

Ch07. 메뉴와 대화상자 2

모바일앱프로그래밍

김지심교수

학습목표



메뉴를 작성할 수 있다.
토스트를 다양하게 출력할 수 있다.
대화상자를 사용할 수 있다.

목차

1. 메뉴
2. 대화상자



02



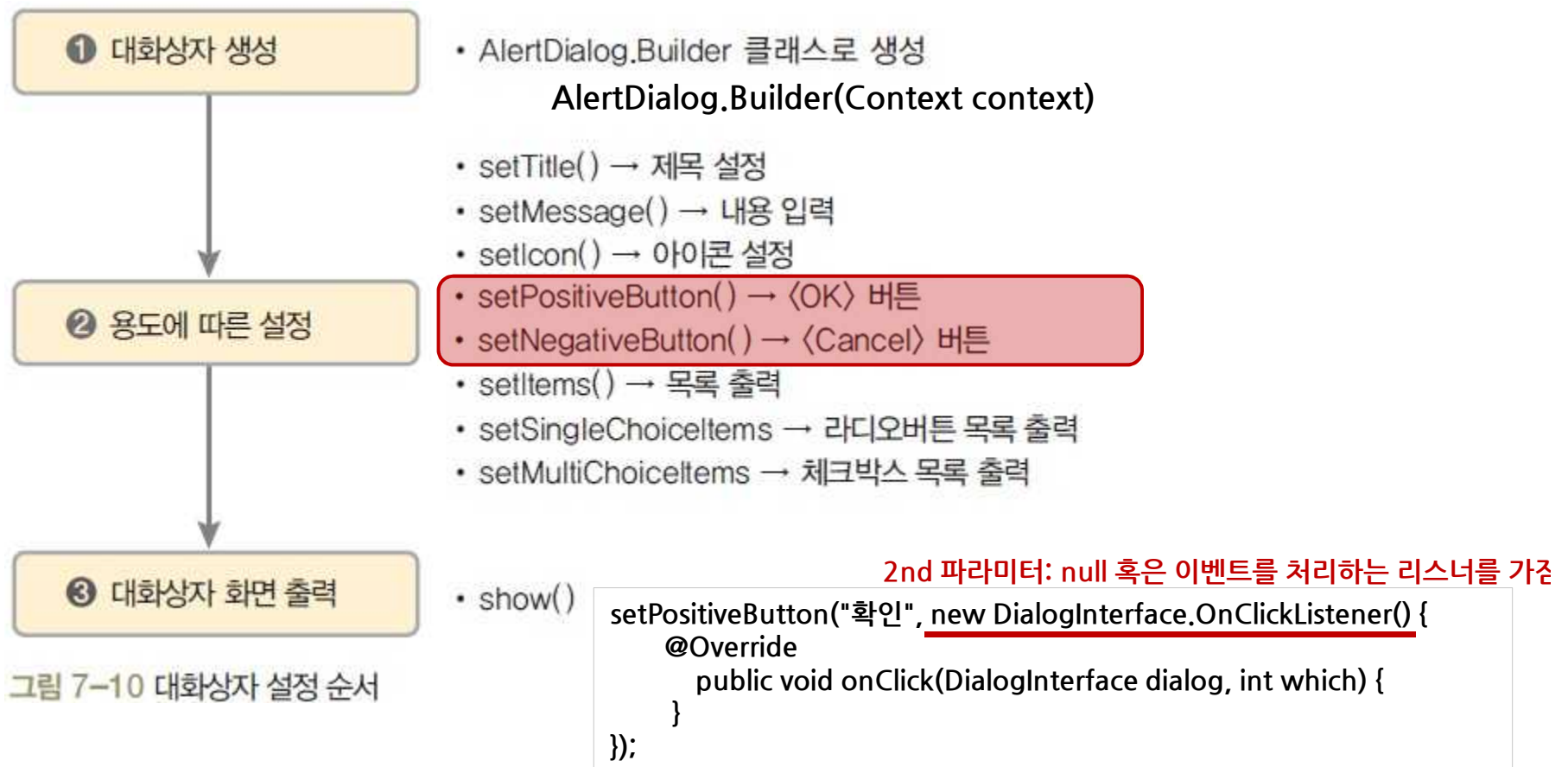
대화상자



3. 대화상자 | 기본 대화상자

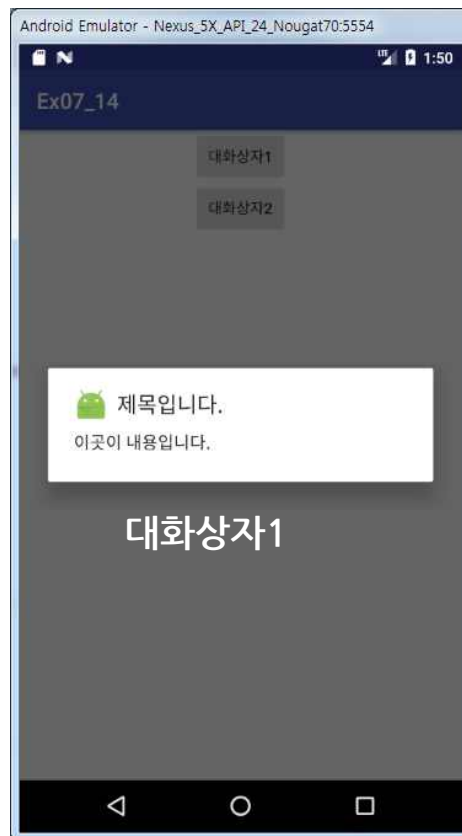
대화상자(Dialog)

사용자에게 중요한 사항을 알려주고 선택하게 함



3. 대화상자 | 기본 대화상자

예제7-14: Ex07_14
기본 대화상자 예제



3. 대화상자 | 기본 대화상자

activity_main.xml

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      android:id="@+id/activity_main"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      android:orientation="vertical"
8      android:gravity="center_horizontal"
9      tools:context="com.example.kjs.ex07_14.MainActivity">
10
11      <Button
12          android:id="@+id/button1"
13          android:layout_width="wrap_content"
14          android:layout_height="wrap_content"
15          android:text="대화상자1" />
16      <Button
17          android:id="@+id/button2"
18          android:layout_width="wrap_content"
19          android:layout_height="wrap_content"
20          android:text="대화상자2" />
21
22  </LinearLayout>
23
```





3. 대화상자 | 기본 대화상자

MainActivity.java
변수 선언

```
11 public class MainActivity extends AppCompatActivity {  
12  
13     @Override  
14     protected void onCreate(Bundle savedInstanceState) {  
15         super.onCreate(savedInstanceState);  
16         setContentView(R.layout.activity_main);  
17  
18         final Button button1 = (Button) findViewById(R.id.button1);  
19         final Button button2 = (Button) findViewById(R.id.button2);  
