



2일차 : 실습교재

옥 상 훈

Blog: <http://okgosu.tistory.com>

Web Site: <http://okgosu.net>

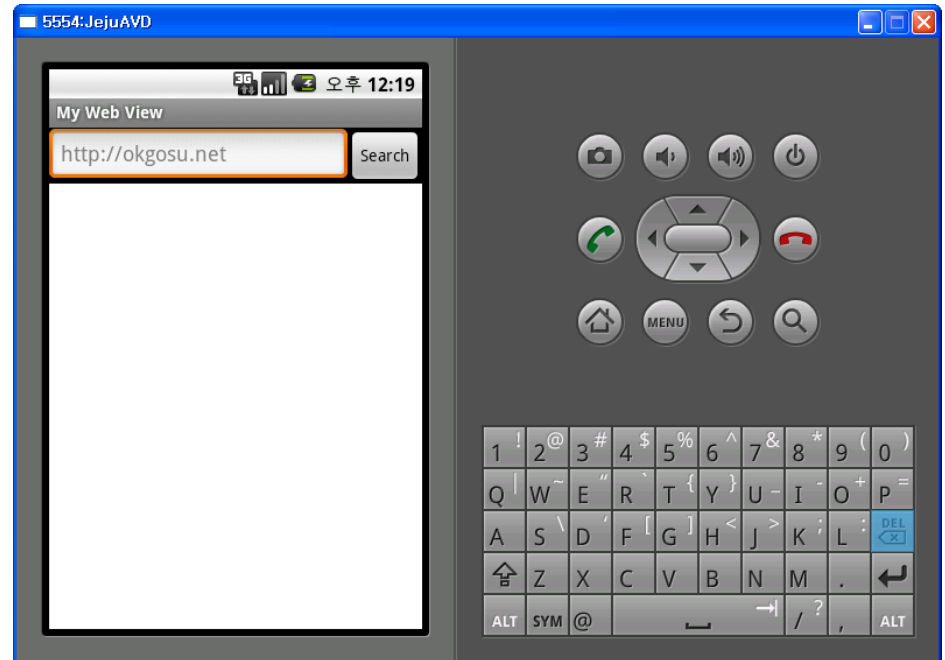
twitter @okgosu

2일차 실습



My Web View

- 목표
 - 에디트텍스트, 버튼, 웹뷰 활용법
- 작성할 파일
 - layout: my_web_view.xml
 - Activity: MyWebView.java



My Web View 1 단계 - 대상: my_web_view.xml

- ❑ LinearLayout 생성
 - ❑ android:orientation= *"vertical"*
 - ❑ android:layout_width= *"fill_parent"*
 - ❑ android:layout_height= *"fill_parent"*



My Web View 2 단계 - 대상: my_web_view.xml

❑ LinearLayout 안에 1) LinearLayout 2) WebView 추가

❑ LinearLayout 속성

```
<LinearLayout  
    android:orientation= "horizontal"  
    android:layout_width= "fill_parent"  
    android:layout_height= "wrap_content">  
</LinearLayout>
```

❑ WebView 속성

```
<WebView android:id= "@+id/webkit"  
    android:layout_width= "fill_parent"  
    android:layout_height= "fill_parent"/>
```



My Web View 3 단계 - 대상: my_web_view.xml

❑ 2단계의 LinearLayout 안에 1) EditText, 2) Button 추가

❑ EditText 속성

```
<EditText  
    android:id="@+id/url"  
    android:layout_height="wrap_content"  
    android:layout_width="wrap_content"  
    android:lines="1"  
    android:layout_weight="1"  
    android:hint="http://okgosu.net"/>
```

❑ Button 속성

```
<Button  
    android:id="@+id/go_button"  
    android:layout_height="wrap_content"  
    android:layout_width="wrap_content"  
    android:text="Search"/>
```



My Web View 4 단계 - 대상: MyWebView.java

- ❑ MyWebView 액티비티 생성
- ❑ onCreate() 함수 오버라이딩
 - ❑ setContentView(R.layout.*my_web_view*);
 - ❑ **final** Button btn = (Button)findViewById(R.id.*go_button*);



My Web View 5 단계 - 대상: MyWebView.java

□ 클래스 import

```
import android.webkit.WebView;  
import android.widget.Button;  
import android.widget.EditText;
```



My Web View 6 단계 - 대상: MyWebView.java

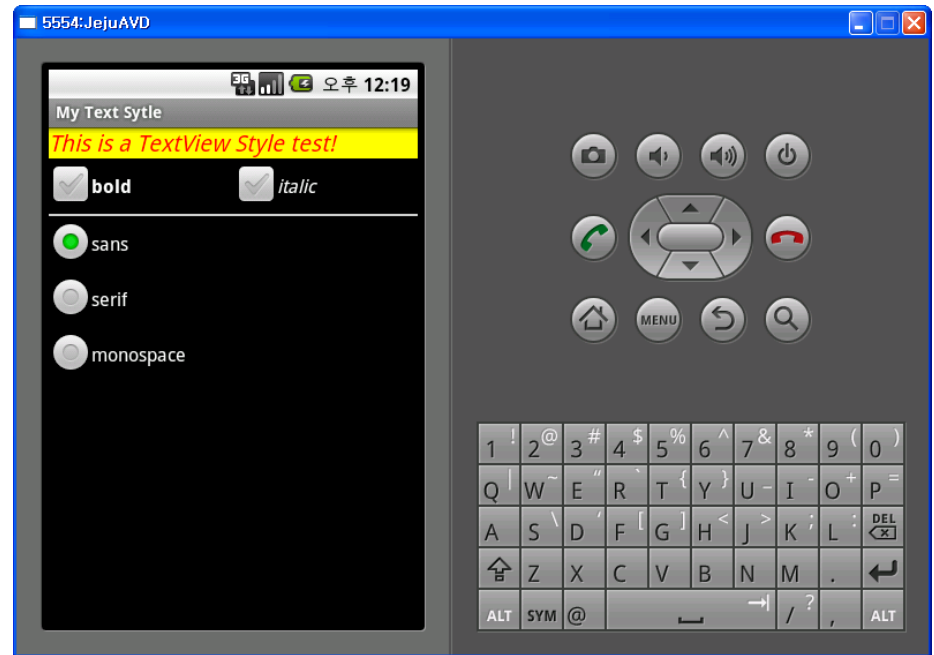
□ 버튼 클릭 핸들러 구현

```
btn.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        WebView browser = (WebView)findViewById(R.id.webkit);  
        EditText ed = (EditText)findViewById(R.id.url);  
        browser.loadUrl(ed.getText().toString());  
  
        // 현재의 액티비티 안에서만 웹페이지가 로딩되도록  
        browser.setWebViewClient(new WebViewClient() {  
            public boolean shouldOverrideUrlLoading(WebView view, String url) {  
                view.loadUrl(url);  
                return true;  
            }  
        });  
    }  
});
```



