안드로이드 프로그래밍
(미니 프로젝트)

1-1. BaseAdapter 예제 프로그램

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="주소관리"
        android:textSize="30sp"
        android:layout_gravity="center"/>
    <ListView
        android:id="@+id/list"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>
</LinearLayout>
```

item.xml

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:padding="10sp">
    <TextView
        android:id="@+id/name"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="이름"
        android:textSize="20sp"
        android:layout_alignParentLeft="true"/>
    <TextView
        android:id="@+id/tel"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="전화번호"
        android:textSize="15sp"
        android:layout_alignParentRight="true"/>
    <TextView
        android:id="@+id/addr"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="주소"
        android:textSize="15sp"
        android:layout_below="@id/name"/>
</RelativeLayout>
```

```java
public class MainActivity extends Activity {
    ArrayList<Item> array;
    MyAdapter adapter;

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

        array=new ArrayList<Item>();
        Item item=null;
        item=new Item("홍길동", "001-0001-0001", "인천 서구 경서동");
        array.add(item);
        item=new Item("이순신", "001-0002-0002", "인천 남구 용현동");
        array.add(item);

        adapter=new MyAdapter();
        ListView list=(ListView)findViewById(R.id.list);
        list.setAdapter(adapter);
    }

    class Item {
        String name;
        String tel;
        String addr;
        public Item(String name, String tel, String addr) {
            this.name = name;
            this.tel = tel;
            this.addr = addr;
        }
    }

    class MyAdapter extends BaseAdapter {
        public int getCount() {
            return array.size();
        }
        public Object getItem(int arg0) {
            return null;
        }
        public long getItemId(int arg0) {
            return 0;
        }

        public View getView(final int id, View convertView, ViewGroup parent) {
            LayoutInflater inflater=getLayoutInflater();
            if(convertView == null){
                convertView=inflater.inflate(R.layout.item, parent, false);
            }
            TextView name=(TextView)convertView.findViewById(R.id.name);
            TextView tel=(TextView)convertView.findViewById(R.id.tel);
            TextView addr=(TextView)convertView.findViewById(R.id.addr);

            name.setText(array.get(id).name);
            tel.setText(array.get(id).tel);
            addr.setText(array.get(id).addr);
            return convertView;
        }
    }
}
```
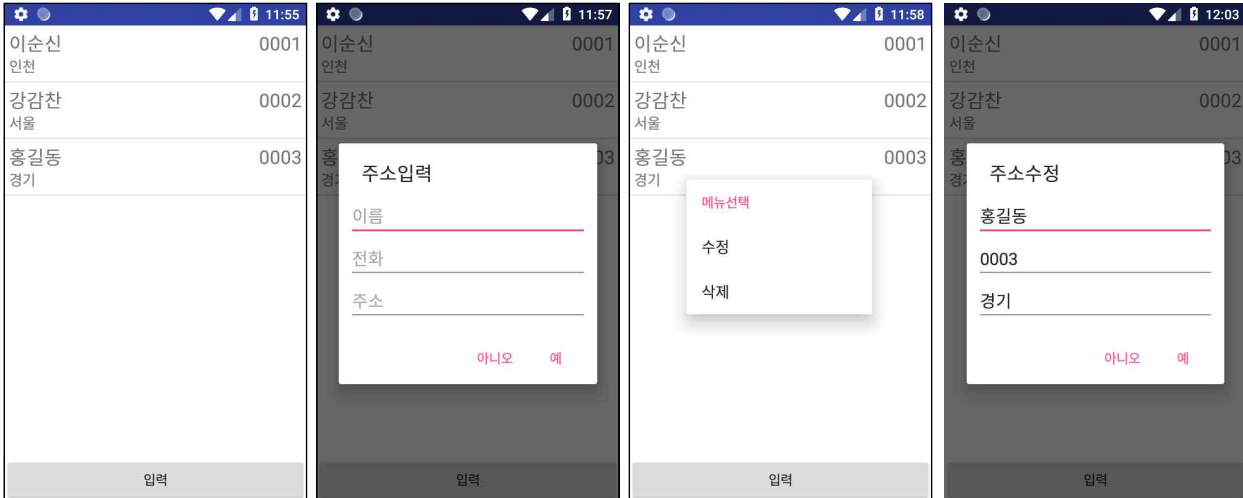
```xml
<style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">
    …
    <item name="windowNoTitle">true</item>
    …
</style>
```

1-2. CursorAdapter 예제 프로그램

```xml
<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">
    <ListView
        android:id="@+id/list"
        android:layout_width="match_parent"
        android:layout_height="0px"
        android:layout_weight="1"/>
    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="입력"
        android:onClick="mClick"/>
</LinearLayout>
```

item.xml

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="5sp">
    <TextView
        android:id="@+id/name"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="20sp"
        android:layout_alignParentLeft="true"/>
    <TextView
        android:id="@+id/tel"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="20sp"
        android:layout_alignParentRight="true"/>
    <TextView
        android:id="@+id/addr"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="15sp"
        android:layout_below="@id/tel"/>
</RelativeLayout>
```

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="10sp">
    <EditText
        android:id="@+id/name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="이름"/>
    <EditText
        android:id="@+id/tel"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="전화"/>
    <EditText
        android:id="@+id/addr"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="주소"/>
</LinearLayout>
```

```java
public class AddressDB extends SQLiteOpenHelper{
    public AddressDB(Context context) {
        super(context, "address.db", null, 1);
    }
    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL("create table address(_id integer primary key autoincrement, name text, tel text, addr text);");
        db.execSQL("insert into address(name, tel, addr) values('이순신','0001','인천');");
        db.execSQL("insert into address(name, tel, addr) values('강감찬','0002','서울');");
        db.execSQL("insert into address(name, tel, addr) values('홍길동','0003','경기');");
    }
    @Override
    public void onUpgrade(SQLiteDatabase arg0, int arg1, int arg2) {
    }
}
```

```java
public class MainActivity extends Activity {
    AddressDB helper;
    SQLiteDatabase db;

    MyAdapter adapter;
    ListView list;
    Cursor cursor;

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

        helper=new AddressDB(this);
        db=helper.getWritableDatabase();

        cursor=db.rawQuery("select * from address", null);
        adapter = new MyAdapter(this, cursor);

        list=(ListView)findViewById(R.id.list);
        list.setAdapter(adapter);
        registerForContextMenu(list);
    }
```

```java
SimpleCursorAdapter adapter;
 ...

adapter=new SimpleCursorAdapter(
        list.getContext(),
        R.layout.item,
        cursor,
        new String[] {"name", "tel", "addr"},
        new int[] {R.id.name, R.id.tel, R.id.addr});
```

```java
public void mClick(View v) {
    LayoutInflater inflater=getLayoutInflater();
    final LinearLayout input=(LinearLayout)inflater.inflate(R.layout.input, null);

    AlertDialog.Builder box1= new AlertDialog.Builder(this);
    box1.setTitle("주소입력");
    box1.setView(input);

    box1.setPositiveButton("예", new DialogInterface.OnClickListener() {
        public void onClick(DialogInterface arg0, int arg1) {
            EditText name=(EditText)input.findViewById(R.id.name);
            EditText tel=(EditText)input.findViewById(R.id.tel);
            EditText addr=(EditText)input.findViewById(R.id.addr);

            String strName=name.getText().toString();
            String strTel=tel.getText().toString();
            String strAddr=addr.getText().toString();

            String sql="insert into address(name, tel, addr) values(";
            sql=sql + "'" + strName + "',";
            sql=sql + "'" + strTel + "',";
            sql=sql + "'" + strAddr + "');";
            db.execSQL(sql);

            Toast.makeText(MainActivity.this, "입력되었습니다.", 0).show();

            cursor=db.rawQuery("select * from address", null);
            adapter.changeCursor(cursor);
        }
    });
    box1.setNegativeButton("아니오", null);
    box1.show();
}

public void onCreateContextMenu(ContextMenu menu, View v, ContextMenuInfo menuInfo) {
    //AdapterContextMenuInfo info=(AdapterContextMenuInfo)menuInfo;
    //Toast.makeText(MainActivity.this, "position::" + info.position + ":" + cursor.getString(1), 0).show();

    menu.setHeaderTitle("메뉴선택");
    menu.add(0, 1, 0, "수정");
    menu.add(0, 2, 0, "삭제");
}

public boolean onContextItemSelected(MenuItem item) {
    //AdapterContextMenuInfo info=(AdapterContextMenuInfo)item.getMenuInfo();
    //Toast.makeText(MainActivity.this, "position::" + info.position + ":" + cursor.getString(1), 0).show();

    LayoutInflater inflater=getLayoutInflater();
    LinearLayout input=(LinearLayout)inflater.inflate(R.layout.input, null);

    final int id = cursor.getInt(0);
    String strName = cursor.getString(1);
    String strTel = cursor.getString(2);
    String strAddr = cursor.getString(3);

    final EditText name=(EditText)input.findViewById(R.id.name);
    final EditText tel=(EditText)input.findViewById(R.id.tel);
    final EditText addr=(EditText)input.findViewById(R.id.addr);

    name.setText(strName);
    tel.setText(strTel);
    addr.setText(strAddr);
```

```java
        switch(item.getItemId()){
        case 1:
            AlertDialog.Builder box1=new AlertDialog.Builder(this);
            box1.setView(input);
            box1.setTitle("주소수정");
            box1.setPositiveButton("예", new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface arg0, int arg1) {
                    String strName=name.getText().toString();
                    String strTel=tel.getText().toString();
                    String strAddr=addr.getText().toString();
                    String sql="update address set ";

                    sql=sql + "name='" + strName + "',";
                    sql=sql + "tel='" + strTel + "',";
                    sql=sql + "addr='" + strAddr + "'";
                    sql=sql + " where _id=" + id;
                    db.execSQL(sql);

                    cursor=db.rawQuery("select * from address", null);
                    adapter.changeCursor(cursor);
                    Toast.makeText(MainActivity.this, "수정되었습니다.", 0).show();
                }
            });
            box1.setNegativeButton("아니오", null);
            box1.show();
            break;
        case 2:
            AlertDialog.Builder box2=new AlertDialog.Builder(this);
            box2.setMessage("삭제하시겠습니까?");
            box2.setPositiveButton("예", new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface arg0, int arg1) {
                    String sql="delete from address where _id=" + id;
                    db.execSQL(sql);

                    cursor=db.rawQuery("select * from address", null);
                    adapter.changeCursor(cursor);
                    Toast.makeText(MainActivity.this, "삭제되었습니다.", 0).show();
                }
            });
            box2.setNegativeButton("아니오", null);
            box2.show();
        }
        return true;
    }

    class MyAdapter extends CursorAdapter {
        public MyAdapter(Context context, Cursor c) {
            super(context, c);
        }
        public void bindView(View view, Context context, Cursor cursor) {
            final TextView name=(TextView)view.findViewById(R.id.name);
            TextView tel=(TextView)view.findViewById(R.id.tel);
            TextView addr=(TextView)view.findViewById(R.id.addr);

            name.setText(cursor.getString(1));
            tel.setText(cursor.getString(2));
            addr.setText(cursor.getString(3));
        }
        public View newView(Context context, Cursor coursor, ViewGroup parent) {
            LayoutInflater inflater=getLayoutInflater();
            return inflater.inflate(R.layout.item, parent, false);
        }
    }
}
```
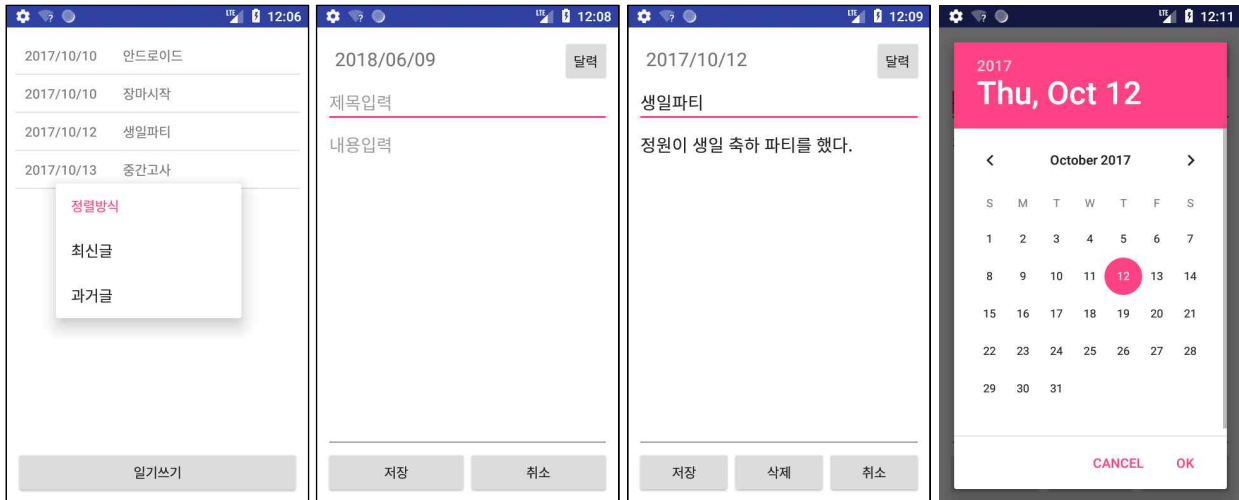
1-3. 일기장(Diary) 예제 프로그램

```xml
<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">
    <ListView
        android:id="@+id/list"
        android:layout_width="match_parent"
        android:layout_height="0px"
        android:layout_weight="1"/>
    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="일기쓰기"
        android:onClick="mClick"/>
</LinearLayout>
```

item.xml

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal"
    android:padding="5sp">
    <TextView
        android:id="@+id/wdate"
        android:layout_width="0px"
        android:layout_height="wrap_content"
        android:layout_weight="1"/>
    <TextView
        android:id="@+id/subject"
        android:layout_width="0px"
        android:layout_height="wrap_content"
        android:layout_weight="3"
        android:paddingLeft="10sp"/>
</LinearLayout>
```

input.xml

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <TextView
```

```xml
            android:id="@+id/wdate"
            android:layout_width="0px"
            android:layout_height="wrap_content"
            android:layout_weight="5"
            android:padding="10sp"/>
        <Button
            android:id="@+id/bntCal"
            android:layout_width="0px"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="달력"
            android:onClick="mClick"/>
    </LinearLayout>
    <EditText
        android:id="@+id/subject"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="제목입력"/>
    <EditText
        android:id="@+id/content"
        android:layout_width="match_parent"
        android:layout_height="0px"
        android:layout_weight="1"
        android:hint="내용입력"
        android:gravity="top"/>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <Button
            android:id="@+id/btnSave"
            android:layout_width="0px"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="저장"
            android:onClick="mClick"/>
        <Button
            android:id="@+id/btnDelete"
            android:layout_width="0px"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="삭제"
            android:onClick="mClick"/>
        <Button
            android:id="@+id/btnCancel"
            android:layout_width="0px"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="취소"
            android:onClick="mClick"/>
    </LinearLayout>
</LinearLayout>
```

DiaryDB.java

```java
public class DiaryDB extends SQLiteOpenHelper{
    public DiaryDB(Context context) {
        super(context, "diary.db", null, 1);
    }

    public void onCreate(SQLiteDatabase db) {
        db.execSQL("create table diary(_id integer primary key autoincrement, wdate text, subject text, content text);");
        db.execSQL("insert into  diary(wdate, subject, content) values('2017/10/10', '안드로이드', '안드로이드 어렵지 않아요!');");
        db.execSQL("insert into  diary(wdate, subject, content) values('2017/10/10', '장마시작', '장마가 시작되었다.');");
        db.execSQL("insert into  diary(wdate, subject, content) values('2017/10/12', '생일파티', '정원이 생일 축하 파티를 했다.');");
        db.execSQL("insert into  diary(wdate, subject, content) values('2017/10/13', '중간고사', '중간고사 시험을 아주 잘 봤다.');");
    }
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("drop table if exists diary");
        onCreate(db);
    }
}
```

```java
public class MainActivity extends Activity {
    DiaryDB helper;
    SQLiteDatabase db;
    Cursor cursor;
    MyAdapter adapter;
    ListView list;

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

        helper=new DiaryDB(this);
        db=helper.getWritableDatabase();
        cursor=db.rawQuery("select * from diary", null);
        adapter=new MyAdapter(this, cursor);

        list=(ListView)findViewById(R.id.list);
        list.setAdapter(adapter);
        list.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
                Intent intent=new Intent(MainActivity.this, Update.class);
                intent.putExtra("id", cursor.getInt(0));
                startActivity(intent);
            }
        });
        registerForContextMenu(list);
    }

    public void mClick(View v){
        Intent intent=new Intent(this, Input.class);
        startActivity(intent);
    }

    class MyAdapter extends CursorAdapter{
        public MyAdapter(Context context, Cursor c) {
            super(context, c);
        }
        public void bindView(View view, Context context, Cursor cursor) {
            TextView wdate=(TextView)view.findViewById(R.id.wdate);
            TextView subject=(TextView)view.findViewById(R.id.subject);
            wdate.setText(cursor.getString(1));
            subject.setText(cursor.getString(2));
        }
        public View newView(Context context, Cursor cursor, ViewGroup parent) {
            LayoutInflater inflater=getLayoutInflater();
            return inflater.inflate(R.layout.item, parent, false);
        }
    }

    public void onCreateContextMenu(ContextMenu menu, View v, ContextMenuInfo menuInfo) {
            menu.setHeaderTitle("정렬방식");
            menu.add(0, 1, 0, "최신글");
            menu.add(0, 2, 0, "과거글");

            super.onCreateContextMenu(menu, v, menuInfo);
    }

    public boolean onContextItemSelected(MenuItem item) {
        switch(item.getItemId()){
        case 1:
            cursor=db.rawQuery("select * from diary order by wdate desc", null);
            break;
        case 2:
            cursor=db.rawQuery("select * from diary order by wdate", null);
        }
        adapter.changeCursor(cursor);
        return super.onContextItemSelected(item);
    }

    protected void onRestart() {
        cursor=db.rawQuery("select * from diary order by wdate", null);
        adapter.changeCursor(cursor);
        super.onRestart();
    }
}
```

```java
public class Input extends Activity{
    DiaryDB helper;
    SQLiteDatabase db;
    TextView wdate;
    int mYear, mMonth, mDay;
    EditText subject, content;
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.input);

      wdate=(TextView)findViewById(R.id.wdate);
        subject=(EditText)findViewById(R.id.subject);
        content=(EditText)findViewById(R.id.content);
        findViewById(R.id.btnDelete).setVisibility(View.GONE);

        helper=new DiaryDB(this);
        db=helper.getWritableDatabase();

        GregorianCalendar calendar=new GregorianCalendar();
        mYear=calendar.get(Calendar.YEAR);
        mMonth=calendar.get(Calendar.MONTH);
        mDay=calendar.get(Calendar.DAY_OF_MONTH);

        wdate.setText(String.format("%04d/%02d/%02d", mYear, mMonth+1, mDay));
    }
    public void mClick(View v){
        switch(v.getId()){
        case R.id.bntCal:
            new DatePickerDialog(this, mDate, mYear, mMonth-1, mDay).show();
            break;
        case R.id.btnSave:
            String srtWdate=wdate.getText().toString();
            String strSubject=subject.getText().toString();
            String strContent=content.getText().toString();

            String sql="insert into diary(wdate, subject, content) values(";
            sql=sql + "" + srtWdate + ",";
            sql=sql + "" + strSubject + ",";
            sql=sql + "" + strContent + ")";

            db.execSQL(sql);
            Toast.makeText(this, "저장되었습니다.", 0).show();
            finish();
        case R.id.btnCancel:
            finish();
        }
    }
    DatePickerDialog.OnDateSetListener mDate=new DatePickerDialog.OnDateSetListener() {
        public void onDateSet(DatePicker view, int year, int month, int day) {
            mYear=year;
            mMonth=month + 1;
            mDay=day;
            wdate.setText(String.format("%04d/%02d/%02d", mYear, mMonth, mDay));
        }
    };
}
```

```java
public class Update extends Activity{
    DiaryDB helper;
    SQLiteDatabase db;
    Cursor cursor;
    TextView wdate;
    EditText subject, content;
    int id;
    int mYear, mMonth, mDay;

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

        wdate=(TextView)findViewById(R.id.wdate);
```

```
        subject=(EditText)findViewById(R.id.subject);
        content=(EditText)findViewById(R.id.content);

        helper=new DiaryDB(this);
        db=helper.getWritableDatabase();

        Intent intent = getIntent();
        id=intent.getIntExtra("id", 0);
        cursor = db.rawQuery("select * from diary where _id=" + id, null);
        while(cursor.moveToNext()){
            wdate.setText(cursor.getString(1));
            subject.setText(cursor.getString(2));
            content.setText(cursor.getString(3));
        }
    }
    public void mClick(View v){
        switch(v.getId()){
        case R.id.bntCal:
            String date=wdate.getText().toString();
            mYear = Integer.parseInt(date.substring(0, 4));
            mMonth = Integer.parseInt(date.substring(5, 7));
            mDay = Integer.parseInt(date.substring(8, 10));
            new DatePickerDialog(this, mDate, mYear, mMonth-1, mDay).show();
            break;
        case R.id.btnSave:
            String strWdate=wdate.getText().toString();
            String strSubject=subject.getText().toString();
            String strContent=content.getText().toString();

            String sql="update diary set ";
            sql=sql + "wdate='" + strWdate + "',";
            sql=sql + "subject='" + strSubject + "',";
            sql=sql + "content='" + strContent + "' ";
            sql=sql + "where _id=" + id;
            db.execSQL(sql);
            Toast.makeText(this, "수정되었습니다.", 0).show();
            finish();
            break;
        case R.id.btnDelete:
            AlertDialog.Builder box=new AlertDialog.Builder(this);
            box.setMessage("삭제하시겠습니까?");
            box.setPositiveButton("예", new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface arg0, int arg1) {
                    String sql="delete from diary where _id=" + id;
                    db.execSQL(sql);
                    Toast.makeText(Update.this, "삭제되었습니다.", 0).show();
                    finish();
                }
            });
            box.setNegativeButton("아니오", null);
            box.show();
            break;
        case R.id.btnCancel:
            finish();
        }
    }
    DatePickerDialog.OnDateSetListener mDate=new DatePickerDialog.OnDateSetListener() {
        public void onDateSet(DatePicker view, int year, int month, int day) {
            mYear=year;
            mMonth=month + 1;
            mDay=day;
            wdate.setText(String.format("%04d/%02d/%02d", mYear, mMonth, mDay));
        }
    };
}
```

AndroidMenifest.xml

```
<application>
    ...
    <activity android:name=".Input"/>
    <activity android:name=".Update"/>
    ...
</application>
```

1-4. NaverAPI(뉴스검색) 예제 프로그램

API28 이후에서 Http에 접근하려면 cleartext HTTP를 활성화 시켜야한다.
<application>속성에 android:usesCleartextTraffic="true" 추가

activity_main.xml

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="0px"
        android:layout_weight="1"
        android:orientation="horizontal">
        <EditText
            android:id="@+id/query"
            android:layout_width="0px"
            android:layout_height="wrap_content"
            android:layout_weight="5"
            android:text="커피"/>
        <Button
            android:id="@+id/search"
            android:layout_width="0px"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="검색"
            android:onClick="mClick"/>
    </LinearLayout>
    <ListView
        android:id="@+id/list"
        android:layout_width="match_parent"
        android:layout_height="0px"
        android:layout_weight="10"/>
    <Button
        android:id="@+id/more"
        android:layout_width="match_parent"
        android:layout_height="0px"
        android:layout_weight="1"
        android:text="더보기"
        android:onClick="mClick"/>
</LinearLayout>
```

item.xml

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:padding="10sp">
    <TextView
        android:id="@+id/title"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:singleLine="true"
        android:textColor="#FF0000"/>
    <TextView
        android:id="@+id/description"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:singleLine="true"
        android:layout_below="@id/title"/>
</RelativeLayout>
```

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="10sp" >
    <TextView
        android:id="@+id/title"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="15sp"/>
    <WebView
        android:id="@+id/web"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>
</LinearLayout>
```

```java
public class MainActivity extends Activity {
    ArrayList<HashMap<String, String>> array;
    EditText query;
    ListView list;
    Button search, more;
    String strQuery="커피";
    int display=10;
    MyAdapter adapter;
    ProgressDialog progress;

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

        list=(ListView)findViewById(R.id.list);
        search=(Button)findViewById(R.id.search);
        more=(Button)findViewById(R.id.more);
        query=(EditText)findViewById(R.id.query);

        new NaverThread().execute();

        search.setOnClickListener(new View.OnClickListener() {
            public void onClick(View arg0) {
                display=10;
                strQuery=query.getText().toString();
                new NaverThread().execute();
            }
        });
        more.setOnClickListener(new View.OnClickListener() {
            public void onClick(View arg0) {
                display=display+10;
                new NaverThread().execute();
            }
        });

        list.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
                HashMap<String, String> map=array.get(position);

                Intent intent=new Intent(MainActivity.this, Detail.class);
                intent.putExtra("title", map.get("title"));
                intent.putExtra("link", map.get("link"));
                startActivity(intent);
            }
        });
    }

    //네이버 서버 접속 쓰레드
    class NaverThread extends AsyncTask<String, String, String>{
        protected String doInBackground(String... strings) {
            return connectNaver();
        }
        protected void onPreExecute() {
            progress=new ProgressDialog(MainActivity.this);
            progress.show();
            progress.setMessage("검색중입니다.");
            super.onPreExecute();
        }
```

```java
        protected void onPostExecute(String result) {
            parserJSON(result);
            adapter=new MyAdapter();
            list.setAdapter(adapter);
            progress.dismiss();
        }
    }
    //네이버 서버 접속
    private String connectNaver(){
        String clientId = "QTGGIhSFbIsoY_XSKPKG"; //애플리케이션 클라이언트 아이디값"
        String clientSecret = "bITB0GHh_B"; //애플리케이션 클라이언트 시크릿값"
        try{
            String text = URLEncoder.encode(strQuery, "UTF-8");
            String apiURL = "https://openapi.naver.com/v1/search/news.json?query="+ text + "&display=" + display; //json
            //String apiURL = "https://openapi.naver.com/v1/search/news.xml?query="+ text; //xml
            URL url = new URL(apiURL);
            HttpURLConnection con = (HttpURLConnection)url.openConnection();
            con.setRequestMethod("GET");
            con.setRequestProperty("X-Naver-Client-Id", clientId);
            con.setRequestProperty("X-Naver-Client-Secret", clientSecret);
            int responseCode = con.getResponseCode();
            BufferedReader br;
            if(responseCode==200) { // 정상 호출
                br = new BufferedReader(new InputStreamReader(con.getInputStream()));
            }else{ // 에러 발생
                br = new BufferedReader(new InputStreamReader(con.getErrorStream()));
            }
            String inputLine;
            StringBuffer response = new StringBuffer();
            while ((inputLine = br.readLine()) != null) {
                response.append(inputLine);
            }
            br.close();
            return response.toString();
        }catch(Exception e) {
            return e.toString();
        }
    }
    //JSON 파서
    public void parserJSON(String result){
        array=new ArrayList<HashMap<String, String>>();
        try{
            JSONArray jArray = new JSONObject(result).getJSONArray("items");
            for(int i=0;i<jArray.length();i++){
                JSONObject data = jArray.getJSONObject(i);
                String title=data.getString("title");
                String link=data.getString("link");
                String description=data.getString("description");

                HashMap<String, String> map=new HashMap<String, String>();
                map.put("title", title);
                map.put("link", link);
                map.put("description", description);
                array.add(map);
            }
        }catch(Exception e){ }
    }
    //MyAdapter 정의
    class MyAdapter extends BaseAdapter{
        public int getCount() {
            return array.size();
        }
        public Object getItem(int arg0) { return null; }
        public long getItemId(int arg0) { return 0; }
        public View getView(int position, View convertView, ViewGroup parent) {
            LayoutInflater inflater=getLayoutInflater();
            if(convertView==null){ convertView=inflater.inflate(R.layout.item, null); }
            HashMap<String, String> map=array.get(position);

            TextView title=(TextView)convertView.findViewById(R.id.title);
            TextView description=(TextView)convertView.findViewById(R.id.description);
            title.setText(Html.fromHtml(map.get("title")));
            description.setText(Html.fromHtml(map.get("description")));
            return view;
        }
    }
}
```

```java
public class Detail extends Activity{
    ProgressDialog progress;
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.detail);

        Intent intent=getIntent();
        String url=intent.getStringExtra("link");
        String title=intent.getStringExtra("title");

        TextView text=(TextView)findViewById(R.id.title);
        text.setText(Html.fromHtml(title));

        WebView web=(WebView)findViewById(R.id.web);

        progress=new ProgressDialog(this);
        progress.setMessage("페이지를 불러오는 중입니다.");
        progress.show();

        web.setWebViewClient(new MyWebView());
        web.getSettings().setJavaScriptEnabled(true);
        web.loadUrl(url);
    }

    public class MyWebView extends WebViewClient{
        public boolean shouldOverrideUrlLoading(WebView view, String url) {
            return super.shouldOverrideUrlLoading(view, url);
        }
        public void onPageFinished(WebView view, String url) {
            progress.dismiss();
            super.onPageFinished(view, url);
        }
    }
}
```

```xml
...
<uses-permission android:name="android.permission.INTERNET"/>
...
```
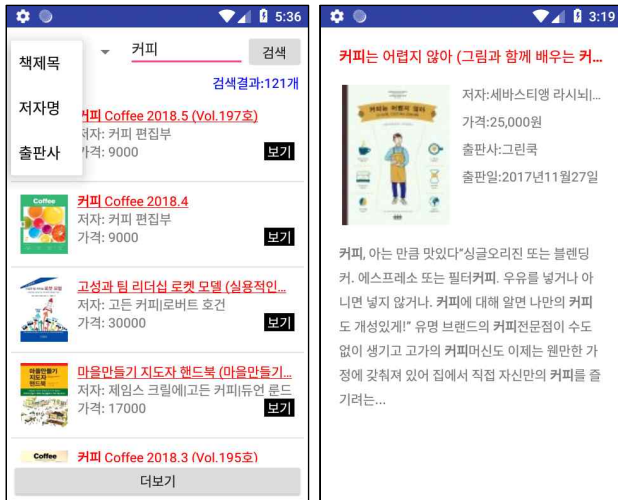
```java
Spinner spinner;
String[] enCategory=new String[]{"news.json", "blog.json", "cafearticle.json"};
String[] koCategory=new String[]{"뉴스", "블로그", "카페"};
String strCategory;

spinner=(Spinner)findViewById(R.id.spinner);
ArrayAdapter<String> adSpin=new ArrayAdapter<String>(this, android.R.layout.simple_spinner_item, koCategory);
adSpin.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
spinner.setAdapter(adSpin);

spinner.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
    public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {
        strCategory=enCategory[position];
    }
    public void onNothingSelected(AdapterView<?> parent) {
    }
});

String apiURL = "https://openapi.naver.com/v1/search/" + strCategory + "?query="+ text + "&display=" + display;
```

1-5. NaverAPI(도서검색) 예제 프로그램

activity_main.xml

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="0px"
        android:layout_weight="1"
        android:orientation="horizontal">
        <Spinner
            android:id="@+id/spin"
            android:layout_width="0px"
            android:layout_height="wrap_content"
            android:layout_weight="1" />
        <EditText
            android:id="@+id/query"
            android:layout_width="0px"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="커피" />
        <Button
            android:id="@+id/search"
            android:layout_width="0px"
            android:layout_height="wrap_content"
            android:layout_weight="0.5"
            android:text="검색"
            android:onClick="mClick"/>
    </LinearLayout>
    <TextView
        android:id="@+id/total"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="right"
        android:layout_margin="5sp"
        android:text="검색결과"/>
    <ListView
        android:id="@+id/list"
        android:layout_width="match_parent"
        android:layout_height="0px"
        android:layout_weight="10"/>
    <Button
        android:id="@+id/more"
        android:layout_width="match_parent"
        android:layout_height="0px"
        android:layout_weight="1"
        android:text="더보기"
        android:onClick="mClick"/>
</LinearLayout>
```

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:padding="10sp">
        <ImageView
            android:id="@+id/image"
            android:layout_width="50sp"
            android:layout_height="70sp"
            android:src="@drawable/ic_launcher_background"
            android:layout_marginRight="10sp"/>
        <TextView
            android:id="@+id/subject"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_toRightOf="@id/image"
            android:singleLine="true"
            android:textColor="#FF0000"
            android:text="커피"/>
        <TextView
            android:id="@+id/author"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_toRightOf="@id/image"
            android:layout_below="@id/subject"
            android:text="저자" />
        <TextView
            android:id="@+id/price"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_toRightOf="@id/image"
            android:layout_below="@id/author"
            android:text="가격" />
        <TextView
            android:id="@+id/description"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_below="@id/image"
            android:text="내용입니다."/>
        <TextView
            android:id="@+id/btnView"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_alignParentRight="true"
            android:layout_alignBaseline="@id/price"
            android:text=" 보기 "
            android:background="#000000"
            android:textColor="#FFFFFF"/>
</RelativeLayout>
```

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center_horizontal"
    android:padding="20sp">
    <TextView
        android:id="@+id/subject"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:singleLine="true"
        android:textSize="15sp"
        android:layout_marginBottom="20sp"
        android:textColor="#FF0000"/>
    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <ImageView
            android:id="@+id/image"
            android:layout_width="120sp"
            android:layout_height="150sp"
            android:layout_marginRight="10sp"
            android:src="@drawable/ic_launcher_background"/>
        <TextView
            android:id="@+id/author"
```

```xml
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_toRightOf="@id/image"
                android:layout_marginBottom="10sp"
                android:singleLine="true"
                android:text="저자" />
            <TextView
                android:id="@+id/price"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_toRightOf="@id/image"
                android:layout_below="@id/author"
                android:layout_marginBottom="10sp"
                android:text="가격" />
            <TextView
                android:id="@+id/publisher"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_toRightOf="@id/image"
                android:layout_below="@id/price"
                android:layout_marginBottom="10sp"
                android:text="출판사" />
            <TextView
                android:id="@+id/pubdate"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_toRightOf="@id/image"
                android:layout_below="@id/publisher"
                android:text="출판일" />
    </RelativeLayout>
    <TextView
        android:id="@+id/description"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20sp"
        android:lineSpacingExtra="10dip"
        android:text="내용입니다." />
</LinearLayout>
```

strings.xml

```xml
<string-array name="ko_book">
    <item>책제목</item>
    <item>저자명</item>
    <item>출판사</item>
</string-array>
<string-array name="en_book">
    <item>d_titl</item>
    <item>d_auth</item>
    <item>d_publ</item>
</string-array>
```