20     }
```


3. 대화상자 | 기본 대화상자

MainActivity.java
대화상자1 처리

```
21   
24
25
26
27 
28
29
31
```

```
button1.setOnClickListener((v) -> {
    AlertDialog.Builder dlg = new AlertDialog.Builder(MainActivity.this);
    dlg.setTitle("제목입니다.");
    dlg.setMessage("이곳이 내용입니다.");
    dlg.setIcon(R.mipmap.ic_launcher);
    dlg.show();
});
```

Builder 패턴
객체의 복잡한 생성과정을 분리하여
구현

생성자의 파라미터가 많을 때 주로 사용

3. 대화상자 | 기본 대화상자

MainActivity.java 대화상자2 처리

```
32 button2.setOnClickListener(new View.OnClickListener() {
33
34     public void onClick(View v) {
35
36         AlertDialog.Builder dlg = new AlertDialog.Builder(MainActivity.this);
37         dlg.setTitle("제목입니다");
38         dlg.setMessage("이곳에 내용입니다");
39         dlg.setIcon(R.mipmap.ic_launcher);
40         dlg.setPositiveButton("확인", new DialogInterface.OnClickListener() {
41             @Override
42             public void onClick(DialogInterface dialog, int which) {
43                 Toast.makeText(getApplicationContext(), "확인을 눌렀음", Toast.LENGTH_SHORT).show();
44             }
45         });
46         dlg.show();
47     }
48 });
49
50 }
```

3. 대화상자 | 기본 대화상자

대화상자에 다양한 형태의 목록을 출력할 때

기본 목록 형식의 데이터 출력

```
setItems(CharSequence[] items, DialogInterface.OnClickListener listener)
```

onClick메소드의 파라미터인 which로 배열 데이터의 position을 구하여 해당 위치의 항목값을 리턴

라디오 버튼으로 목록 출력

```
setSingleChoiceItems(CharSequence[] items, int checkedItem, DialogInterface.OnClickListener listener)
```

int checkedItem 기본적으로 선택된 항목

체크박스로 목록 출력

