**프로젝트 최종 보고서**

**Animalunteer**

팀 : DUNKIN

IT정보공학과 201812730 김다영

IT정보공학과 201812763 박예린

**목차**

1. 개발 앱의 제목 및 목적

2. 앱 사용자 범위

3. 앱 구조 및 인터페이스 설명

4. 개발 내용

5. 향후 발전 계획

6. 참고 자료

7. 유튜브 링크

**1. 개발 앱의 제목 및 목적**

|  |  |
| --- | --- |
| 제목 | Animalunteer |
| 목적 | 유기동물을 위한 후원이나 봉사를 하고싶지만, 어디서 어떻게 해야 하는지 잘 모르는 사람들을 위해 신뢰성 있는 정보를 제공하는 것을 목적으로 만들었다. |

**2. 앱 사용자 범위**

- 전라도 거주자

- 평소 유기동물 봉사나 후원에 관심이 있는 사람

- 거주지 근처에 있는 유기동물 보호센터를 찾고자 하는 사람

- 봉사와 후원이 필요해 유기동물 보호센터의 홍보가 필요한 사람

**3. 앱 구조 및 인터페이스 설명**

|  |  |
| --- | --- |
| 지도이(가) 표시된 사진  자동 생성된 설명 |  |
| **Map Fragment** | **Recruit Fragment** |
|  | 텍스트이(가) 표시된 사진  자동 생성된 설명 |
| **Memo Fragment** | **Information fragment** |

* 전체 구성

- FrameLayout을 이용해서 세 가지 xml 파일이 BottomNavigationView에 따라 전환될 수 있도록 구현했다.

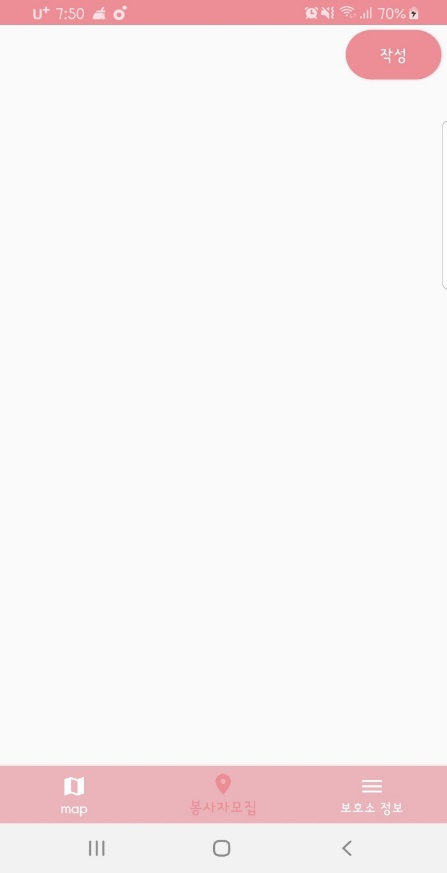
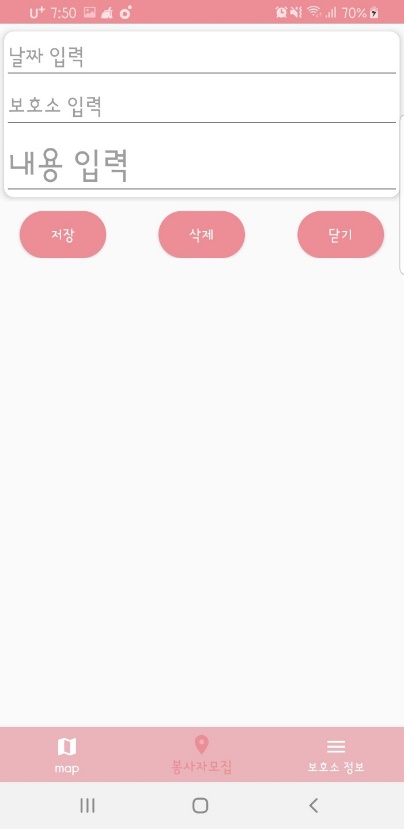
* Map Fragment

- 처음 앱이 실행되면 Map Fragment가 기본으로 보이고 마커 클릭 시 말풍선을 보여주고, 말풍선 클릭 시 관련 정보를 Dialog를 통해 띄워준다.  
지도이(가) 표시된 사진

