**인턴 오재우 “Forget about it” 어플리케이션 최종 보고서**

**목차**

2

2

2

2

2

3

3

3

3

4

4

4

4

4

4

4

5

6

6

6

6

1. 기획 단계 ...........................................................................................................................................................
   1. 어플리케이션 이름 ................................................................................................................................
   2. 어플리케이션 설명 ................................................................................................................................
   3. 활용 공공데이터 API 설명 ................................................................................................................
      1. 기상청 중기예보정보 조회 서비스 API ..............................................................................
      2. 기상청 동네예보정보 조회 서비스 API ..............................................................................
      3. 기상청 생활기상지수 조회 서비스 API ..............................................................................
      4. 한국수자원공사 실시간 수도정보 API .................................................................................
      5. 한국환경공단 대기오염정보 조회 서비스 API .................................................................
      6. 한국환경공단 측정소정보 조회 서비스 API .....................................................................
   4. 주요 기능 설명 .......................................................................................................................................
2. 거주지역 설정 (위치서비스, 수동) .........................................................................................
3. 매시간 데이터 자동 업데이트 .................................................................................................
4. 푸시 알림 항목 설정 ...................................................................................................................
5. 푸시 알림 시간 설정 ...................................................................................................................
6. 위험 수준 도달 시 푸시 알림 기능 .....................................................................................
   1. UI 예시 .......................................................................................................................................................
7. 개발 단계 ...........................................................................................................................................................
   1. 공공데이터 API 개발신청 ..................................................................................................................
      1. 공공데이터 API 개발신청 방법 ..............................................................................................
      2. 공공데이터 API 개발신청 확인 ..............................................................................................
   2. 개발환경 .....................................................................................................................................................

6

7

7

7

7

8

8

9

* 1. 코드 구조 및 설명 .................................................................................................................................
     1. 어플리케이션 구조 및 개요 ......................................................................................................
        1. Java class ...................................................................................................................................
        2. Layout (xml) .............................................................................................................................
        3. Manifest .....................................................................................................................................
        4. Gradle .........................................................................................................................................
     2. 각 파일 별 설명 .............................................................................................................................
  2. 진행 상황 ...................................................................................................................................................

1. **기획 단계**
2. 어플리케이션 이름

Forget about it!

1. 어플리케이션 설명

“앱을 깔고 신경을 끄세요.”

앱 하나로 물, 자외선, 비, 미세먼지에 대한 걱정이 사라집니다.

사용자가 설정한 지역의

사용자가 수질, 자외선 지수, 일기예보, 미세먼지 정보를 매시간마다 업데이트하여 위험수준 도달 시 푸시 알림을 보내며, 각 항목별로 알람을 원하는 시간대를 설정할 수 있습니다.

1. 활용 공공데이터 API 설명
2. 기상청 중기예보정보 조회 서비스 API
   1. 중기전망조회
      1. 기상전망을 텍스트 형식으로 출력한다.

(예시: 이번 예보기간에는 북태평양고기압의 영향으로 남부지방은 대체로 맑겠으나, 중부지방은 그 가장자리에 들어 구름이 많겠습니다…)

