



군산대학교
KUNSAN NATIONAL UNIVERSITY

Advanced User Interface

모바일 응용

주요 강의자료 인용 : 알짜만 골라배우는, 시작하세요 안드로이드, Professional Android

화면 전환- Forwarding

❖ 화면 Forwarding의 구현

✓ APIDemo/app/forwarding



화면 전환- Forwarding

```
public class Forwarding extends Activity
{
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);

        setContentView(R.layout.forwarding);

        // Watch for button clicks.
        Button goButton = (Button)findViewById(R.id.go);
        goButton.setOnClickListener(mGoListener);
    }

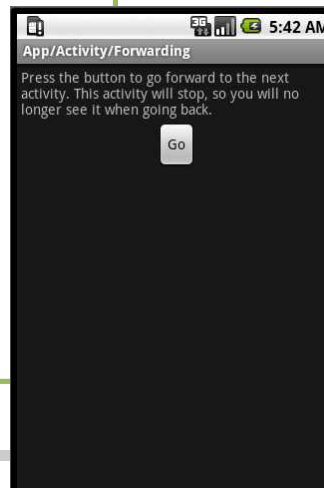
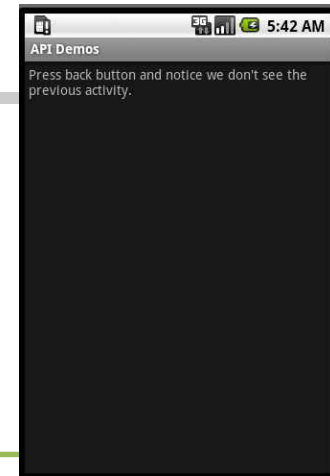
    private OnClickListener mGoListener = new OnClickListener()
    {
        public void onClick(View v)
        {
            // Here we start the next activity, and then call finish()
            // so that our own will stop running and be removed from the
            // history stack.
            Intent intent = new Intent();
            intent.setClass(Forwarding.this, ForwardTarget.class);
            startActivity(intent);
            finish();
        }
    };
}
```

click



```
public class ForwardTarget extends Activity
{
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);

        setContentView(R.layout.forward_target);
    }
}
```



화면 전환 – 결과값 받기

❖ 화면 결과값 받기의 구현

✓ APIDemo/app/ReceiveResult



화면 전환 – 결과값 받기

❖ 화면 결과값 받기의 구현

```
public class ReceiveResult extends Activity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        // Be sure to call the super class.
        super.onCreate(savedInstanceState);

        // See assets/res/any/layout/hello_world.xml for this
        // view layout definition, which is being set here as
        // the content of our screen.
        setContentView(R.layout.receive_result);

        // Retrieve the TextView widget that will display results.
        mResults = (TextView)findViewById(R.id.results);

        // This allows us to later extend the text buffer.
        mResults.setText(mResults.getText(), TextView.BufferType.EDITABLE);

        // Watch for button clicks.
        Button getButton = (Button)findViewById(R.id.get);
        getButton.setOnClickListener(mGetListener);
    }
}
```

```
@Override
protected void onActivityResult(int requestCode, int resultCode,
    Intent data) {
    // requestCode는 MultipleChild Activity들로 부터 구분하기 위해 사용
    if (requestCode == GET_CODE) {

        // We will be adding to our text.
        Editable text = (Editable)mResults.getText();

        // This is a standard resultCode
        if (resultCode == RESULT_CANCELED) {
            text.append("(cancelled)");

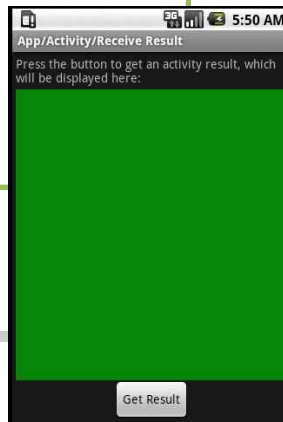
            // Our protocol with the sending activity is that it will send
            // text in 'data' as its result.
        } else {
            text.append("(okay ");
            text.append(Integer.toString(resultCode));
            text.append(" ");
            if (data != null) {
                text.append(data.getAction());
            }
        }

        text.append("\n");
    }
}
```

// Definition of the one requestCode we use for receiving results.
static final private int GET_CODE = 0;