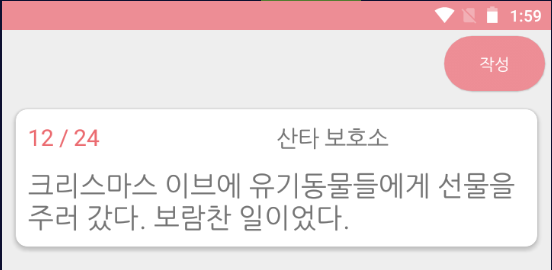
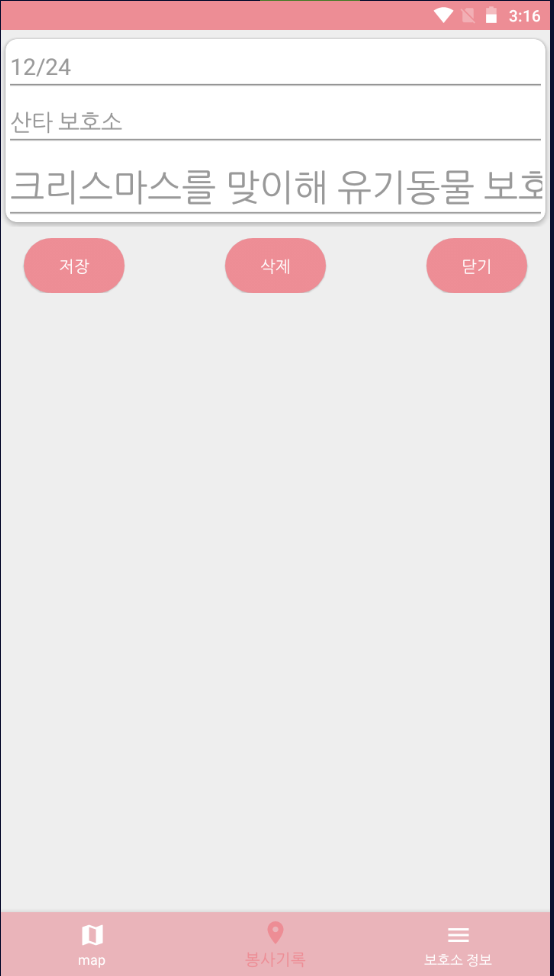
자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명

* Recruit Fragment layout & Memo Fragment

- 봉사자 모집 버튼을 누르면 Recrui Fragment layout이 나오게 되고, 버튼을 클릭하면 memo Fragment 화면이 나오면서 봉사에 대한 간단한 기록을 할 수 있게끔 만들었다.  
 

- 봉사 모집 글을 작성해 저장하면 Recruitment Fragment layout에 CardView를 통해 보여준다. 또 저장된 list를 클릭하면 수정화면으로 이동해서 내용을 수정할 수 있다.

* Information Fragment

- 보호소 정보 버튼을 누르면 앱의 목적에 관한 주요 내용들이 나온다.

**4. 개발 내용**

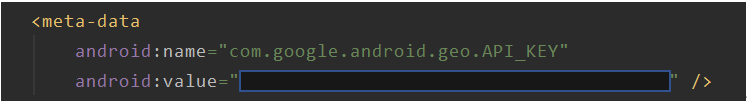
**1) BottomNavigationView**

1. Fragment 연동  
   - BottomNavigationView를 이용해 Fragment로 화면전환을 구성했다.  
     
   

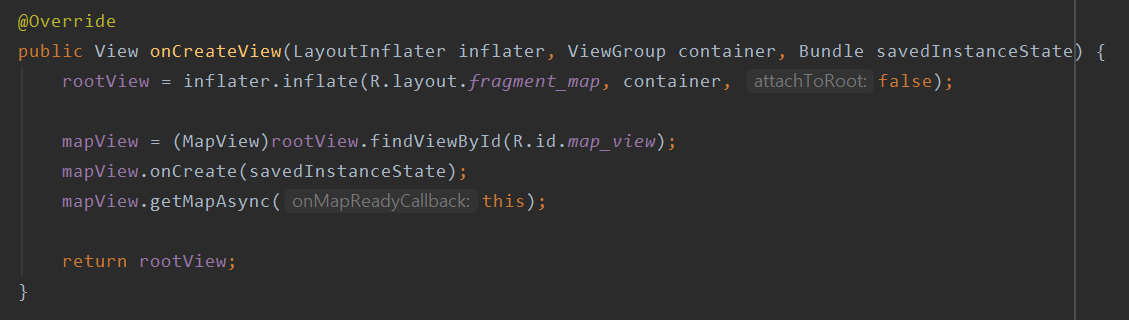
**2) Map Fragment**

1. GoogleMap API,

1) Google platform에서 받아온 API 키 (AndroidManifest.xml)

