

헬로, 안드로이드

9주차 - 로컬 데이터 저장하기

강대기

동서대학교 컴퓨터정보공학부

학습 목표

- 게임이나 프로그램에 옵션을 추가하는 방법을 알아본다.
- 이전의 프로그램을 계속하기 위해 상태를 저장하는 방법을 알아본다.
- 게임에서 현재의 실행 위치를 저장하는 방법에 대해 알아본다.
- 내부의 파일 시스템을 읽고 쓰는 방법에 대해 알아본다.
- SD 카드에 접근하는 방법에 대해 알아본다.
- 여러 UI 위젯들에 대해 알아본다.
- 음악 재생에 있어 Service 클래스를 이용하는 방법을 알아본다.

차례

- 스토쿠에 옵션 추가하기
- 이전 게임 계속하기
- 현재 위치 기억하기
- 내부 파일 시스템 액세스하기
- SD 카드 액세스하기
- 위젯
- 메뉴
- 다이얼로그 (대화 상자)
- 여러 UI 뷰들
- 음악 재생에 Service 클래스를 배경으로 사용
- 요약
- 퀴즈
- 연습문제

스도쿠에 옵션 추가하기

- 메뉴를 누르고 설정을 선택하면 Settings 액티비티가 시작되어 사용자가 게임 옵션을 변경 가능
- Settings 는 PreferenceActivity의 확장이므로 , Settings 값은 프로그램의 기본설정 영역에 저장 (Settings.java)
- 옵션 키(음악과 힌트)는 /res/xml/settings.xml과 동일
- Music.Play() 수정하여 음악의 기본 설정을 확인하도록 함 (Music.java)
- 힌트의 기본 설정 확인을 위해 PuzzleView 수정 (PuzzleView.java)

/src/org/example/sudoku/Settings.jav

a

```
public class Settings extends PreferenceActivity {  
    // Option names and default values  
    private static final String OPT_MUSIC = "music";  
    private static final boolean OPT_MUSIC_DEF = true;  
    private static final String OPT_HINTS = "hints";  
    private static final boolean OPT_HINTS_DEF = true;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        addPreferencesFromResource(R.xml.settings);  
    }  
  
    /** Get the current value of the music option */  
    public static boolean getMusic(Context context) {  
        return PreferenceManager.getDefaultSharedPreferences(context)  
            .getBoolean(OPT_MUSIC, OPT_MUSIC_DEF);  
    }  
  
    /** Get the current value of the hints option */  
    public static boolean getHints(Context context) {  
        return PreferenceManager.getDefaultSharedPreferences(context)  
            .getBoolean(OPT_HINTS, OPT_HINTS_DEF);  
    }  
}
```

```
/src/org/example/sudoku/Music.java
```

```
/** Stop old song and start new one */
```

```
public static void play(Context context, int  
resource) {
```

```
    stop(context);
```

```
    // Start music only if not disabled in preferences
```

```
if (Settings.getMusic(context)) {
```

```
    mp = MediaPlayer.create(context, resource);
```

```
    mp.setLooping(true);
```

```
    mp.start();
```

```
    }
```

```
    }
```

```
/src/org/example/sudoku/PuzzleView.
```

```
java
```

```
if (Settings.getHints(getContext())) {  
    // Draw the hints...  
  
    // Pick a hint color based on #moves left  
    Paint hint = new Paint();  
    int c[] = { getResources().getColor(R.color.puzzle_hint_0),  
               getResources().getColor(R.color.puzzle_hint_1),  
               getResources().getColor(R.color.puzzle_hint_2), };  
    Rect r = new Rect();  
    for (int i = 0; i < 9; i++) {  
        for (int j = 0; j < 9; j++) {  
            int movesleft = 9 - game.getUsedTiles(i, j).length;  
            if (movesleft < c.length) {  
                getRect(i, j, r);  
                hint.setColor(c[movesleft]);  
                canvas.drawRect(r, hint);  
            }  
        }  
    }  
}
```

이전 게임 계속하기

- 스마트폰 게임의 경우, 사용자가 아무 때고 게임을 중도에 그만둘 수 있는데, 나중에 이어서 할 수 있게 해야 함
- 게임의 상태 정보 저장 - 퍼즐의 상태는 한 타일 당 한 문자, 즉 81 개의 문자들의 문자열로 저장
- 퍼즐 데이터 키 상수(PREF_PUZZLE) 및 이전 게임의 계속을 위한 상수(DIFFICULTY_CONTINUE) (Game.java)
- onPause() 에서 게임 저장 (Game.java)
- getPuzzle() 에서 저장된 게임 다시 읽음. 이 때, DIFFICULTY_CONTINUE 확인 (Game.java)
- 계속하기 버튼이 눌리면, Sudoku.onClick()에 DIFFICULTY_CONTINUE 를 전달하고 startGame()을 호출하는 코드를 추가. startGame()은 Game 액티비티에 난이도 전달. Game.onCreate()는 Intent.getIntExtra()를 호출해 난이도 정보를 읽은 뒤, 그 정보를 getPuzzle()에 전달
- 액티비티가 저절로 사라졌다가 다시 시작되는 경우에도 게임 복원해야 함. Game.onCreate() 메서드 변경 (Game.java)

/src/org/example/sudoku/Game.java

```
private static final String PREF_PUZZLE = "puzzle";
protected static final int DIFFICULTY_CONTINUE = -1;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    Log.d(TAG, "onCreate");
    int diff = getIntent().getIntExtra(KEY_DIFFICULTY, DIFFICULTY_EASY);
    puzzle = getPuzzle(diff);
    calculateUsedTiles();
    puzzleView = new PuzzleView(this);
    setContentView(puzzleView);
    puzzleView.requestFocus();
    // ...
    // If the activity is restarted, do a continue next time
    getIntent().putExtra(KEY_DIFFICULTY, DIFFICULTY_CONTINUE);
}

@Override
protected void onPause() {
    super.onPause();
    Log.d(TAG, "onPause");
    Music.stop(this);
    // Save the current puzzle
    getPreferences(MODE_PRIVATE).edit().putString(PREF_PUZZLE, toPuzzleString(puzzle)).commit();
}

/** Given a difficulty level, come up with a new puzzle */
private int[] getPuzzle(int diff) {
    String puz;
    switch (diff) {
        case DIFFICULTY_CONTINUE: puz = getPreferences(MODE_PRIVATE).getString(PREF_PUZZLE, easyPuzzle); break;
        case DIFFICULTY_HARD: puz = hardPuzzle; break;
        case DIFFICULTY_MEDIUM: puz = mediumPuzzle; break;
        case DIFFICULTY_EASY: default: puz = easyPuzzle; break;
    }
    return fromPuzzleString(puz);
}
```

