    rotate_open_anim = AnimationUtils.loadAnimation(mContext,
R.anim.rotate_open_anim);
    rotate_close_anim = AnimationUtils.loadAnimation(mContext,
R.anim.rotate_close_anim);

    add_btn = (FloatingActionButton) findViewById(R.id.add_btn);
    ranking_btn = (FloatingActionButton) findViewById(R.id.ranking_btn);
    feedback_btn = (FloatingActionButton) findViewById(R.id.feedback_btn);
    ranking_text = (TextView) findViewById(R.id.ranking_text);
    feedback_text = (TextView) findViewById(R.id.feedback_text);
    /* 플로팅 버튼 관련 */

    cities_db = new citiesDatabase(getApplicationContext());
    graph_db = new GraphDatabase(getApplicationContext());
    cities_db.open();
    graph_db.open();

    Button btnallremove = (Button) findViewById(R.id.btn_allremove);

    try {
        // 맵을 가져옵니다
        mapFragment = (SupportMapFragment)
getSupportFragmentManager().findFragmentById(R.id.map);
        searchView.setOnQueryTextListener(new
SearchView.OnQueryTextListener() {
            @Override
            public boolean onQueryTextSubmit(String s) {
                String location = searchView.getQuery().toString();
                // 검색창 커지는 것 방지
                searchView.setIconified(false); // 실행 시 자동으로

                List<Address> addressList = null;

                if (location != null || !location.equals("")) {
                    Geocoder geocoder = new Geocoder(MainActivity.this);
                    try {
                        addressList = geocoder.getFromLocationName(location,
1);

                        searchView.clearFocus();
                    } catch (IOException e) {
                        e.printStackTrace();
                    }
                    try {
                        Address address = addressList.get(0);
                        LatLng latLng = new LatLng((address.getLatitude()),
address.getLongitude());

                        mMap.animateCamera(CameraUpdateFactory.newLatLngZoom(latLng, 11));
                    } catch (Exception e) {
                        Toast.makeText(getApplicationContext(), "지역이
없습니다.", Toast.LENGTH_LONG).show();

```

```

        }
    }
    return false;
}

@Override
public boolean onQueryTextChange(String newText) {
    return false;
}

});
mapFragment.getMapAsync(this);
SupportMapFragment mapFragment = (SupportMapFragment)
getSupportFragmentManager().findFragmentById(R.id.map);
mapFragment.getMapAsync(this);
} catch (Exception ignored) {
}

/* 플로팅 버튼 관련 */
add_btn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        toggleFab();
    }
});

ranking_btn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent = new Intent(getApplicationContext(),
RankRecommend.class);
        startActivity(intent);
        toggleFab();
    }
});

feedback_btn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent(MainActivity.this,
SecondScreen.class);
        startActivity(intent);
        toggleFab();
    }
});

/* 플로팅 버튼 관련 */

// Spinner 등급 콤보박스 관련 코드
Spinner gradeSpinner = (Spinner) findViewById(R.id.spinner_grade);
ArrayAdapter gradeAdapter = ArrayAdapter.createFromResource(this,
R.array.spinner_grade, android.R.layout.simple_spinner_item);

gradeAdapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_i
tem);
gradeSpinner.setAdapter(gradeAdapter);

// Spinner 지역 콤보박스 관련 코드
Spinner locationSpinner = (Spinner) findViewById(R.id.spinner_location);
ArrayAdapter locationAdapter = ArrayAdapter.createFromResource(this,

```

```

R.array.spinner_location, android.R.layout.simple_spinner_item);

locationAdapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
locationSpinner.setAdapter(locationAdapter);

gradeSpinner.setOnItemSelectedListener(new
AdapterView.OnItemSelectedListener() {
    @Override
    public void onItemSelected(AdapterView<?> adapterView, View view, int
position, long id) {
        String grade;
        grade = adapterView.getItemAtPosition(position).toString();
// 콤보박스 문자열 가져오기
        removeAllMarkers();
        createMarker(grade);
    }

    @Override
    public void onNothingSelected(AdapterView<?> adapterView) {

    }
});

locationSpinner.setOnItemSelectedListener(new
AdapterView.OnItemSelectedListener() {
    @Override
    public void onItemSelected(AdapterView<?> adapterView, View view, int
position, long id) {
        String location;
        location = adapterView.getItemAtPosition(position).toString();
        try {
            Field popup = Spinner.class.getDeclaredField("mPopup");
            popup.setAccessible(true);

            ListPopupWindow window = (ListPopupWindow)
popup.get(locationSpinner);
            window.setHeight(1150);
        } catch (NoSuchFieldException | IllegalAccessException e) {
            e.printStackTrace();
        }

        removeAllMarkers();
        createMarker(location);
    }

    @Override
    public void onNothingSelected(AdapterView<?> adapterView) {

    }
});

btnallremove.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {

        createMarker("지도 초기화");
    }
});

```

```

        gradeSpinner.setSelection(0);
        locationSpinner.setSelection(0);
    }
});
}

GoogleMap.OnInfoWindowClickListener markerClickListener = marker -> {
    String markerId = marker.getTitle();
    //선택한 타겟위치
    Intent intent = new Intent(MainActivity.this, GraphActivity.class);
    intent.putExtra("region", markerId);
    intent.putExtra("response", "0");
    startActivity(intent);
};

// 맵에 대한 작업을 세우는 곳(마커같은 기능)
@Override
public void onMapReady(GoogleMap googleMap) {
    mMap = googleMap;
    googleMap.setPadding(0, 400, 0, 0); // 나침반 위치를 패딩으로
    옮깁니다.
    mMap.getUiSettings().setMapToolbarEnabled(false); // 구글맵 툴바를
    없앱니다.
    this.mMap.setOnInfoWindowClickListener(markerClickListener);

    // 지도상 대한민국 중심, 지도 위 경도 기반 위치 가져오기(무주군 위치)
    LatLng center = new LatLng(35.965500, 127.885751);
    mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center, (float) 6.9));
    // 해당 지역으로 카메라 이동

    Thread loading = new Thread() {
        @Override
        public void run() {
            super.run();
            Intent intent = new Intent(MainActivity.this, Loading.class);
            intent.putExtra("activity", "main");
            startActivity(intent);
        }
    };

}

// 검색창 밑 도,광역시 버튼 클릭 시 String location 매개변수로 구분하여 마커를 찍는
메소드입니다.
private void createMarker(String location) {
    Cursor selecttable = cities_db.selectColumns();
    for (int i = 0; i < selecttable.getCount(); i++) {
        selecttable.moveToPosition(i);
        String columns_do = selecttable.getString(1);
        String columns_sigungu = selecttable.getString(2);
        String columns_locationinfo = selecttable.getString(3);

        double D_latitude =
        Double.parseDouble(Objects.requireNonNull(columns_locationinfo.split(" ")[0]));
        double D_longitude =
        Double.parseDouble(Objects.requireNonNull(columns_locationinfo.split(" ")[1]));
        double D_sunshineduration =
        Double.parseDouble(Objects.requireNonNull(columns_locationinfo.split(" ")[2]));

```



```

        double D_stationNum =
Double.parseDouble(Objects.requireNonNull(columns_locationinfo.split(" ")[3]));
        double D_hazard = Double.parseDouble(columns_locationinfo.split("
")[4]);
        double D_architecture =
Double.parseDouble(Objects.requireNonNull(columns_locationinfo.split(" ")[5]));
        String city_do = columns_do;
        String city_sigungu = columns_sigungu;

        // 등급 수식입니다. 계산된 값이 높을수록
        double grade = 55 - (((((10 - D_sunshineduration) *
Math.log10(D_stationNum) + 1) / 100)
        * (1 + Math.log10(1 + Math.pow(D_hazard, 2))) *
Math.log10(D_architecture)) * 10);

        // 지도 위.경도 기반 위치 가져오기
        LatLng latLng = new LatLng(D_latitude, D_longitude);

        // 지역 마커 객체 생성, 위치, 제목, 더보기 내용 설정
        MarkerOptions markerOptions = new MarkerOptions();
        markerOptions.position(latLng);
        markerOptions.title(city_do + " " + city_sigungu);
        markerOptions.snippet(D_latitude + ", " + D_longitude);

        // 등급마다 마커 표시를 하는 등급 콤보박스 번호들
        switch (location) {
            case "1등급":
                // 대한민국 지도상 중심 (무주군 위치)
                LatLng center = new LatLng(35.965500, 127.885751);
                mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 6.9));
                // 해당 지역으로 카메라 이동
                for (int j = 0; j < 154; j++) {
                    // 초록색3 1등급
                    if (grade >= 0 && grade < 20.0) {
markerOptions.icon(BitmapDescriptorFactory.defaultMarker(120f));
                        // 구글맵에 설정된 마커 추가
                        Marker locationMarker =
mMap.addMarker(markerOptions); // 구글맵에 설정된 마커 추가
                        AllMarkers.add(locationMarker); // 배열에 마커 추가
                    } else break;
                }
                break;
            case "2등급":
                // 대한민국 지도상 중심 (무주군 위치)
                center = new LatLng(35.965500, 127.885751);
                mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 6.9));
                // 해당 지역으로 카메라 이동
                for (int j = 0; j < 154; j++) {
                    // 초록색2 2등급
                    if (grade >= 20.0 && grade < 30.0) {
markerOptions.icon(BitmapDescriptorFactory.defaultMarker(100f));
                        // 구글맵에 설정된 마커 추가
                        Marker locationMarker =
mMap.addMarker(markerOptions); // 구글맵에 설정된 마커 추가
                        AllMarkers.add(locationMarker); // 배열에 마커 추가
                    } else break;
                }
            }

```

