

메뉴와 대화상자

Instructor: Park, JoonSeok (pjs50@pusan.ac.kr)

메뉴

- ▶ 메뉴는 사용자에게 여러 개 중 하나를 선택할 수 있는 방식을 제공
- ▶ 옵션 메뉴(option menu) 와 컨텍스트 메뉴(context menu)로 구분
- ▶ 메뉴를 사용하는 방법은 메뉴 XML 파일을 생성한 후 Java에서 호출하는 방법과, XML 파일 없이 Java 코드만으로 메뉴를 생성하는 방법이 있음



메뉴 - 옵션 메뉴

▶ XML을 이용한 방법

① 메뉴 폴더 생성 및 메뉴 XML 파일 생성 · 편집

메뉴 코딩



② Java 코딩:
onCreateOptionsMenu() 메소드 오버라이딩

메뉴 파일 등록



③ Java 코딩:
onOptionsItemSelected() 메소드 오버라이딩

메뉴 선택 시 작동할 내용 코딩



메뉴 – 옵션 메뉴

▶ XML 파일 형식

```
<menu>
    <item
        android:id="@+id/항목1아이디"
        android:title="항목1 제목"/>
    <item
        android:id="@+id/항목2아이디"
        android:title="항목2 제목"/>
</menu>
```

메뉴 - 옵션 메뉴

▶ onCreateOptionsMenu()

```
public boolean onCreateOptionsMenu(Menu menu) {  
    super.onCreateOptionsMenu(menu);  
    MenuInflater mInflater = getMenuInflater();  
    mInflater.inflate(R.menu.메뉴XML아이디, menu);  
    return true;  
}
```

인플레이터(Inflater)는 풍선에 바람을 넣어서 실제 객체로 만들어 사용한다는 의미로, 메뉴 인플레이터(MenuInflater) 개체는 메뉴 XML 파일을 Java 코드에서 가져와서 사용하는 것이다.



메뉴 - 옵션 메뉴

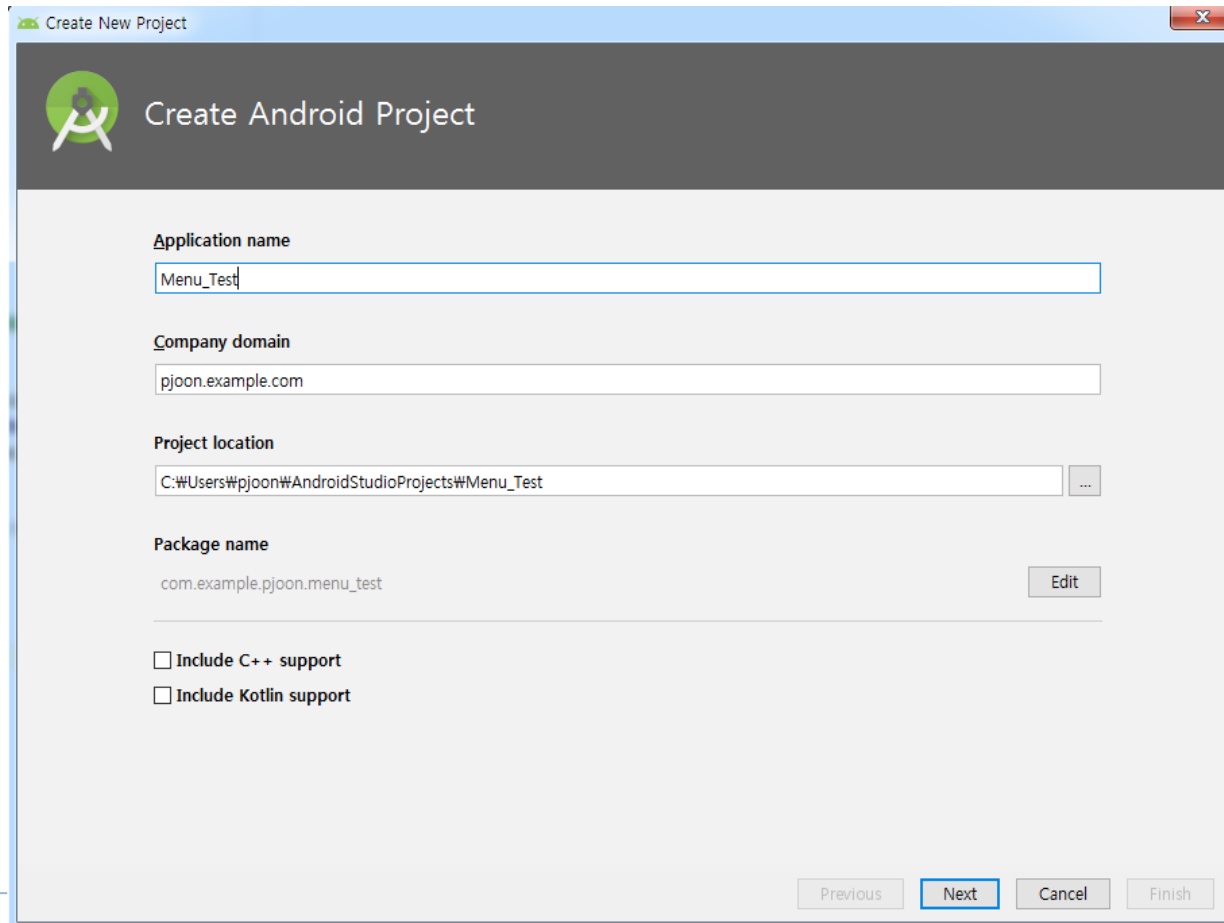
▶ onOptionsItemSelected()

```
public boolean onOptionsItemSelected(MenuItem item) {  
    switch (item.getItemId()) {  
        case R.id.항목1아이디:  
            항목1을 선택했을 때 실행할 코드  
            return true;  
        case R.id.항목2아이디:  
            항목2를 선택했을 때 실행할 코드  
            return true;  
    }  
    return false;  
}
```