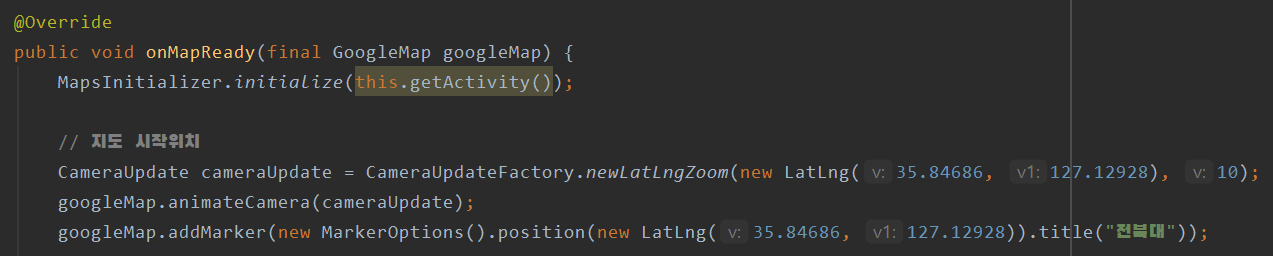


2) MapFragment.java



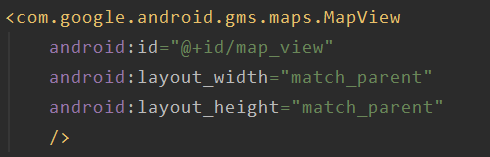
* GoogleMap 초기화
* getMapAsync : Map Callback 등록

2)

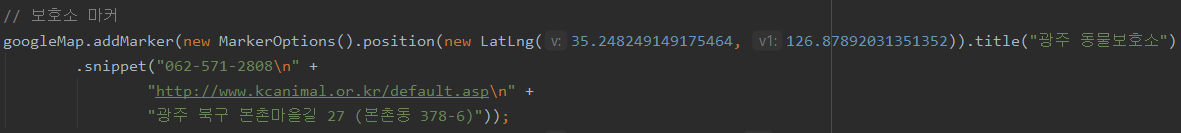
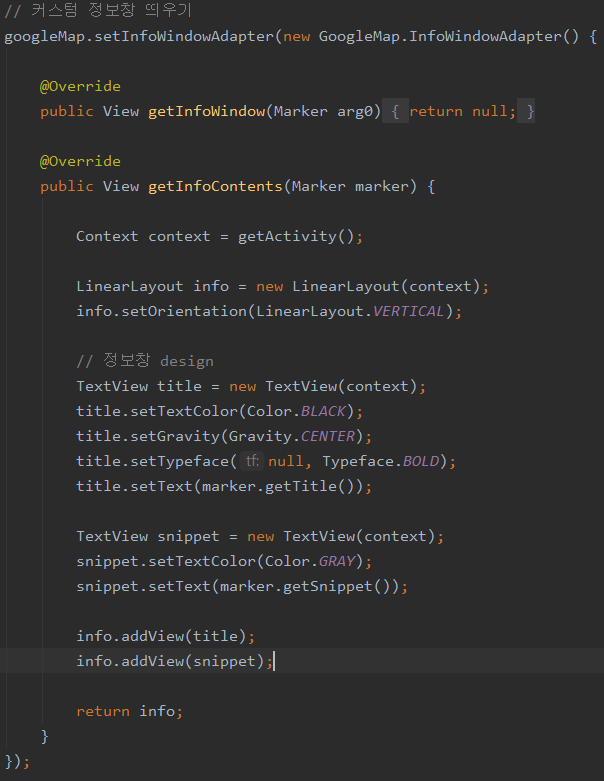
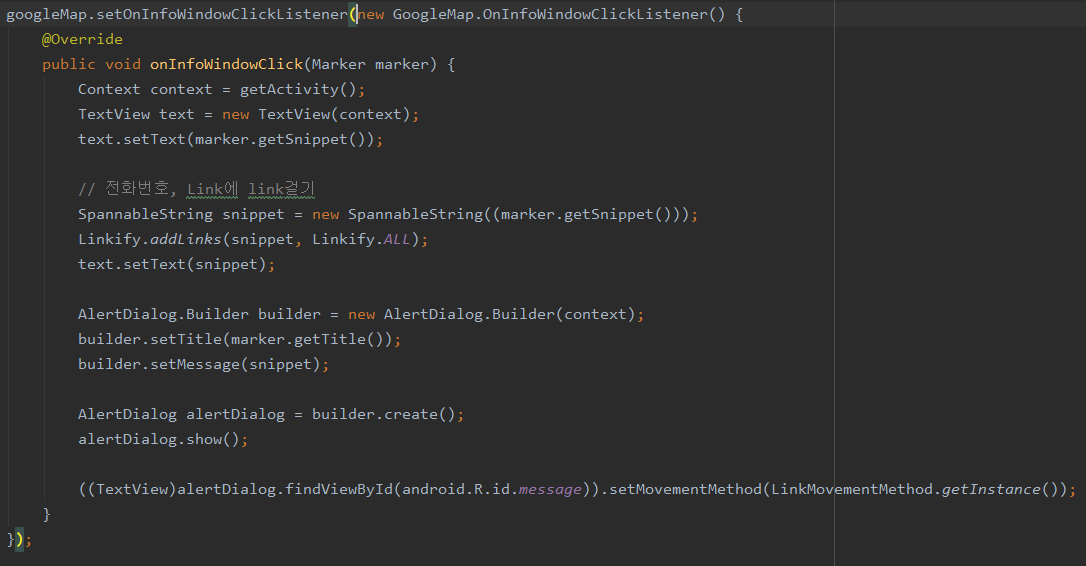
  