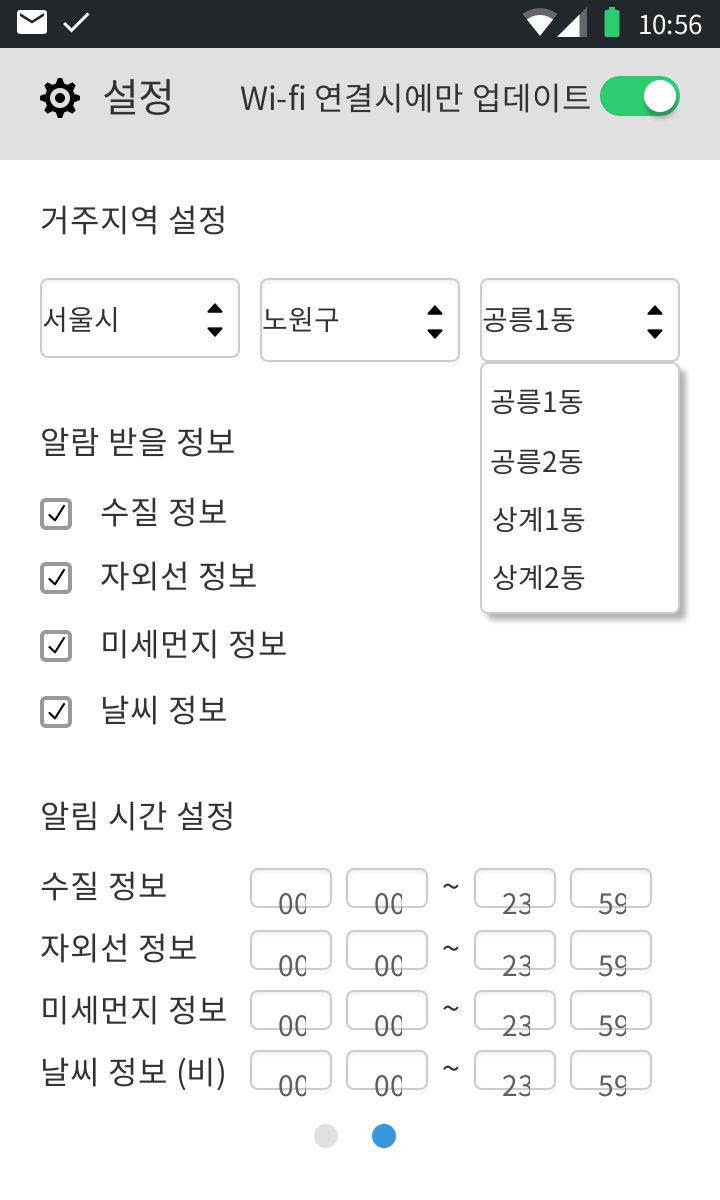
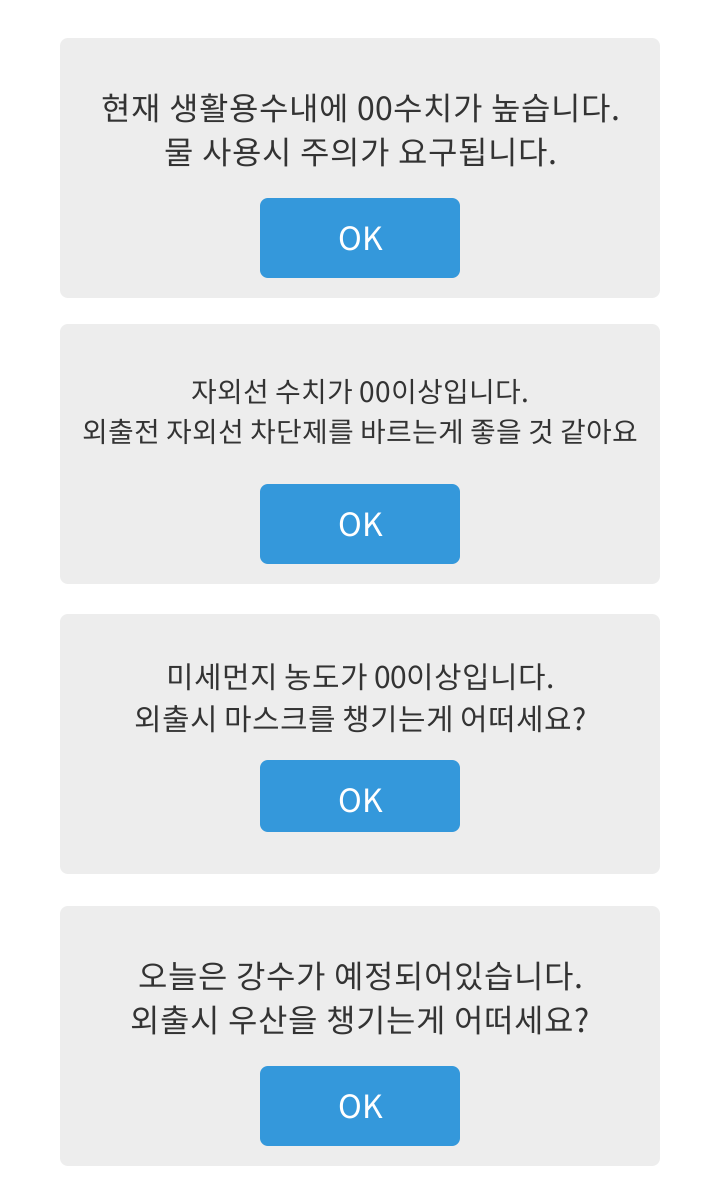
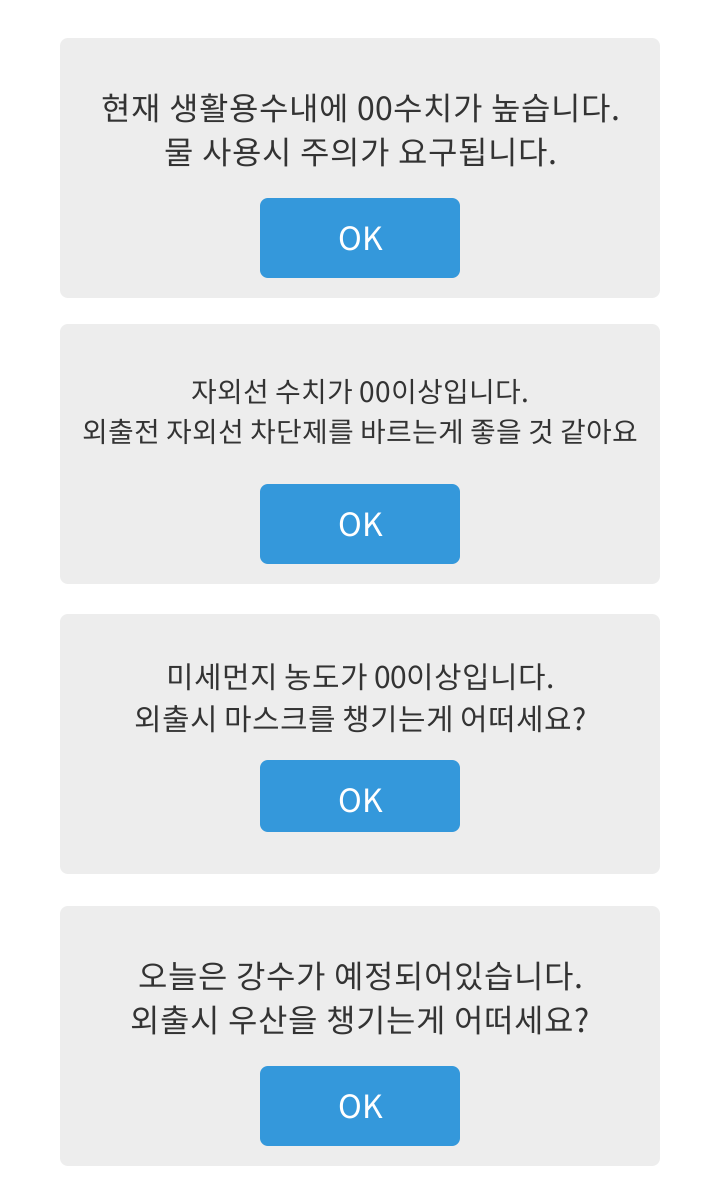
* 1. 중기기온조회
     1. 3~10일 후의 최저/최고기온을 출력한다.
  2. 중기육상예보조회
     1. 3~10일 후의 날씨정보를 출력한다

