



# PreferenceActivity(실습)

---

배 희호 교수  
경북대학교  
스마트IT과



# PreferenceActivity

- Android 프로그래밍에서 **설정 Activity**를 만들어야 할 때가 있음
- 설정 Activity들은 보통 List로 구성되어 있음
- 리스트를 컨버팅하면서 만든다고 해도, 텍스트만 나오는 row가 있고, 체크 박스도 혼재된 row, 입력창이 들어가야하는 row가 있을 수 있음
  - 이럴 때 편하게 사용할 수 있는 방법이 바로 PreferenceActivity를 상속받아서 만드는 것



# PreferenceActivity



- PreferenceActivity를 사용하기 위한 2단계
  - Layout 작성
  - Activity 작성
- Layout 작성
  - UI에 보여줄 Preference들을 XML에 정의해야 함
  - PreferenceActivity는 Activity가 들어있지만 res/layout에 XML을 저장하지 않고, res/xml 폴더에 파일이 들어 있음
  - PreferenceActivity에 사용되는 Layout xml은 일반 Activity에 사용하는 XML과 작성법이 조금 다름
  - 일반 Activity에 사용하는 default layout이 LinearLayout 이라면, Preference Activity에 사용하는 default layout은 PreferenceScreen



# PreferenceActivity

## ■ Activity 작성

- PreferenceActivity를 처음 작성할 때 달라진 부분은 하나 뿐임
- 보통의 Activity에서 Layout을 설정할 때 setContentView(R.layout.파일이름)으로 설정한다면,
- PreferenceActivity는 Layout을 설정할 때 addPreferencesFromResource(R.xml.파일이름)과 같이 함
- 일반적인 Activity라면 Click Event를 OnClickListener로 받음
- PreferenceActivity는 Click Event를 OnPreferenceClickListener로 받아야 함



# PreferenceActivity



- XML에 Preference 정의하는 절차
  - 먼저 "res" 디렉토리에 "xml" 이름으로 디렉토리를 생성
  - res/xml 안에 원하는 preference xml 파일을 생성
    - 일반적으로 preference.xml과 같이 생성
    - 생성한 XML 파일을 열어보면 아래와 같이 <PreferenceScreen>을 root로 하여 만들어진 것을 확인할 수 있음

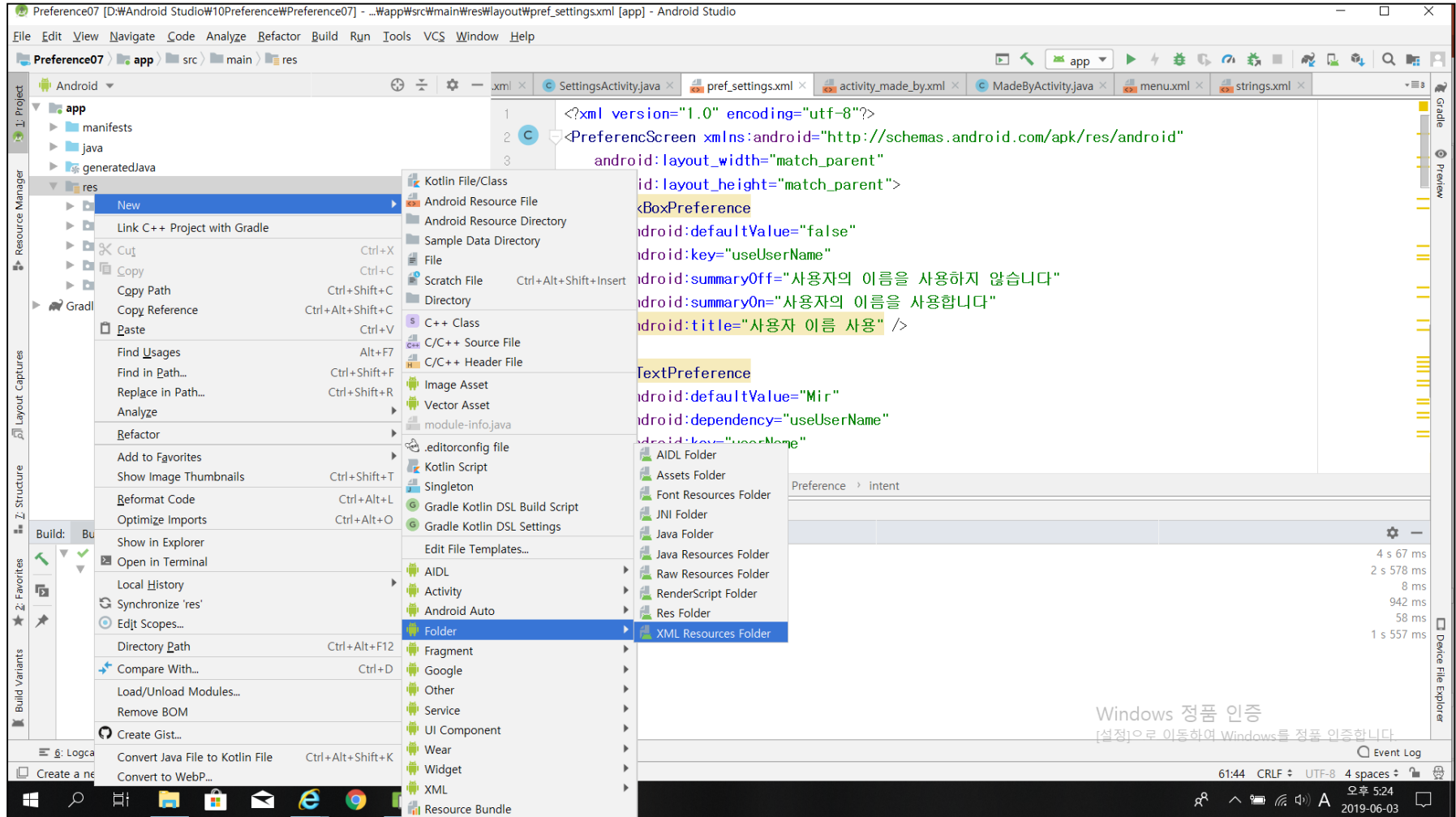
```
<?xml version="1.0" encoding="utf-8"?>  
<PreferenceScreen xmlns:android="http://schemas.android.com/apk/res/android">  
  
</PreferenceScreen>
```

- 생성한 xml 파일을 열어 원하는 Preference를 추가



# PreferenceActivity

## ■ [New]–[Folder]–[XML Resources Folder]





# PreferenceActivity

## [New] - [XML resource file]

The screenshot shows the Android Studio interface. The 'New' menu is open, and 'XML resource file' is selected. The XML editor displays the following code:

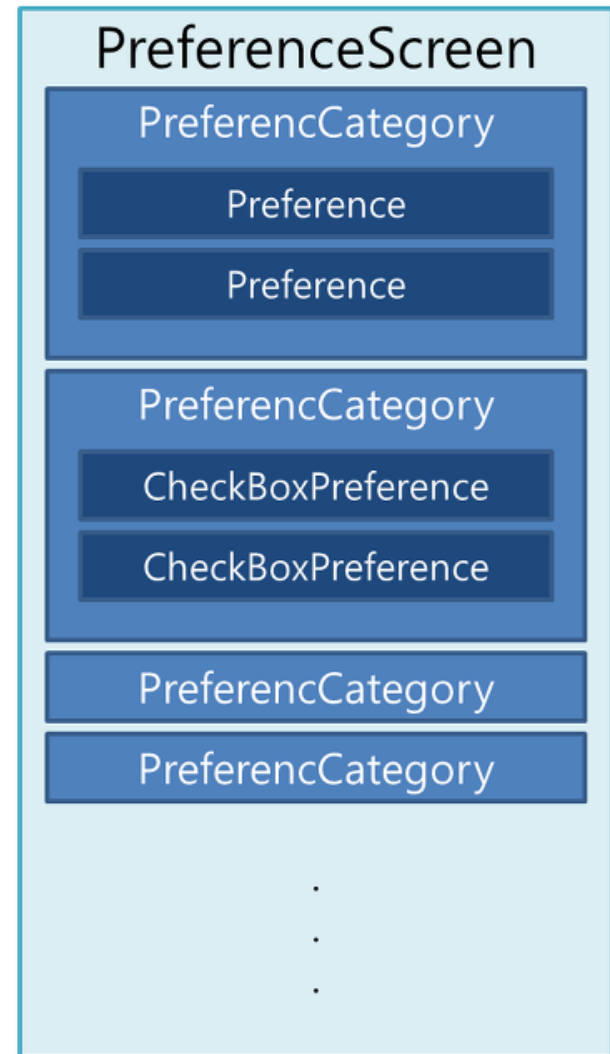
```
<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen xmlns:android="http://schemas.android.com/apk/res/android">
    <include layout="@layout/activity_made_by" />
    <Preference
        android:defaultValue="false"
        android:key="useUserName"
        android:summaryOff="사용자의 이름을 사용하지 않습니다"
        android:summaryOn="사용자의 이름을 사용합니다"
        android:title="사용자 이름 사용" />
    <include layout="@layout/activity_made_by" />
    <Preference
        android:defaultValue="Mir"
        android:dependency="useUserName"
        android:key="userName"
        android:maxLines="1"
        android:title="사용자 이름 사용" />
</PreferenceScreen>
```

The bottom of the screen shows the Windows taskbar with the time 6:44 and date 2019-06-03.