메뉴 – 옵션 메뉴

- ▶ 옵션 메뉴를 만들고, 옵션 메뉴를 클릭하면 레이아웃의 배경색이 바뀌는 앱



Create New Project

Create Android Project

Application name
Menu_Test

Company domain
pjoon.example.com

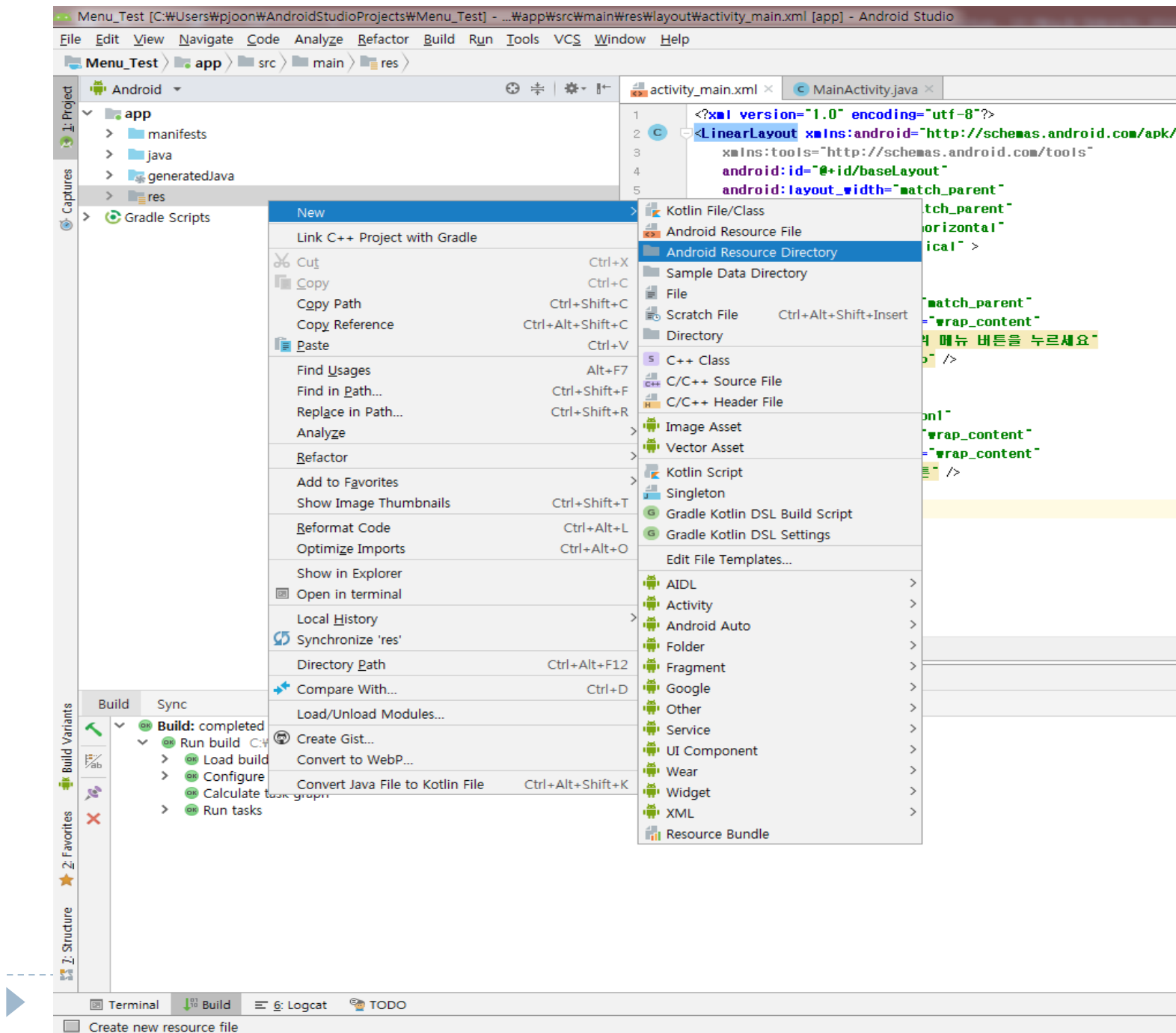
Project location
C:\Users\pjoon\AndroidStudioProjects\Menu_Test

Package name
com.example.pjoon.menu_test

☐ Include C++ support
☐ Include Kotlin support

Previous Next Cancel Finish

```
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/baseLayout"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   android:gravity="center_horizontal"
8   android:orientation="vertical" >
9
10   <TextView
11     android:layout_width="match_parent"
12     android:layout_height="wrap_content"
13     android:text="오른쪽 위 메뉴 버튼을 누르세요"
14     android:textSize="20dp" />
15
16   <Button
17     android:id="@+id/button1"
18     android:layout_width="wrap_content"
19     android:layout_height="wrap_content"
20     android:text="이건 버튼" />
21
22 </LinearLayout>
```

Project
Captures

Android
app
manifests
java
generatedJava
res
drawable
layout
mipmap
values
Gradle Scripts

activity_main.xml x MainActivity.java x
1 <?xml version="1.0" encoding="utf-8"?>

New Resource Directory

Directory name: menu

Resource type: menu

Source set: main

Available qualifiers:
Country Code
Network Code
Locale
Layout Direction
Smallest Screen Width
Screen Width
Screen Height
Size
Ratio
Orientation
UI Mode
Night Mode
Density
Touch Screen