(예시: 맑음, 구름많음, 비 등)

1. 기상청 동네예보정보 조회 서비스 API
   1. 초단기실황조회
      1. 현재 날씨정보(기온, 강수량, 하늘상태, 습도, 강수형태 등)를 출력한다.
   2. 초단기예보조회
      1. 1~3 시간 후의 날씨정보(기온, 강수량, 하늘상태, 습도, 강수형태 등)를 출력한다.
   3. 동네예보조회
      1. 현재부터 이틀 후까지의 날씨정보(강수확률, 강수형태, 습도, 하늘상태, 최저/최고기온 등)를 출력한다.
2. 기상청 생활기상지수 조회 서비스 API
   1. 더위체감지수, 열지수, 불쾌지수, 식중독지수, 자외선지수 등을 출력한다.
3. 한국수자원공사 실시간 수도정보 API
   1. 공급지역 정수장 코드 조회 서비스
      1. 정수장과 정수장의 위치(주소)를 제공한다
   2. 정수장 코드 조회 서비스
      1. 정수장과 정수장 코드 정보를 제공한다.
   3. 1시간 수질정보 조회 서비스
      1. 입력한 특정 정수장의 수질 정보를 출력한다.
4. 한국환경공단 대기오염정보 조회 서비스 API
   1. 측정소별 실시간 측정정보 조회
      1. 측정소명을 입력하여 실시간 대기오염정보를 출력한다.
      2. 위치서비스를 이용한 대기정보 출력 시 활용
   2. 시군구별 실시간 평균정보 조회
      1. 시도별 검색을 통해 시군구별 실시간 대기오염 정보를 출력한다.
      2. 직접 위치 설정 시 활용
   3. 대기질 예보통보 조회
      1. 오늘부터 모레까지의 대기오염 정보를 출력한다.
      2. 대기질 예보에 활용
5. 한국환경공단 측정소정보 조회 서비스 API
   1. TM 기준좌표 조회
      1. 읍면동 이름으로 TM 좌표를 출력한다.
      2. 해당 정보를 근접측정소 목록 조회에 사용한다.
   2. 근접측정소 목록 조회
      1. TM 좌표로 근접 측정소 목록을 출력한다.
      2. 해당 정보를 측정소별 실시간 측정정보 조회에 사용한다.
6. 주요 기능 설명
7. 거주지역 설정
   1. 위치서비스 혹은 직접 본인의 지역을 설정한다.
8. 매시간 데이터 자동 업데이트
   1. 정해진 시간마다 로컬DB를 최신화한다.
9. 푸시 알림 항목 설정
   1. 사용자가 원하는 푸시 알림 항목을 설정한다.
10. 푸시 알림 시간 설정
    1. 사용자가 푸시 알림을 받길 원하는 시간대를 설정한다.
11. 위험 수준 도달 시 푸시 알림 기능
    1. 사용자가 설정한 항목과 시간대에 수치가 위험 수준에 도달하면 사용자에게 푸시 알림을 보낸다.
12. UI 예시

<설정 화면>

<메인 화면>



<푸시 알림 예시 2>

<푸시 알림 예시 1>

1. **개발 단계**
2. 공공데이터 API 개발신청
3. 공공데이터 API 개발신청 방법
4. 공공데이터 포털(<https://www.data.go.kr/>)에 접속한다.
5. 계정 생성 후, 원하는 데이터를 검색한다.
6. 해당 데이터를 클릭한 후, 활용신청 버튼을 클릭한다.

(대부분의 데이터가 자동승인으로 바로 인증키가 발급되나, 최초 2,3시간까지 인증키 오류가 발생할 수 있다. 자동승인 데이터가 아닌 경우, 활용 허가를 받을 때까지 기다려야 한다.)

1. 공공데이터 API 개발신청 확인
2. 개발환경
3. Android Studio 3.1.4
4. Android Studio 파일을 다운로드 받는다. (<https://developer.android.com/studio/?hl=ko>)
5. 다운로드한 파일을 실행시킨다.
6. 설치 마법사의 안내에 따라 Android Studio를 설치한다.

(JDK가 설치된 위치를 런처 스크립트가 찾지 못 할 경우 환경 변수를 따로 설정해주어야 한다. Start 메뉴 > Computer > System Properties > Advanced System Properties를 선택한 후, Advanced 탭 > Environment Variables를 열고 JDK 폴더(예시: C:\Program Files\Java\jdk1.8.0\_77)를 가리키는 새 시스템 변수 JAVA\_HOME을 추가한다.

1. 새로운 도구와 기타 API가 나오게 되면, Android Studio가 팝업으로 알려준다. 또는 Help > Check for Update를 클릭하여 업데이트를 확인할 수 있다.
2. 코드 구조 및 설명
3. 어플리케이션 구조 및 개요
4. Java class: 어플리케이션의 기능을 구현한다.
5. layout(xml): 어플리케이션의 layout (UI)을 구현한다.
6. Manifest: Android 시스템이 앱의 코드를 실행하기 전에 확보해야 하는 앱에 대한 필수 정보를 시스템에 제공한다. (예시: 어플리케이션 이름, 구조, 안드로이드 버전, 권한 등)
7. AndroidManifest.xml
8. Gradle: Java class 파일을 안드로이드 환경에서 별도의 설정없이 빌드하여 실행할 수 있도록 도와주는 빌드 툴로, design/layout 등의 버전을 설정한다.
9. build.gradle (app)
10. 각 파일 별 설명

**MainActivity (Java Class)**

* 어플리케이션의 main activity를 담당하는 자바 클래스 파일로, 전체적인 UI와 틀을 제공
* activity\_main xml 파일을 인자로 UI를 제공
* 총 두개의 페이지를 생성하여 각각 main과 setting layout을 return
* 실제 코드