MainActivity.java

```java
public class MainActivity extends Activity {
    ArrayList<HashMap<String, String>> array;
    Spinner spin;
    EditText query;
    Button search, more;
    ListView list;
    MyAdapter adapter;
    String strQuery="커피", strKey="d_titl";
    int display=5, intTotal, clickedPosition=-1;

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

        spin=(Spinner)findViewById(R.id.spin);
        query=(EditText)findViewById(R.id.query);
        search=(Button)findViewById(R.id.search);
        list=(ListView)findViewById(R.id.list);
        more=(Button)findViewById(R.id.more);
        total=(TextView)findViewById(R.id.total);
```

```java
        ArrayAdapter<CharSequence> adSpin=
                ArrayAdapter.createFromResource(this, R.array.ko_book, android.R.layout.simple_spinner_item);
        adSpin.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
        spin.setAdapter(adSpin);
        spin.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
            public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {
                String[] book=getResources().getStringArray(R.array.en_book);
                strKey=book[position];
            }
            public void onNothingSelected(AdapterView<?> parent) { }
        });
        search.setOnClickListener(new View.OnClickListener(){
            public void onClick(View v) {
                display=5;
                clickedPosition=-1;
                strQuery=query.getText().toString();
                new ThreadNaver().execute();
            }
        });
        more.setOnClickListener(new OnClickListener() {
            public void onClick(View v) {
                display=display + 5;
                new ThreadNaver().execute();
            }
        });
        new ThreadNaver().execute();
}

//네이버 서버 접속
private String connectNaver(){
    String clientId = "QTGGlhSFbIsoY_XSKPKG"; //애플리케이션 클라이언트 아이디값"
    String clientSecret = "bITB0GHh_B"; //애플리케이션 클라이언트 시크릿값"
    try {
        String text = URLEncoder.encode(strQuery, "UTF-8");
        String apiURL = "https://openapi.naver.com/v1/search/book_adv.xml?";
        apiURL=apiURL + strKey + "=" + text;
        apiURL=apiURL + "&display=" + display;
        URL url = new URL(apiURL);
        HttpURLConnection con = (HttpURLConnection)url.openConnection();
        con.setRequestMethod("GET");
        con.setRequestProperty("X-Naver-Client-Id", clientId);
        con.setRequestProperty("X-Naver-Client-Secret", clientSecret);
        int responseCode = con.getResponseCode();
        BufferedReader br;
        if(responseCode==200) { // 정상 호출
            br = new BufferedReader(new InputStreamReader(con.getInputStream()));
        }else{ // 에러 발생
            br = new BufferedReader(new InputStreamReader(con.getErrorStream()));
        }
        String inputLine;
        StringBuffer response = new StringBuffer();
        while ((inputLine = br.readLine()) != null) {
            response.append(inputLine);
        }
        br.close();
        System.out.println(response.toString());
        return response.toString();
    }catch(Exception e) {
        System.out.println(e);
        return e.toString();
    }
}

//네이버서버 접속 쓰레드
class ThreadNaver extends AsyncTask<String, String, String>{
    protected String doInBackground(String... strings) {
        return connectNaver();
    }

    protected void onPreExecute() {
        progress=new ProgressDialog(MainActivity.this);
        progress.setMessage("검색중입니다.");
        progress.show();
    }
```

```java
        protected void onPostExecute(String result) {
            progress.dismiss();
            parserXML(result);

            adapter=new MyAdapter();
            list.setAdapter(adapter);
            total.setText("검색결과:" + intTotal + "개");
        }
    }

//XML 파서
public void parserXML(String result){
    array=new ArrayList<HashMap<String, String>>();
    try{
        DocumentBuilderFactory factory=DocumentBuilderFactory.newInstance();
        DocumentBuilder builder=factory.newDocumentBuilder();

        InputStream is = new ByteArrayInputStream(result.getBytes("utf-8"));
        Document document=builder.parse(is);
        Element element=document.getDocumentElement();

        NodeList total=element.getElementsByTagName("total");
        String strTotal=total.item(0).getTextContent();
        intTotal=Integer.parseInt(strTotal);

        NodeList records=element.getElementsByTagName("item");
        for(int i=0; i<records.getLength(); i++){
            Node record=records.item(i);
            NodeList fields=record.getChildNodes();

            HashMap<String, String> map=new HashMap<String, String>();
            for(int j=0; j<fields.getLength(); j++){
                Node field=fields.item(j);
                if(field.getNodeName().equals("title")){
                    String title=field.getTextContent();
                    map.put("title", title);
                }else if(field.getNodeName().equals("author")){
                    String author=field.getTextContent();
                    map.put("author", author);
                }else if(field.getNodeName().equals("price")){
                    String price=field.getTextContent();
                    map.put("price", price);
                }else if(field.getNodeName().equals("image")){
                    String image=field.getTextContent();
                    map.put("image", image);
                }else if(field.getNodeName().equals("description")){
                    String description=field.getTextContent();
                    map.put("description", description);
                }else if(field.getNodeName().equals("publisher")){
                    String publisher=field.getTextContent();
                    map.put("publisher", publisher);
                }else if(field.getNodeName().equals("pubdate")){
                    String pubdate=field.getTextContent();
                    map.put("pubdate", pubdate);
                }
            }
            array.add(map);
        }
    }catch(Exception e){ }
}

//MyAdapter 정의
class MyAdapter extends BaseAdapter{
    public int getCount() { return array.size(); }
    public Object getItem(int position) { return null; }
    public long getItemId(int position) { return 0; }

    public View getView(final int position, View convertView, ViewGroup parent) {
        LayoutInflater inflater=getLayoutInflater();
        if(convertView==null){
            convertView=inflater.inflate(R.layout.item, null);
        }
```

```java
            final HashMap<String, String> map=array.get(position);

            ImageView image=(ImageView)convertView.findViewById(R.id.image);
            TextView title=(TextView)convertView.findViewById(R.id.subject);
            TextView author=(TextView)convertView.findViewById(R.id.author);
            TextView price=(TextView)convertView.findViewById(R.id.price);
            final TextView description=(TextView)convertView.findViewById(R.id.description);
            TextView btnView=(TextView)convertView.findViewById(R.id.btnView);

            title.setText(Html.fromHtml("<u>" + map.get("title") + "</u>"));
            author.setText("저자: " + Html.fromHtml(map.get("author")));
            price.setText("가격: " + map.get("price"));
            description.setText(Html.fromHtml(map.get("description")));

            new ThreadImage(image, map.get("image")).execute();

            btnView.setOnClickListener(new OnClickListener() {
                public void onClick(View v) {
                    Intent intent=new Intent(MainActivity.this, Detail.class);

                    intent.putExtra("title", map.get("title"));
                    intent.putExtra("author", map.get("author"));
                    intent.putExtra("image", map.get("image"));
                    intent.putExtra("price", map.get("price"));
                    intent.putExtra("publisher", map.get("publisher"));
                    intent.putExtra("pubdate", map.get("pubdate"));
                    intent.putExtra("description", map.get("description"));
                    startActivity(intent);
                }
            });
            convertView.setOnClickListener(new OnClickListener(){
                public void onClick(View v) {
                    if(clickedPosition==position){
                        clickedPosition=-1;
                    }else{
                        clickedPosition=position;
                    }
                    adapter.notifyDataSetChanged();
                }
            });
            if(clickedPosition==position){
                description.setVisibility(View.VISIBLE);
            }else{
                description.setVisibility(View.GONE);
            }
            return convertView;
        }
    }

    //이미지 쓰레드
    class ThreadImage extends AsyncTask<String, String, Bitmap>{
        ImageView image;
        String url;

        public ThreadImage(ImageView image, String url) {
            this.image = image;
            this.url = url;
        }

        protected Bitmap doInBackground(String... strings) {
            Bitmap bitmap=null;
            try{
                InputStream is=new URL(url).openStream();
                bitmap= BitmapFactory.decodeStream(is);
                is.close();
            }catch(Exception e){ }
            return bitmap;
        }

        protected void onPostExecute(Bitmap result) {
            image.setImageBitmap(result);
        }
    }
}
```

```java
public class Detail extends Activity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.detail);

        Intent intent=getIntent();
        String strSubject=intent.getStringExtra("title");
        String strAuthor=intent.getStringExtra("author");
        String strImage=intent.getStringExtra("image");

        //String을 숫자포맷으로 출력
        DecimalFormat NDF=new DecimalFormat("#,###");
        String strPrice=NDF.format(Integer.parseInt(intent.getStringExtra("price")));

        //String을 날짜포맷으로 출력
        String strPubdate=intent.getStringExtra("pubdate");
        SimpleDateFormat beforeSDF=new SimpleDateFormat("yyyyMMdd");
        SimpleDateFormat afterSDF=new SimpleDateFormat("yyyy년MM월dd일");
        try{
            Date date= beforeSDF.parse(strPubdate);
            strPubdate=afterSDF.format(date);
        }catch(Exception e){}

        String strPublisher=intent.getStringExtra("publisher");
        String strDescription=intent.getStringExtra("description");

        TextView subject=(TextView)findViewById(R.id.subject);
        TextView author=(TextView)findViewById(R.id.author);
        ImageView image=(ImageView)findViewById(R.id.image);
        TextView price=(TextView)findViewById(R.id.price);
        TextView publisher=(TextView)findViewById(R.id.publisher);
        TextView pubdate=(TextView)findViewById(R.id.pubdate);
        TextView description=(TextView)findViewById(R.id.description);

        new ThreadImage(image, strImage).execute();
        subject.setText(Html.fromHtml(strSubject));
        author.setText("저자:" + strAuthor);
        price.setText("가격:" + strPrice + "원");
        publisher.setText("출판사:" +strPublisher);
        pubdate.setText("출판일:" + strPubdate);
        description.setText(Html.fromHtml(strDescription));
    }
}
```

1-6. NaverAPI(영화검색) 예제 프로그램

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:padding="10sp">
    <ImageView
        android:id="@+id/image"
        android:layout_width="50sp"
        android:layout_height="70sp"
        android:src="@drawable/ic_launcher_background"
        android:layout_marginRight="10sp"/>
    <TextView
        android:id="@+id/subject"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_toRightOf="@id/image"
        android:singleLine="true"
        android:textColor="#FF0000"
        android:text="커피"/>
    <TextView
        android:id="@+id/director"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_toRightOf="@id/image"
        android:layout_below="@id/subject"
        android:text="감독" />
    <TextView
        android:id="@+id/actor"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_toRightOf="@id/image"
        android:layout_below="@id/director"
        android:text="주연" />
    <RatingBar
        style="?android:attr/ratingBarStyleSmall"
        android:id="@+id/rating"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_toRightOf="@id/image"
        android:layout_below="@id/actor"
        android:numStars="10"
        android:stepSize="0.01"
        android:visibility="gone"/>
    <TextView
        android:id="@+id/btnView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentRight="true"
        android:layout_alignBaseline="@id/rating"
        android:text=" 별점 "
        android:background="#000000"
        android:textColor="#FFFFFF"/>
</RelativeLayout>
```

```xml
<string-array name="genre_name">
    <item>드라마</item>
    <item>액션</item>
    <item>코미디</item>
    <item>공포</item>
    <item>로맨스</item>
    <item>모험</item>
    <item>판타지</item>
</string-array>
<string-array name="genre_code">
    <item>1</item>
    <item>19</item>
    <item>11</item>
    <item>4</item>
    <item>5</item>
    <item>6</item>
    <item>2</item>
</string-array>
```

```java
public class MainActivity extends Activity {
    ArrayList<HashMap<String, String>> array;
    Spinner spin;
    EditText query;
    Button search, more;
    ListView list;
    TextView total;
    MyAdapter adapter;
    String strQuery="커피";
    int display=5;
    int intTotal;
    String strKey;
    int clickedPosition=-1;
    ProgressDialog progress;

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

        spin=(Spinner)findViewById(R.id.spin);
        query=(EditText)findViewById(R.id.query);
        search=(Button)findViewById(R.id.search);
        list=(ListView)findViewById(R.id.list);
        more=(Button)findViewById(R.id.more);
        total=(TextView)findViewById(R.id.total);

        ArrayAdapter adSpin=
                ArrayAdapter.createFromResource(this, R.array.genre_name, android.R.layout.simple_spinner_item);
        adSpin.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
        spin.setAdapter(adSpin);

        spin.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
            public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {
                String[] code= getResources().getStringArray(R.array.genre_code);
                strKey=code[position];
            }
            public void onNothingSelected(AdapterView<?> prent) { }
        });
        search.setOnClickListener(new View.OnClickListener(){
            public void onClick(View v) {
                display=5;
                clickedPosition=-1;
                strQuery=query.getText().toString();
                new ThreadNaver().execute();
            }
        });
        more.setOnClickListener(new OnClickListener() {
            public void onClick(View v) {
                display=display + 5;
                new ThreadNaver().execute();
            }
        });
        new ThreadNaver().execute();
    }
```

```java
//네이버 서버 접속
private String connectNaver(){
    String clientId = "QTGGlhSFbIsoY_XSKPKG"; //애플리케이션 클라이언트 아이디값"
    String clientSecret = "bITB0GHh_B"; //애플리케이션 클라이언트 시크릿값"
    try{
        String text = URLEncoder.encode(strQuery, "UTF-8");
        String apiURL = "https://openapi.naver.com/v1/search/movie.json";
        apiURL=apiURL + "?query=" + text;
        apiURL=apiURL + "&genre=" + strKey;
        apiURL=apiURL + "&display=" + display;
        URL url = new URL(apiURL);
        HttpURLConnection con = (HttpURLConnection)url.openConnection();
        con.setRequestMethod("GET");
        con.setRequestProperty("X-Naver-Client-Id", clientId);
        con.setRequestProperty("X-Naver-Client-Secret", clientSecret);
        int responseCode = con.getResponseCode();
        BufferedReader br;
        if(responseCode==200) { // 정상 호출
            br = new BufferedReader(new InputStreamReader(con.getInputStream()));
        }else{ // 에러 발생
            br = new BufferedReader(new InputStreamReader(con.getErrorStream()));
        }
        String inputLine;
        StringBuffer response = new StringBuffer();
        while ((inputLine = br.readLine()) != null) {
            response.append(inputLine);
        }
        br.close();
        System.out.println(response.toString());
        return response.toString();
    }catch(Exception e) {
        System.out.println(e.toString());
        return e.toString();
    }
}
//네이버서버 접속 쓰레드
class ThreadNaver extends AsyncTask<String, Integer, String>{
    protected String doInBackground(String... strings) {
        return connectNaver();
    }
    protected void onPreExecute() {
        progress=new ProgressDialog(MainActivity.this);
        progress.setMessage("검색중입니다.");
        progress.show();
    }
    protected void onPostExecute(String result) {
        progress.dismiss();
        parserJSON(result);
        adapter=new MyAdapter();
        list.setAdapter(adapter);
        total.setText("검색결과:" + intTotal + "개");
    }
}
//JSON 파서
public void parserJSON(String result){
    array=new ArrayList<HashMap<String, String>>();
    try{
        JSONObject object=new JSONObject(result);
        String strTotal=object.getString("total");
        intTotal=Integer.parseInt(strTotal);
        JSONArray jArray = object.getJSONArray("items");
        for(int i=0;i<jArray.length();i++){
            JSONObject data = jArray.getJSONObject(i);
            String title=data.getString("title");
            String director=data.getString("director");
            String actor=data.getString("actor");
            String image=data.getString("image");
            String rating=data.getString("userRating");
            HashMap<String, String> map=new HashMap<String, String>();
            map.put("title", title);
            map.put("director", director);
            map.put("actor", actor);
            map.put("image", image);
            map.put("rating", rating);
            array.add(map);
        }
    }catch(Exception e){ }
}
```

```
//MyAdapter 정의
class MyAdapter extends BaseAdapter{
    public int getCount() { return array.size(); }
    public Object getItem(int position) { return null; }
    public long getItemId(int position) { return 0; }

    public View getView(final int position, View convertView, ViewGroup parent) {
        LayoutInflater inflater=getLayoutInflater();
        if(convertView==null){
            convertView=inflater.inflate(R.layout.item, null);
        }

        final HashMap<String, String> map=array.get(position);

        ImageView image=(ImageView)convertView.findViewById(R.id.image);
        TextView title=(TextView)convertView.findViewById(R.id.subject);
        TextView director=(TextView)convertView.findViewById(R.id.director);
        TextView actor=(TextView)convertView.findViewById(R.id.actor);
        RatingBar rating=(RatingBar)convertView.findViewById(R.id.rating);
        TextView btnView=(TextView)convertView.findViewById(R.id.btnView);

        title.setText(Html.fromHtml("<u>" + map.get("title") + "</u>"));
        director.setText("저자: " + Html.fromHtml(map.get("director")));
        actor.setText("주연: "  + Html.fromHtml(map.get("actor")));
        rating.setRating(Float.parseFloat(map.get("rating")));
        new ThreadImage(image, map.get("image")).execute();

        btnView.setOnClickListener(new OnClickListener() {
            public void onClick(View v) {
                if(clickedPosition==position){
                    clickedPosition=-1;
                }else{
                    clickedPosition=position;
                }
                adapter.notifyDataSetChanged();
            }
        });

        if(clickedPosition==position){
            rating.setVisibility(View.VISIBLE);
        }else{
            rating.setVisibility(View.GONE);
        }
        return convertView;
    }
}

//이미지 쓰레드
class ThreadImage extends AsyncTask<String, String, Bitmap>{
    ImageView image;
    String url;
    public ThreadImage(ImageView image, String url) {
        this.image = image;
        this.url = url;
    }
    protected Bitmap doInBackground(String... strings) {
        Bitmap bitmap=null;
        try{
            InputStream is=new URL(url).openStream();
            bitmap= BitmapFactory.decodeStream(is);
            is.close();
        }catch(Exception e){ }
        return bitmap;
    }
    protected void onPostExecute(Bitmap result) {
        image.setImageBitmap(result);
    }
}
}
```
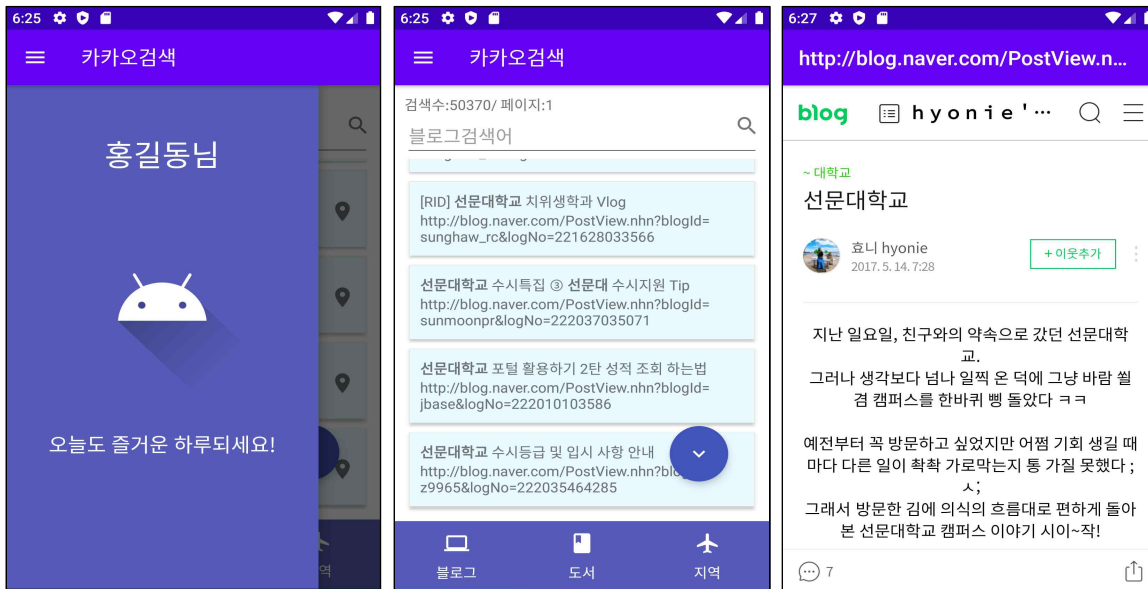
1-7. 카카오 검색 API

- 블로그검색 프로그램

AndroidMainfest.xml

```xml
<uses-permission android:name="android.permission.INTERNET" />
<application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/AppTheme"
    android:usesClearTextTraffic="true">
...
```

---

build.gradle (Module:app)

```
implementation 'com.android.support:design:30.0.0'
implementation 'com.squareup.picasso:picasso:2.5.2'
```

---

activity_main.xml

```xml
<androidx.drawerlayout.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:id="@+id/drawerLayout">
    <include layout="@layout/drawer_view"/>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">
        <androidx.viewpager.widget.ViewPager
            android:id="@+id/pager"
            android:layout_width="match_parent"
            android:layout_height="0sp"
            android:layout_weight="1"/>
        <com.google.android.material.tabs.TabLayout
            android:id="@+id/tab"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="#5559b4"
            app:tabTextColor="#ffffff"/>
    </LinearLayout>
</androidx.drawerlayout.widget.DrawerLayout>
```

---

drawer_view.xml

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="300sp"
    android:layout_height="match_parent"
    android:background="#5559b4"
    android:orientation="vertical"
    android:id="@+id/drawerView"
    android:layout_gravity="start">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textColor="#ffffff"
        android:text="홍길동님"
        android:textSize="30sp"
        android:layout_gravity="center"
        android:layout_marginTop="50sp"/>
    <ImageView
        android:layout_width="200sp"
        android:layout_height="200sp"
        android:layout_gravity="center"
        android:src="@drawable/ic_launcher_foreground"
        android:layout_marginTop="30sp"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:textColor="#ffffff"
        android:textSize="20sp"
        android:text="오늘도 즐거운 하루되세요!"
        android:layout_marginTop="30sp"/>
</LinearLayout>
```

fragment_blog.xml

```xml
<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"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    tools:context=".BlogFragment"
    android:padding="10sp">
    <TextView
        android:id="@+id/txtTotal"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="검색수:"/>
    <EditText
        android:id="@+id/edtSearch"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="검색어"
        android:maxLines="1"
        android:layout_below="@id/txtTotal"/>
    <ImageView
        android:id="@+id/btnSearch"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/ic_search"
        android:layout_alignParentRight="true"
        android:layout_below="@id/txtTotal"/>
    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/btnMore"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/ic_more"
        android:layout_alignParentBottom="true"
        android:layout_alignParentRight="true"
        android:layout_margin="30sp"
        android:backgroundTint="#3F51B5"
        app:borderWidth="0sp"/>
    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/listBlog"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_below="@id/edtSearch"/>
</RelativeLayout>
```

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#E3F3FA"
    android:padding="10sp"
    android:layout_margin="5sp"
    android:orientation="vertical"
    android:elevation="3sp"
    android:id="@+id/layout">
    <TextView
        android:id="@+id/title"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="제목이 들어갑니다............"/>
    <TextView
        android:id="@+id/link"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="URL주소가 들어갑니다..........."/>
</LinearLayout>
```

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".WebActivity">
    <WebView
        android:id="@+id/web"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>
</LinearLayout>
```

```java
public class Kakao {
    public static String connect(String apiURL) {
        try {
            URL url = new URL(apiURL);
            HttpURLConnection con = (HttpURLConnection)url.openConnection();
            con.setRequestMethod("GET");
            con.setRequestProperty("Authorization", "KakaoAK 4c9947b8b0448265ca15036c39a7fa4e");
            int responseCode = con.getResponseCode();
            BufferedReader br;

            if(responseCode==200) {   // 정상 호출인 경우
                br = new BufferedReader(new InputStreamReader(con.getInputStream(), "UTF-8"));
            } else {   //에러 발생가 발생한 경우
                br = new BufferedReader(new InputStreamReader(con.getErrorStream()));
            }

            String inputLine;
            StringBuffer response = new StringBuffer();
            while ((inputLine = br.readLine()) != null) {
                response.append(inputLine);
            }
            br.close();
            //System.out.println(response.toString());
            return response.toString();
        }catch (Exception e) {
            return e.toString();
        }
    }
}
```

```java
public class MainActivity extends AppCompatActivity {
    DrawerLayout drawerLayout;
    LinearLayout drawerView;
    TabLayout tab;
    ViewPager pager;
    ArrayList<Fragment> array;

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

        getSupportActionBar().setTitle("카카오검색");
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
        getSupportActionBar().setHomeAsUpIndicator(R.drawable.ic_menu);

        drawerLayout=findViewById(R.id.drawerLayout);
        drawerView=findViewById(R.id.drawerView);
        tab=findViewById(R.id.tab);
        pager=findViewById(R.id.pager);

        tab.addTab(tab.newTab().setText("블로그"));
        tab.addTab(tab.newTab().setText("도서"));
        tab.addTab(tab.newTab().setText("지역"));

        tab.getTabAt(0).setIcon(R.drawable.ic_blog);
        tab.getTabAt(1).setIcon(R.drawable.ic_baseline_book);
        tab.getTabAt(2).setIcon(R.drawable.ic_local);

        PagerAdapter ad=new PagerAdapter(getSupportFragmentManager());
        pager.setAdapter(ad);

        pager.addOnPageChangeListener(new TabLayout.TabLayoutOnPageChangeListener(tab));
        tab.addOnTabSelectedListener(new TabLayout.OnTabSelectedListener() {
            @Override
            public void onTabSelected(TabLayout.Tab tab) {
                pager.setCurrentItem(tab.getPosition());
            }
            @Override
            public void onTabUnselected(TabLayout.Tab tab) { }
            @Override
            public void onTabReselected(TabLayout.Tab tab) { }
        });
    }

    class PagerAdapter extends FragmentStatePagerAdapter{
        public PagerAdapter(@NonNull FragmentManager fm) {
            super(fm);
        }
        @NonNull
        @Override
        public Fragment getItem(int position) {
            switch (position){
                case 0:
                    return new BlogFragment();
                case 1:
                    return new BookFragment();
                case 2:
                    return new LocalFragment();
                default:
                    return null;
            }
        }
        @Override
        public int getCount() {
            return 3;
        }
    }
    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        switch (item.getItemId()){
            case android.R.id.home:
                drawerLayout.openDrawer(drawerView);
        }
        return super.onOptionsItemSelected(item);
    }
}
```

```java
public class BlogFragment extends Fragment {
    String url="https://dapi.kakao.com/v2/search/blog";
    String query="선문대학교";
    int page=1, total=0;
    boolean is_end;
    TextView txtTotal;
    EditText edtSearch;
    ImageView btnSearch;
    RecyclerView listBlog;
    FloatingActionButton btnMore;

    ArrayList<HashMap<String, String>> arrayBlog=new ArrayList<>();
    BlogAdapter blogAdapter=new BlogAdapter();

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {
        View view=inflater.inflate(R.layout.fragment_blog, container, false);

        edtSearch=view.findViewById(R.id.edtSearch);
        edtSearch.setHint("블로그검색어");
        txtTotal=view.findViewById(R.id.txtTotal);

        listBlog=view.findViewById(R.id.listBlog);
        listBlog.setLayoutManager(new LinearLayoutManager(getActivity()));
        listBlog.setAdapter(blogAdapter);

        new KakaoThread().execute();

        btnMore=view.findViewById(R.id.btnMore);
        btnMore.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if(!is_end) {                    page = page + 1;
                    new KakaoThread().execute();
                }else{
                    Toast.makeText(getActivity(),"마지막페이지입니다.", Toast.LENGTH_SHORT).show();
                }
            }
        });

        ImageView btnSearch=view.findViewById(R.id.btnSearch);
        btnSearch.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                query=edtSearch.getText().toString();
                page=page+1;
                new KakaoThread().execute();
            }
        });
        return view;
    }

    class KakaoThread extends AsyncTask<String,String,String>{  //카카오 스레드
        @Override
        protected String doInBackground(String... strings) {
            String s=Kakao.connect(url+"?query="+query+"&page="+page);
            return s;
        }

        @Override
        protected void onPostExecute(String s) {
            super.onPostExecute(s);
            blogParser(s);
            blogAdapter.notifyDataSetChanged();
            listBlog.scrollToPosition(arrayBlog.size()-1);
            txtTotal.setText("검색수:" + total + "/ 페이지:" + page);
        }
    }

    public void blogParser(String s){  //데이타 파서
        try{
            JSONObject jObject=new JSONObject(s).getJSONObject("meta");
            total=jObject.getInt("total_count");
            is_end=jObject.getBoolean("is_end");
            JSONArray jArray=new JSONObject(s).getJSONArray("documents");

            for(int i=0; i<jArray.length(); i++){
```

```java
                JSONObject obj=jArray.getJSONObject(i);
                HashMap<String,String> map=new HashMap<>();
                map.put("title", obj.getString("title"));
                map.put("link", obj.getString("url"));
                arrayBlog.add(map);
            }
        }catch(Exception e){ System.out.pringln(e.toString(); }
    }

    class BlogAdapter extends RecyclerView.Adapter<BlogAdapter.ViewHolder>{   //블로그 어댑터
        @NonNull
        @Override
        public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
            View view=getActivity().getLayoutInflater().inflate(R.layout.item_blog,parent,false);
            return new ViewHolder(view);
        }

        @Override
        public void onBindViewHolder(@NonNull ViewHolder holder, int position) {
            final HashMap<String,String> map=arrayBlog.get(position);
            holder.title.setText(Html.fromHtml(map.get("title")));
            holder.link.setText(map.get("link"));
            holder.layout.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    Intent intent=new Intent(getActivity(), WebActivity.class);
                    intent.putExtra("link", map.get("link"));
                    intent.putExtra("title", map.get("title"));
                    startActivity(intent);
                }
            });
        }

        @Override
        public int getItemCount() { return arrayBlog.size(); }

        public class ViewHolder extends RecyclerView.ViewHolder {
            TextView title,link;
            LinearLayout layout;
            public ViewHolder(@NonNull View itemView) {
                super(itemView);
                layout=itemView.findViewById(R.id.layout);
                title=itemView.findViewById(R.id.title);
                link=itemView.findViewById(R.id.link);
            }
        }
    }
}
```

WebActivity.java

```java
public class WebActivity extends AppCompatActivity {
    WebSettings webSettings;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_web);

        Intent intent=getIntent();
        String title= intent.getStringExtra("title");
        String url=intent.getStringExtra("link");

        getSupportActionBar().setTitle(Html.fromHtml(url));

        WebView web=findViewById(R.id.web);
        web.setWebViewClient(new MyWebView());
        webSettings=web.getSettings();
        webSettings.setJavaScriptEnabled(true);
        web.loadUrl(url);
    }
    class MyWebView extends WebViewClient{
        @Override
        public boolean shouldOverrideUrlLoading(WebView view, String url) {
            return super.shouldOverrideUrlLoading(view, url);
        }
    }
}
```
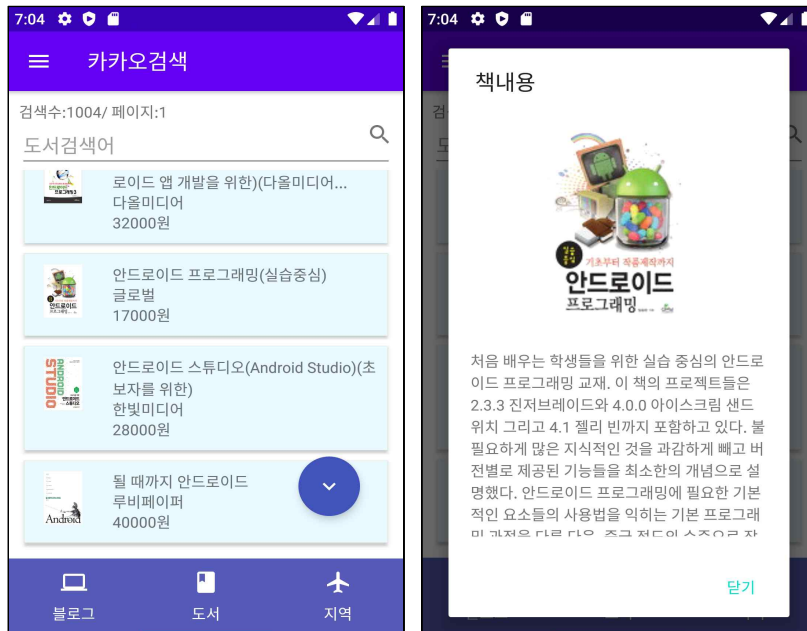
- **도서검색 프로그램**

fragment_book.xml

```
...
<androidx.recyclerview.widget.RecyclerView
    android:id="@+id/listBook"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_below="@id/edtSearch"/>
</RelativeLayout>
```

item_book.xml

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#E3F3FA"
    android:padding="10sp"
    android:layout_margin="5sp"
    android:elevation="3sp">
    <ImageView
        android:id="@+id/thumbnail"
        android:layout_width="50sp"
        android:layout_height="50sp"
        android:src="@mipmap/ic_launcher"
        android:layout_marginRight="20sp"/>
    <TextView
        android:id="@+id/title"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="제목이 들어갑니다............"
        android:layout_toRightOf="@id/thumbnail"/>
    <TextView
        android:id="@+id/publisher"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="출판사 들어갑니다............"
        android:layout_toRightOf="@id/thumbnail"
        android:layout_below="@id/title"/>
    <TextView
        android:id="@+id/price"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="가격이 들어갑니다............"
        android:layout_toRightOf="@id/thumbnail"
        android:layout_below="@id/publisher"/>
</RelativeLayout>
```
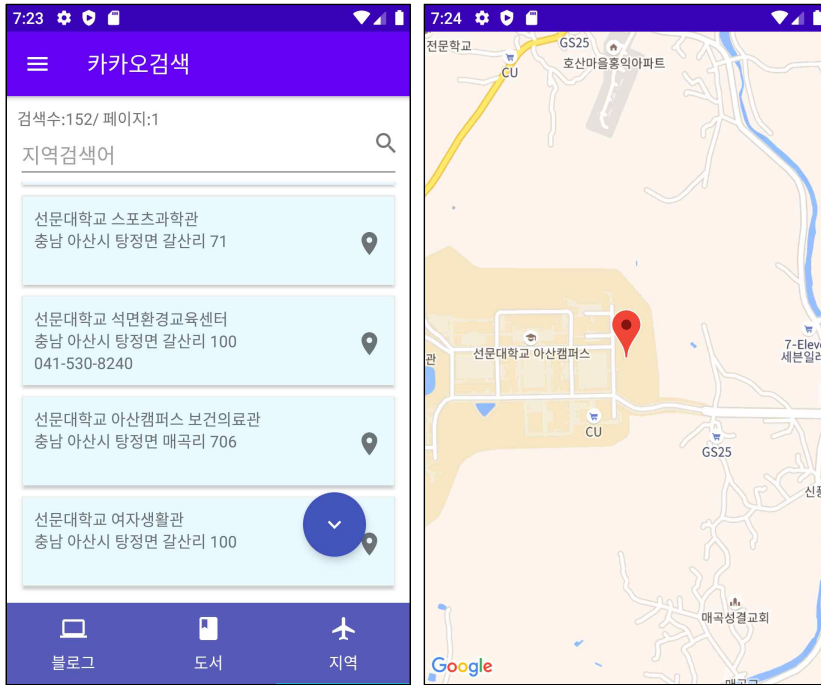
```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="20sp"
    android:orientation="vertical">
    <ImageView
        android:id="@+id/thumbnail"
        android:layout_width="200sp"
        android:layout_height="200sp"
        android:layout_gravity="center"
        android:layout_marginBottom="20sp"/>
    <TextView
        android:id="@+id/description"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
</LinearLayout>
```

