

# MainActivity

- 오늘의 공지
  - 키워드 즉시 탐색
    - findNotice(String): void
      - NoticeScraper.find(Context, String, Date): boolean
  - 결과 출력
    - 성공 – AlertDialog
    - 예외 – Toast
- 설정
  - SettingsActivity

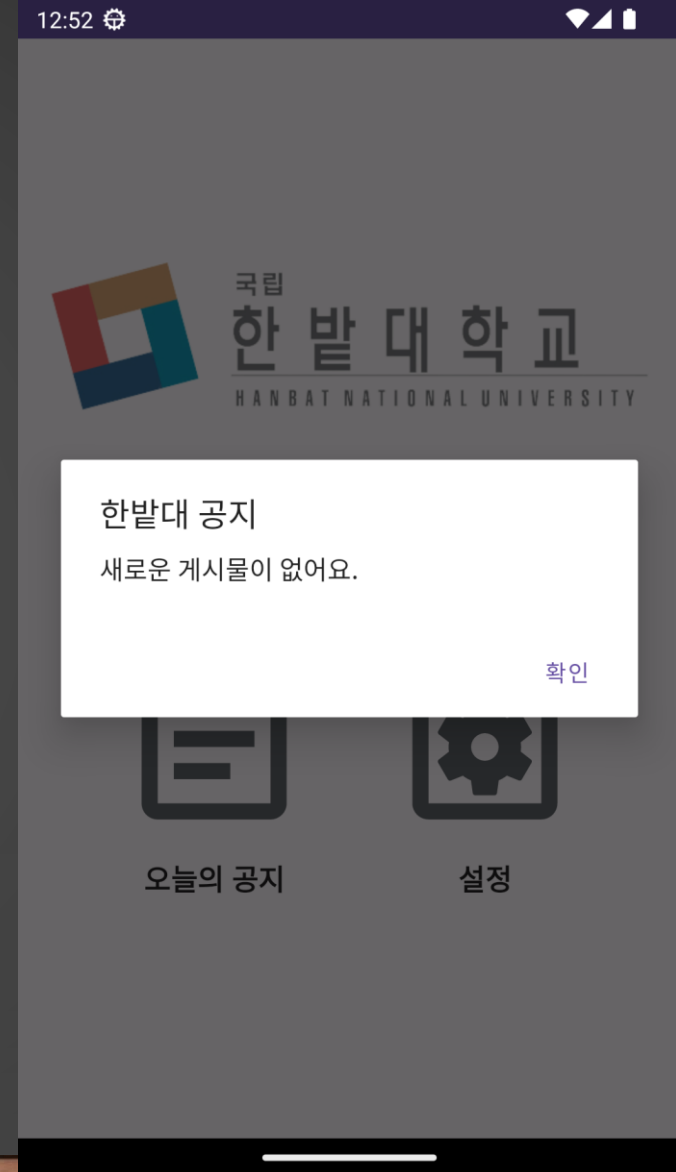


```

private void findNotice(String keyword) {
    binding.buttonFindNotice.setEnabled(false);
    binding.buttonSettings.setEnabled(false);
    binding.progressFindNotice.setVisibility(View.VISIBLE);

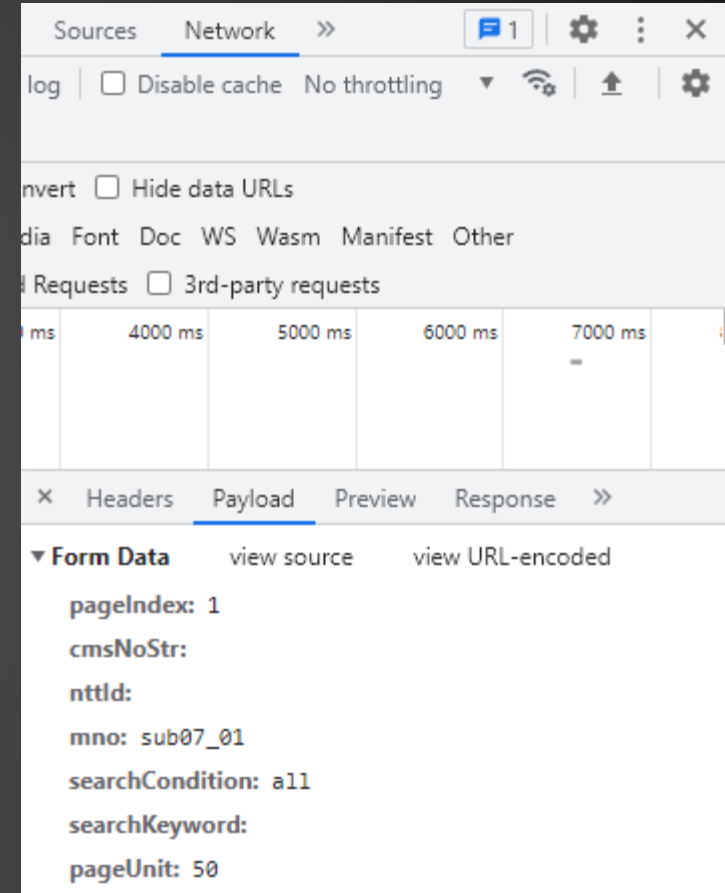
    new Thread(() -> {
        try {
            if (NoticeScraper.find(this, keyword, new Date())) {
                runOnUiThread(() -> showNoticeDialog(true));
            } else {
                runOnUiThread(() -> showNoticeDialog(false));
            }
        } catch (Exception e) {
            runOnUiThread(() -> {
                Toast.makeText(getApplicationContext(), R.string.error_jsoup_exception,
                    Toast.LENGTH_SHORT).show();
            });
        } finally {
            runOnUiThread(() -> {
                binding.buttonFindNotice.setEnabled(true);
                binding.buttonSettings.setEnabled(true);
                binding.progressFindNotice.setVisibility(View.INVISIBLE);
            });
        }
    }).start();
}