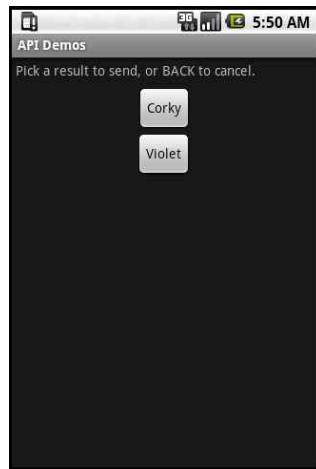
```
private OnClickListener mGetListener = new OnClickListener() {
    public void onClick(View v) {
        // Start the activity whose result we want to retrieve. The
        // result will come back with request code GET_CODE.
        Intent intent = new Intent(ReceiveResult.this, SendResult.class);
        startActivityForResult(intent, GET_CODE);
    }
};

private TextView mResults;
}
```



화면 전환 – 결과값 받기

❖ 화면 결과값 받기의 구현



```
public class SendResult extends Activity
{
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        // Be sure to call the super class.
        super.onCreate(savedInstanceState);

        setContentView(R.layout.send_result);

        // Watch for button clicks.
        Button button = (Button)findViewById(R.id.corky);
        button.setOnClickListener(mCorkyListener);
        button = (Button)findViewById(R.id.violet);
        button.setOnClickListener(mVioletListener);
    }

    private OnClickListener mCorkyListener = new OnClickListener()
    {
        public void onClick(View v)
        {
            setResult(RESULT_OK, (new Intent()).setAction("Corky!"));
            finish();
        }
    };

    private OnClickListener mVioletListener = new OnClickListener()
    {
        public void onClick(View v)
        {
            // To send a result, simply call setResult() before your
            // activity is finished.
            setResult(RESULT_OK, (new Intent()).setAction("Violet!"));
            finish();
        }
    };
}
```

실습

❖ 게임 설정 화면 만들기 예

✓ 심리 테스트 만들기

— 예

- <http://blog.naver.com/cyjsumin?Redirect=Log&logNo=30081310608>
- <http://blog.naver.com/cyjsumin?Redirect=Log&logNo=30080900184>

<div><div></div><div>동의</div></div>	<div><div><input type="radio"/> 초보자 <input type="radio"/> 중급자</div><div>취미</div><div><input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/></div><div>next</div></div>	<div><div></div><div>next back</div></div>	<div><div></div><div>저장 back</div></div>
설문조사 소개	설문조사1	설문조사2	설문조사 종합

백그라운드 프로그래밍

❖ 백그라운드 UI 프로그래밍의 세가지 방법

- Handler 클래스를 이용하는 방법
 - Message Handler 클래스를 이용하여 프로그래밍
 - 화면 UI가 On View 상태인 경우 유용
- View 객체의 postDelayed() 메소드를 이용하는 방법
 - 실제로는 내부적으로 Handler 클래스를 이용하나, Handler를 직접 호출하지 않으므로 코드가 약간 깔끔해지는 효과가 있음
- AlarmManager
 - OS의 Alarm 기능을 이용하여 프로그래밍
 - 화면이 보이지 않다가 특정 시간이 되면 기동하도록 할때 유용

백그라운드 프로그래밍

❖ 메시지 핸들러(Message Handler) 프로그래밍

✓ Handler란?

- android.os.Handler 패키지 클래스로 존재
- Activity 당 하나씩 생성할 수 있으며, 생성하기만 하면 자동 등록됨

✓ Handler가 인스턴스와 통신하는 방법

- Message를 이용하는 방법
- Runnable 객체를 이용하는 방법

백그라운드 프로그래밍

❖ Handler 프로그래밍-message

- ✓ Handler의 생성과 메시지를 수신할 메소드를 구현

```
public Handler handler = new Handler() {  
    @Override  
    public void handleMessage(Message msg)  
    {  
        ←  
    }  
};
```

```
switch( msg.what) {  
  
}
```

- ✓ 메시지 만들기

```
Message msg = handler.obtainMessage();  
msg.what = 1234;  
msg.arg1 = 1111;  
msg.arg2 = 2222;  
msg.object = Object;
```

```
Message msg = Message.obtain( handler, 1234, 111, 222 );
```

- ✓ 메시지 보내기

```
handler.sendMessage(msg);
```

백그라운드 프로그래밍

❖ Handler 프로그래밍 – post, postDelayed

- ✓ postDelayed 를 이용하여 Runnable 등록하기

```
Handler mHandler = new Handler();  
mHandler.postDelayed(new Runnable() {  
    //Do Something  
}, 3000);
```