```java
public class BookFragment extends Fragment {
    String url="https://dapi.kakao.com/v3/search/book?target=title";
    String query="안드로이드";
    ...
    //bookParser
    public void bookParser(String s){
        try{
            ...
            JSONArray jArray=new JSONObject(s).getJSONArray("documents");
            for(int i=0; i<jArray.length(); i++){
                JSONObject obj=jArray.getJSONObject(i);
                HashMap<String,String> map=new HashMap<>();
                map.put("title", obj.getString("title"));
                map.put("thumbnail", obj.getString("thumbnail"));
                map.put("publisher", obj.getString("publisher"));
                map.put("price", obj.getString("price"));
                map.put("description", obj.getString("contents"));
                arrayBook.add(map);
            }
        }catch(Exception e){ }
    }
    //BookAdapter
    class BookAdapter extends RecyclerView.Adapter<BookAdapter.ViewHolder>{
        @NonNull
        @Override
        public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
            View view=getActivity().getLayoutInflater().inflate(R.layout.item_book, parent, false);
            return new ViewHolder(view);
        }
        @Override
        public void onBindViewHolder(@NonNull ViewHolder holder, int position) {
            final HashMap<String, String> map=arrayBook.get(position);
            ...
            String thumbnail=map.get("thumbnail");
            if(!thumbnail.equals("")) Picasso.with(getActivity()).load(map.get("thumbnail")).into(holder.thumbnail);

            holder.thumbnail.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    LinearLayout layout=(LinearLayout)getLayoutInflater().inflate(R.layout.item_description,null);
                    TextView description=layout.findViewById(R.id.description);
                    description.setText(map.get("description"));
                    ImageView thumbnail=layout.findViewById(R.id.thumbnail);
                    Picasso.with(getActivity()).load(map.get("thumbnail")).into(thumbnail);

                    AlertDialog.Builder box=new AlertDialog.Builder(getActivity());
                    box.setTitle("책내용");
                    box.setPositiveButton("닫기", null);
                    box.setView(layout);
                    box.show();
                }
            });
        }
    ...
}
```

- **지역검색 프로그램**

```
...
<androidx.recyclerview.widget.RecyclerView
    android:id="@+id/listBook"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_below="@id/edtSearch"/>
</RelativeLayout>
```

item_local.xml

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#E3F3FA"
    android:padding="l0sp"
    android:layout_margin="5sp"
    android:elevation="3sp">
    <TextView
        android:id="@+id/name"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="장소명입니다."/>
    <TextView
        android:id="@+id/address"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="주소입니다."
        android:layout_below="@id/name"/>
    <TextView
        android:id="@+id/tel"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="전화번호입니다."
        android:layout_below="@id/address"/>
    <ImageView
        android:id="@+id/map"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/ic_location"
        android:layout_alignParentRight="true"
        android:layout_below="@id/name"/>
</RelativeLayout>
```

activity_maps.xml

```xml
<fragment xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:map="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/map"
    android:name="com.google.android.gms.maps.SupportMapFragment"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MapsActivity" />
```

LocalFragment.xml

```java
public class LocalFragment extends Fragment {
    String url="https://dapi.kakao.com/v2/local/search/keyword.json";
    String query="선문대학교";
    ...
    //데이터 파서
    public void localParser(String s){
        try{
            ...
            JSONArray jArray=new JSONObject(s).getJSONArray("documents");

            for(int i=0; i<jArray.length(); i++){
                JSONObject obj=jArray.getJSONObject(i);
                HashMap<String,String> map=new HashMap<>();
                map.put("name", obj.getString("place_name"));
                map.put("address", obj.getString("address_name"));
                map.put("tel", obj.getString("phone"));
                map.put("x", obj.getString("x"));
                map.put("y", obj.getString("y"));
                arrayLocal.add(map);
            }
        }catch(Exception e){ }
    }
}
```

MapsActivity.java

```java
public class MapsActivity extends FragmentActivity implements OnMapReadyCallback {
    private GoogleMap mMap;
    double x;
    double y;
    String title;

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

        Intent intent=getIntent();
        x = Double.parseDouble(intent.getStringExtra("x"));
        y = Double.parseDouble(intent.getStringExtra("y"));
        title = intent.getStringExtra("title");

        SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()
                .findFragmentById(R.id.map);
        mapFragment.getMapAsync(this);
    }

    @Override
    public void onMapReady(GoogleMap googleMap) {
        mMap = googleMap;

        LatLng latLng = new LatLng(y, x);
        mMap.addMarker(new MarkerOptions().position(latLng).title(title));
        mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));
        mMap.moveCamera(CameraUpdateFactory.zoomTo(15));
    }
```

1-8 크롤링 (Crawling)

```xml
<uses-permission android:name="android.permission.INTERNET"/>
<application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/AppTheme"
    android:usesCleartextTraffic="true">
    …
```
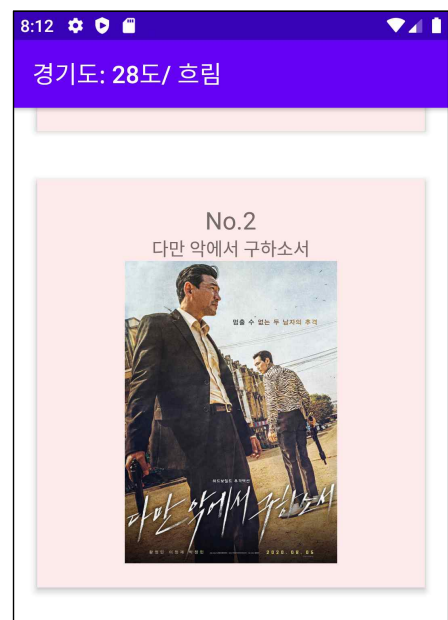
build.gradle (Module:app)

```gradle
implementation 'org.jsoup:jsoup:1.13.1'
implementation 'com.android.support:design:30.0.0'
implementation 'com.squareup.picasso:picasso:2.5.2'
…
```

activity_main.xml

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/listCVG"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>
</RelativeLayout>
```

item_cgv.xml

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="20sp"
    android:gravity="center"
    android:background="#F6E3E3"
    android:layout_margin="20sp"
    android:elevation="5sp">
    <TextView
        android:id="@+id/rank"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="여기는 순위출력"
        android:textSize="20sp"/>
    <TextView
        android:id="@+id/title"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="여기는 제목출력"
        android:textSize="15sp"/>
    <ImageView
        android:id="@+id/image"
        android:layout_width="200sp"
        android:layout_height="250sp"
        android:src="@mipmap/ic_launcher"/>
</LinearLayout>
```

```java
public class CVGAdapter extends RecyclerView.Adapter<CVGAdapter.ViewHolder> {
    Context context;
    ArrayList<HashMap<String, String>> arrayCGV;

    public CVGAdapter(Context context, ArrayList<HashMap<String, String>> arrayCGV) {
        this.context = context;
        this.arrayCGV = arrayCGV;
    }
    @NonNull
    @Override
    public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        View view= LayoutInflater.from(parent.getContext()).inflate(.layout.item_cgv, parent,false);
        return new ViewHolder(view);
    }
    @Override
    public void onBindViewHolder(@NonNull ViewHolder holder, int position) {
        holder.rank.setText(arrayCGV.get(position).get("rank"));
        holder.title.setText(arrayCGV.get(position).get("title"));
        Picasso.with(context).load(arrayCGV.get(position).get("image")).into(holder.image);
    }
    @Override
    public int getItemCount() {
        return arrayCGV.size();
    }

    public class ViewHolder extends RecyclerView.ViewHolder {
        TextView rank, title;
        ImageView image;
        public ViewHolder(@NonNull View itemView) {
            super(itemView);
            rank=itemView.findViewById(R.id.rank);
            title=itemView.findViewById(R.id.title);
            image=itemView.findViewById(R.id.image);
        }
    }
}
```

```java
public class MainActivity extends AppCompatActivity {
    ArrayList<HashMap<String,String>> arrayCGV=new ArrayList<>();
    CVGAdapter cvgAdapter;
    ArrayList<HashMap<String,String>> arrayDaum=new ArrayList<>();
    int index;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        new CVGThread().execute();
        new DaumThread().execute();
    }

    class CVGThread extends AsyncTask<String,String,String>{
        @Override
        protected String doInBackground(String... strings) {
            try{
                Document doc= Jsoup.connect("http://www.cgv.co.kr/movies").get();
                Elements elements=doc.select(".sect-movie-chart ol");
                for(Element e:elements.select("li")){
                    HashMap<String, String> map=new HashMap<String, String>();
                    map.put("rank", e.select(".rank").text());
                    map.put("title", e.select(".title").text());
                    map.put("image", e.select("img").attr("src"));
                    if(!e.select(".rank").text().equals("")) arrayCGV.add(map);
                }
            }catch(Exception e){ System.out.println(e.toString()); }
            return null;
        }
```

```java
        @Override
        protected void onPostExecute(String s) {
            super.onPostExecute(s);
            System.out.println("데이타갯수:" +arrayCGV.size());

            RecyclerView listCVG=findViewById(R.id.listCVG);
            listCVG.setLayoutManager(new LinearLayoutManager(MainActivity.this));
            cvgAdapter=new CVGAdapter(MainActivity.this, arrayCGV);
            listCVG.setAdapter(cvgAdapter);
        }
    }

    //다음 날씨 스레드
    class DaumThread extends AsyncTask<String, String, String>{
        @Override
        protected void onPostExecute(String s) {
            super.onPostExecute(s);
            System.out.println("다음날씨데이타갯수:" + arrayDaum.size());
            getSupportActionBar().setTitle("다음날씨");

            BackThread backThread=new BackThread();
            backThread.setDaemon(true);
            backThread.start();
        }

        @Override
        protected String doInBackground(String... strings) {
            try {
                HashMap<String, Object> object=new HashMap<String, Object>();
                Document doc=Jsoup.connect("http://www.daum.net").get();
                Elements elements=doc.select(".list_weather");
                for(Element e:elements.select("li")) {
                    HashMap<String, String> map=new HashMap<String, String>();
                    map.put("part", e.select(".txt_part").text());
                    map.put("temper", e.select(".txt_temper").text());
                    map.put("ico", e.select(".ico_ws").text());
                    arrayDaum.add(map);
                }
            }catch(Exception e) {
                System.out.println(e.toString());
            }
            return null;
        }
    }

    class BackThread extends Thread{
        @Override
        public void run() {
            super.run();
            index=0;
            while(true){
                handler.sendEmptyMessage(0);
                index++;
                if(index==arrayDaum.size()) index=0;
                try{ Thread.sleep(1000); }catch(Exception e){ }
            }
        }
    }

    Handler handler=new Handler(){
        @Override
        public void handleMessage(@NonNull Message msg) {
            super.handleMessage(msg);
            String part=arrayDaum.get(index).get("part");
            String temper=arrayDaum.get(index).get("temper");
            String ico=arrayDaum.get(index).get("ico");
            getSupportActionBar().setTitle(part+": "+ temper + "도/ " +ico);
        }
    };
}
```
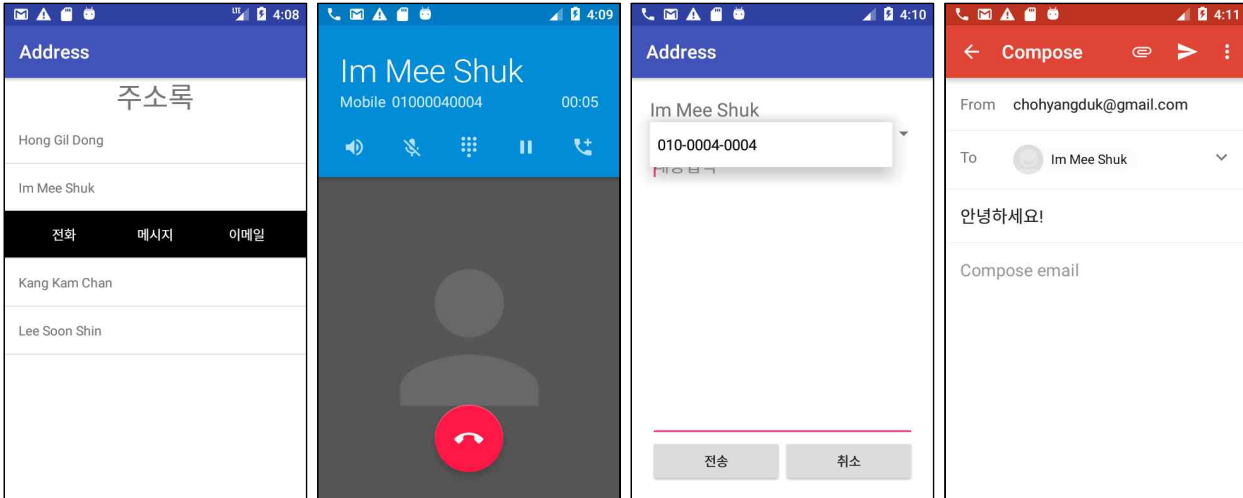
1-9 주소록 예제 프로그램

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"  android:layout_height="match_parent"
     android:orientation="vertical">
    <TextView
        android:layout_width="wrap_content"  android:layout_height="wrap_content"
        android:text="주소록"
        android:textSize="30sp"  android:layout_gravity="center"/>
    <ListView
        android:id="@+id/list"
        android:layout_width="match_parent"  android:layout_height="match_parent"/>
</LinearLayout>
```

item.xml

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"  android:layout_height="match_parent"
    android:orientation="vertical">
    <TextView
        android:id="@+id/name"
        android:layout_width="wrap_content"  android:layout_height="wrap_content"
        android:layout_margin="15sp"/>
    <LinearLayout
        android:id="@+id/content"
        android:layout_width="match_parent"  android:layout_height="wrap_content"
        android:orientation="horizontal"  android:padding="15sp"
        android:visibility="gone">
        <TextView
            android:id="@+id/phone"  android:text="전화"
            android:layout_width="0px"  android:layout_weight="1"  android:layout_height="wrap_content"
            android:textColor="#FFFFFF"  android:gravity="center"/>
        <TextView
            android:id="@+id/message"  android:text="메시지"
            android:layout_width="0px"  android:layout_weight="1"  android:layout_height="wrap_content"
            android:textColor="#FFFFFF"  android:gravity="center"/>
        <TextView
            android:id="@+id/email"  android:text="이메일"
            android:layout_width="0px"  android:layout_weight="1"  android:layout_height="wrap_content"
            android:textColor="#FFFFFF"  android:gravity="center"/>
    </LinearLayout>
</LinearLayout>
```

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"  android:layout_height="match_parent"
    android:orientation="vertical"  android:padding="20sp">
    <TextView
        android:id="@+id/name"  android:textSize="30sp"
        android:layout_width="wrap_content"  android:layout_height="wrap_content"/>
    <Spinner
        android:id="@+id/target"
        android:layout_width="match_parent"  android:layout_height="wrap_content"/>
    <EditText
        android:id="@+id/content"  android:hint="내용입력"  android:gravity="top"
        android:layout_width="match_parent"  android:layout_height="0px"  android:layout_weight="1"/>
    <LinearLayout
        android:layout_width="match_parent"  android:layout_height="wrap_content"  android:orientation="horizontal">
        <Button
            android:id="@+id/btnSend"  android:text="전송"
            android:layout_width="0px"  android:layout_weight="1"  android:layout_height="wrap_content"/>
        <Button
            android:id="@+id/btnCancel"  android:text="전송"
            android:layout_width="0px"  android:layout_weight="1"  android:layout_height="wrap_content"/>
    </LinearLayout>
</LinearLayout>
```

```java
public class PermissionActivity extends AppCompatActivity {
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_permission);

        if(Build.VERSION.SDK_INT < 23) {
            goIndexActivity();
        }
        else{
            checkAndRequestPermission();
        }
    }
    private void goIndexActivity(){  /* 인덱스 액티비티를 실행하고 현재 액티비티를 종료한다. */
        Intent intent=new Intent (this, MainActivity.class);
        startActivity(intent);
        finish();
    }

    /* 권한을 확인하고 권한이 부여되어 있지 않다면 권한을 요청하고 요청되어 있으면 인덱스 액티비티로 이동한다. */
    private void checkAndRequestPermission(){
        int intGranted = PackageManager.PERMISSION_GRANTED;
        if(ContextCompat.checkSelfPermission(this, Manifest.permission.READ_CONTACTS) != intGranted
                || ContextCompat.checkSelfPermission(this, Manifest.permission.SEND_SMS) != intGranted
                || ContextCompat.checkSelfPermission(this, Manifest.permission.CALL_PHONE) != intGranted) {

            ActivityCompat.requestPermissions(this,
                    new String[]{Manifest.permission.READ_CONTACTS,
                    Manifest.permission.SEND_SMS,
                    Manifest.permission.CALL_PHONE},100);
        }else{
            goIndexActivity();
        }
    }
    /* 권한 요청 결과를 보고 모든 권한이 승인되었을 경우에는 메인 액티비티로 이동하고 아니면 현재 액티비티를 종료한다. */
    public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {
        boolean isAllGranted = true;
        switch (requestCode){
            case 100:
                for(int i=0; i<permissions.length; i++){
                    if(grantResults[i] != PackageManager.PERMISSION_GRANTED){
                        isAllGranted=false;
                    }
                }
                if(isAllGranted){
                    goIndexActivity();
                }else{
                    Toast.makeText(this, "퍼미션을 얻지 못했습니다!", Toast.LENGTH_SHORT).show();
                    finish();
                }
                break;
        }
    }
}
```

```java
public class MainActivity extends AppCompatActivity {
    ArrayList<MyItem> array;
    MyAdapter adapter;
    int clickedPosition = -1;

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

        readContracts();
        adapter = new MyAdapter();
        ListView list = (ListView) findViewById(R.id.list);
        list.setAdapter(adapter);
    }

    //MyItem 정의
    class MyItem {
        int id;
        String name;
        ArrayList<String> phone;
        ArrayList<String> email;

        public MyItem(int id, String name, ArrayList<String> phone, ArrayList<String> email) {
            this.id = id;
            this.name = name;
            this.phone = phone;
            this.email = email;
        }
    }

    //readContracts
    private void readContracts() {
        array = new ArrayList<MyItem>();
        ContentResolver cr = getContentResolver();
        Cursor cursor = cr.query(ContactsContract.Contacts.CONTENT_URI, null, null, null,
                                ContactsContract.Contacts.DISPLAY_NAME);

        int intId = cursor.getColumnIndex(ContactsContract.Contacts._ID);
        int intName = cursor.getColumnIndex(ContactsContract.Contacts.DISPLAY_NAME);

        while (cursor.moveToNext()) {
            ArrayList<String> phone = new ArrayList<String>();
            ArrayList<String> email = new ArrayList<String>();

            String id = cursor.getString(intId);
            Cursor cursor2 = cr.query(ContactsContract.CommonDataKinds.Phone.CONTENT_URI, null,
                    ContactsContract.CommonDataKinds.Phone.CONTACT_ID + "=?", new String[]{id}, null);

            while (cursor2.moveToNext()) {
                String number =
                        cursor2.getString(cursor2.getColumnIndex(ContactsContract.CommonDataKinds.Phone.NUMBER));
                phone.add(number);
            }
            cursor2.close();

            Cursor cursor3 = cr.query(ContactsContract.CommonDataKinds.Email.CONTENT_URI, null,
                    ContactsContract.CommonDataKinds.Email.CONTACT_ID + "=?", new String[]{id}, null);
            while (cursor3.moveToNext()) {
                String mail = cursor3.getString(cursor3.getColumnIndex(ContactsContract.CommonDataKinds.Email.DATA));
                email.add(mail);
            }
            cursor3.close();

            MyItem item = new MyItem(intId, cursor.getString(intName), phone, email);
            array.add(item);
        }
        cursor.close();
    }
```

```java
//MyAdapter 정의
class MyAdapter extends BaseAdapter {
    public int getCount() { return array.size(); }
    public Object getItem(int position) { return null; }
    public long getItemId(int position) { return 0; }

    public View getView(final int position, View convertView, ViewGroup parent) {
        LayoutInflater inflater = getLayoutInflater();
        if (convertView == null) {
            convertView = inflater.inflate(R.layout.item, parent, false);
        }

        TextView name = (TextView) convertView.findViewById(R.id.name);
        TextView phone = (TextView) convertView.findViewById(R.id.phone);
        TextView message = (TextView) convertView.findViewById(R.id.message);
        TextView email = (TextView) convertView.findViewById(R.id.email);
        LinearLayout content = (LinearLayout) convertView.findViewById(R.id.content);

        name.setText(array.get(position).name);

        phone.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                if (array.get(position).phone.size() > 0) {
                    Uri uri = Uri.parse("tel:" + array.get(position).phone.get(0));
                    Intent intent = new Intent(Intent.ACTION_CALL, uri);

                    if (ActivityCompat.checkSelfPermission(MainActivity.this,
                                Manifest.permission.CALL_PHONE) != PackageManager.PERMISSION_GRANTED) {
                        return;
                    }
                    startActivity(intent);
                }
            }
        });

        message.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                Intent intent=new Intent(MainActivity.this, MessageActivity.class);
                intent.putExtra("name", array.get(position).name);
                intent.putExtra("phone", array.get(position).phone);
                startActivity(intent);
            }
        });

        email.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                if(array.get(position).email.size()>0) {
                    Uri uri = Uri.parse("mailto:" + array.get(position).email.get(0));
                    Intent intent = new Intent(Intent.ACTION_SENDTO, uri);
                    intent.putExtra(Intent.EXTRA_SUBJECT, "안녕하세요!");
                    startActivity(intent);
                }
            }
        });

        convertView.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                if(clickedPosition==position){
                    clickedPosition=-1;
                }else{
                    clickedPosition=position;
                }
                adapter.notifyDataSetChanged();
            }
        });
        if(clickedPosition==position){
            content.setVisibility(View.VISIBLE);
        }else{
            content.setVisibility(View.GONE);
        }
        return convertView;
    }
}
```

```java
public class MessageActivity extends AppCompatActivity {
    TextView name;
    Spinner spin;

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

        name=(TextView)findViewById(R.id.name);
        spin=(Spinner)findViewById(R.id.spin);

        Intent intent=getIntent();
        String strName=intent.getStringExtra("name");
        ArrayList arPhone=intent.getStringArrayListExtra("phone");

        name.setText(strName);
        ArrayAdapter adPhone = new ArrayAdapter(this, android.R.layout.simple_spinner_item, arPhone);
        adPhone.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
        spin.setAdapter(adPhone);
    }

    public void mClick(View v){
        switch(v.getId()){
            case R.id.btnSend:
                SmsManager sms=SmsManager.getDefault();

                EditText message=(EditText)this.findViewById(R.id.content);
                String strMessage=message.getText().toString();
                String phone=(String)spin.getSelectedItem();

                sms.sendTextMessage(phone, null, strMessage, null, null);
                Toast.makeText(this, "메시지를 전송했습니다.", Toast.LENGTH_SHORT).show();
                break;
            case R.id.btnCancel:
                Toast.makeText(this, "취소되었습니다.", Toast.LENGTH_SHORT).show();
        }
        Intent intent=new Intent(this, MainActivity.class);
        finish();
    }
}
```

```xml
<uses-permission android:name="android.permission.READ_CONTACTS"/>
<uses-permission android:name="android.permission.SEND_SMS"/>
<uses-permission android:name="android.permission.CALL_PHONE"/>
```

2-1. Firebase 인증 예제 프로그램

Firebase는 구글에서 웹과 모바일 개발에 필요한 기능을 제공하는 백엔드 서비스이다. 쉽게 말해 백엔드 개발을 통해 서버를 따로 설계, 구현하지 않고 프론트엔드 개발에 집중할 수 있도록 도와주는 서비스이다. 기능으로는 실시간 데이터베이스, 간편한 사용자인증, 클라우드 저장소, 호스팅, 앱 테스트와 수익 창출을 도와주는 등 다양한 기능을 제공해준다.

• **Firebase Connect**

1) Firebase 콘솔에 접속한다. (https://console.firebase.google.com/)
2) 새로운 프로젝트를 생성한다.
3) 프로젝트 안에 새로운 안드로이드앱을 생성한 후 google-services.json 파일을 다운로드 받는다.
4) 메뉴의 [Android]를 [Project]로 바꾸고 다운로드 받은 google-services.json을 [project]-[app] 아래에 복사한다.
5) 프로젝트 수준의 build.gradle의 내용을 아래와 같이 수정한다.

```
buildscript {
    repositories {
        // Check that you have the following line (if not, add it):
        google()  // Google's Maven repository
    }

    dependencies {
        ...
        // Add this line
        classpath 'com.google.gms:google-services:4.3.10'
    }
}

allprojects {
    ...
    repositories {
        // Check that you have the following line (if not, add it):
        google()  // Google's Maven repository
        ...
    }
}
```

※ 6)~7)을 설정하지 않은 경우

```
db=FirebaseDatabase.getInstance("https://positive-notch-259405-default-rtdb.firebaseio.com/");
```

6) 앱 수준의 build.gradle의 내용을 아래와 같이 수정한다.

```
apply plugin: 'com.android.application'

// Add this line
apply plugin: 'com.google.gms.google-services'

dependencies {
    // Import the Firebase BoM
    implementation platform('com.google.firebase:firebase-bom:29.0.0')
    // Add the dependencies for the desired Firebase products
    // https://firebase.google.com/docs/android/setup#available-libraries
}
```
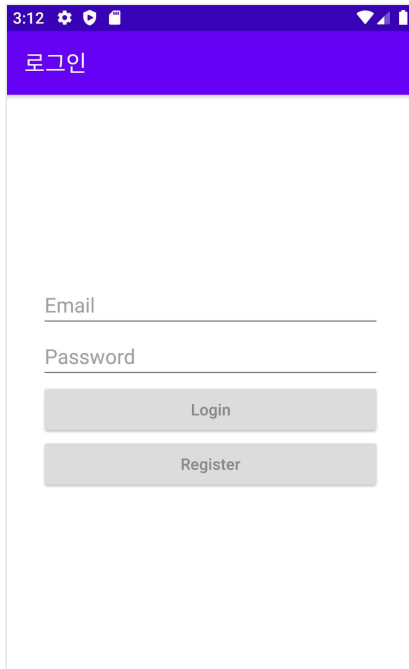
7) 'java.lang.nosuchmethoderror: no static method metafactory' 에러가 발생한 경우 자바 호환성에 대한 명시를 한다.

```
android {
    ...
    compileOptions{
        sourceCompatibility JavaVersion.VERSION_1_8
        targetCompatibility JavaVersion.VERSION_1_8
    }
}
```

- Firebase Authentication

1) [Tools]-[Firebase] 메뉴를 선택한다.
2) [Authentication] - [Email and password authentication] 메뉴를 선택한다.
3) [Add Firebase Authentication to your app] 생성된 앱에 인증관련에 필요한 dependencies를 설정한다.
4) [launch in browser]를 실행하여 콘솔로 이동한다.
5) [생성된 프로젝트]-[Firebase]-[Authentication]-[Sign-in method]-[이메일/비밀번호] 상태를 사용으로 설정한다.

[activity_main.xml]

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    tools:context=".MainActivity"
    android:orientation="vertical"
    android:layout_gravity="center_vertical"
    android:padding="30sp">
    <EditText
        android:id="@+id/edtEmail"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Email"/>
    <EditText
        android:id="@+id/edtPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Password"
        android:inputType="textPassword"/>
    <Button
        android:id="@+id/btnLogin"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Login"/>
    <Button
        android:id="@+id/btnRegister"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Register"/>
</LinearLayout>
```

```java
public class MainActivity extends AppCompatActivity {
    EditText edtEmail, edtPassword;
    Button btnRegister, btnLogin;
    String strEmail, strPassword;
    private FirebaseAuth mAuth;

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

        mAuth = FirebaseAuth.getInstance();
        edtEmail=(EditText)findViewById(R.id.edtEmail);
        edtPassword=(EditText)findViewById(R.id.edtPassword);
        btnRegister=(Button)findViewById(R.id.btnRegister);
        btnLogin=(Button)findViewById(R.id.btnLogin);

        btnRegister.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                strEmail=edtEmail.getText().toString();
                if(strEmail.length()<8){
                    Toast.makeText(MainActivity.this,"비밀번호 8자리이상 입력", Toast.LENGTH_SHORT).show();
                }else{
                    strPassword=edtPassword.getText().toString();
                    registerUser(strEmail, strPassword);
                }
            }
        });
        btnLogin.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                strEmail=edtEmail.getText().toString();
                strPassword=edtPassword.getText().toString();
                loginUser(strEmail, strPassword);
            }
        });
    }

    public void registerUser(String strEmail, String strPassword){
        mAuth.createUserWithEmailAndPassword(strEmail, strPassword)
                .addOnCompleteListener(this, new OnCompleteListener<AuthResult>() {
                    public void onComplete(@NonNull Task<AuthResult> task) {
                        if (task.isSuccessful()) {
                            FirebaseUser user = mAuth.getCurrentUser();
                            Toast.makeText(MainActivity.this, "사용자등록성공", Toast.LENGTH_SHORT).show();
                        } else {
                            Toast.makeText(MainActivity.this, "사용자등록실패", Toast.LENGTH_SHORT).show();
                        }
                    }
                });
    }

    public void loginUser(String email, String password){
        mAuth.signInWithEmailAndPassword(email, password)
                .addOnCompleteListener(this, new OnCompleteListener<AuthResult>() {
                    public void onComplete(@NonNull Task<AuthResult> task) {
                        if (task.isSuccessful()) {
                            Toast.makeText(MainActivity.this, "인증성공", Toast.LENGTH_SHORT).show();
                            FirebaseUser user = mAuth.getCurrentUser();
                            Intent intent=new Intent(MainActivity.this, MemoActivity.class);
                            startActivity(intent);
                            finish();
                        } else {
                            Toast.makeText(MainActivity.this, "인증실패", Toast.LENGTH_SHORT).show();
                        }
                    }
                });
    }
}
```
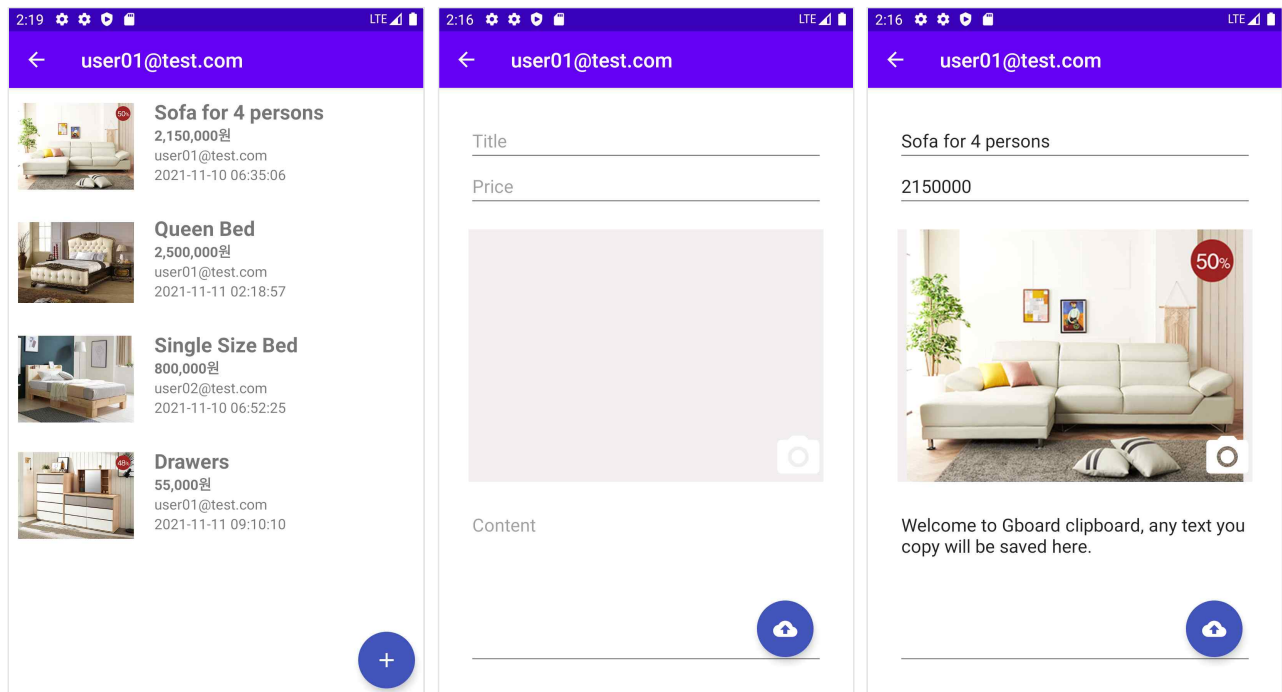
2-2. Firebase 상품관리 예제 프로그램

- **Firebase Firestore와 Storage**

1) [Tools]-[Firebase] 메뉴를 선택한다.