```



# NoticeScraper

- 학교 공지사항 스크래핑 – jsoup
  - Document
    - Request URL, Form data, User agent, Referrer
    - HTTP POST 요청
  - Elements
    - CSS selector
  - Return value
    - boolean



```

public static boolean find(Context context, String keyword, Date date) throws Exception {
    Document document = Jsoup.connect(URL).data(getData())
        .userAgent(WebSettings.getDefaultUserAgent(context))
        .referrer(URL).timeout(5000).post();

    Elements elements = document.select("table[class^=board_list]")
        .select("tbody > tr:not(.notice)");
    Elements subjects = elements.select("td.subject > a");
    Elements regDates = elements.select("td.regDate");

    Gson gson = new Gson();
    Type type = new TypeToken<Collection<String>>() {}.getType();
    List<String> keywords = gson.fromJson(keyword, type);

    SimpleDateFormat dateFormat = new SimpleDateFormat("yyyy-MM-dd", Locale.getDefault());

    for (int i = 0; i < elements.size(); ++i) {
        String reg = regDates.get(i).text();
        if (reg.equals(dateFormat.format(date))) {
            for (String subject : keywords) {
                if (subjects.get(i).text().contains(subject)) return true;
            }
        } else {
            Date regDate = dateFormat.parse(reg);
            if (regDate != null && regDate.before(date)) break;
        }
    }

    return false;
}

```

```

private static Map<String, String> getData() {
    Map<String, String> data = new HashMap<>();
    data.put("pageIndex", "1");
    data.put("cmsNoStr", "");
    data.put("nttId", "");
    data.put("mno", "sub07_01");
    data.put("searchCondition", "all");
    data.put("searchKeyword", "");
    data.put("pageUnit", "50");

    return data;
}