/src/org/example/sudoku/Sudoku.jav

a

```
public void onClick(View v) {  
    switch (v.getId()) {  
        case R.id.continue_button:  
            startGame(Game.DIFFICULTY_CONTINUE);  
            break;  
            // ...  
  
        case R.id.about_button:  
            Intent i = new Intent(this, About.class);  
            startActivity(i);  
            break;  
            // More buttons go here (if any) ...  
        case R.id.new_button:  
            openNewGameDialog();  
            break;  
        case R.id.exit_button:  
            finish();  
            break;  
    }  
}
```

현재 위치 기억하기

- 스도쿠 게임 도중, 화면 방향을 바꾸면 커서가 위치를 잃음 - 사용자가 만든 View인 PuzzleView를 사용하기 때문에 자동으로 상태 저장을 안함
- 지속적(Persistent) 상태와는 달리 인스턴스(Instance) 상태는 영구적이지 않고, 안드로이드 어플리케이션 스택의 Bundle 클래스에 저장됨
- 인스턴스 상태는 커서의 위치와 같은 소량의 정보에 사용됨
- Activity.onSaveInstanceState() 는 안드로이드는 뷰 계층구조를 따라 내려오며 ID를 가진 모든 뷰에서 View.onSaveInstanceState() 호출함. onRestoreInstanceState()도 동일한 작업을 함. 여기서 ID는 XML에서 오지만, PuzzleView는 우리가 만든 것이므로 setId() 메서드로 직접 설정함
- onSaveInstanceState()는 Bundle에 슈퍼클래스를 호출해서 받은 상태 저장 후, 자신의 정보 저장해서 반환.
- onRestoreInstanceState()는 반대로 저장된 정보를 빼내는 것으로, 자기 정보를 먼저 빼내고, 슈퍼 클래스의 상태 정보를 빼내도록 함

/src/org/example/sudoku/PuzzleView.

java

```
private static final String SELX = "selX";  
private static final String SELY = "selY";  
private static final String VIEW_STATE = "viewState";  
private static final int ID = 42;
```

```
public PuzzleView(Context context) {  
    super(context);  
    this.game = (Game) context;  
    setFocusable(true);  
    setFocusableInTouchMode(true);  
    setId(ID);  
}
```

@Override

```
protected Parcelable onSaveInstanceState() {  
    Parcelable p = super.onSaveInstanceState();  
    Log.d(TAG, "onSaveInstanceState");  
    Bundle bundle = new Bundle();  
    bundle.putInt(SELX, selX);  
    bundle.putInt(SELY, selY);  
    bundle.putParcelable(VIEW_STATE, p);  
    return bundle;  
}
```

@Override

```
protected void onRestoreInstanceState(Parcelable state) {  
    Log.d(TAG, "onRestoreInstanceState");  
    Bundle bundle = (Bundle) state;  
    select(bundle.getInt(SELX), bundle.getInt(SELY));  
    super.onRestoreInstanceState(bundle.getParcelable(VIEW_STATE));  
    return;  
}
```

내부 파일 시스템 액세스하기

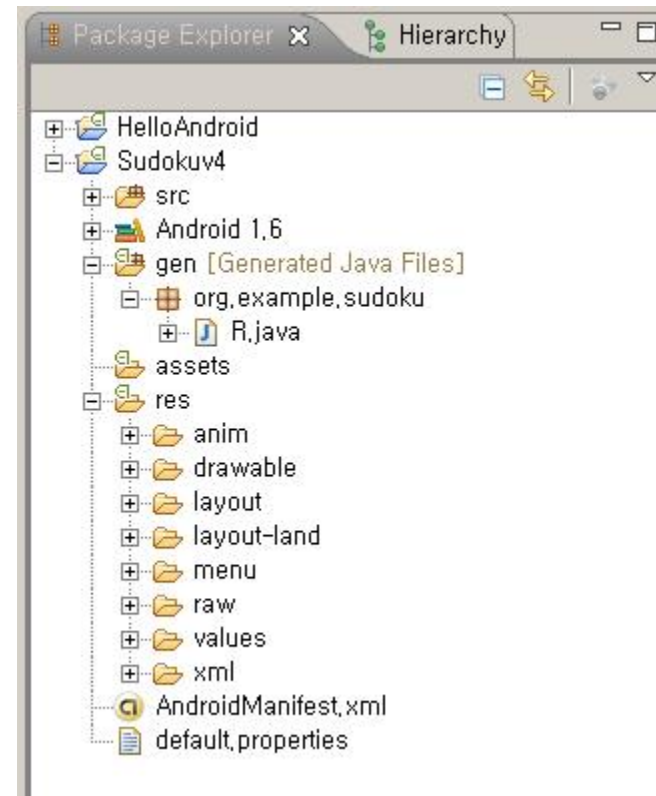
- 안드로이드는 리눅스 기반이므로 실제 파일 시스템이 있음. 플래시 메모리에 저장됨
- java.io 의 모든 입출력 루틴은 사용 가능함
- 주로 액세스하는 곳은 패키지 비공개 디렉토리인 /data/data/package_name
- Context 클래스와 이로부터 확장된 Activity 클래스는 helper 메서드들을 지원함
 - deleteFile(), fileList(), openFileInput(), openFileOutput()
- 결국 내부 메모리이므로 최대 1~2 메가를 넘기지 않도록 하고, 공간이 부족해서 생기는 오류에 조심

SD 카드 액세스하기

- 용량이 커서 비디오/오디오 저장에 이상적이며, 실행되는 프로그램 코드를 저장할 수는 없음
- 명령 - `mksdcard 256M c:\temp\sd.img`
- 8M 이하로 너무 작으면 에뮬레이터가 무시하고, 너무 크면 로딩 시간이 오래 걸림
- 에뮬레이터 실행 시 옵션에서 “-sdcard c:\temp\sd.img” 으로 SD 카드 이미지 경로 지정
- 파일 업로드는 다음과 같이 함
 - `adb push c:\code\samplevideo.mp4 /sdcard/samplevideo.mp4`
- Video.java의 onCreate() 메서드 수정
 - `video.setVideoPath("/sdcard/samplevideo.mp4");`

위젯 (Widget)

- Widget Package
 - Application screen 상에서 사용되는 UI 요소를 포함하는 패키지
 - View 또는 sub-class를 상속받음
- Android UI Structure



android.widget.Button

- 특징
 - push-button widget 클래스
 - user의 행위를 실행시켜 주는 것
 - Listener 등록 필요
- Sub-Class
 - CheckBox, RadioButton, ToggleButton
 - CompoundButton의 하위 클래스
 - checked / unchecked 2가지 상태를 가지는 버튼