- OnMapReadyCallback interface를 구현하여 onMapReady method 정의

* 지도의 시작 위치를 전북대로 설정

3) fragment\_map.xml

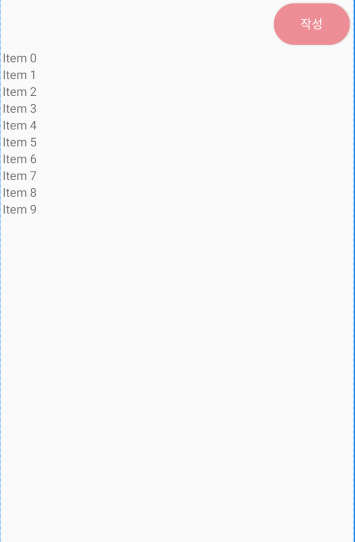


* GoogleMap을 Fragment로 띄우기 위해 MapView 사용

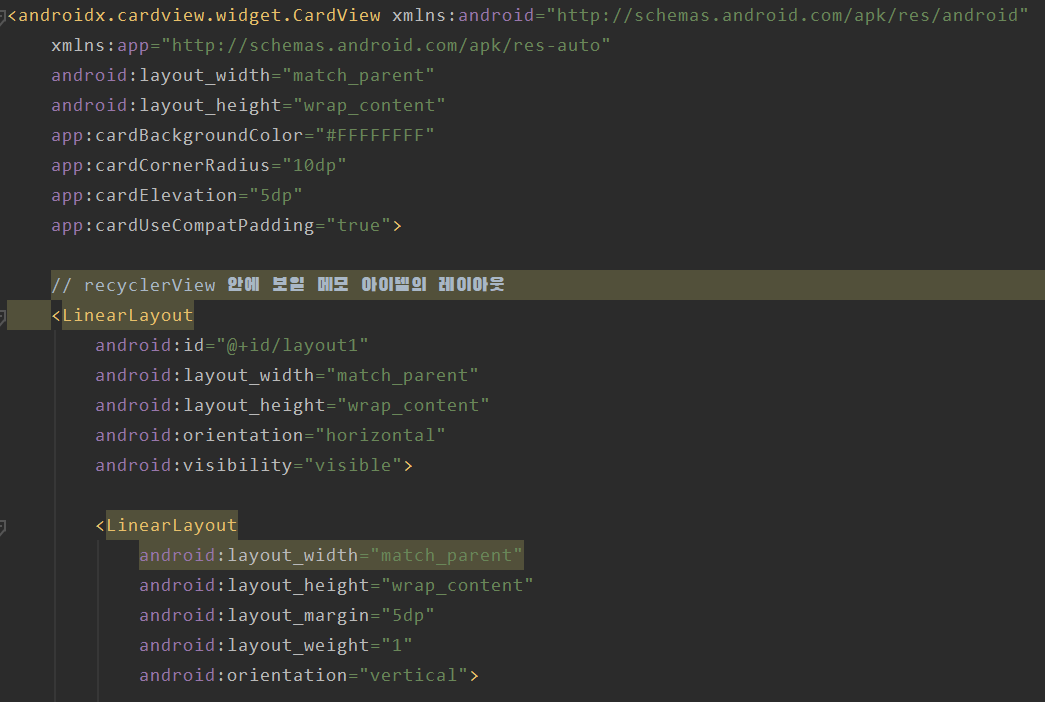
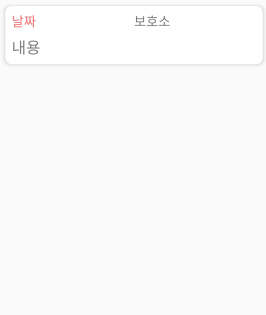
1. Mark 찍기  
     