- ✓ postDelayed에 Runnable 직접 등록하기

```
View.postDelayed(Runnable action, long delayMillis)
```

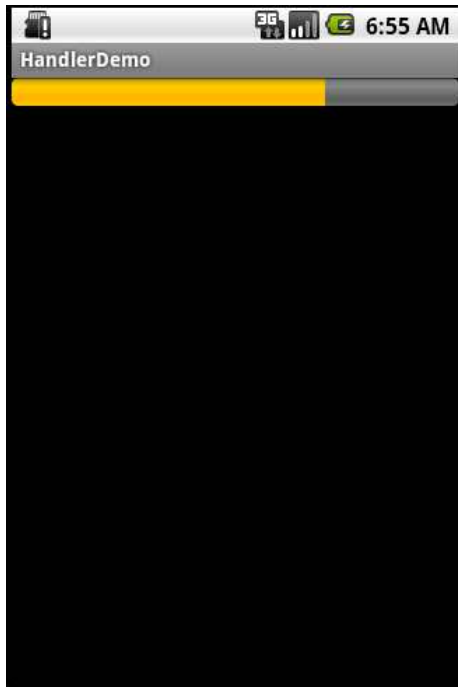
백그라운드 프로그래밍

❖ AlarmManager를 이용하기

```
final Intent intent = new Intent(context, MyService.class);  
final PendingIntent pending = PendingIntent.getService(context, 0, intent, 0);  
  
Calendar cal = nextScheduleTime();  
  
AlarmManager am = (AlarmManager)context.getSystemService(ALARM_SERVICE);  
am.setRepeating(AlarmManager.RTC_WAKEUP, cal.getTimeInMillis(), 5 * 60 * 1000, pending);
```

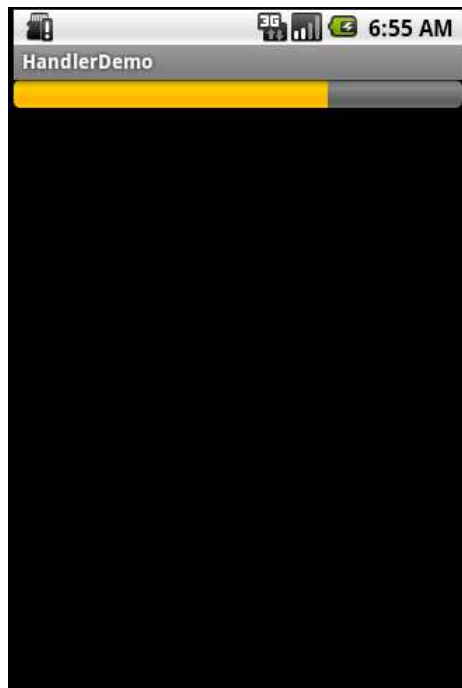
백그라운드 프로그래밍

❖ Handler 프로그래밍 예제



```
public class HandlerDemo extends Activity {  
    ProgressBar bar;  
    Handler handler=new Handler() {  
        @Override  
        public void handleMessage(Message msg) {  
            bar.incrementProgressBy(5);  
        }  
    };  
    AtomicBoolean isRunning=new AtomicBoolean(false);  
}
```

백그라운드 프로그래밍



```
@Override
public void onCreate(Bundle icle) {
    super.onCreate(icle);
    setContentView(R.layout.main);
    bar=(ProgressBar)findViewById(R.id.progress);
}

public void onStart() {
    super.onStart();
    bar.setProgress(0);

    Thread background=new Thread(new Runnable() {
        public void run() {
            try {
                for (int i=0;i<20 && isRunning.get();i++) {
                    Thread.sleep(1000);
                    handler.sendMessage(handler.obtainMessage());
                }
            } catch (Throwable t) {
                // just end the background thread
            }
        }
    });

    isRunning.set(true);
    background.start();
}

public void onStop() {
    super.onStop();
    isRunning.set(false);
}
```

AdapterView 컨테이너들

❖ Adapter를 이용할수 있는 View 컨테이너들

- ✓ ListView
- ✓ GridView
- ✓ GalleryView

❖ Adapter 클래스들

- ✓ CursorAdapter
- ✓ ArrayAdapter
- ✓ ListAdapter

AdapterView 컨테이너들

❖ ArrayAdapter 사용하기

```
private String[] items = { "item 1", "item 2", ...};  
ArrayAdapter arradt = new ArrayAdapter<String>( this, R.layout.textview, items);
```