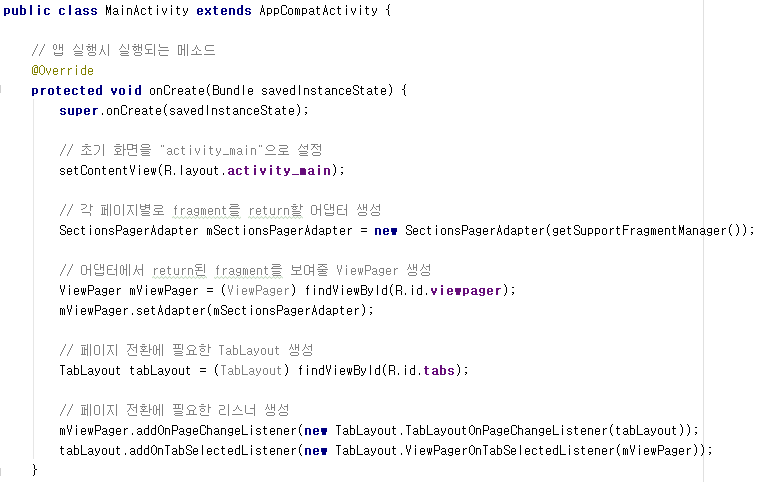
`

**Position=1**

`

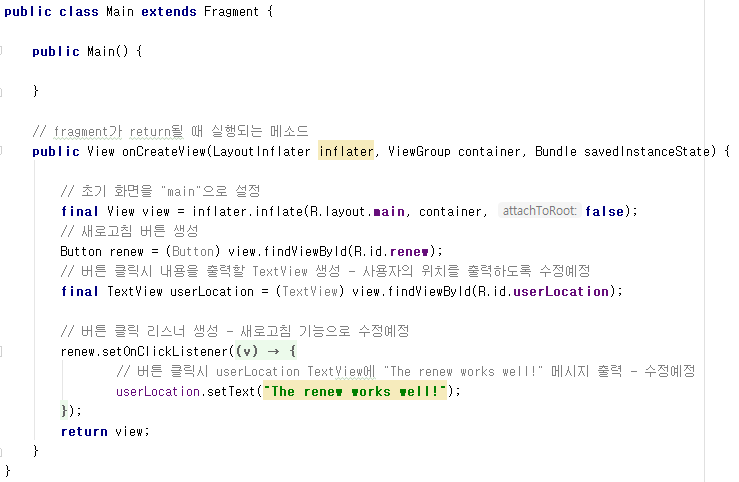
**Position=0**

`



**Main (Java Class)**

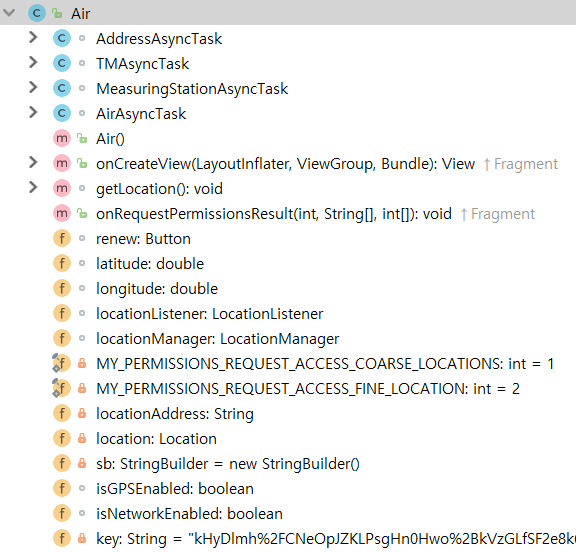
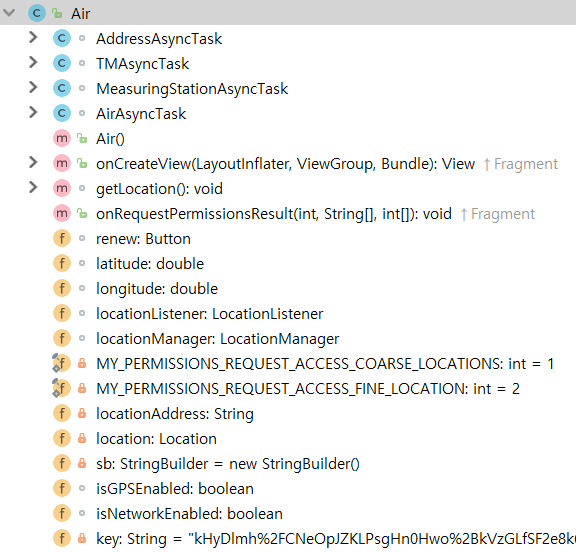
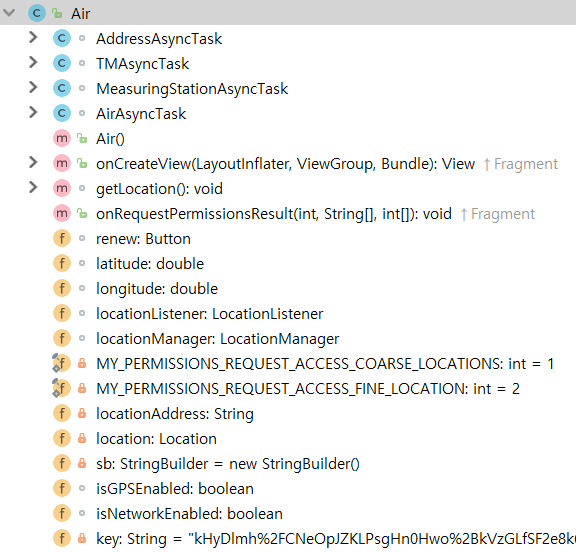
* MainActivity 클래스에서 페이지 번호가 0일 때 return되는 fragment
* main xml 파일을 인자로 water, lifeIndex, air, weather layout을 한 화면에서 출력
* 실제 코드