2) [Firestore]-[Read and write documents with Cloud Firestore] 메뉴를 선택한다.

3) [Add Cloud Firestore to your app] 생성된 앱에 Firesore 관련에 필요한 dependencies를 설정한다.

4) [Storage]-[Upload and download a file with Cloud Storage] 메뉴를 선택한다.

5) [Add Cloud Storage to your app] 생성된 앱에 Storage 관련에 필요한 dependencies를 설정한다.

6) [launch in browser]를 실행하여 콘솔로 이동한다.

7) [생성된 프로젝트]-[Firestore Database]-[규칙]에서 read, write 규칙을 'true'로 변경한다.

    allow read, write: if true;.

8) [생선된 프로젝트]-[Storage]-[규칙]에서 read, write 규칙을 'true'로 변경한다.

[mainfests]-[AndroidManifest.xml]

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android package="com.example.ex01">
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.CAMERA" />
    <application ...>
        <provider
            android:name="androidx.core.content.FileProvider"
            android:authorities="${applicationId}"
            android:exported="false"
            android:grantUriPermissions="true">
            <meta-data
                android:name="android.support.FILE_PROVIDER_PATHS"
                android:resource="@xml/file_path" />
        </provider>
    </application>
</manifest>
```

[res]-[xml]-[file_path.xml]

```
<paths xmlns:android="http://schemas.android.com/apk/res/android">
    <!--getCacheDir() 내부저장소 -->
    <cache-path name="cache" path="."/>
</paths>
```

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".ProductActivity">
    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/list"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>
    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/register"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_alignParentRight="true"
        android:backgroundTint="#3F51B5"
        app:borderWidth="0sp"
        android:layout_marginBottom="10sp"
        android:layout_marginRight="10sp"
        android:src="@drawable/ic_register"/>
</RelativeLayout>
```

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:padding="10sp">
    <ImageView
        android:id="@+id/image"
        android:layout_width="300px"
        android:layout_height="250px"
        android:src="@mipmap/ic_launcher"
        android:layout_marginRight="20sp"/>
    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_toRightOf="@id/image"
        android:orientation="vertical">
        <TextView
            android:id="@+id/name"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:textSize="20sp"
            android:textStyle="bold"
            android:text="여기에는 상품명출력"/>
        <TextView
            android:id="@+id/price"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:textStyle="bold"
            android:text="여기에는 상품가격출력"/>
        <TextView
            android:id="@+id/email"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="여기에는 이메일출력"/>
        <TextView
            android:id="@+id/date"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="여기에는 등록일출력"/>
    </LinearLayout>
</RelativeLayout>
```

```java
public class ProductVO {
    private String id;
    private String name;
    private int price;
    private String content;
    private String email;
    private String date;
    private String url;
    private String image;
    //... setter()/ getter()
}
```

```java
public class ProductAdapter extends RecyclerView.Adapter<ProductAdapter.ViewHolder> {
    Context context;
    ArrayList<ProductVO> array;

    public ProductAdapter(Context context, ArrayList<ProductVO> array) {
        this.context = context;
        this.array = array;
    }

    @NonNull
    @Override
    public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        View view= LayoutInflater.from(context).inflate(R.layout.item_product, parent, false);
        return new ViewHolder(view);
    }

    @Override
    public void onBindViewHolder(@NonNull ViewHolder holder, int position) {
        ProductVO vo=array.get(position);
        holder.name.setText(vo.getName());
        DecimalFormat df=new DecimalFormat("#,###원");
        holder.price.setText(df.format(vo.getPrice()));
        holder.date.setText(vo.getDate());
        holder.email.setText(vo.getEmail());
        Picasso.with(context).load(vo.getUrl()).into(holder.image);

        holder.image.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent=new Intent(context, ReadActivity.class);
                intent.putExtra("id", vo.getId());
                context.startActivity(intent);
            }
        });
    }

    @Override
    public int getItemCount() {
        return array.size();
    }

    public class ViewHolder extends RecyclerView.ViewHolder {
        TextView name, price, date, email;
        ImageView image;
        public ViewHolder(@NonNull View itemView) {
            super(itemView);
            name=itemView.findViewById(R.id.name);
            price=itemView.findViewById(R.id.price);
            date=itemView.findViewById(R.id.date);
            email=itemView.findViewById(R.id.email);
            image=itemView.findViewById(R.id.image);
        }
    }
}
```

```java
public class ProductActivity extends AppCompatActivity {
    FirebaseUser user;
    FirebaseFirestore db;
    ProductAdapter adapter;
    RecyclerView list;
    ArrayList<ProductVO> array = new ArrayList<>();

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

        user = FirebaseAuth.getInstance().getCurrentUser();
        getSupportActionBar().setTitle(user.getEmail());
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);

        list = findViewById(R.id.list);
        adapter = new ProductAdapter(this, array);
        list.setLayoutManager(new LinearLayoutManager(this));
        list.setAdapter(adapter);

        FloatingActionButton register=findViewById(R.id.register);
        register.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent=new Intent(ProductActivity.this, RegisterActivity.class);
                startActivity(intent);
            }
        });
        readProducts();
    }

    public void readProducts(){
        db=FirebaseFirestore.getInstance();
        array.clear();
        db.collection("products").get().addOnCompleteListener(new OnCompleteListener<QuerySnapshot>() {
                @Override
                public void onComplete(@NonNull Task<QuerySnapshot> task) {
                    if(task.isSuccessful()){
                        for(QueryDocumentSnapshot doc: task.getResult()){
                            ProductVO vo=new ProductVO();
                            vo.setId(doc.getId());
                            vo.setName(doc.getData().get("name").toString());
                            vo.setContent(doc.getData().get("content").toString());
                            vo.setPrice(Integer.parseInt(doc.getData().get("price").toString()));
                            vo.setImage(doc.getData().get("image").toString());
                            vo.setUrl(doc.getData().get("url").toString());
                            vo.setEmail(doc.getData().get("email").toString());
                            vo.setDate(doc.getData().get("date").toString());
                            array.add(vo);
                            adapter.notifyDataSetChanged();
                        }
                    }
                }
            });
    }

    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        switch (item.getItemId()){
            case android.R.id.home:
                finish();
        }
        return super.onOptionsItemSelected(item);
    }

    protected void onRestart() {
        readProducts();
        super.onRestart();
    }
}
```

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".RegisterActivity"
    android:padding="30sp">
    <EditText
        android:id="@+id/name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Title"/>
    <EditText
        android:id="@+id/price"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/name"
        android:layout_marginBottom="20sp"
        android:hint="Price"/>
    <ImageView
        android:id="@+id/image"
        android:layout_width="match_parent"
        android:layout_height="250sp"
        android:layout_below="@id/price"
        android:background="#ECE8E8"
        android:layout_marginBottom="20sp"/>
    <ImageView
        android:id="@+id/photo"
        android:layout_width="50sp"
        android:layout_height="50sp"
        android:src="@drawable/ic_photo"
        android:layout_alignBottom="@id/image"
        android:layout_alignRight="@id/image"/>
    <EditText
        android:id="@+id/content"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_below="@id/image"
        android:hint="Content"
        android:gravity="top"/>
    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/register"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_alignParentRight="true"
        android:backgroundTint="#3F51B5"
        app:borderWidth="0sp"
        android:layout_marginBottom="10sp"
        android:layout_marginRight="10sp"
        android:src="@drawable/ic_upload"/>
</RelativeLayout>
```

```java
public class RegisterActivity extends AppCompatActivity {
    EditText name, price, content;
    String strName, strPrice, strContent, strFile, strUrl, strImage, strEmail;
    FirebaseFirestore db;
    FirebaseUser user;
    ImageView image;
    StorageReference storageRef;

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

        checkPermission(); //퍼미션 체크
        user = FirebaseAuth.getInstance().getCurrentUser();
        strEmail = user.getEmail();
        getSupportActionBar().setTitle(strEmail);
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
```

```java
            name=findViewById(R.id.name);
            price=findViewById(R.id.price);
            content=findViewById(R.id.content);
            FloatingActionButton register=findViewById(R.id.register);
            image=findViewById(R.id.image);

            storageRef = FirebaseStorage.getInstance().getReference();
            db=FirebaseFirestore.getInstance();
            register.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    strName=name.getText().toString();
                    strPrice=price.getText().toString();
                    strContent=content.getText().toString();

                    if(strName.equals("") || strPrice.equals("") || strContent.equals("") || strFile==null){
                        Toast.makeText(RegisterActivity.this, "내용을 입력하세요!", Toast.LENGTH_SHORT).show();
                    }else {  //이미지 업로드 및 상품정보 등록
                        Uri file = Uri.fromFile(new File(strFile));
                        strImage = strFile.substring(strFile.lastIndexOf("/") + 1);
                        StorageReference imageRef = storageRef.child("images/" + strImage);
                        imageRef.putFile(file)
                            .addOnSuccessListener(new OnSuccessListener<UploadTask.TaskSnapshot>() {
                                @Override
                                public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {
                                    Task<Uri> result = taskSnapshot.getStorage().getDownloadUrl();
                                    result.addOnSuccessListener(new OnSuccessListener<Uri>() {
                                        @Override
                                        public void onSuccess(Uri uri) {
                                            strUrl = uri.toString();
                                            insertProduct();  //상품정보등록
                                        }
                                    });
                                }
                            });
                    }
                }
            });

            ImageView photo=findViewById(R.id.photo);
            photo.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    AlertDialog.Builder box=new AlertDialog.Builder(RegisterActivity.this);
                    box.setTitle("이미지선택방법");
                    box.setPositiveButton("앨범", new DialogInterface.OnClickListener() {
                        @Override
                        public void onClick(DialogInterface dialogInterface, int i) {  //앨범을 선택한 경우
                            Intent intent=new Intent(Intent.ACTION_PICK, MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
                            startActivityForResult(intent, 1);
                        }
                    });
                    box.setNegativeButton("카메라", new DialogInterface.OnClickListener() {
                        @Override
                        public void onClick(DialogInterface dialogInterface, int i) {  //카메라를 선택한 경우
                            Intent intent= new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
                            try {  //카메라 파일명(/storage/emulated/0/Android/data/패키지명/files/Pictures/파일명)
                                File storageDir=getCacheDir();  //캐시저장소 사용
                                File file=File.createTempFile("img_",".jpg",storageDir);
                                strFile=file.getAbsolutePath();
                                Uri photoURI = FileProvider.getUriForFile(RegisterActivity.this, getPackageName(), file);
                                intent.putExtra(MediaStore.EXTRA_OUTPUT, photoURI);
                                startActivityForResult(intent, 0);
                            } catch (Exception e) { System.out.println("error..." + e.toString()); }
                        }
                    });
                    box.show();
                }
            });
        }
```

```java
    protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
        switch (requestCode) {
            case 1: //앨범 이미지를 선택한 경우
                //image.setImageURI(data.getData()); uri로 이미지를 출력
                //앨범 파일명(/storage/emulated/0/DCIM/Camera/파일명)
                Cursor cursor = getContentResolver().query(data.getData(), null, null, null, null);
                cursor.moveToFirst();
                strFile = cursor.getString(cursor.getColumnIndex(MediaStore.Images.Media.DATA));
                cursor.close();
            case 0:
                Bitmap bitmap= BitmapFactory.decodeFile(strFile);
                image.setImageBitmap(bitmap);
                break;
        }
        super.onActivityResult(requestCode, resultCode, data);
    }

    public void insertProduct(){
        HashMap<String, Object> product=new HashMap<>();
        product.put("name", strName);
        product.put("price", Integer.parseInt(strPrice));
        product.put("content", strContent);
        product.put("image", strImage);
        product.put("url", strUrl);
        SimpleDateFormat sdf=new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
        product.put("date", sdf.format(new Date()));
        product.put("email", strEmail);

        db.collection("products")
            .add(product)
            .addOnSuccessListener(new OnSuccessListener<DocumentReference>() {
                @Override
                public void onSuccess(DocumentReference documentReference) {
                    Toast.makeText(RegisterActivity.this, "상품등록 성공!", Toast.LENGTH_SHORT).show();
                    finish();
                }
            })
            .addOnFailureListener(new OnFailureListener() {
                @Override
                public void onFailure(@NonNull Exception e) {
                    Toast.makeText(RegisterActivity.this, "상품등록 실패!", Toast.LENGTH_SHORT).show();
                }
            });
    }

    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        switch (item.getItemId()){
            case android.R.id.home:
                finish();
        }
        return super.onOptionsItemSelected(item);
    }

    public void checkPermission(){
        String[] permissions= { Manifest.permission.READ_EXTERNAL_STORAGE,
                                Manifest.permission.WRITE_EXTERNAL_STORAGE,
                                Manifest.permission.CAMERA };

        ArrayList<String> noPermissions=new ArrayList<>();
        for(String permission:permissions){
            if(ActivityCompat.checkSelfPermission(this, permission) != PackageManager.PERMISSION_GRANTED){
                noPermissions.add(permission);
            }
        }
        if(noPermissions.size() > 0){
            String[] reqPermissions=noPermissions.toArray(new String[noPermissions.size()]);
            ActivityCompat.requestPermissions(this, reqPermissions,100);
        }
    }
}
```

```java
public class ReadActivity extends AppCompatActivity {
    EditText name, price, content;
    ImageView image;
    FloatingActionButton register;
    String id, strEmail;
    ProductVO vo = new ProductVO();
    FirebaseFirestore db;
    FirebaseUser user;

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

        user = FirebaseAuth.getInstance().getCurrentUser();
        strEmail = user.getEmail();
        getSupportActionBar().setTitle(strEmail);
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);

        Intent intent=getIntent();
        id = intent.getStringExtra("id");
        name=findViewById(R.id.name);
        price=findViewById(R.id.price);
        content=findViewById(R.id.content);
        image=findViewById(R.id.image);

        register=findViewById(R.id.register);
        register.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if(!strEmail.equals(vo.getEmail())){
                    Toast.makeText(ReadActivity.this, "상품등록자가 아닙니다!", Toast.LENGTH_SHORT).show();
                    return;
                }
                vo.setName(name.getText().toString());
                vo.setPrice(Integer.parseInt(price.getText().toString()));
                vo.setContent(content.getText().toString());
                db.collection("products").document(id).set(vo).addOnSuccessListener(new OnSuccessListener<Void>() {
                    @Override
                    public void onSuccess(Void aVoid) {
                        finish();
                    }
                });
            }
        });

        db=FirebaseFirestore.getInstance();
        db.collection("products").document(id).get().addOnCompleteListener(new OnCompleteListener<DocumentSnapshot>() {
            @Override
            public void onComplete(@NonNull Task<DocumentSnapshot> task) {
                if(task.isSuccessful()){
                    DocumentSnapshot doc = task.getResult();
                    vo.setId(id);
                    vo.setName(doc.get("name").toString());
                    vo.setPrice(Integer.parseInt(doc.get("price").toString()));
                    vo.setContent(doc.get("content").toString());
                    vo.setImage(doc.get("image").toString());
                    vo.setUrl(doc.get("url").toString());
                    vo.setEmail(doc.get("email").toString());
                    vo.setDate(doc.get("date").toString());
                    name.setText(vo.getName());
                    price.setText(String.valueOf(vo.getPrice()));
                    content.setText(vo.getContent());
                    Picasso.with(ReadActivity.this).load(vo.getUrl()).into(image);
                }
            }
        });
    }
}
```
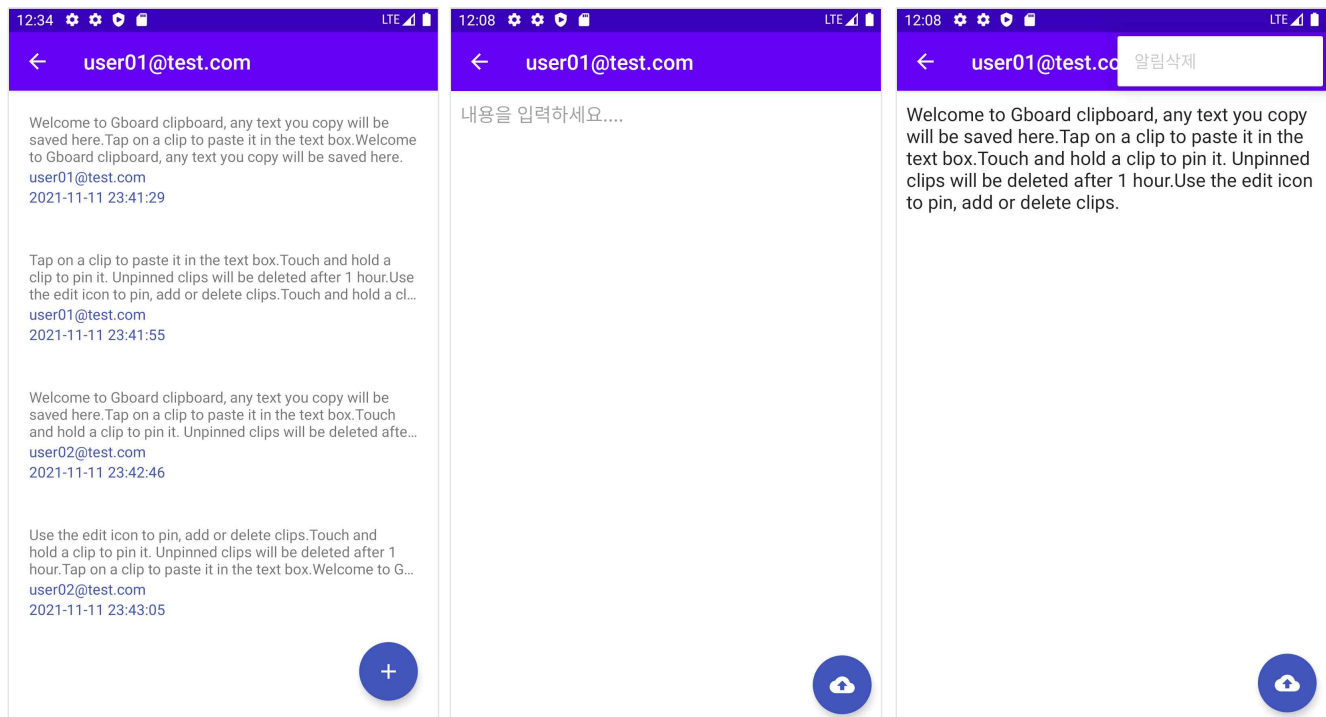
2-3. Firebase 알림 예제 프로그램

• **Firebase Realtime Database**

1) [Tools]-[Firebase] 메뉴를 선택한다.
2) [Realtime Database]-[Save and retrieve data] 메뉴를 선택한다.
3) [Add the Realtime Database to your app] 생성된 앱에 Realtime Database 관련에 필요한 dependencies를 설정한다.
4) [launch in browser]를 실행하여 콘솔로 이동한다.
5) [생성된 프로젝트]-[Realtime Database] 새로운 데이터베이스를 생성한다.
6) [생성된 프로젝트]-[Firestore]-[Realtime Database]-[규칙]에서 read, write 규칙을 'true'로 변경한다.

[res]-[layout]-[item-notice.xml]

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:padding="20sp"
    android:orientation="vertical"
    android:id="@+id/item">
    <TextView
        android:id="@+id/content"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="여기에 내용을 출력합니다."
        android:maxLines="3"
        android:ellipsize="end"/>
    <TextView
        android:id="@+id/email"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="여기에 이메일을 출력합니다."
        android:textColor="#3F51B5"/>
    <TextView
        android:id="@+id/date"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="여기에 날짜를 출력합니다."
        android:textColor="#3F51B5"/>
</LinearLayout>
```

[java]-[com.example.ex01]-[NoticeVO.java]

```java
public class NoticeVO implements Serializable {
    private String key;
    private String content;
    private String email;
    private String date;
    //...setter() gettert()
}
```

[res]-[menu]-[menu_notice.xml]

```xml
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/delete"
        android:title="알림삭제"/>
</menu>
```

[java]-[com.example.ex01]-[NoticeAdapter.java]

```java
public class NoticeAdapter extends RecyclerView.Adapter<NoticeAdapter.ViewHolder>{
    Context context;
    ArrayList<NoticeVO> array;

    public NoticeAdapter(Context context, ArrayList<NoticeVO> array) {
        this.context = context;
        this.array = array;
    }

    @NonNull
    @Override
    public NoticeAdapter.ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        View view=LayoutInflater.from(context).inflate(R.layout.item_notice, parent, false);
        return new ViewHolder(view);
    }

    @Override
    public void onBindViewHolder(@NonNull NoticeAdapter.ViewHolder holder, int position) {
        NoticeVO vo=array.get(position);
        holder.content.setText(vo.getContent());
        holder.email.setText(vo.getEmail());
        holder.date.setText(vo.getDate());
        holder.item.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent=new Intent(context, NoticeReadActivity.class);
                intent.putExtra("vo", vo);
                context.startActivity(intent);
            }
        });
    }

    @Override
    public int getItemCount() {
        return array.size();
    }

    public class ViewHolder extends RecyclerView.ViewHolder {
        TextView content, email, date;
        LinearLayout item;
        public ViewHolder(@NonNull View itemView) {
            super(itemView);
            content=itemView.findViewById(R.id.content);
            email=itemView.findViewById(R.id.email);
            date=itemView.findViewById(R.id.date);
            item=itemView.findViewById(R.id.item);
        }
    }
}
```

```java
public class NoticeActivity extends AppCompatActivity {
    FirebaseUser user;
    ArrayList<MemoVO> array=new ArrayList<>();
    FirebaseDatabase db;
    DatabaseReference ref;
    NoticeAdapter noticeAdapter;

    @Override
    protected void onRestart() {
        array.clear();
        callList();
        super.onRestart();
    }

    public void callList(){
        ref.addChildEventListener(new ChildEventListener() {
            @Override
            public void onChildAdded(@NonNull DataSnapshot snap, @Nullable String previousChildName) {
                NoticeVO vo=snap.getValue(NoticeVO.class);
                array.add(vo);
                memoAdapter.notifyDataSetChanged();
            }
            @Override
            public void onChildChanged(@NonNull DataSnapshot snapshot, @Nullable String previousChildName) {

            }
            ...
        });
    }

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(activity_notice_register.xml]);

        user= FirebaseAuth.getInstance().getCurrentUser();
        db=FirebaseDatabase.getInstance("https://positive-notch-259405-default-rtdb.firebaseio.com/");
        ref=db.getReference("memos").child(user.getUid());
        callList();

        noticeAdapter = new NoticeAapter(this, array);
        RecyclerView list=findViewById(R.id.list);
        list.setAdapter(memoAdapter);
        list.setLayoutManager(new LinearLayoutManager(this));

        getSupportActionBar().setTitle(user.getEmail());
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);

        FloatingActionButton edit=findViewById(R.id.register);
        edit.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent=new Intent(ReadActivity.this, InsertActivity.class);
                startActivity(intent);
            }
        });
    }

    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        switch (item.getItemId()){
            case android.R.id.home:
                finish();
                break;
        }
        return super.onOptionsItemSelected(item);
    }
}
```

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".NoticeRegisterActivity">
    <EditText
        android:id="@+id/content"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:hint="내용을 입력하세요...."
        android:gravity="top"
        android:padding="10sp"
        android:background="@null"/>
    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/register"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_alignParentRight="true"
        android:backgroundTint="#3F51B5"
        app:borderWidth="0sp"
        android:layout_marginBottom="10sp"
        android:layout_marginRight="10sp"
        android:src="@drawable/ic_upload"/>
</RelativeLayout>
```

```java
public class NoticeRegisterActivity extends AppCompatActivity {
    FirebaseUser user;
    EditText content;
    FirebaseDatabase db;
    DatabaseReference dbRef;

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

        user = FirebaseAuth.getInstance().getCurrentUser();
        getSupportActionBar().setTitle(user.getEmail());
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
        db=FirebaseDatabase.getInstance();
        content = findViewById(R.id.content);

        FloatingActionButton register=findViewById(R.id.register);
        register.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if(content.getText().toString().equals("")){
                    Toast.makeText(NoticeRegisterActivity.this, "내용을 입력하세요!", Toast.LENGTH_SHORT).show();
                }else{
                    NoticeVO vo=new NoticeVO();
                    vo.setContent(content.getText().toString());
                    SimpleDateFormat sdf=new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
                    vo.setDate(sdf.format(new Date()));
                    vo.setEmail(user.getEmail());
                    dbRef=db.getReference("notices").push();  //"notices"/자동키 생성
                    vo.setKey(dbRef.getKey());
                    dbRef.setValue(vo);
                    Toast.makeText(NoticeRegisterActivity.this, "저장완료!", Toast.LENGTH_SHORT).show();
                    finish();
                }
            }
        });
    }
}
```

```java
public class NoticeReadActivity extends AppCompatActivity {
    FirebaseUser user;
    FirebaseDatabase db;
    DatabaseReference dbRef;
    EditText content;
    NoticeVO vo=new NoticeVO();

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

        user = FirebaseAuth.getInstance().getCurrentUser();
        getSupportActionBar().setTitle(user.getEmail());
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);

        content=findViewById(R.id.content);
        Intent intent = getIntent();
        vo=(NoticeVO)intent.getSerializableExtra("vo");
        content.setText(vo.getContent());

        db=FirebaseDatabase.getInstance();
        dbRef=db.getReference("notices").child(vo.getKey());

        FloatingActionButton register=findViewById(R.id.register);
        register.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if(!vo.getEmail().equals(user.getEmail())){
                    Toast.makeText(NoticeReadActivity.this, "작성자가 아니므로 수정 불가능!", Toast.LENGTH_SHORT).show();
                }else{
                    vo.setContent(content.getText().toString());
                    dbRef.setValue(vo);
                    finish();
                }
            }
        });
    }

    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        switch (item.getItemId()) {
            case android.R.id.home:
                finish();
                break;
            case R.id.delete:
                AlertDialog.Builder box=new AlertDialog.Builder(this);
                box.setMessage("삭제하실래요?");
                box.setPositiveButton("예", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialogInterface, int i) {
                        dbRef.removeValue();
                        finish();
                    }
                });
                box.setNegativeButton("아니오", null);
                box.show();
                break;
        }
        return super.onOptionsItemSelected(item);
    }

    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.menu_notice, menu);
        if(vo.getEmail().equals(user.getEmail())) {
            menu.findItem(R.id.delete).setEnabled(true);
        }else {
            menu.findItem(R.id.delete).setEnabled(false);
        }
        return super.onCreateOptionsMenu(menu);
    }
}
```
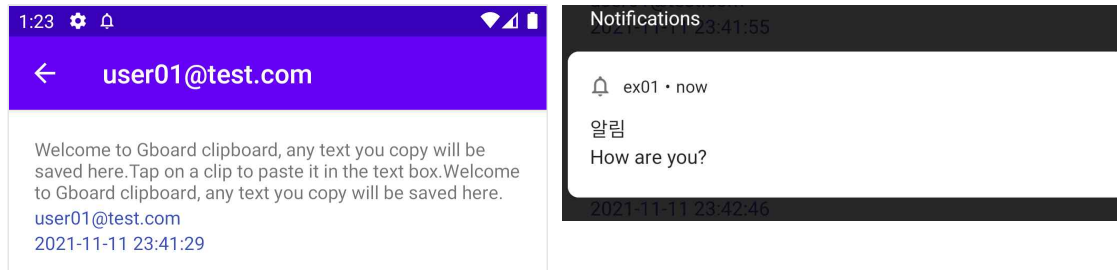
- **Firebase Cloud Messaging**

1) [Tools]-[Firebase] 메뉴를 선택한다.

2) [Cloud Messaging]-[Set up Firebase Cloud Messaging] 메뉴를 선택한다.

3) [Add FCM to your app] 생성된 앱에 FCM 관련에 필요한 dependencies를 설정한다.

[mainfests]-[AndroidManifest.xml]

```xml
<manifest xmlns:android="http://schemas.android.com/apk/res/android" package="com.example.ex01">
    <uses-permission android:name="android.permission.INTERNET"/>
    <application ...>
        ...
        <service android:name=".MessageService" android:exported="false">
            <intent-filter>
                <action android:name="com.google.firebase.MESSAGING_EVENT"/>
            </intent-filter>
        </service>
        <meta-data
            android:name="com.google.firebase.messaging.default_notification_icon"
            android:resource="@drawable/ic_notification"/>
    </application>
</manifest>
```

[java]-[com.example.ex01]-[MessageService.java]

```java
public class MessageService extends FirebaseMessagingService {
    @Override
    public void onNewToken(@NonNull String s) {
        super.onNewToken(s);
    }

    @Override
    public void onMessageReceived(@NonNull RemoteMessage remoteMessage) {
        if(remoteMessage.getNotification() != null){
            String messageBody = remoteMessage.getNotification().getBody();
            String messageTitle= remoteMessage.getNotification().getTitle();

            Intent intent = new Intent(this, MainActivity.class);
            PendingIntent pendingIntent= PendingIntent.getActivity(this,0,intent,0);

            String channelId="channel_01";
            NotificationCompat.Builder builder = new NotificationCompat.Builder(this, channelId);
            builder.setContentTitle(messageTitle);
            builder.setContentText(messageBody);
            builder.setSmallIcon(R.drawable.ic_notification);
            builder.setAutoCancel(true);
            builder.setContentIntent(pendingIntent);

            NotificationManager manager=(NotificationManager)getSystemService(NOTIFICATION_SERVICE);

            if(Build.VERSION.SDK_INT>=26){
                String channelName="channel_notification";
                NotificationChannel channel=
                    new NotificationChannel(channelId, channelName, NotificationManager.IMPORTANCE_DEFAULT);
                manager.createNotificationChannel(channel);
            }
            manager.notify(0, builder.build());
        }
        super.onMessageReceived(remoteMessage);
    }
}
```

4) FCM 메시지 보내기를 테스트하기위해 [launch in browser]를 실행하여 콘솔로 이동한다.

5) [생성된 프로젝트]-[Cloud Messaging]-[Send your first message] 메뉴를 선택한다.

6) [알림 제목]-[알림 텍스트]-[다음] 버튼을 클릭한다.

7) [사용자 타겟팅 조건]-[다음]-[검토]-[게시] 버튼을 클릭하여 앱에서 알림이 왔는지 확인한다.