# PreferenceActivity

- PreferenceScreen 태그 안에는 복수의 PreferenceCategory가 들어갈 수 있음
- 또 PreferenceCategory 안에는 복수의 Preference들이 들어갈 수 있음







# PreferenceActivity

## ■ 태그 종류

태그 종류	설명
PreferenceScreen	설정화면 단위, 중첩 가능하며 중첩된 내용은 별도의 화면에 나옴
PreferenceCategory	설정 여러 개를 시각적으로 묶어서 표현
CheckboxPreference	체크박스가 나오는 설정
EditTextPreference	글 입력을 위한 설정
ListPreference	항목 다이얼로그를 위한 설정
MultiSelectListPreference	항목 다이얼로그인데 체크박스가 자동 추가되어 여러 선택 가능
RingtonePreference	알림음 선택을 위한 설정
SwitchPreference	스위치를 이용한 설정



# PreferenceActivity



## ■ 중요 XML 속성

XML 속성	설명
android:key	Preferences 데이터 저장/불러올 때 사용되는 키 (key) 지정
android:title	Preference 항목의 제목을 지정
android:summary	Preference 항목을 자세히 설명하는 문자열 지정
android:enabled	Preference 항목의 활성화 비활성화 여부 지정 (true/false)
android:selectable	Preference 항목의 선택 가능 여부 결정 (true/false)
android:order	Preference 항목이 표시되는 순서를 결정 (0-based 정수 사용: 낮은 값 먼저 보임) 순서가 명시적으로 지정되지 않는다면 XML에 정의된 순서대로 표시됨



# PreferenceActivity



## ■ 중요 메소드

중요 메소드	설명
void setKey/setTitle/setSummary	Preference 항목의 키, 제목, 설명을 지정
void setEnabled(boolean)	Preference 항목의 활성화/비활성화 여부 지정
void setSelectable(boolean)	Preference 항목의 선택 가능 여부 지정
void setLayoutResource()	Preference 항목에 표시할 view를 'R.layout.레이아웃파일이름' 형식으로 파라미터로 넘김
void setOnPreferenceChangeListener()	Preference 항목에 변화가 있으면 호출할 callback 메소드 지정
void setOnPreferenceClickListener()	Preference 항목에 click 이벤트가 발생하면 호출할 callback 메소드 지정



# PreferenceActivity

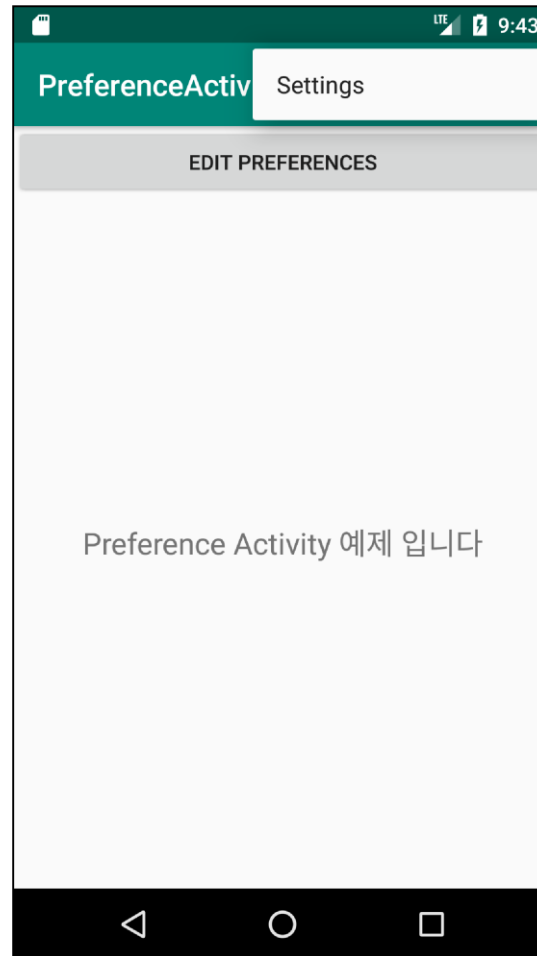


- 이 Preference의 태그의 속성들은 간단하게는 "키(KEY)"와 "타이틀(TITLE)" / "키(KEY)"와 "타이틀(TITLE)", 그리고 "값(VALUE)"로 이루어져 있음
- 다른 속성들이 많이 있지만, 이것들만 사용해도 어렵지 않게 설정의 기능들을 구현해 낼 수 있음



# PreferenceActivity 예제 1

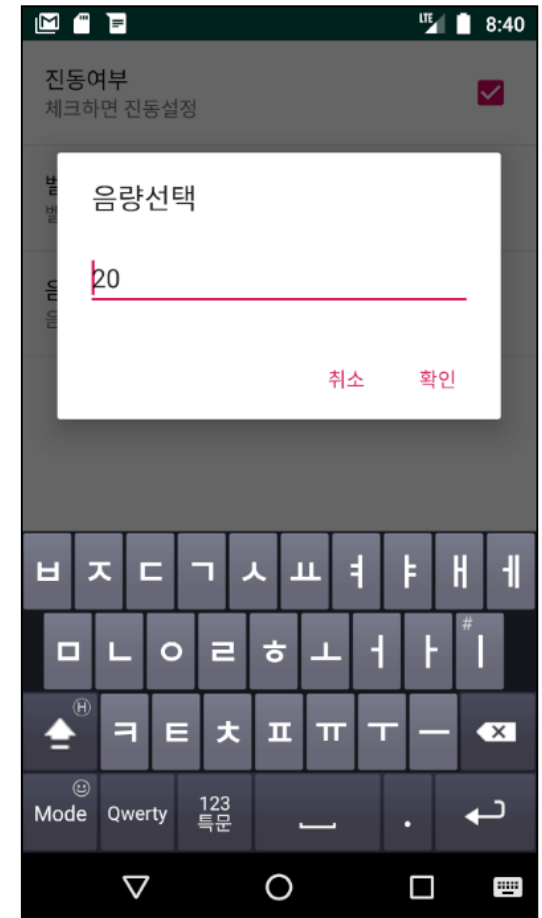
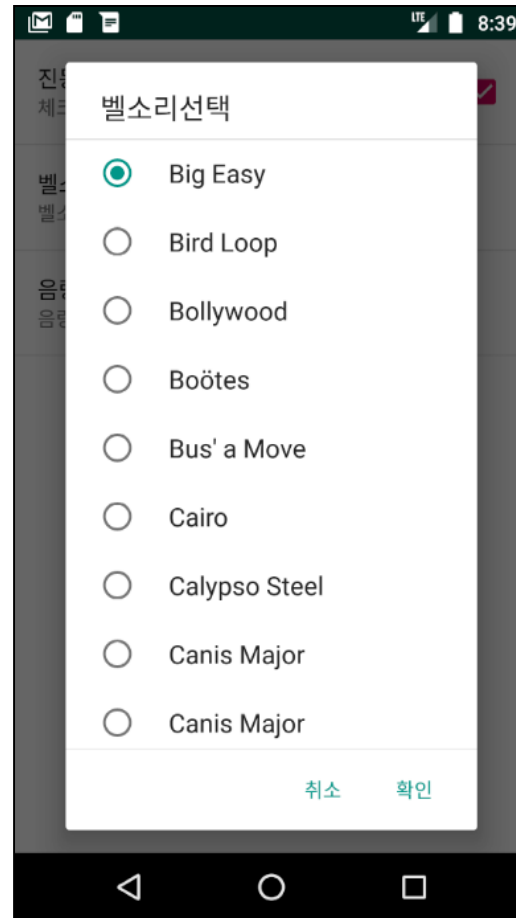
- 다음과 같이 애플리케이션의 옵션 메뉴를 활용하여 환경을 설정하는 프로그램을 만들어보자