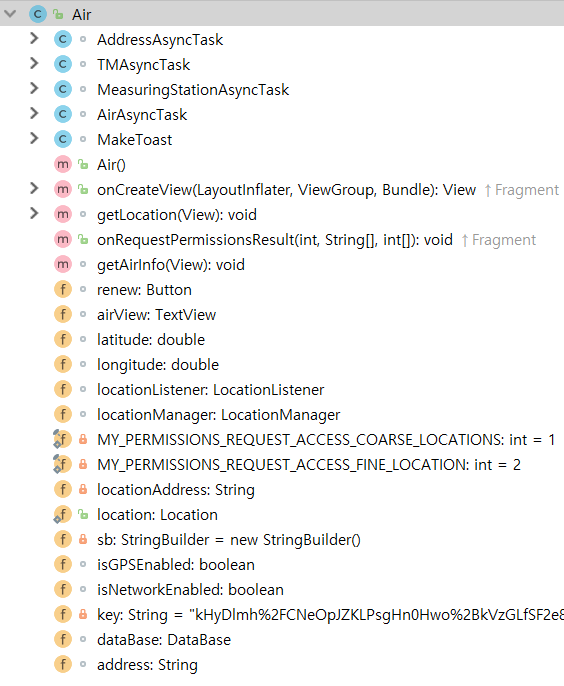




**Air (Java Class)**

* air xml 파일을 인자로 미세먼지 정보를 출력하는 fragment
* main layout 상에서 출력
* 클래스 구성

는 클래스, 는 생성자와 메소드, 는 필드를 나타낸다.

`

**주요 기능**

* 세부 코드 설명: 기본 생성자 선언과 필드 선언 (Constructor, Field)



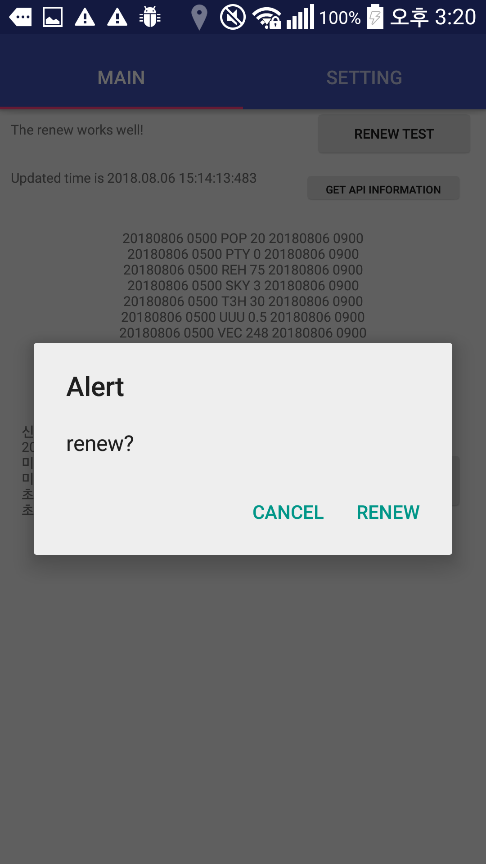


오류 발생을 막기위해 기본 생성자를 생성해주고, 클래스에 필요한 필드들을 선언해준다.

* 세부 코드 설명: onCreateView (Method) – 1

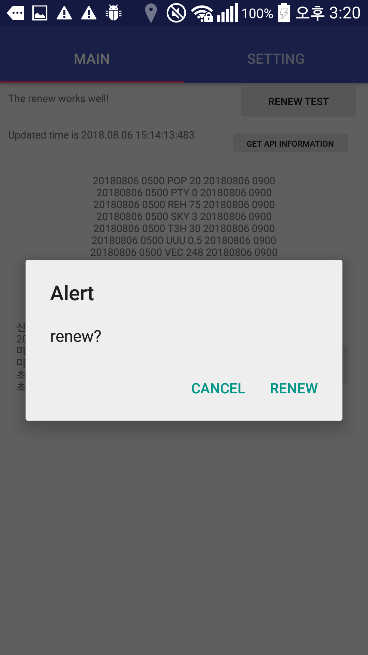
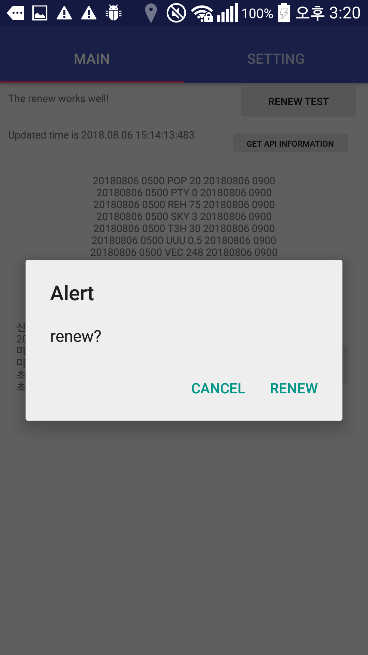
onCreateView는 해당 클래스(fragment)가 실행될 때 View를 설정해주는 메소드이다. 따라서 초기화면을 설정해주고, 자바 클래스의 main 메소드와 비슷한 역할을 한다.

onCreateView 메소드는 버튼 클릭시 AlertDialog를 통해 사용자에게 실행 여부를 확인한 뒤, 위치서비스를 통해 위경도 좌표를 얻어오는 getLocation 메소드를 실행한다. 성공적으로 좌표를 얻어와 해당정보가 location 필드에 저장이 되어있으면, 위경도 좌표를 통해 주소를 얻어오는 AddressAsyncTask 클래스, 주소를 통해 TM 좌표를 얻는 TMAsyncTask 클래스, TM 좌표를 통해 인근 측정소 정보를 얻는 MeasuringStationAsyncTask 클래스, 측정소를 통해 대기오염 정보를 얻어 출력하는 AirAsyncTask가 순서대로 실행된다. getLocation 메소드가 제대로 실행되지 않아 location 필드가 “null” 상태이면, 아무 작동없이 사용자의 다음 행동을 기다린다.