```

# SettingsActivity

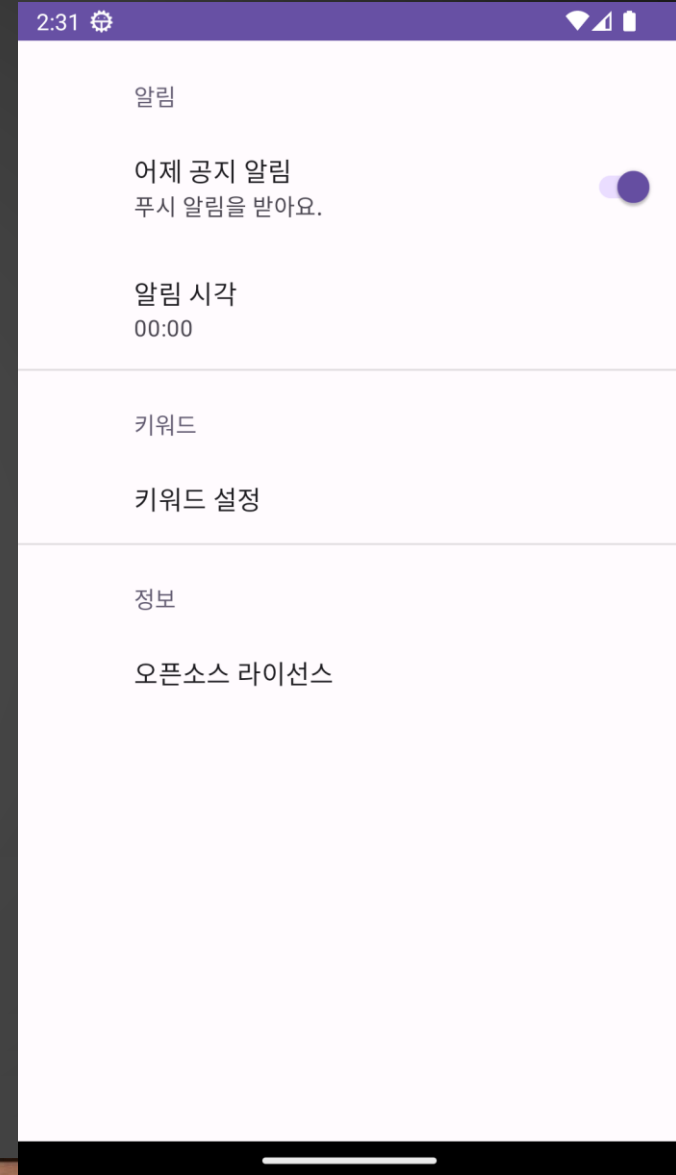
- 환경 설정
  - FragmentContainerView
    - SettingsFragment

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.fragment.app.FragmentContainerView xmlns:andro
xmlns:tools="http://schemas.android.com/tools"
android:id="@+id/fragment_settings"
android:name="kr.ac.hanbat.notice.SettingsFragment"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".SettingsActivity"
tools:viewBindingIgnore="true">
</androidx.fragment.app.FragmentContainerView>
```



# SettingsFragment

- 어제 공지 알림
  - 키워드 푸시 알림 여부
- 알림 시각
  - TimePickerDialog
- 키워드 설정
  - KeywordActivity
- 오픈소스 라이선스
  - OSS licenses plugin



```

pushPreference.setOnPreferenceChangeListener(new Preference.OnPreferenceChangeListener() {
    @Override
    public boolean onPreferenceChange(@NonNull Preference preference, Object newValue) {
        if (getActivity() == null) return false;

        if ((boolean)newValue) {
            String keyword = preferences.getString(getString(R.string.pref_keyword), "");
            if (keyword.equals("")) {
                Toast.makeText(getActivity(), R.string.error_empty_keyword,
                    Toast.LENGTH_SHORT).show();

                return false;
            } else if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.S &&
                !manager.canScheduleExactAlarms()) {
                showExactAlarmDialog();
                return false;
            } else {
                AlarmUtils.set(getActivity());
                return true;
            }
        } else {
            AlarmUtils.cancel(getActivity());
            return true;
        }
    }
});

```

```

timePreference.setOnPreferenceClickListener(new Preference.OnPreferenceClickListener() {
    @Override
    public boolean onPreferenceClick(@NonNull Preference preference) {
        String[] time = timePreference.getSummary().toString().split(":");
        int hourOfDay = Integer.parseInt(time[0]);
        int minute = Integer.parseInt(time[1]);