# PreferenceActivity 예제 1

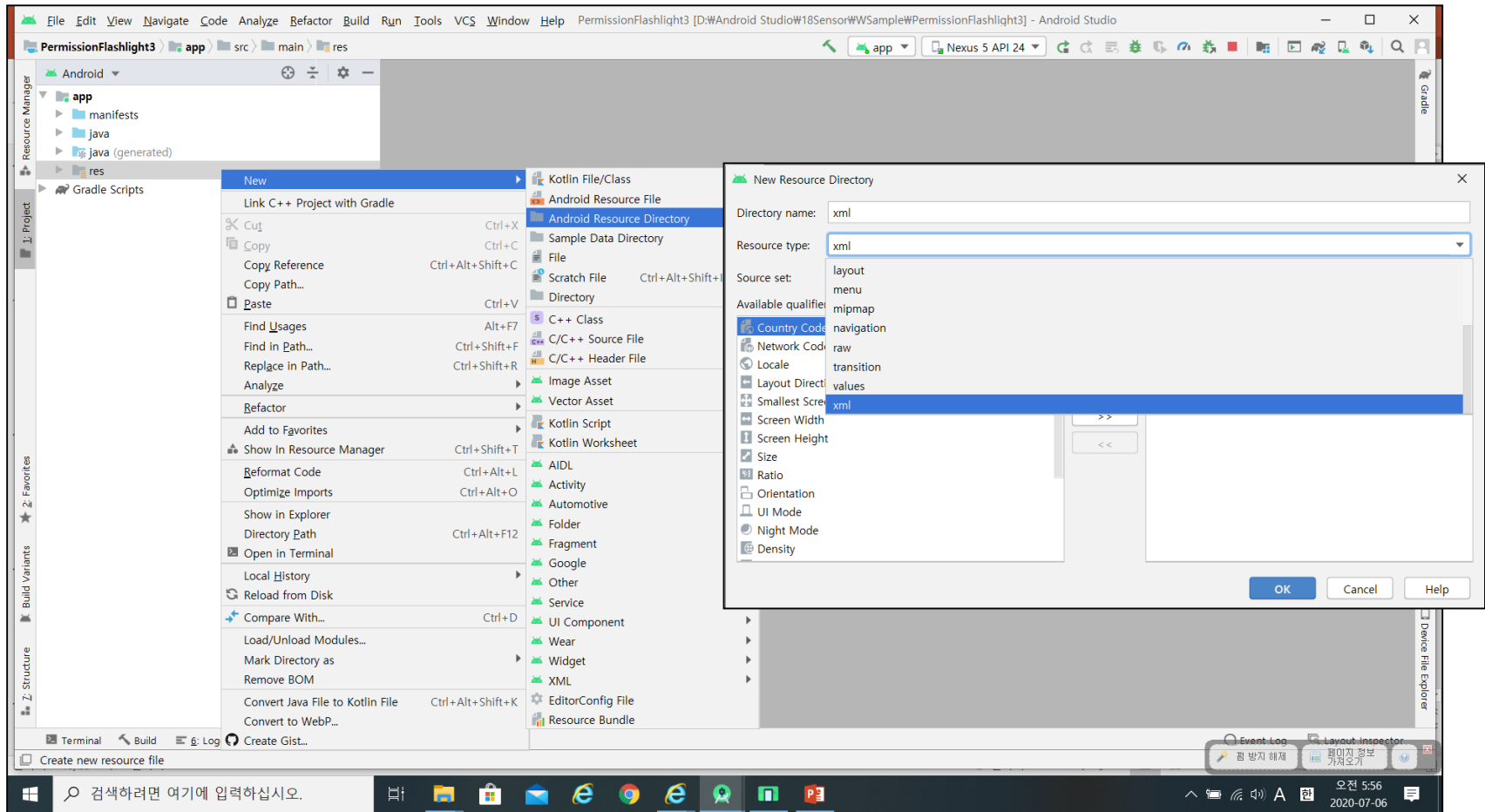
## ■ 실행 화면





# PreferenceActivity 예제 1

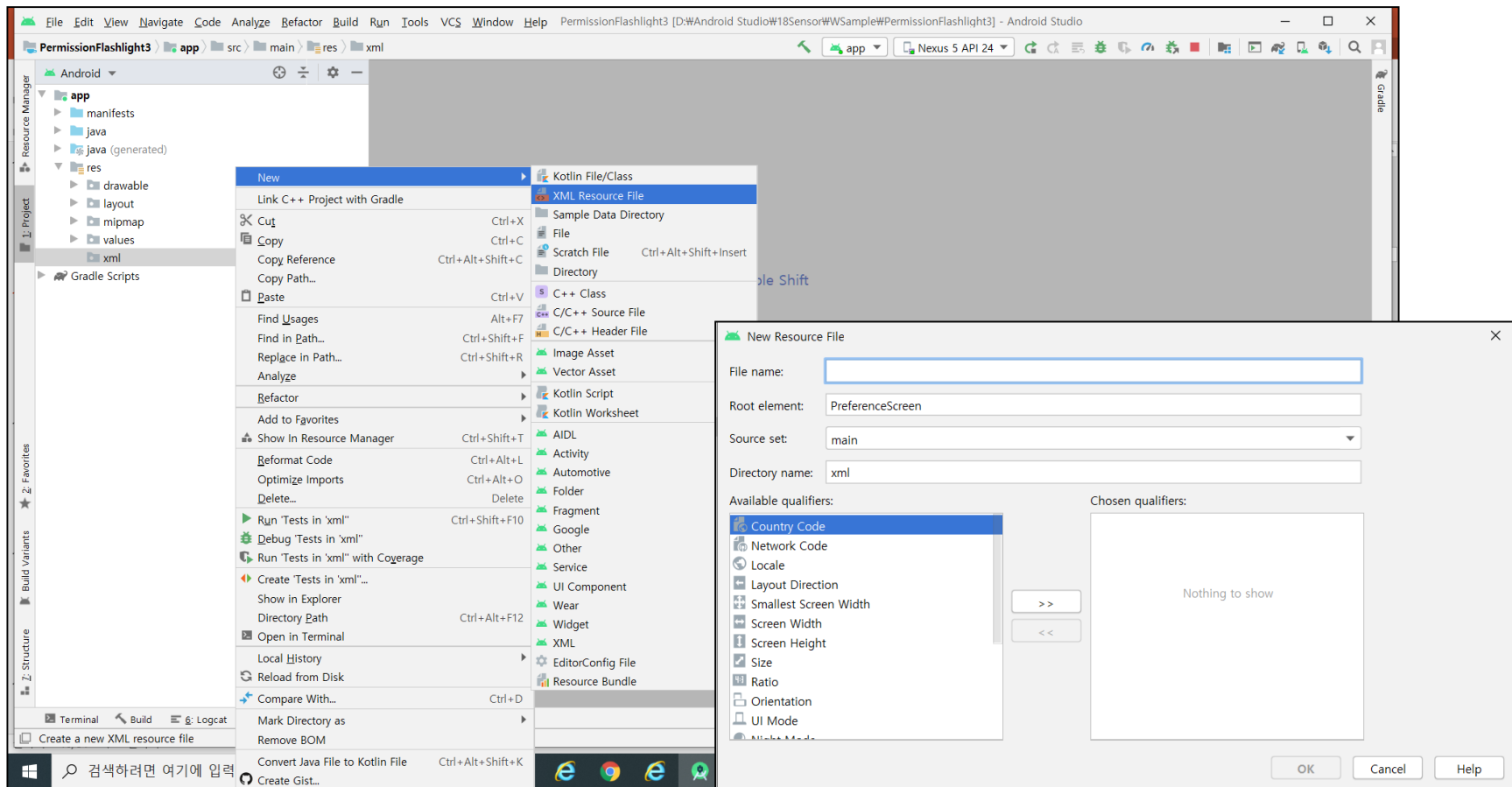
- PreferenceActivity를 만들기 위한 setting 방법
- res 폴더에 xml이라는 이름의 폴더 생성





# PreferenceActivity 예제 1

- PreferenceActivity를 만들기 위한 setting 방법
  - 생성한 xml폴더안에 자신이 원하는 이름으로 PreferenceActivity를 디자인 할 xml파일을 생성







# PreferenceActivity 예제 1

## ■ 기본 속성

- android:title – Preference의 텍스트 지정
- android:summary – title아래의 작은 설명
- android:key – SharedPreferences로 저장할 때 키
- android:defaultValue – 기본 값



# PreferenceActivity 예제 1

## ■ 사용자 인터페이스

```
<?xml version="1.0" encoding="utf-8"?>
<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"
    android:orientation="vertical"
    tools:context=".MainActivity">
    <Button
        android:id="@+id/button"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Edit Preferences" />
    <TextView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center"
        android:text="Preference Activity 예제 입니다"
        android:textSize="20dp" />
</LinearLayout>
```



# PreferenceActivity 예제 1

## ■ pref\_activity.xml (R.xml 폴더)

```
<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen xmlns:android="http://schemas.android.com/apk/res/android">

    <CheckBoxPreference
        android:defaultValue="false"
        android:key="check"
        android:summary="체크하면 진동설정"
        android:title="진동여부" />

    <RingtonePreference
        android:key="ringtone"
        android:showDefault="true"
        android:showSilent="true"
        android:summary="벨소리를 선택하시오"
        android:title="벨소리선택" />

</PreferenceScreen>
```