Chosen qualifiers:
Nothing to show

OK Cancel

Android

app
manifests
java
generatedJava
res
drawable
layout
menu
mipmap
values
Gradle Scripts

Menu_Test [C:\Users\pjoon\AndroidStudioProjects\Menu_Test] - ...App\src\main\res\layout\activity_main.xml [app] - Android Studio

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

Menu_Test > app > src > main > res > menu >

Android

Project

- app
 - manifests
 - java
 - generatedJava
 - res
 - drawable
 - layout
 - menu
 - mipmap
 - values
 - Gradle Scripts

Captures

activity_main.xml x MainActivity.java x

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android..
3   xmlns:tools="http://schemas.android.com/tools"
4   android:id="@+id/baseLayout"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   android:gravity="center_horizontal"
8   android:orientation="vertical" >

```

New

- Link C++ Project with Gradle
- Cut Ctrl+X
- Copy Ctrl+C
- Copy Path Ctrl+Shift+C
- Copy Reference Ctrl+Alt+Shift+C
- Paste Ctrl+V
- Find Usages Alt+F7
- Find in Path... Ctrl+Shift+F
- Replace in Path... Ctrl+Shift+R
- Analyze
- Refactor
- Add to Favorites
- Show Image Thumbnails Ctrl+Shift+T
- Reformat Code Ctrl+Alt+L