Listener 등록

```
Button btn01 = (Button)findViewById(R.id.btn01);  
  
btn01.setOnClickListener(new OnClickListener(){  
  
    @Override  
    public void onClick(View arg0) {  
        Intent oneInt = new Intent();  
        oneInt.putExtra(NAME, "Tom");  
        setResult(RES_OK, oneInt);  
        finish();  
    }  
  
});
```

Button example

```
<LinearLayout
  android:layout_width="wrap_content"
  android:layout_height="wrap_content"
  android:orientation="vertical">
  <Button android:id="@+id/button_normal"
    android:text="@string/buttons_1_normal"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
  <Button android:id="@+id/button_small"
    style="?android:attr/buttonStyleSmall"
    android:text="@string/buttons_1_small"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
  <ToggleButton android:id="@+id/button_toggle"
    android:text="@string/buttons_1_toggle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
</LinearLayout>
```

android.widget.ImageButton

- 특징
 - image를 표시하는 button widget 클래스
 - user의 행위를 실행시켜 주는 것
 - Listener 등록 필요
- subclass
 - ZoomButton

ImageButton example

```
<LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:layout_width="fill_parent"
  android:layout_height="fill_parent"
  android:orientation="vertical">
  <ImageButton
    android:layout_width="100dip"
    android:layout_height="50dip"
    android:src="@android:drawable/sym_action_call" />
  <ImageButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@android:drawable/sym_action_chat" />
  <ImageButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@android:drawable/sym_action_email" />
</LinearLayout>
```

ImageButton example

```
<LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:layout_width="fill_parent"
  android:layout_height="fill_parent"
  android:orientation="vertical">
  <ImageButton
    android:layout_width="100dip"
    android:layout_height="50dip"
    android:src="@android:drawable/sym_action_call" />
  <ImageButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@android:drawable/sym_action_chat" />
  <ImageButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@android:drawable/sym_action_email" />
</LinearLayout>
```

메뉴 (android.view.Menu)

- Menu Interface

- 특징

- Menu 안의 item을 관리하는 Interface
 - 모든 Activity는 option menu 지원

- Menu 추가 방법

- MenuInflater 를 사용해서 menu XML file 추가
 - code를 통해서 추가
 - onOptionsItemSelected(MenuItem)
 - onContextItemSelected(MenuItem)

Menu 종류

Menu Type	Description
Context Menus	shortcut, icon, submenu 지원하지 않음
Options Menus	<ul style="list-style-type: none">• menu 타이틀만 표시• 체크 표시 아이콘 메뉴 지원 하지 않음• 보여지는 item 개수보다 menu item이 많을 경우, 'more' 라고 표시
Sub Menu	<ul style="list-style-type: none">• 메뉴 icon 지원하지 않음• sub menu 지원

Menu 관련 메서드

Prototype	Description
<code>public MenuItem add(int titleRes)</code>	<ul style="list-style-type: none">- menu에 새로운 item 추가- item 표시할 String의 Resource ID- 가장 최근에 추가된 메뉴 item 리턴
<code>public MenuItem add(CharSequence title)</code>	item을 표시할 text String
<code>public MenuItem add(int groupId, int itemId, int order, CharSequence title)</code>	<ul style="list-style-type: none">- item이 속할 group 지정, NONE – 그룹 지정 하지 않음- item의 unique ID, NONE- item 순서, NONE- item을 표시할 text String
<code>public MenuItem add(int groupId, int itemId, int order, int titleRes)</code>	item을 표시할 String의 Resource ID

AndroidManifest.xml 파일에 activity 추가

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
package="com.min.summary"
android:versionCode="1" android:versionName="1.0.0">
<application android:icon="@drawable/icon"
    android:label="@string/app_name">

<activity android:name=".MenuTest" android:label="@string/app_name">
<intent-filter>
<action android:name="android.intent.action.MAIN" />
<category android:name="android.intent.category.LAUNCHER" />
</intent-filter>
</activity>

<activity android:name=".Item" android:label="Menu_Item Activity">
</activity>

</application>
</manifest>
```

Menu Interface Example

```
public class MenuTest extends Activity {
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.scroll_view);
    }
    public boolean onCreateOptionsMenu(Menu menu) {
        menu.add(0, 0, 0, "Go Item View");
        return super.onCreateOptionsMenu(menu);
    }
    public boolean onOptionsItemSelected(MenuItem item) {
        switch( item.getItemId() )
        {
            case 0:
                Intent newActivity = new Intent(this, Item.class );
                try{
                    startActivity( newActivity );
                }catch(ActivityNotFoundException e){ }
                break;
        }
        return super.onOptionsItemSelected( item );
    }
}
```

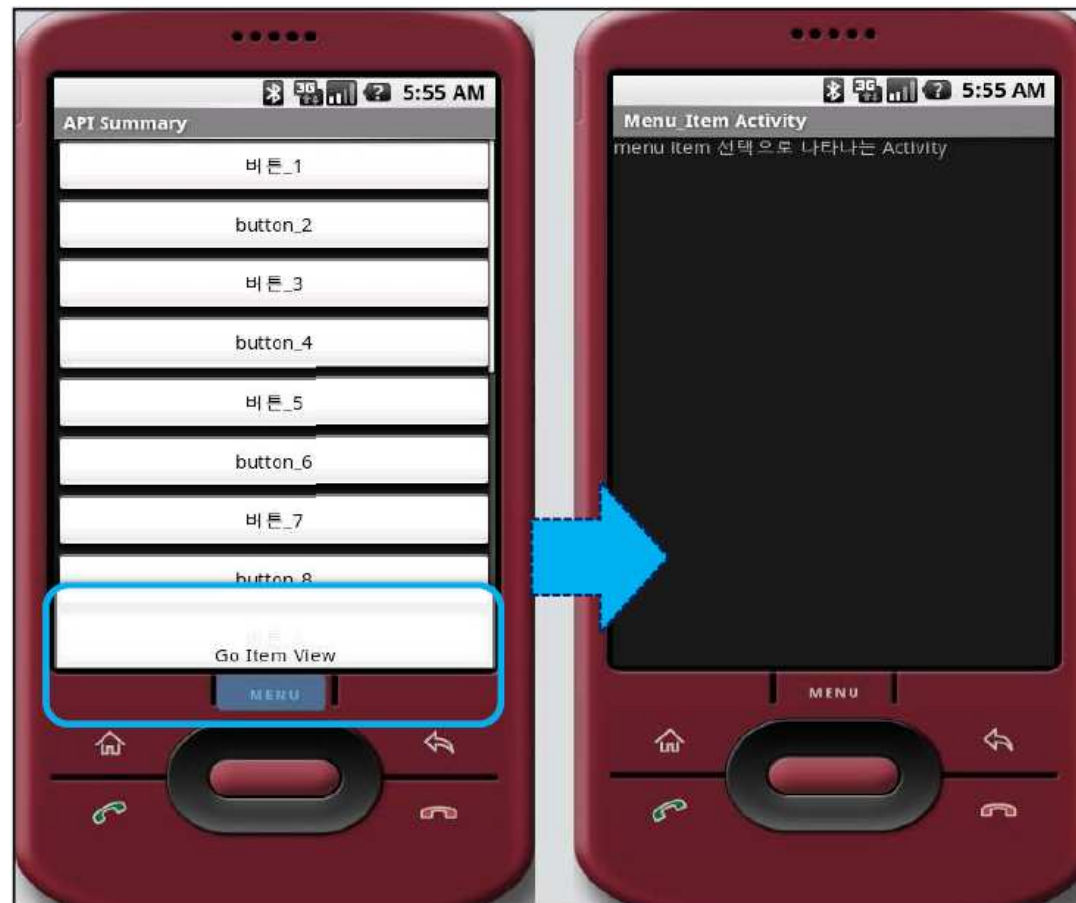
Menu 선택 시 화면 Activity