# PreferenceActivity 예제 1

## ■ pref\_activity.xml (R.xml 폴더)

```
<EditTextPreference  
    android:defaultValue="20"  
    android:key="edit"  
    android:summary="음량을 선택하십시오"  
    android:title="음량선택" />  
</PreferenceScreen>
```





# PreferenceActivity 예제 1

## ■ MainActivity.JAVA

```
public class MainActivity extends AppCompatActivity {

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

        Button button = findViewById(R.id.button);
        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(MainActivity.this, PreferenceActivity.class);
                startActivity(intent);
            }
        });
    }
}
```



# PreferenceActivity 예제 1

## ■ MainActivity.JAVA

@Override

```
public boolean onCreateOptionsMenu(Menu menu) {  
    menu.add(0,1,0, "Settings");  
    return true;  
}
```

@Override

```
public boolean onOptionsItemSelected(MenuItem item) {  
    if (item.getItemId() == 1) {  
        Intent intent = new Intent(this, PreferenceActivity.class);  
        startActivity(intent);  
        return true;  
    }  
    return false;  
}
```



# PreferenceActivity 예제 1

## ■ MainActivity.JAVA

@Override

```
public boolean onOptionsItemSelected(MenuItem item) {  
    if (item.getItemId() == 1) {  
        Intent intent = new Intent(this, PreferenceActivity.class);  
        startActivity(intent);  
        return true;  
    }  
    return false;  
}
```



# PreferenceActivity 예제 1

## ■ PreferenceActivity.JAVA

```
public class PreferenceActivity extends android.preference.PreferenceActivity
    implements Preference.OnPreferenceClickListener {
    CheckBoxPreference vibrator;
```

@Override

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    addPreferencesFromResource(R.xml.pref_activity);
```

```
    vibrator = (CheckBoxPreference) findPreference("check");
    vibrator.setOnPreferenceClickListener(this);
    RingtonePreference ringtone =
        (RingtonePreference) findPreference("ringtone");
    ringtone.setOnPreferenceClickListener(this);
```





# PreferenceActivity 예제 1

## ■ PreferenceActivity.JAVA

```
final EditTextPreference volume =  
    (EditTextPreference) findPreference("edit");  
volume.setOnPreferenceChangeListener(  
    new Preference.OnPreferenceChangeListener() {  
        @Override  
        public boolean onPreferenceChange(Preference preference,  
                                           Object newValue) {  
            String msg = String.format("Volume을 %s에서 %s로 조정하였음",  
                                       volume.getText().toString(), newValue);  
            Toast.makeText(getApplicationContext(), msg,  
                          Toast.LENGTH_SHORT).show();  
            return true;  
        }  
    });  
}
```



# PreferenceActivity 예제 1

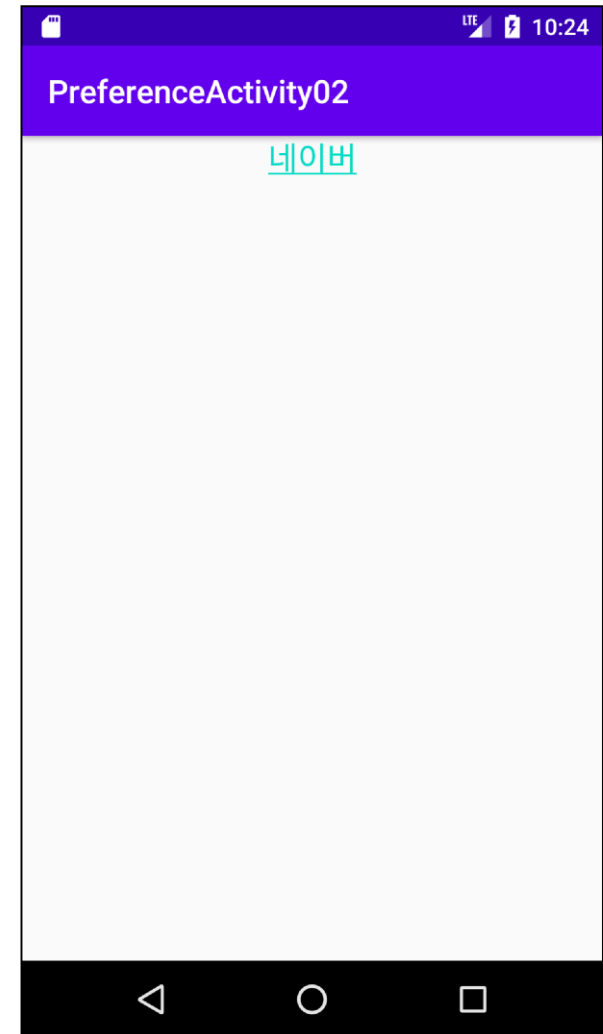
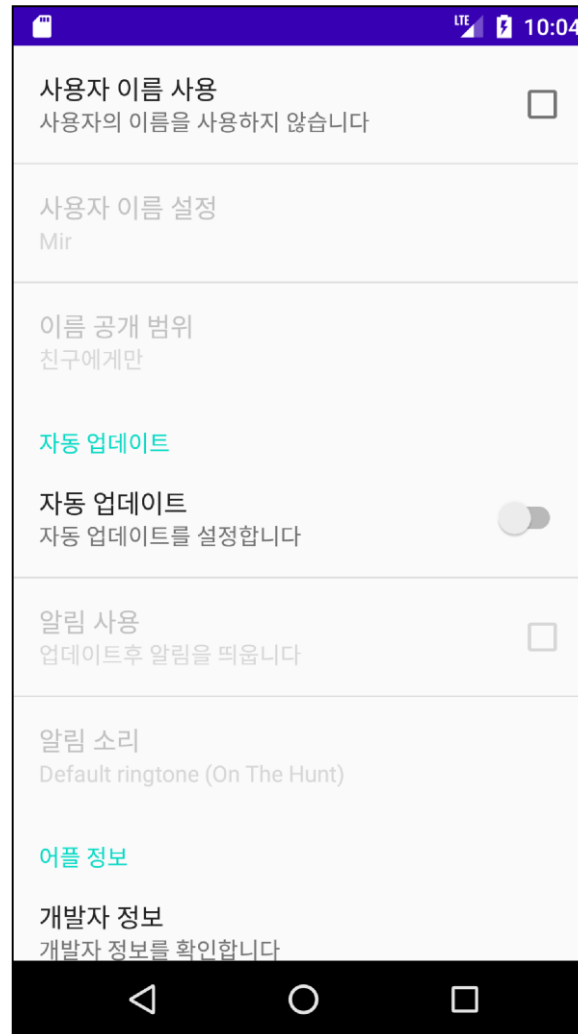
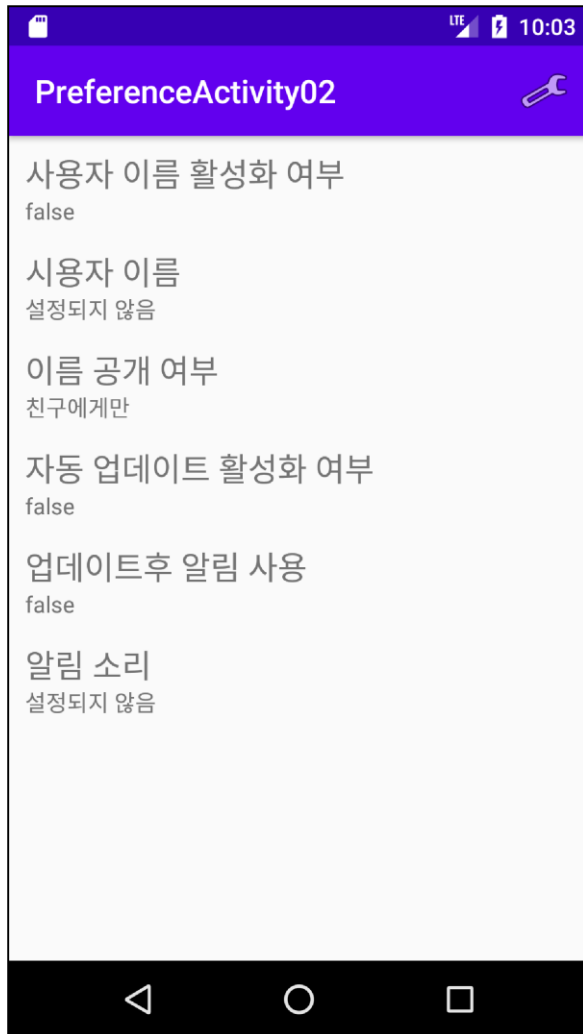
## ■ PreferenceActivity.JAVA

@Override