My Text Style

- 목표
 - 텍스트 스타일 설정, 체크박스, 라디오버튼 활용법
- 작성할 파일
 - layout: my_text_style.xml
 - Activity: MyTextStyle.java



My Text Style 1 단계 - 대상: my_text_style.xml

- ❑ LinearLayout 생성
 - ❑ android:orientation= *"vertical"*
 - ❑ android:layout_width= *"fill_parent"*
 - ❑ android:layout_height= *"fill_parent"*



My Text Style 2 단계 - 대상: my_text_style.xml

☐ LinearLayout 안에 1) TextView 2) LinearLayout 3) View 4) RadioGroup 추가

☐ 1) TextView 속성

- ☐ android:id= "@+id/tv1"
- ☐ android:layout_width= "fill_parent"
- ☐ android:layout_height= "wrap_content"
- ☐ android:text= "This is a TextView Style test!"
- ☐ android:typeface= "sans"
- ☐ android:textColor= "#FF0000"
- ☐ android:background= "#FFFF00"
- ☐ android:textSize= "20sp"
- ☐ android:layout_gravity= "left"

☐ 2) LinearLayout 속성

- ☐ android:layout_width= "fill_parent"
- ☐ android:orientation= "horizontal"
- ☐ android:layout_height= "wrap_content"



My Text Style 2 단계 - 대상: my_text_style.xml

- ❑ 3) View 속성
 - ❑ `android:background= "#CCCCCC"`
 - ❑ `android:layout_width= "fill_parent"`
 - ❑ `android:layout_height= "2sp"`

- ❑ 4) RadioGroup 속성
 - ❑ `android:id= "@+id/rdg"`
 - ❑ `android:layout_width= "wrap_content"`
 - ❑ `android:layout_height= "wrap_content"`



My Text Style 3 단계 - 대상: my_text_style.xml

- ❑ 2번째 LinearLayout 안에 CheckBox 2개 추가
 - ❑ 1) CheckBox
 - ❑ android:text= "*bold*"
 - ❑ android:textStyle= "*bold*"
 - ❑ android:id= "@+id/CheckBox01"
 - ❑ android:layout_weight= "1"
 - ❑ android:layout_width= "*wrap_content*"
 - ❑ android:layout_height= "*wrap_content*"
 - ❑ 2) CheckBox
 - ❑ android:text= "*italic*"
 - ❑ android:textStyle= "*italic*"
 - ❑ android:id= "@+id/CheckBox02"
 - ❑ android:layout_weight= "1"
 - ❑ android:layout_width= "*wrap_content*"
 - ❑ android:layout_height= "*wrap_content*"



My Text Style 4 단계 - 대상: my_text_style.xml

☐ RadioGroup 안에 Radio 버튼 3개 추가

☐ 1) RadioButton

- ☐ android:text= "*sans*"
- ☐ android:id= "@+id/rb1"
- ☐ android:checked= "*true*"
- ☐ android:layout_width= "*wrap_content*"
- ☐ android:layout_height= "*wrap_content*"

☐ 2) RadioButton

- ☐ android:text= "*serif*"
- ☐ android:id= "@+id/rb2"
- ☐ android:layout_width= "*wrap_content*"
- ☐ android:layout_height= "*wrap_content*"

☐ 3) RadioButton

- ☐ android:text= "*monospace*"
- ☐ android:id= "@+id/rb3"
- ☐ android:layout_width= "*wrap_content*"
- ☐ android:layout_height= "*wrap_content*"



My Text Style 5 단계 - 대상: MyTextStyle.java

- ❑ MyTextStyle 액티비티 생성
 - ❑ **public class** MyTextStyle **extends** Activity

- ❑ 변수 선언
 - ❑ **int** tvStyle1;
 - ❑ **int** tvStyle2;
 - ❑ **private** String mytype="sans";
 - ❑ RadioGroup rdg;



My Text Style 6 단계 - 대상: MyTextStyle.java

- ❑ 클래스 import
 - ❑ **import** android.widget.CheckBox;
 - ❑ **import** android.widget.CompoundButton;
 - ❑ **import** android.widget.RadioGroup;
 - ❑ **import** android.widget.TextView;

- ❑ onCreate() 함수 오버라이딩
 - ❑ setContentView(R.layout.*my_text_style*);
 - ❑ TextView tv1 = (TextView)findViewById(R.id.*tv1*);
 - ❑ CheckBox cb1 = (CheckBox)findViewById(R.id.*CheckBox01*);
 - ❑ CheckBox cb2 = (CheckBox)findViewById(R.id.*CheckBox02*);
 - ❑ rdg = (RadioGroup)findViewById(R.id.*rdg*);



My Text Style 7 단계 - 대상: MyTextStyle.java

- onCreate() 함수 내의 cb1 에 setOnCheckedChangeListener 추가 후 onCheckedChanged 함수 구현

```
cb1.setOnCheckedChangeListener(new  
android.widget.CompoundButton.OnCheckedChangeListener() {  
  
    public void onCheckedChanged(CompoundButton buttonView, boolean  
isChecked) {  
        if(isChecked) {  
            tvStyle1 = Typeface.BOLD;  
        } else {  
            tvStyle1 = Typeface.NORMAL;  
        }  
        Typeface face=Typeface.create(mytype, tvStyle1|tvStyle2);  
        tv1.setTypeface(face, tvStyle1|tvStyle2);  
    }  
});
```