   - position : mark의 위치에 대한 위도 및 경도 값 지정  
   - title : 사용자가 마커를 눌렀을 때 정보창에 표시되는 문자열  
   - snippet : 제목 아래에 표시되는 추가 텍스트
2. 커스텀 정보창 띄우기  
     
   - 정보창은 보통 1줄만 보여주기 때문에 2줄 이상이 나오게 하기 위해 커스텀 했다. 이는 한 눈에 보기 좋게 만들기 위해 추가했다.
3. 전화번호, E-mail 등에 Link 걸기  
     
   - 정보창의 snippet에 나오는 문자열들 중 전화 번호, E-mail에 Link를 걸어 클릭 시 관련 창이 나오게 만들었다. 해당 링크에 관한 정보를 좀 더 편하게 접하게 하기 위해서 추가했다.

**3) DB를 이용해 봉사 기록 작성, 수정, 삭제**

1. RecyclerView를 통해 작성한 메모들을 보여준다. (fragment\_recruit.xml)

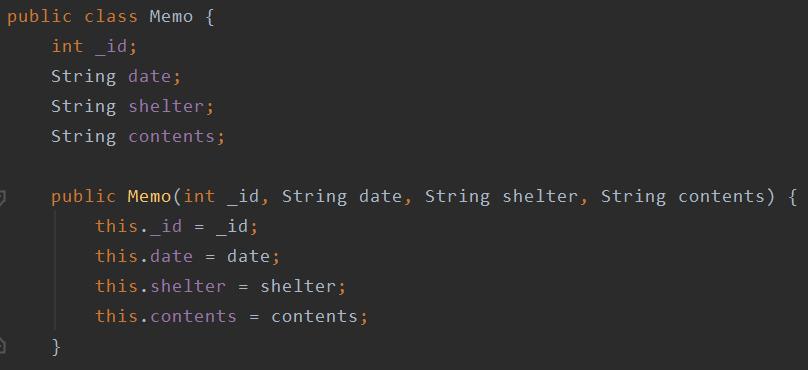
 