```
public boolean onPreferenceClick(Preference preference) {  
    if (preference.getKey().equals("check")) {  
        if (vibrator.isChecked()) {  
            Toast.makeText(getApplicationContext(), "진동을 선택 했음",  
                                Toast.LENGTH_SHORT).show();  
        } else {  
            Toast.makeText(getApplicationContext(), "진동을 해제 했음",  
                                Toast.LENGTH_SHORT).show();  
        }  
    } else if (preference.getKey().equals("ringtone")) {  
        Toast.makeText(getApplicationContext(), "ringtone 설정",  
                                Toast.LENGTH_SHORT).show();  
    }  
    return true;  
}
```



# PreferenceActivity 예제 2





# PreferenceActivity 예제 2

## ■ 사용자 인터페이스

```
<?xml version="1.0" encoding="utf-8"?>
<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"
    android:orientation="vertical"
    android:padding="10dp"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="20dp"
        android:text="사용자 이름 활성화 여부" />

    <TextView
        android:id="@+id/userNameEnable"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
```



# PreferenceActivity 예제 2

## ■ 사용자 인터페이스

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="15dp"
    android:textSize="20dp"
    android:text="사용자 이름" />

<TextView
    android:id="@+id/userName"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="15dp"
    android:textSize="20dp"
    android:text="이름 공개 여부" />
```



# PreferenceActivity 예제 2

## ■ 사용자 인터페이스

```
<TextView
    android:id="@+id/userNameOpen"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="15dp"
    android:textSize="20dp"
    android:text="자동 업데이트 활성화 여부" />

<TextView
    android:id="@+id/autoUpdateEnable"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
```



# PreferenceActivity 예제 2

## ■ 사용자 인터페이스

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="15dp"
    android:textSize="20dp"
    android:text="업데이트후 알림 사용" />

<TextView
    android:id="@+id/updateNofiti"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="15dp"
    android:textSize="20dp"
    android:text="알림 소리" />
```



# PreferenceActivity 예제 2

## ■ 사용자 인터페이스

```
<TextView
    android:id="@+id/notifiSound"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
</LinearLayout>
```





# PreferenceActivity 예제 2

## ■ MainActivity.JAVA

```
public class MainActivity extends AppCompatActivity {
    SharedPreferences preferences;
    TextView userNameEnable, userName, userNameOpen, autoUpdateEnable,
        updateNofiti, notifiSound;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        preferences = PreferenceManager.getDefaultSharedPreferences(this);

        userNameEnable = findViewById(R.id.userNameEnable);
        userName = findViewById(R.id.userName);
        userNameOpen = findViewById(R.id.userNameOpen);
        autoUpdateEnable = findViewById(R.id.autoUpdateEnable);
        updateNofiti = findViewById(R.id.updateNofiti);
        notifiSound = findViewById(R.id.notifiSound);
        getPreferencesData();
    }
```



# PreferenceActivity 예제 2

## ■ MainActivity.JAVA

@Override

```
protected void onResume() {  
    super.onResume();  
    getPreferencesData();  
}
```

```
private void getPreferencesData() {  
    userNameEnable.setText("" + preferences.getBoolean("useUserName",  
                                                         false));  
    userName.setText(preferences.getString("userName", "설정되지 않음"));  
    autoUpdateEnable.setText("" + preferences.getBoolean("autoUpdate",  
                                                         false));  
    updateNofiti.setText("" + preferences.getBoolean("useUpdateNofiti",  
                                                      false));  
    String[] array = getResources().getStringArray(R.array.userNameOpen);  
    int index = getArrayIndex(R.array.userNameOpen_values,  
                              preferences.getString("userNameOpen", "0"));
```



# PreferenceActivity 예제 2

## ■ MainActivity.JAVA

```
if (index != -1) {
    String userNameOpenString = array[index];
    userNameOpen.setText(userNameOpenString);
} else
    userNameOpen.setText("");
String Sound = preferences.getString("autoUpdate_ringtones", "설정되지 않음");
if (TextUtils.isEmpty(Sound)) {
    notiSound.setText("무음으로 설정됨");
} else {
    Ringtone ringtone = RingtoneManager.getRingtone(getBaseContext(),
        Uri.parse(Sound));
    if (ringtone == null) {
        notiSound.setText(null);
    } else {
        String name = ringtone.getTitle(this);
        notiSound.setText(name);
    }
}
}
```



# PreferenceActivity 예제 2

## ■ MainActivity.JAVA

```
private int getArrayIndex(int array, String findIndex) {  
    String[] arrayString = getResources().getStringArray(array);  
    for (int e = 0; e < arrayString.length; e++) {  
        if (arrayString[e].equals(findIndex))  
            return e;  
    }  
    return -1;  
}
```



# PreferenceActivity 예제 2

## ■ MainActivity.JAVA

@Override

```
public boolean onCreateOptionsMenu(Menu menu) {  
    getMenuInflater().inflate(R.menu.menu, menu);  
    return true;  
}
```

@Override

```
public boolean onOptionsItemSelected(MenuItem item) {  
    if (item.getItemId() == R.id.action_settings) {  
        Intent SettingActivity = new Intent(this, SettingsActivity.class);  
        startActivity(SettingActivity);  
    }  
    return super.onOptionsItemSelected(item);  
}
```



```
public class SettingsActivity extends PreferenceActivity {
```

```
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    addPreferencesFromResource(R.xml.pref_settings);

    setOnPreferenceChange(findPreference("userName"));
    setOnPreferenceChange(findPreference("userNameOpen"));
    setOnPreferenceChange(findPreference("autoUpdate_ringtone"));
}
```

```
private void setOnPreferenceChange(Preference mPreference) {
    mPreference.setOnPreferenceChangeListener(preferenceChangeListener);
    onPreferenceChangeListener.onPreferenceChange(mPreference,
        PreferenceManager.getDefaultSharedPreferences(
            mPreference.getContext()).getString(
                mPreference.getKey(), ""));
}
```



# PreferenceActivity 예제 2

## ■ SettingActivity.JAVA

```
private Preference.OnPreferenceChangeListener preferenceChangeListener =  
    new Preference.OnPreferenceChangeListener() {
```

*@Override*

```
public boolean onPreferenceChange(Preference preference,  
                                   Object newValue) {  
    String stringValue = newValue.toString();  
    if (preference instanceof EditTextPreference) {  
        preference.setSummary(stringValue);  
    } else if (preference instanceof ListPreference) {  
        ListPreference listPreference = (ListPreference) preference;  
        int index = listPreference.findIndexOfValue(stringValue);  
        preference.setSummary(index >= 0 ?  
                                listPreference.getEntries()[index] : null);  
    } else if (preference instanceof RingtonePreference) {
```



# PreferenceActivity 예제 2

## ■ SettingActivity.JAVA

```
if (TextUtils.isEmpty(stringValue)) {
    preference.setSummary("무음으로 설정됨");
} else {
    Ringtone ringtone = RingtoneManager.getRingtone(
        preference.getContext(), Uri.parse(stringValue));
    if (ringtone == null) {
        preference.setSummary(null);
    } else {
        String name = ringtone.getTitle(preference.getContext());
        preference.setSummary(name);
    }
}
return true;
};
}
```





# PreferenceActivity 예제 2

## ■ pref\_settings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen xmlns:android=
    "http://schemas.android.com/apk/res/android" >

    <CheckBoxPreference
        android:defaultValue="false"
        android:key="useUserName"
        android:summaryOff="사용자의 이름을 사용하지 않습니다"
        android:summaryOn="사용자의 이름을 사용합니다"
        android:title="사용자 이름 사용" />

    <EditTextPreference
        android:defaultValue="Mir"
        android:dependency="useUserName"
        android:key="userName"
        android:maxLines="1"
        android:selectAllOnFocus="true"
        android:singleLine="true"
        android:title="사용자 이름 설정" />
```



# PreferenceActivity 예제 2

## ■ pref\_settings.xml

```
<ListPreference
    android:defaultValue="0"
    android:dependency="useUserName"
    android:entries="@array/userNameOpen"
    android:entryValues="@array/userNameOpen_values"
    android:key="userNameOpen"
    android:negativeButtonText="@null"
    android:positiveButtonText="@null"
    android:title="이름 공개 범위" />

<PreferenceCategory android:title="자동 업데이트" >
    <SwitchPreference
        android:defaultValue="false"
        android:key="autoUpdate"
        android:summary="자동 업데이트를 설정합니다"
        android:switchTextOff="OFF"
        android:switchTextOn="ON"
        android:title="자동 업데이트" />
```