My Text Style 8 단계 - 대상: MyTextStyle.java

- onCreate() 함수 내의 cb2 에 setOnCheckedChangeListener 추가 후 onCheckedChanged 함수 구현

```
cb2.setOnCheckedChangeListener(new android.widget.CompoundButton.OnCheckedChangeListener() {  
    public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {  
        if(isChecked) {  
            tvStyle2 = Typeface.ITALIC;  
        } else {  
            tvStyle2 = Typeface.NORMAL;  
        }  
        Typeface face=Typeface.create(mytype, tvStyle1|tvStyle2);  
        tv1.setTypeface(face, tvStyle1|tvStyle2);  
    }  
});
```



My Text Style 9 단계 - 대상: MyTextStyle.java

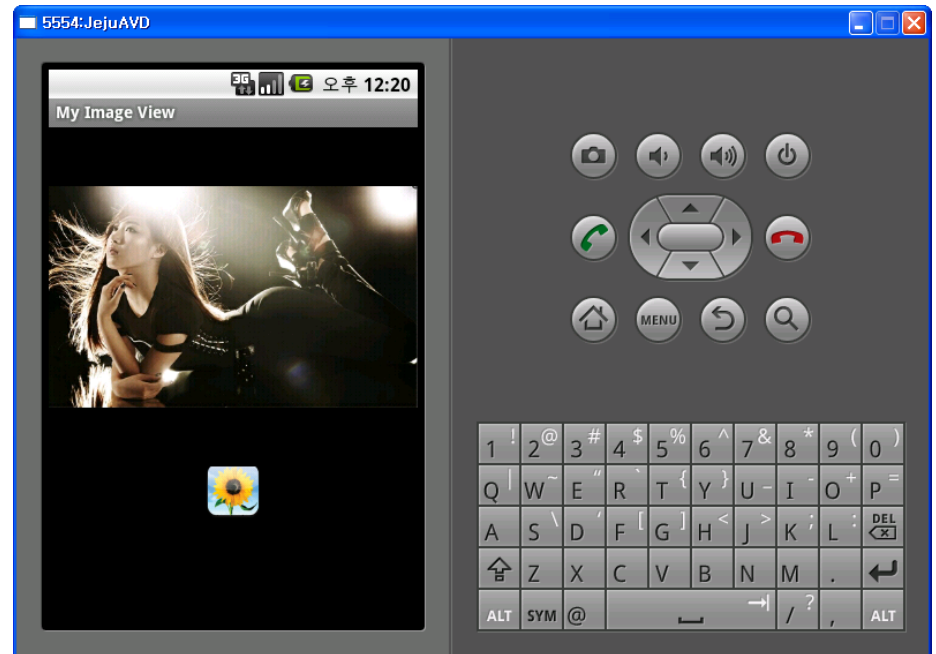
□ onCreate() 함수 내의 rdg 에 setOnCheckedChangeListener 추가 후 onCheckedChanged 함수 구현

```
rdg.setOnCheckedChangeListener(new android.widget.RadioGroup.OnCheckedChangeListener() {  
    @Override  
    public void onCheckedChanged(RadioGroup group, int checkedId) {  
        switch(checkedId){  
            case R.id.rb1:  
                mytype = "sans";  
                break;  
            case R.id.rb2:  
                mytype = "serif";  
                break;  
            case R.id.rb3:  
                mytype = "monospace";  
                break;  
        }  
        Typeface face=Typeface.create(mytype, 0);  
        tv1.setTypeface(face);  
    }  
});
```



My Image View

- 목표
 - 이미지, 이미지버튼, 배경화면 설정 방법, 터치이벤트 처리
- 작성할 파일
 - layout: my_image_view.xml
 - Activity: MyImageView.java



My Image View 1단계 – 대상: my_image_view.xml

- ❑ LinearLayout 생성
 - ❑ android:orientation= *"vertical"*
 - ❑ android:layout_width= *"wrap_content"*
 - ❑ android:layout_height= *"wrap_content"*



My Image View 2 단계 - 대상: my_image_view.xml

❑ LinearLayout 안에 1) ImageView 2) ImageButton추가

❑ Image 속성

```
<ImageView android:id="@+id/img"  
    android:layout_weight="1"  
    android:layout_width="wrap_content"  
    android:layout_height="fill_parent"/>
```

❑ ImageButton속성

```
<ImageButton android:id="@+id/imgBtn"  
    android:src="@drawable/my_image_btn"  
    android:layout_gravity="bottom|center"  
    android:layout_width="48px"  
    android:layout_height="48px"/>
```



My Image View 3 단계 - 대상: *@drawable/my_image_btn.xml*

❑ drawable 폴더에 다음 이미지 복사

❑ *drawable/wallpaper_icon.png*

❑ *drawable/wallpaper_icon_dw.png*

❑ *drawable/wallpaper_icon_focus.png*



❑ drawable 폴더에 다음 xml 파일 추가

```
<?xml version="1.0" encoding="UTF-8"?>
<selector xmlns:android="http://schemas.android.com/apk/res/android">
<item android:state_pressed="true" android:drawable="@drawable/wallpaper_icon_dw" />
<item android:state_focused="true" android:drawable="@drawable/wallpaper_icon_focus" />
<item android:drawable="@drawable/wallpaper_icon" />
</selector>
```



My Image View 4 단계 - 대상: MyImageView.java

- ❑ MyImageView 액티비티 생성
- ❑ onCreate() 함수 오버라이딩
 - ❑ setContentView(R.layout.*my_image_view*);



My Image View 5 단계 - 대상: MyImageView.java

❑ onCreate() 함수에

❑ 이미지, 이미지 버튼 선언 및 관련 클래스 import

```
ImageView iv;  
ImageButton imgBtn;
```

❑ 이미지, 이미지 버튼 아이디 참조

```
imgBtn = (ImageButton)findViewById(R.id.imgBtn);  
iv = (ImageView)findViewById(R.id.img);
```

❑ 이미지 설정

```
iv.setImageResource(R.drawable.icon);
```

❑ 배경이미지 매니저 인스턴스

```
final WallpaperManager wallpaperManager = WallpaperManager.getInstance(this);
```