- Optimize Imports Ctrl+Alt+O
- Delete... Delete
- Run 'Tests in 'menu'' Ctrl+Shift+F10
- Debug 'Tests in 'menu''
- Run 'Tests in 'menu'' with Coverage
- Create 'Tests in 'menu''...
- Show in Explorer
- Open in terminal
- Local History
- Synchronize 'menu'
- Directory Path Ctrl+Alt+F12
- Compare With... Ctrl+D
- Load/Unload Modules...
- Create Gist...
- Convert Java File to Kotlin File Ctrl+Alt+Shift+K

Kotlin File/Class

- Menu resource file
- Sample Data Directory
- File
- Scratch File Ctrl+Alt+Shift+Insert
- Directory
- C++ Class
- C/C++ Source File
- C/C++ Header File
- Image Asset
- Vector Asset
- Kotlin Script
- Singleton
- Gradle Kotlin DSL Build Script
- Gradle Kotlin DSL Settings
- Edit File Templates...
- AIDL
- Activity
- Android Auto
- Folder
- Fragment
- Google
- Other
- Service
- UI Component
- Wear
- Widget
- XML
- Resource Bundle

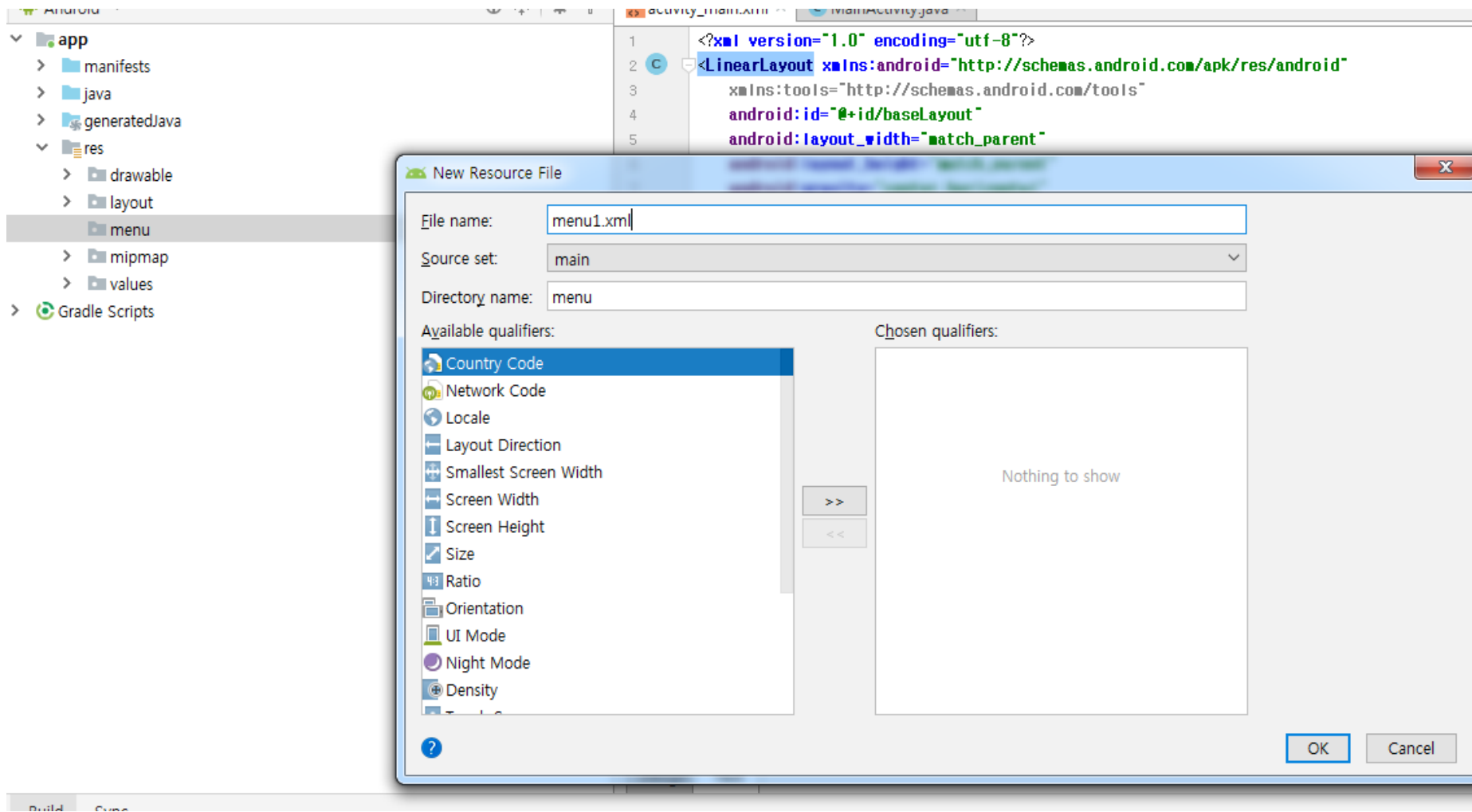
Build Sync

Build: completed successfully at 2019

- Run build C:\Users\pjoon\Android
- Load build
- Configure build
- Calculate task graph
- Run tasks

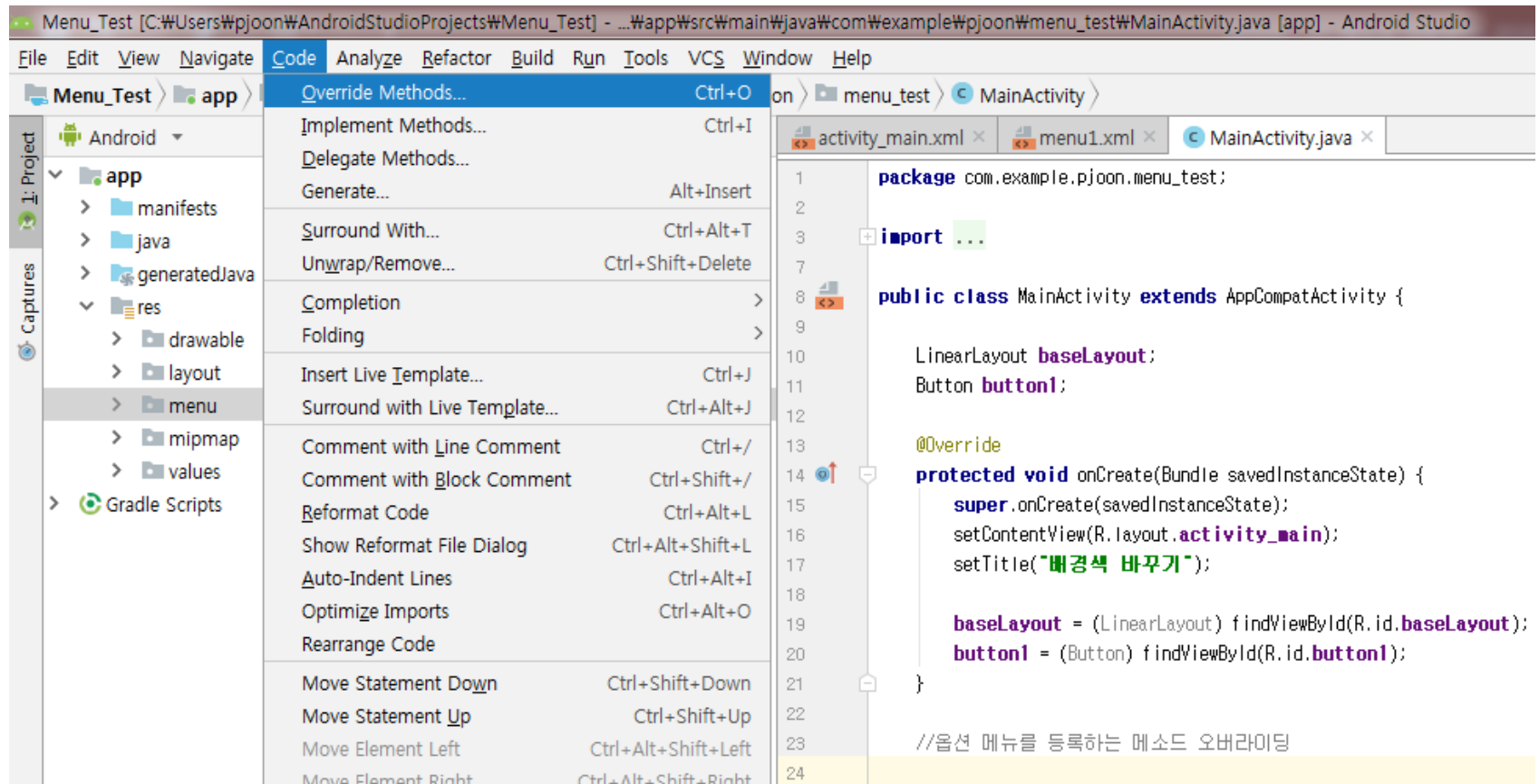
Terminal Build Logcat TODO

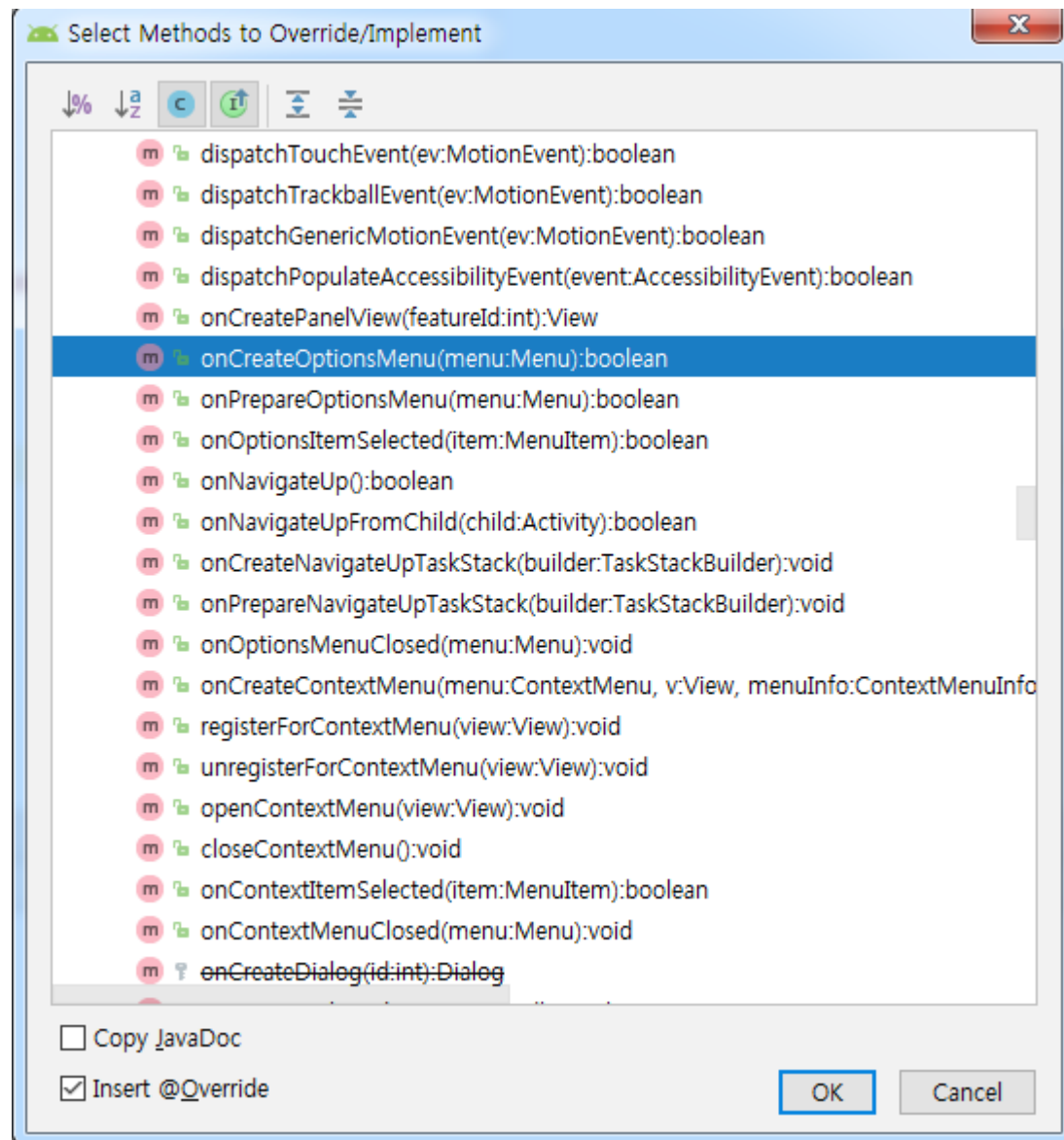
Create a new Menu resource file



