

The background of the slide is a dark, grayscale image of a city skyline at night. On the right side, a large Ferris wheel is visible, and on the left, several skyscrapers are illuminated. The entire scene is reflected in a body of water at the bottom. A white rectangular border is centered on the slide, enclosing the title and the course information.

Java Programing

1344033 컴퓨터수학과

정 재혁

개요

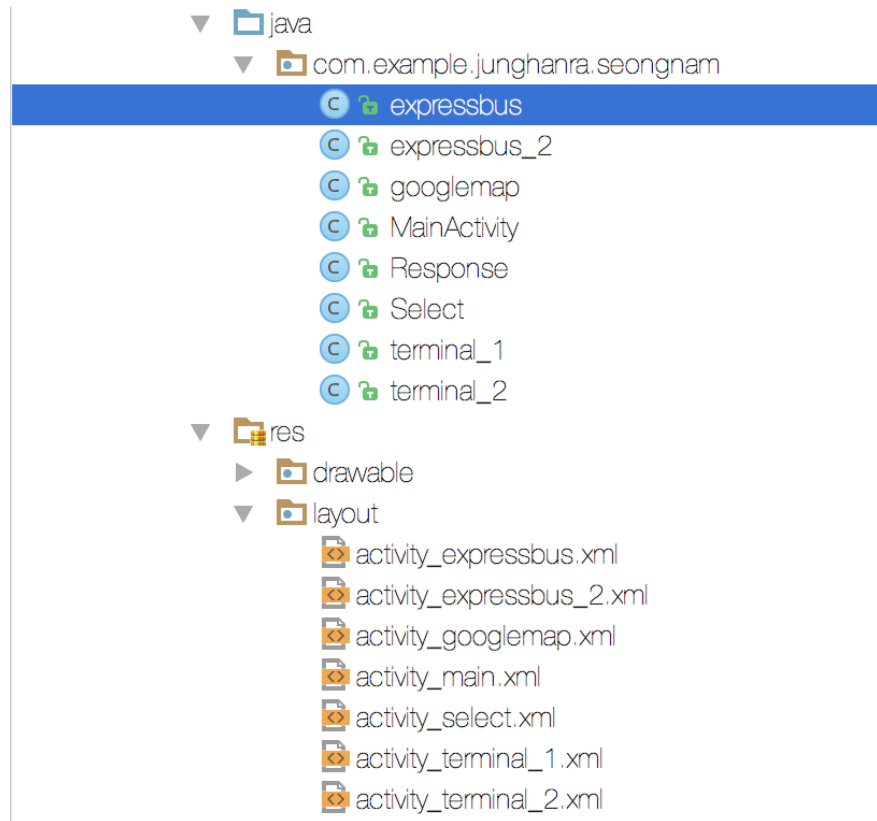


Java로 무엇을 만들수 있을까 생각하다가
안드로이드를 개발할 때 Java를 사용한다는 얘기를 듣고 선택
(Java는 플랫폼 독립적이라는 성격 때문에 안드로이드 개발에 적합)