My Image View 6 단계 - 대상: MyImageView.java

□ 이미지에 터치 이벤트 리스너

```
iv.setOnTouchListener(new OnTouchListener() {  
    @Override  
    public boolean onTouch(View v, MotionEvent event) {  
        loadImages();  
        return false;  
    }  
});
```

□ 이미지버튼에 클릭 이벤트 리스너 (배경화면 설정)

```
imgBtn.setOnClickListener(new OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        iv.setDrawingCacheEnabled(true);  
        try {  
            wallpaperManager.setBitmap(iv.getDrawingCache());  
            finish();  
        } catch (IOException e) {  
            e.printStackTrace();  
        }  
    }  
});
```



My Image View 7 단계 - 대상: MyImageView.java

□ 웹 이미지 로딩

```
int num = 1;
private void loadImages() {
    try {
        URL url = new URL("http://okgosu.net/pds/blackgg/" + num + ".jpg");
        URLConnection conn = url.openConnection();
        conn.connect();
        BufferedInputStream bis = new BufferedInputStream(conn.getInputStream());
        Bitmap bm = BitmapFactory.decodeStream(bis);
        bis.close();
        iv.setImageBitmap(bm);
        num++;
        if(num>9) num = 1;
    } catch (IOException e) {
        Log.w("okgosu", e );
    }
}
```



My Image View 8 단계 - 대상: AndroidManifest.xml

□ permission 추가

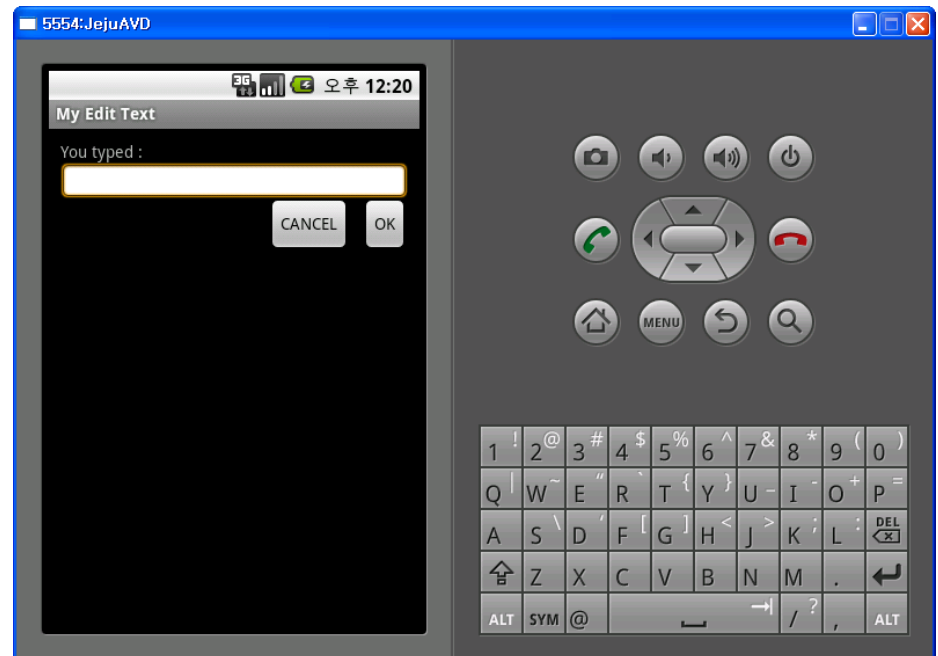
```
<uses-permission android:name="android.permission.INTERNET"> </uses-permission>
```

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



My Edit Text

- 목표
 - 글자 입력수 출력, RelativeLayout, addTextChangedListener
- 작성할 파일
 - layout: my_edit_text.xml
 - Activity: MyEditText.java



My Edit Text 1단계 – 대상: my_edit_text.xml

☐ **RelativeLayout** 생성

- ☐ android:layout_width= "*fill_parent*"
- ☐ android:layout_height= "*wrap_content*"
- ☐ android:padding= "*10dp*"



My Edit Text 2단계 – 대상: my_edit_text.xml

❑ **RelativeLayout** 안에 1) TextView 2) EditText 추가

```
<TextView
    android:id="@+id/label"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="You typed :"/>
<EditText
    android:id="@+id/entry"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:background="@android:drawable/editbox_background"
    android:layout_below="@id/label"/>
```



My Edit Text 3단계 – 대상: my_edit_text.xml

□ EditText 밑에 Button 2개 추가

```
<Button
    android:id="@+id/ok"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/entry"
    android:layout_alignParentRight="true"
    android:layout_marginLeft="10dip"
    android:text="OK" />
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_toLeftOf="@id/ok"
    android:layout_alignTop="@id/ok"
    android:text="CANCEL" />
```



My Edit Text 4 단계 - 대상: MyEditText.java

- ❑ MyEditText 액티비티 생성
- ❑ 멤버변수 선언
 - ❑ TextView tv;
 - ❑ EditText ed;
- ❑ onCreate() 함수 오버라이딩
 - ❑ setContentView(R.layout.*my_edit_text*);
 - ❑ tv = (TextView)findViewById(R.id.*label*);
 - ❑ ed = (EditText)findViewById(R.id.*entry*);



My Edit Text 5 단계 - 대상: MyEditText.java

□ ed에 TextChangeListener 구현

```
ed.addTextChangedListener(new TextWatcher() {  
    @Override  
    public void onTextChanged(CharSequence s, int start, int before, int count) {  
        // TODO Auto-generated method stub  
        tv.setText("You typed: " + ed.getText().length());  
    }  
  
    @Override  
    public void beforeTextChanged(CharSequence s, int start, int count,  
    int after) {  
        // TODO Auto-generated method stub  
    }  
  
    @Override  
    public void afterTextChanged(Editable s) {  
        // TODO Auto-generated method stub  
    }  
});
```



My Scroll View

- 목표
 - 스크롤뷰 사용법, 반복문으로 여러 화면 생성
- 작성할 파일
 - layout: my_scroll_view.xml
 - Activity: MyScrollView.java



My Scroll View 1단계 – 대상: my_scroll_view.xml

☐ **ScrollView** 생성

- ☐ android:layout_width= "*fill_parent*"
- ☐ android:layout_height= "*wrap_content*"