        TimePickerDialog dialog = new TimePickerDialog(
            getActivity(), android.R.style.Theme_Holo_Light_Dialog_NoActionBar_MinWidth,
            new TimePickerDialog.OnTimeSetListener() {
                @Override
                public void onTimeSet(TimePicker view, int hourOfDay, int minute) {
                    Calendar calendar = Calendar.getInstance();
                    calendar.set(Calendar.HOUR_OF_DAY, hourOfDay);
                    calendar.set(Calendar.MINUTE, minute);

                    String time = new SimpleDateFormat(
                        "HH:mm", Locale.getDefault()).format(calendar.getTime());
                    editor.putString(getString(R.string.pref_time), time);
                    editor.apply();
                    timePreference.setSummary(time);

                    if (getActivity() != null) {
                        AlarmUtils.set(getActivity());
                    }
                }
            }, hourOfDay, minute, false);
        dialog.getWindow().setBackgroundDrawableResource(android.R.color.transparent);
        dialog.show();

        return true;
    }
});

```

# AlarmUtils

- `set(Context): void`
  - BroadcastReceiver 설정
    - Doze mode 고려 (AllowWhileIdle)
      - 안드로이드 6+
      - 10초 이내의 wake lock 유지 가능
    - SCHEDULE\_EXACT\_ALARM 권한
      - 안드로이드 12+
- `cancel(Context): void`
  - AlarmManager, PendingIntent 취소

```
관리자: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator\AppData\Local\Android\Sdk\platform-tools>
    RTC_WAKEUP #23: Alarm{7741e4f type 0 origWhen 1685977200000 whe
        tag=*walarm*:kr.ac.hanbat.notice/.AlarmReceiver
        operation=PendingIntent{dc729dc: PendingIntentRecord{8b60ae5
PS C:\Users\Administrator\AppData\Local\Android\Sdk\platform-tools>
```



```
public static void set(Context context) {
    AlarmManager manager = (AlarmManager)context.getSystemService(Context.ALARM_SERVICE);

    Intent intent = new Intent(context, AlarmReceiver.class);
    PendingIntent pendingIntent = PendingIntent.getBroadcast(
        context, REQUEST_CODE, intent, PendingIntent.FLAG_IMMUTABLE);

    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.S &&
        !manager.canScheduleExactAlarms()) {

        AlarmManagerCompat.setAndAllowWhileIdle(
            manager, AlarmManager.RTC_WAKEUP,
            getCalendar(context).getTimeInMillis(), pendingIntent);
    } else {
        AlarmManagerCompat.setExactAndAllowWhileIdle(
            manager, AlarmManager.RTC_WAKEUP,
            getCalendar(context).getTimeInMillis(), pendingIntent);
    }
}
```

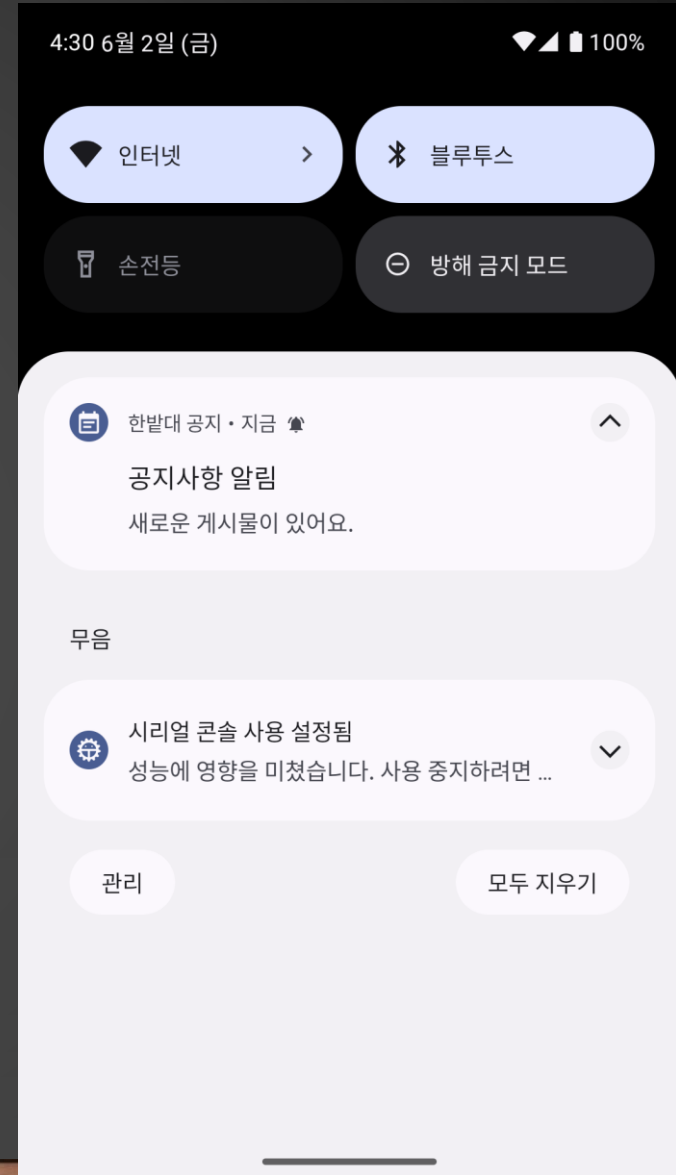
```
public static void cancel(Context context) {
    AlarmManager manager = (AlarmManager)context.getSystemService(Context.ALARM_SERVICE);