```
setMultiChoiceItems(CharSequence[] items, boolean[] checkedItems,  
DialogInterface.OnMultiChoiceClickListener listener)
```

boolean[] checkedItems 각 항목별로 기본적으로 선택된 여부

3. 대화상자 | 목록 대화상자

예제7-18: Ex07_18

목록 대화상자 예제

대화상자에 리스트 형태의 목록을 출력하여 선택한 안드로이드 버전을 버튼에 출력



3. 대화상자 | 목록 대화상자

activity_main.xml

1개 버튼 디자인

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      android:id="@+id/activity_main"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      tools:context="com.example.kjs.ex07_18.MainActivity">
8
9      <Button
10         android:text="Button"
11         android:layout_width="wrap_content"
12         android:layout_height="wrap_content"
13         android:id="@+id/button1"
14         android:layout_weight="1" />
15  </LinearLayout>
16
```


3. 대화상자 | 목록 대화상자

MainActivity.java

변수 선언 및 리스트 배열 선언

```
9 public class MainActivity extends AppCompatActivity {  
10  
11     Button button1;  
12  
13     @Override  
14     protected void onCreate(Bundle savedInstanceState) {  
15         super.onCreate(savedInstanceState);  
16         setContentView(R.layout.activity_main);  
17  
18         button1 = (Button) findViewById(R.id.button1);  
19  
20         final String[] versionArray = new String[] {"젤리빈", "킷캣", "물리팝"};  
21     }
```

3. 대화상자 | 목록 대화상자

MainActivity.java

setItems(문자배열, 리스너) 메소드를 사용해 해당 위치의 항목 리턴

```
21
22     AlertDialog.Builder dlg = new AlertDialog.Builder(MainActivity.this);
23
24     dlg.setTitle("사용하고 있는 버전은?");
25     dlg.setIcon(R.mipmap.ic_launcher);
26     dlg.setItems(versionArray, new DialogInterface.OnClickListener() {
27         @Override
28         public void onClick(DialogInterface dialog, int which) {
29             button1.setText(versionArray[which]);
30         }
31     });
32
33     dlg.setPositiveButton("닫기", null);
34     dlg.show();
35 }
36
37
```

3. 대화상자 | 목록 대화상자

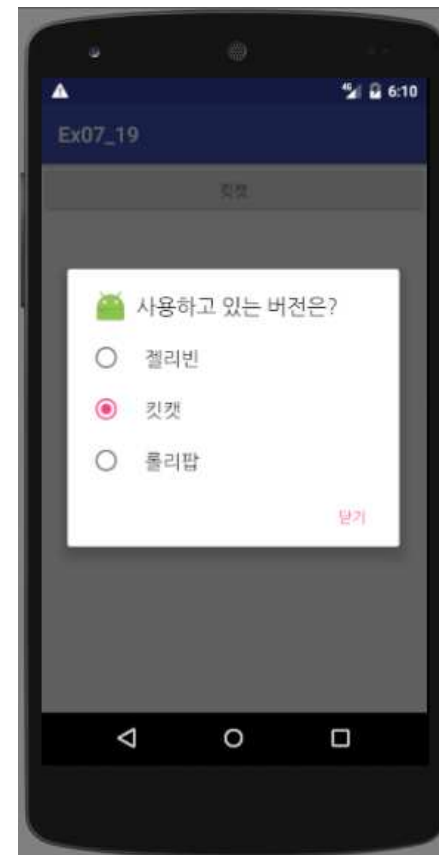
예제 7-9: Ex07_19

라디오버튼 대화상자 예제

setItems() 대신 setSingleChoiceItems()을 사용하면 라디오 버튼으로 항목 제시
선택 후 대화상자가 닫히지 않음

activity_main.xml

앞 예제랑 동일



3. 대화상자 | 목록 대화상자

MainActivity.java

앞 예제와 동일한 부분임

```
1 package com.example.kjs.ex07_19;
2
3 import android.content.DialogInterface;
4 import android.os.Bundle;
5 import android.support.v7.app.AlertDialog;
6 import android.support.v7.app.AppCompatActivity;
7 import android.widget.Button;
8
9 public class MainActivity extends AppCompatActivity {
10
11     Button button1;
12
13     @Override
14     protected void onCreate(Bundle savedInstanceState) {
15         super.onCreate(savedInstanceState);
16         setContentView(R.layout.activity_main);
17
18         button1 = (Button) findViewById(R.id.button1);
19
20         final String[] versionArray = new String[] { "젤리빈", "킷캣", "롤리팝" };
21
22         AlertDialog.Builder dlg = new AlertDialog.Builder(MainActivity.this);
23
24         dlg.setTitle("사용하고 있는 버전은?");
25         dlg.setIcon(R.mipmap.ic_launcher);
```

3. 대화상자 | 목록 대화상자

MainActivity.java

setItems() 대신 setSingleChoiceItems() 사용

```
26         dlg.setSingleChoiceItems(versionArray, 0, new DialogInterface.OnClickListener() {
27             @Override
28             public void onClick(DialogInterface dialog, int which) {
29                 button1.setText(versionArray[which]);
30             }
31         });
32
33         dlg.setPositiveButton("닫기", null);
34         dlg.show();
35     }
36 }
37
```