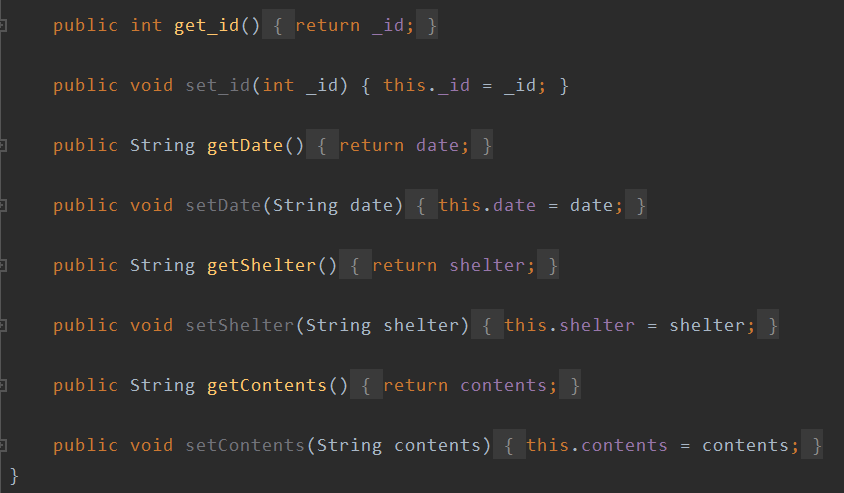
memo\_item.xml

* recyclerView 안에 보일 메모 아이템의 레이아웃
* 날짜, 보호소, 봉사 내용을 표시

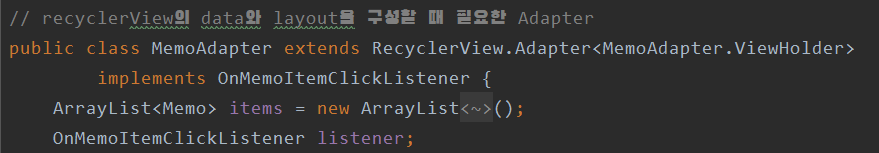
1. Memo.java



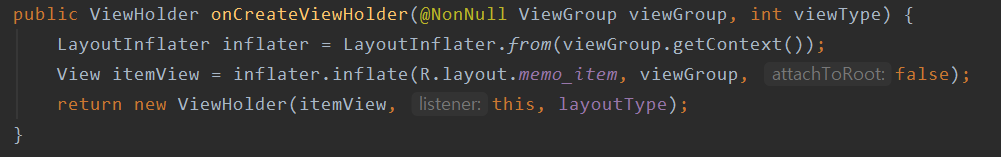


* 메모 안에 넣을 변수 설정

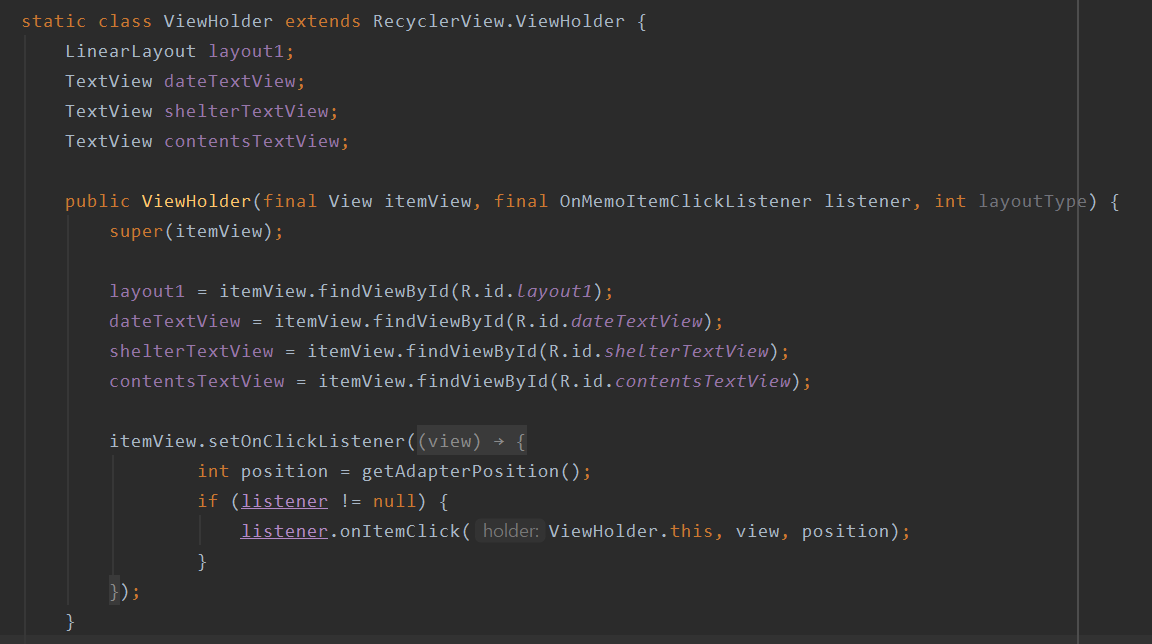
1. MemoAdapter.java



* items : 사용할 아이템 정의

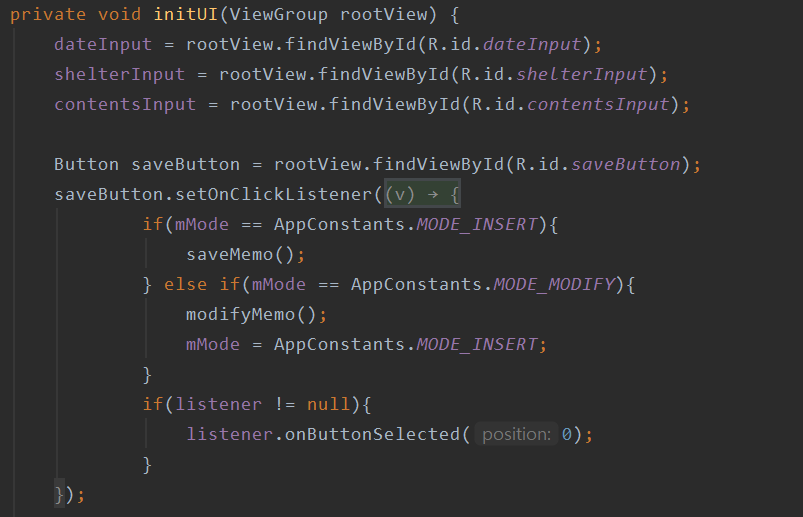


* Adapter는 ViewHolder를 이용해 화면에 아이템을 띄워줌



* 각 아이템을 위한 뷰는 memo\_item.xml에서 정의한 레이아웃을 이용해 뷰 객체로 만들고, 이 객체가 뷰 홀더에 담겨짐
* 뷰 홀더를 정의하는 부분에는 memo\_item.xml의 레이아웃 안에 들어 있는 위젯을 찾아 변수에 할당한 후 Memo 객체가 설정될 때 그 객체에 들어 있는 데이터를 위젯에 설정함