```

        }
        break;
    case "3등급":
        // 대한민국 지도상 중심 (무주군 위치)
        center = new LatLng(35.965500, 127.885751);
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 6.9));
        // 해당 지역으로 카메라 이동
        for (int j = 0; j < 154; j++) {
            // 초록색1 3등급
            if (grade >= 30.0 && grade < 40.0) {

markerOptions.setIcon(BitmapDescriptorFactory.defaultMarker(80f));
                // 구글맵에 설정된 마커 추가
                Marker locationMarker =
mMap.addMarker(markerOptions); // 구글맵에 설정된 마커 추가
                AllMarkers.add(locationMarker); // 배열에 마커 추가
            } else break;
        }
        break;
    case "4등급":
        // 대한민국 지도상 중심 (무주군 위치)
        center = new LatLng(35.965500, 127.885751);
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 6.9));
        // 해당 지역으로 카메라 이동
        for (int j = 0; j < 154; j++) {
            // 주황색2 4등급
            if (grade >= 40.0 && grade < 45.0) {

markerOptions.setIcon(BitmapDescriptorFactory.defaultMarker(50f));
                // 구글맵에 설정된 마커 추가
                Marker locationMarker =
mMap.addMarker(markerOptions); // 구글맵에 설정된 마커 추가
                AllMarkers.add(locationMarker); // 배열에 마커 추가
            } else break;
        }
        break;
    case "5등급":
        // 대한민국 지도상 중심 (무주군 위치)
        center = new LatLng(35.965500, 127.885751);
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 6.9));
        // 해당 지역으로 카메라 이동
        for (int j = 0; j < 154; j++) {
            // 주황색1 5등급
            if (grade >= 45.0 && grade < 50.0) {

markerOptions.setIcon(BitmapDescriptorFactory.defaultMarker(20f));
                // 구글맵에 설정된 마커 추가
                Marker locationMarker =
mMap.addMarker(markerOptions); // 구글맵에 설정된 마커 추가
                AllMarkers.add(locationMarker); // 배열에 마커 추가
            } else break;
        }
        break;
    case "6등급":
        // 대한민국 지도상 중심 (무주군 위치)
        center = new LatLng(35.965500, 127.885751);
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 6.9));
        // 해당 지역으로 카메라 이동
        for (int j = 0; j < 154; j++) {

```

```

        // 빨간색3 6등급
        if (grade >= 50.0 && grade < 52.0) {

markerOptions.icon(BitmapDescriptorFactory.defaultMarker(10f));
            // 구글맵에 설정된 마커 추가
            Marker locationMarker =
mMap.addMarker(markerOptions); // 구글맵에 설정된 마커 추가
            AllMarkers.add(locationMarker); // 배열에 마커 추가
        } else break;
    }
    break;
    case "7등급":
        // 대한민국 지도상 중심 (무주군 위치)
        center = new LatLng(35.965500, 127.885751);
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 6.9)); // 해당 지역으로 카메라 이동
        for (int j = 0; j < 154; j++) {
            // 빨간색2 7등급
            if (grade >= 52.0 && grade < 54.0) {

markerOptions.icon(BitmapDescriptorFactory.defaultMarker(5f));
                // 구글맵에 설정된 마커 추가
                Marker locationMarker =
mMap.addMarker(markerOptions); // 구글맵에 설정된 마커 추가
                AllMarkers.add(locationMarker); // 배열에 마커 추가
            } else break;
        }
        break;
    case "8등급":
        // 대한민국 지도상 중심 (무주군 위치)
        center = new LatLng(35.965500, 127.885751);
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 6.9)); // 해당 지역으로 카메라 이동
        for (int j = 0; j < 154; j++) {
            // 빨간색1 8등급
            if (grade >= 54.0 && grade < 56.0) {

markerOptions.icon(BitmapDescriptorFactory.defaultMarker(0f));
                // 구글맵에 설정된 마커 추가
                Marker locationMarker =
mMap.addMarker(markerOptions); // 구글맵에 설정된 마커 추가
                AllMarkers.add(locationMarker); // 배열에 마커 추가
            } else break;
        }
        break;
    }

    if (city_do.equals(location)) {
        switch (location) {
            case "서울특별시": {
                // 지도상 서울특별시 중심, 지도 위 정도 기반 위치
가져오기 (서울특별시 마포구로 초기화 되있는 상태)
                LatLng center = new LatLng(37.5415, 126.9499);
                mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 9)); // 해당 지역으로 카메라 이동

                break;
            }
        }
    }

```

```

    }
    case "대전광역시": {
        // 지도상 대전광역시 중심, 지도 위 경도 기반 위치
        가져오기 (대전광역시 중구로 초기화 되있는 상태)
        LatLng center = new LatLng(36.315, 127.3953);
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 9)); // 해당 지역으로 카메라 이동

        break;
    }
    case "광주광역시": {
        // 지도상 광주광역시 중심, 지도 위 경도 기반 위치
        가져오기 (광주광역시 서구로 초기화 되있는 상태)
        LatLng center = new LatLng(35.152, 126.8903);
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 9)); // 해당 지역으로 카메라 이동

        break;
    }
    case "부산광역시": {
        // 지도상 부산광역시 중심, 지도 위 경도 기반 위치
        가져오기 (부산광역시 연제구로 초기화 되있는 상태)
        LatLng center = new LatLng(35.1845, 129.0761);
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 9)); // 해당 지역으로 카메라 이동

        break;
    }
    case "인천광역시": {
        // 지도상 인천광역시 중심, 지도 위 경도 기반 위치
        가져오기 (인천광역시 중구로 초기화 되있는 상태)
        LatLng center = new LatLng(37.4404, 126.4613);
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 9)); // 해당 지역으로 카메라 이동

        break;
    }
    case "경기도": {
        // 지도상 경기도 중심, 지도 위 경도 기반 위치 가져오기 (경기도
        구리시로 초기화 되있는 상태)
        LatLng center = new LatLng(37.5964, 127.1492);
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 8.0)); // 해당 지역으로 카메라 이동

        break;
    }
    case "강원도": {
        // 지도상 강원도 중심, 지도 위 경도 기반 위치 가져오기 (강원도
        평창군으로 초기화 되있는 상태)
        LatLng center = new LatLng(37.6736, 128.7061);
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 8.0)); // 해당 지역으로 카메라 이동

        break;
    }

```

```

    }
    case "충청남도": {
        // 지도상 충청남도 중심, 지도 위 경도 기반 위치 가져오기 (충청남도
아산시로 초기화 되있는 상태)
        LatLng center = new LatLng(36.7575, 126.8774);
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 8.0));
        // 해당 지역으로 카메라 이동

        break;
    }
    case "충청북도": {
        // 지도상 충청북도 중심, 지도 위 경도 기반 위치 가져오기 (충청북도
청주시로 초기화 되있는 상태)
        LatLng center = new LatLng(36.5536, 127.5488);
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 8.0));
        // 해당 지역으로 카메라 이동

        break;
    }
    case "전라남도": {
        // 지도상 전라남도 중심, 지도 위 경도 기반 위치 가져오기 (전라남도
보성군으로 초기화 되있는 상태)
        LatLng center = new LatLng(34.8295, 127.1518);
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 8.0));
        // 해당 지역으로 카메라 이동

        break;
    }
    case "전라북도": {
        // 지도상 전라북도 중심, 지도 위 경도 기반 위치 가져오기 (전라북도
전주시로 초기화 되있는 상태)
        LatLng center = new LatLng(35.8386, 127.1431);
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 8.0));
        // 해당 지역으로 카메라 이동

        break;
    }
    case "경상남도": {
        // 지도상 경상남도 중심, 지도 위 경도 기반 위치 가져오기 (경상남도
창원시로 초기화 되있는 상태)
        LatLng center = new LatLng(35.1904, 128.5624);
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 8.0));
        // 해당 지역으로 카메라 이동

        break;
    }
    case "경상북도": {
        // 지도상 경상북도 중심, 지도 위 경도 기반 위치 가져오기 (경상북도
청송군으로 초기화 되있는 상태)
        LatLng center = new LatLng(36.3446, 129.0634);
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center,
(float) 8.0));
        // 해당 지역으로 카메라 이동

```

```

        break;
    }
}
for (int j = 0; j < 154; j++) {
    // 초록색3 1등급
    if (grade >= 0 && grade < 20.0)

markerOptions.icon(BitmapDescriptorFactory.defaultMarker(120f));
    // 초록색2 2등급
    else if (grade >= 20.0 && grade < 30.0)

markerOptions.icon(BitmapDescriptorFactory.defaultMarker(100f));
    // 초록색1 3등급
    else if (grade >= 30.0 && grade < 40.0)

markerOptions.icon(BitmapDescriptorFactory.defaultMarker(80f));
    // 주황색2 4등급
    else if (grade >= 40.0 && grade < 45.0)

markerOptions.icon(BitmapDescriptorFactory.defaultMarker(50f));
    // 주황색1 5등급
    else if (grade >= 45.0 && grade < 50.0)

markerOptions.icon(BitmapDescriptorFactory.defaultMarker(20f));
    // 빨간색3 6등급
    else if (grade >= 50.0 && grade < 52.0)

markerOptions.icon(BitmapDescriptorFactory.defaultMarker(10f));
    // 빨간색2 7등급
    else if (grade >= 52.0 && grade < 54.0)

markerOptions.icon(BitmapDescriptorFactory.defaultMarker(5f));
    // 빨간색1 8등급
    else

markerOptions.icon(BitmapDescriptorFactory.defaultMarker(0f));

    // 구글맵에 설정된 마커 추가
    Marker locationMarker = mMap.addMarker(markerOptions);
// 구글맵에 설정된 마커 추가
    AllMarkers.add(locationMarker);        // 배열에 마커 추가
}
}

if (location.equals("지도 초기화")) {
    removeAllMarkers();
    // 지도상 서울특별시 중심, 지도 위 경도 기반 위치 가져오기(영동군으로
초기화 되있는 상태)
    LatLng center = new LatLng(35.965500, 127.885751);
    mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(center, (float)
6.9));
    // 해당 지역으로 카메라 이동
}
}

private void toggleFab() {

    if (isFabOpen) {

```

```

        ranking_btn.startAnimation(rotate_close_anim);
        feedback_btn.startAnimation(rotate_close_anim);
        ranking_btn.setClickable(false);
        feedback_btn.setClickable(false);
        ranking_text.setVisibility(View.INVISIBLE);
        feedback_text.setVisibility(View.INVISIBLE);
        isFabOpen = false;
    } else {
        ranking_btn.startAnimation(rotate_open_anim);
        feedback_btn.startAnimation(rotate_open_anim);
        ranking_btn.setClickable(true);
        feedback_btn.setClickable(true);
        ranking_text.setVisibility(View.VISIBLE);
        feedback_text.setVisibility(View.VISIBLE);
        isFabOpen = true;
    }
}