개발 과정

안드로이드

개발 과정



안드로이드는 기본적으로 Activity에
java 파일과 Layout으로 구성

개발과정

공공데이터포털 API

DATA 공공데이터포털 .GO .KR

데이터셋 | 활용사례 | 참여마당 | 정보공유 | 전체메뉴

마이페이지

오른API >

- 개발계정
- 활용현황
- 운영계정
- 인증키 발급현황

DATA

나의 문의

나의 관심

회원정보

공공데이터 제공신청

개발계정

☞ > 마이페이지 > OPEN API > 개발계정

활용자가 개발계정을 신청중인 단계

신청	· 반려	0건
0건	· 보류	0건

활용자의 개발계정신청이 승인되어 활용중인 단계

활용	· 변경신청	0건
2건	· 반려	0건
	· 보류	0건

활용자가 중지신청하여 운영이 중지된 단계

중지		0건
----	--	----

상세검색

총 2 건

[승인] 고속버스정보 서비스

서비스유형 : REST 분류 : 지역개발 > 지역및도시 제공기관 : 국토교통부

신청일:2017-05-30 [연장신청] 만료예정 :2019-05-30

[승인] 시외버스정보 서비스

서비스유형 : REST 분류 : 지역개발 > 지역및도시 제공기관 : 국토교통부

신청일:2017-05-30 [연장신청] 만료예정 :2019-05-30

<< < 1 > >>

```
<items>
  <arrplacnm>광주종합버스터미널</arrplacnm>
  <charge>25500</charge>
  <depplandtime>06시 00분</depplandtime>
  <level>우등</level>
  <routeid>NAEK120500</routeid>
</items>
<items>
  <arrplacnm>광주종합버스터미널</arrplacnm>
  <charge>25500</charge>
  <depplandtime>06시 50분</depplandtime>
  <level>우등</level>
  <routeid>NAEK120500</routeid>
</items>
<items>
  <arrplacnm>광주종합버스터미널</arrplacnm>
  <charge>17300</charge>
  <depplandtime>07시 30분</depplandtime>
  <level>일반</level>
  <routeid>NAEK120500</routeid>
</items>
<items>
  <arrplacnm>광주종합버스터미널</arrplacnm>
  <charge>25500</charge>
  <depplandtime>08시 10분</depplandtime>
  <level>우등</level>
  <routeid>NAEK120500</routeid>
</items>
<items>
  <arrplacnm>광주종합버스터미널</arrplacnm>
  <charge>25500</charge>
  <depplandtime>09시 30분</depplandtime>
  <level>우등</level>
  <routeid>NAEK120500</routeid>
</items>
<items>
  <arrplacnm>광주종합버스터미널</arrplacnm>
  <charge>25500</charge>
  <depplandtime>09시 40분</depplandtime>
  <level>우등</level>
  <routeid>NAEK120500</routeid>
</items>
```

공공데이터 포털에서 시외버스터미널&고속버스터미널 API사용
(총 4개 xml사용 각각 시외버스 고속버스 로 , 터미널명&터미널ID와
터미널명&터미널ID&출발시간&요금 으로 이루어진 xml)

Java Class 파일

개발과정

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_terminal_1);
    ListView = (ListView)findViewById(R.id.ListView1);

    final ArrayList<Response> list = xmlParser();

    final String[] data = new String[list.size()];
    for(int i=0;i<list.size();i++) {
        data[i] = list.get(i).getarrplacnm();
    }
    ArrayAdapter<String> adapter = new ArrayAdapter<>(getApplicationContext(),
        android.R.layout.simple_list_item_1,
        data);
    ListView.setAdapter(adapter);
    EditText editTextFilter = (EditText)findViewById(R.id.editTextFilter);
    editTextFilter.addTextChangedListener(new TextWatcher() {
        @Override
        public void afterTextChanged(Editable edit) {
            String filterText = edit.toString(); if (filterText.length() > 0)
            { listView.setFilterText(filterText); } else { listView.clearTextFilter(); }
        }

        @Override
        public void beforeTextChanged(CharSequence s, int start, int count, int after) {}

        @Override
        public void onTextChanged(CharSequence s, int start, int before, int count) {}
    });
    listView.setOnItemClickListener((parent, v, position, id) -> {
        String content = RouteidVector.get(position);
        Intent intent = new Intent(getApplicationContext(), terminal_2.class);
        intent.putExtra("content",content);
        startActivity(intent);
    });
}
```

```
protected ArrayList<Response> xmlParser() {
    RouteidVector = new Vector<String>();
    ArrayList<Response> arrayList = new ArrayList<>();
    InputStream is = getResources().openRawResource(R.raw.intercity_bus);

    // xmlPullParser
    try {
        XmlPullParserFactory factory = XmlPullParserFactory.newInstance();
        XmlPullParser parser = factory.newPullParser();
        parser.setInput(new InputStreamReader(is, "UTF-8"));
        int eventType = parser.getEventType();
        Response items = null;

        while(eventType != XmlPullParser.END_DOCUMENT) {
            switch (eventType) {
                case XmlPullParser.START_TAG:
                    String startTag = parser.getName();
                    if(startTag.equals("items")) {
                        items = new Response();
                    }
                    if(startTag.equals("arrplacnm")) {
                        items.setarrplacnm(parser.nextText());
                    }
                    if(startTag.equals("routeid")){
                        items.setrouteid(parser.nextText());
                    }

                    break;
                case XmlPullParser.END_TAG:
                    String endTag = parser.getName();
                    if(endTag.equals("items")) {
                        RouteidVector.add(items.getrouteid());
                        arrayList.add(items);
                    }

                    break;
            }
            eventType = parser.next();
        }
    }
}
```

Java 클래스 파일에는 해당 Layout에 무엇을 하는지 담겨 있음.
(xml pull parser를 이용하여 각 데이터를 vector에 저장후 , Layout- listview에 뿌려줌)

Java class 파일

개발과정