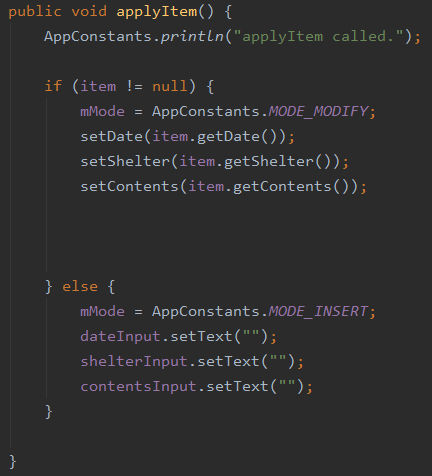
1. MemoFragment.java



* MemoFragment에 대한 View들을 객체로 연결
* saveButton : 저장 버튼을 눌렀을 때 Insert 모드이면 saveMemo()를 호출해 메모를 새로 저장, Modify 모드이면 modifyMemo()를 호출해 메모를 수정



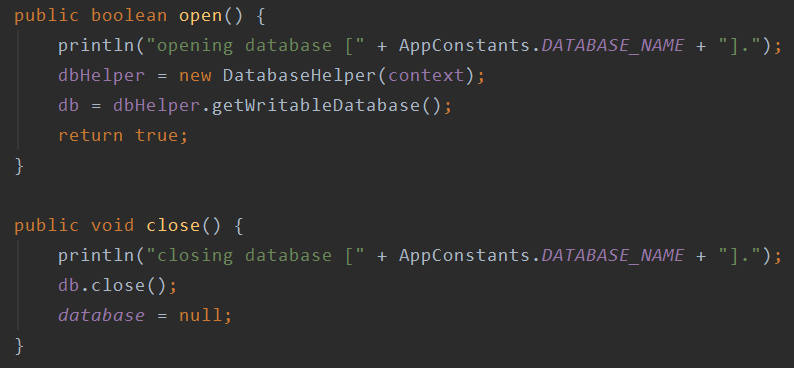
* deleteButton : 삭제 버튼을 눌렀을 때 deleteMemo()를 호출해 메모를 삭제
* closeButton : 닫기 버튼을 눌렀을 때 fragment\_recruit로 이동



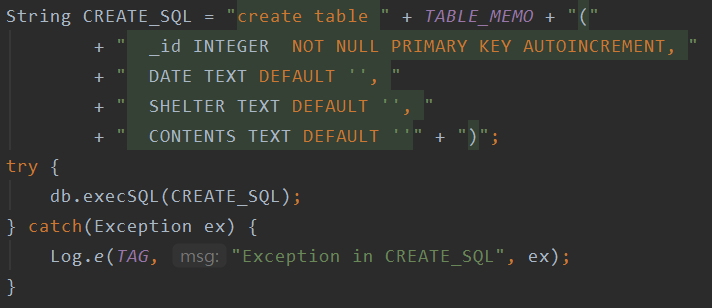
* 이 전의 코드에서 받은 ‘item = null’ 값을 통해 밑의 else 문으로 이동해 해당 메모의 내용들을 다 초기화 시킨다.