• **로그인한 이메일 ID아래에 토큰값을 Realtime Database에 저장한다.**

[java]-[com.example.ex01]-[MainActivity.java]

```java
public class MainActivity extends AppCompatActivity {
    EditText edtEmail, edtPassword;
    Button btnRegister, btnLogin;
    String strEmail, strPassword, token;
    private FirebaseAuth mAuth;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        getSupportActionBar().setTitle("로그인");
        ...
    }

    //사용자 등록 버튼을 클릭한 경우
    public void registerUser(String strEmail, String strPassword){
        mAuth.createUserWithEmailAndPassword(strEmail, strPassword)
            .addOnCompleteListener(this, new OnCompleteListener<AuthResult>() {
                public void onComplete(@NonNull Task<AuthResult> task) {
                    if (task.isSuccessful()) {
                        FirebaseUser user = mAuth.getCurrentUser();
                        Toast.makeText(MainActivity.this, "사용자등록성공", Toast.LENGTH_SHORT).show();
                    } else {
                        Toast.makeText(MainActivity.this, "사용자등록실패", Toast.LENGTH_SHORT).show();
                    }
                }
            });
    }

    //로그인 버튼을 클릭한 경우
    public void loginUser(String email, String password){
        mAuth.signInWithEmailAndPassword(email, password)
            .addOnCompleteListener(this, new OnCompleteListener<AuthResult>() {
                public void onComplete(@NonNull Task<AuthResult> task) {
                    if (task.isSuccessful()) {
                        Toast.makeText(MainActivity.this, "인증성공", Toast.LENGTH_SHORT).show();
                        FirebaseUser user = mAuth.getCurrentUser();

                        //현재 토큰값을 가져와서 사용자 ID아래에 저장
                        FirebaseMessaging.getInstance().getToken()
                            .addOnCompleteListener(new OnCompleteListener<String>() {
                                @Override
                                public void onComplete(@NonNull Task<String> task) {
                                    token = task.getResult();
                                    FirebaseDatabase db=FirebaseDatabase.getInstance();
                                    HashMap<String, Object> map=new HashMap<>();
                                    map.put("token", token);
                                    db.getReference("FCMID/" + user.getUid()).setValue(map);
                                }
                            });

                        Intent intent=new Intent(MainActivity.this, NoticeActivity.class);
                        startActivity(intent);
                        finish();
                    } else {
                        Toast.makeText(MainActivity.this, "인증실패", Toast.LENGTH_SHORT).show();
                    }
                }
            });
    }
}
```

```java
public class NoticeRegisterActivity extends AppCompatActivity {
    ...
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        ...
        FloatingActionButton register=findViewById(R.id.register);
        register.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                ....
                db.getReference("FCMID/").addValueEventListener(new ValueEventListener() {
                    @Override
                    public void onDataChange(@NonNull DataSnapshot snapshot) {
                        for (DataSnapshot snap : snapshot.getChildren()) {  //FCM에 등록된 User ID 수만큼 반복
                            String user_id=snap.getKey();
                            HashMap<String, Object> map=(HashMap<String, Object>)snap.getValue();
                            String token =(String)map.get("token");
                            //보내는 User를 제외하는 경우 if( !user.getUid().equals(user_id) )
                            new ThreadSend(token).start();
                        }
                    }
                    @Override
                    public void onCancelled(@NonNull DatabaseError error) {
                    }
                });
                Toast.makeText(NoticeRegisterActivity.this, "저장완료!", Toast.LENGTH_SHORT).show();
                finish();
            }
        });
    }

    class ThreadSend extends Thread{
        String FCM_URL ="https://fcm.googleapis.com/fcm/send";
        String SERVER_KEY = "AAAAsL3IqTE:..."; //[Firebase 프로젝트]-[프로젝트설정]-[클라우드 메시징]-[서버키]
        String token;

        public ThreadSend(String token) {
            this.token = token;
        }

        @Override
        public void run() {
            try{
                //FCM 메시지 생성
                JSONObject root=new JSONObject();
                JSONObject notification = new JSONObject();
                notification.put("body", content.getText().toString());
                notification.put("title", "알림");
                root.put("notification", notification);
                root.put("to", token);

                //FCM 메시지 보내기
                URL url=new URL(FCM_URL);
                HttpURLConnection conn=(HttpURLConnection)url.openConnection();
                conn.setRequestMethod("POST");
                conn.setDoOutput(true);
                conn.setDoInput(true);
                conn.addRequestProperty("Authorization", "key=" + SERVER_KEY);
                conn.setRequestProperty("Accept", "application/json");
                conn.setRequestProperty("content-type", "application/json");
                OutputStream os=conn.getOutputStream();
                os.write(root.toString().getBytes("utf-8"));
                os.flush();
                conn.getResponseCode();
            }catch(Exception e){
                System.out.println("오류...................:" + e.toString());
            }
        }
    }
}
```
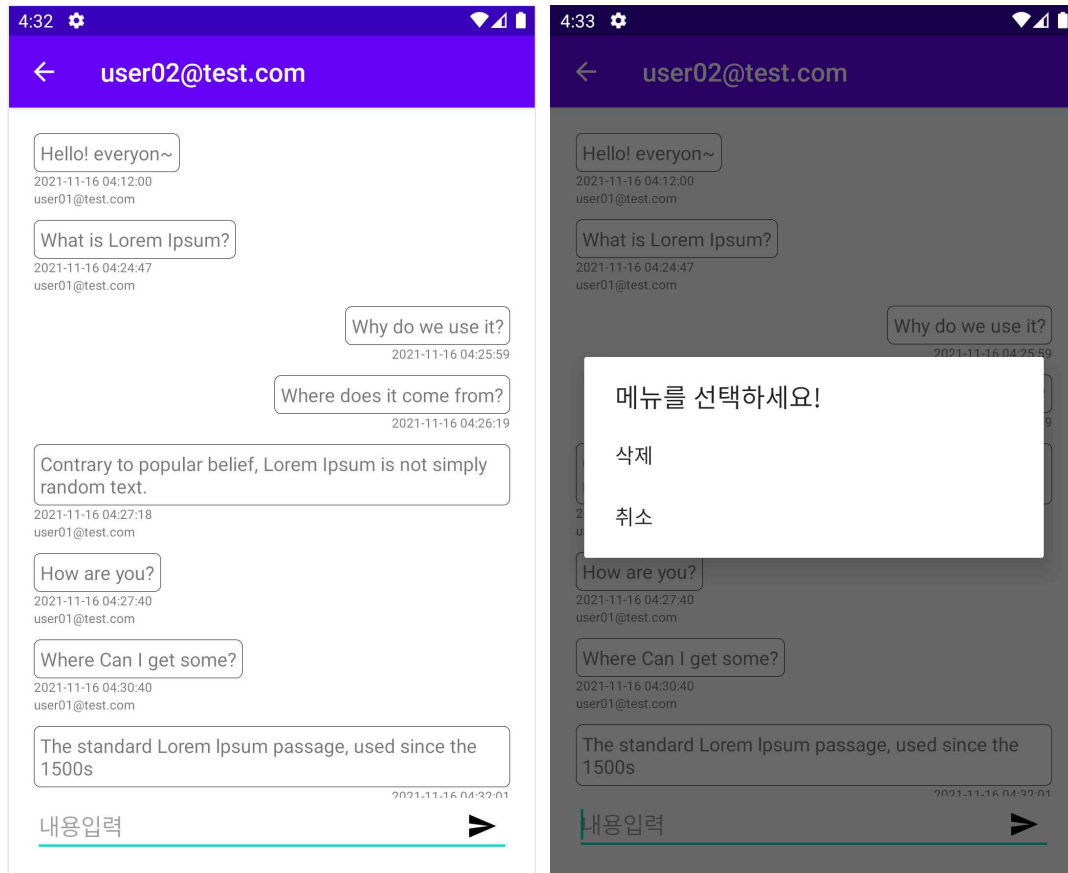
2-4. Firebase 채팅 예제 프로그램

[res]-[layout]-[activity_chat.xml]

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".ChatActivity"
    android:orientation="vertical"
    android:padding="20sp">
    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/list"
        android:layout_width="match_parent"
        android:layout_height="0sp"
        android:layout_weight="1"/>
    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <EditText
            android:id="@+id/content"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="내용입력"/>
        <ImageView
            android:id="@+id/send"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:src="@drawable/ic_send"
            android:layout_alignParentRight="true"
            android:padding="10sp"/>
    </RelativeLayout>
</LinearLayout>
```

```
chats
  -MoawRxVaH_QIuuVtHmb
    content: "Hello! everyon~"
    date: "2021-11-16 04:12:00"
    email: "user01@test.com"
    key: "-MoawRxVaH_QIuuVtHmb"
```

[res]-[drawable]-[bubble.xml]

```xml
<shape xmlns:android="http://schemas.android.com/apk/res/android">
    <solid
        android:color="#ffffff"/>
    <stroke
        android:width="0.5sp"
        android:color="#000000"/>
    <corners android:radius="5sp"/>
</shape>
```

[rex]-[layout]-[item_chat.xml]

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:layout_marginBottom="10sp"
    android:id="@+id/item">
    <TextView
        android:id="@+id/content"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="글내용........"
        android:textSize="15sp"
        android:padding="5sp"
        android:background="@drawable/bubble"/>
    <TextView
        android:id="@+id/date"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="글쓴날짜시간"
        android:textSize="10sp"/>
    <TextView
        android:id="@+id/email"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="글쓴이"
        android:textSize="10sp"/>
</LinearLayout>
```

[java]-[com.example.ex01]-[ChatVO.java]

```java
public class ChatVO {
    private String key;
    private String content;
    private String email;
    private String date;
    //setter(), getter() 생략
}
```

```java
public class ChatAdapter extends RecyclerView.Adapter<ChatAdapter.ViewHolder> {
    Context context;
    ArrayList<ChatVO> array;
    String user;
    DatabaseReference dbRef;
    public ChatAdapter(Context context, ArrayList<ChatVO> array, String user, DatabaseReference dbRef) {
        this.context = context;
        this.array = array;
        this.user = user;
        this.dbRef = dbRef;
    }
    public ChatAdapter.ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        View view= LayoutInflater.from(context).inflate(R.layout.item_chat, parent, false);
        return new ViewHolder(view);
    }

    public void onBindViewHolder(@NonNull ChatAdapter.ViewHolder holder, int position) {
        ChatVO vo=array.get(position);
        LinearLayout.LayoutParams paramsContent=(LinearLayout.LayoutParams)holder.content.getLayoutParams();
        LinearLayout.LayoutParams paramsEmail=(LinearLayout.LayoutParams)holder.email.getLayoutParams();
        LinearLayout.LayoutParams paramsDate=(LinearLayout.LayoutParams)holder.date.getLayoutParams();

        if(user.equals(vo.getEmail())) {
            paramsContent.gravity = Gravity.RIGHT;
            paramsEmail.gravity = Gravity.RIGHT;
            paramsDate.gravity = Gravity.RIGHT;
            holder.email.setVisibility(View.GONE);

            holder.content.setOnLongClickListener(new View.OnLongClickListener() {
                public boolean onLongClick(View view) {
                    AlertDialog.Builder box=new AlertDialog.Builder(context);
                    box.setTitle("메뉴를 선택하세요!");
                    box.setItems(new String[]{"삭제", "취소"}, new DialogInterface.OnClickListener() {
                        public void onClick(DialogInterface dialogInterface, int i) {
                            if(i==0) {
                                dbRef.child(vo.getKey()).removeValue();
                                array.remove(vo);
                            }
                        }
                    });
                    box.show();
                }
            });
        }else {
            paramsContent.gravity = Gravity.LEFT;
            paramsEmail.gravity = Gravity.LEFT;
            paramsDate.gravity = Gravity.LEFT;
            holder.email.setVisibility(View.VISIBLE);
        }
        holder.date.setText(vo.getDate());
        holder.email.setText(vo.getEmail());
        holder.content.setText(vo.getContent());
    }

    public int getItemCount() {
        return array.size();
    }

    public class ViewHolder extends RecyclerView.ViewHolder {
        TextView content, date, email;
        LinearLayout item;
        public ViewHolder(@NonNull View itemView) {
            content=itemView.findViewById(R.id.content);
            date=itemView.findViewById(R.id.date);
            email=itemView.findViewById(R.id.email);
            item=itemView.findViewById(R.id.item);
        }
    }
}
```

```java
public class ChatActivity extends AppCompatActivity {
    FirebaseUser user;
    FirebaseDatabase db;
    DatabaseReference dbRef;
    ChatAdapter adapter;
    ArrayList<ChatVO> array=new ArrayList<>();
    RecyclerView list;

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

        user = FirebaseAuth.getInstance().getCurrentUser();
        getSupportActionBar().setTitle(user.getEmail());
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
        db = FirebaseDatabase.getInstance();
        dbRef= db.getReference("chats");

        adapter = new ChatAdapter(this, array, user.getEmail(), dbRef);
        list=findViewById(R.id.list);
        list.setLayoutManager(new LinearLayoutManager(this));
        list.setAdapter(adapter);

        dbRef.addChildEventListener(new ChildEventListener() {
            @Override
            public void onChildAdded(@NonNull DataSnapshot snapshot, @Nullable String previousChildName) {
                ChatVO vo=snapshot.getValue(ChatVO.class);
                array.add(vo);
                list.scrollToPosition(array.size()-1);
                adapter.notifyDataSetChanged();
            }
            @Override
            public void onChildRemoved(@NonNull DataSnapshot snapshot) {
                adapter.notifyDataSetChanged();
            }
            ...
        });

        EditText content=findViewById(R.id.content);
        ImageView send=findViewById(R.id.send);
        send.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String strContent=content.getText().toString();
                if(strContent.equals("")) {
                    Toast.makeText(ChatActivity.this, "내용을 입력하세요!", Toast.LENGTH_SHORT).show();
                }else{
                    ChatVO vo=new ChatVO();
                    vo.setEmail(user.getEmail());
                    vo.setContent(strContent);
                    SimpleDateFormat sdf=new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
                    vo.setDate(sdf.format(new Date()));
                    dbRef=db.getReference("chats").push();  //"chats"/자동키 생성
                    vo.setKey(dbRef.getKey());
                    dbRef.setValue(vo);
                    content.setText("");
                }
            }
        });
    }

    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        switch (item.getItemId()) {
            case android.R.id.home:
                finish();
        }
        return super.onOptionsItemSelected(item);
    }
}
```
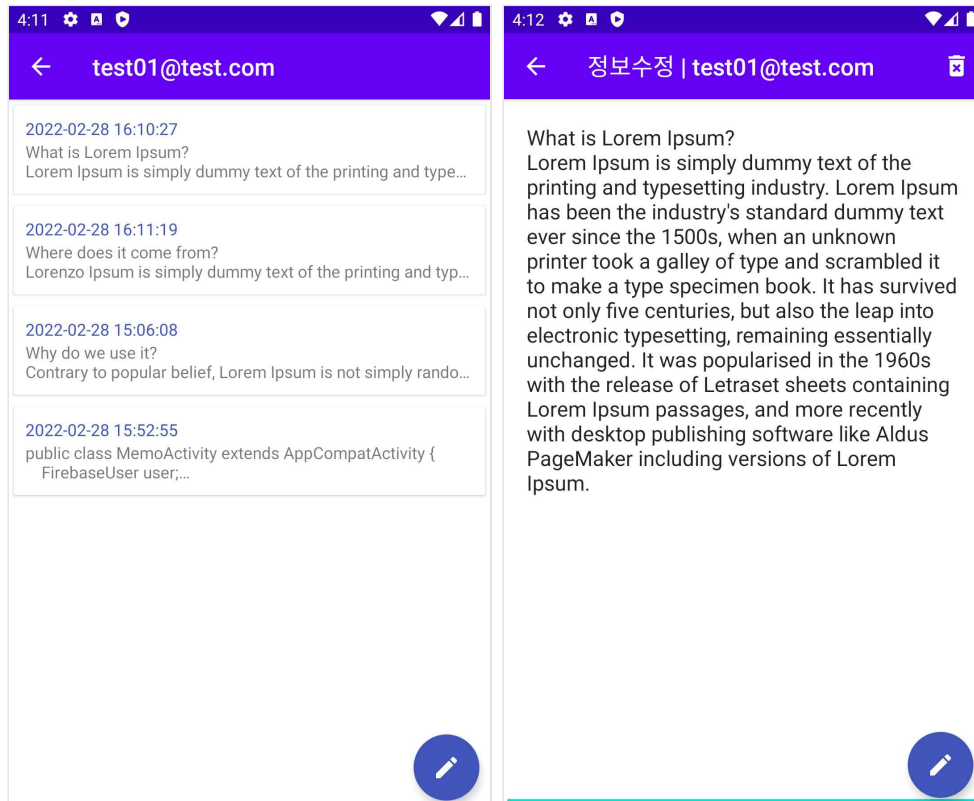
2-5. Firebase 메모장 예제 프로그램

```xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayoutxmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MemoActivity">

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/list"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>

    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/edit"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/ic_edit"
        android:backgroundTint="#3F51B5"
        android:layout_alignParentBottom="true"
        android:layout_alignParentRight="true"
        android:layout_margin="10sp"
        app:borderWidth="0sp"/>
</RelativeLayout>
```

[java]-[com.example.ex01]-[MemoVO.java]

```java
public class MemoVO implements Serializable {
    private String key;
    private String email;
    private String date;
    private String contents;

    //setter(), getter() 생략
}
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.cardview.widget.CardView
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="5sp"
    android:id="@+id/item">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:padding="10sp">
        <TextView
            android:id="@+id/date"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="메모작성날짜"
            android:textColor="#3F51B5"/>
        <TextView
            android:id="@+id/contents"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="메모내용"
            android:lines="2"
            android:ellipsize="end"/>
    </LinearLayout>
</androidx.cardview.widget.CardView>
```

```java
public class MemoActivity extends AppCompatActivity    {
    FirebaseUser user;
    ArrayList<MemoVO> array=new   ArrayList<>();
    FirebaseDatabasedb;
    DatabaseReference ref;
    MemoAdaptermemoAdapter=new MemoAdapter();

    @Override
    protected   void onRestart() {
        array.clear();
        callList();
        super.onRestart();
    }

    public void callList(){
        ref.addChildEventListener(new ChildEventListener() {
            @Override
            public void onChildAdded(@NonNull DataSnapshot snap, @Nullable String   previousChildName) {
                MemoVOvo=snap.getValue(MemoVO.class);
                array.add(vo);
                System.out.println("....." + array.size() + ":" + vo.toString());
                memoAdapter.notifyDataSetChanged();
            }
             @Override
             public void onChildChanged(@NonNull DataSnapshot snapshot, @Nullable   String previousChildName) {
            }
            ...
        });
    }

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

        user=   FirebaseAuth.getInstance().getCurrentUser();
        db=FirebaseDatabase.getInstance("https://positive-notch-259405-default-rtdb.firebaseio.com/");
        ref=db.getReference("memos").child(user.getUid());
```

```java
        callList();

        RecyclerView list=findViewById(R.id.list);
        list.setAdapter(memoAdapter);
        list.setLayoutManager(new   LinearLayoutManager(this));

        getSupportActionBar().setTitle(user.getEmail());
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);

        FloatingActionButton edit=findViewById(R.id.edit);
        edit.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent=new Intent(MemoActivity.this, InsertActivity.class);
                startActivity(intent);
            }
        });
    }

    @Override
    public   booleanonOptionsItemSelected(@NonNull MenuItem item) {
        switch (item.getItemId()){
            case android.R.id.home:
            finish();
            break;
        }
        return   super.onOptionsItemSelected(item);
    }

    //메모 어댑터정의
    class MemoAdapter extends RecyclerView.Adapter<MemoAdapter.ViewHolder>{
        @NonNull
        @Override
        public   MemoAdapter.ViewHolderonCreateViewHolder(@NonNull ViewGroup parent, int   viewType) {
            View view=getLayoutInflater().inflate(R.layout.item,parent,false);
            return new ViewHolder(view);
        }

        @Override
        public   void onBindViewHolder(@NonNull MemoAdapter.ViewHolder holder, int position) {
            final MemoVOvo=array.get(position);
            holder.contents.setText(vo.getContents());
            holder.date.setText(vo.getDate());
            holder.item.setOnClickListener(new   View.OnClickListener() {
                @Override
                public void onClick(View v) {
                    Intent intent=new Intent(MemoActivity.this, ReadActivity.class);
                    intent.putExtra("vo", vo);
                    startActivity(intent);
                }
            });
        }

        @Override
        public   int getItemCount() {
            return array.size();
        }

        public   class ViewHolder extends RecyclerView.ViewHolder {
            TextView contents, date;
            CardView item;
            public ViewHolder(@NonNull View itemView) {
                super(itemView);
                contents=itemView.findViewById(R.id.contents);
                date=itemView.findViewById(R.id.date);
                item=itemView.findViewById(R.id.item);
            }
        }
    }
}
```

```xml
<RelativeLayoutxmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto" xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".InsertActivity">
    <EditText
        android:id="@+id/contents"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:hint="메모작성"
        android:gravity="start"
        android:padding="20sp"/>
     <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/edit"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/ic_edit"
        android:backgroundTint="#3F51B5"
        android:layout_alignParentBottom="true"
        android:layout_alignParentRight="true"
        android:layout_margin="10sp"
        app:borderWidth="0sp"/>
</RelativeLayout>
```

```java
public class InsertActivity extends AppCompatActivity {
    FirebaseUser user;
    EditText contents;
    FirebaseDatabasedb;
    DatabaseReference ref;

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

        contents=findViewById(R.id.contents);

        user= FirebaseAuth.getInstance().getCurrentUser();
        getSupportActionBar().setTitle("메모작성 | " + user.getEmail());
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);

        db=FirebaseDatabase.getInstance("https://positive-notch-259405-default-rtdb.firebaseio.com/");

        //저장버튼을 클릭한 경우
        FloatingActionButton edit=findViewById(R.id.edit);
        edit.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String strContents=contents.getText().toString();
                if(strContents.equals("")){
                    Toast.makeText(InsertActivity.this, "메모내용을 입력하세요!", Toast.LENGTH_SHORT).show();
                }else {
                    MemoVOvo=new MemoVO();
                    vo.setContents(strContents);
                    SimpleDateFormatsdf=new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
                    vo.setDate(sdf.format(new Date()));
                    vo.setEmail(user.getEmail());
                    ref=db.getReference("memos").child(user.getUid()).push();  //memos/user.getUid()/자동키 생성
                    vo.setKey(ref.getKey());
                    ref.setValue(vo);
                    finish();
                }
            }
        });
    }
}
```

```java
//저장된 User 정보출력
refUser=db.getReference("users").child(user.getUid());
refUser.get().addOnCompleteListener(new OnCompleteListener<DataSnapshot>() {
    @Override
    public void onComplete(@NonNull Task<DataSnapshot> task) {
        userVO=task.getResult().getValue(UserVO.class);
        if(userVO != null) {  //데이터가 있으면
            name.setText(userVO.getName());
            address.setText(userVO.getAddress());
            if (!userVO.getImage().equals("") || userVO.getImage() != null) {
                Picasso.with(MypageActivity.this).load(userVO.getImage()).into(image);
            }
        }
    }
});

//User 정보저장
refUser=db.getReference("users").child(user.getUid());
vo.setKey(refUser.getKey());  //키:users/user.getUid()
refUser.setValue(vo);
```

[res]-[menu]-[read.xml]

```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/delete"
        android:title="삭제"
        app:showAsAction="always"
        android:icon="@drawable/ic_delete"/>
</menu>
```

[java]-[com.example.ex01]-[ReadActity.java]

```java
public class ReadActivity extends AppCompatActivity {
    FirebaseUser user;
    MemoVO;
    EditText contents;
    FirebaseDatabasedb;
    DatabaseReference ref;

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

        db=FirebaseDatabase.getInstance("https://positive-notch-259405-default-rtdb.firebaseio.com/");
```

```java
        user=FirebaseAuth.getInstance().getCurrentUser();
        getSupportActionBar().setTitle("정보수정 | " + user.getEmail());
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);

        Intent   intent =getIntent();
        vo=(MemoVO)intent.getSerializableExtra("vo");

        contents=findViewById(R.id.contents);
        contents.setText(vo.getContents());

        FloatingActionButton edit=findViewById(R.id.edit);
        edit.setOnClickListener(new View.OnClickListener() {  //수정버튼을 클릭한 경우
            @Override
            public void onClick(View v) {
                final String strContent=contents.getText().toString();
                if(!vo.getContents().equals(strContent)){  //내용이 변경된 경우
                    AlertDialog.Builder box=new   AlertDialog.Builder(ReadActivity.this);
                    box.setTitle("질의");
                    box.setMessage("변경된 내용을 수정하실래요?");
                    box.setPositiveButton("예", new DialogInterface.OnClickListener() {
                        @Override
                        public void   onClick(DialogInterface dialog, int which) {
                            vo.setContents(strContent);  //메모수정
                            SimpleDateFormatsdf=new SimpleDateFormat("yyyy-MM-dd   HH:mm:ss");
                            vo.setDate(sdf.format(new Date()));
                            ref=db.getReference("memos").child(user.getUid()+"/" +vo.getKey());
                            ref.setValue(vo);
                            finish();
                        }
                    });
                    box.setNegativeButton("아니오",null);
                    box.show();
                }
            }
        });
    }

    @Override
    public booleanonOptionsItemSelected(@NonNull MenuItem item) {
        switch (item.getItemId()){
            case android.R.id.home:
                finish();
                break;
            case R.id.delete:
                AlertDialog.Builder box=new   AlertDialog.Builder(this);
                box.setTitle("질의");
                box.setMessage("메모를 삭제하실래요?");
                box.setPositiveButton("예", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialog, int which) {
                        ref=db.getReference("memos").child(user.getUid()+"/"    +vo.getKey());
                        ref.removeValue();
                        finish();
                    }
                });
                box.setNegativeButton("아니오", null);
                box.show();
                break;
        }
        return super.onOptionsItemSelected(item);
    }

    @Override
    public booleanonCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.read, menu);
        return super.onCreateOptionsMenu(menu);
    }
}
```

3-1. 사용자 관리 프로그램

▪ **userinfo 테이블 생성 및 데이터 입력**

[userinfo] 테이블 생성

```sql
create table userinfo (
    id varchar(8) primary key not null,
    name varchar(10) not null,
    password varchar(8) not null,
);
```

[userinfo] 테이블 데이터 입력

```sql
insert into userinfo(name, id, password) values("김재영", "spider", "pass");
insert into userinfo(name, id, password) values("박철규", "iceman", "pass");
insert into userinfo(name, id, password) values("변재희", "sparrow", "pass");
insert into userinfo(name, id, password) values("진석영", "batman", "pass");
```

● **테스트: http://localhost:8080/user/list.jsp json-simple-xxx.jar 파일을 [톰캣서버]-[lib] 폴더에 복사한다.**

[WebContent]-[list.jsp]

```jsp
<%@page import="java.sql.*, org.json.simple.*"%>
<%
    response.setCharacterEncoding("UTF-8");
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/webdb", "root", "1234");

        String sql="select * from userinfo";
        PreparedStatement ps=con.prepareStatement(sql);
        ResultSet rs=ps.executeQuery();

        JSONArray array=new JSONArray();
        while(rs.next()){
            JSONObject object=new JSONObject();
            object.put("id", rs.getString("id"));
            object.put("password", rs.getString("password"));
            object.put("name", rs.getString("name"));
            array.add(object);
        }
        out.print(array.toJSONString());
    }catch(Exception e){
        response.setStatus(400);
    }
%>
```

● **테스트: http://localhost:8080/user/insert.jsp?id=user00&pass=1234&name=홍길동**

[WebContent]-[insert.jsp]

```jsp
<%@page import="java.sql.*"%>
<%
    request.setCharacterEncoding("UTF-8");
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/webdb", "root", "1234");

        String id=request.getParameter("id");
        String password=request.getParameter("password");
        String name=request.getParameter("name");

        String sql="insert into userinfo(id, password, name) values(?,?,?)";
        PreparedStatement ps=con.prepareStatement(sql);
        ps.setString(1, id);
        ps.setString(2, password);
        ps.setString(3, name);
        ps.execute();
    }catch(Exception e){
        response.setStatus(400);
    }
%>
```

- 테스트: http://localhost:8080/user/read.jsp?id=user00

```
<%@page import="java.sql.*, org.json.simple.*"%>
<%
    response.setCharacterEncoding("UTF-8");
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(jdbc:mysql://localhost:3306/webdb", "root", "1234");

        String id=request.getParameter("id");
        String sql="select * from userinfo where id=?";

        PreparedStatement ps=con.prepareStatement(sql);
        ps.setString(1, id);
        ResultSet rs=ps.executeQuery();

        JSONObject object=new JSONObject();
        if(rs.next()){
            object.put("id", rs.getString("id"));
            object.put("password", rs.getString("password"));
            object.put("name", rs.getString("name"));
        }
        out.print(object.toJSONString());

    }catch(Exception e){
        response.setStatus(400);
    }
%>
```

- 테스트: http://localhost:8080/user/update.jsp?id=user00&password=1234&김길동

[WebContent]-[delete.jsp]

```
<%@page import="java.sql.*"%>
<%
    request.setCharacterEncoding("UTF-8");
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(jdbc:mysql://localhost:3306/webdb", "root", "1234");

        String id=request.getParameter("id");
        String password=request.getParameter("password");
        String name=request.getParameter("name");

        System.out.println("id=" + id);
        String sql="update userinfo set password=?, name=? where id=?";
        PreparedStatement ps=con.prepareStatement(sql);
        ps.setString(1, password);
        ps.setString(2, name);
        ps.setString(3, id);
        ps.execute();

    }catch(Exception e){
        response.setStatus(400);
    }
%>
```

- 테스트: http://localhost:8080/user/delete.jsp?id=user00

[WebContent]-[update.jsp]

```
<%@page import="java.sql.*"%>
<%
    request.setCharacterEncoding("UTF-8");
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(jdbc:mysql://localhost:3306/webdb", "root", "1234");
        request.setCharacterEncoding("UTF-8");

        String id=request.getParameter("id");

        String sql="delete from userinfo where id=?";
        PreparedStatement ps=con.prepareStatement(sql);
        ps.setString(1, id);
        ps.execute();
    }catch(Exception e){
        response.setStatus(400);
    }
%>
```
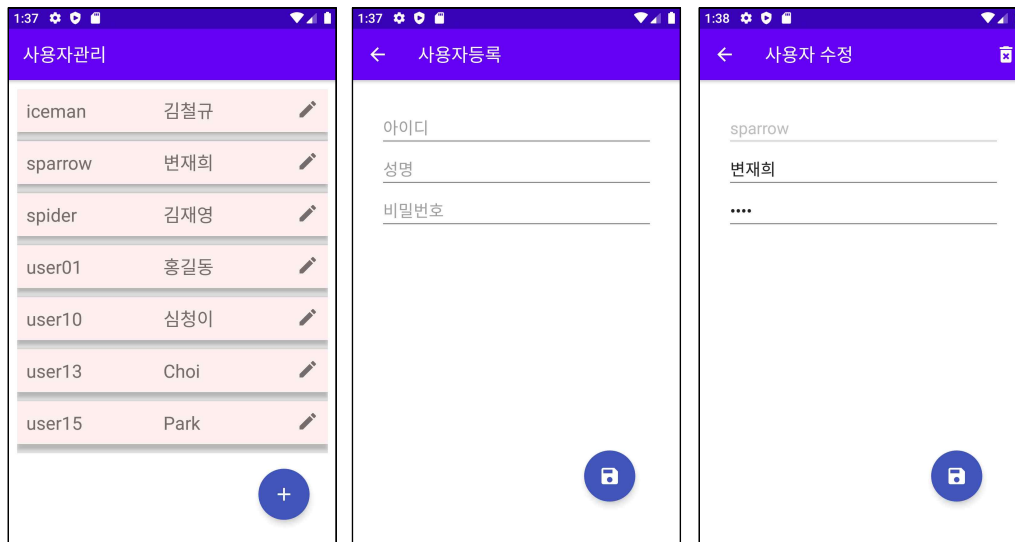
- **웹프로그램 설치**

1) [Eclipse]-[File]-[Export]-[Web] 프로젝트를 user.war파일로 생성한다.

2) 프로젝트.war 파일을 [c:\program Files]-[톰캣]-[webapps]에 복사한다.

3) [서비스]에서 톰캣 서버와 My-SQL DB 서버를 다시 구동한다.

4) [cmd]에서 ipconfig 명령으로 현재 PC의 IP를 검색한다.

5) 브라우저에서 http://IP Address:8080/user/list.jsp로 목록을 출력해 본다.