My Scroll View 2단계 – 대상: my_scroll_view.xml

❑ **ScrollView**에 LinearLayout 추가

```
<LinearLayout android:id="@+id/LinearLayout01"  
    android:orientation="vertical"  
    android:layout_width="fill_parent"  
    android:layout_height="wrap_content"/>
```



My Scroll View 3단계 – 대상: MyScrollView.java

- ❑ MyScrollView 액티비티 생성
- ❑ 멤버변수 선언
 - ❑ `LinearLayout layout;`
- ❑ `onCreate()` 함수 오버라이딩
 - ❑ `setContentView(R.layout.my_scroll_view);`
 - ❑ `layout = (LinearLayout) findViewById(R.id.LinearLayout01);`
 - ❑ `createWidgets();`



My Scroll View 4단계 – 대상: MyScrollView.java

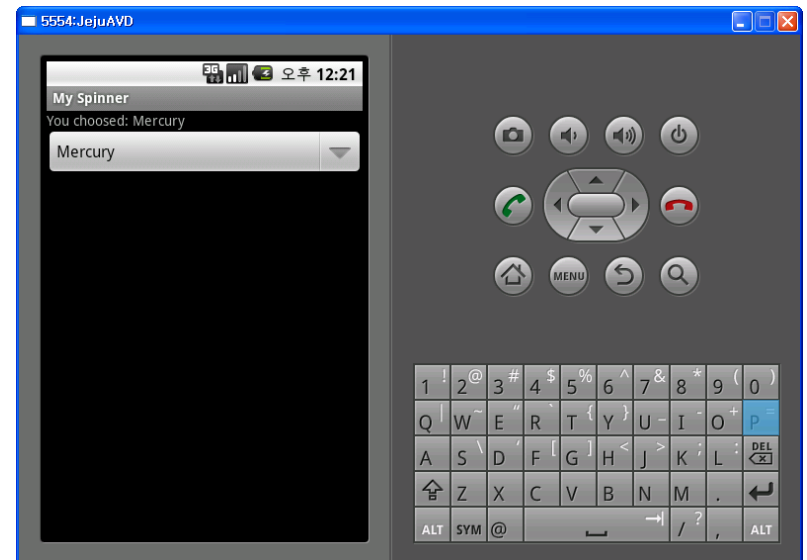
❑ createWidgets() 구현

```
private void createWidgets(){  
    for (int i = 1; i < 15; i++) {  
        TextView textView = new TextView(this);  
        textView.setText("Text View:" + i);  
        LinearLayout.LayoutParams p = new LinearLayout.LayoutParams(  
            LinearLayout.LayoutParams.FILL_PARENT,  
            LinearLayout.LayoutParams.WRAP_CONTENT  
        );  
        layout.addView(textView, p);  
        Button btn = new Button(this);  
        btn.setText("[Button]: " + i);  
        btn.setTextColor(0xFF0000FF);  
        layout.addView(btn);  
    }  
}
```



My Spinner

- ❑ 목표
 - ❑ Spinner 이벤트 처리, ArrayAdapter 사용법
- ❑ 작성할 파일
 - ❑ layout: my_spinner.xml
 - ❑ Activity: MySpinner.java



My Spinner 1단계 – 대상: my_spinner.xml

- ❑ LinearLayout 생성
 - ❑ android:orientation= *"vertical"*
 - ❑ android:layout_width= *"fill_parent"*
 - ❑ android:layout_height= *"wrap_content"*



My Spinner 2 단계 - 대상: my_spinner.xml

□ LinearLayout 안에 1) TextView 2) Spinner 추가

□ TextView 속성

```
<TextView android:id="@+id/tv_sp"  
    android:layout_width="fill_parent"  
    android:layout_height="wrap_content"  
    android:text="[You Choose]"/>
```

□ Spinner 속성

```
<Spinner android:id="@+id/spinner1"  
    android:layout_width="fill_parent"  
    android:layout_height="wrap_content"  
    android:drawSelectorOnTop="true"  
    android:prompt="@string/app_name" />
```



My Spinner 3단계 – 대상: MySpinner.java

- ❑ MySpinner 액티비티 생성
- ❑ 멤버변수 선언
 - ❑ Spinner s1;
- ❑ onCreate() 함수 오버라이딩
 - ❑ setContentView(R.layout.my_spinner);
 - ❑ s1 = (Spinner) findViewById(R.id.*spinner1*);



My Spinner 4단계 – 대상: MySpinner.java

- ❑ onCreate() 함수에 ArrayAdapter 추가
 - ❑ String[] items = {"aaa", "bbb", "ccc"};
 - ❑ ArrayAdapter <String> adapter = **new** ArrayAdapter<String> (**this**, android.R.layout.*simple_spinner_item*, items);
 - ❑ adapter.setDropDownViewResource(android.R.layout.*simple_spinner_dropdown_item*);
 - ❑ **s1.setAdapter(adapter);**



My Spinner 5단계 – 대상: MySpinner.java

□ onCreate() 함수에 Spinner 아이템 선택 이벤트 리스너 추가

```
s1.setOnItemClickListener(new OnItemSelectedListener() {  
    @Override  
        public void onItemClick(AdapterView<?> arg0, View arg1,  
            int pos, long arg3) {  
  
            TextView tv = (TextView)findViewById(R.id.tv_sp);  
tv.setText("You choosed: " + s1.getSelectedItem().toString());  
        }  
  
        @Override  
        public void onNothingSelected(AdapterView<?> arg0) {  
        }  
    });
```



My Tab

- 목표
 - Tab 화면 구성, 동적 Tab 추가
- 작성할 파일
 - my_tab.xml
 - MyTab.java



My Tab 1단계 – 대상: my_tab.xml

- ❑ LinearLayout 생성
 - ❑ android:orientation= *"vertical"*
 - ❑ android:layout_width= *"fill_parent"*
 - ❑ android:layout_height= *" fill_parent "*