// 지도 상의 마커를 모두 지웁니다.
private void removeAllMarkers() {
    for (Marker locationMarker : AllMarkers) {
        locationMarker.remove();
    }
    AllMarkers.clear();
}
}

<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:focusableInTouchMode="true"
    tools:context=".MainActivity"
    android:orientation="horizontal">

    <fragment
        android:id="@+id/map"
        android:name="com.google.android.gms.maps.SupportMapFragment"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        tools:context=".MapsActivity"/>

    <SearchView
        android:id="@+id/sv_location"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="end|top"
        android:layout_margin="3dp"
        android:background="@drawable/searchview"
        android:iconifiedByDefault="false"
        android:queryHint="Search..."
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.444"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="@+id/map"
        app:layout_constraintVertical_bias="0.0" />

```

```

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="120dp"
    android:layout_gravity="bottom|start"
    android:layout_marginLeft="20dp"
    android:layout_marginBottom="30dp"
    android:orientation="vertical" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="등급"
        android:fontFamily="@font/nurir"
        android:layout_marginLeft="0.5dp"
        android:layout_marginBottom="2dp"
        android:textColor="#000000"
        android:textSize="10dp"
        android:textStyle="bold" />

    <LinearLayout
        android:layout_width="20dp"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

        <ImageButton
            android:id="@+id/src_gradation"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginStart="0dp"
            android:layout_weight="1"
            android:background="@drawable/gradation_2"
            android:contentDescription="@string/todo" />

        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:orientation="vertical">

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginLeft="4dp"
                android:layout_marginTop="2dp"
                android:text="1"
                android:textColor="#000000"
                android:textSize="8sp" />

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginLeft="4dp"
                android:layout_marginTop="2dp"
                android:text="2"
                android:textColor="#000000"
                android:textSize="8sp" />

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"

```



```

        android:layout_marginLeft="4dp"
        android:layout_marginTop="2dp"
        android:text="3"
        android:textColor="#000000"
        android:textSize="8sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="4dp"
    android:layout_marginTop="2dp"
    android:text="4"
    android:textColor="#000000"
    android:textSize="8sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="4dp"
    android:layout_marginTop="3dp"
    android:text="5"
    android:textColor="#000000"
    android:textSize="8sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="4dp"
    android:layout_marginTop="2dp"
    android:text="6"
    android:textColor="#000000"
    android:textSize="8sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="4dp"
    android:layout_marginTop="2dp"
    android:text="7"
    android:textColor="#000000"
    android:textSize="8sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="4dp"
    android:layout_marginTop="2dp"
    android:text="8"
    android:textColor="#000000"
    android:textSize="7sp" />
</LinearLayout>
</LinearLayout>
</LinearLayout>

<!-- <HorizontalScrollView-->
<!--     android:layout_width="match_parent"-->
<!--     android:layout_height="wrap_content"-->
<!--     android:layout_marginTop="55dp"-->
<!--     android:scrollbars="none"-->

```

```

<!--      >-->
<!--      -->
<!--      </HorizontalScrollView>-->

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="40dp"
    android:gravity="left"
    android:layout_marginTop="55dp"
    android:orientation="horizontal">

    <Spinner
        android:id="@+id/spinner_grade"
        android:layout_width="wrap_content"
        android:layout_height="match_parent"
        android:layout_margin="3dp"
        android:gravity="center"
        android:background="@drawable/searchview"
        android:popupBackground="@drawable/searchview"/>

    <Spinner
        android:id="@+id/spinner_location"
        android:layout_width="wrap_content"
        android:layout_height="match_parent"
        android:layout_margin="3dp"
        android:gravity="center"
        android:background="@drawable/searchview"
        android:popupBackground="@drawable/searchview"/>

    <Button
        android:id="@+id/btn_allremove"
        android:layout_width="wrap_content"
        android:layout_height="match_parent"
        android:layout_margin="3dp"
        android:text="지도 초기화"
        android:textSize="16sp"
        android:background="@drawable/searchview"/>

</LinearLayout>

<com.google.android.material.floatingactionbutton.FloatingActionButton
    android:id="@+id/add_btn"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="end|bottom"
    android:layout_marginEnd="24dp"
    android:layout_marginRight="24dp"
    android:layout_marginBottom="24dp"
    android:clickable="true"
    android:focusable="true"
    android:src="@drawable/ic_baseline_arrow_circle_up_24" />

<LinearLayout
    android:id="@+id/ranking_btnlayout"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="bottom|end"
    android:layout_marginBottom="104dp"

```

```

        android:orientation="horizontal">

        <TextView
            android:id="@+id/ranking_text"
            android:layout_width="80dp"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:gravity="center"
            android:layout_marginRight="16dp"
            android:background="@drawable/searchview"
            android:text="(도) 별 랭킹"
            android:textColor="@color/white"
            android:visibility="invisible"/>

        <com.google.android.material.floatingactionbutton.FloatingActionButton
            android:id="@+id/ranking_btn"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="end|bottom"
            app:backgroundTint="@color/white"
            android:layout_marginEnd="24dp"
            android:layout_marginRight="24dp"
            android:clickable="true"
            android:focusable="true"
            android:src="@drawable/ic_baseline_library_books_24"
            android:visibility="invisible" />

    </LinearLayout>

    <LinearLayout
        android:id="@+id/feedback_btnlayout"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom|end"
        android:layout_marginBottom="184dp"
        android:orientation="horizontal">

        <TextView
            android:id="@+id/feedback_text"
            android:layout_width="80dp"
            android:layout_height="20dp"
            android:layout_gravity="center"
            android:gravity="center"
            android:layout_marginRight="16dp"
            android:background="@drawable/searchview"
            android:text="피드백"
            android:textColor="@color/white"
            android:visibility="invisible"/>

        <com.google.android.material.floatingactionbutton.FloatingActionButton
            android:id="@+id/feedback_btn"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="end|bottom"
            android:layout_marginEnd="24dp"
            android:layout_marginRight="24dp"
            android:clickable="true"
            android:focusable="true"
            android:rotationX="1"

```

```

        app:backgroundTint="@color/white"
        android:src="@drawable/ic_baseline_textsms_24"
        android:visibility="invisible" />

    </LinearLayout>

</FrameLayout>

```

## 7.2 순위 추천

```

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.Gravity;
import android.view.View;
import android.widget.Button;
import android.widget.ImageButton;

public class RankRecommend extends AppCompatActivity {
    public void BtnConnect(Button[] CityTOPArr) {
        CityTOPArr[0] = (Button) findViewById(R.id.bt1);
        CityTOPArr[1] = (Button) findViewById(R.id.bt2);
        CityTOPArr[2] = (Button) findViewById(R.id.bt3);
        CityTOPArr[3] = (Button) findViewById(R.id.bt4);
        CityTOPArr[4] = (Button) findViewById(R.id.bt5);
        CityTOPArr[5] = (Button) findViewById(R.id.bt6);
        CityTOPArr[6] = (Button) findViewById(R.id.bt7);
        CityTOPArr[7] = (Button) findViewById(R.id.bt8);
        CityTOPArr[8] = (Button) findViewById(R.id.bt9);
        CityTOPArr[9] = (Button) findViewById(R.id.bt10);
        CityTOPArr[10] = (Button) findViewById(R.id.bt11);
        CityTOPArr[11] = (Button) findViewById(R.id.bt12);
        CityTOPArr[12] = (Button) findViewById(R.id.bt13);
    }
    public void CityBtnConnect(final Button[] CityTOPArr, String[] cities)
    {
        for(int i = 0;i<13;i++)
        {
            CityTOPArr[i].setTextSize(15);
            CityTOPArr[i].setGravity(50);
            CityTOPArr[i].setText(cities[i]);
            CityTOPArr[i].setGravity(Gravity.CENTER);

        }
    }

    public void onClick(View v) {
        Intent intent = new Intent(getApplicationContext(),
RankRecommendSub.class);
        intent.putExtra("province", "서울특별시");
        if (v.getId() == R.id.bt1) {
            intent.putExtra("province", "서울특별시");

```

```

    }
    else if (v.getId() == R.id.bt2) {
        intent.putExtra("province", "대전광역시");
    }
    else if (v.getId() == R.id.bt3) {
        intent.putExtra("province", "광주광역시");
    }
    else if (v.getId() == R.id.bt4) {
        intent.putExtra("province", "부산광역시");
    }
    else if (v.getId() == R.id.bt5) {
        intent.putExtra("province", "인천광역시");
    }
    else if (v.getId() == R.id.bt6) {
        intent.putExtra("province", "전라북도");
    }
    else if (v.getId() == R.id.bt7) {
        intent.putExtra("province", "전라남도");
    }
    else if (v.getId() == R.id.bt8) {
        intent.putExtra("province", "충청북도");
    }
    else if (v.getId() == R.id.bt9) {
        intent.putExtra("province", "충청남도");
    }
    else if (v.getId() == R.id.bt10) {
        intent.putExtra("province", "강원도");
    }
    else if (v.getId() == R.id.bt11) {
        intent.putExtra("province", "경상남도");
    }
    else if (v.getId() == R.id.bt12) {
        intent.putExtra("province", "경상북도");
    }
    else if (v.getId() == R.id.bt13) {
        intent.putExtra("province", "경기도");
    }
    startActivity(intent);
}