3. 대화상자 | 목록 대화상자

예제7-20: Ex07_20

Checkbox를 사용하기 위해 setMultiChoiceItems()를 사용
화면은 앞 예제와 동일

```
1 package com.example.kjs.ex07_20;
2
3 import android.content.DialogInterface;
4 import android.os.Bundle;
5 import android.support.v7.app.AlertDialog;
6 import android.support.v7.app.AppCompatActivity;
7 import android.widget.Button;
8
9 public class MainActivity extends AppCompatActivity {
10
11     Button button1;
12
13     @Override
14     protected void onCreate(Bundle savedInstanceState) {
15         super.onCreate(savedInstanceState);
16         setContentView(R.layout.activity_main);
17
18         button1 = (Button) findViewById(R.id.button1);
19
20         final String[] versionArray = new String[] {"젤리빈", "킷캣", "물리팝"};
21         final boolean[] checkArray = new boolean[] {true, false, false};
22
23         AlertDialog.Builder dlg = new AlertDialog.Builder(MainActivity.this);
24
25         dlg.setTitle("사용하고 있는 버전은?");
26         dlg.setIcon(R.mipmap.ic_launcher);
```

3. 대화상자 | 목록 대화상자

MainActivity.java

```
27 dig.setMultiChoiceItems(versionArray, checkArray, new DialogInterface.OnMultiChoiceClickListener() {  
28     @Override  
29     public void onClick(DialogInterface dialog, int which, boolean isChecked) {  
30         button1.setText(versionArray[which]);  
31     }  
32 });  
33  
34 dig.setPositiveButton("닫기", null);  
35 dig.show();  
36 }  
37 }  
38
```

→ 확인 버튼을 클릭하면, 선택된 항목들이 모두 버튼에 출력되도록 수정하시오!

3. 대화상자 | 목록 대화상자

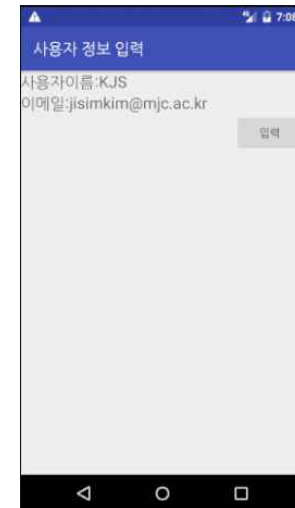
실습7-3: Project07_03

사용자정보 입력 예제

대화상자에 필요한 화면을 직접 설계하여 대화상자로 활용



확인



취소



3. 대화상자 | 목록 대화상자

화면을 직접 설계하여 대화상자로 활용하는 절차

- 1) xml 화면을 인플레이션: 뷰로 사용하기 위해서 인플레이션
`View.inflate(Context, int resource, Viewgroup)`
- 2) 생성된 대화상자에 설정
`dlg.setView(인플레이션된 뷰)`