- **사용자 관리 앱프로그램**

AndroidManifest.xml

```
<uses-permission android:name="android.permission.INTERNET" />
<application
    ...
    android:usesCleartextTraffic="true">
```

[build.gradle] (Module:app)

```
dependencies {
    ...
    implementation 'com.android.support:design:30.0.0'
    implementation 'com.squareup.retrofit2:retrofit:2.2.0'
    implementation 'com.squareup.retrofit2:converter-gson:2.2.0'
}
```

activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto" xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:layout_margin="10sp">
    <ListView
        android:id="@+id/listUser"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:dividerHeight="10sp"/>
    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/btnAdd"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/ic_add"
        android:layout_alignParentBottom="true"
        android:layout_alignParentRight="true"
        android:layout_margin="20sp"
        app:borderWidth="0sp"
        android:backgroundTint="#3F51B5"/>
</RelativeLayout>
```

item_user.xml

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#F6E7E7"
    android:padding="10sp"
    android:elevation="5sp">
    <TextView
        android:id="@+id/txtId"
        android:layout_width="150sp"
        android:layout_height="wrap_content"
        android:text="여기는 아이디"
        android:textSize="20sp"/>
    <TextView
        android:id="@+id/txtName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="여기는 이름"
        android:layout_toRightOf="@id/txtId"
        android:textSize="20sp"/>
    <ImageView
        android:id="@+id/btnRead"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/ic_edit"
        android:layout_alignParentRight="true"/>
</RelativeLayout>
```

activity_add.xml

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".AddActivity"
    android:padding="30sp"
    android:paddingTop="200sp">
    <EditText
        android:id="@+id/edtId"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="아이디"/>
    <EditText
        android:id="@+id/edtName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="성명"
        android:layout_below="@id/edtId"/>
    <EditText
        android:id="@+id/edtPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="비밀번호"
        android:inputType="textPassword"
        android:layout_below="@id/edtName"/>
    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/btnSave"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/ic_save"
        android:layout_alignParentRight="true"
        android:layout_alignParentBottom="true"
        android:layout_margin="20sp"
        app:borderWidth="0sp"
        android:backgroundTint="#3F51B5"/>
</RelativeLayout>
```

```
public class UserVO {
    private String id;
    private String name;
    private String password;
    setter()/getter() 메서드 추가
}
```

```
public interface RemoteService {
    public static final String BASE_URL="http://172.30.1.2:8080/user/";

    @GET("list.jsp")
    Call<List<UserVO>> listUser();

    @GET("read.jsp")
    Call<UserVO> readUser(@Query("id") String id);

    @POST("insert.jsp")
    Call<Void> insertUser(
        @Query("id") String id,
        @Query("name") String name,
        @Query("password") String password);

    @POST("delete.jsp")
    Call<Void> deleteUser(@Query("id") String id);

    @POST("update.jsp")
    Call<Void> updateUser(
        @Query("id") String id,
        @Query("name") String name,
        @Query("password") String password);
}
```

```
public class MainActivity extends AppCompatActivity {
    Retrofit retrofit;
    RemoteService remoteService;
    List<UserVO> arrayUser=new ArrayList<>();
    UserAdapter userAdapter=new UserAdapter();
    ListView listUser;

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

        getSupportActionBar().setTitle("사용자관리");

        FloatingActionButton btnAdd=findViewById(R.id.btnAdd);
        btnAdd.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent=new Intent(MainActivity.this, AddActivity.class);
                startActivityForResult(intent, 1);
            }
        });

        listUser=findViewById(R.id.listUser);
        listUser.setAdapter(userAdapter);

        retrofit=new Retrofit.Builder()
            .baseUrl(BASE_URL)
            .addConverterFactory(GsonConverterFactory.create())
            .build();
        remoteService=retrofit.create(RemoteService.class);
```

```java
        Call<List<UserVO>> call=remoteService.listUser();
        call.enqueue(new Callback<List<UserVO>>() {
            @Override
            public void onResponse(Call<List<UserVO>> call, Response<List<UserVO>> response) {
                arrayUser=response.body();
                userAdapter.notifyDataSetChanged();
            }
            @Override
            public void onFailure(Call<List<UserVO>> call, Throwable t) { }
        });
    }

    class UserAdapter extends BaseAdapter{
        @Override
        public int getCount() {
            return arrayUser.size();
        }
        @Override
        public View getView(final int i, View view, ViewGroup viewGroup) {
            view= getLayoutInflater().inflate(R.layout.item_user, viewGroup, false);

            TextView txtId=view.findViewById(R.id.txtId);
            txtId.setText(arrayUser.get(i).getId());
            TextView txtName=view.findViewById(R.id.txtName);
            txtName.setText(arrayUser.get(i).getName());

            final String strId=arrayUser.get(i).getId();
            ImageView btnRead=view.findViewById(R.id.btnRead);
            btnRead.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    Intent intent=new Intent(MainActivity.this, ReadActivity.class);
                    intent.putExtra("id", strId);
                    startActivityForResult(intent, 2);
                }
            });
            return view;
        }
        @Override
        public Object getItem(int i) {
            return null;
        }
        @Override
        public long getItemId(int i) {
            return 0;
        }
    }

    @Override
    protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
        if(requestCode==1 && resultCode==RESULT_OK){
            Toast.makeText(this,"저장완료", Toast.LENGTH_SHORT).show();
        }else if(requestCode==2 && resultCode==RESULT_OK){
            Toast.makeText(this,"수정완료", Toast.LENGTH_SHORT).show();
        }else if(requestCode==2 && resultCode==3) {
            Toast.makeText(this, "삭제완료", Toast.LENGTH_SHORT).show();
        }
        Call<List<UserVO>> call=remoteService.listUser();
        call.enqueue(new Callback<List<UserVO>>() {
            @Override
            public void onResponse(Call<List<UserVO>> call, Response<List<UserVO>> response) {
                arrayUser=response.body();
                userAdapter.notifyDataSetChanged();
            }
            @Override
            public void onFailure(Call<List<UserVO>> call, Throwable t) { }
        });
        super.onActivityResult(requestCode, resultCode, data);
    }
}
```

```java
public class AddActivity extends AppCompatActivity {
    Retrofit retrofit;
    RemoteService remoteService;

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

        getSupportActionBar().setTitle("사용자등록");
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);

        retrofit=new Retrofit.Builder()
                .baseUrl(BASE_URL)
                .addConverterFactory(GsonConverterFactory.create())
                .build();
        remoteService=retrofit.create(RemoteService.class);

        FloatingActionButton btnSave=findViewById(R.id.btnSave);
        btnSave.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                AlertDialog.Builder box=new AlertDialog.Builder(AddActivity.this);
                box.setTitle("질의");
                box.setMessage("저장하실래요?");

                box.setPositiveButton("예", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialogInterface, int i) {
                        UserVO userVO=new UserVO();
                        EditText edtId=findViewById(R.id.edtId);
                        userVO.setId(edtId.getText().toString());
                        EditText edtName=findViewById(R.id.edtName);
                        userVO.setName(edtName.getText().toString());
                        EditText edtPassword=findViewById(R.id.edtPassword);
                        userVO.setPassword(edtPassword.getText().toString());

                        Call<Void> call=remoteService.insertUser(userVO.getId(), userVO.getName(), userVO.getPassword());
                        call.enqueue(new Callback<Void>() {
                            @Override
                            public void onResponse(Call<Void> call, Response<Void> response) {
                                setResult(RESULT_OK);
                                finish();
                            }

                            @Override
                            public void onFailure(Call<Void> call, Throwable t) {
                            }
                        });
                    }
                });

                box.setNegativeButton("아니오", null);
                box.show();
            }
        });
    }

    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        switch (item.getItemId()){
            case android.R.id.home:
                finish();
                break;
        }
        return super.onOptionsItemSelected(item);
    }
}
```

```java
public class ReadActivity extends AppCompatActivity {
    Retrofit retrofit;
    RemoteService remoteService;
    EditText edtId;
    EditText edtName;
    EditText edtPassword;
    Intent intent;

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

        getSupportActionBar().setTitle("사용자 수정");
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);

        edtId=findViewById(R.id.edtId);
        edtName=findViewById(R.id.edtName);
        edtPassword=findViewById(R.id.edtPassword);

        retrofit=new Retrofit.Builder()
                .baseUrl(BASE_URL)
                .addConverterFactory(GsonConverterFactory.create())
                .build();
        remoteService=retrofit.create(RemoteService.class);

        intent=getIntent();

        Call<UserVO> call=remoteService.readUser(intent.getStringExtra("id"));
        call.enqueue(new Callback<UserVO>() {
            @Override
            public void onResponse(Call<UserVO> call, Response<UserVO> response) {
                //결과값을 UserVO에 저장한다.
                UserVO userVO=response.body();

                edtId.setText(userVO.getId());
                edtId.setEnabled(false);
                edtName.setText(userVO.getName());
                edtPassword.setText(userVO.getPassword());
            }

            @Override
            public void onFailure(Call<UserVO> call, Throwable t) {
                System.out.println(t.toString());
            }
        });

        FloatingActionButton btnSave=findViewById(R.id.btnSave);
        btnSave.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                AlertDialog.Builder box=new AlertDialog.Builder(ReadActivity.this);
                box.setTitle("질의");
                box.setMessage("수정하실래요?");

                box.setPositiveButton("예", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialogInterface, int i) {
                        UserVO vo=new UserVO();
                        vo.setId(edtId.getText().toString());
                        vo.setName(edtName.getText().toString());
                        vo.setPassword(edtPassword.getText().toString());
```

```java
                    Call<Void> call=remoteService.updateUser(vo.getId(), vo.getName(), vo.getPassword());
                    call.enqueue(new Callback<Void>() {
                        @Override
                        public void onResponse(Call<Void> call, Response<Void> response) {
                            setResult(RESULT_OK);
                            //MainActivity의 onActivityResult 메서드로 이동한다.
                            finish();
                        }
                        @Override
                        public void onFailure(Call<Void> call, Throwable t) {
                            System.out.println(t.toString());
                        }
                    });
                }
            });

            box.setNegativeButton("아니오", null);
            box.show();
        }
    });
}


@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {
    switch (item.getItemId()){
        //ActionBar의 뒤로가기 버튼을 클릭한 경우
        case android.R.id.home:
            finish();
            break;
        case R.id.itemDelete:
            AlertDialog.Builder box=new AlertDialog.Builder(this);
            box.setTitle("질의");
            box.setMessage("삭제하실래요?");

            box.setPositiveButton("예", new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialogInterface, int i) {
                    Call call=remoteService.deleteUser(intent.getStringExtra("id"));
                    call.enqueue(new Callback() {
                        @Override
                        public void onResponse(Call call, Response response) {
                            setResult(3);  //resultCode가 3인 경우 삭제
                            finish();
                        }

                        @Override
                        public void onFailure(Call call, Throwable t) {
                            System.out.println(t.toString());
                        }
                    });
                }
            });
            box.setNegativeButton("아니오", null);
            box.show();
    }
    return super.onOptionsItemSelected(item);
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.memu_read, menu);
    return super.onCreateOptionsMenu(menu);
}
}
```

3-2. 상품관리 프로그램

- userinfo 테이블 생성 및 데이터 입력

```
create table produtinfo(
    code char(5) primary key not null,
    pname varchar(50) not null,
    price int,
    image varchar(50),
);
```

```
insert into productinfo(code, pname, price, image) values('P0001', '더블침대', 2000000, 'img07.jpg');
insert into productinfo(code, pname, price, image) values('P0002', '가죽소파', 1000000, 'img04.jpg');
insert into productinfo(code, pname, price, image) values('P0003', '5단 서랍장', 50000, 'img06.jpg');
insert into productinfo(code, pname, price, image) values('P0004', '6인용식탁', 800000, 'img12.jpg');
```

- 상품 관리 웹프로그램

```
<host>
    ...
    <Context docBase="c:/Temp" path="/product/image"/>
    ...
</host>
```

```
<%@page import="java.sql.*, org.json.simple.*"%>
<%
    response.setCharacterEncoding("UTF-8");
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/webdb", "root", "1234");
        String sql="select * from productinfo";
        PreparedStatement ps=con.prepareStatement(sql);
        ResultSet rs=ps.executeQuery();

        JSONArray array=new JSONArray();
        while(rs.next()){
            JSONObject object=new JSONObject();
            object.put("code", rs.getString("code"));
            object.put("pname", rs.getString("pname"));
            object.put("price", rs.getInt("price"));
            object.put("image", rs.getString("image"));
            array.add(object);
        }
        out.print(array.toJSONString());
    }catch(Exception e){ response.setStatus(400); }
%>
```

```
<html>
    <body>
        <form method="post" action="insert.jsp" enctype="multipart/form-data">
            <div>코드: <input type="text" name="code"></div>
            <div>이름: <input type="text" name="pname"></div>
            <div>가격: <input type="text" name="price"></div>
            <div>파일: <input type="file" name="image" accept="image/*"></div>
            <div>
                <input type="reset" value="취소"><input type="submit" value="저장">
            </div>
        </form>
    </body>
</html>
```

- 파일 업로드를 위한 라이브러리(cos.jar) 파일을 [톰캣서버]-[lib]에 복사한 후 서버를 다시 구동한다.

[WebContent]-[insertjsp]

```
<%@ page import="java.sql.*, java.io.*"%>
<%@ page import="com.oreilly.servlet.*, com.oreilly.servlet.multipart.*" %>
<%
    request.setCharacterEncoding("UTF-8");
    String uploadPath="c:/Temp/";
    File mdPath=new File(uploadPath);

    MultipartRequest multi = new MultipartRequest(
        request, uploadPath,
        1024 * 1024 * 10,
        "UTF-8",
        new DefaultFileRenamePolicy()
    );

    String code=multi.getParameter("code");
    String pname=multi.getParameter("pname");
    int price=Integer.parseInt(multi.getParameter("price"));
    String image=multi.getFilesystemName("image");

    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/webdb", "root", "1234");
        String sql="insert into productinfo(code, pname, price, image) values(?,?,?,?)";
        PreparedStatement ps=con.prepareStatement(sql);
        ps.setString(1, code);
        ps.setString(2, pname);
        ps.setInt(3, price);
        ps.setString(4, image);
        ps.execute();
    }catch(Exception e){
        System.out.println(e.toString());
        response.setStatus(400);
    }
%>
```

[WebContent]-[delete.jsp]

```
<%@page import="java.sql.*, java.io.*"%>
<%
    response.setCharacterEncoding("UTF-8");
    String uploadPath="c:/Temp/";
    File mdPath=new File(uploadPath);
    if(!mdPath.exists()) mdPath.mkdir();

    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/webdb", "root", "1234");
        String code=request.getParameter("code");
        String sql="select * from productinfo where code=?";
        PreparedStatement ps=con.prepareStatement(sql);
        ps.setString(1, code);
        ResultSet rs=ps.executeQuery();

        if(rs.next()){
            String image=rs.getString("image");
            File file=new File(uploadPath + image);
            if(file.exists()){ file.delete(); }
        }

        sql="delete from productinfo where code=?";
        ps=con.prepareStatement(sql);
        ps.setString(1, code);
        ps.execute();
    }catch(Exception e){
        System.out.println(e.toString());
        response.setStatus(400);
    }
%>
```
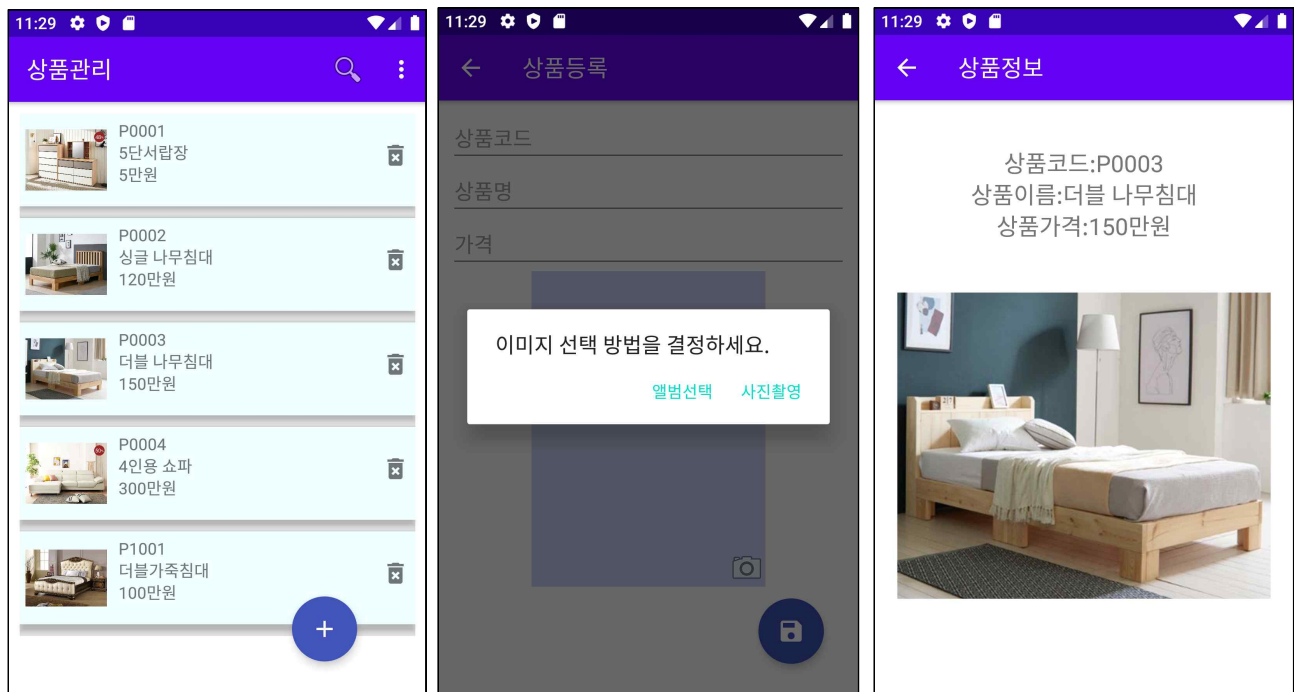
- **웹프로그램 설치**

1) [Eclipse]-[File]-[Export]-[Web] 프로젝트를 project.war파일로 생성한다.

2) 프로젝트.war 파일을 [c:\program Files]-[톰캣]-[webapps]에 복사한다.

3) [서비스]-[톰캣] 서버를 다시 구동한다.

4) [cmd]에서 ipconfig 명령으로 현재 PC의 IP를 검색한다.

5) 브라우저에서 http://IP Address:8080/product/list.jsp로 목록을 출력해 본다.

- **상품 관리 앱프로그램**

[build.gradle (Module:app)]

```
dependencies {
    implementation   'com.squareup.retrofit2:retrofit:2.2.0'
    implementation 'com.squareup.retrofit2:converter-gson:2.2.0'
    implementation 'com.squareup.picasso:picasso:2.5.2'
}
```

[AndroidManifest.xml]

```xml
<manifest xmlns:android="http://schemas.android.com/apk/res/android" package="com.example.product">
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>
    <uses-permission android:name="android.permission.CAMERA"/>
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme"
        android:usesCleartextTraffic="true"
        android:requestLegacyExternalStorage="true">
        <provider
            android:name="androidx.core.content.FileProvider"
            android:authorities="com.example.product.fileProvider"
            android:exported="false"
            android:grantUriPermissions="true">
            <meta-data
                android:name="android.support.FILE_PROVIDER_PATHS"
                android:resource="@xml/file_paths"/>
        </provider>
    </application>
</manifest>
```

```xml
<paths xmlns:android="http://schemas.android.com/apk/res/android">
    <external-path name="images" path="Android/data/com.example.productl/files/Pictures"/>
</paths>
```

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:layout_margin="10sp">
    <ListView
        android:id="@+id/listProduct"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:dividerHeight="10sp"/>
    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/btnAdd"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/ic_add"
        android:layout_alignParentBottom="true"
        android:layout_alignParentRight="true"
        android:layout_marginBottom="20sp"
        android:layout_marginRight="50sp"
        android:backgroundTint="#3F51B5"
        app:borderWidth="0sp"/>
</RelativeLayout>
```

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:padding="5sp"
    android:background="#E9F8FA"
    android:elevation="5sp"
    android:id="@+id/item">
    <ImageView
        android:id="@+id/image"
        android:layout_width="70sp"
        android:layout_height="70sp"
        android:src="@mipmap/ic_launcher"
        android:layout_marginRight="10sp"/>
    <TextView
        android:id="@+id/txtCode"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="상품코드"
        android:layout_toRightOf="@id/image"/>
    <TextView
        android:id="@+id/txtName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="상품이름"
        android:layout_toRightOf="@id/image"
        android:layout_below="@id/txtCode"/>
    <TextView
        android:id="@+id/txtPrice"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="상품가격"
        android:layout_toRightOf="@id/image"
        android:layout_below="@id/txtName"/>
    <ImageView
        android:id="@+id/btnDelete"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/ic_delete"
        android:layout_alignParentRight="true"
        android:layout_below="@id/txtCode"/>
</RelativeLayout>
```

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:orientation="vertical"
    android:padding="10sp">
    <EditText
        android:id="@+id/edtCode"  android:hint="상품코드"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
    <EditText
        android:id="@+id/edtPname"  android:hint="상품명"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/edtCode"/>
    <EditText
        android:id="@+id/edtPrice"  android:hint="가격"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/edtPname"/>
    <ImageView
        android:id="@+id/image"
        android:layout_width="200sp" android:layout_height="270sp"
        android:background="#C8CFF4"
        android:layout_centerHorizontal="true"
        android:layout_below="@id/edtPrice"/>
    <ImageView
        android:id="@+id/btnImage"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignBottom="@id/image"
        android:layout_alignEnd="@id/image"
        android:background="#C8CFF4"
        android:src="@android:drawable/ic_menu_camera"/>
    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/btnSave"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_alignParentRight="true"
        android:layout_margin="20sp"
        android:backgroundTint="#3F51B5" app:borderWidth="0sp"
        android:src="@drawable/ic_save"/>
</RelativeLayout>
```

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".ReadActivity"
    android:orientation="vertical"
    android:padding="20sp"
    android:gravity="center">
    <TextView
        android:id="@+id/txtCode"  android:text="상품코드"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="20sp"/>
    <TextView
        android:id="@+id/txtPname"  android:text="상품이름"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="20sp"/>
    <TextView
        android:id="@+id/txtPrice"  android:text="상품가격"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="20sp"/>
    <ImageView
        android:id="@+id/image"
        android:layout_width="350sp"
        android:layout_height="350sp"
        android:src="@mipmap/ic_launcher"/>
</LinearLayout>
```

```java
public class ProductVO {
    private String code;
    private String pname;
    private int price;
    private String image;
    ... setter()/getter() 메서드 추가
}
```

```java
public interface RemoteService {
    public static final String BASE_URL="http://192.168.0.7:8088/product/";

    @GET("list.jsp")
    Call<List<ProductVO>> listProduct(@Query("order") String order,@Query("query") String query);
    @GET("read.jsp")
    Call<ProductVO> readProduct(@Query("code") String code);
    @Multipart
    @POST("insert.jsp")
    Call<ResponseBody> uploadProduct(@Part("code")  RequestBody strCode,@Part("pname") RequestBody strPname,
            @Part("price") RequestBody strPrice, @Part MultipartBody.Part image);
    @POST("delete.jsp")
    Call<Void> deleteProduct(@Query("code") String code);
}
```

```java
public class MainActivity extends AppCompatActivity {
    Retrofit retrofit;
    RemoteService remoteService;
    List<ProductVO> arrayProduct=new ArrayList<>();
    ProductAdapter productAdapter=new ProductAdapter();
    String strOrder="code";
    String strQuery="";
    final static int ADD_ACTIVITY=2;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        getSupportActionBar().setTitle("상품관리");

        FloatingActionButton btnAdd=findViewById(R.id.btnAdd);
        btnAdd.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent=new Intent(MainActivity.this, AddActivity.class);
                startActivityForResult(intent, ADD_ACTIVITY);
            }
        });
        ListView list=findViewById(R.id.listProduct);
        list.setAdapter(productAdapter);

        retrofit=new Retrofit.Builder().baseUrl(BASE_URL).addConverterFactory(GsonConverterFactory.create()).build();
        remoteService=retrofit.create(RemoteService.class);
        callData(strOrder, strQuery);
    }
    public void callData(String order, String query){
        Call<List<ProductVO>> call=remoteService.listProduct(order, query);
        call.enqueue(new Callback<List<ProductVO>>() {
            @Override
            public void onResponse(Call<List<ProductVO>> call, Response<List<ProductVO>> response) {
                arrayProduct=response.body();
                productAdapter.notifyDataSetChanged();
            }
            @Override
            public void onFailure(Call<List<ProductVO>> call, Throwable t) { }
        });
    }
```

```
class ProductAdapter extends BaseAdapter{
    @Override
    public int getCount() {
        return arrayProduct.size();
    }

    @Override
    public View getView(final int i, View view, ViewGroup viewGroup) {
        view=getLayoutInflater().inflate(R.layout.item_product, viewGroup, false);
        TextView txtCode=view.findViewById(R.id.txtCode);
        TextView txtPname=view.findViewById(R.id.txtName);
        TextView txtPrice=view.findViewById(R.id.txtPrice);

        txtCode.setText(arrayProduct.get(i).getCode());
        txtPname.setText(arrayProduct.get(i).getPname());
        txtPrice.setText(arrayProduct.get(i).getPrice() + "만원");
        ImageView image=view.findViewById(R.id.image);
        Picasso.with(MainActivity.this).load(BASE_URL + "image/" + arrayProduct.get(i).getImage()).into(image);

        RelativeLayout item=view.findViewById(R.id.item);
        item.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent=new Intent(MainActivity.this, ReadActivity.class);
                intent.putExtra("code", arrayProduct.get(i).getCode());
                startActivityForResult(intent, 1);
            }
        });

        final String code=arrayProduct.get(i).getCode();
        ImageView btnDelete=view.findViewById(R.id.btnDelete);
        btnDelete.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                AlertDialog.Builder box=new AlertDialog.Builder(MainActivity.this);
                box.setTitle("질의");
                box.setMessage("삭제하실래요?");
                box.setPositiveButton("예", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialogInterface, int i) {
                        Call<Void> call=remoteService.deleteProduct(code);
                        call.enqueue(new Callback<Void>() {
                            @Override
                            public void onResponse(Call<Void> call, Response<Void> response) {
                                Toast.makeText(MainActivity.this,"삭제완료!", Toast.LENGTH_SHORT).show();
                                callData(strOrder, strQuery);
                            }
                            @Override
                            public void onFailure(Call<Void> call, Throwable t) {
                            }
                        });
                    }
                });
                box.setNegativeButton("아니오", null);
                box.show();
            }
        });
        return view;
    }

    @Override
    public Object getItem(int i) {
        return null;
    }

    @Override
    public long getItemId(int i) {
        return 0;
    }
}
```

```java
    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.menu_main, menu);
        MenuItem itemSearch=menu.findItem(R.id.itemSearch);
        SearchView searchView=(SearchView)itemSearch.getActionView();
        searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener() {
            @Override
            public boolean onQueryTextSubmit(String s) { return false; }
            @Override
            public boolean onQueryTextChange(String s) {
                strQuery=s;
                callData(strOrder, strQuery);
                return false;
            }
        });
        return super.onCreateOptionsMenu(menu);
    }

    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        switch (item.getItemId()){
            case R.id.itemCode:
                strOrder="code";
                break;
            case R.id.itemPname:
                strOrder="pname";
                break;
            case R.id.itemDesc:
                strOrder="desc";
                break;
            case R.id.itemAsc:
                strOrder="asc";
                break;
        }
        callData(strOrder,strQuery);
        return super.onOptionsItemSelected(item);
    }
    @Override
    protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
        if(requestCode==ADD_ACTIVITY && resultCode==RESULT_OK){
            Toast.makeText(this, "상품등록완료", Toast.LENGTH_SHORT).show();
            callData("code", "");
        }
        super.onActivityResult(requestCode, resultCode, data);
    }
}
```

[AddActivity.java]

```java
public class AddActivity extends AppCompatActivity {
    ImageView image;
    Retrofit retrofit;
    RemoteService remoteService;
    EditText code, pname, price;
    String strFile;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        permissionCheck();
        setContentView(R.layout.activity_add);
        getSupportActionBar().setTitle("상품등록");
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);

        code=findViewById(R.id.edtCode);
        pname=findViewById(R.id.edtPname);
        price=findViewById(R.id.edtPrice);

        image=findViewById(R.id.image);
        ImageView btnImage=findViewById(R.id.btnImage);
```

```java
        btnImage.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view){
                AlertDialog.Builder box=new AlertDialog.Builder(AddActivity.this);
                box.setTitle("이미지 선택 방법을 결정하세요.");

                box.setPositiveButton("사진촬영", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialogInterface, int i) {
                        Intent intent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
                        try {
                            File storageDir = getExternalFilesDir(Environment.DIRECTORY_PICTURES);
                            File file=File.createTempFile("img_", ".jpg", storageDir);
                            if(file != null){
                                strFile = file.getAbsolutePath();
                                Uri photoURI=FileProvider.getUriForFile(AddActivity.this,
                                                        "com.example.product.fileProvider", file);
                                intent.putExtra(MediaStore.EXTRA_OUTPUT, photoURI);
                                startActivityForResult(intent, 0);
                            }
                        }catch(Exception e){ }
                    }
                });
                box.setNegativeButton("앨범선택", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialogInterface, int i) {
                        Intent intent=new Intent(Intent.ACTION_PICK, MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
                        startActivityForResult(intent, 1);
                    }
                });
                box.show();
            }
        });
        retrofit=new Retrofit.Builder().baseUrl(BASE_URL).addConverterFactory(GsonConverterFactory.create()).build();
        remoteService=retrofit.create(RemoteService.class);

        FloatingActionButton btnSave=findViewById(R.id.btnSave);
        btnSave.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                File file=new File(strFile);
                RequestBody mFile = RequestBody.create(MediaType.parse("multipart/form-data"), file);
                MultipartBody.Part fileToUpload = MultipartBody.Part.createFormData("image", file.getName(), mFile);
                RequestBody strCode=RequestBody.create(MediaType.parse("multipart/form-data"), code.getText().toString());
                RequestBody strPname=RequestBody.create(MediaType.parse("multipart/form-data"),
                                                pname.getText().toString());
                RequestBody strPrice=RequestBody.create(MediaType.parse("multipart/form-data"), price.getText().toString());

                Call<ResponseBody> call=remoteService.uploadProduct(strCode, strPname, strPrice, fileToUpload);
                call.enqueue(new Callback<ResponseBody>() {
                    @Override
                    public void onResponse(Call<ResponseBody> call, Response<ResponseBody> response) {
                        setResult(RESULT_OK);
                        finish();
                    }
                    @Override
                    public void onFailure(Call<ResponseBody> call, Throwable t) { }
                });
            }
        });
    }
    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        switch (item.getItemId()){
            case android.R.id.home:
                finish();
                break;
        }
        return super.onOptionsItemSelected(item);
    }
```

```java
    @Override
    protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
        if(requestCode==1){  //갤러리인 경우
            try{
                String[] projection = { MediaStore.Images.Media.DATA };
                Cursor cursor = getContentResolver().query(data.getData(), projection, null, null, null);
                cursor.moveToFirst();
                strFile = cursor.getString(cursor.getColumnIndex(projection[0]));
                cursor.close();
                image.setImageBitmap(MediaStore.Images.Media.getBitmap(getContentResolver(), data.getData()));
            }catch (Exception e){}
        }else if(requestCode ==0){  //카메라인 경우
            Bitmap bmpImage=BitmapFactory.decodeFile(strFile); //Bitmap bitmapImage=(Bitmap)data.getExtras().get("data");
            image.setImageBitmap(bmpImage);
        }
        super.onActivityResult(requestCode, resultCode, data);
    }
    public void permissionCheck(){
        String[] permissions= {Manifest.permission.READ_EXTERNAL_STORAGE,
                Manifest.permission.WRITE_EXTERNAL_STORAGE, Manifest.permission.CAMERA };
        ArrayList<String> checkPermission=new ArrayList<>();
        for(String permission:permissions){
            if(ActivityCompat.checkSelfPermission(this, permission) != PackageManager.PERMISSION_GRANTED){
                checkPermission.add(permission);
            }
        }
        if(checkPermission.size() > 0){
            String[] reqPermission=checkPermission.toArray(new String[checkPermission.size()]);
            ActivityCompat.requestPermissions(this, reqPermission,100);
        }
    }
}
```

[ReadActivity.java]

```java
public class ReadActivity extends AppCompatActivity {
    Retrofit retrofit;
    RemoteService remoteService;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_read);
        getSupportActionBar().setTitle("상품정보");
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
        Intent intent=getIntent();
        retrofit=new Retrofit.Builder().baseUrl(BASE_URL).addConverterFactory(GsonConverterFactory.create()).build();
        remoteService=retrofit.create(RemoteService.class);

        Call<ProductVO> call=remoteService.readProduct(intent.getStringExtra("code"));
        call.enqueue(new Callback<ProductVO>() {
            @Override
            public void onResponse(Call<ProductVO> call, Response<ProductVO> response) {
                ProductVO vo=response.body();
                TextView txtCode=findViewById(R.id.txtCode);
                TextView txtPname=findViewById(R.id.txtPname);
                TextView txtPrice=findViewById(R.id.txtPrice);
                txtCode.setText("상품코드:" +vo.getCode());
                txtPname.setText("상품이름:" + vo.getPname());
                txtPrice.setText("상품가격:" + vo.getPrice() + "만원");
                ImageView image=findViewById(R.id.image);
                Picasso.with(ReadActivity.this).load(BASE_URL + "image/" + vo.getImage()).into(image);
            }
            @Override
            public void onFailure(Call<ProductVO> call, Throwable t) { }
        });
    }
    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        switch (item.getItemId()){
            case android.R.id.home:
                finish();
        }
        return super.onOptionsItemSelected(item);
    }
}
```

3-3. 맛집 관리 프로그램

▪ foodinfo 테이블 생성 및 데이터 입력

[foodinfo] 테이블 생성

```sql
create table foodinfo(
    seq int not null auto_increment primary key,
    name varchar(20) not null,
    tel varchar(20) not null,
    address varchar(50) not null,
    latitude double not null,
    longitude double not null,
);
```

[foodinfo] 데이터 입력

```sql
insert into foodinfo(name, tel, address, latitude, longitude)
values('쿠바', '032-565-6969', '인천광역시 서구 경서동 950-14', 37.54089, 126.6799233);
insert into foodinfo(name, tel, address, latitude, longitude)
values('테르미니', '02-972-0036', '서울특별시 노원구 공릉동 27-20', 37.6263383, 127.0919478);
insert into foodinfo(name, tel, address, latitude, longitude)
values('애플민트', '02-975-3710', '서울특별시 노원구 공릉동 113', 37.6263176, 127.0240964);
```
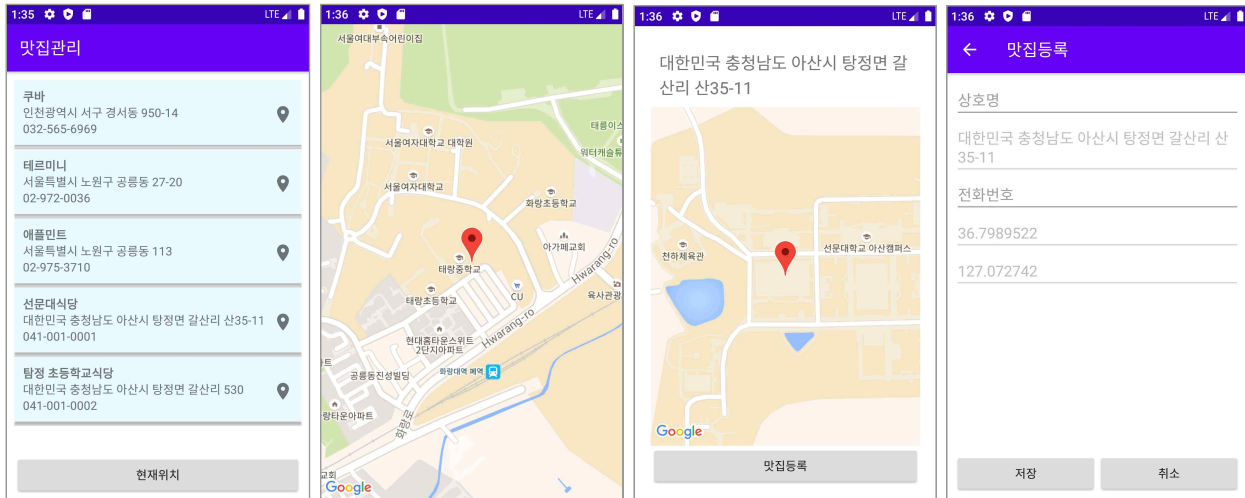
▪ 맛집 관리 웹프로그램

[WebContent]-[list.jsp]