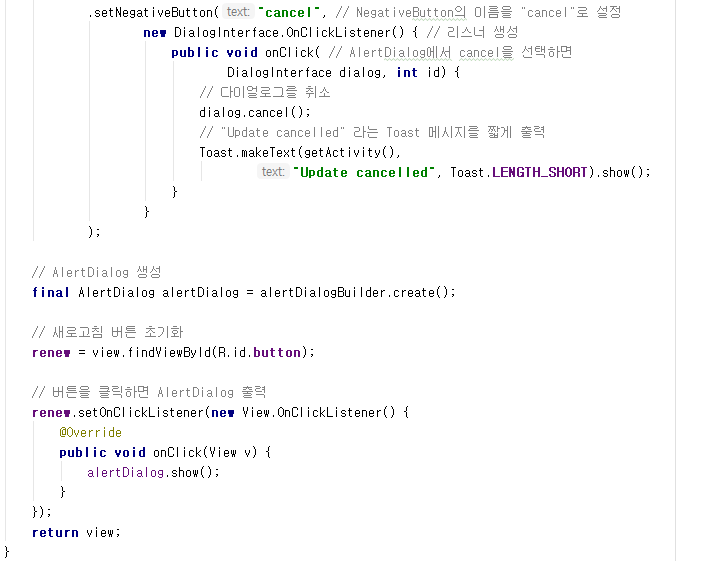
`



뒷장에 계속

* 세부 코드 설명: onCreateView (Method) - 2

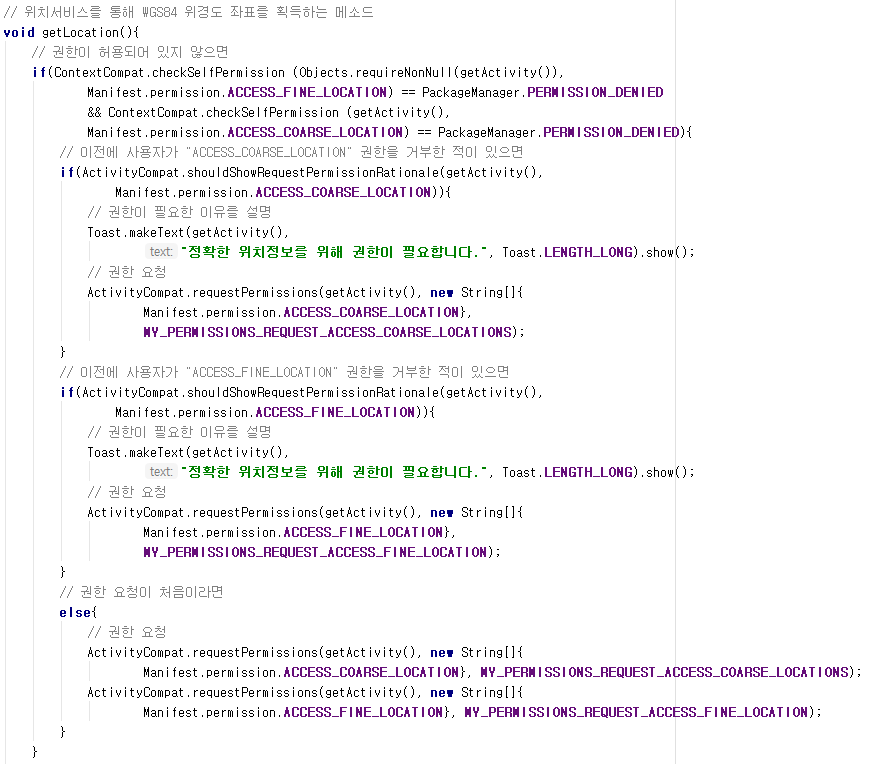
**주요 기능**



* 세부 코드 설명: getLocation (Method) - 1

뒷장에 계속

**권한 요청**



getLocation은 위치서비스를 통해 위경도 좌표를 얻는 메소드이다. 이 메소드가 제대로 실행되지 않으면 위경도 좌표를 통해 주소를 얻어오는 서브클래스인 “AddressAsyncTask”가 제대로 실행되지 않기 때문에 가장 핵심적인 메소드라 할 수 있다.

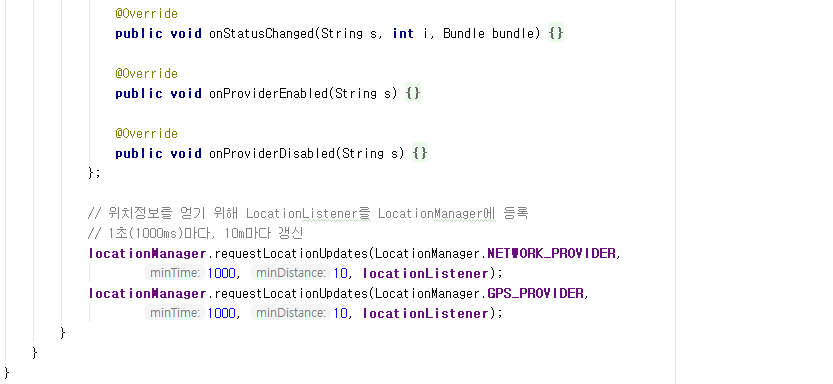
Run Console 창에서 "The application may be doing too much work on its main thread" 메시지가 출력되게 하는 요인일 가능성이 있고, 위치정보는 현재 구현중인 다른 클래스들에서도 사용될 예정이기 때문에 하나의 메인 클래스로 만드는 것을 고려하고 있다.

getLocation 메소드는 onCreateView 메소드에서 실행되면 해당 기기의 위치서비스 권한 허용여부를 확인한 뒤, 위치정보를 갱신한다.