3. 대화상자 | 목록 대화상자

activity_main.xml

화면 디자인

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     android:layout_width="match_parent"
4     android:layout_height="match_parent"
5     android:orientation="vertical" >
6     <LinearLayout
7         android:layout_width="wrap_content"
8         android:layout_height="wrap_content"
9         android:orientation="vertical">
10         <TextView
11             android:id="@+id/tvName"
12             android:layout_width="wrap_content"
13             android:layout_height="wrap_content"
14             android:text="사용자 이름:"
15             android:ems="15"
16             android:textSize="20sp" />
17
18         <TextView
19             android:id="@+id/tvEmail"
20             android:layout_width="wrap_content"
21             android:layout_height="wrap_content"
22             android:text="이메일:"
23             android:ems="15"
24             android:textSize="20sp" />
25     </LinearLayout>
26
27     <RelativeLayout
28         android:layout_width="wrap_content"
29         android:layout_height="wrap_content">
30         <Button
31             android:id="@+id/button1"
32             android:layout_width="wrap_content"
33             android:layout_height="wrap_content"
34             android:layout_alignParentRight="true"
35             android:text="입력" />
36     </RelativeLayout>
37 </LinearLayout>
```



3. 대화상자 | 목록 대화상자

activity_main.xml: dialog1.xml

대화상자용 dialog1.xml 파일을 디자인하고 뷰 객체화하여
AlertDialog로 setting하기 위해,

대화상자에서 사용할 레이아웃 XML 파일(dialog1) 생성

[res] > [layout] > [New] > [Layout resource file] > dialog1.xml

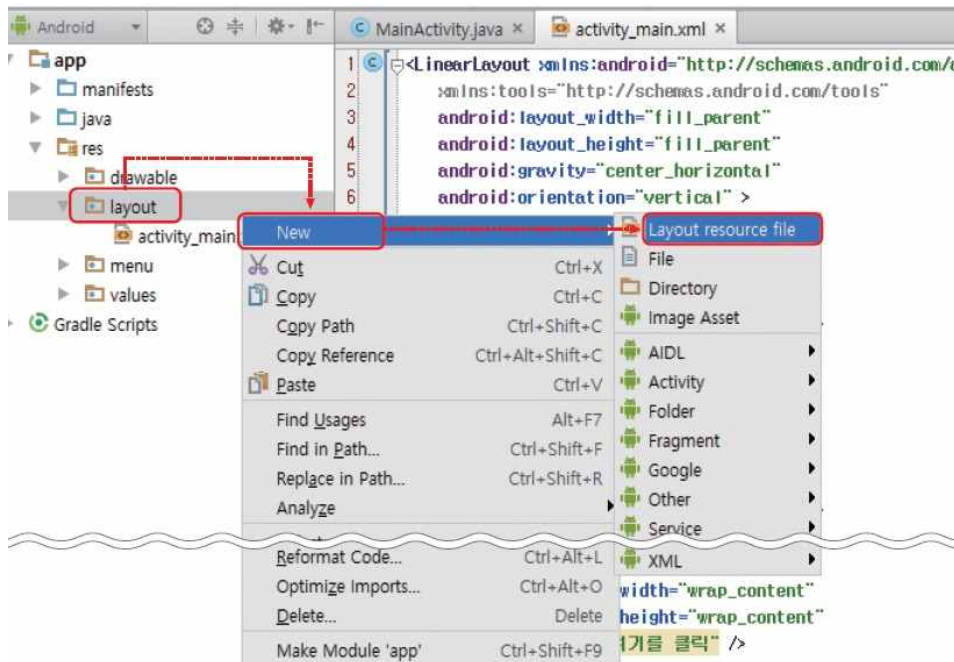


그림 7-12 레이아웃 XML 파일 생성 1

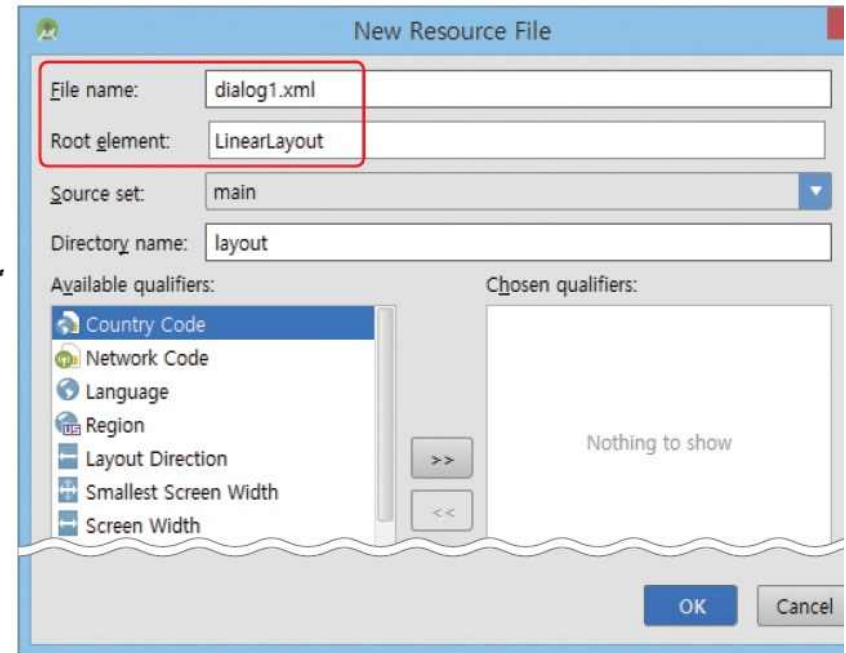
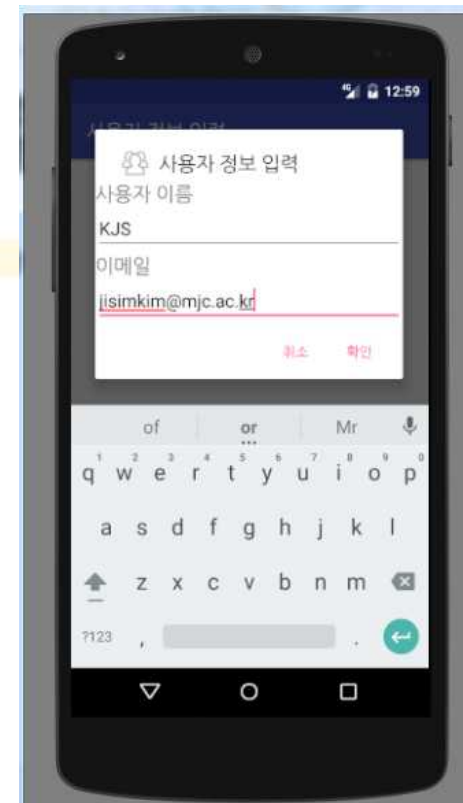


그림 7-13 레이아웃 XML 파일 생성 2

3. 대화상자 | 목록 대화상자

activity_main.xml: dialog1.xml