My Tab 2단계 – 대상: my_tab.xml

- LinearLayout 안에 TabHost , FrameLayout 추가

```
<TabHost android:id="@+id/tabhost"  
    android:layout_width="fill_parent"  
    android:layout_height="fill_parent">  
    <FrameLayout android:id="@android:id/tabcontent"  
        android:paddingTop="100dip"  
        android:layout_width="fill_parent"  
        android:layout_height="fill_parent">  
        </FrameLayout>  
    </TabHost>
```



My Tab 3단계 – 대상: my_tab.xml

□ TabHost 안에 1) TabWidget 추가

```
<TabWidget android:id= "@android:id/tabs"  
  android:layout_width= "fill_parent"  
  android:layout_height= "wrap_content">  
</TabWidget>
```



My Tab 4단계 – 대상: my_tab.xml

☐ FrameLayout 안에 1) AnalogClock 2) Button 추가

☐ 1) AnalogClock

```
<AnalogClock android:id="@+id/AnalogClock01"  
    android:layout_width="fill_parent"  
    android:layout_height="fill_parent"/>
```

☐ 2) Button

```
<Button android:text="버튼추가" android:id="@+id/TabButton01"  
    android:layout_width="fill_parent"  
    android:layout_height="fill_parent"/>
```



My Tab 5단계 – 대상: MyTab.java

- ❑ MyTab 액티비티 생성
- ❑ onCreate() 함수 오버라이딩
 - ❑ setContentView(R.layout.*my_tab*);
 - ❑ **// 탭 초기화**
 - ❑ **final** TabHost tabs=(TabHost)findViewById(R.id.*tabhost*);
 - ❑ tabs.setup();



My Tab 6단계 – 대상: MyTab.java

- ❑ onCreate() 함수 오버라이딩 (이어서)
 - ❑ 첫번째 탭화면 추가
 - ❑ `TabHost.TabSpec spec = tabs.newTabSpec("tag1");`
 - ❑ `spec.setContent(R.id.AnalogClock01);`
 - ❑ `spec.setIndicator("시계");`
 - ❑ `tabs.addTab(spec);`
 - ❑ 두번째 탭화면 추가
 - ❑ `spec = tabs.newTabSpec("tag2");`
 - ❑ `spec.setContent(R.id.TabButton01);`
 - ❑ `spec.setIndicator("버튼");`
 - ❑ `tabs.addTab(spec);`
 - ❑ 현재 탭화면 지정
 - ❑ `tabs.setCurrentTab(0);`



My Tab 7단계 – 대상: MyTab.java

- ❑ onCreate() 함수 오버라이딩 (이어서)
 - ❑ 버튼 클릭시 동적 탭화면 추가되도록 클릭핸들러 추가

```
Button btn = (Button)findViewById(R.id.TabButton01);  
btn.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
    }  
});
```



My Tab 8단계 – 대상: MyTab.java

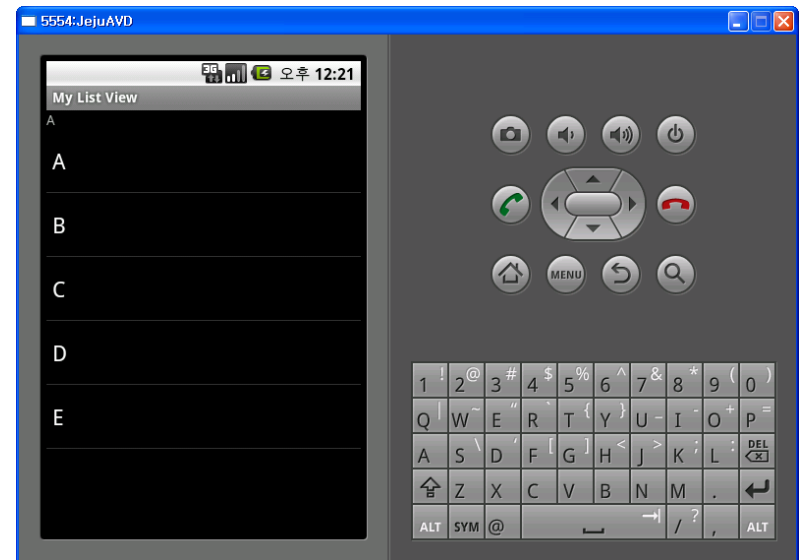
- 버튼 클릭 핸들러 안에 다음 내용 추가
 - 버튼 클릭시 동적 탭화면 추가되도록

```
TabHost.TabSpec spec = tabs.newTabSpec("tag1");
spec.setContent(new TabContentFactory() {
    @Override
        public View createTabContent(String tag) {
            return (new DatePicker(MyTab.this));
        }
});
spec.setIndicator("날짜");
tabs.addTab(spec);
tabs.setCurrentTab(tabs.getChildCount());
```



My List View

- ❑ 목표
 - ❑ ListActivity, ListView 사용방법, 이벤트 처리, ArrayAdapter 사용법
 - ❑ 옵션메뉴, 컨텍스트 메뉴 사용법
- ❑ 작성할 파일
 - ❑ layout: my_list_view.xml
 - ❑ Activity: MyListView.java



My List View 1단계 – 대상: my_list_view.xml

- ❑ LinearLayout 생성
 - ❑ android:orientation= *"vertical"*
 - ❑ android:layout_width= *"fill_parent"*
 - ❑ android:layout_height= *"fill_parent"*



My List View 2단계 – 대상: my_list_view.xml

❑ LinearLayout 안에 1) TextView 2) ListView 추가

❑ 1) TextView

```
<TextView android:id="@+id/listItem"  
    android:text="Selection"  
    android:layout_width="fill_parent"  
    android:layout_height="wrap_content"/>
```

❑ 2) ListView 추가

```
<ListView  
    android:id="@android:id/list"  
    android:drawSelectorOnTop="false"  
    android:layout_width="fill_parent"  
    android:layout_height="fill_parent"/>
```



My List View 3단계 – 대상: MyListView.java

- ❑ MyListView 액티비티 생성
 - ❑ ListActivity 를 extends함

- ❑ 멤버변수 선언
 - ❑ TextView itemSelected;
 - ❑ String[] items = {"AAA", "BBB", "CCC"};

- ❑ onCreate() 함수 오버라이딩
 - ❑ setContentView(R.layout.*my_list_view*);
 - ❑ itemSelected = (TextView)findViewById(R.id.*listItem*);