```jsp
<%@page import="java.sql.*, org.json.simple.*"%>
<%
    response.setCharacterEncoding("UTF-8");
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/webdb", "root", "1234");
        String sql="select * from foodinfo";
        PreparedStatement ps=con.prepareStatement(sql);
        ResultSet rs=ps.executeQuery();
        JSONArray array=new JSONArray();
        while(rs.next()){
            JSONObject object=new JSONObject();
            object.put("seq", rs.getInt("seq"));
            object.put("name", rs.getString("name"));
            object.put("tel", rs.getString("tel"));
            object.put("address", rs.getString("address"));
            object.put("latitude", rs.getString("latitude"));
            object.put("longitude", rs.getString("longitude"));
            array.add(object);
        }
        out.print(array.toJSONString());
    }catch(Exception e){ out.print(e.toString()); }
%>
```

[WebContent]-[insert.jsp]

```jsp
<%@page import="java.sql.*"%>
<%
    request.setCharacterEncoding("UTF-8");
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/webdb", "root", "1234");
        String name=request.getParameter("name");
        String tel=request.getParameter("tel");
        String address=request.getParameter("address");
        String latitude=request.getParameter("latitude");
        String longitude=request.getParameter("longitude");
        String sql="insert into foodinfo(name, tel, address, latitude, longitude) values(?,?,?,?,?)"
        PreparedStatement ps=con.prepareStatement(sql);
        ps.setString(1, name);
        ps.setString(2, tel);
        ps.setString(3, address);
        ps.setDouble(4, Double.parseDouble(latitude));
        ps.setDouble(5, Double.parseDouble(longitude));
        ps.execute();
    }catch(Exception e){ out.print(e.toString());}
%>
```

- 맛집 관리 앱프로그램

```
dependencies {
    ...
    implementation  'com.squareup.retrofit2:retrofit:2.2.0'
    implementation 'com.squareup.retrofit2:converter-gson:2.2.0'
}
```

AndroidManifest.xml

```
<uses-permission android:name="android.permission.INTERNET"/>
<application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme"
        android:usesCleartextTraffic="true">
```

[res]-[layout]-[activity_main.xml]

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:orientation="vertical"
    android:padding="10sp">
    <ListView
        android:id="@+id/listFood"
        android:layout_width="match_parent"
        android:layout_height="0sp"
        android:layout_weight="1"
        android:dividerHeight="5sp"/>
    <Button
        android:id="@+id/btnLocation"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="현재위치"/>
</LinearLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#E3F3FA"
    android:padding="10sp"
    android:elevation="3sp"
    android:layout_margin="10sp">
    <TextView
        android:id="@+id/txtName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="상호명"
        android:textStyle="bold"/>
    <TextView
        android:id="@+id/txtTel"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="전화번호"
        android:layout_below="@id/txtName"/>
    <TextView
        android:id="@+id/txtAddress"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="주소"
        android:layout_below="@id/txtTel"/>
    <ImageView
        android:id="@+id/btnLocation"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/ic_location"
        android:layout_alignParentRight="true"
        android:layout_below="@id/txtName"/>
</RelativeLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:map="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="20sp">
    <TextView
        android:id="@+id/txtAddress"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="클릭한곳의 주소"
        android:textSize="20sp"
        android:padding="10sp"/>
    <fragment
        android:id="@+id/map"
        android:name="com.google.android.gms.maps.SupportMapFragment"
        android:layout_width="match_parent"
        android:layout_height="0sp"
        android:layout_weight="1"
        tools:context=".CurrentActivity" />
    <Button
        android:id="@+id/btnAdd"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="맛집등록"/>
</LinearLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<fragment xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:map="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/map"
    android:name="com.google.android.gms.maps.SupportMapFragment"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MapsActivity" />
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="10sp">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="0px"
        android:layout_weight="1"
        android:orientation="vertical">
        <EditText
            android:id="@+id/edtName"
            android:hint="상호명"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"/>
        <EditText
            android:id="@+id/edtAddress"
            android:hint="주소"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:enabled="false"/>
        <EditText
            android:id="@+id/edtTel"
            android:hint="전화번호"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"/>
        <EditText
            android:id="@+id/edtLatitude"
            android:hint="위도"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:enabled="false"/>
        <EditText
            android:id="@+id/edtLongitude"
            android:hint="경도"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:enabled="false"/>
    </LinearLayout>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:gravity="end">
        <Button
            android:id="@+id/btnSave"
            android:text="저장"
            android:layout_width="0px"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:onClick="mClick"/>
        <Button
            android:id="@+id/btnCancel"
            android:text="취소"
            android:layout_width="0px"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:onClick="mClick"/>
    </LinearLayout>
</LinearLayout>
```

```java
public class FoodVO {
    private int seq;
    private String name;
    private String tel;
    private String address;
    private Double latitude;
    private Double longitude;
    ... setter()/getter() 생략
}
```

```java
public interface RemoteService {
    public static final String BASE_URL="http://172.30.1.24:8080/food/";
    @GET("list.jsp")
    Call<List<FoodVO>> listFood();

    @POST("insert.jsp")
    Call<Void> insertFood(@Query("name") String name, @Query("address") String address,
                          @Query("tel") String tel, @Query("latitude") Double latitude,
                          @Query("longitude") Double longitude);
}
```

```java
public class MainActivity extends AppCompatActivity {
    Retrofit retrofit;
    RemoteService remoteService;
    List<FoodVO> arrayFood=new ArrayList<>();
    FoodAdapter foodAdapter=new FoodAdapter();

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

        ListView listFood=findViewById(R.id.listFood);
        listFood.setAdapter(foodAdapter);

        retrofit=new Retrofit.Builder().baseUrl(BASE_URL).addConverterFactory(GsonConverterFactory.create()).build();
        remoteService=retrofit.create(RemoteService.class);

        Call<List<FoodVO>> call=remoteService.listFood();
        call.enqueue(new Callback<List<FoodVO>>() {
            @Override
            public void onResponse(Call<List<FoodVO>> call, Response<List<FoodVO>> response) {
                arrayFood=response.body();
                foodAdapter.notifyDataSetChanged();
            }
            @Override
            public void onFailure(Call<List<FoodVO>> call, Throwable t) {

            }
        });

        Button btnLocation=findViewById(R.id.btnLocation);
        btnLocation.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent=new Intent(MainActivity.this, CurrentActivity.class);
                intent.putExtra("latitude", 36.7989522);
                intent.putExtra("longitude", 127.072742);
                startActivity(intent);
            }
        });
    }
```

```java
class FoodAdapter extends BaseAdapter{
    @Override
    public int getCount() { return arrayFood.size(); }

    @Override
    public Object getItem(int i) { return null; }

    @Override
    public long getItemId(int i) { return 0; }

    @Override
    public View getView(int i, View view, ViewGroup viewGroup) {
        view=getLayoutInflater().inflate(R.layout.item_food, viewGroup,false);

        TextView txtName=view.findViewById(R.id.txtName);
        TextView txtAddress=view.findViewById(R.id.txtTel);
        TextView txtTel=view.findViewById(R.id.txtAddress);

        final FoodVO vo=arrayFood.get(i);
        txtName.setText(vo.getName());
        txtAddress.setText(vo.getAddress());
        txtTel.setText(vo.getTel());

        ImageView btnLocation=view.findViewById(R.id.btnLocation);
        btnLocation.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent=new Intent(MainActivity.this, MapsActivity.class);
                intent.putExtra("latitude", vo.getLatitude());
                intent.putExtra("longitude", vo.getLongitude());
                intent.putExtra("name", vo.getName());
                intent.putExtra("tel", vo.getTel());
                startActivity(intent);
            }
        });
        return view;
    }
}
}
```

[MapsActivity.java]

```java
public class MapsActivity extends FragmentActivity implements OnMapReadyCallback {
    private GoogleMap mMap;
    double latitude;
    double longitude;
    String name, tel;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_maps);
        SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()
                                    .findFragmentById(R.id.map);
        mapFragment.getMapAsync(this);
        Intent intent=getIntent();
        latitude=intent.getDoubleExtra("latitude",0);
        longitude=intent.getDoubleExtra("longitude", 0);
        name=intent.getStringExtra("name");
        tel=intent.getStringExtra("tel");
    }

    @Override
    public void onMapReady(GoogleMap googleMap) {
        mMap = googleMap;
        LatLng latLng = new LatLng(latitude, longitude);
        MarkerOptions mMarker=new MarkerOptions();
        mMarker.position(latLng);
        mMarker.title(name);
        mMarker.snippet(tel);
        mMap.addMarker(mMarker);
        mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));
        mMap.moveCamera(CameraUpdateFactory.zoomTo(16));
    }
}
```

```java
public class CurrentActivity extends FragmentActivity implements OnMapReadyCallback {
    private GoogleMap mMap;
    double latitude;
    double longitude;
    TextView txtAddress;

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

        SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()
                .findFragmentById(R.id.map);
        mapFragment.getMapAsync(this);

        Intent intent=getIntent();
        latitude=intent.getDoubleExtra("latitude",0);
        longitude=intent.getDoubleExtra("longitude", 0);

        txtAddress=findViewById(R.id.txtAddress);
        getAddress(new LatLng(latitude, longitude));

        Button btnAdd=findViewById(R.id.btnAdd);
        btnAdd.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent1=new Intent(CurrentActivity.this, AddActivity.class);
                intent1.putExtra("latitude", latitude);
                intent1.putExtra("longitude", longitude);
                intent1.putExtra("address", txtAddress.getText().toString());
                startActivity(intent1);
            }
        });
    }

    @Override
    public void onMapReady(GoogleMap googleMap) {
        mMap = googleMap;

        LatLng latLng = new LatLng(latitude, longitude);

        mMap.addMarker(new MarkerOptions().position(latLng).title("현재위치"));
        mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));
        mMap.moveCamera(CameraUpdateFactory.zoomTo(16));
        mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));

        mMap.setOnMapClickListener(new GoogleMap.OnMapClickListener() {
            @Override
            public void onMapClick(LatLng latLng) {
                latitude=latLng.latitude;
                longitude=latLng.longitude;

                mMap.clear();
                MarkerOptions mOptions=new MarkerOptions();
                mOptions.title("위치");
                mOptions.position(latLng);
                mOptions.snippet(latitude + "/" + longitude);
                mMap.addMarker(mOptions);

                getAddress(latLng);
            }
        });
    }

    //주소구하기
    public void getAddress(LatLng latLng){
        Geocoder geocoder=new Geocoder(CurrentActivity.this, Locale.KOREAN);
        try{
            List<Address> list=geocoder.getFromLocation(latLng.latitude, latLng.longitude, 1);
            if(!list.isEmpty()){
                String strAddress=list.get(0).getAddressLine(0).toString();
                txtAddress.setText(strAddress);
            }
        }catch(Exception e){ }
    }
}
```

```java
public class AddActivity extends AppCompatActivity {
    EditText name, tel, address, edtLatitude, edtLongitude;
    RemoteService remoteService;
    Retrofit retrofit;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_add);
        getSupportActionBar().setTitle("맛집등록");
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);

        retrofit=new Retrofit.Builder().baseUrl(BASE_URL).addConverterFactory(GsonConverterFactory.create()).build();
        remoteService=retrofit.create(RemoteService.class);

        name=findViewById(R.id.edtName);
        tel=findViewById(R.id.edtTel);
        address=findViewById(R.id.edtAddress);
        edtLatitude=findViewById(R.id.edtLatitude);
        edtLongitude=findViewById(R.id.edtLongitude);

        Intent intent=getIntent();
        address.setText(intent.getStringExtra("address"));
        double longitude=intent.getDoubleExtra("longitude", 0);
        edtLongitude.setText(longitude + "");
        double latitude=intent.getDoubleExtra("latitude", 0);
        edtLatitude.setText(latitude + "");

        Button btnSave=findViewById(R.id.btnSave);
        btnSave.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                AlertDialog.Builder box=new AlertDialog.Builder(AddActivity.this);
                box.setTitle("질의");
                box.setMessage("저장하실래요?");

                box.setPositiveButton("예", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialogInterface, int i) {
                        FoodVO vo=new FoodVO();
                        vo.setAddress(address.getText().toString());
                        vo.setName(name.getText().toString());
                        vo.setTel(tel.getText().toString());

                        String strLatitude=edtLatitude.getText().toString();
                        vo.setLatitude(Double.parseDouble(strLatitude));

                        String strLongitude=edtLongitude.getText().toString();
                        vo.setLongitude(Double.parseDouble(strLongitude));

                        if(vo.getName().equals("") || vo.getTel().equals("")){
                            Toast.makeText(AddActivity.this, 상호명과 전화번호를 입력하세요!",Toast.LENGTH_SHORT).show();
                        }else{
                            Call<Void> call=remoteService.insertFood(vo.getName(),
                                    vo.getAddress(),vo.getTel(),vo.getLatitude(), vo.getLongitude());
                            call.enqueue(new Callback<Void>() {
                                @Override
                                public void onResponse(Call<Void> call, Response<Void> response) {
                                    Toast.makeText(AddActivity.this,"저장완료",Toast.LENGTH_SHORT).show();
                                    finish();
                                    Intent intent=new Intent(AddActivity.this, MainActivity.class);
                                    startActivity(intent);
                                }
                                @Override
                                public void onFailure(Call<Void> call, Throwable t) {
                                }
                            });
                        }
                    }
                });
                box.setNegativeButton("아니오", null);
                box.show();
            }
        });
    }
}
```

3-4. 학사 관리 프로그램

▪ 학사 관리 웹프로그램

• 테스트: http://localhost:8080/haksa/list_students.jsp

[WebContent]-[list_students.jsp]

```jsp
<%@page import="java.sql.*, org.json.simple.*"%>
<%
    response.setCharacterEncoding("UTF-8");
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/haksa", "root", "1234");

        String sql="select * from students order by dept, scode";
        PreparedStatement ps=con.prepareStatement(sql);
        ResultSet rs=ps.executeQuery();

        JSONArray array=new JSONArray();
        while(rs.next()){
            JSONObject object=new JSONObject();
            object.put("dept", rs.getString("dept"));
            object.put("scode", rs.getString("scode"));
            object.put("sname", rs.getString("sname"));
            array.add(object);
        }
        out.print(array.toJSONString());

    }catch(Exception e){
        System.out.println(e.toString());
        response.setStatus(400);
    }
%>
```

• 테스트: http://localhost:8080/haksa/list_enrollments.jsp?scode=92454018

[WebContent]-[list_enrollments.jsp]

```jsp
<%@page import="java.sql.*, org.json.simple.*"%>
<%
    response.setCharacterEncoding("UTF-8");
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/haksa", "root", "1234");

        String scode=request.getParameter("scode");
        String sql="select e.scode, e.lcode, c.lname, e.edate from enrollments e, courses c where e.lcode=c.lcode and
                    scode=? order by e.lcode";

        PreparedStatement ps=con.prepareStatement(sql);
        ps.setString(1, scode);
        ResultSet rs=ps.executeQuery();

        JSONArray array=new JSONArray();
        while(rs.next()){
            JSONObject object=new JSONObject();
            object.put("lcode", rs.getString("e.lcode"));
            object.put("lname", rs.getString("c.lname"));
            object.put("edate", rs.getString("e.edate"));
            array.add(object);
        }
        out.print(array.toJSONString());
    }catch(Exception e){
        System.out.println(e.toString());
        response.setStatus(400);
    }
%>
```

- 테스트: http://localhost:8080/haksa/list_courses.jsp

[WebContent]-[list_courses.jsp]

```
<%@page import="java.sql.*, org.json.simple.*"%>
<%
    response.setCharacterEncoding("UTF-8");
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/haksa", "root", "1234");

        String sql="select * from courses order by lcode";
        PreparedStatement ps=con.prepareStatement(sql);
        ResultSet rs=ps.executeQuery();

        JSONArray array=new JSONArray();
        while(rs.next()){
            JSONObject object=new JSONObject();
            object.put("lcode", rs.getString("lcode"));
            object.put("lname", rs.getString("lname"));
            array.add(object);
        }
        out.print(array.toJSONString());
    }catch(Exception e){
        System.out.println(e.toString());
        response.setStatus(400);
    }
%>
```

- 테스트: http://localhost:8080/haksa/insert_enrollments.jsp?scode=92454018&lcode=E221

[WebContent]-[insert_enrollments.jsp]
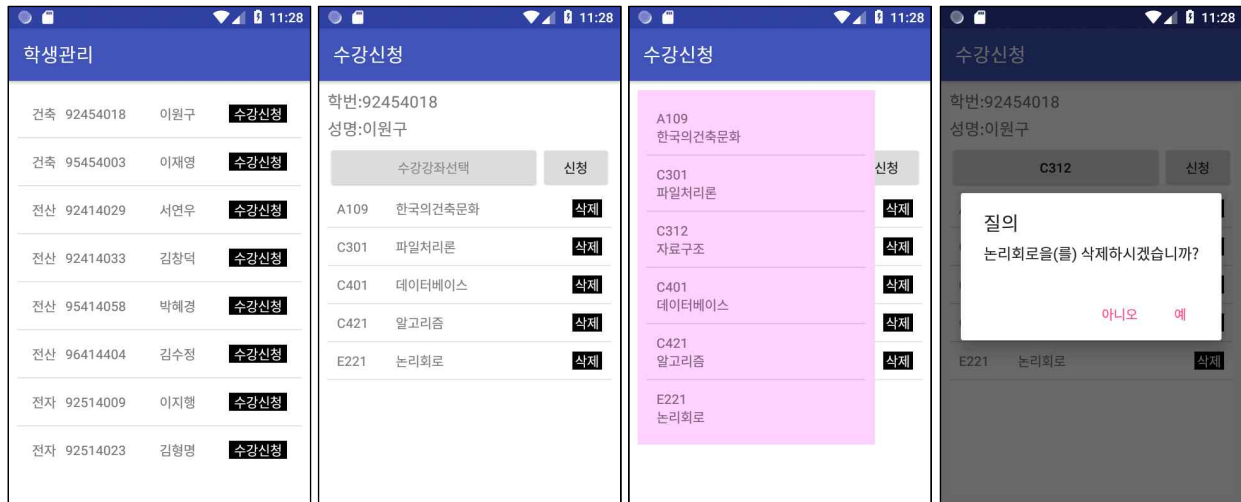
```
<%@page import="java.sql.*, java.util.Date, java.text.*"%>
<%
    request.setCharacterEncoding("UTF-8");
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/haksa", "root", "1234");

        String scode=request.getParameter("scode");
        String lcode=request.getParameter("lcode");
        Date today=new Date();
        SimpleDateFormat sdf=new SimpleDateFormat("yyyy-MM-dd");
        String edate=sdf.format(today);

        String sql="insert into enrollments(scode, lcode, edate, grade) values(?,?,?,0)";
        PreparedStatement ps=con.prepareStatement(sql);
        ps.setString(1, scode);
        ps.setString(2, lcode);
        ps.setString(3, edate);
        ps.execute();
    }catch(Exception e){
        System.out.println(e.toString());
        response.setStatus(400);
    }
%>
```

- 테스트: http://localhost:8080/haksa/delete_enrollments.jsp?lcode=E221

[WebContent]-[delete_enrollments.jsp]

```
<%@page import="java.sql.*, java.util.Date, java.text.*"%>
<%
    request.setCharacterEncoding("UTF-8");
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/haksa", "root", "1234");

        String scode=request.getParameter("scode");
        String lcode=request.getParameter("lcode");

        String sql="delete from enrollments where scode=? and lcode=?";
        PreparedStatement ps=con.prepareStatement(sql);
        ps.setString(1, scode);
        ps.setString(2, lcode);
        ps.execute();
    }catch(Exception e){
        System.out.println(e.toString());
        response.setStatus(400);
    }
%>
```

- 학사 관리 앱프로그램

[build.gradle (Module:app)]

```
dependencies {
    …
    implementation 'com.squareup.retrofit2:retrofit:2.2.0'
    implementation 'com.squareup.retrofit2:converter-gson:2.2.0'
}
```

AndroidManifest.xml

```
<uses-permission android:name="android.permission.INTERNET" />
```

[res]-[layout]-[activity_main.xml]

```
<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:orientation="vertical"
    android:padding="10sp">
    <ListView
        android:id="@+id/list"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
</android.support.constraint.ConstraintLayout>
```

[res]-[layout]-[item_students.xml]

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"  android:layout_height="match_parent"  android:padding="15sp">
    <TextView
        android:id="@+id/dept"  android:text="학과"
        android:layout_width="0px"  android:layout_height="wrap_content"  android:layout_weight="1"/>
    <TextView
        android:id="@+id/scode"  android:text="학번"
        android:layout_width="wrap_content"  android:layout_height="wrap_content"  android:layout_weight="1"/>
    <TextView
        android:id="@+id/sname"  android:text="성명"
        android:layout_width="wrap_content"  android:layout_height="wrap_content"  android:layout_weight="1"/>
    <TextView
        android:id="@+id/btn"  android:text="수강신청"
        android:layout_width="wrap_content" android:layout_height="wrap_content"  android:layout_weight="0.3"
        android:background="#000000"  android:textColor="#FFFFFF"
        android:gravity="center"/>
</LinearLayout>
```

```xml
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"  android:layout_height="match_parent"  android:padding="10sp">
    <LinearLayout
        android:id="@+id/enrollments"
        android:layout_width="match_parent"  android:layout_height="match_parent"
        android:background="#FFFFFF"  android:orientation="vertical">
        <TextView
            android:id="@+id/scode"  android:text="학번"
            android:layout_width="wrap_content"  android:layout_height="wrap_content"
            android:textSize="17sp"  android:layout_marginBottom="5sp"/>
        <TextView
            android:id="@+id/sname"  android:text="성명"
            android:layout_width="wrap_content"  android:layout_height="wrap_content"
            android:textSize="17sp"  android:layout_marginBottom="5sp"/>
        <LinearLayout
            android:layout_width="match_parent"  android:layout_height="wrap_content"  android:orientation="horizontal">
            <Button
                android:id="@+id/btnCourses"  android:hint="수강강좌선택"
                android:layout_width="0px"  android:layout_height="wrap_content"  android:layout_weight="3"
                android:onClick="mClick"/>
            <Button
                android:id="@+id/btnEnroll"  android:text="신청"
                android:layout_width="0px"  android:layout_height="wrap_content"  android:layout_weight="1"
                android:onClick="mClick"/>
        </LinearLayout>
        <ListView
            android:id="@+id/list"
            android:layout_width="match_parent"  android:layout_height="0px"  android:layout_weight="1"/>
    </LinearLayout>
    <LinearLayout
        android:id="@+id/fragmentCourses"
        android:layout_width="250sp"  android:layout_height="wrap_content"
        android:orientation="vertical"  android:background="#FFCCFF">
    </LinearLayout>
</FrameLayout>
```

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"  android:layout_height="match_parent"
    android:orientation="horizontal"  android:padding="10sp">
    <TextView
        android:id="@+id/lcode"  android:text="강좌번호"
        android:layout_width="0px"  android:layout_height="wrap_content"  android:layout_weight="1"/>
    <TextView
        android:id="@+id/lname"  android:text="강좌번호"
        android:layout_width="0px"  android:layout_height="wrap_content"  android:layout_weight="3"/>
    <TextView
        android:id="@+id/btnDelete"  android:text="삭제"
        android:layout_width="0px"  android:layout_height="wrap_content"  android:layout_weight="0.5"
        android:background="#000000"  android:textColor="#ffffff"
        android:gravity="center"/>
</LinearLayout>
```

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"  android:layout_height="match_parent"
    android:orientation="vertical"  android:padding="10sp">
    <ListView
        android:id="@+id/list"
        android:layout_width="match_parent"  android:layout_height="match_parent"/>
</LinearLayout>
```

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"  android:layout_height="match_parent"
    android:orientation="vertical"  android:padding="10sp">
    <ListView
        android:id="@+id/list"
        android:layout_width="match_parent"  android:layout_height="match_parent"/>
</LinearLayout>
```

```java
public class StudentsVO {
    private String scode;
    private String sname;
    private String dept;
    ...
    setter()/getter() 메서드 생략
}
```

```java
public class EnrollmentsVO {
    private String lcode;
    private String lname;
    private String edate;
    ...
    setter()/getter() 메서드 생략
}
```

```java
public class CoursesVO {
    private String lcode;
    private String lname;
    ...
    setter()/getter() 메서드 생략
}
```

```java
public interface RemoteService {
    public static final String BASE_URL="http://172.30.1.52:8080/haksa/";

    @GET("list_students.jsp")
    Call<List<StudentsVO>> listStudents();

    @GET("list_courses.jsp")
    Call<List<CoursesVO>> listCourses();

    @GET("list_enrollments.jsp")
    Call<List<EnrollmentsVO>> listEnrollments(@Query("scode") String scode);

    @POST("insert_enrollments.jsp")
    Call<Void> insertEnrollments(@Query("scode") String scode, @Query("lcode") String lcode);

    @POST("delete_enrollments.jsp")
    Call<Void> deleteEnrollments(@Query("scode") String scode, @Query("lcode") String lcode);
}
```

```java
public class MainActivity extends AppCompatActivity {
    Retrofit retrofit;
    RemoteService rs;
    MyAdapter adapter;
    List<StudentsVO> students;
    ListView list;

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

        adapter=new MyAdapter();
        list=(ListView)findViewById(R.id.list);

        retrofit = new Retrofit.Builder()
                .baseUrl(BASE_URL)
                .addConverterFactory(GsonConverterFactory.create())
                .build();
        rs = retrofit.create(RemoteService.class);
    }

    protected void onResume() {
        Call<List<StudentsVO>> call = rs.listStudents();
        call.enqueue(new Callback<List<StudentsVO>>() {
```

```
                public void onResponse(Call<List<StudentsVO>> call, Response<List<StudentsVO>> response) {
                    students = response.body();
                    System.out.println("학생 데이타갯수:" + students.size());
                    adapter.notifyDataSetChanged();
                    list.setAdapter(adapter);
                }
                public void onFailure(Call<List<StudentsVO>> call, Throwable t) { }
            });
            super.onResume();
        }

    class MyAdapter extends BaseAdapter{
        public int getCount() { return students.size(); }
        public Object getItem(int position) { return null; }
        public long getItemId(int position) { return 0; }

        public View getView(int position, View convertView, ViewGroup parent) {
            LayoutInflater inflater=getLayoutInflater();
            if(convertView==null){
                convertView=inflater.inflate(R.layout.item_students, null);
            }

            final StudentsVO vo=students.get(position);

            TextView dept=(TextView)convertView.findViewById(R.id.dept);
            dept.setText(vo.getDept());
            TextView scode=(TextView)convertView.findViewById(R.id.scode);
            scode.setText(vo.getScode());
            TextView sname=(TextView)convertView.findViewById(R.id.sname);
            sname.setText(vo.getSname());
            TextView btn=(TextView)convertView.findViewById(R.id.btn);

            btn.setOnClickListener(new View.OnClickListener() {
                public void onClick(View v) {
                    Intent intent=new Intent(MainActivity.this, EnrollmentsActivity.class);
                    intent.putExtra("scode", vo.getScode());
                    intent.putExtra("sname", vo.getSname());
                    startActivity(intent);
                }
            });
            return convertView;
        }
    }
}
```

```
public class EnrollmentsActivity extends AppCompatActivity {
    Retrofit retrofit;
    RemoteService rs;
    List<CoursesVO> courses;
    List<EnrollmentsVO> enrollments;
    TextView scode, sname;
    String strScode, strSname;
    EnrollAdapter enrollAdapter;
    ListView list;
    Button btnCourses;

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

        scode=(TextView)findViewById(R.id.scode);
        sname=(TextView)findViewById(R.id.sname);
        btnCourses=(Button)findViewById(R.id.btnCourses);

        Intent intent=getIntent();
        strScode=intent.getStringExtra("scode");
        strSname=intent.getStringExtra("sname");
        scode.setText("학번:" + strScode);
        sname.setText("성명:" + strSname);

        retrofit = new Retrofit.Builder()
                .baseUrl(BASE_URL)
                .addConverterFactory(GsonConverterFactory.create())
                .build();
        rs = retrofit.create(RemoteService.class);
```

```java
        enrollAdapter=new EnrollAdapter();
        list=(ListView)findViewById(R.id.list);
}

public void mClick(View v){
    switch (v.getId()){
        case R.id.btnCourses:
            FragmentManager fm=getFragmentManager();
            Fragment fragment=fm.findFragmentById(R.id.fragmentCourses);
            if(fragment==null) {
                FragmentTransaction tr = fm.beginTransaction();
                CoursesFragment cf = new CoursesFragment();
                tr.add(R.id.fragmentCourses, cf, "courses");
                tr.commit();
            }
            break;
        case R.id.btnEnroll:
            String strLcode=btnCourses.getText().toString();
            System.out.println("Lcode:" + strLcode + ", Scode:" + strScode);
            Call<Void> call=rs.insertEnrollments(strScode, strLcode);

            call.enqueue(new Callback<Void>() {
                public void onResponse(Call<Void> call, Response<Void> response) {
                    Toast.makeText(EnrollmentsActivity.this, "신청되었습니다.", Toast.LENGTH_SHORT).show();
                    onResume();
                }
                public void onFailure(Call<Void> call, Throwable t) { }
            });
        }
}

protected void onResume() {
    Call<List<EnrollmentsVO>> call = rs.listEnrollments(strScode);
    call.enqueue(new Callback<List<EnrollmentsVO>>() {
        public void onResponse(Call<List<EnrollmentsVO>> call, Response<List<EnrollmentsVO>> response) {
            enrollments=response.body();
            System.out.println("수강신청 데이타갯수:" + enrollments.size());
            enrollAdapter.notifyDataSetChanged();
            list.setAdapter(enrollAdapter);
        }
        public void onFailure(Call<List<EnrollmentsVO>> call, Throwable t) { }
    });
    super.onResume();
}

class EnrollAdapter extends BaseAdapter{
    public int getCount() { return enrollments.size(); }
    public Object getItem(int position) { return null; }
    public long getItemId(int position) { return 0; }

    public View getView(int position, View convertView, ViewGroup parent) {
        LayoutInflater inflater=getLayoutInflater();
        if(convertView==null){
            convertView=inflater.inflate(R.layout.item_enrollments, null);
        }
        final EnrollmentsVO vo=enrollments.get(position);

        TextView lcode=(TextView)convertView.findViewById(R.id.lcode);
        lcode.setText(vo.getLcode());
        TextView lname=(TextView)convertView.findViewById(R.id.lname);
        lname.setText(vo.getLname());

        TextView btnDelete=(TextView)convertView.findViewById(R.id.btnDelete);
        btnDelete.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                AlertDialog.Builder box=new AlertDialog.Builder(EnrollmentsActivity.this);
                box.setTitle("질의");
                box.setMessage(vo.getLname()+"을(를) 삭제하시겠습니까?");

                box.setPositiveButton("예", new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog, int which) {
                        Call<Void> call=rs.deleteEnrollments(strScode, vo.getLcode());

                        call.enqueue(new Callback<Void>() {
                            public void onResponse(Call<Void> call, Response<Void> response) {
                                Toast.makeText(EnrollmentsActivity.this, "삭제되었습니다.", Toast.LENGTH_SHORT).show();
                                onResume();
                            }
                            public void onFailure(Call<Void> call, Throwable t) { }
```

```
                });
            }
        });
        box.setNegativeButton("아니오", null);
        box.show();
        }
    });
    return convertView;
    }
  }
}
```

```
public class CoursesFragment extends Fragment{
    Retrofit retrofit;
    RemoteService rs;
    CoursesAdapter coursesAdapter;
    ListView list;
    List<CoursesVO> courses;

    @Nullable
    @Override
    public View onCreateView(LayoutInflater inflater, @Nullable ViewGroup container, Bundle savedInstanceState) {
        LinearLayout layout=(LinearLayout)inflater.inflate(R.layout.flagment_courses, container, false);

        retrofit = new Retrofit.Builder()
                .baseUrl(BASE_URL)
                .addConverterFactory(GsonConverterFactory.create())
                .build();
        rs = retrofit.create(RemoteService.class);

        coursesAdapter=new CoursesAdapter();
        list=(ListView)layout.findViewById(R.id.list);

        Call<List<CoursesVO>> call = rs.listCourses();
        call.enqueue(new Callback<List<CoursesVO>>() {
            public void onResponse(Call<List<CoursesVO>> call, Response<List<CoursesVO>> response) {
                courses=response.body();
                System.out.println("강좌 데이타갯수:" + courses.size());
                coursesAdapter.notifyDataSetChanged();
                list.setAdapter(coursesAdapter);
            }
            public void onFailure(Call<List<CoursesVO>> call, Throwable t) { }
        });
        return layout;
    }

    class CoursesAdapter extends BaseAdapter{
        public int getCount() { return courses.size(); }
        public Object getItem(int position) { return null; }
        public long getItemId(int position) {return 0; }

        public View getView(int position, View convertView, ViewGroup parent) {
            LayoutInflater inflater=getActivity().getLayoutInflater();
            if(convertView==null){
                convertView=inflater.inflate(R.layout.item_courses, parent, false);
            }
            final CoursesVO vo=courses.get(position);
            TextView lcode=(TextView)convertView.findViewById(R.id.lcode);
            lcode.setText(vo.getLcode());
            TextView lname=(TextView)convertView.findViewById(R.id.lname);
            lname.setText(vo.getLname());

            convertView.setOnClickListener(new View.OnClickListener() {
                public void onClick(View v) {
                    Button btnCourses=(Button)getActivity().findViewById(R.id.btnCourses);
                    btnCourses.setText(vo.getLcode());
                    getActivity().getFragmentManager().beginTransaction().remove(CoursesFragment.this).commit();
                }
            });
            return convertView;
        }
    }
}
```

4-1. 사용자관리 프로그램 (Node.js)

• 데이터베이스 생성하기

Workbench를 실행한 후 데이터베이스와 테이블을 생성하고 Sample 데이터를 입력한다.

tbl_user 테이블 생성

```
createdatabase webdb;
use webdb;

create table tbl_user(
  uid varchar(50) not null,
  upw varchar(50) not null,
  uname varchar(100) not null,
  primary key(uid)
);
```

tbl_user 데이터 입력

```
insert into tbl_user(uid, upw, uname) values('user00', 'pass', '홍길동');
insert into tbl_user(uid, upw, uname) values('user01', 'pass', '심청이');
insert into tbl_user(uid, upw, uname) values('user02', 'pass', '강감찬');
insert into tbl_user(uid, upw, uname) values('user03', 'pass', '이순신');
insert into tbl_user(uid, upw, uname) values('user04', 'pass', '이몽룡');
```

• 노드 환경 구성하기

1) 프로젝트 디렉터리를 생성한 후 생성된 디렉터리로 이동한다.

```
C:₩project>
```

2) 익스프레스 프로젝트를 쉽게 생성해 주는 express-generator 모듈을 전역으로 설치한다. 이렇게 express-generator를 사용해서 프로젝트를 생성하면 기본적인 모듈들이 자동으로 pakage.json에 추가된다.

```
C:₩project>npm install express-generator -g
```

3) user 디렉터리에서 아래 명령어를 실행해서 프로젝트 디렉터리를 생성한다. 이때 템플릿 엔진은 ejs로 설치할 것이므로 -e 옵션을 추가한다.

```
C:₩project>express -e web
```

4) c:₩project₩web₩public 디렉터리에 있는 디렉터리 이름을 다음처럼 변경한다. 긴 이름을 그대로 사용해도 되지만 개발할 때 조금이나마 편하려면 이름을 변경하는 것이 좋다.

```
images -> img
javascripts -> js
stylesheets -> css
```

5) package.json에 추가된 모듈을 설치한다. npm install 명령어를 사용하면 package.json에 선언된 모듈이 자동으로 설치된다. 이때 명령어는 web 디렉터리 아래에서 실행한다.

```
C:₩project₩web>npm install
```

6) 추가로 필요한 모듈을 설치한다. 이때는 설치 옵션에 --save를 주어 package.json에 설치된 모듈을 추가해야한다. 이렇게 추가해 놓으면 어떤 모듈을 사용하고 있는지 알 수 있으며 나중에 동일한 프로젝트를 다른 곳에 생성할 때 pakage.json을 복사해서 설치하면 매우 편리하다. 파일 업로드 모듈인 formidable과 mysql 연결 모듈인 mysql을 함께 설치하자.

```
C:₩project₩web>npm install formidable mysql --save
```

• 노드 실행하기

1) 노드는 npm start 명령어를 사용해서 실행할 수 있다. 참고로, 실행을 종료하고 싶다면 CTRL + C를 입력하면 된다.

```
C:\project\web>npm start
```

2) 결과는 브라우저에 http://localhost:3000으로 접속해서 확인할 수 있다. 노드는 기본적으로 3000번 포트에서 동작하도록 되어 있다. 다른 포트 번호로 변경하려면 c:\data\user\web\bin 디렉터리의 www파일의 내용을 수정해야 한다.