❖ CursorAdapter 사용하기

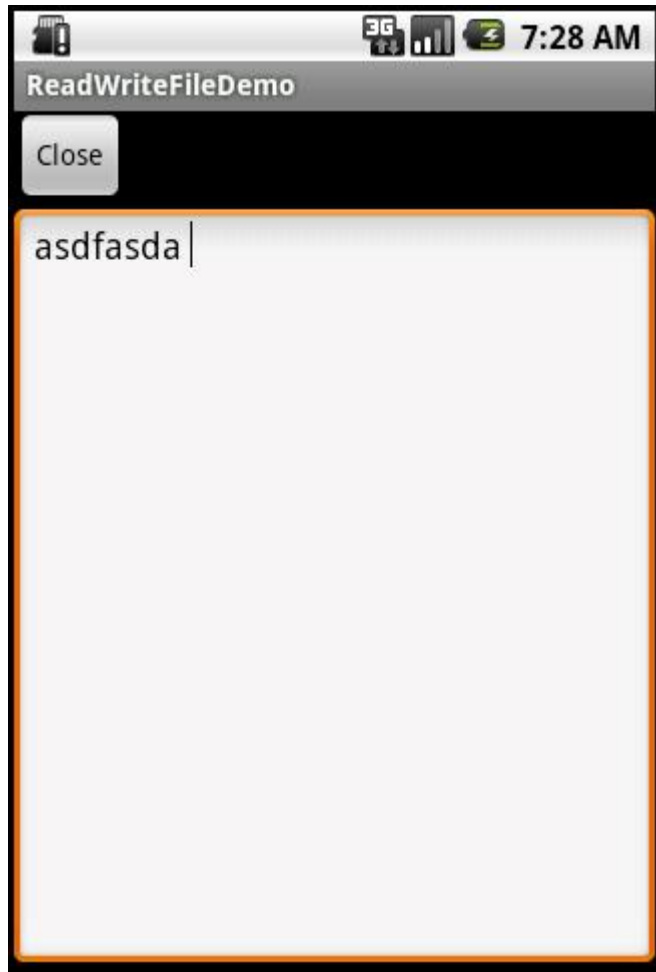
```
Cursor names = managedQuery( Contacts.Phones.CONTENT_URI, null, null, null, null);  
  
startManagingCursor(names);  
ListAdapter adapater = new SimpleCursorAdapter(  
    this, R.layout.two_text,  
    names, new String[] {  
        Contacts.Phones.NAME, Contatcs.Phones.NUMBER},  
    new int[] {  
        R.id.scratch_text1, R.id.scratch_text2  
    }  
);
```

AdapterView 컨테이너들

❖ 자료를 AdapterView 컨테이너에 Binding 하기

```
((ListView)findViewById(R.id.list)).setAdapter(adapte);
```

간단한 File ReadWrite



```
public class ReadWriteFileDemo extends Activity {  
    private final static String NOTES="notes.txt";  
    private EditText editor;  
  
    @Override  
    public void onCreate(Bundle icle) {  
        super.onCreate(icle);  
        setContentView(R.layout.main);  
        editor=(EditText)findViewById(R.id.editor);  
  
        Button btn=(Button)findViewById(R.id.close);  
  
        btn.setOnClickListener(new Button.OnClickListener() {  
            public void onClick(View v) {  
                finish();  
            }  
        });  
    }  
}
```

간단한 File ReadWrite

```
public void onResume() {
    super.onResume();

    try {
        InputStream in=openFileInput(NOTES);

        if (in!=null) {
            InputStreamReader tmp=new InputStreamReader(in);
            BufferedReader reader=new BufferedReader(tmp);
            String str;
            StringBuilder buf=new StringBuilder();

            while ((str = reader.readLine()) != null) {
                buf.append(str+"\n");
            }

            in.close();
            editor.setText(buf.toString());adfasd
        }
    } catch (java.io.FileNotFoundException e) {
        // that's OK, we probably haven't created it yet
    }
    catch (Throwable t) {
        Toast
        .makeText(this, "Exception: "+t.toString(), 2000)
        .show();
    }
}
```

```
public void onPause() {
    super.onPause();

    try {
        OutputStreamWriter out=
            new OutputStreamWriter(openFileOutput(NOTES, 0));

        out.write(editor.getText().toString());
        out.close();
    }
    catch (Throwable t) {
        Toast
        .makeText(this, "Exception: "+t.toString(), 2000)
        .show();
    }
}
```