    Intent intent = new Intent(context, AlarmReceiver.class);
    PendingIntent pendingIntent = PendingIntent.getBroadcast(
        context, REQUEST_CODE, intent,
        PendingIntent.FLAG_IMMUTABLE | PendingIntent.FLAG_NO_CREATE);

    if (pendingIntent != null) {
        manager.cancel(pendingIntent);
        pendingIntent.cancel();
    }
}
```

# AlarmReceiver

- 알람 재설정 – onReceive
- 키워드 탐색
  - findNotice(Context, String): void
    - Network I/O – PendingResult
    - 10초 이내에 반환 (ANR)
- 푸시 알림
  - NotificationChannel 생성
    - 안드로이드 8+
  - POST\_NOTIFICATIONS 권한
    - 안드로이드 13+



```

private void findNotice(Context context, String keyword) {
    Calendar calendar = Calendar.getInstance();
    calendar.add(Calendar.DAY_OF_MONTH, -1);

    PendingResult pendingResult = goAsync();
    new Thread(() -> {
        try {
            if (NoticeScraper.find(context, keyword, calendar.getTime())) {
                showNotification(context);
            }
        } catch (Exception ignored) {}
        finally {
            pendingResult.finish();
        }
    }).start();
}

```

```

private void showNotification(Context context) {
    Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse(URL));
    PendingIntent pendingIntent = PendingIntent.getActivity(
        context, REQUEST_CODE, intent,
        PendingIntent.FLAG_IMMUTABLE | PendingIntent.FLAG_ONE_SHOT);

    NotificationCompat.Builder builder = new NotificationCompat.Builder(
        context, context.getString(R.string.notice_channel_id));
    builder.setSmallIcon(R.drawable.ic_stat_notify)
        .setContentTitle(context.getString(R.string.notice_channel_name))
        .setContentText(context.getText(R.string.msg_notice_found))
        .setPriority(NotificationCompat.PRIORITY_DEFAULT)
        .setVisibility(NotificationCompat.VISIBILITY_PUBLIC)
        .setContentIntent(pendingIntent)
        .setAutoCancel(true);