* 세부 코드 설명: getLocation (Method) - 2

**위치 갱신**

**위치서비스 요청**



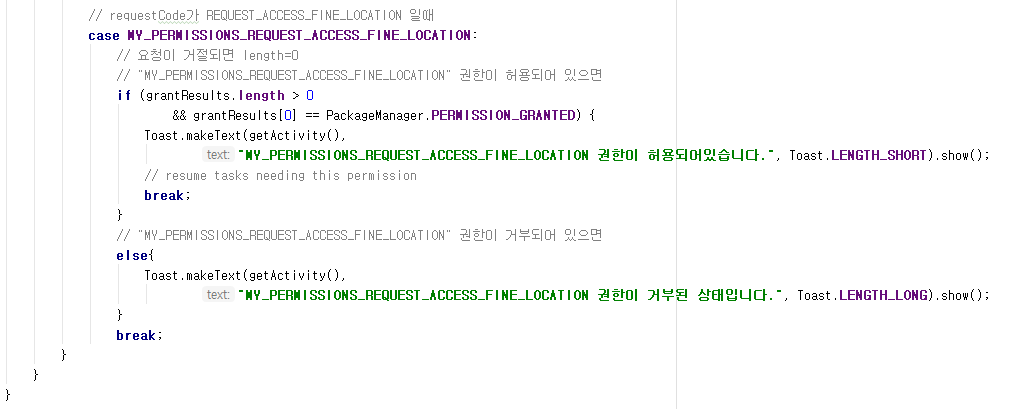
* 세부 코드 설명: onRequestPermissionsResult (Method)

onRequestPermissionsResult는 권한의 유무를 확인하고 권한이 있을 시 해당 권한을 요청하는 메소드이다.

onRequestPermissionsResult 메소드는 따로 호출하지 않아도 권한 유무 확인 및 요청이 필요할 시 프로그램이 자동으로 호출하여 권한 유무에 따라 메소드 내에서 오버라이딩으로 지정된 동작을 실행한다.

현재 코드에서는 권한의 종류와 허용 유무에 따라 사용자에게 권한 허용유무를 알리는 간단한 Toast 메시지가 출력되도록 했으며, 필요에 따라 기능을 추가할 예정이다.

**권한에 따른 구분**



* 세부 코드 설명: AddressAsyncTask (Sub Class)

AddressAsyncTask는 getLocation 메소드에서 얻은 위치정보를 활용해 주소를 구하는 클래스이다. 이를 위해 위경도로 주소를 구하는 기본 Library Geocoder을 사용하였다.

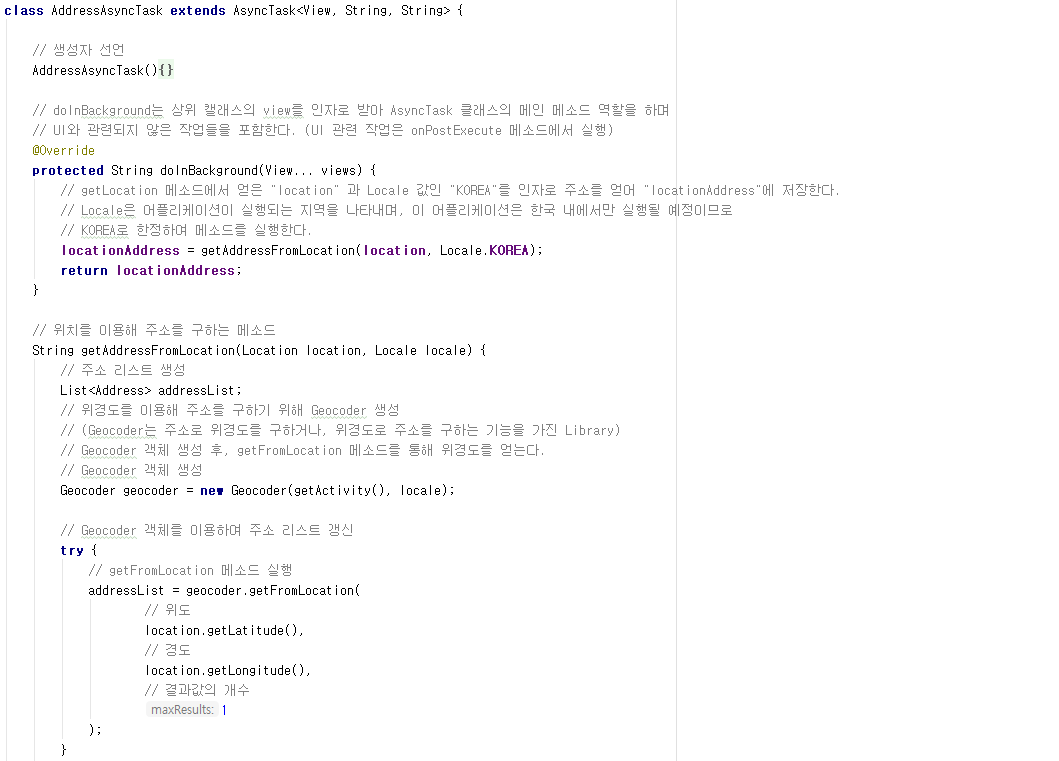
메인클래스인 Air의 onCreateView에서 Geocoder의 메소드인 getFromLocation을 실행하면 UI관련 쓰레드에서 네트워크 작업을 하면서 충돌이 일어나기 때문에 따로 Asynctask 서브 클래스를 만들어 메소드를 실행한다.