```
public class Response {
    String arrplacnm, routeid, depplandtime, charge, level;
    public String getarrplacnm() { return arrplacnm; }

    public void setarrplacnm(String arrplacnm) { this.arrplacnm = arrplacnm; }
    public String getrouteid() { return routeid; }

    public void setrouteid(String routeid) { this.routeid = routeid; }

    public String getdepplandtime() { return depplandtime; }

    public void setdepplandtime(String depplandtime) { this.depplandtime = depplandtime; }
    public String getcharge() { return charge; }

    public void setcharge(String charge) { this.charge = charge; }
    public String getlevel(){return level;}
    public void setlevel(String level) { this.level=level; }

}
```

```
try {
    XmlPullParserFactory factory = XmlPullParserFactory.newInstance();
    XmlPullParser parser = factory.newPullParser();
    parser.setInput(new InputStreamReader(is, "UTF-8"));
    int eventType = parser.getEventType();
    Response items = null;

    while(eventType != XmlPullParser.END_DOCUMENT) {
        switch (eventType) {
            case XmlPullParser.START_TAG:
                String startTag = parser.getName();
                if(startTag.equals("items")) {
                    items = new Response();
                }

                if (startTag.equals("routeid")) {
                    items.setrouteid(parser.nextText());
                    Log.d("TEST", items.getrouteid());
                }

                if (startTag.equals("depplandtime")) {
                    items.setdepplandtime(parser.nextText());
                }

                if (startTag.equals("charge")) {
                    items.setcharge(parser.nextText());
                }

                break;
            case XmlPullParser.END_TAG:
                String endTag = parser.getName();
                if(endTag.equals("items")) {
                    arrayList.add(items);
                    if(data1.equals(items.getrouteid())){
                        Depplandtime.add(items.getdepplandtime());
                        Charge.add(items.getcharge());
                    }
                }
                break;
        }
        eventType = parser.next();
    }
}
```


Layout

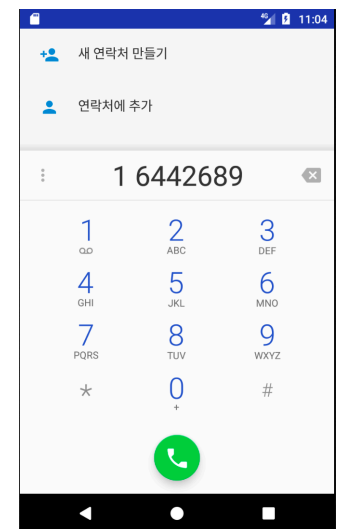
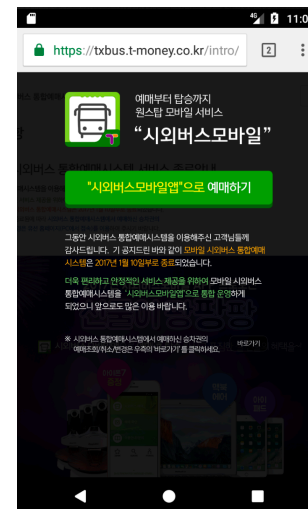
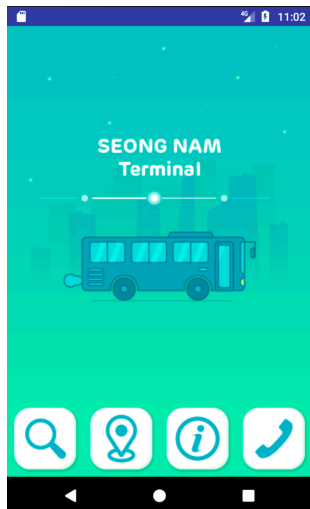
개발과정

```
<RelativeLayout 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.example.junghanra.seongnam.terminal_1">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="26sp"
        android:layout_marginRight="8dp"
        android:id="@+id/textView1"
        android:text="터미널 검색" />
    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_toRightOf="@id/textView1"
        android:id="@+id/editTextFilter"/>
    <ListView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_below="@id/editTextFilter"
        android:textFilterEnabled="true"
        android:id="@+id/listview1"/>
</RelativeLayout>
```

Layout파일에 화면에 구성될 것들을 생성

결과

결과



결과