My List View 4단계 – 대상: MyListView.java

- ❑ onCreate() 함수 오버라이딩 (이어서)
 - ❑ ArrayAdapter 설정
 - ❑ setListAdapter(**new** ArrayAdapter<String>(**this**, android.R.layout.*simple_list_item_1*, items));

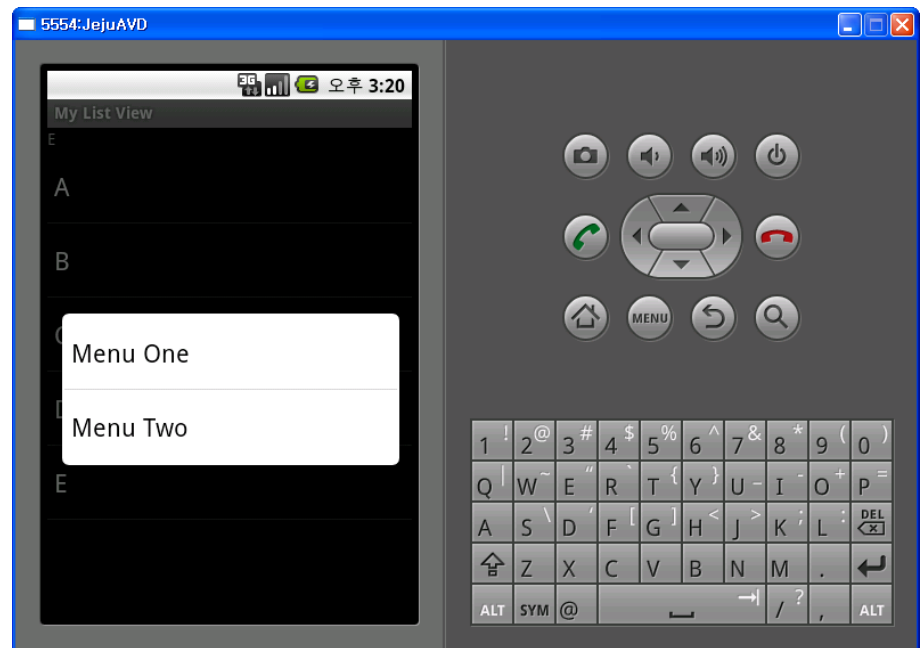
- ❑ ListActivity의 onItemClick 구현

```
public void onItemClick(ListView l, View v, int position, long id) {  
    itemSelected.setText(items[position]);  
}
```



My List View 5단계 – 대상: MyListView.java

- ❑ 컨텍스트 메뉴 생성을 위한 설정
 - ❑ onCreate() 함수에 다음 추가
 - ❑ `registerForContextMenu(getListView());`



My List View 6단계 – 대상: MyListView.java

❑ 컨텍스트 메뉴 생성을 위한 함수 추가

❑ 컨텍스트 메뉴 생성

@Override

```
public void onCreateContextMenu(ContextMenu menu, View v,  
ContextMenuItem menuInfo) {  
    createMyMenu(menu);  
}
```

❑ 메뉴 아이템 추가

```
private void createMyMenu(Menu menu) {  
    menu.add(Menu.NONE, Menu.FIRST, Menu.NONE, "Menu One");  
    menu.add(Menu.NONE, Menu.FIRST+1, Menu.NONE, "Menu Two");  
}
```



My List View 6단계 – 대상: MyListView.java

□ 메뉴 선택시 실행될 함수 지정

```
@Override
public boolean onOptionsItemSelected(MenuItem item) {
    return (myContextMenuChoice(item)||super.onOptionsItemSelected(item));
}

private boolean myContextMenuChoice(MenuItem item) {
    Toast.makeText(this, item.getTitle(), Toast.LENGTH_SHORT).show();
    return true;
}
```



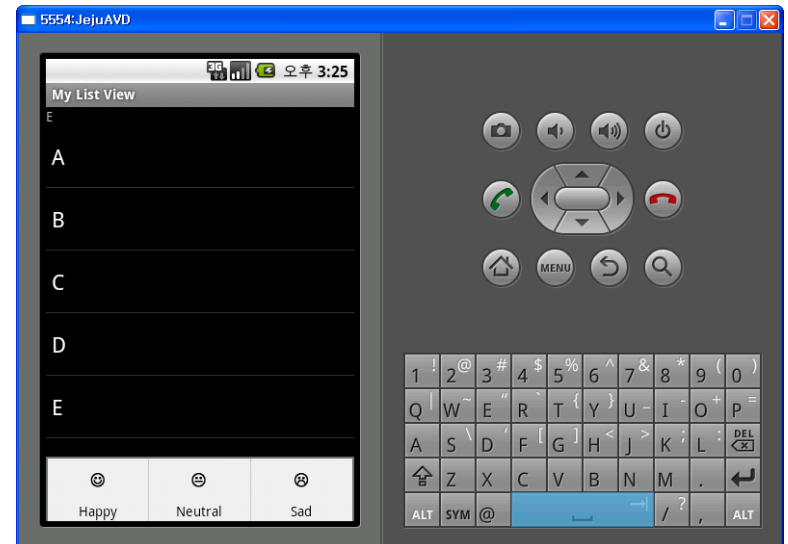
My List View 7단계 – 대상: MyListView.java

□ 메뉴를 위한 xml추가 : res/menu/title_icon.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:id="@+id/happy" android:title="Happy"
        android:icon="@drawable/stat_happy" />
    <item android:id="@+id/neutral" android:title="Neutral"
        android:icon="@drawable/stat_neutral" />
    <item android:id="@+id/sad" android:title="Sad"
        android:icon="@drawable/stat_sad" />
</menu>
```

□ 메뉴 xml의 아이콘 확인

- res/drawable/stat_happy.png
- res/drawable/stat_neutral.png
- res/drawable/stat_sad.png



My List View 8단계 – 대상: MyListView.java

□ 옵션 메뉴 생성을 위한 함수 추가