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_rank_recommend);
    Button[] CityTOPArr = new Button[13];
    BtnConnect(CityTOPArr);
    String[] cities = {"서울특별시", "대전광역시", "광주광역시", "부산광역시",
        "인천광역시", "전라북도", "전라남도", "충청북도",
        "충청남도", "강원도", "경상남도", "경상북도", "경기도"};
    CityBtnConnect(CityTOPArr, cities);

    ImageButton back = findViewById(R.id.back2);
    back.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            finish();
        }
    });
}
}

```

```

import androidx.annotation.NonNull;
import androidx.annotation.RequiresApi;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.content.ContextCompat;

import android.content.Intent;
import android.content.res.Resources;
import android.database.Cursor;
import android.graphics.Color;
import android.graphics.Typeface;
import android.graphics.drawable.GradientDrawable;

import android.graphics.fonts.Font;
import android.os.Build;
import android.os.Bundle;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.ImageButton;
import android.widget.LinearLayout;
import android.widget.TextView;

import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.HashMap;

public class RankRecommendSub extends AppCompatActivity {
    City[] cityArr = new City[154];
    citiesDatabase cityDatabase;

    private void QuickKSort(City[] arr, int left, int right) {
        if (left < right) {
            int q = Partition(arr, left, right);
            QuickKSort(arr, left, q - 1);
            QuickKSort(arr, q + 1, right);
        }
    }

    private int Partition(City[] arr, int left, int right) {
        int low = left;
        int high = right + 1;
        City pivot = arr[left];
        do {
            do {
                low++;
            } while (low <= right && pivot.grade > arr[low].grade);
            do {
                high--;
            } while (high >= left && pivot.grade < arr[high].grade);
            if (low < high) {
                City t = arr[low];

```

```

        arr[low] = arr[high];
        arr[high] = t;

    }
    while (low < high);
    arr[left] = arr[high];
    arr[high] = pivot;
    return high;
}

@RequiresApi(api = Build.VERSION_CODES.O)
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_ranking_recommend_sub);
    cityDatabase = new citiesDatabase(getApplicationContext());
    cityDatabase.open();

    //인텐트에서 도 광역시 정보를 받음
    final Intent intent = getIntent();
    String province = intent.getExtras().getString("province");
    TextView t = findViewById(R.id.tbx);
    t.setText(province);

    Cursor values = cityDatabase.selectColumns();
    for (int i = 0; i < values.getCount(); i++) {
        values.moveToPosition(i);
        cityArr[i] = new City(values.getString(1), values.getString(2),
values.getString(3));
    }

    for (int i = 0; i < values.getCount(); i++) {
        System.out.println(cityArr[i].province + " " + cityArr[i].name + " "
+ cityArr[i].grade);
    }

    QuickKSort(cityArr, 0, cityArr.length - 1);
    final GradientDrawable drawable = (GradientDrawable)
ContextCompat.getDrawable(this, R.drawable.maingrad);
    LinearLayout l = findViewById(R.id.list);
    int rank = 1;

    //도 광역시 이름이 province 인 지역들만 가져와서 버튼으로 동적생성
    for (int i = 0; i < 154; i++) {
        if (cityArr[i].province.contentEquals(t.getText())) {
            LinearLayout.LayoutParams r = new
LinearLayout.LayoutParams(ViewGroup.LayoutParams.MATCH_PARENT,
ViewGroup.LayoutParams.WRAP_CONTENT);
            r.setMargins(0, 10, 0, 20);
            final Button b = new Button(this);
            String text = rank + "위 " + t.getText() + " " +
cityArr[i].name;
            b.setText(text);
            b.setBackground(drawable);
            b.setTextColor(Color.BLACK);

```

```

        b.setHeight(160);
        b.setTextSize(18);
        b.setLayoutParams(r);
        Typeface font = getResources().getFont(R.font.font);
        b.setTypeface(font);
        l.addView(b);
        rank++;
        b.setOnClickListener(v -> {
            Intent graphIntent = new
Intent (RankRecommendSub.this,GraphActivity.class);
            String buttonText = b.getText().toString();
            String[] strArr = buttonText.split(" ");
            String reComposition = strArr[1] + " " + strArr[2];
            graphIntent.putExtra("region",reComposition);
            graphIntent.putExtra("response","0");
            startActivity(graphIntent);
        });
        b.setBackground(drawable);
    }
}

//뒤로가기 버튼 함수구현
ImageButton back = (ImageButton) findViewById(R.id.back);
back.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        finish();
    }
});
}
}

```

## XML

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".RankRecommend"
    android:orientation="vertical"
    android:background="#ffffff">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:background="#FFFFFF">
        <ImageButton
            android:layout_width="40dp"
            android:layout_height="40dp"
            android:background="@drawable/back"
            android:textColor="#252525"
            android:textSize="20dp"
            android:layout_margin="10dp"
            android:textStyle="bold"
            android:id="@+id/back2"
        ></ImageButton>
    </LinearLayout>

```



```

<TextView
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center_vertical"
    android:text="지역 선택"
    android:textSize="30dp"
    android:layout_marginLeft="80dp"
    android:padding="10dp"
    android:textColor="#034305"
    android:fontFamily="@font/font"
></TextView>
</LinearLayout>
<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:background="#000000"
    android:layout_marginLeft="10dp"
    android:layout_marginRight="10dp"></View>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:textColor="#FFFFFF"
    android:background="#FFFFFF"
    android:padding="10dp">
    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
    >
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
    >
    <Button android:padding="20dp"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:orientation="horizontal"
        android:background="@drawable/focusd"
        android:gravity="left"
        android:onClick="onClick"
        android:textSize="15sp"
        android:textColor="#000000"
        android:fontFamily="@font/font"
        android:id="@+id/bt1"
        android:layout_marginBottom="10dp">
    </Button>
    <Button android:padding="20dp"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:orientation="horizontal"
        android:onClick="onClick"
        android:background="@drawable/focusd"
        android:gravity="left"
        android:textSize="15sp"
        android:textColor="#000000"
        android:fontFamily="@font/font"
        android:id="@+id/bt2"

```

```

        android:layout_marginBottom="10dp"
    >
</Button>
<Button android:padding="20dp"
    android:layout_width="match_parent"
    android:layout_height="60dp"
    android:onClick="onClick"
    android:orientation="horizontal"
    android:background="@drawable/focusd"
    android:gravity="left"
    android:textSize="15sp"
    android:textColor="#000000"
    android:fontFamily="@font/font"
    android:id="@+id/bt3"
    android:layout_marginBottom="10dp"
    >
</Button>
<Button android:padding="20dp"
    android:layout_width="match_parent"
    android:layout_height="60dp"
    android:orientation="horizontal"
    android:onClick="onClick"
    android:background="@drawable/focusd"
    android:gravity="left"
    android:textSize="15sp"
    android:textColor="#000000"
    android:fontFamily="@font/font"
    android:id="@+id/bt4"
    android:layout_marginBottom="10dp"
    >
</Button>
<Button android:padding="20dp"
    android:layout_width="match_parent"
    android:layout_height="60dp"
    android:onClick="onClick"
    android:orientation="horizontal"
    android:background="@drawable/focusd"
    android:gravity="left"
    android:textSize="15sp"
    android:textColor="#000000"
    android:fontFamily="@font/font"
    android:id="@+id/bt5"
    android:layout_marginBottom="10dp"
    >
</Button>
<Button android:padding="20dp"
    android:layout_width="match_parent"
    android:layout_height="60dp"
    android:orientation="horizontal"
    android:onClick="onClick"
    android:background="@drawable/focusd"
    android:gravity="left"
    android:textSize="15sp"
    android:textColor="#000000"
    android:fontFamily="@font/font"
    android:id="@+id/bt6"
    android:layout_marginBottom="10dp"
    >
</Button>

```

```

<Button android:padding="20dp"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:onClick="onClick"
        android:orientation="horizontal"
        android:background="@drawable/focusd"
        android:gravity="left"
        android:textSize="15sp"
        android:textColor="#000000"
        android:fontFamily="@font/font"
        android:id="@+id/bt7"
        android:layout_marginBottom="10dp"
    >
</Button>
<Button android:padding="20dp"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:onClick="onClick"
        android:orientation="horizontal"
        android:background="@drawable/focusd"
        android:gravity="left"
        android:textSize="15sp"
        android:textColor="#000000"
        android:fontFamily="@font/font"
        android:id="@+id/bt8"
        android:layout_marginBottom="10dp"
    >
</Button>
<Button android:padding="20dp"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:onClick="onClick"
        android:orientation="horizontal"
        android:background="@drawable/focusd"
        android:gravity="left"
        android:textSize="15sp"
        android:textColor="#000000"
        android:fontFamily="@font/font"
        android:id="@+id/bt9"
        android:layout_marginBottom="10dp"
    >
</Button>
<Button android:padding="20dp"
        android:layout_width="match_parent"
        android:onClick="onClick"
        android:layout_height="60dp"
        android:orientation="horizontal"
        android:background="@drawable/focusd"
        android:gravity="left"
        android:textSize="15sp"
        android:textColor="#000000"
        android:fontFamily="@font/font"
        android:id="@+id/bt10"
        android:layout_marginBottom="10dp"
    >
</Button>
<Button android:padding="20dp"
        android:layout_width="match_parent"
        android:layout_height="60dp"

```

```

        android:orientation="horizontal"
        android:background="@drawable/focusd"
        android:onClick="onClick"
        android:gravity="left"
        android:textSize="15sp"
        android:textColor="#000000"
        android:fontFamily="@font/font"
        android:id="@+id/bt11"
        android:layout_marginBottom="10dp"
    >
</Button>
<Button android:padding="20dp"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:orientation="horizontal"
        android:onClick="onClick"
        android:background="@drawable/focusd"
        android:gravity="left"
        android:textSize="15sp"
        android:textColor="#000000"
        android:fontFamily="@font/font"
        android:id="@+id/bt12"
        android:layout_marginBottom="10dp"
    >
</Button>
<Button android:padding="20dp"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:orientation="horizontal"
        android:onClick="onClick"
        android:background="@drawable/focusd"
        android:gravity="left"
        android:textSize="15sp"
        android:textColor="#000000"
        android:fontFamily="@font/font"
        android:id="@+id/bt13"
        android:layout_marginBottom="10dp"
    >
</Button>
</LinearLayout>
</ScrollView>
</LinearLayout>
</LinearLayout>

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:tools="http://schemas.android.com/tools"
    android:orientation="vertical"
    tools:context=".RankRecommendSub"
    android:background="#FFFFFF">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:background="#FFFFFF">
        <ImageButton
            android:id="@+id/back"
            android:layout_width="40dp"