```
public class Item extends Activity{
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        TextView tv = new TextView(this);
        tv.setText("menu item 선택으로 나타나는 Activity");
        setContentView(tv);
    }
}
```

Menu 선택 시 화면 Activity

```
public class Item extends Activity{
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        TextView tv = new TextView(this);
        tv.setText("menu item 선택으로 나타나는 Activity");
        setContentView(tv);
    }
}
```

Menu Interface Example



다이얼로그 (대화 상자) - android.app.Dialog

- 특징
 - Dialog를 위한 Base Class
 - Activity에서 Dialog 생성, 저장 관리하는 기능 제공
- Sub Classes

Class	Description
AlertDialog	1 ~ 3개 버튼을 가지는 Dialog
-- CharacterPickerDialog	강조된 문자를 선택하는 Dialog
-- DatePickerDialog	calendar를 view로 사용해서 보여주는 클래스
-- ProgressDialog	progress indicator, 추가 text message, view를 보여주는 클래스
TimePickerDialog	- 하루 시간을 24hour 또는 AM/PM 모드로 선택하는 view를 나타내는 Dialog class

android.app.Dialog

- Dialog & Activity 관련 Classes

Method	Description
public final boolean requestWindowFeature(int featureId)	- 확장된 window 기능 사용 가능여 부 - android.view.window에 정의된 기 능

int	DEFAULT_FEATURES	65	0x00000041
int	FEATURE_CONTEXT_MENU	6	0x00000006
int	FEATURE_CUSTOM_TITLE	7	0x00000007
int	FEATURE_INDETERMINATE_PROGRESS	5	0x00000005
int	FEATURE_LEFT_ICON	3	0x00000003
int	FEATURE_NO_TITLE	1	0x00000001
int	FEATURE_OPTIONS_PANEL	0	0x00000000
int	FEATURE_PROGRESS	2	0x00000002
int	FEATURE_RIGHT_ICON	4	0x00000004
int	ID_ANDROID_CONTENT	16908290	0x01020002
int	PROGRESS_END	10000	0x00002710
int	PROGRESS_INDETERMINATE_OFF	-4	0xffffffc
int	PROGRESS_INDETERMINATE_ON	-3	0xffffffd
int	PROGRESS_SECONDARY_END	30000	0x00007530
int	PROGRESS_SECONDARY_START	20000	0x00004e20
int	PROGRESS_START	0	0x00000000
int	PROGRESS_VISIBILITY_OFF	-2	0xffffffe
int	PROGRESS_VISIBILITY_ON	-1	0xfffffff

android.app.Dialog

- Dialog & Activity 관련 Classes
 - android.app.Activity package Method

Method	Description
public Window getWindow()	- activity를 위한 Window 조회해서 리턴
protected Dialog onCreateDialog(int id)	- Dialog 생성을 위해 호출되는 callback method - activity에 의해서 생성 & 저장됨
protected void onPrepareDialog(int id, Dialog dialog)	- Dialog 보여주기 전에 update 할 것이 존재 할 경우 overriding
public final void showDialog(int id)	- Dialog 보여주는 method
public final void dismissDialog(int id)	- ShowDialog()를 통해서 보여준 Dialog를 해제시켜주는 메소드
public final void removeDialog(int id)	- Dialog를 제거하는 method

android.app.Dialog

- AlertDialog Class (android.app.AlertDialog)
 - 특징
 - Dialog의 subclass
 - 1 ~ 3개의 button 표시
 - setMessage() : 문자열 출력
 - 좀 더 복잡한 view : FrameLayout 사용
- 관련 Method

Method	Description
public AlertDialog.Builder setIcon (int iconId) / setIcon (Drawable icon)	- title에 사용될 Drawable한 resource Id 설정
public AlertDialog.Builder setTitle(int titleId) / setTitle(CharSequence title)	- 주어진 resource id / string을 title로 설정
public AlertDialog.Builder setPositiveButton (int textId, DialogInterface.OnClickListener listener)	- 긍정 버튼이 눌러 졌을 때 처리 할 listener 등록 - Listener의 onClick() overriding 해야 함
public AlertDialog.Builder setNegativeButton(int textId, DialogInterface.OnClickListener listener)	- 부정 버튼이 눌러 졌을 때 처리 할 listener 등록 - Listener의 onClick() overriding 해야 함

android.app.Dialog 관련 Method 계속

Method	Description
<code>public AlertDialog.Builder setItems(int itemId, DialogInterface.OnClickListener listener) setItems(CharSequence[] items, DialogInterface.OnClickListener listener)</code>	<ul style="list-style-type: none">- Dialog 안에 표시되는 List item 설정- listener에 의해서 선택된 item 처리- itemId : R.array 처럼 배열 type
<code>public AlertDialog.Builder setSingleChoiceItems(int itemId, int checkedItem, DialogInterface.OnClickListener listener) setSingleChoiceItems(CharSequence[] items, int checkedItem, DialogInterface.OnClickListener listener)</code>	<ul style="list-style-type: none">- Dialog 안에 표시되는 List item 설정- 선택된 item text 오른쪽에 체크 마크 표시- itemId : R.array 처럼 배열 type
<code>public AlertDialog.Builder setPositiveButton (int textId, DialogInterface.OnClickListener listener)</code>	<ul style="list-style-type: none">- 긍정 버튼이 눌러 졌을 때 처리 할 listener 등록- Listener의 onClick() overriding 해야 함
<code>public AlertDialog.Builder setNegativeButton(int textId, DialogInterface.OnClickListener listener) / setNegativeButton(CharSequence text, DialogInterface.OnClickListener listener)</code>	<ul style="list-style-type: none">- 부정 버튼이 눌러 졌을 때 처리 할 listener 등록- Listener의 onClick() overriding
<code>public AlertDialog.Builder setNeutralButton (int textId, DialogInterface.OnClickListener listener) / setNeutralButton (CharSequence text, DialogInterface.OnClickListener listener)</code>	<ul style="list-style-type: none">- 중간 버튼이 눌러 졌을 때 처리 할 Listener 등록- Listener의 onClick() overriding

android.content.DialogInterface class

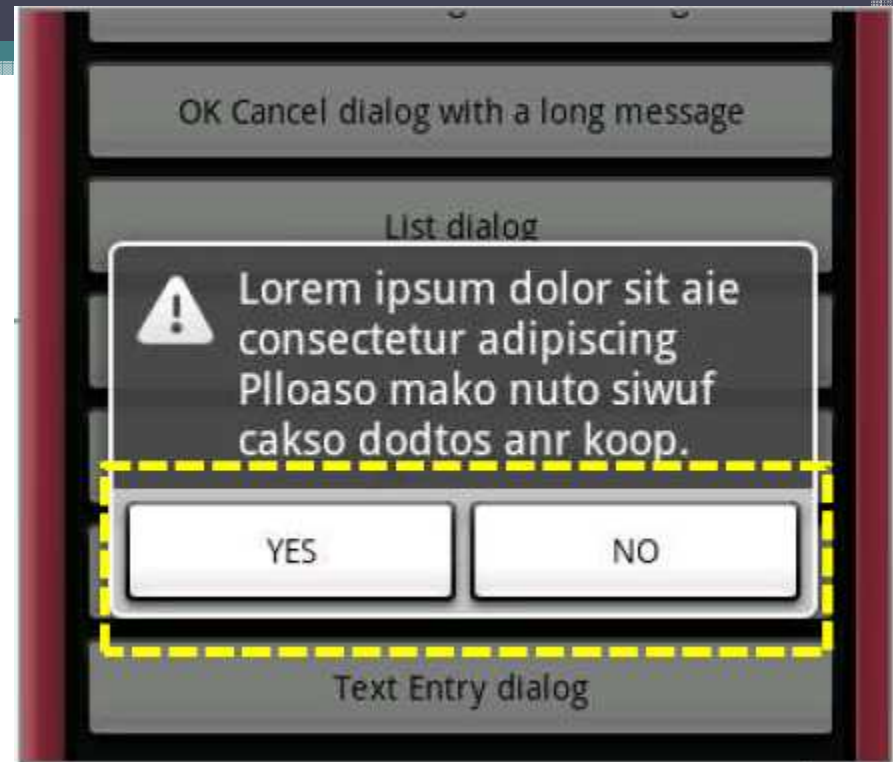
- Dialog 창의 버튼에 대한 처리를 하는 Interface

Classes	Description
DialogInterface.OnCancelListener	- cancel 되었을 때 처리할 일
DialogInterface.OnClickListener	- click 되었을 때 처리 할 일
DialogInterface.OnDismissListener	- dismiss 되었을 때 처리 할 일
DialogInterface.OnKeyListener	- key event 발생되었을 때 처리 할 일
DialogInterface.OnMultiChoiceClickListener	- 다중 선택 시 처리 할 일

AlertDialog class

예제 01