```
1 <?xml version="1.0" encoding="utf-8"?>
2 <menu xmlns:android="http://schemas.android.com/apk/res/android">
3     <item
4         android:id="@+id/itemRed"
5         android:title="배경색 (빨강)">
6     </item>
7     <item
8         android:id="@+id/itemGreen"
9         android:title="배경색 (초록)">
10    </item>
11    <item
12        android:id="@+id/itemBlue"
13        android:title="배경색 (파랑)">
14    </item>
15    <item android:title="버튼 변경 >> ">
16        <menu>
17            <item
18                android:id="@+id/subRotate"
19                android:title="버튼 45도 회전"/>
20            <item
21                android:id="@+id/subSize"
22                android:title="버튼 2배 확대"/>
23        </menu>
24    </item>
25
26 </menu>
```

```
1 package com.example.pjoon.menu_test;
2
3 import ...
4
5
6
7
8 public class MainActivity extends AppCompatActivity {
9
10     LinearLayout baseLayout;
11     Button button1;
12
13     @Override
14     protected void onCreate(Bundle savedInstanceState) {
15         super.onCreate(savedInstanceState);
16         setContentView(R.layout.activity_main);
17         setTitle("배경색 바꾸기");
18
19         baseLayout = (LinearLayout) findViewById(R.id.baseLayout);
20         button1 = (Button) findViewById(R.id.button1);
21     }
22 }
23
```





24
25
26
27
28
29
30
31
32
33
34
35

//옵션 메뉴를 등록하는 메소드 오버라이딩

@Override

```
public boolean onCreateOptionsMenu(Menu menu) {  
    super.onCreateOptionsMenu(menu);  
    MenuInflater inflater = getMenuInflater();  
    inflater.inflate(R.menu.menu1, menu);  
    return true;  
}
```