```
1 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
2     android:layout_width="match_parent"
3     android:layout_height="match_parent"
4     android:orientation="vertical" >
5     <TextView
6         android:layout_width="wrap_content"
7         android:layout_height="wrap_content"
8         android:text="사용자 이름"
9         android:textSize="20sp" />
10
11     <EditText
12         android:id="@+id/dlgEdt1"
13         android:layout_width="match_parent"
14         android:layout_height="wrap_content" />
15
16     <TextView
17         android:layout_width="wrap_content"
18         android:layout_height="wrap_content"
19         android:text="이메일"
20         android:textSize="20sp" />
21
22     <EditText
23         android:id="@+id/dlgEdt2"
24         android:layout_width="match_parent"
25         android:layout_height="wrap_content" />
26 </LinearLayout>
```



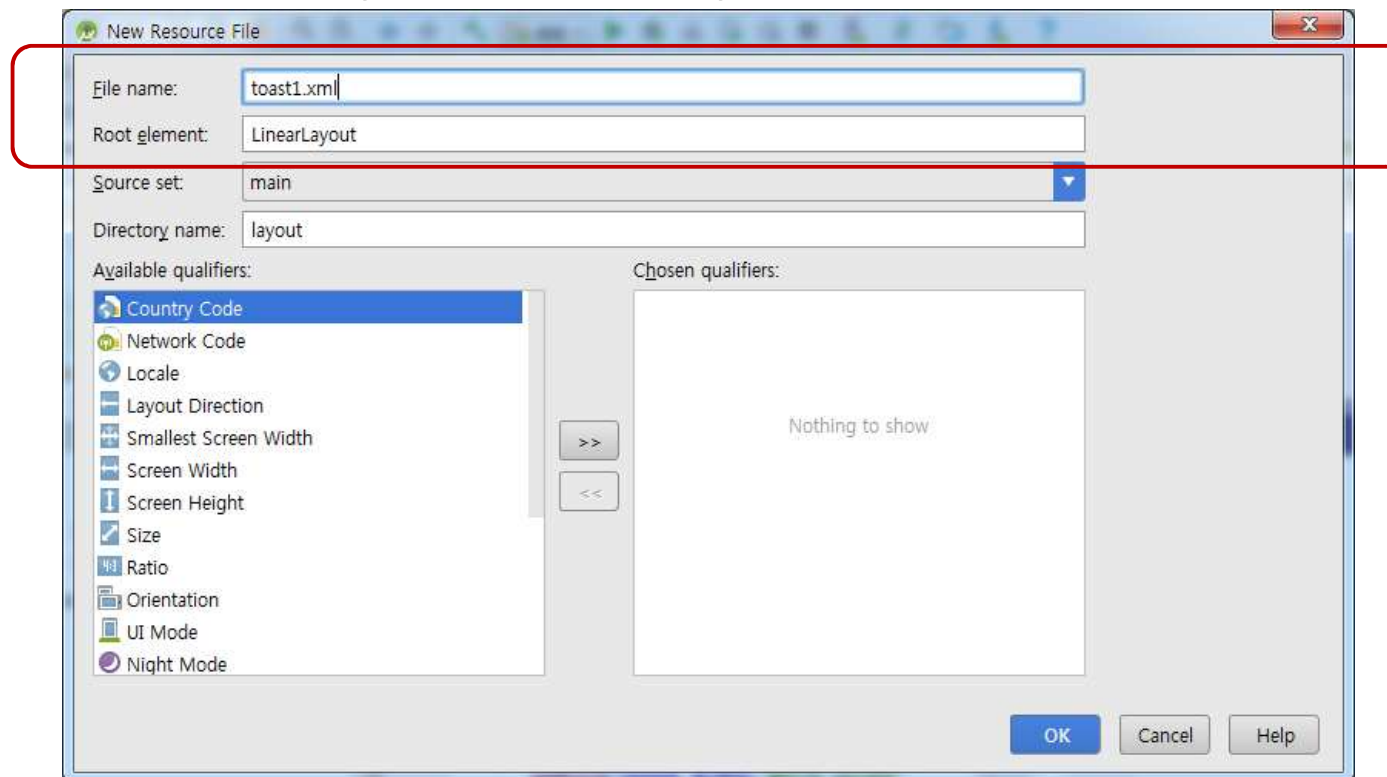
3. 대화상자 | 목록 대화상자

activity_main.xml: toast1.xml

취소시 출력할 토스트 메시지용 XML 디자인

toast1.xml 파일을 아래와 같은 방식으로 생성

[res] > [layout] > [New] > [Layout resource file] > toast1.xml



3. 대화상자 | 목록 대화상자

toast1.xml 파일을 디자인

이미지(btn_star_big_on)를 drawable에 등록 후 디자인

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     android:layout_width="match_parent"
4     android:layout_height="match_parent"
5     android:background="#ff0000"
6     android:layout_gravity="center">
7     <ImageView
8         android:layout_width="wrap_content"
9         android:layout_height="wrap_content"
10        android:id="@+id/imageView1"
11        android:src="@drawable/btn_star_big_on"/>
12
13    <TextView
14        android:layout_width="wrap_content"
15        android:layout_height="wrap_content"
16        android:id="@+id/toastText1"
17        android:text="TextView"
18        android:textSize="20sp"/>
19
20    <ImageView
21        android:layout_width="wrap_content"
22        android:layout_height="wrap_content"
23        android:id="@+id/imageView2"
24        android:src="@drawable/btn_star_big_on"/>
25 </LinearLayout>
```