**메소드 실행 (주소 구하기)**

**저장 후 리턴**

**객체 생성**

**getLocation에서 받은 위치정보**



* 세부 코드 설명: TMAsyncTask (Sub Class) - 1

`

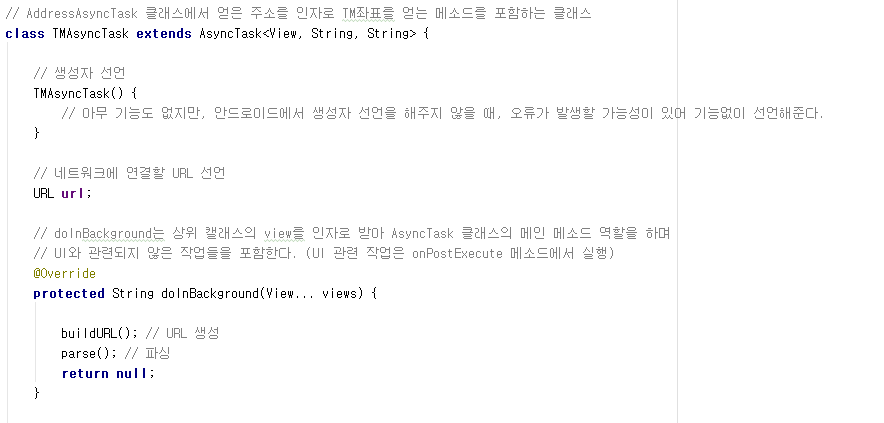
**URL 생성**

**URL 생성 메소드**

뒷장에 계속

`

**메소드 실행 (URL 생성 후 파싱)**



TMAsyncTask는 AddressAsyncTask에서 Geocoder의 메소드 getFromLocation으로 얻은 주소를 이용하여 TM좌표를 구하는 클래스이다. 이를 위해 한국환경공단에서 제공하는 “측정소정보 조회 서비스”의 “TM 기준좌표 조회” API를 활용한다.

doInBackground 메소드 내에서 buildURL 메소드와 parse 메소드를 실행시켜 각각 URL을 만들고 그 URL을 이용해 네트워크 접속 후, 파싱을 통해 TM좌표를 원하는 형태(tmX tmY)로 가져온다.

* 세부 코드 설명: TMAsyncTask (Sub Class) – 2



* 세부 코드 설명: TMAsyncTask (Sub Class) – 3



* 세부 코드 설명: MeasuringStationAsyncTask (Sub Class)

TMAsyncTask에서 얻은 TM좌표를 통해 인근 측정소 정보를 가져온다.

* 세부 코드 설명: AirAsyncTask (Sub Class)

MeasuringStationAsyncTask에서 얻은 측정소에서 측정된 미세먼지 데이터를 가져온다.

* 세부 코드 설명: MakeToast (Sub Class)

Toast 메시지를 출력할 때, 쓰레드 관련 오류 발생을 방지하기 위해 생성자로 선언해준다.

* 세부 코드 설명: getAirInfo (Method)

미세먼지 데이터를 얻기 위한 getLocation 메소드와 AddressAsyncTask, TMAsyncTask, MeasuringStationAsyncTask, AirAsyncTask 클래스를 포함하는 메소드

PushPopup 클래스에서 매 시각 파싱 기능을 구현하기 위해 만들었으나, 미구현

Water, LifeIndex, Weather

* 구현중 혹은 미구현, Air 클래스와 흡사한 구조를 가지므로 생략

Setting

* setting xml 파일을 인자로 사용자 설정 정보를 출력하는 fragment
* 어플리케이션의 설정(거주지역, 알림 시간 등)과 관련된 기능 제공
* 사용자는 거주지역 설정 기능만 사용가능
* 알림 시간 설정 UI는 있으나, 기능 미구현
* 엑셀파일에서 데이터를 읽어 로컬DB에 저장하는 기능 (앱 최초 실행 시 동작)
* 매 시각 5분마다 알람(기능 요청)을 보냄 (이 부분에 파싱 기능 추가되어야 함)

Database

* openDB 메소드에서 DB가 없으면 새로 DB를 생성한다
* 현재 구현된 미세먼지 데이터(“Air”), 엑셀파일에서 읽어오는 행정구역 데이터(“SiSI”, “SiSiU”)의 테이블(총 3개)을 onCreate 메소드에서 자동으로 생성한다.
* 행정구역 데이터(“SiSI”, “SiSiU”)는 앱 최초 실행시 자동으로 저장되며 1~2분의 시간이 소요된다
* 미세먼지 데이터(“Air”)는 앱에서 업데이트 버튼을 누르면 데이터 파싱을 통해 얻은 미세먼지 정보를 업데이트한다.
* 각 테이블에 대한 insert, update, delete, select, count 메소드가 구현되어있다.

PushPopup

* Setting 클래스에서 요청한 알람을 수신하는 클래스
* onReceive 메소드에서 매 시각 파싱 기능을 추가해야 한다
* 데이터가 기준치를 벗어날 때 팝업기능을 추가해야 한다

PopupActivity

* popup.xml 파일을 인자로 UI관련 작업을 하는 클래스 (미구현)

1. 진행 상황
2. 공공데이터 데이터 파싱 및 로컬DB 저장 기능 – 위치데이터 및 파싱 메소드 구현
3. 매 시각 5분 후 마다 자동으로 파싱 및 설정된 시간내 알림 기능 - 구현 중단
4. 공공데이터 분석 – 디테일 부족
5. 최종 보고서 – 미완성