    NotificationManagerCompat manager = NotificationManagerCompat.from(context);
    manager.notify(NOTIFICATION_ID, builder.build());
}

```

# BootReceiver

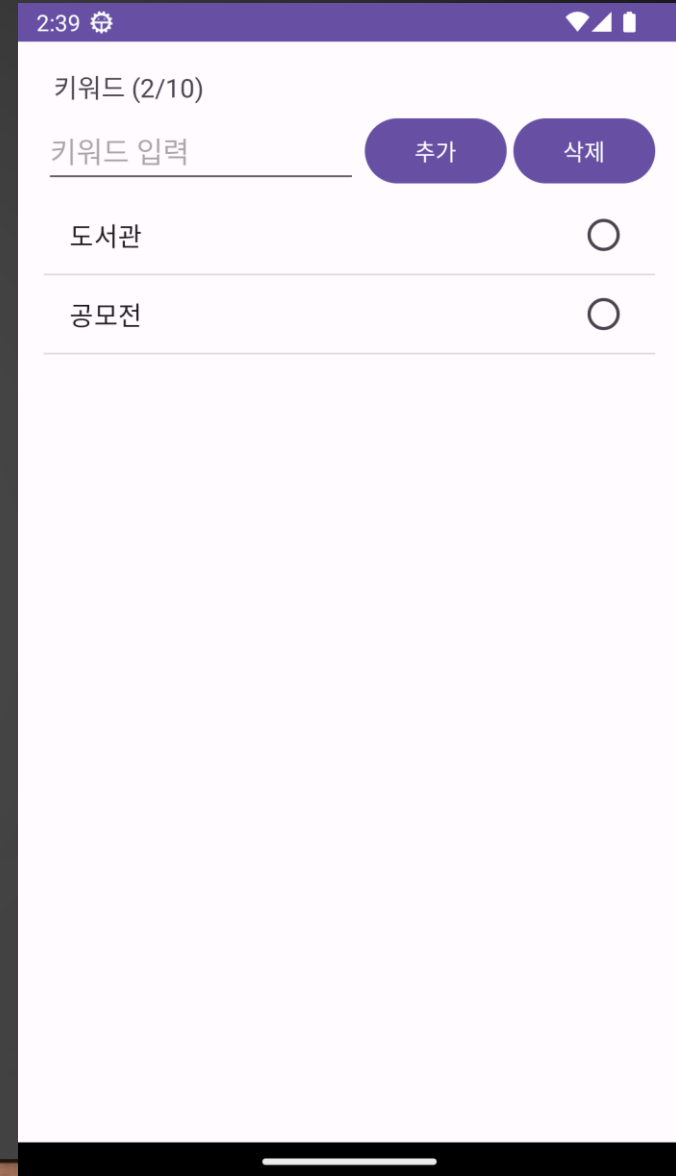
- 기기를 종료하면 모든 알람이 취소
  - 재부팅 시 자동으로 알람 재설정
- `RECEIVE_BOOT_COMPLETED` 권한
  - Intent filter: `ACTION_BOOT_COMPLETED`

```
public class BootReceiver extends BroadcastReceiver {  
    @Override  
    public void onReceive(Context context, Intent intent) {  
        if (!intent.getAction().equals("android.intent.action.BOOT_COMPLETED")) return;  
  
        SharedPreferences preferences = PreferenceManager.getDefaultSharedPreferences(context);  
        if (preferences.getBoolean(context.getString(R.string.pref_push), false)) {  
            AlarmUtils.set(context);  
        }  
    }  
}
```



# KeywordActivity

- 키워드 저장 알고리즘
  - ArrayList to JSON String
    - 직렬화/역직렬화 – Gson
  - String to SharedPreferences
- 결과 출력
  - ListView



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

    binding = ActivityKeywordBinding.inflate(getLayoutInflater());
    setContentView(binding.getRoot());

    preferences = PreferenceManager.getDefaultSharedPreferences(this);
    String keyword = preferences.getString(getString(R.string.pref_keyword), "");

    Gson gson = new Gson();
    Type type = new TypeToken<Collection<String>>() {}.getType();
    List<String> keywords = gson.fromJson(keyword, type);

    ArrayAdapter<String> adapter = new ArrayAdapter<>(
        this, android.R.layout.simple_list_item_single_choice, keywords);
    binding.listKeywords.setAdapter(adapter);
    updateKeywordViews(keywords.size());
}
```

```
private void setKeywordPreference(List<String> keywords) {
    Gson gson = new Gson();
    String keyword = gson.toJson(keywords);

    SharedPreferences.Editor editor = preferences.edit();
    editor.putString(getString(R.string.pref_keyword), keyword);
    editor.apply();
}
```