```
public class SoDialogTEST extends Activity {  
-----  
public void onCreate(Bundle savedInstanceState) {  
-----  
    protected Dialog onCreateDialog(int id) {  
        AlertDialog myAlert;  
        switch(id) {  
        case DIALOG_OK_CANCEL_NO_MESSAGE:  
            myAlert = new AlertDialog.Builder( SoDialogTEST.this)  
                .setIcon( R.drawable.alert_dialog_icon )  
                .setTitle( R.string.alert_dialog_two_buttons_title)  
                .setPositiveButton( R.string.alert_dialog_ok, new DialogInterface.OnClickListener(){  
                    @Override public void onClick(DialogInterface arg0, int arg1) { }  
                })  
                .setNegativeButton( R.string.alert_dialog_cancel, new DialogInterface.OnClickListener(){  
                    @Override public void onClick(DialogInterface arg0, int arg1) { }  
                })  
                .create();  
            return myAlert;  
        }  
    }  
}
```



AlertDialog class

예제 02

```
public class SoDialogTEST extends Activity {  
-----  
public void onCreate(Bundle savedInstanceState) {  
-----  
protected Dialog onCreateDialog(int id) {  
    AlertDialog myAlert;  
    switch(id)  
    {  
        case DIALOG_LIST:  
            myAlert = new AlertDialog.Builder(SoDialogTEST.this)  
                .setTitle(R.string.select_dialog)  
                .setItems(R.array.select_dialog_items, new DialogInterface.OnClickListener() {  
                    public void onClick(DialogInterface dialog, int which) {  
                        String[] items = getResources().getStringArray(R.array.select_dialog_items);  
                        new AlertDialog.Builder(SoDialogTEST.this)  
                            .setMessage("You selected: " + which + ", " + items[which])  
                            .show();  
                    }  
                })  
            .create();  
            return myAlert;  
        }  
    }  
}
```



android.widget.ImageView

- 특징
 - image를 표시하는 class
 - 다양한 source (resource, content provider ..)로부터 이미지 로딩
 - image 치수 계산 layout manager에 사용됨
 - 다양한 표시 옵션 제공
- Nested Class

Constant	Value
matrix	0
fitXY	1
fitStart	2
fitCenter	3
fitEnd	4
center	5
centerCrop	6
centerInside	7

Nested Class	Description
ImageView.ScaleType	- image, view의 범위를 위한 option class

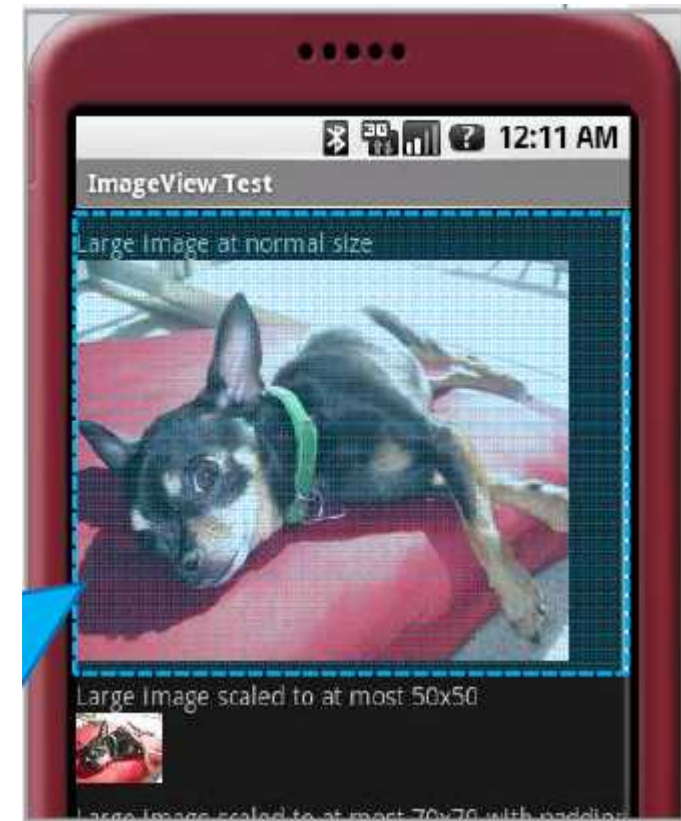
- 관련 Sub-Classes – ImageButton, ZoomButton

ImageView class - XML Attributes

Attributes Name	Related Methods	Description
android:adjustViewBounds	setAdjustViewBounds(boolean)	- ImageView를 drawable한 비율에 맞추기 설정
android:maxHeight	setMaxHeight(int)	- view의 최대 height 설정
android:maxLength	setMaxLength(int)	- view의 최대 width 설정
android:scaleType	setScaleType(ImageView.ScaleType)	- ImageView에 맞게 image를 어떻게 제어할지 설정
android:src	setImageResource(int)	- ImageView에 그려질 수 있는 content 설정
android:tint	setColorFilter(int, PorterDuff.Mode)	- Image의 tint color 설정

ImageView class - 예제 01

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView
  xmlns:android="http://schemas.android.com/apk/res/a
  ndroid"
  android:layout_width="fill_parent"
  android:layout_height="fill_parent">
  <LinearLayout android:layout_width="fill_parent"
  android:layout_height="fill_parent"
  android:orientation="vertical">