```
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    //createMyMenu(menu);
    new MenuInflater(getApplicationContext()).inflate(R.menu.title_icon, menu);
    return super.onCreateOptionsMenu(menu);
}
```

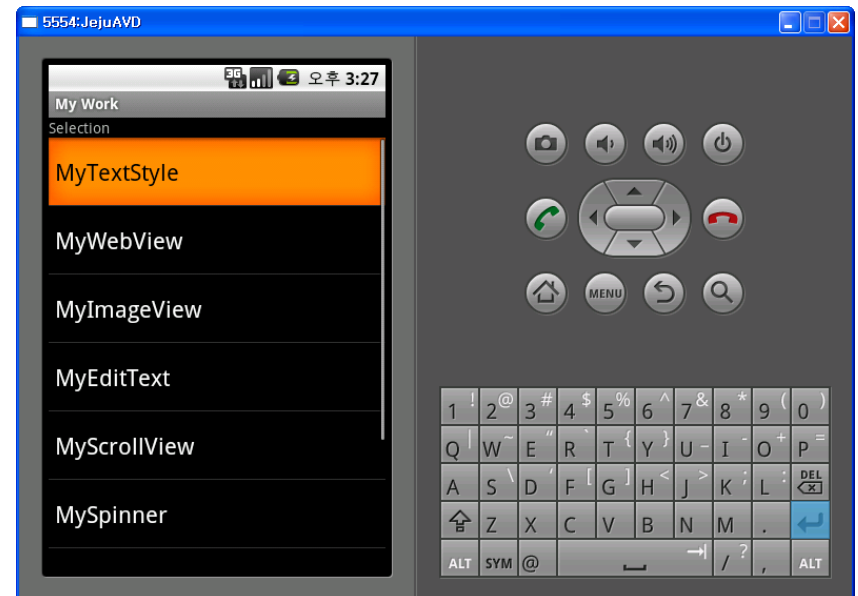
□ 옵션 메뉴 선택 이벤트 리스너 처리

```
@Override
public boolean onOptionsItemSelected(MenuItem item) {
    return (myContextMenuChoice(item)||super.onOptionsItemSelected(item));
}
```



My Main List

- ❑ 목표
 - ❑ Intent를 이용한 액티비티 이동
 - ❑ List 사용법
- ❑ 작성할 파일
 - ❑ layout: my_main_list.xml
 - ❑ Activity: MyMainList.java



My Main List – 1단계 : my_main_list.xml

- ❑ LinearLayout 생성
 - ❑ android:orientation= *"vertical"*
 - ❑ android:layout_width= *"fill_parent"*
 - ❑ android:layout_height= *"fill_parent"*



My Main List – 2단계 : my_main_list.xml

❑ LinearLayout 안에 1) TextView 2) ListView 추가

❑ 1) TextView

```
<TextView android:id="@+id/listItem"  
    android:text="Selection"  
    android:layout_width="fill_parent"  
    android:layout_height="wrap_content"/>
```

❑ 2) ListView 추가

```
<ListView  
    android:id="@android:id/list"  
    android:drawSelectorOnTop="false"  
    android:layout_width="fill_parent"  
    android:layout_height="fill_parent"/>
```



My Main List – 3단계 : MyMainList.java

- ❑ **ListActivity**를 extends하여
 - ❑ MyMainList.java 생성

- ❑ 멤버변수 추가
 - ❑ TextView itemSelected;
 - ❑ String[] items =
{"MyTextStyle", "MyWebView", "MyImageView", "MyEditText", "MyScrollView", "MySpinner", "MyListView", "MyTab"};

- ❑ onCreate() 함수 오버라이딩
 - ❑ setContentView(R.layout.my_main_list);
 - ❑ itemSelected = (TextView)findViewById(R.id.listItem);



My Main List – 4단계 : MyMainList.java

- ❑ onCreate() 함수 오버라이딩 이어서
 - ❑ ArrayAdapter 설정
 - ❑ setListAdapter(**new** ArrayAdapter<String>(**this**, android.R.layout.*simple_list_item_1*, items));
- ❑ onItemClick 함수 오버라이딩
 - ❑ **my.work**는 각자의 패키지명으로 설정

```
public void onItemClick(ListView l, View v, int position, long id) {  
    // 인텐트 호출  
}
```



My Main List – 5단계 : MyMainList.java

- 인텐트 호출
 - 명시적 호출
 - act 이름으로 호출하기
 - `startActivity(new Intent("lotte.net.MyWebView"));`
 - 클래스 이름으로 호출하기
 - `startActivity(new Intent(this, MyWebView.class));`
 - 클래스 문자열 이름으로 호출하기

```
try {  
    startActivity(new  
        Intent(this, Class.forName("my.work." +  
            items[position])));  
} catch (ClassNotFoundException e) {  
    e.printStackTrace();  
}
```



My Main List – 5단계 : AndroidManifest.xml

□ 액티비티 등록 확인

```
<activity android:name=".MyTextStyle" android:label="My Text Sytle">
    <intent-filter> <action android:name="okgosu.net.MyTextStyle" /> </intent-filter>
</activity>
<activity android:name=".MyWebView" android:label="My Web View">
    <intent-filter> <action android:name="okgosu.net.MyWebView " /> </intent-filter>
</activity>
<activity android:name=".MyImageView" android:label="My Image View">
    <intent-filter> <action android:name="okgosu.net.MyImageView" /> </intent-filter>
</activity>
<activity android:name=".MyEditText" android:label="My Edit Text">
    <intent-filter> <action android:name="okgosu.net.MyEditText" /> </intent-filter>
</activity>
<activity android:name=".MyScrollView" android:label="My Scroll View">
    <intent-filter> <action android:name="okgosu.net.MyScrollView" /> </intent-filter>
</activity>
<activity android:name=".MySpinner" android:label="My Spinner">
    <intent-filter> <action android:name="okgosu.net.MySpinner" /> </intent-filter>
</activity>
<activity android:name=".MyListView" android:label="My List View">
    <intent-filter> <action android:name="okgosu.net.MyListView" /> </intent-filter>
</activity>
<activity android:name=".MyTab" android:label="My Tab View">
    <intent-filter> <action android:name="okgosu.net.MyTabView" /> </intent-filter>
</activity>
```



수고하셨습니다 ^^/



open source project

How will you shape it?



by OkGosu.Net