# PreferenceActivity 예제 2

## ■ pref\_settings.xml

```
<CheckBoxPreference
    android:defaultValue="false"
    android:dependency="autoUpdate"
    android:key="useUpdateNofiti"
    android:summary="업데이트후 알림을 띄웁니다"
    android:title="알림 사용" />
```

```
<RingtonePreference
    android:defaultValue="content://settings/system/notification_sound"
    android:dependency="useUpdateNofiti"
    android:key="autoUpdate_ringtone"
    android:ringtoneType="notification"
    android:showSilent="true"
    android:title="알림 소리" />
```

```
</PreferenceCategory>
```



# PreferenceActivity 예제 2

## ■ pref\_settings.xml

```
<PreferenceCategory android:title="어플 정보" >
    <Preference
        android:summary="개발자 정보를 확인합니다"
        android:title="개발자 정보" >
        <intent
            android:targetClass="com.bae.preferenceactivity02.MadeActivity"
            android:targetPackage="com.bae.preferenceactivity02" />
    </Preference>
    <Preference
        android:summary="개발자에게 메일을 보냅니다"
        android:title="메일 보내기" >
        <intent
            android:action="android.intent.action.SENDTO"
            android:data="mailto:hhbae@kbu.ac.kr" />
    </Preference>
</PreferenceCategory>
</PreferenceScreen>
```



# PreferenceActivity 예제 2

## ■ activity\_made.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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=".MadeActivity">

    <TextView
        android:id="@+id/text"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="20dp"
        android:gravity="center"/>

</LinearLayout>
```



# PreferenceActivity 예제 2

## ■ MainActivity.JAVA

```
public class MainActivity extends AppCompatActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_made);  
  
        TextView textView = findViewById(R.id.text);  
        textView.setLinksClickable(true);  
        textView.setMovementMethod(LinkMovementMethod.getInstance());  
        String urlNaver = "<a href='http://www.naver.com'>네이버</a><br>";  
  
        textView.setText(Html.fromHtml(urlNaver));  
    }  
}
```



# PreferenceActivity 예제 2

## ■ arrays.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string-array name="userNameOpen">
        <item>항상 공개</item>
        <item>친구에게만</item>
        <item>비공개</item>
    </string-array>
    <string-array name="userNameOpen_values">
        <item>1</item>
        <item>0</item>
        <item>-1</item>
    </string-array>
</resources>
```



# PreferenceActivity 예제 2

- Google의 Android는 매년 새로운 버전이 나오면서 새로운 기능을 제공하는 API를 발표함
- 새로운 버전이다 보니 이전의 버전에서 새로운 기능 또는 UI 등을 사용하기 힘든 상황
- 그래서 Google은 Android Support Library(Support 라이브러리)들을 제공해서 하위 버전에서도 상위 버전의 기능과 UI를 사용할 수 있게 지원
- 그래서 Android 개발자들은 기본 API와 더불어 Support Library를 살펴봐야 함





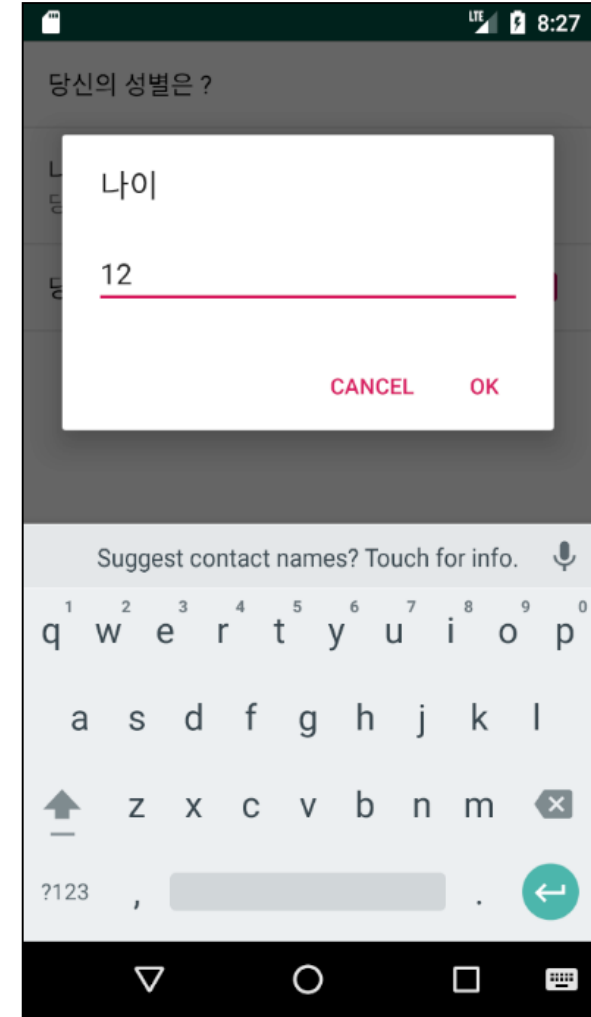
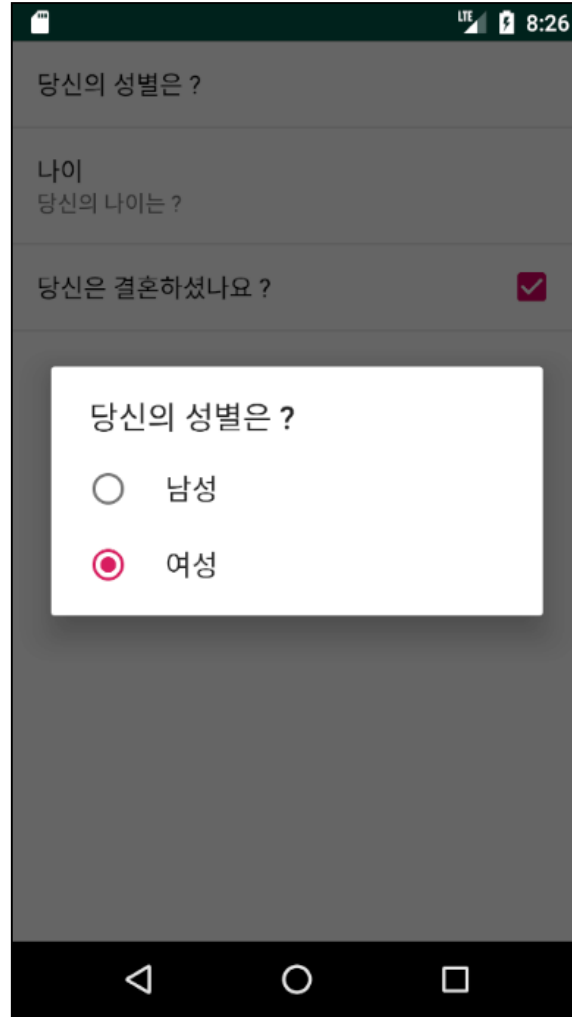
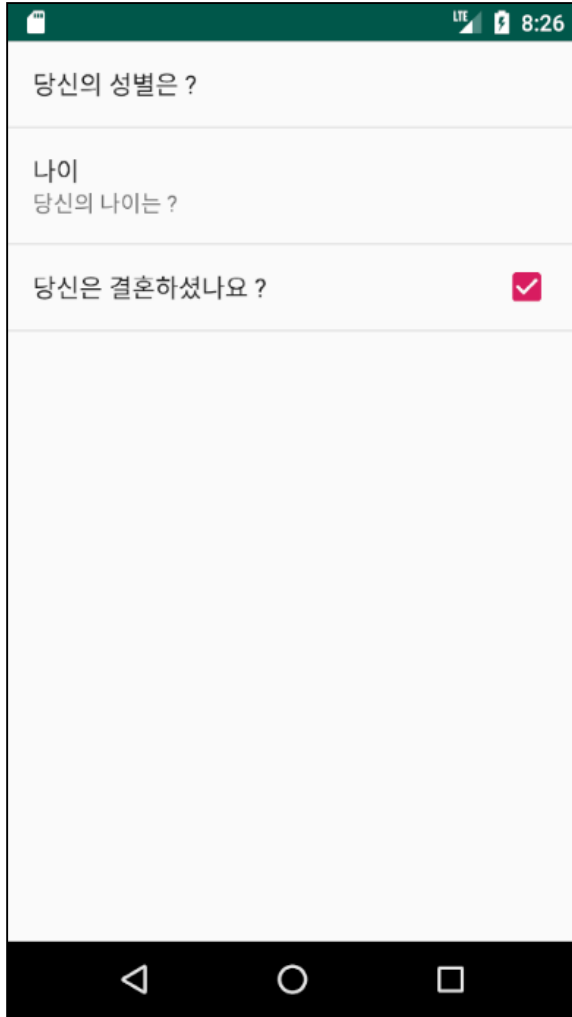
# PreferenceActivity 예제 3