  <TextView android:layout_width="fill_parent"
  android:layout_height="wrap_content"
  android:paddingTop="10dip"
  android:text="@string/image_view_large_normal" />
  <ImageView android:src="@drawable/sample_1"
  android:adjustViewBounds="true"
  android:layout_width="wrap_content"
  android:layout_height="wrap_content" />
```



android.widget.ImageButton

- 특징
 - image 버튼을 표시하는 class
 - press, click 가능
- XML Attributes

Attributes	Description
android.widget.ImageView	- ImageView 속성 상속 받음
android.view.View	- View 속성 상속 받음

android.webkit.WebView

- 특징
 - web browsing을 위한 tool 제공하는 클래스 중 하나
 - web page 표시하는 View
 - WebKit 엔진 사용
- 주의 사항
 - Internet 접속 및 web page 로드를 위한 설정
 - AndroidManifest.xml 에 추가

```
<manifest
  xmlns:android="http://schemas.android.com/apk/res/android"
  package="com.example.android.apis">
  <uses-permission
    android:name="android.permission.INTERNET" />
  ...
</manifest>
```

android.webkit.WebView 관련 Method

Method	Description
public boolean canGoBack()	- back history item을 가졌는지 여부
public boolean canGoBackOrForward(int steps)	- 지정된 수 만큼 go back / forward 가능 여부
public Picture capturePicture()	- 현재 표시되고 있는 webView를 캡처하는 기능
public void clearView()	- view를 지우고, onDraw()는 흰색 배경을 drawing
public String[] getHttpAuthUsernamePassword(String host, String realm)	- Http 인증 username, password 조회
public String getTitle ()	- 현재 page의 title 가져오는 기능
public void loadData (String data, String mimeType, String encoding)	- webView에 주어진 데이터 로드
public void loadUrl(String url)	- 주어진 url 로드
public void loadDataWithBaseUrl (String baseUrl, String data, String mimeType, String encoding, String failUrl)	- webView에 주어진 데이터 로드하고, -주어진 URL은content의 base URL로 적용 (base URL : page 로딩 시 보여줄 화면)

android.webkit.WebView 예제

```
public class WebBrowserMain extends ListActivity {

    //-----
    //- Member Variable
    //-----
    private final String LOG_TAG=" ==> WebBrowserMain : ";
    private String[] strLists = {"Call the Browser", "Using WebView"};

    //-----
    //- Overriding Method
    //-----
    //- List 생성
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        Log.i(LOG_TAG, "onCreate()");
        //- ListActivity 경우 기본 제공 List layout 존재 하기 때문에
        //- List Item 설정 및 data 설정
        ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
            android.R.layout.simple_list_item_1, strLists);
        setListAdapter(adapter);
    }
}
```

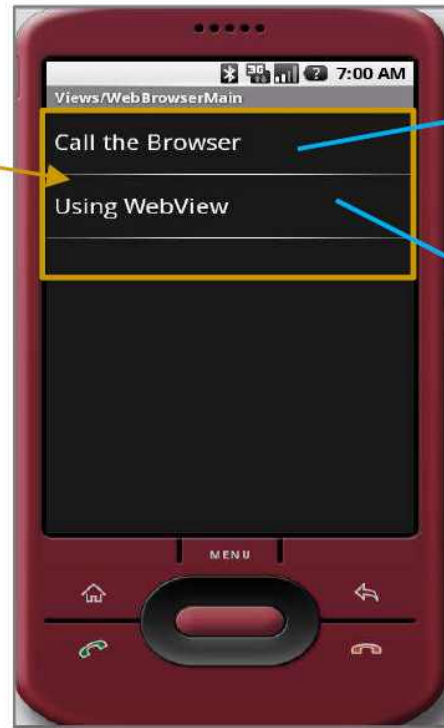
android.webkit.WebView 예제

```
public class WebBrowserMain extends ListActivity {  
-----  
    protected void onListItemClick(ListView l, View v, int position, long id) {  
        super.onListItemClick( l, v, position, id );  
        switch(position)  
        {  
            case 0:  
                Log.i(LOG_TAG, "onListItemClick() - goBrowser");  
                Uri homeURI = Uri.parse("http://www.google.co.kr");  
                Intent goBrowser = new Intent( Intent.ACTION_VIEW, homeURI );  
                startActivity(goBrowser);  
                break;  
  
            case 1:  
                Log.i(LOG_TAG, "onListItemClick() - goGoogle");  
                Intent goMyBrowser = new Intent( this, WebBrowser.class );  
                startActivity( goMyBrowser );  
                break;  
        }  
    }  
}
```