```

```

        android:layout_height="40dp"
        android:background="@drawable/back"
        android:textColor="#252525"
        android:textSize="20dp"
        android:layout_margin="10dp"
        android:textStyle="bold"
    ></ImageButton>

    <TextView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center_vertical"
        android:text="지역 선택"
        android:textSize="30dp"
        android:layout_marginLeft="80dp"
        android:padding="10dp"
        android:textColor="#034305"
        android:fontFamily="@font/font"
        android:id="@+id/tbx"
    ></TextView>
</LinearLayout>
<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:background="#000000"
    android:layout_marginLeft="10dp"
    android:layout_marginRight="10dp"></View>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:textColor="#FFFFFF"
    android:background="#FFFFFF"
    android:padding="10dp">
    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
    >
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:orientation="vertical"
            android:id="@+id/list"/>
    </ScrollView>
</LinearLayout>
</LinearLayout>

```

## 7.3 그래프

```

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentManager;
import androidx.fragment.app.FragmentPagerAdapter;

```

```

import androidx.viewpager.widget.ViewPager;

import android.content.Context;
import android.content.Intent;
import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.ImageButton;
import android.widget.TextView;

import com.google.android.material.tabs.TabLayout;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.nio.file.FileSystemLoopException;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.HashMap;
import java.util.List;

public class GraphActivity extends AppCompatActivity {

    TabLayout tabLayout;
    ViewPager viewPager;
    TextView province;
    ImageButton restart;
    ImageButton back;
    String getValueFromOtherClass;
    String response;
    GraphDatabase graphDatabase;
    ArrayList<ArrayList<String>> arrayLists;
    ArrayList<String> arrayList;
    ArrayList<String> rising_point_of_landslide_comparingTo_last_year;
    ArrayList<String> number_of_generator_for_each_years;
    ArrayList<String> generating_amount;

    Thread loading = new Thread() {
        @Override
        public void run() {
            try {
                Intent loading = new Intent(GraphActivity.this, Loading.class);
                loading.putExtra("activity", "graph");
                startActivity(loading);
            } catch (Exception ignored) {
            }
        }
    };

    generatingDatabase generatingDatas;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_graph);
        //데이터 베이스 열고 생성
        generatingDatas = new generatingDatabase(getApplicationContext());
    }
}

```

```

generatingDatas.open();
generatingDatas.create();

//데이터를 fragment로 보낼 ArrayList 생성
rising_point_of_landslide_comparingTo_last_year = new ArrayList<>();
number_of_generator_for_each_years = new ArrayList<>();
generating_amount = new ArrayList<>();

//메인화면에서 Intent를 통해서 지역 값 받아오기
Intent intent1 = getIntent();
getValueFromOtherClass = intent1.getStringExtra("region");

//선택된 지역의 시군구 전력 데이터 찾기
String generatingValues;
String[] splitGeneratingValues;

Cursor generatingDatasCursor =
generatingDatas.selectWhatIWant(getValueFromOtherClass);
if(generatingDatasCursor.getCount() > 0){
    generatingDatasCursor.moveToPosition(0);
    generatingValues = generatingDatasCursor.getString(2);
    splitGeneratingValues = generatingValues.split(", ");
    generating_amount.addAll(Arrays.asList(splitGeneratingValues));
}

//컴포넌트들 액티비티에서 받기
tabLayout = findViewById(R.id.tab_layout);
viewPager = findViewById(R.id.view_pager);
province = findViewById(R.id.province);
restart = findViewById(R.id.restart);
back = findViewById(R.id.back);

//큰타이틀 설정
province.setText(getValueFromOtherClass);

//그래프 데이터베이스 열기
graphDatabase = new GraphDatabase(getApplicationContext());
graphDatabase.open();

//그래프에 관한 데이터 베이스 열기
Cursor GraphCursor =
graphDatabase.selectedWhatIWant(getValueFromOtherClass);
//그래프에 관한 데이터 베이스를 찾고 위에서 생성한 ArrayList에 저장
if (GraphCursor.getCount() != 0) {
    for (int i = 0; i < GraphCursor.getCount(); i++) {
        GraphCursor.moveToPosition(i);
        number_of_generator_for_each_years.add(GraphCursor.getString(3));
    }
    rising_point_of_landslide_comparingTo_last_year.add(GraphCursor.getString(5));
}

//슬라이드 탭의 이름을 설정하기 위한 ArrayList생성후 탭 이름 넣기

```

```

        arrayList = new ArrayList<>();
        arrayList.add("발진소 수량");
        arrayList.add("산사태 증감");
        arrayList.add("발진량");

        System.out.println("잘지냈어요 여기봐요 = ");

        //ArrayList에 ArrayList<String>들 넣기
        arrayLists = new ArrayList<>();
        arrayLists.add(number_of_generator_for_each_years);
        arrayLists.add(rising_point_of_landslide_comparingTo_last_year);
        arrayLists.add(generating_amount);

        //fragment들로 데이터 넘겨주기
        prepareViewPager(viewPager, arrayList, arrayLists);

        //fragment 완전 set
        tabLayout.setupWithViewPager(viewPager);

        //뒤로가기 버튼 리스너
        back.setOnClickListener(view -> {
            if (loading.getState() != Thread.State.NEW) {
                loading.interrupt();
            }
            loading.start();
            finish();
        });

        //다시시작 리스너
        restart.setOnClickListener(view -> {
            if (loading.getState() != Thread.State.NEW) {
                loading.interrupt();
            }
            loading.start();
            Intent intent = new Intent(GraphActivity.this, GraphActivity.class);
            intent.putExtra("region", getValueFromOtherClass);
            startActivity(intent);
            finish();
        });

        //전화면으로 데이터 response 0를 받았을 경우 재시작
        prepareViewPager(viewPager, arrayList, arrayLists);

    }

    // 다음 fragment로 데이터 넘겨주기
    private void prepareViewPager(ViewPager viewPager, ArrayList<String>
arrayList, ArrayList<ArrayList<String>> arrayLists) {
        MainAdapter adapter = new MainAdapter(getSupportFragmentManager());
        MainFragment fragment = new MainFragment();

```



```

String[] names_of_charts = new String[3];
names_of_charts[0] = "number_of_generator_for_each_years";
names_of_charts[1] = "rising_point_of_landslide_comparingTo_last_year";
names_of_charts[2] = "generating_amount";

for (int i = 0; i < arrayList.size(); i++) {
    Bundle bundle = new Bundle();
    bundle.putStringArrayList(names_of_charts[i], arrayLists.get(i));
    fragment.setArguments(bundle);

    adapter.addFragment(fragment, arrayList.get(i));

    fragment = new MainFragment();
}
viewPager.setAdapter(adapter);

}

//fragment를 위한 어댑터
private static class MainAdapter extends FragmentPagerAdapter {

    ArrayList<String> arrayList = new ArrayList<>();

    List<Fragment> fragmentList = new ArrayList<>();

    public void addFragment(Fragment fragment, String title) {
        arrayList.add(title);
        fragmentList.add(fragment);
    }

    public MainAdapter(@NonNull FragmentManager fm) {
        super(fm);
    }

    @NonNull
    @Override
    public Fragment getItem(int position) {
        return fragmentList.get(position);
    }

    @Override
    public int getCount() {
        return fragmentList.size();
    }

    @Nullable
    @Override
    public CharSequence getPageTitle(int position) {
        return arrayList.get(position);
    }
}

```

```

}

import android.graphics.Color;
import android.os.Bundle;

import androidx.annotation.NonNull;
import androidx.fragment.app.Fragment;

import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.LinearLayout;
import android.widget.TextView;

import com.github.mikephil.charting.charts.BarChart;
import com.github.mikephil.charting.charts.LineChart;
import com.github.mikephil.charting.components.AxisBase;
import com.github.mikephil.charting.components.XAxis;
import com.github.mikephil.charting.components.YAxis;
import com.github.mikephil.charting.data.BarData;
import com.github.mikephil.charting.data.BarDataSet;
import com.github.mikephil.charting.data.BarEntry;
import com.github.mikephil.charting.data.Entry;
import com.github.mikephil.charting.data.LineData;
import com.github.mikephil.charting.data.LineDataSet;
import com.github.mikephil.charting.formatter.ValueFormatter;

import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Calendar;
import java.util.Date;
import java.util.List;
import java.util.Locale;

import lecho.lib.hellocharts.gesture.ContainerScrollType;
import lecho.lib.hellocharts.model.Axis;
import lecho.lib.hellocharts.model.AxisValue;
import lecho.lib.hellocharts.model.Column;
import lecho.lib.hellocharts.model.ColumnChartData;
import lecho.lib.hellocharts.model.SubcolumnValue;
import lecho.lib.hellocharts.model.Viewport;
import lecho.lib.hellocharts.view.ColumnChartView;

/**
 * A simple {@link Fragment} subclass.
 * Use the {@link MainFragment#newInstance} factory method to
 * create an instance of this fragment.
 */
public class MainFragment extends Fragment {

    // TODO: Rename parameter arguments, choose names that match
    // the fragment initialization parameters, e.g. ARG_ITEM_NUMBER
    private static final String ARG_PARAM1 = "param1";
    private static final String ARG_PARAM2 = "param2";
    // TODO: Rename and change types of parameters
    String mParam1;
    String mParam2;

    public MainFragment() {