- 다음과 같이 애플리케이션의 환경을 설정하는 프로그램을 만들어보자



# PreferenceActivity 예제 3

## ■ 실행 화면





# PreferenceActivity 예제 3



```
<?xml version="1.0" encoding="utf-8" ?>
<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"
    android:orientation="vertical"
    tools:context=".MainActivity">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:text="ID를 입력하세요" />
    <EditText
        android:id="@+id/editText1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:hint="ID를 입력"
        android:ems="10" >
    <requestFocus />
    </EditText>
```



# PreferenceActivity 예제 3

## ■ 사용자 인터페이스

<TextView

```
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:layout_marginTop="30dp"  
android:text="비밀번호를 입력하세요" />
```

<EditText

```
android:id="@+id/editText2"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:inputType="textPassword"  
android:hint="비밀번호를 입력하세요"  
android:ems="10"/>
```



# PreferenceActivity 예제 3

## ■ 사용자 인터페이스

<CheckBox

```
    android:id="@+id/checkBox"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_marginTop="10dp"  
    android:text="자동 로그인" />
```

<Button

```
    android:id="@+id/button"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text="로그인"/>
```

</LinearLayout>



# PreferenceActivity 예제 3

## ■ JAVA 프로그램

```
public class MainActivity extends AppCompatActivity {
```

```
    @Override
```

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

```
        final EditText editText1 = findViewById(R.id.editText1);
```

```
        final EditText editText2 = findViewById(R.id.editText2);
```

```
        final CheckBox checkBox = findViewById(R.id.checkBox);
```

```
        final SharedPreferences setting = getSharedPreferences("setting",  
                                                                MODE_PRIVATE);
```

```
        editText1.setText(setting.getString("ID", ""));
```

```
        editText2.setText(setting.getString("PW", ""));
```

```
        String id = editText1.getText().toString();
```

```
        String pw = editText2.getText().toString();
```



# PreferenceActivity 예제 3

```
if (id.equals("") && pw.equals("")) {
    checkBox.setChecked(false);
    Toast.makeText(getApplicationContext(), "로그인 기록 없음",
                                   Toast.LENGTH_SHORT).show();
} else
    checkBox.setChecked(true);
checkBox.setOnCheckedChangeListener(new CompoundButton.
                                   OnCheckedChangeListener() {

    @Override
    public void onCheckedChanged(CompoundButton buttonView,
                                   boolean isChecked) {

        if (isChecked) {
            editText1.setText(setting.getString("ID", ""));
            editText2.setText(setting.getString("PW", ""));
        } else {
            editText1.setText("");
            editText2.setText("");
            checkBox.setChecked(false);
        }
    }
});
```



# PreferenceActivity 예제 3

## ■ JAVA 프로그램

```
Button button = findViewById(R.id.button);
button.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (checkBox.isChecked()) {
            editText1.setText(setting.getString("ID", ""));
            editText2.setText(setting.getString("PW", ""));
        }
        String id = editText1.getText().toString();
        String pw = editText2.getText().toString();
        if (id.equals("") || pw.equals("")) {
            Toast.makeText(getApplicationContext(), "입력해주세요",
                Toast.LENGTH_SHORT).show();
        } else {
```





# PreferenceActivity 예제 3

## ■ JAVA 프로그램

```
Toast.makeText(getApplicationContext(), "Login ID " + id,
                                           Toast.LENGTH_SHORT).show();

editText1.setText("");
editText2.setText("");
checkBox.setChecked(false);
editText1.requestFocus();

SharedPreferences.Editor editor = setting.edit();
editor.putString("ID", id);
editor.putString("PW", pw);
editor.apply();
editor.commit();
    }
}
});
}
```



# PreferenceActivity 예제 3

## ■ JAVA 프로그램

@Override

```
public boolean onCreateOptionsMenu(Menu menu) {  
    getMenuInflater().inflate(R.menu.menu, menu);  
    return true;  
}
```

@Override

```
public boolean onOptionsItemSelected(MenuItem item) {  
    switch (item.getItemId()) {  
        case R.id.action_settings:  
            Intent intent = new Intent(MainActivity.this, Setting.class);  
            startActivity(intent);  
            return true;  
        }  
    return false;  
}
```



# PreferenceActivity 예제 3

## ■ menu.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
      xmlns:app="http://schemas.android.com/apk/res-auto">
  <item
    android:id="@+id/action_settings"
    android:orderInCategory="100"
    android:title="@string/action_settings"
    app:showAsAction="never" />
</menu>
```



# PreferenceActivity 예제 3

## ■ settings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen xmlns:android="http://schemas.android.com/apk/res/android"
    android:key="Pref_Screen">

    <ListPreference
        android:defaultValue="0"
        android:dialogTitle="당신의 성별은 ?"
        android:entries="@array/gender"
        android:entryValues="@array/gender_values"
        android:key="gender"
        android:negativeButtonText="@null"
        android:positiveButtonText="@null"
        android:title="당신의 성별은 ?" />
```



# PreferenceActivity 예제 3

## ■ settings.xml

```
<EditTextPreference
    android:key="age"
    android:defaultValue="0"
    android:summary="당신의 나이는 ?"
    android:title="나이" />

<CheckBoxPreference
    android:key="married"
    android:defaultValue="true"
    android:title="당신은 결혼하셨나요 ?" />
</PreferenceScreen>
```



# PreferenceActivity 예제 3

## ■ Setting.JAVA

```
public class Setting extends PreferenceActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        addPreferencesFromResource(R.xml.settings);  
    }  
}
```



# PreferenceActivity 예제 3

## ■ arrays.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string-array name="gender">
        <item>남성</item>
        <item>여성</item>
    </string-array>