android.webkit.WebView 예제

- WebView 예제

◆ WebBrowserMain
- Android System에서
지원하는 기본 List



android.webkit.WebView 예제

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/main" android:layout_width="fill_parent" android:layout_height="fill_parent"
    android:clickable="true">
    <EditText android:id="@+id/insert_url" android:layout_width="fill_parent"
        android:layout_height="wrap_content" android:autoText="true"/>
    <WebView android:id="@+id/browser"
        android:layout_width="fill_parent" android:layout_height="fill_parent"
        android:layout_below="@id/insert_url" android:clickable="true"/>
    <LinearLayout android:layout_width="fill_parent" android:layout_height="wrap_content"
        android:orientation="horizontal" android:layout_alignParentBottom="true">
    <Button android:id="@+id/btn_goback" android:layout_height="wrap_content"
        android:layout_width="wrap_content" android:text="@string/go_back"/>
    <Button android:id="@+id/btn_reload" android:layout_height="wrap_content"
        android:layout_width="wrap_content" android:text="@string/go_reload"/>
    <Button android:id="@+id/btn_zoomout" android:layout_height="wrap_content"
        android:layout_width="wrap_content" android:text="@string/zoomOut"/>
    <Button android:id="@+id/btn_goURL" android:layout_height="wrap_content"
        android:layout_width="wrap_content" android:text="@string/go_URL"/>
    </LinearLayout>
</RelativeLayout>
```

android.webkit.WebView 예제

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
<!-- ===== -->
<!-- myBrowser examples strings -->
<!-- ===== -->
<string name="go_back">뒤로</string>
<string name="go_home">홈</string>
<string name="go_reload">다시</string>
<string name="zoomIn">확대</string>
<string name="zoomOut">축소</string>
<string name="go_URL">url...</string>
<string name="default_msg">url 입력하세요.</string>
</resources>
```


android.webkit.WebView 예제

```
public class WebBrowser extends Activity {
    //-----
    //- Member Variable
    //-----
    private final String LOG_TAG=" ==> WebBrowser : ";
    private String strData = "<html><body>My Browser<br><a
href='http://www.naver.com'>Naver!</a></body></html>";
    private String strMimeType = "text/html";
    private String strEncoding = "utf-8";
    WebView mBrowserView;
    //-----
    //- Overriding Method
    //-----
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        Log.i(LOG_TAG, "onCreate() ");
        setContentView( R.layout.mybrowser );
        mBrowserView = (WebView)findViewById( R.id.browser );
        mBrowserView.loadData( strData, strMimeType, strEncoding );
    }
}
```

android.webkit.WebView 예제

```
protected void onCreate(Bundle savedInstanceState) {  
    -----  
    //- Button 처리  
    Button btnGoBack = (Button)findViewById(R.id.btn_goback);  
    btnGoBack.setOnClickListener(new OnClickListener(){  
        @Override  
        public void onClick(View arg0) {  
            Log.i(LOG_TAG, " btnGoBack ");  
            mBrowserView.goBack();  
        }  
    });  
    Button btnGoHome = (Button)findViewById(R.id.btn_gohome);  
    btnGoHome.setOnClickListener(new OnClickListener(){  
        public void onClick(View arg0) {  
            Log.i(LOG_TAG, " btnGoHome ");  
            mBrowserView.loadUrl("http://www.naver.com");  
        }  
    });  
}
```

android.webkit.WebView 예제

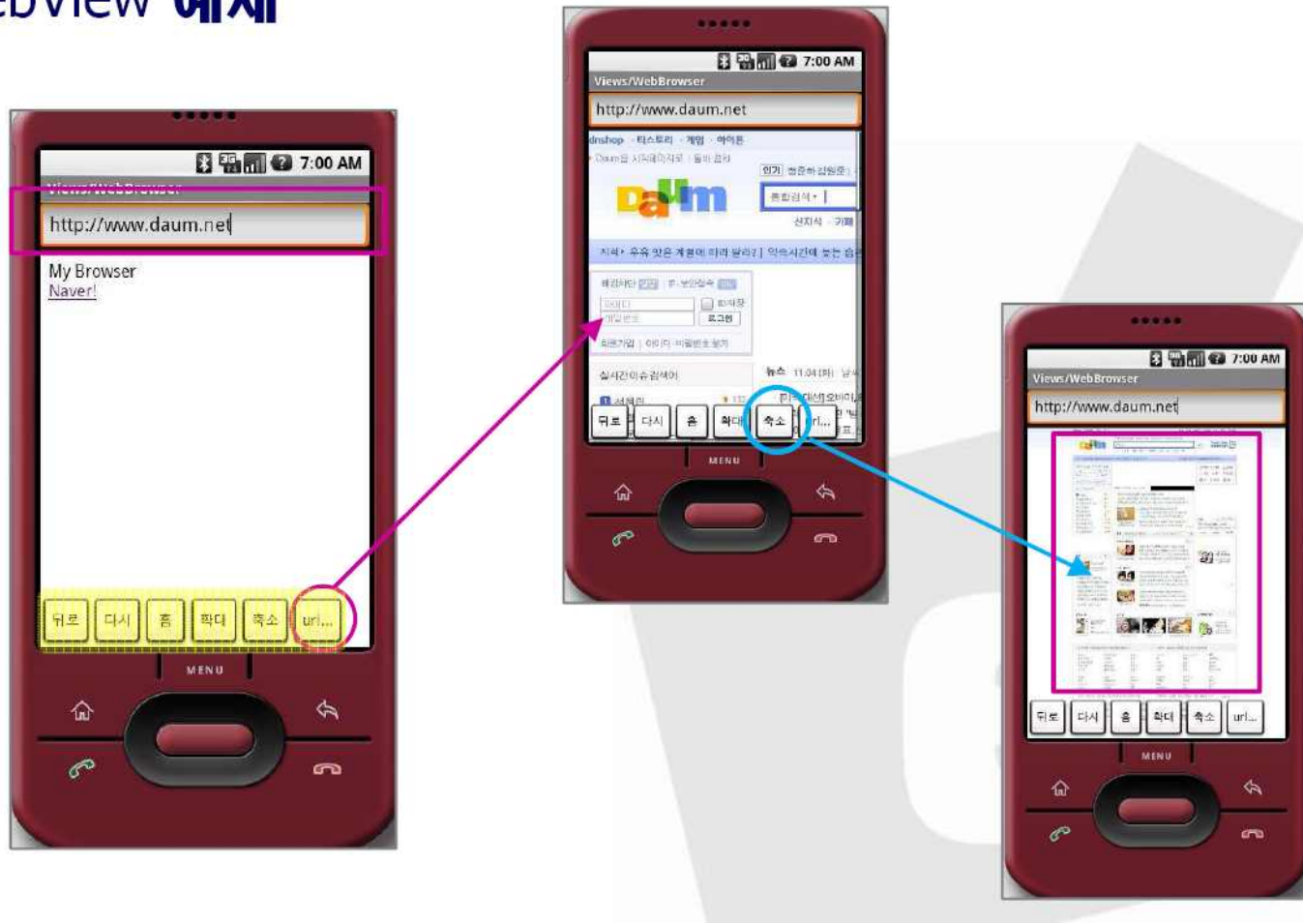
```
protected void onCreate(Bundle savedInstanceState) {  
    -----  
    //- Button 처리  
    Button btnReload = (Button)findViewById( R.id.btn_reload );  
    btnReload.setOnClickListener( new OnClickListener() {  
        public void onClick(View arg0) {  
            Log.i(LOG_TAG, " btnReload ");  
            mBrowserView.reload();  
        }  
    });  
    Button btnZoomIn = (Button)findViewById( R.id.btn_zommin);  
    btnZoomIn.setOnClickListener( new OnClickListener(){  
        public void onClick(View arg0) {  
            Log.i(LOG_TAG, " btnZoomIn ");  
            mBrowserView.zoomIn();  
        }  
    });  
};
```