```

```

    // Required empty public constructor
}

// TODO: Rename and change types and number of parameters
public static MainFragment newInstance(String param1, String param2) {
    MainFragment fragment = new MainFragment();
    Bundle args = new Bundle();
    args.putString(ARG_PARAM1, param1);
    args.putString(ARG_PARAM2, param2);
    fragment.setArguments(args);
    return fragment;
}

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (getArguments() != null) {
        mParam1 = getArguments().getString(ARG_PARAM1);
        mParam2 = getArguments().getString(ARG_PARAM2);
    }
}

@Override
public View onCreateView(@NonNull LayoutInflater inflater, ViewGroup
container,
                        Bundle savedInstanceState) {
    View view = inflater.inflate(R.layout.fragment_main, container, false);

    BarChart barchart = view.findViewById(R.id.barchart);
    LineChart linechart = view.findViewById(R.id.linechart);
    TextView titles = view.findViewById(R.id.title);
    //try {
    Bundle bundle = getArguments();

    System.out.println("fragment 실행");
    //number_of_generator_for_each_years 데이터 Main으로 부터 받기 시도

    ArrayList<String> number_of_generator_for_each_years;

    try {
        assert bundle != null;
        number_of_generator_for_each_years =
bundle.getStringArrayList("number_of_generator_for_each_years");
        for (String n : number_of_generator_for_each_years) {
            System.out.println("generators = " + n);
        }
        for (String n : number_of_generator_for_each_years) {
            System.out.println(n);
        }
    }

    ArrayList<String> year = new ArrayList<>();
    ArrayList<BarEntry> entries = new ArrayList<>();

```

```

        int count = 0;
        int maxValue = 0;
        for (int i = 2015; i < 2015 +
number_of_generator_for_each_years.size(); i++) {
            //수정코드
            entries.add(new
BarEntry(Float.parseFloat(number_of_generator_for_each_years.get(count)),
count));

            year.add(String.valueOf(i));

            //전코드
            if
(Integer.parseInt(number_of_generator_for_each_years.get(count)) > maxValue) {
                maxValue =
Integer.parseInt(number_of_generator_for_each_years.get(count));
            }
            count++;
        }
        XAxis xAxis = barchart.getXAxis();
        xAxis.setPosition(XAxis.XAxisPosition.BOTTOM);

        System.out.println("개수 size = " + entries.size());
        BarDataSet barDataSet = new BarDataSet(entries, "태양광 발전소 개수");
        barchart.animateY(3000);
        BarData BData = new BarData(year, barDataSet);
        barchart.setData(BData);
        barchart.setDescription(null);
        titles.setText("연별 태양광 발전소 등록수");
    } catch (Exception ignored) {
    }

    //rising_point_of_landslide_comparingTo_last_year 데이터 Main으로 부터 받기
시도

    ArrayList<String> rising_point_of_landslide_comparingTo_last_year;
    try {
        rising_point_of_landslide_comparingTo_last_year =
bundle.getStringArrayList("rising_point_of_landslide_comparingTo_last_year");

        for (String n : rising_point_of_landslide_comparingTo_last_year) {
            System.out.println("rising " + n);
        }

        ArrayList<String> year = new ArrayList<>();
        ArrayList<BarEntry> entries = new ArrayList<>();

        int count1 = 0;
        for (int i = 2015; i < 2015 +
rising_point_of_landslide_comparingTo_last_year.size(); i++) {
            entries.add(new
BarEntry(Float.parseFloat(rising_point_of_landslide_comparingTo_last_year.get(co
unt1)), count1));
            year.add(String.valueOf(i));
            count1++;
        }
        XAxis xAxis = barchart.getXAxis();
        xAxis.setPosition(XAxis.XAxisPosition.BOTTOM);

```

```

        System.out.println("개수 size = " + entries.size());
        BarDataSet barDataSet = new BarDataSet(entries, "%");
        barchart.animateY(3000);
        BarData BData = new BarData(year, barDataSet);
        barchart.setData(BData);
        barchart.setDescription(null);
        String title = "연별 산사태 증감률";
        titles.setText(title);

    } catch (Exception ignored) {
    }

    try {
        ArrayList<String> generating_amount;
        generating_amount = bundle.getStringArrayList("generating_amount");

        String response = "";
        for (String n : generating_amount) {
            System.out.print(n + ",");
            response += n;
            //response.append(n);
        }
        String res = "";
        System.out.println("response" + response);
        if (response.equals("ul") || generating_amount.size() < 1 ||
response.equals("")) {
            titles.setText("데이터 없음");
            res = "데이터 없음";
        }

        barchart.setVisibility(BarChart.INVISIBLE);
        linechart.setVisibility(LineChart.VISIBLE);

        SimpleDateFormat dateFormat = new SimpleDateFormat("yyyy-MM",
Locale.getDefault());
        Calendar cal = Calendar.getInstance();
        Date stringToDate = dateFormat.parse("2015-01");
        float max = 0;

        ArrayList<String> year = new ArrayList<>();
        ArrayList<Entry> entries = new ArrayList<>();
        int count = 0;

        assert stringToDate != null;
        cal.setTime(stringToDate);
        boolean flag = false;
        for (int i = 0; i < generating_amount.size(); i++) {
            try {
                if (!generating_amount.get(i).equals("0") &&
!generating_amount.get(i + 1).equals("0")) {

```

```

        flag = true;
    } else if (generating_amount.get(i).equals("0") &&
generating_amount.get(i + 1).equals("0")) {
        flag = false;
    }
    } catch (Exception e) {
        flag = false;
    }

    if (flag) {
        entries.add(new
BarEntry(Float.parseFloat(generating_amount.get(i)), count));
        year.add(dateFormat.format(cal.getTime()));
        count++;
        if (Float.parseFloat(generating_amount.get(i)) > max) {
            max = Float.parseFloat(generating_amount.get(i));
        }
    }

    cal.add(Calendar.MONTH, 1);
}

System.out.println("개수 size = " + entries.size());
LineDataSet barDataSet = new LineDataSet(entries, "MW");
linechart.animateY(3000);
LineData LData = new LineData(year, barDataSet);
linechart.setData(LData);
linechart.setDescription(null);
XAxis xAxis = linechart.getXAxis();
xAxis.setPosition(XAxis.XAxisPosition.BOTTOM);

if (!res.equals("데이터 없음")) {
    String title = "월별 태양광 발전량";
    titles.setText(title);
}

} catch (Exception ignored) {
}

return view;

}

}

```

## XML

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"

```

```

android:background="#ffffff"
android:orientation="vertical"
tools:context="com.App.SolarPing.GraphActivity">

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#ffffff"
    android:layout_weight="5.9"
    android:orientation="vertical">

    <LinearLayout
        android:layout_margin="10dp"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="horizontal">

        <ImageButton
            android:id="@+id/back"
            android:layout_width="match_parent"
            android:layout_weight="1"
            android:layout_marginStart="10dp"
            android:layout_height="match_parent"
            android:background="@drawable/back"
            android:contentDescription="@string/todo" />

        <TextView
            android:id="@+id/province"
            android:layout_width="match_parent"
            android:textSize="20sp"
            android:textStyle="bold"
            android:fontFamily="@font/font"
            android:textColor="@color/black"
            android:gravity="center"
            android:layout_height="match_parent"
            android:layout_weight="0.3"/>

        <ImageButton
            android:id="@+id/restart"
            android:background="@drawable/restart"
            android:layout_width="match_parent"
            android:layout_marginEnd="10dp"
            android:layout_weight="1"
            android:layout_height="match_parent" />

    </LinearLayout>

    <!--      <TextView-->
    <!--          android:id="@+id/empty"-->
    <!--          android:layout_width="wrap_content"-->
    <!--          android:layout_weight="0.5"-->
    <!--          android:layout_marginTop="40dp"-->
    <!--          android:layout_height="match_parent"-->
    <!--          android:layout_gravity="center"-->
    <!--          android:textSize="40sp" />-->

</LinearLayout>
<View
    android:layout_width="match_parent"
    android:layout_height="2dp"
    android:layout_marginStart="10dp"
    android:layout_marginEnd="10dp"
    android:background="#3a3a3a"/>

```

```

<androidx.viewpager.widget.ViewPager
    android:id="@+id/view_pager"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_weight="1" />

<com.google.android.material.tabs.TabLayout
    android:id="@+id/tab_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_weight="6.1"
    android:background="@drawable/tabtheme"
    android:fontFamily="@font/font"
    app:tabIndicatorColor="#ccffcc"
    app:tabIndicatorGravity="top"
    app:tabSelectedTextColor="#6FED6F"
    app:tabTextColor="@android:color/black" />

</LinearLayout>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    tools:context="com.App.SolarPing.GraphActivity"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#ffffff"
    android:orientation="vertical">

    <!-- TODO: Update blank fragment layout -->

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_gravity="center"
        android:layout_marginTop="30dp"
        android:paddingTop="10dp"
        android:background="@drawable/titleradiusandcolor"
        android:gravity="center"
        android:layout_marginStart="10dp"
        android:layout_marginEnd="10dp"
        android:paddingBottom="10dp"
        android:layout_height="wrap_content">
        <TextView
            android:id="@+id/title"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:text="데이터 없음"
            android:textStyle="bold"
            android:textColor="@color/black"
            android:fontFamily="@font/font"
            android:textSize="25sp"/>
    </LinearLayout>
    <FrameLayout
        android:id="@+id/graphPlace"
        android:layout_marginTop="60dp"
        android:layout_marginBottom="20dp"
        android:layout_marginStart="20dp"
        android:layout_marginEnd="20dp"

```



```

        android:layout_width="match_parent"
        android:layout_height="340dp">
        <com.github.mikephil.charting.charts.BarChart
            android:id="@+id/barchart"
            android:layout_width="match_parent"
            android:visibility="visible"
            android:layout_height="match_parent"/>
        <com.github.mikephil.charting.charts.LineChart
            android:id="@+id/linechart"
            android:layout_width="match_parent"
            android:visibility="invisible"
            android:layout_height="match_parent"/>

```

## 7.4 피드백