3. 대화상자 | 목록 대화상자

MainActivity.java

변수 선언

```
11 public class MainActivity extends AppCompatActivity {  
12  
13     TextView tvName, tvEmail;  
14     Button button1;  
15     EditText d1gEdtName, d1gEdtEmail;  
16     TextView toastText;  
17     View dialogView, toastView;  
18  
19     dialog1.xml 파일을 뷰로 inflate하기 위한 것  
20     @Override  
21     protected void onCreate(Bundle savedInstanceState) {  
22         super.onCreate(savedInstanceState);  
23         setContentView(R.layout.activity_main);  
24  
25         setTitle("사용자 정보 입력");  
26  
27         tvName = (TextView) findViewById(R.id.tvName);  
28         tvEmail = (TextView) findViewById(R.id.tvEmail);  
29         button1 = (Button) findViewById(R.id.button1);  
30     }  
31 }
```


3. 대화상자 | 목록 대화상자

MainActivity.java

button1 변수를 클릭했을 때 대화상자 출력토록 리스너 작성

dialog1.xml을 View로서 인플레이팅하고 AlertDialog에 설정

onClick() 메소드 안에서 대화상자를 생성하고 <확인>과 <취소>를 추가
<확인>, <취소> 버튼 클릭 시 실행할 내용은 일단 null로 입력

drawable에 ic_menu_allfriends 복사

```
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51

button1.setOnClickListener(new View.OnClickListener() {

    public void onClick(View v) {
        dialogView = (View) View.inflate(MainActivity.this,
            R.layout.dialog1, null);

        AlertDialog.Builder dlg = new AlertDialog.Builder(
            MainActivity.this);
        dlg.setTitle("사용자 정보 입력");
        dlg.setIcon(R.drawable.ic_menu_allfriends);
        dlg.setView(dialogView);
        dlg.setPositiveButton("확인", null);
        dlg.setNegativeButton("취소", null);
        dlg.show();
    }
});
```

우선 null로 완성하고 다음 페이지와 같이 수정하여라!