• 노드 서버 프로그램 작성 (입력, 수정 출력)

```
C:/project/web/app.js에 아래내용 추가
```

```
...

app.use('/user', require('./routes/user'));

...
```

```
C:/project/web/routes/user.js 생성
```

```javascript
var express = require("express");
var router = express.Router();

//데이터베이스 연결
var mysql=require('mysql');

var db=mysql.createPool({
    connectionLimit:100,
    host:'localhost',
    user:'root',
    password:'1234',
    database:'webdb'
});

//전체사용자 출력
router.get("/list", function(req, res) {
    var sql="select * from tbl_user";

    db.query(sql, function(err, rows){
        console.log("rows:" + rows.length);
        res.status(200).json(rows);
    });
});
//특정사용자 출력
router.get("/read/:uid", function(req, res) {
    var uid=req.params.uid;
    var sql="select * from tbl_user where uid=?";

    db.query(sql, [uid], function(err, rows){
        res.status(200).json(rows[0]);
    });
});

//사용자 등록
router.post("/insert", function(req, res){
    var uid=req.body.uid;
    var upw=req.body.upw;
    var uname=req.body.uname;
    var sql="insert into tbl_user(uid, upw, uname) values(?, ?, ?)";

    db.query(sql,[uid, upw, uname], function(err, result){
        if(err){
            return res.sendStatus(400);
        }else{
            return res.sendStatus(200);
        }
    });
});
```

```javascript
//사용자 삭제
router.post('/delete', function(req, res){
    var uid=req.body.uid;
    var sql="delete from tbl_user where uid=?";

    db.query(sql,[uid], function(err, result){
        if(err){
            return res.sendStatus(400);
        }else{
            return res.sendStatus(200);
        }
    });
});

//사용자 수정
router.post("/update", function(req, res){
    var uid=req.body.uid;
    var upw=req.body.upw;
    var uname=req.body.uname;
    var sql="update tbl_user set upw=?, uname=? where uid=?";

    db.query(sql,[upw, uname, uid], function(err, result){
        if(err){
            return res.sendStatus(400);
        }else{
            return res.sendStatus(200);
        }
    });
});

//외부에서 사용가능하도록 지정
module.exports = router;
```
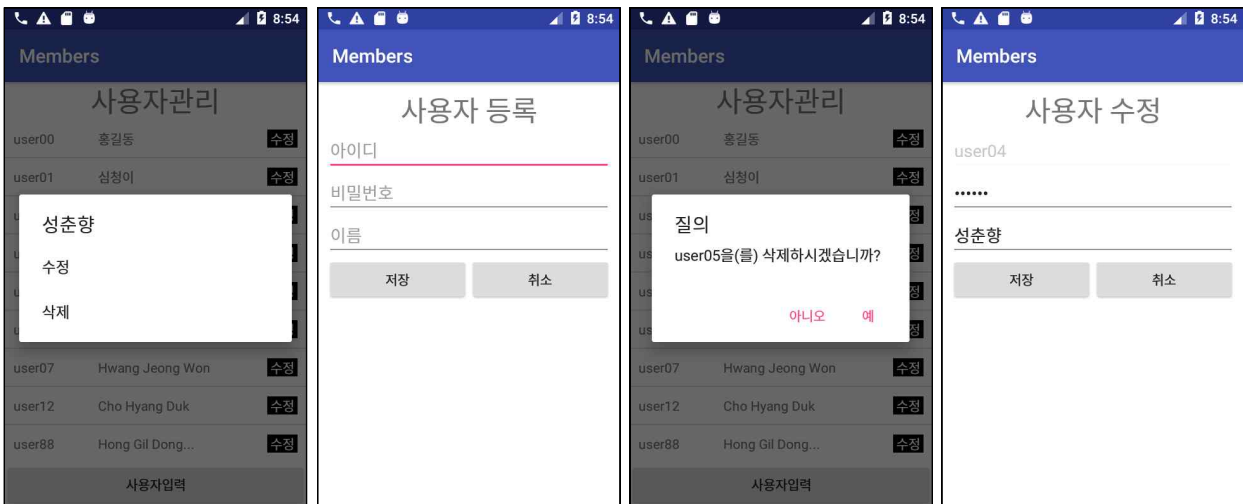
- **안드로이드 프로그래밍**

---

activity_main.xml

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="사용자관리"
        android:textSize="30sp"
        android:layout_gravity="center"/>
    <ListView
        android:id="@+id/list"
```

```xml
            android:layout_width="match_parent"
            android:layout_height="0px"
            android:layout_weight="1"/>
        <Button
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="사용자입력"
            android:onClick="mClick"/>
    </LinearLayout>
```

item.xml

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:padding="10sp">
    <TextView
        android:id="@+id/uid"
        android:layout_width="0px"
        android:layout_weight="1"
        android:layout_height="wrap_content"
        android:text="아이디" />
    <TextView
        android:id="@+id/uname"
        android:layout_width="0px"
        android:layout_weight="2"
        android:layout_height="wrap_content"
        android:text="성명" />
    <TextView
        android:id="@+id/btnUpdate"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text=" 수정 "
        android:gravity="center"
        android:background="#000000"
        android:textColor="#FFFFFF"/>
</LinearLayout>
```

activity_insert.xml

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="10sp">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="사용자 등록"
        android:textSize="30sp"
        android:layout_gravity="center"/>
    <EditText
        android:id="@+id/uid"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="아이디"/>
    <EditText
        android:id="@+id/upw"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textPassword"
        android:hint="비밀번호"/>
    <EditText
        android:id="@+id/uname"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="이름"/>
```

```xml
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <Button
            android:id="@+id/btnInsert"
            android:layout_width="0px"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="저장"
            android:onClick="mClick"/>
        <Button
            android:id="@+id/btnCancel"
            android:layout_width="0px"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="취소"
            android:onClick="mClick"/>
    </LinearLayout>
</LinearLayout>
```

activity_update.xml

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="10sp">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="사용자 수정"
        android:textSize="30sp"
        android:layout_gravity="center"/>
    <EditText
        android:id="@+id/uid"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:enabled="false"/>
    <EditText
        android:id="@+id/upw"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textPassword"/>
    <EditText
        android:id="@+id/uname"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <Button
            android:id="@+id/btnInsert"
            android:layout_width="0px"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="저장"
            android:onClick="mClick"/>
        <Button
            android:id="@+id/btnCancel"
            android:layout_width="0px"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="취소"
            android:onClick="mClick"/>
    </LinearLayout>
</LinearLayout>
```

```java
public class GetHttpURL extends AsyncTask<String, String, String> {
    @Override
    protected String doInBackground(String... strings) {
        String baseURL="http://172.30.1.4:8080/user";
        try {
            String apiURL = baseURL + strings[0];

            URL url = new URL(apiURL);
            HttpURLConnection con = (HttpURLConnection) url.openConnection();
            con.setRequestMethod("GET");

            BufferedReader br;
            if ( con.getResponseCode() == 200) {
                br = new BufferedReader(new InputStreamReader((con.getInputStream()), "UTF-8"));
            } else {
                br = new BufferedReader(new InputStreamReader(con.getErrorStream()));
            }

            String inputLine;
            StringBuffer response = new StringBuffer();
            while ((inputLine = br.readLine()) != null) {
                response.append(inputLine);
            }
            br.close();

            System.out.println("result=" + response.toString());
            return response.toString();
        } catch (Exception e) {
            return e.toString();
        }
    }
}
```

```java
public class PostHttpURL extends AsyncTask<String, String, String>{
    @Override
    protected String doInBackground(String... strings) {
        String baseURL = "http://172.30.1.4:8080/user";
        try {
            String apiURL = baseURL + strings[0];
            String strParams = strings[1];

            URL url = new URL(apiURL);
            HttpURLConnection con = (HttpURLConnection) url.openConnection();
            con.setRequestMethod("POST");

            OutputStream os=con.getOutputStream();
            os.write(strParams.getBytes("UTF-8"));
            os.flush();
            os.close();

            System.out.println("결과코드:" + con.getResponseCode());
            if(con.getResponseCode() == 200) {
                return "SUCCESS";
            }else {
                return "FAIL";
            }
        } catch (Exception e) {
            return e.toString();
        }
    }
}
```

```java
public class MainActivity extends AppCompatActivity {
    ArrayList<HashMap<String, String>> array;
    MyAdapter adapter;

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

        try {
            String result = new GetHttpURL().execute("/list.jsp").get();
            parserUser(result);
            adapter=new MyAdapter();

        }catch(Exception e){}

        ListView list=(ListView)findViewById(R.id.list);
        list.setAdapter(adapter);
        registerForContextMenu(list);
    }

    public void mClick(View v){
        Intent intent=new Intent(this, InsertActivity.class);
        startActivity(intent);
    }

    @Override
    public void onCreateContextMenu(ContextMenu menu, View v, ContextMenu.ContextMenuInfo menuInfo) {
        int position = ((AdapterView.AdapterContextMenuInfo)menuInfo).position;
        HashMap<String, String> map=array.get(position);

        String strUname=map.get("uname");
        menu.setHeaderTitle(strUname);
        menu.add(0, 1, 0, "수정");
        menu.add(0, 2, 0, "삭제");
        super.onCreateContextMenu(menu, v, menuInfo);
    }

    @Override
    public boolean onContextItemSelected(MenuItem item) {
        AdapterView.AdapterContextMenuInfo info=(AdapterView.AdapterContextMenuInfo)item.getMenuInfo();
        int position=info.position;

        HashMap<String, String> map=array.get(position);
        final String strUid=map.get("uid");

        switch(item.getItemId()){
            case 1:
                Intent intent=new Intent(this, UpdateActivity.class);
                intent.putExtra("uid", strUid);
                startActivity(intent);
                break;
            case 2:
                AlertDialog.Builder box=new AlertDialog.Builder(this);
                box.setTitle("질의");
                box.setMessage(strUid + "을(를) 삭제하시겠습니까?");
                box.setPositiveButton("예", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialog, int which) {
                        PostHttpURL postHttpURL = new PostHttpURL();
                        try {
                            StringBuffer params = new StringBuffer();
                            params.append("uid=" + strUid);
                            String strParams = params.toString();

                            String result = postHttpURL.execute("/delete.jsp", strParams).get();
                            if(result=="SUCCESS"){
                                Toast.makeText(MainActivity.this, "삭제되었습니다!", Toast.LENGTH_SHORT).show();
```

```
                    }
                    try {
                        result = new GetHttpURL().execute("/list.jsp").get();
                        parserUser(result);
                        adapter.notifyDataSetChanged();
                    }catch(Exception e){}

                }catch (Exception e){}
            }
        });
        box.setNegativeButton("아니오", null);
        box.show();
        break;
    }
    return super.onContextItemSelected(item);
}


@Override
protected void onResume() {
    try {
        String result = new GetHttpURL().execute("/list.jsp").get();
        parserUser(result);
        adapter.notifyDataSetChanged();
    }catch(Exception e){}
    super.onResume();
}

class MyAdapter extends BaseAdapter{
    @Override
    public int getCount() {
        return array.size();
    }
    @Override
    public Object getItem(int position) {
        return null;
    }
    @Override
    public long getItemId(int position) {
        return 0;
    }
    @Override
    public View getView(int position, View convertView, ViewGroup parent) {
        LayoutInflater inflater=getLayoutInflater();
        if(convertView==null){
            convertView=inflater.inflate(R.layout.item, parent, false);
        }
        HashMap<String, String> map=new HashMap<String, String>();
        map=array.get(position);

        final String strUid=map.get("uid");
        String strUname=map.get("uname");
        TextView uid=(TextView)convertView.findViewById(R.id.uid);
        TextView uname=(TextView)convertView.findViewById(R.id.uname);
        TextView btnUpdate=(TextView)convertView.findViewById(R.id.btnUpdate);

        uid.setText(strUid);
        uname.setText(strUname);

        btnUpdate.setOnClickListener(new View.OnClickListener(){
            @Override
            public void onClick(View v) {
                Intent intent=new Intent(MainActivity.this, UpdateActivity.class);
                intent.putExtra("uid", strUid);
                startActivity(intent);
            }
        });
        return convertView;
    }
}
public void parserUser(String result) {
    array = new ArrayList<HashMap<String, String>>();
```

```
            try {
                JSONArray jArray = new JSONArray(result);
                for (int i = 0; i < jArray.length(); i++) {
                    JSONObject data = jArray.getJSONObject(i);
                    String uid = data.getString("uid");
                    String upw = data.getString("upw");
                    String uname = data.getString("uname");
                    HashMap<String, String> map = new HashMap<String, String>();
                    map.put("uid", uid);
                    map.put("upw", upw);
                    map.put("uname", uname);
                    array.add(map);
                }
            } catch (Exception e){}
    }
}
```

```
public class InsertActivity extends AppCompatActivity {
    EditText uid, upw, uname;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_insert);

        uid=(EditText)findViewById(R.id.uid);
        upw=(EditText)findViewById(R.id.upw);
        uname=(EditText)findViewById(R.id.uname);
    }
    public void mClick(View v){
        switch(v.getId()) {
            case R.id.btnInsert:
                String strUid = uid.getText().toString();
                String strUpw = upw.getText().toString();
                String strUname = uname.getText().toString();
                try {
                    StringBuffer params = new StringBuffer();
                    params.append("uid=" + strUid);
                    params.append("&upw=" + strUpw);
                    params.append("&uname=" + strUname);
                    String strParams = params.toString();
                    String result = new PostHttpURL().execute("/register.jsp", strParams).get();
                    if (result=="SUCCESS") {
                        Toast.makeText(this, "저장되었습니다!", Toast.LENGTH_SHORT).show();
                    }else{
                        Toast.makeText(this, "저장되지 않았습니다!", Toast.LENGTH_SHORT).show();
                    }
                    finish();
                } catch (Exception e){}
            case R.id.btnCancel:
                finish();
        }
    }
}
```

```
public class UpdateActivity extends AppCompatActivity {
    EditText uid, upw, uname;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_update);
        Intent intent=getIntent();
        String strUid=intent.getStringExtra("uid");

        uid = (EditText)findViewById(R.id.uid);
        upw = (EditText)findViewById(R.id.upw);
        uname = (EditText)findViewById(R.id.uname);
```

```
                uid.setText(strUid);
                try {
                    String result = new GetHttpURL().execute("/read.jsp?uid=" + strUid).get();
                    JSONObject object = new JSONObject(result);
                    String strUpw= object.getString("upw");
                    String strUname = object.getString("uname");

                    upw.setText(strUpw);
                    uname.setText(strUname);
                }catch(Exception e){}
            }
            public void mClick(View v){
                switch(v.getId()) {
                    case R.id.btnInsert:
                        AlertDialog.Builder box=new AlertDialog.Builder(this);
                        box.setTitle("질의");
                        box.setMessage("수정하시겠습니까?");
                        box.setNegativeButton("아니오", null);
                        box.setPositiveButton("예", new DialogInterface.OnClickListener() {
                            @Override
                            public void onClick(DialogInterface dialog, int which) {
                                String strUid = uid.getText().toString();
                                String strUpw = upw.getText().toString();
                                String strUname = uname.getText().toString();
                                try {
                                    StringBuffer params = new StringBuffer();
                                    params.append("uid=" + strUid);
                                    params.append("&upw=" + strUpw);
                                    params.append("&uname=" + strUname);
                                    String strParams = params.toString();
                                    String result = new PostHttpURL().execute("/update.jsp", strParams).get();
                                    if (result=="SUCCESS") {
                                        Toast.makeText(UpdateActivity.this, "수정되었습니다!", Toast.LENGTH_SHORT).show();
                                    }else{
                                        Toast.makeText(UpdateActivity.this, "수정되지 않았습니다!", Toast.LENGTH_SHORT).show();
                                    }
                                    finish();
                                } catch (Exception e) { }
                            }
                        });
                        box.show();
                        break;
                    case R.id.btnCancel:
                        finish();
                }
            }
        }
```
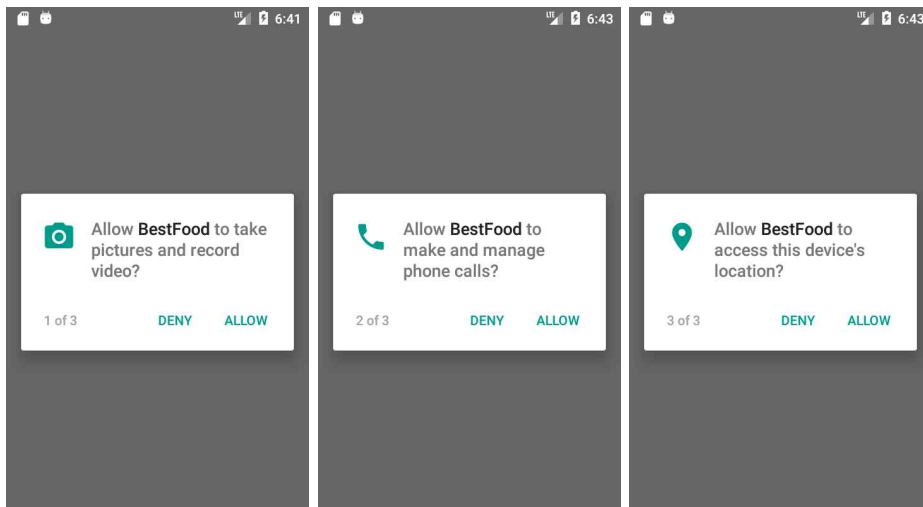
- **권한 화면 구성하기**

```java
public class PermissionActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_permission);
        if(Build.VERSION.SDK_INT < 23) {
            goIndexActivity();
        }else{
            checkAndRequestPermission();
        }
    }

    /* 인덱스 액티비티를 실행하고 현재 액티비티를 종료한다. */
    private void goIndexActivity(){
        Intent intent=new Intent (this, IndexActivity.class);
        startActivity(intent);
        finish();
    }

    /* 권한을 확인하고 권한이 부여되어 있지 않다면 권한을 요청하고 요청되어 있으면 인덱스 액티비티로 이동한다. */
    private void checkAndRequestPermission(){
        int intGranted = PackageManager.PERMISSION_GRANTED;
        if(ContextCompat.checkSelfPermission(this, Manifest.permission.CAMERA) != intGranted
            || ContextCompat.checkSelfPermission(this, Manifest.permission.READ_PHONE_STATE) != intGranted
            || ContextCompat.checkSelfPermission(this, Manifest.permission.ACCESS_FINE_LOCATION) != intGranted) {

            ActivityCompat.requestPermissions(this,new String[]{Manifest.permission.CAMERA,
                                                    Manifest.permission.READ_PHONE_STATE,
                                                    Manifest.permission.ACCESS_FINE_LOCATION},100);
        }else{
            goIndexActivity();
        }
    }

    /* 권한 요청 결과를 보고 모든 권한이 승인되었을 경우에는 메인 액티비티로 이동하고 아니면 현재 액티비티를 종료한다. */
    @Override
    public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {
        boolean isAllGranted = true;
        switch (requestCode){
            case 100:
                for(int i=0; i<permissions.length; i++){
                    if(grantResults[i] != PackageManager.PERMISSION_GRANTED){
                        isAllGranted=false;
                    }
                }
                if(isAllGranted){
                    goIndexActivity();
                }else{
                    Toast.makeText(this, "퍼미션을 얻지 못했습니다!", Toast.LENGTH_SHORT).show();
                    finish();
                }
                break;
        }
    }
}
```

```xml
...
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.CAMERA" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
...
```

4-2. BR

```java
public class MainActivity extends AppCompatActivity {
    NotificationManager mNotiManager;
    static final int NAPNOTI = 1;

    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        mNotiManager = (NotificationManager) getSystemService(NOTIFICATION_SERVICE);
    }

    public void mOnClick(View v) {
        Toast.makeText(MainActivity.this, "안녕히 주무세요!", Toast.LENGTH_SHORT).show();
        v.postDelayed(new Runnable() {
            @Override
            public void run() {
                Intent intent = new Intent(MainActivity.this, NapEnd.class);
                intent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
                PendingIntent content = PendingIntent.getActivity(MainActivity.this, 0, intent, 0);
                if(Build.VERSION.SDK_INT>=26) {
                    NotificationChannel mChannel = new NotificationChannel("channel_01",
                                            "channel_alam", NotificationManager.IMPORTANCE_LOW);
                    mChannel.setDescription("알람");
                    mChannel.enableLights(true);
                    mChannel.setLightColor(Color.RED);
                    mChannel.enableVibration(true);
                    mChannel.setVibrationPattern(new long[]{100, 200, 300, 400, 500, 400, 300, 200, 400});
                    mNotiManager.createNotificationChannel(mChannel);
                }

                NotificationCompat.Builder builder = new NotificationCompat.Builder(MainActivity.this)
                        .setContentTitle("기상시간")
                        .setContentText("일어나세요! 일할시간이예요...")
                        .setSmallIcon(R.drawable.icon_noti)
                        .setContentIntent(content);

                if(Build.VERSION.SDK_INT>=26) {
                    builder.setChannelId("channel_01");
                }

                Notification noti = builder.build();
                mNotiManager.notify(NAPNOTI, noti);
            }
        }, 5 * 1000);
    }
}
```

```java
public class DetectFree extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_detect_free);
    }

    public void mOnClick(View v){
        v.postDelayed(new Runnable() {
            @Override
            public void run() {
                BroadcastReceiver freeBR=new FreeBR();
                IntentFilter filter=new IntentFilter("freeWIFI");
                registerReceiver(freeBR, filter);

                Intent intent=new Intent();
                intent.setAction("freeWIFI");
                sendBroadcast(intent);
            }
        }, 5*1000);
    }
}
```

```java
public class AlarmTest extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_alarm_test);
    }

    public void mOnClick(View v){
        AlarmManager am=(AlarmManager)getSystemService(Context.ALARM_SERVICE);
        Intent intent;
        PendingIntent sender;

        switch(v.getId()){
            case R.id.onetime:
                //예약에 의해 호출될 BR 지정
                intent=new Intent(this, AlarmReceiver.class);
                sender=PendingIntent.getBroadcast(this,0,intent,0);
                am.set(AlarmManager.RTC, System.currentTimeMillis() + 5*1000, sender);
                break;

            case R.id.repeat:
            case R.id.stop:
                intent=new Intent(this, DisplayScore.class);
                sender=PendingIntent.getBroadcast(this, 0, intent, 0);
                //6초당 한 번 알람 등록
                if(v.getId()==R.id.repeat){
                    am.setRepeating(AlarmManager.RTC_WAKEUP, System.currentTimeMillis(), 6*1000,sender);
                }else {
                    am.cancel(sender);
                }
                break;
        }
    }
}
```

4-3. ViewPager

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:orientation="vertical">
    <android.support.design.widget.TabLayout
        android:id="@+id/tab"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
    <android.support.v4.view.ViewPager
        android:id="@+id/pager"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>
</LinearLayout>
```

```java
public class FragmentMoive extends Fragment {
    @Nullable
    @Override
    public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup container,
                                                    @Nullable Bundle savedInstanceState) {
        View view=inflater.inflate(R.layout.fragment_movie, container, false);
        return view;
    }
}
```

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="영화검색"/>
</LinearLayout>
```

```
public class FragmentBook extends Fragment {
    @Nullable
    @Override
    public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup container,
                                                    @Nullable Bundle savedInstanceState) {
        View view=inflater.inflate(R.layout.fragment_book, container, false);
        return view;
    }
}
```

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="영화검색"/>
</LinearLayout>
```

```
public class MainActivity extends AppCompatActivity {
    ArrayList<Fragment> fragments;
    PagerAdapter adapter;
    TabLayout tab;
    ViewPager pager;

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

        tab=findViewById(R.id.tab);
        tab.addTab(tab.newTab().setText("영화"));
        tab.addTab(tab.newTab().setText("도서"));
        tab.getTabAt(0).setIcon(R.drawable.icon_movie);
        tab.getTabAt(1).setIcon(R.drawable.icon_book);

        fragments=new ArrayList<Fragment>();
        fragments.add(new FragmentMoive());
        fragments.add(new FragmentBook());

        TabLayout tab=findViewById(R.id.tab);
        pager=findViewById(R.id.pager);
        adapter=new PagerAdapter(getSupportFragmentManager(), fragments);
        pager.setAdapter(adapter);

        pager.addOnPageChangeListener(new TabLayout.TabLayoutOnPageChangeListener(tab));
        tab.addOnTabSelectedListener(new TabLayout.OnTabSelectedListener() {
            @Override
            public void onTabSelected(TabLayout.Tab tab) {
                pager.setCurrentItem(tab.getPosition());
            }

            @Override
            public void onTabUnselected(TabLayout.Tab tab) {
```

```
            }

            @Override
            public void onTabReselected(TabLayout.Tab tab) {

            }
        });
    }

    private class PagerAdapter extends FragmentStatePagerAdapter{
        ArrayList<Fragment> fragments;

        public PagerAdapter(FragmentManager fm, ArrayList<Fragment> fragments) {
            super(fm);
            this.fragments = fragments;
        }

        @Override
        public Fragment getItem(int i) {
            return fragments.get(i);
        }

        @Override
        public int getCount() {
            return fragments.size();
        }
    }
}
```

4-4. 공유

```
dependencies {
    ...

    implementation 'com.soundcloud.android:android-crop:1.0.1@aar'

    ...
}
```

[AndroidManifest.xml]

```
<application>
  ...

  <provider
    android:name="android.support.v4.content.FileProvider"
    android:authorities="${applicationId}.provider"
    android:exported="false"
    android:grantUriPermissions="true">
    <meta-data
      android:name="android.support.FILE_PROVIDER_PATHS"
      android:resource="@xml/provider_paths"/>
  </provider>
  <activity android:name="com.soundcloud.android.crop.CropImageActivity"
    android:screenOrientation="portrait"
    android:theme="@style/Theme.AppCompat.NoActionBar"/>

  ...
</appicaion>
```

```xml
<Resource xmlns:android="http://schemas.android.com/apk/res/android">
  <paths xmlns:android="http://schemas.android.com/apk/res/android">
    <external-path name="external_files" path="."/>
  </paths>
</Resource>
```

```xml
<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=".Main2Activity"
    android:orientation="vertical"
    android:gravity="center">
    <Button
        android:id="@+id/btn"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="텍스트 공유"/>
    <ImageView
        android:id="@+id/img"
        android:layout_width="150sp"
        android:layout_height="120sp"
        android:src="@drawable/img01"/>
    <Button
        android:id="@+id/btn1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="캡처화면 공유"/>
</LinearLayout>
```

```java
public class Main2Activity extends AppCompatActivity {
    File screenShot;
    Uri uriFile;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);
        permissionCheck();
        Button btn=findViewById(R.id.btn);
        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent sendIntent = new Intent();
                sendIntent.setAction(Intent.ACTION_SEND);
                sendIntent.putExtra(Intent.EXTRA_TEXT, "This is my text to send.");
                sendIntent.setType("text/plain");
                //startActivity(sendIntent);

                List targetIntents = new ArrayList<Intent>();
                startActivity(Intent.createChooser(sendIntent, "선택"));
            }
        });

        ImageView img=findViewById(R.id.img);
        img.setOnClickListener(new View.OnClickListener(){
            @Override
            public void onClick(View v) {
                Uri imageUri = Uri.parse("android.resource://" + getPackageName()+ "/drawable/" + "img01");
                Intent shareIntent = new Intent();
                shareIntent.setAction(Intent.ACTION_SEND);
```

```java
                shareIntent.putExtra(Intent.EXTRA_STREAM, imageUri);
                shareIntent.setType("image/*");
                startActivity(Intent.createChooser(shareIntent, "선택"));

            }
        });

        Button btnl = findViewById(R.id.btnl);
        btnl.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                View rootView = getWindow().getDecorView();
                screenShot = ScreenShot(rootView);
                uriFile = Uri.fromFile(screenShot);
                if(screenShot!=null){
                    Crop.of(uriFile, uriFile).asSquare().start(Main2Activity.this, 100);
                }
            }
        });
    }
    @Override
    protected void onActivityResult(int requestCode, int resultCode, Intent data) {
        super.onActivityResult(requestCode, resultCode, data);
        File cropFile = screenShot;

        if(requestCode ==100){
            if (resultCode == RESULT_OK) {
                cropFile = new File(Crop.getOutput(data).getPath());
            }

            if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.N) {  // API 24 이상 일경우..
                uriFile = FileProvider.getUriForFile(getApplicationContext(),
                        getApplicationContext().getPackageName() + ".provider", cropFile);
            } else {  // API 24 미만 일경우..
                uriFile = Uri.fromFile(cropFile);
            }
            Intent shareIntent = new Intent();
            shareIntent.setAction(Intent.ACTION_SEND);
            shareIntent.putExtra(Intent.EXTRA_STREAM, uriFile);
            shareIntent.setType("image/*");
            startActivity(Intent.createChooser(shareIntent, "선택"));
        }
    }

    public File ScreenShot(View view){
        view.setDrawingCacheEnabled(true);  //화면에 뿌릴때 캐시를 사용하게 한다
        Bitmap screenBitmap = view.getDrawingCache();   //캐시를 비트맵으로 변환
        String filename = "screenshot.png";
        File file = new File(Environment.getExternalStorageDirectory() + "/Pictures", filename);
                                                    //Pictures폴더 screenshot.png 파일

        FileOutputStream os = null;
        try{
            os = new FileOutputStream(file);
            screenBitmap.compress(Bitmap.CompressFormat.PNG, 90, os);   //비트맵을 PNG파일로 변환
            os.close();
        }catch (Exception e){
            System.out.println(e.toString());
            return null;
        }
        view.setDrawingCacheEnabled(false);
        return file;
    }

    public void permissionCheck(){
        if(ActivityCompat.checkSelfPermission(this, Manifest.permission.WRITE_EXTERNAL_STORAGE)
                                                        != PackageManager.PERMISSION_GRANTED){
            ActivityCompat.requestPermissions(this, new String[]{Manifest.permission.WRITE_EXTERNAL_STORAGE}, 100);
        }
    }
}
```

4-6. 툴바

- **액티비티에 포함된 기본 ActionBar 사용하지 않게 만들기.**

[res]-[values]-[styles]-[styles.xml]

```
<style name="AppTheme" parent="Theme.AppCompat.Light.NoActionBar">
```

- **액티비티의 레이아웃 xml에 Toolbar 위젯 추가**

[res]-[layout]-[activity_main.xml]

```
<android.support.v7.widget.Toolbar
  android:id="@+id/toolbar"
  android:layout_width="match_parent"
  android:layout_height="wrap_content"
  android:background="?attr/colorPrimaryDark">
  <TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="여기는 툴바입니다."
    android:layout_gravity="left"/>
<android.support.v7.widget.Toolbar/>
```

- **Toolbar를 액티비티의 앱바(App Bar)로 지정**

[java]-[com.example.chatting]-[MainActivity.java]

```
Toolbar toolbar=findViewById(R.id.toolbar);
setSupportActionBar(toolbar);

//App Bar의 타일틀 값을 지운다.
getSupportActionBar().setTitle("");

//Toolbar에 대한 참조를 획득한다.
ActionBar ab = getSupportActionBar();
```