    <string-array name="gender_values">
        <item>0</item>
        <item>1</item>
    </string-array>
</resources>
```



# PreferenceActivity 예제 3



## ■ strings.xml

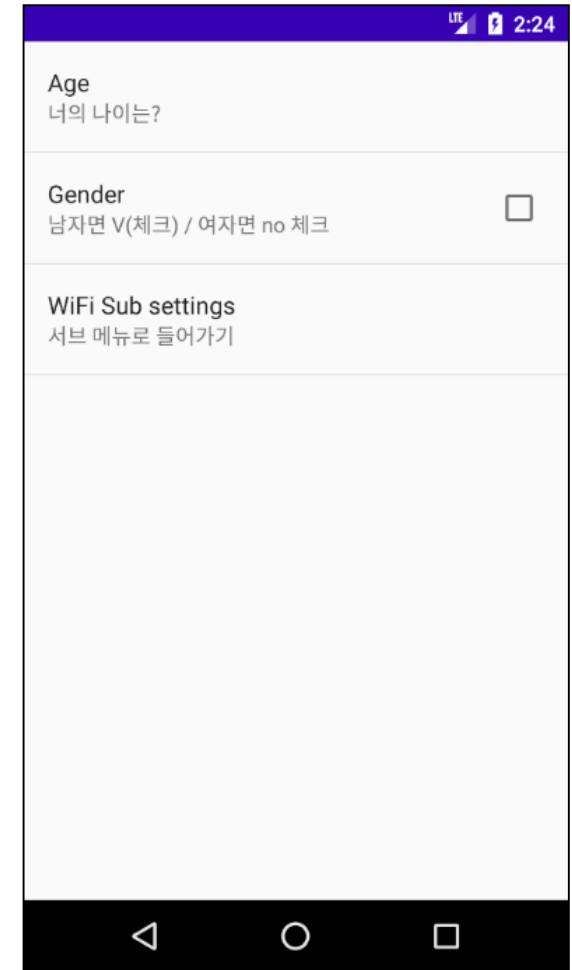
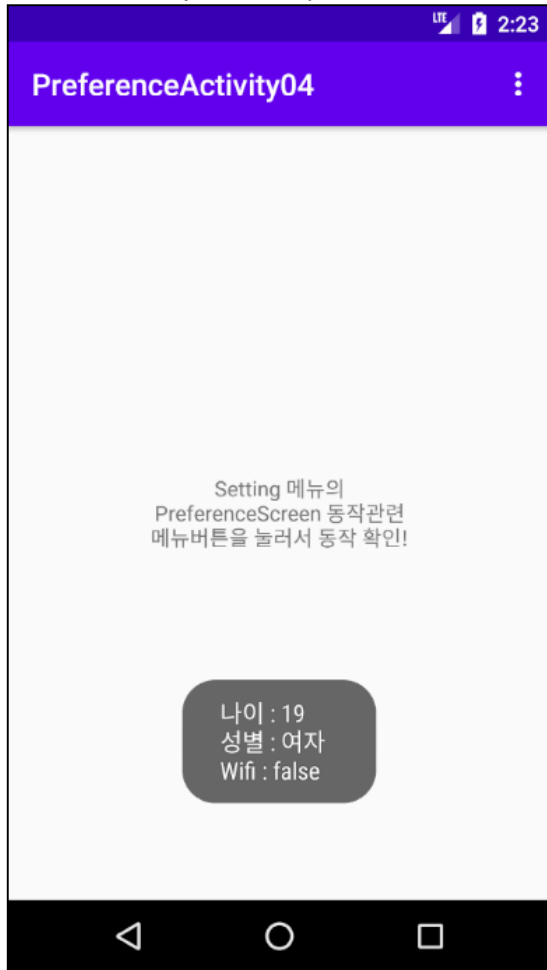
```
<resources>
  <string name="app_name">PreferenceActivity03</string>
  <string name="action_settings">Settings</string>
</resources>
```





# PreferenceActivity 예제 4

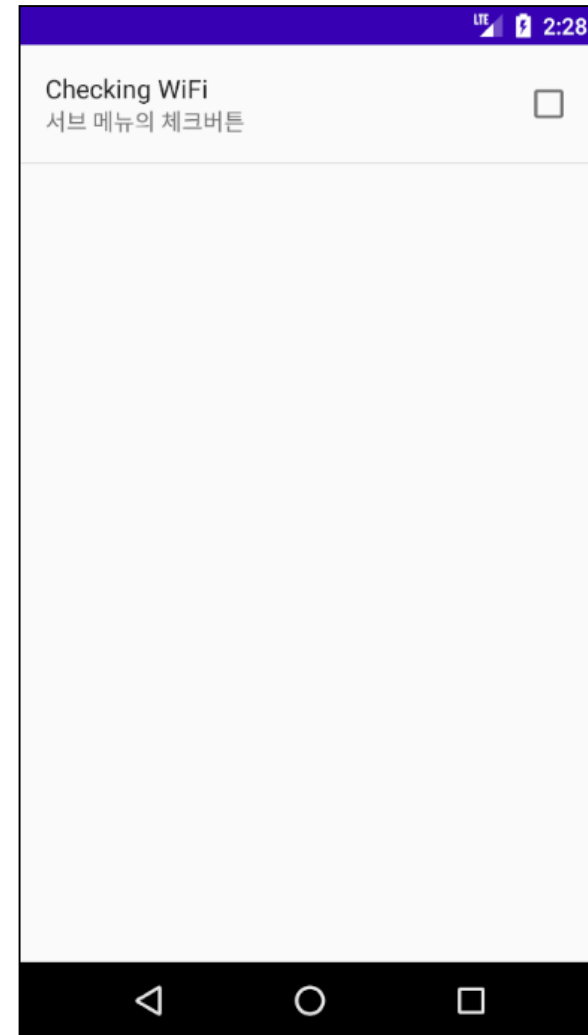
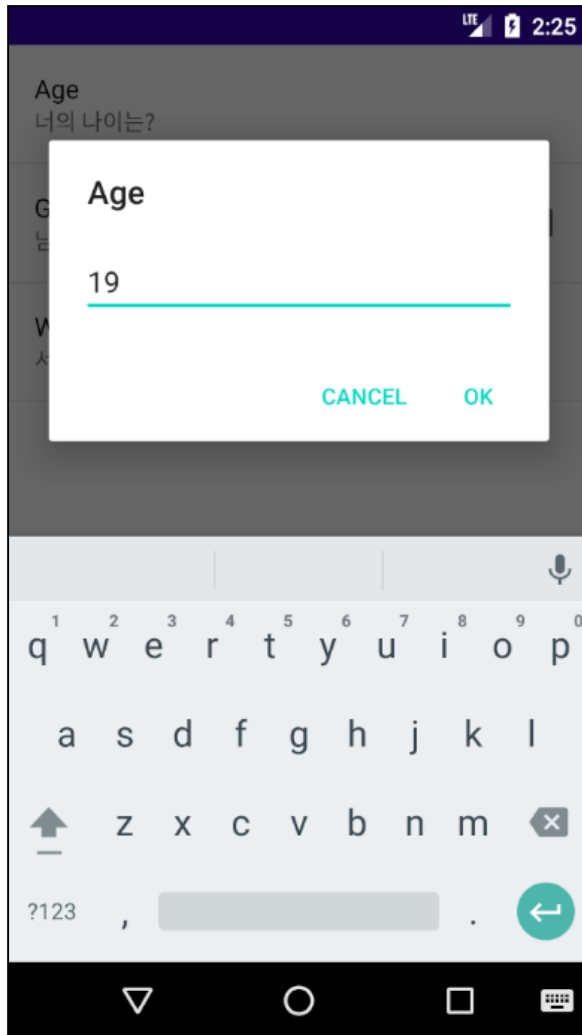
- 다음과 같이 애플리케이션의 환경을 설정하는 프로그램을 만들어보자





# PreferenceActivity 예제 4

## ■ 실행 화면





# PreferenceActivity 예제 4

## ■ 사용자 인터페이스

```
<?xml version="1.0" encoding="utf-8"?>
<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=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:gravity="center"
        android:text="@string/context" />
</LinearLayout>
```



# PreferenceActivity 예제 4

## ■ MainActivity.JAVA

```
public class MainActivity extends AppCompatActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
    }  
  
    public boolean onCreateOptionsMenu(Menu menu) {  
        super.onCreateOptionsMenu(menu);  
        MenuInflater inflater = getMenuInflater();  
        inflater.inflate(R.menu.menu, menu);  
        return true;  
    }  
}
```



# PreferenceActivity 예제 4

## ■ MainActivity.JAVA

```
public boolean onOptionsItemSelected(MenuItem item) {  
    switch(item.getItemId()){  
        case R.id.settings:  
            Intent intent = new Intent(this, PreferenceActivity.class);  
            startActivity(intent);  
            break;  
        default:  
            Toast.makeText(this, "아무것도 없음", Toast.LENGTH_SHORT).show();  
            break;  
    }  
    return super.onOptionsItemSelected(item);  
}  
  
protected void onResume() {  
    super.onResume();  
    viewPopupStatus();  
}
```



# PreferenceActivity 예제 4

## ■ MainActivity.JAVA

```
public void viewPopupStatus() {  
    String current = "";  
    current += "나이 : " + PreferenceActivity.getAge(this);  
    if(PreferenceActivity.getGender(this))  
        current += "\n성별 : 남자\n";  
    else  
        current += "\n성별 : 여자\n";  
    current += "Wifi : " + PreferenceActivity.getWifi(this);  
    Toast.makeText(this, current, Toast.LENGTH_LONG).show();  
}
```



# PreferenceActivity 예제 4

## ■ preference.xml

```
<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen xmlns:android
                    ="http://schemas.android.com/apk/res/android">
    <EditTextPreference
        android:key="age"
        android:title="Age"
        android:summary="너의 나이는?"
        android:defaultValue="19"/>

    <CheckBoxPreference
        android:key="gender"
        android:title="Gender"
        android:summary="남자면 V(체크) / 여자면 no 체크"
        android:defaultValue="true"/>
```



# PreferenceActivity 예제 4

## ■ preference.xml

```
<!-- SubMenu 관련 -->
<PreferenceScreen
    android:key="submenuwifi"
    android:title="WiFi Sub settings"
    android:summary="서브 메뉴로 들어가기">

    <CheckBoxPreference
        android:key="wifi"
        android:title="Checking WiFi"
        android:summary="서브 메뉴의 체크버튼" />
    </PreferenceScreen>
</PreferenceScreen>
```





# PreferenceActivity 예제 4

## ■ PreferenceActivity.JAVA

```
public class PreferenceActivity extends android.preference.PreferenceActivity {  
    static String USER_AGE = "age";  
    static String AGE_VALUE = null;  
    static String USER_GENDER = "gender";  
    static boolean GENDER_VALUE = true;  
    static String USER_WIFI = "wifi";  
    static boolean WIFI_VALUE = true;
```

### @Override

```
public void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    addPreferencesFromResource(R.xml.preference);  
}
```

```
public static String getAge(Context context){  
    return PreferenceManager.getDefaultSharedPreferences(context).  
        getString(USER_AGE, AGE_VALUE);  
}
```



# PreferenceActivity 예제 4

## ■ PreferenceActivity.JAVA

```
public static boolean getGender(Context context){  
    return PreferenceManager.getDefaultSharedPreferences(context).  
        getBoolean(USER_GENDER, GENDER_VALUE);  
}  
  
public static boolean getWifi(Context context) {  
    return PreferenceManager.getDefaultSharedPreferences(context).  
        getBoolean(USER_WIFI, WIFI_VALUE);  
}  
}
```



# PreferenceActivity 예제 4

## ■ menu.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/settings"
        android:title="Setting"
        android:alphabeticShortcut="a" />

    <item
        android:id="@+id/dummy"
        android:title="Dummy" />
</menu>
```



# PreferenceActivity 예제 4

## ■ arrays.xml

```
<resources>
  <string name="app_name">PreferenceActivity04</string>
  <string name="context">Setting 메뉴의
    \nPreferenceScreen 동작관련
    \n메뉴버튼을 눌러서 동작 확인! </string>
</resources>
```