```

17     super.onCreate(savedInstanceState);
18     setContentView(R.layout.activity_main);
19     setTitle("배경색 바꾸기");
20
21     baselayout = (LinearLayout) findViewById(R.id.
22     button1 = (Button) findViewById(R.id.button1);
23 }
24
25 //옵션 메뉴를 등록하는 메소드 오버라이딩
26
27
28 @Override
29 public boolean onCreateOptionsMenu(Menu menu) {
30     super.onCreateOptionsMenu(menu);
31     MenuInflater mInflater = getMenuInflater();
32     mInflater.inflate(R.menu.menu1, menu);
33     return true;
34 }
35
36
37 //메뉴를 클릭했을때 동작하는 메소드 오버라이딩
38
39
40 }
41

```

MainActivity

Select Methods to Override/Implement

- ☐ onDetachedFromWindow():void
- ☐ hasWindowFocus():boolean
- ☐ dispatchTouchEvent(ev:MotionEvent):boolean
- ☐ dispatchTrackballEvent(ev:MotionEvent):boolean
- ☐ dispatchGenericMotionEvent(ev:MotionEvent):boolean
- ☐ dispatchPopulateAccessibilityEvent(event:AccessibilityEvent):boolean
- ☐ onCreatePanelView(featureId:int):View
- ☐ onPrepareOptionsMenu(menu:Menu):boolean
- ☒ onOptionsItemSelected(item:MenuItem):boolean
- ☐ onNavigateUp():boolean
- ☐ onNavigateUpFromChild(child:Activity):boolean
- ☐ onCreateNavigateUpTaskStack(builder:TaskStackBuilder):void
- ☐ onPrepareNavigateUpTaskStack(builder:TaskStackBuilder):void
- ☐ onOptionsItemSelected(menu:Menu):void
- ☐ onCreateContextMenu(menu:ContextMenu, v:View, menuInfo:ContextMenuInfo)
- ☐ registerForContextMenu(view:View):void
- ☐ unregisterForContextMenu(view:View):void
- ☐ openContextMenu(view:View):void
- ☐ closeContextMenu():void
- ☐ onContextItemSelected(item:MenuItem):boolean
- ☐ onContextMenuClosed(menu:Menu):void
- ☐ onCreateDialog(id:int):Dialog

☐ Copy JavaDoc

☒ Insert @Override

OK

Cancel

//메뉴를 클릭했을때 동작하는 메소드 오버라이딩

@Override

```
public boolean onOptionsItemSelected(MenuItem item) {  
    switch (item.getItemId()) {  
        case R.id.itemRed:  
            baseLayout.setBackgroundColor(Color.RED);  
            return true;  
        case R.id.itemGreen:  
            baseLayout.setBackgroundColor(Color.GREEN);  
            return true;  
        case R.id.itemBlue:  
            baseLayout.setBackgroundColor(Color.BLUE);  
            return true;  
        case R.id.subRotate:  
            button1.setRotation(45);  
            return true;  
        case R.id.subSize:  
            button1.setScaleX(2);  
            return true;  
    }  
    return false;  
}
```



메뉴 - Java 코드를 이용한 메뉴 생성 예제

- ▶ onCreateOptionsMenu() 메소드 안에서 메뉴 XML 파일에 접근하는 대신에 직접 Menu.add() 메소드로 메뉴 항목을 추가

```
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    super.onCreateOptionsMenu(menu);

    menu.add(0, 1, 0, "배경색 (빨간)");
    menu.add(0, 2, 0, "배경색 (초록)");
    menu.add(0, 3, 0, "배경색 (파란)");

    SubMenu sMenu = menu.addSubMenu("버튼 변경 >>");
    sMenu.add(0, 4, 0, "버튼 45도 회전");
    sMenu.add(0, 5, 0, "버튼 2배 확대");

    return true;
}
```



@Override

```
public boolean onCreateOptionsMenu(Menu menu) {  
    super.onCreateOptionsMenu(menu);  
    //MenuInflater inflater = getMenuInflater();  
    //inflater.inflate(R.menu.menu1, menu);  
    /* JAVA 코드만으로 메뉴 구성시*/  
    menu.add( groupId: 0, itemId: 1, order: 0, title: "배경색 (빨강)");  
    menu.add( groupId: 0, itemId: 2, order: 0, title: "배경색 (초록)");  
    menu.add( groupId: 0, itemId: 3, order: 0, title: "배경색 (파랑)");  
  
    SubMenu sMenu = menu.addSubMenu( title: "버튼 변경 >>");  
    sMenu.add( groupId: 0, itemId: 4, order: 0, title: "버튼 45도 회전");  
    sMenu.add( groupId: 0, itemId: 5, order: 0, title: "버튼 2배 확대");  
  
    return true;  
}
```

```

public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
        /*
        case R.id.itemRed:
            baseLayout.setBackgroundColor(Color.RED);
            return true;
        case R.id.itemGreen:
            baseLayout.setBackgroundColor(Color.GREEN);
            return true;
        case R.id.itemBlue:
            baseLayout.setBackgroundColor(Color.BLUE);
            return true;
        case R.id.subRotate:
            button1.setRotation(45);
            return true;
        case R.id.subSize:
            button1.setScaleX(2);
            return true;
        */