3. 대화상자 | 목록 대화상자

MainActivity.java

확인 버튼의 리스너 수정

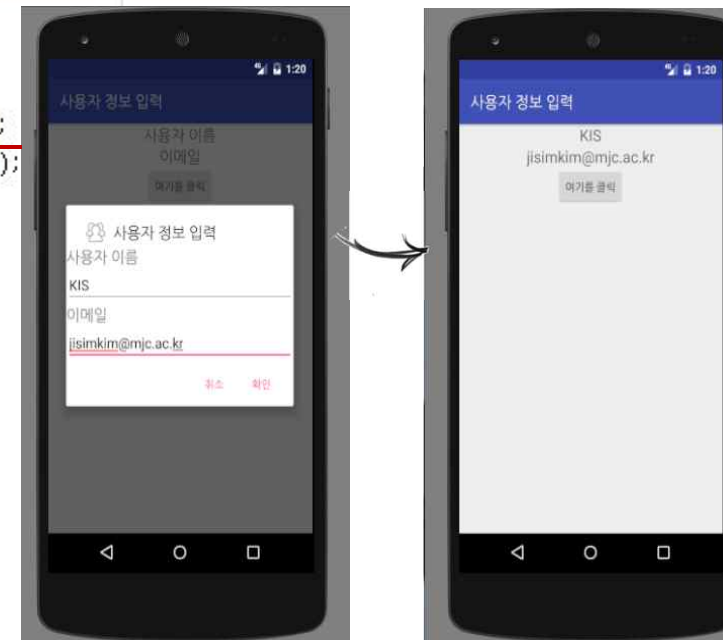
대화상자의 <확인>을 클릭하면 대화상자에서 입력한 사용자 이름과 이메일이 메인 화면(activity_main.xml)의 텍스트뷰에 출력되도록 수정

```
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63

dlg.setPositiveButton("확인", new DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialog, int which) {
        dlgEdtName = (EditText) dialogView.findViewById(R.id.dlgEdt1);
        dlgEdtEmail = (EditText) dialogView.findViewById(R.id.dlgEdt2);

        tvName.setText(dlgEdtName.getText().toString());
        tvEmail.setText(dlgEdtEmail.getText().toString());
    }
});

dlg.setNegativeButton("취소", null);
dlg.show();
}
```



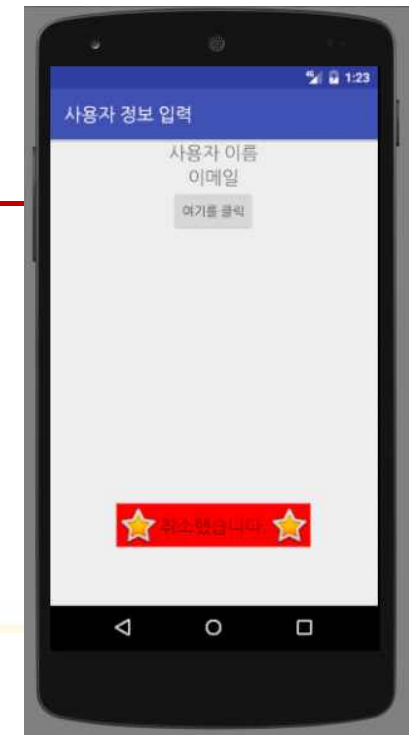
3. 대화상자 | 목록 대화상자

MainActivity.java

취소 버튼의 리스너 수정

대화상자의 <취소>를 클릭했을 때 toast1.xml이 토스트 메시지로 나오도록 수정

```
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
--  
  
dlg.setNegativeButton("취소", new DialogInterface.OnClickListener() {  
    @Override  
    public void onClick(DialogInterface dialog, int which) {  
        Toast toast = new Toast(MainActivity.this);  
        toastView = (View) View.inflate( MainActivity.this,R.layout.toast1, null);  
        toastText = (TextView) toastView.findViewById(R.id.toastText1);  
        toastText.setText("취소했습니다.");  
        toast.setView(toastView);  
        toast.show();  
    }  
});  
dlg.show();  
});  
}
```



3. 대화상자 | 목록 대화상자

실습(직접풀어보기)7-3:Test07_03

▶ 직접 풀어보기 7-3

[실습 7-3]을 다음과 같이 변경하자.

- activity_main.xml의 텍스트뷰를 에디트텍스트로 변경하자.
- <여기를 클릭>을 클릭하면 activity_main.xml의 에디트텍스트 내용이 대화상자의 에디트텍스트에 나타나게 한다.
- 대화상자에서 <확인>을 클릭하면 대화상자의 에디트텍스트 내용이 activity_main.xml의 에디트텍스트 내용으로 변경되게 한다.



3. 대화상자 | 목록 대화상자

실습예제: MyProgressbar



```
3  import android.app.ProgressDialog;
4  import android.os.Bundle;
5  import android.support.v7.app.AppCompatActivity;
6  import android.view.View;
7  import android.widget.Button;
8
9  public class MainActivity extends AppCompatActivity {
10      ProgressDialog dialog;
11
12      @Override
13      protected void onCreate(Bundle savedInstanceState) {
14          super.onCreate(savedInstanceState);
15          setContentView(R.layout.activity_main);
16
17          Button button = (Button) findViewById(R.id.button);
18          button.setOnClickListener(new View.OnClickListener() {
19              @Override
20              public void onClick(View view) {
21                  dialog = new ProgressDialog(MainActivity.this);
22                  dialog.setProgressStyle(ProgressDialog.STYLE_SPINNER);
23                  dialog.setMessage("데이터를 확인하는 중입니다.");
24
25                  dialog.show();
26              }
27          });
28      }
29  }
```