android.webkit.WebView 예제

```
protected void onCreate(Bundle savedInstanceState) {  
    -----  
    //- Button 처리  
    Button btnZoomOut = (Button)findViewById(R.id.btn_zoomout);  
    btnZoomOut.setOnClickListener(new OnClickListener(){  
        public void onClick(View arg0) {  
            Log.i(LOG_TAG, " btnZoomOut ");  
            mBrowserView.zoomOut();  
        }  
    });  
    Button btnURL = (Button)findViewById(R.id.btn_goURL);  
    btnURL.setOnClickListener(new OnClickListener(){  
        public void onClick(View arg0) {  
            Log.i(LOG_TAG, " btnZoomOut ");  
            EditText insertTxt = (EditText)findViewById(R.id.insert_url);  
            String strURL = insertTxt.getText().toString();  
            mBrowserView.loadUrl(strURL);  
        }  
    });  
}
```

android.webkit.WebView 예제

- WebView 예제



음악 재생에 Service 클래스를 배경으로 사용 (한백전자의 예제)

- 프로젝트 생성
 - Project Name – TestSound
 - Package Name – hanback.example.testsound
 - Activity Name – TestSound
 - Application Name – Test Sound
- mp3 가져오기
 - res폴더 아래에 raw 폴더 생성
 - 가져오기(import) – 파일 시스템 선택
- MusicService Class 생성
 - 이름 : MusicService
 - 슈퍼클래스 : android.app.Service
 - onStart(), onDestroy() 구현

MusicService.java

```
package hanback.example.testsound;

import android.app.Service;
import android.content.Intent;
import android.media.MediaPlayer;
import android.os.Binder;
import android.os.IBinder;

public class MusicService extends Service {
    private Binder binder;
    MediaPlayer mp;
    @Override
    public IBinder onBind(Intent arg0) {
        // TODO Auto-generated method stub
        this.binder = new Binder();                return null;
    }
    @Override
    public void onDestroy() {
        // TODO Auto-generated method stub
        this.mp.stop();                          super.onDestroy();
    }
    @Override
    public void onStart(Intent intent, int startId) {
        // TODO Auto-generated method stub
        try
        {
            this.mp = MediaPlayer.create(this, R.raw.night);        this.mp.start();
        }
        catch (Exception e) {}
        super.onStart(intent, startId);
    }
}
```

TestSound.java

```
package hanback.example.testsound;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;

public class TestSound extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        Button playButton = (Button)findViewById(R.id.play_button);
        playButton.setOnClickListener(new OnClickListener() {
            public void onClick(View view)
            {
                Intent intent = new Intent(TestSound.this, MusicService.class);
                startService(intent);
            }
        });
        Button stopButton = (Button)findViewById(R.id.stop_button);
        stopButton.setOnClickListener(new OnClickListener() {
            public void onClick(View view)
            {
                Intent intent = new Intent(TestSound.this, MusicService.class);
                stopService(intent);
            }
        });
    }
}
```


요약

- 게임이나 프로그램에 옵션을 추가하는 방법을 알아보았다.
- 이전의 프로그램을 계속하기 위해 상태를 저장하는 방법을 알아보았다.
- 게임에서 현재의 실행 위치를 저장하는 방법에 대해 알아보았다.
- 내부의 파일 시스템을 읽고 쓰는 방법에 대해 알아보았다.
- SD 카드에 접근하는 방법에 대해 알아보았다.
- 여러 UI 위젯들에 대해 알아보았다.
- 음악 재생에 있어 Service 클래스를 이용하는 방법을 알아보았다.

퀴즈

- 게임이나 프로그램에 옵션을 추가하는 방법을 단계적으로 설명하라.
- 이전의 프로그램을 계속하기 위해 상태를 저장하려면 어떤 메서드들을 오버라이드해야 하는가?
- 지속적 상태와 인스턴스 상태의 차이는 무엇인가?
- 안드로이드에서 내부의 파일 시스템을 액세스하기 위해서는 어떻게 해야 하는가? 특별히 해주어야 할 일은 무엇인가?
- 안드로이드에서 SD 카드 이미지를 만들려면 어떻게 해야 하는가?
- 에뮬레이터 실행 시에 SD 카드 이미지를 지정해서 읽으려면 어떻게 해야 하는가?
- 웹 서버를 만드는 것과 웹 브라우저를 만드는 것 중 어느 것이 더 어려운가?

연습문제

- 안드로이드 용 이미지 뷰어를 만들어라. 이 뷰어는 내부 메모리나 SD 카드의 특정 디렉토리의 이미지 파일들을 읽어 들여 파일 이름들을 리스트로 보여주고 사용자가 선택하면 그것을 디스플레이 해 준다.
- 앞의 웹 브라우저 예제를 기반으로 안드로이드용 웹 브라우저를 만들어 보라. 기능은 구현할 수 있는 모든 기능을 다 동원해서 구현해 보라.