        /*java 코드로 구성시*/
        case 1:
            baseLayout.setBackgroundColor(Color.RED);
            return true;
        case 2:
            baseLayout.setBackgroundColor(Color.GREEN);
            return true;
        case 3:
            baseLayout.setBackgroundColor(Color.BLUE);
            return true;
        case 4:
            button1.setRotation(45);
            return true;
        case 5:
            button1.setScaleX(2);
            return true;
    }
    return false;
}

```



메뉴 – 컨텍스트 메뉴

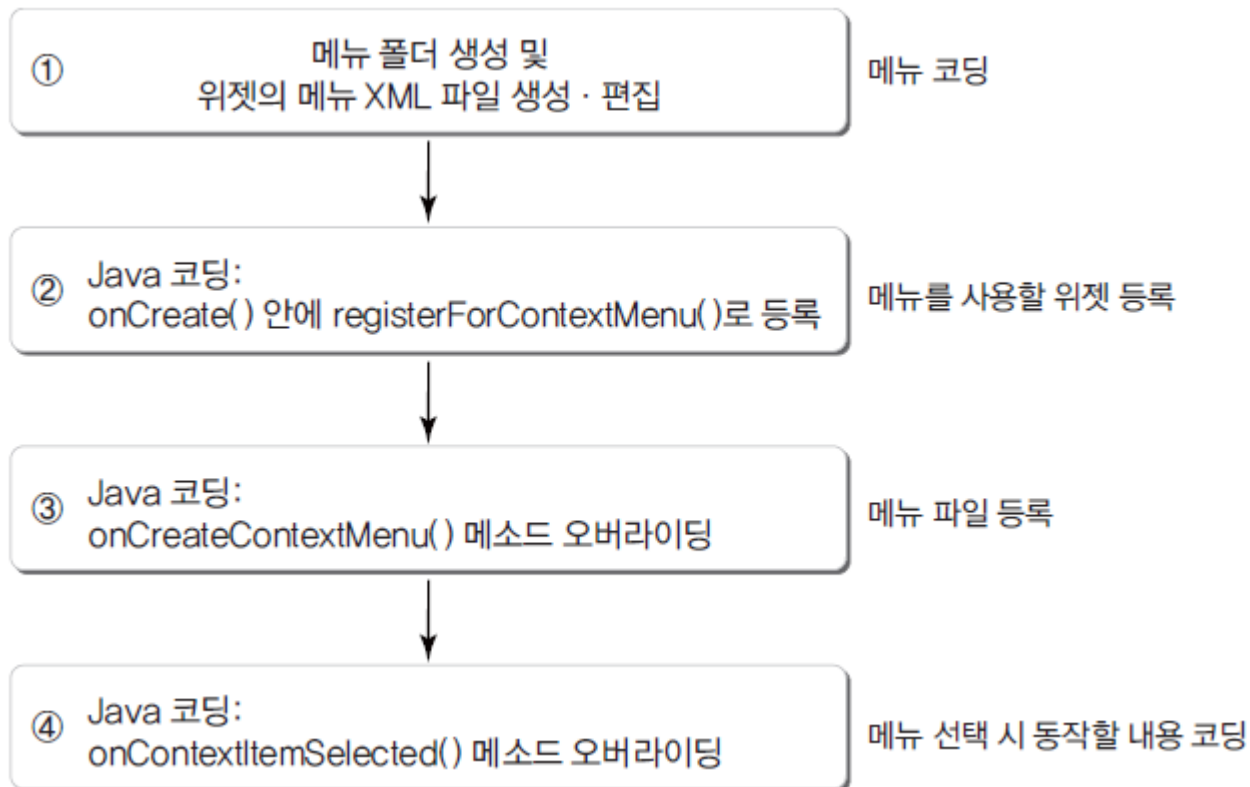
▶ 컨텍스트 메뉴

- ▶ 컨텍스트 메뉴는 위젯을 롱클릭하면 나타남
- ▶ 옵션 메뉴는 1개만 사용 가능했으나, 컨텍스트 메뉴는 각 위젯별로 메뉴를 등록할 수 있으므로 여러 개 사용가능
- ▶ 화면 중앙에 나타나면 Windows의 팝업창과 비슷함



메뉴 – 컨텍스트 메뉴

▶ 설정 순서



메뉴 – 컨텍스트 메뉴

- ▶ 컨텍스트 메뉴 XML 파일
 - ▶ 메뉴 XML 파일은 컨텍스트 메뉴가 나올 위젯마다 별도의 XML 파일로 만들어야 함
 - ▶ 메뉴 XML 문법은 옵션 메뉴와 동일함



메뉴 – 컨텍스트 메뉴

▶ CreateContextMenu()

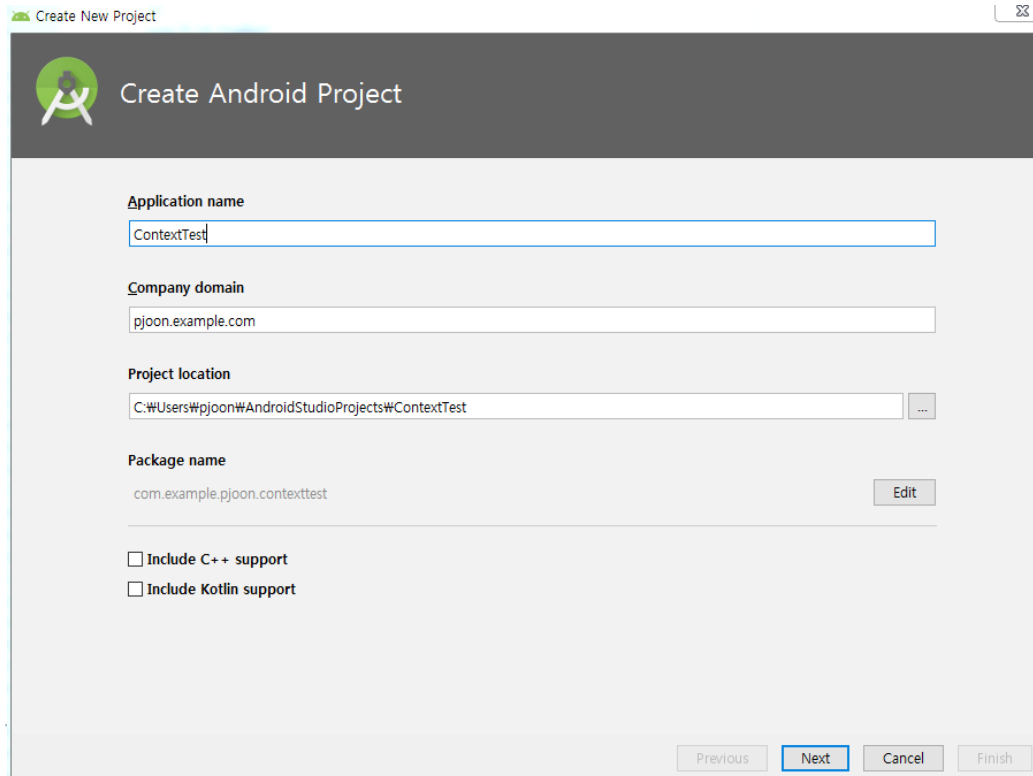
- ▶ 위젯별 컨텍스트 메뉴를 메소드에 if문으로 등록