1. MemoDatabase.java

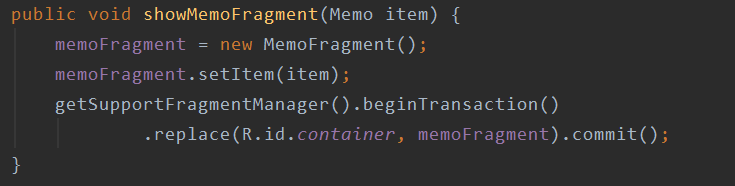
* DB에 접근하기 위해 class 생성



* DB를 열고 닫는 메소드
* DB를 제어하기 위한 dbHelper
* getWirtableDatabase : DB 새로 생성

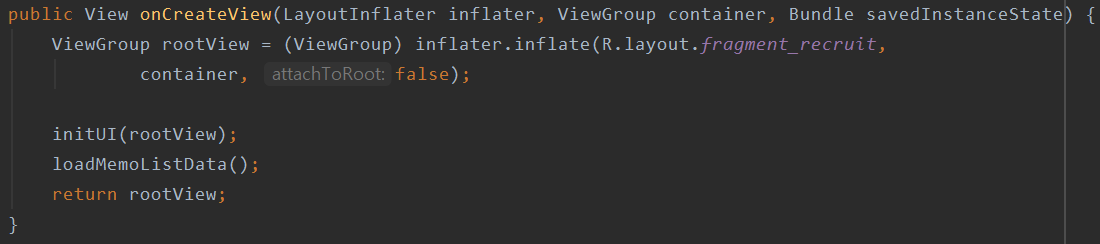


* 테이블 생성
* id, date, shelter, contents 속성

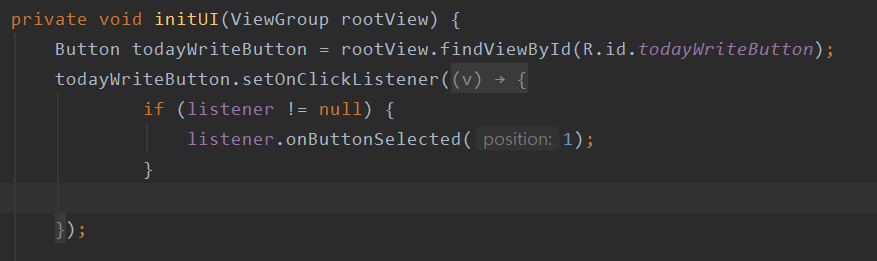
1. MainActivity.java > showMemoFragment   
   

* 메모 아이템을 클릭했을 때 수정할 수 있도록 MemoFragment화면을 띄우는 메소드

1. RecruitFragment.java

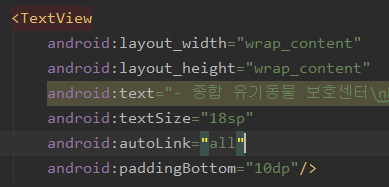


* loadMemoListData : DB에 저장된 리스트를 가져옴



* todayWirteButton : 작성 버튼을 누르면 fragment\_memo로 전환됨

**5) Information fragment**

1. CardView로 배치  
   - CardView 3개를 LinearLayout을 통해 vertical로 쌓아 중요한 내용들을 읽기 수월하도록 만들었다.
2. Link 걸기  
     
   - autoLink를 통해 layout에서 url 형식에 Link가 걸리도록 설정했다.

**5. 향후 발전 계획**

- 사용자가 증가한 경우, 서비스 제공하는 지역을 좀 더 확장할 수 있다.

- 봉사활동에 대해 남긴 후기를 커뮤니티를 통해 다른 사람들과 공유할 수 있다.

- 커뮤니티가 만들어지면 신뢰도가 높은 보호소들에게서 직접 봉사자 모집 지원을 받고 앱을 통해 이를 공고하는 기능을 추가해줄 수 있다. 이 경우 이용자가 보호소에 직접 연락해야 할 일이 줄어든다.

**6. 참고 자료**

<https://mailmail.tistory.com/19>

<https://wonpaper.tistory.com/250>

**7. 유튜브 링크**

<https://youtu.be/qLLG2HRYRas>