```

import androidx.appcompat.app.AppCompatActivity;
import android.content.pm.ActivityInfo;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageButton;
import android.widget.RatingBar;
import android.widget.Toast;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.IgnoreExtraProperties;
import java.text.SimpleDateFormat;
import java.util.Date;

public class SecondScreen extends AppCompatActivity {

    Button star_button;
    RatingBar ratingStars;
    static float myRating = 0;

    private static final String TAG = "SecondScreen";
    private DatabaseReference database;
    EditText titleEdit;
    EditText contentsEdit;
    long now = System.currentTimeMillis();
    Date date = new Date(now);
    SimpleDateFormat sdfNow = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
    String formatDate = sdfNow.format(date);
    ImageButton back;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_second_screen);
        setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);
        database = FirebaseDatabase.getInstance().getReference();
        findViewById(R.id.check).setOnClickListener(new OnClickListener() {

```

```

//뒤로가기버튼
back = findViewById(R.id.back);

//게시판 사용변수
titleEdit = (EditText) findViewById(R.id.titleEditText);
contentsEdit = (EditText) findViewById(R.id.contentsEditText);

//별점 사용변수
star_button = findViewById(R.id.star_check);
ratingStars = findViewById(R.id.ratingBar);
ratingStars.setOnRatingBarChangeListener(new
RatingBar.OnRatingBarChangeListener() {

    @Override
    public void onRatingChanged(RatingBar ratingBar, float v, boolean b)
{

    int rating = (int) v;
    String message = null;

    myRating = (int)ratingBar.getRating();

    switch(rating) {
        case 1:
            message = "더 발전하겠습니다!";
            break;
        case 2:
            message = "분발하겠습니다!";
            break;
        case 3:
            message = "중간은 갔네요!";
            break;
        case 4:
            message = "더더 열심히 하겠습니다!";
            break;
        case 5:
            message = "감사합니다!";
            break;
    }
    Toast.makeText(SecondScreen.this, message,
Toast.LENGTH_SHORT).show();
    });

    star_button.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Toast.makeText(SecondScreen.this, String.valueOf(myRating),
Toast.LENGTH_SHORT).show();
            star_uploader();
        }
    });

    back.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            finish();

```

```

    }
    });
}

View.OnClickListener onClickListener = new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if (v.getId() == R.id.check) {
            uploader();
        }
    }
};

private void uploader(){
    String title = titleEdit.getText().toString();
    String contents = contentsEdit.getText().toString();
    FeedBack feedBack = new FeedBack(title, contents, formatDate);
    database.child("feedback").child(title).setValue(feedBack);

    Toast.makeText(getApplicationContext(), "전송완료", Toast.LENGTH_LONG).show();
}

private void star_uploader(){
    float rs_star = myRating;
    StarCount starcount = new StarCount(rs_star);
    database.child("StarNum").child(formatDate).setValue(starcount);
    Toast.makeText(getApplicationContext(), "전송완료",
    Toast.LENGTH_LONG).show();
}

}

@IgnoreExtraProperties
class StarCount{
    public float starEdit;
    public StarCount(float starEdit){
        this.starEdit = starEdit;
    }
}

@IgnoreExtraProperties
class FeedBack{
    public String title;
    public String contents;
    public String formateDate;
    public FeedBack(String title, String contents, String formateDate){
        this.title = title;
        this.contents = contents;
        this.formateDate = formateDate;
    }
}

```

## XML

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"

```

```

android:background="#ffffff"
android:orientation="vertical"
tools:context="com.App.SolarPing.SecondScreen">

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="70dp"
    android:background="#ffffff"
    android:orientation="vertical">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_margin="10dp"
        android:orientation="horizontal">

        <ImageButton
            android:id="@+id/back"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="10dp"
            android:layout_weight="1"
            android:background="@drawable/back"
            android:contentDescription="@string/todo" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_weight="0.3"
            android:fontFamily="@font/font"
            android:gravity="center"
            android:text="평가 페이지"
            android:textColor="@color/black"
            android:textSize="20sp"
            android:textStyle="bold" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_marginStart="10dp"
            android:layout_weight="1"
            android:contentDescription="@string/todo" />

    </LinearLayout>
</LinearLayout>

<androidx.appcompat.widget.Toolbar
    android:layout_width="match_parent"
    android:layout_height="2dp"
    android:layout_gravity="center_horizontal"
    android:layout_marginStart="15dp"
    android:layout_marginEnd="15dp"
    android:background="?attr/colorPrimary"
    android:minHeight="?attr/actionBarSize"
    android:theme="?attr/actionBarTheme"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"

```

```

        app:layout_constraintHorizontal_bias="0.454"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.119" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center_horizontal"
    android:layout_marginTop="10dp"
    android:fontFamily="@font/nurir"
    android:text="별점을 매겨주세요!"
    android:textColor="@color/black" />

<RatingBar
    android:id="@+id/ratingBar"
    style="@style/Widget.AppCompat.RatingBar"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center_horizontal"
    android:layout_marginTop="20dp"
    android:numStars="5"
    android:progressBackgroundTint="#D1F489"
    android:progressTint="#4CAF50"
    android:rating="0"
    android:visibility="visible" />

<Button
    android:id="@+id/star_check"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center_horizontal"
    android:layout_marginTop="20dp"
    android:backgroundTint="#8BC34A"
    android:fontFamily="@font/nurir"
    android:text="별점!"
    android:textColor="@color/black" />

<androidx.appcompat.widget.Toolbar
    android:id="@+id/toolbar8"
    android:layout_width="match_parent"
    android:layout_height="2dp"
    android:layout_gravity="center_horizontal"
    android:layout_marginStart="15dp"
    android:layout_marginEnd="15dp"
    android:background="?attr/colorPrimary"
    android:minHeight="?attr/actionBarSize"
    android:theme="?attr/actionBarTheme"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.451"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.318" />

<TextView
    android:id="@+id/textView5"
    android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:layout_marginLeft="10dp"
        android:layout_marginTop="10dp"
        android:layout_marginBottom="10dp"
        android:fontFamily="@font/nurir"
        android:text="실시간 FeedBack"
        android:textAllCaps="false"
        android:textColor="@color/black"
        android:textSize="18sp"
        android:textStyle="bold"
        android:typeface="normal" />

```

#### <LinearLayout

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_weight="0.5"
        android:orientation="vertical">

```

#### <TextView

```

        android:id="@+id/textView7"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="10dp"
        android:layout_marginBottom="10dp"
        android:fontFamily="@font/nurir"
        android:text="제목"
        android:textColor="@color/black" />

```

#### <EditText

```

        android:id="@+id/titleEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:backgroundTint="#8BC34A"
        android:ems="10"
        android:inputType="textPersonName"
        android:textColor="@color/black" />

```

#### <TextView

```

        android:id="@+id/textView8"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="10dp"
        android:layout_marginBottom="10dp"
        android:fontFamily="@font/nurir"
        android:text="내용"
        android:textColor="@color/black" />

```

#### <EditText

```

        android:id="@+id/contentsEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:backgroundTint="#8BC34A"
        android:ems="10"
        android:gravity="top"
        android:inputType="textPersonName"
        android:textColor="@color/black" />

```

```

</LinearLayout>

<Button
    android:id="@+id/check"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center_horizontal"
    android:backgroundTint="#8BC34A"
    android:fontFamily="@font/nurir"
    android:text="확인"
    android:textColor="@color/black" />

<androidx.appcompat.widget.Toolbar
    android:id="@+id/toolbar3"
    android:layout_width="match_parent"
    android:layout_height="2dp"
    android:layout_gravity="center_horizontal"
    android:layout_marginStart="15dp"
    android:layout_marginEnd="15dp"
    android:background="?attr/colorPrimary"
    android:minHeight="?attr/actionBarSize"
    android:theme="?attr/actionBarTheme" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:orientation="horizontal">
    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="25"
        android:layout_marginLeft="20dp"
        android:fontFamily="@font/nurir"
        android:text="Tel. 010-1234-5678"
        android:textColor="@color/black" />

        <TextView
            android:id="@+id/textView3"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:layout_marginRight="20dp"
            android:fontFamily="@font/nurir"
            android:text="KY System"
            android:textColor="@color/black" />
    </LinearLayout>

```

## 7.5 인트로

```

public class intro extends AppCompatActivity {
    private final FirebaseDatabase database = FirebaseDatabase.getInstance();
    citiesDatabase cities_db;
    GraphDatabase graph_db;
    Thread DataThread = new Thread() {

```

```

@Override
public void run() {
    super.run();
    cities_db = new citiesDatabase(getApplicationContext());
    graph_db = new GraphDatabase(getApplicationContext());
    cities_db.open();
    graph_db.open();
    cities_db.create();
    graph_db.create();
    getDataFromFirebase();
    getValuesOfRegions();
}
};

Thread IntroThread = new Thread(){
    @Override
    public void run() {
        super.run();

        try {
            Thread.sleep(5000);
            Intent intent = new Intent(intro.this, MainActivity.class);
            startActivity(intent);
            finish();
        }
        catch (Exception ignored){}
    }
};

Thread Loading = new Thread(){
    @Override
    public void run() {
        super.run();
        Intent intent = new Intent(intro.this, Loading.class);
        startActivity(intent);
    }
};

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    //글씨 fading
    setContentView(R.layout.activity_intro);
    TextView t1 = findViewById(R.id.title);
    TextView t2 = findViewById(R.id.t2);
    TextView t3= findViewById(R.id.t3);
    Animation translate = AnimationUtils.loadAnimation(this, R.anim.alpha);
    t1.startAnimation(translate);
    t2.startAnimation(translate);
    t3.startAnimation(translate);

    // 인트로 로딩 회전
    TextView introLoader= findViewById(R.id.introload);
    Drawable introloader = introLoader.getBackground();
    introloader.setAlpha(80);
    Animation rotate = AnimationUtils.loadAnimation(this, R.anim.rotate);
    introLoader.startAnimation(rotate);

```



```

        DataThread.start();
        IntroThread.start();

//        Handler handler = new Handler() {
//            @Override
//            public void handleMessage(Message msg) {
//                if (msg.what == 1) {
//                    Intent intent = new Intent(getApplicationContext(),
MainActivity.class);
//                    startActivity(intent);
//                    finish();
//                }
//            }
//        };
//        IntroThread t = new IntroThread(handler);
//        t.start();

    }

    public void getDataFromFirebase(){
        for (int i = 1; i <= 154; i++) {
            String count = String.format("%s", i);
            DatabaseReference myRef = database.getReference(count);

            myRef.addListenerForSingleValueEvent(new ValueEventListener() {
                @Override
                public void onDataChange(@NonNull DataSnapshot snapshot) {
                    Object value = snapshot.getValue(Object.class);
                    // 문장 양끝 중괄호 제거하는 코드
                    String local_string = String.valueOf(value).substring(1,
String.valueOf(value).length() - 1);

                    // 지역 정보 전체 가져오기
                    assert value != null;
                    String[] locations_information = local_string.split(", ");

                    // 해쉬맵 객체 생성
                    HashMap<String, String> hashMap = new HashMap<>();
                    // 해쉬맵에 값 저장
                    for (String sentence : locations_information) {
                        String[] stringArr = sentence.split("=");
                        hashMap.put(stringArr[0], stringArr[1]);
                    }

                    double D_latitude =
Double.parseDouble(Objects.requireNonNull(hashMap.get("위도")));
                    double D_longitude =
Double.parseDouble(Objects.requireNonNull(hashMap.get("경도")));
                    double D_sunshineduration =
Double.parseDouble(Objects.requireNonNull(hashMap.get("시군구일사량평균")));
                    double D_stationNum =
Double.parseDouble(Objects.requireNonNull(hashMap.get("발전소수량")));
                    double D_hazard =
Double.parseDouble(hashMap.get("산사태취약지역(개수)") +
hashMap.get("산사태취약지역(고속도로 근처)"));
                    double D_architecture =

```

```

Double.parseDouble(Objects.requireNonNull(hashMap.get("시군구별 건축허가현황")));
String city_do = hashMap.get("도*광역시");
String city_sigungu = hashMap.get("시군구");
String anotherInfo = D_latitude + " " + D_longitude + " " +
D_sunshineduration + " " + D_stationNum + " " + D_hazard + " " + D_architecture;

Cursor cities_cursor = cities_db.selectWhatIWant(city_do,
city_sigungu);
if (cities_cursor.getCount() == 0) {
    cities_db.insertColumn(city_do,
city_sigungu,anotherInfo);
} else {
    cities_db.UpdateAll(city_do, city_sigungu,anotherInfo);
}

// 년도와 값이있는 데이터 들만 해쉬에 저장
HashMap<String, String> tmpHash = new HashMap<>();
for (String key : hashMap.keySet()) {
    String val = hashMap.get(key);
    if (!key.equals("시군구") && !key.equals("도*광역시") &&
!key.equals("시군구일사랑평균") && !key.equals("경도") && !key.equals("위도")
&& !key.equals("grade") && !key.equals("시군구별
건축허가현황") && !key.equals("발전소수량") && !key.equals("산사태취약지역 (고속도로
근처)"))
        && !key.equals("산사태취약지역 (개수)")) {
        tmpHash.put(key, val);
    }
}

//그래프에 관한 데이터 저장

Cursor graph_cursor = graph_db.selectedWhatIWant(city_do + "
" + city_sigungu);
if (graph_cursor.getCount() == 0) {
    for (int j = 2015; j < 2015 + tmpHash.size() / 2; j++) {
        graph_db.insertColumn(city_do + " " + city_sigungu,
String.valueOf(j), tmpHash.get(String.valueOf(j)), String.valueOf(j),
tmpHash.get(j + "01(전년누계대비 증감률)"));
    }
} else {
    for (int j = 2015; j < 2015 + tmpHash.size() / 2; j++) {
        graph_db.UpdateAll(city_do + " " + city_sigungu,
String.valueOf(j), tmpHash.get(String.valueOf(j)), String.valueOf(j),
tmpHash.get(j + "01(전년누계대비 증감률)"));
    }
}

@Override
public void onCancelled(@NonNull DatabaseError error) {

}
});

```

```

    }
}

public void getValuesOfRegions() {
    generatingDatabase generatingDatabase = new
generatingDatabase(getApplicationContext());
    generatingDatabase.open();
    generatingDatabase.create();
    final FirebaseDatabase[] database = {FirebaseDatabase.getInstance()};

    for(int i=1;i<=44;i++) {
        DatabaseReference myRef =
database[0].getReference("발전량-"+String.valueOf(i));

        myRef.addListenerForSingleValueEvent(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot snapshot) {

                Object value = snapshot.getValue();
                System.out.println("generating amount = ");
                String a = String.valueOf(value).substring(1,
String.valueOf(value).length() - 1);
                String[] splitedByEqual = a.split("=");

                for(int i=0;i<splitedByEqual.length;i++){
                    if(i == 1){
                        splitedByEqual[i] =
splitedByEqual[i].substring(1,splitedByEqual[i].length() - 5);
                    }
                }

                Cursor cursor =
generatingDatabase.selectWhatIWant(splitedByEqual[2]);
                if (cursor.getCount() == 0) {

generatingDatabase.insertColumn(splitedByEqual[2],splitedByEqual[1]);
                } else {

generatingDatabase.UpdateColumn(splitedByEqual[2],splitedByEqual[1]);

                }

            }

            @Override
            public void onCancelled(@NonNull DatabaseError error) {

            }

        });
    }
}
}

```

## XML

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"

```

```

android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="@drawable/background_intro"
android:gravity="center"
android:orientation="vertical">

<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_centerVertical="true"
        android:orientation="vertical">

        <TextView
            android:id="@+id/title"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:layout_marginBottom="30dp"
            android:background="@android:color/transparent"
            android:fontFamily="@font/nurir"
            android:gravity="center"
            android:text="@string/title"
            android:textSize="30sp" />

        <TextView
            android:id="@+id/t2"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal"
            android:layout_marginBottom="20dp"
            android:fontFamily="@font/nurir"
            android:gravity="center_horizontal"
            android:text="@string/Extra_Subject"
            android:textSize="19sp" />

        <TextView
            android:id="@+id/t3"
            android:layout_width="match_parent"
            android:layout_height="20dp"
            android:layout_gravity="center_horizontal"
            android:background="@android:color/transparent"
            android:fontFamily="@font/nurir"
            android:gravity="center_horizontal"
            android:text="@string/beginningOfApp"
            android:textSize="15sp" />

    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="end">

        <TextView

```

```

        android:id="@+id/introload"
        android:layout_width="70dp"
        android:layout_height="70dp"
        android:layout_gravity="bottom"
        android:layout_marginRight="25dp"
        android:layout_marginBottom="15dp"
        android:background="@drawable/introloading" />

    </LinearLayout>
</RelativeLayout>
</LinearLayout>

```

## 7.6 로딩

```

import androidx.appcompat.app.AppCompatActivity;

import android.annotation.SuppressLint;
import android.content.Intent;
import android.media.Image;
import android.os.Bundle;
import android.os.Handler;
import android.os.Message;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.ImageView;
import android.widget.TextView;

import com.bumptech.glide.Glide;

public class Loading extends AppCompatActivity {

    @SuppressLint("SetTextI18n")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_loading);

        Intent previous_activity = getIntent();
        String activity_checker = previous_activity.getStringExtra("activity");
        TextView t1 = findViewById(R.id.ring);
        Animation translate = AnimationUtils.loadAnimation(this, R.anim.rotate);

        //팁 띄우기
        TextView tiptext = findViewById(R.id.TipText);
        final String[] tips = {"등급별로 1단계에서 8단계까지 나누어져 있어요", "순위  
페이지에서는 각 도*광역시의 발전소설치지역의 순위를 나누어줘요",  
        "그래프페이지의 그래프들은 선택한 지역의 정보예요", "등급별 버튼을 누르면  
등급에 해당하는 지역을 마커로 찍어줘요"};
        int idx = (int) (Math.random()*(tips.length-1));
        tiptext.setText("Tip : "+tips[idx]);

        t1.startAnimation(translate);
        Handler handler = new Handler() {
            @Override

```

```

        public void handleMessage(Message msg) {
            if (msg.what == 1) {
                finish();
            }
        }
    };
    loadingThread t = new loadingThread(handler,activity_checker);
    t.start();
}
class loadingThread extends Thread {
    private Handler handler;
    public loadingThread(Handler handler,String activity) {
        this.handler = handler;this.activity = activity;
    }
    private String activity;
    @Override
    public void run() {
        super.run();
        Message ms = new Message();
        try {
            //Thread.sleep(3500);
            if(activity.equals("main")) {
                Thread.sleep(4000);
            }else if(activity.equals("graph")){
                Thread.sleep(1500);
            }
            ms.what = 1;
            handler.sendMessage(ms.what);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
}

```

## XML

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.App.SolarPing.Loading"
    android:gravity="center"
    android:background="#ffffff"
    android:orientation="vertical"
    android:id="@+id/backgr"
    >
    <RelativeLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        >
        <ImageView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:background="#ffffff"/>

        <LinearLayout
            android:id="@+id/linearLayout1"
            android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:orientation="vertical" android:layout_centerHorizontal="true"
android:layout_centerVertical="true">
    <TextView
        android:id="@+id/ring"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:background="@drawable/loading"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="        loading...."
        android:textColor="@color/black"
        android:textSize="15sp" />

</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="bottom"
    >

    <TextView
        android:id="@+id/TipText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textAlignment="center"
        android:layout_marginStart="40dp"
        android:layout_marginEnd="40dp"
        android:layout_marginBottom="20dp"
        android:gravity="bottom"
        android:textColor="@color/black"
        android:textSize="15sp">

        </TextView>
    </LinearLayout>
</RelativeLayout>

</LinearLayout>

```