```
public void onCreateContextMenu(ContextMenu menu,
    View v, ContextMenuInfo menuInfo) {
    super.onCreateContextMenu(menu, v, menuInfo);

    MenuInflater mInflater = getMenuInflater();
    if(v == 위젯1) {
        mInflater.inflate(R.menu.첫번째메뉴XML파일, menu);
    }
    if (v == 위젯2) {
        mInflater.inflate(R.menu.두번째메뉴XML파일, menu);
    }
}
```

메뉴 – 컨텍스트 메뉴

- ▶ <배경색 변경>버튼을 롱클릭하면 배경색을 변경하는 세가지 컨텍스트 메뉴가 나오고, <버튼 변경>버튼을 롱클릭하면 버튼의 회전과 크기를 변경하는 두가지 컨텍스트 메뉴가 나오는 앱



Create New Project

Create Android Project

Application name
ContextTest

Company domain
pjoon.example.com

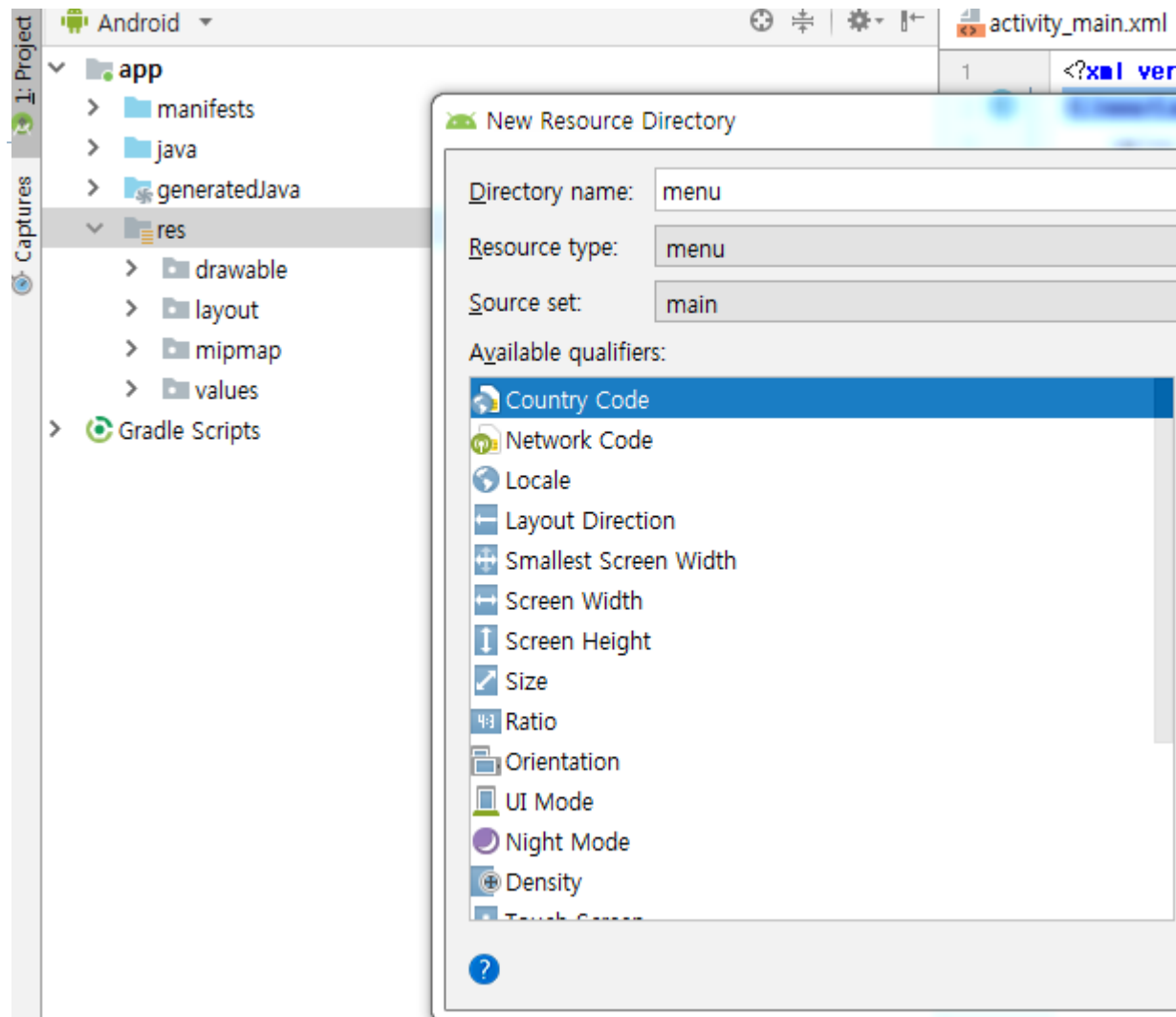
Project location
C:\Users\pjoon\AndroidStudioProjects\ContextTest

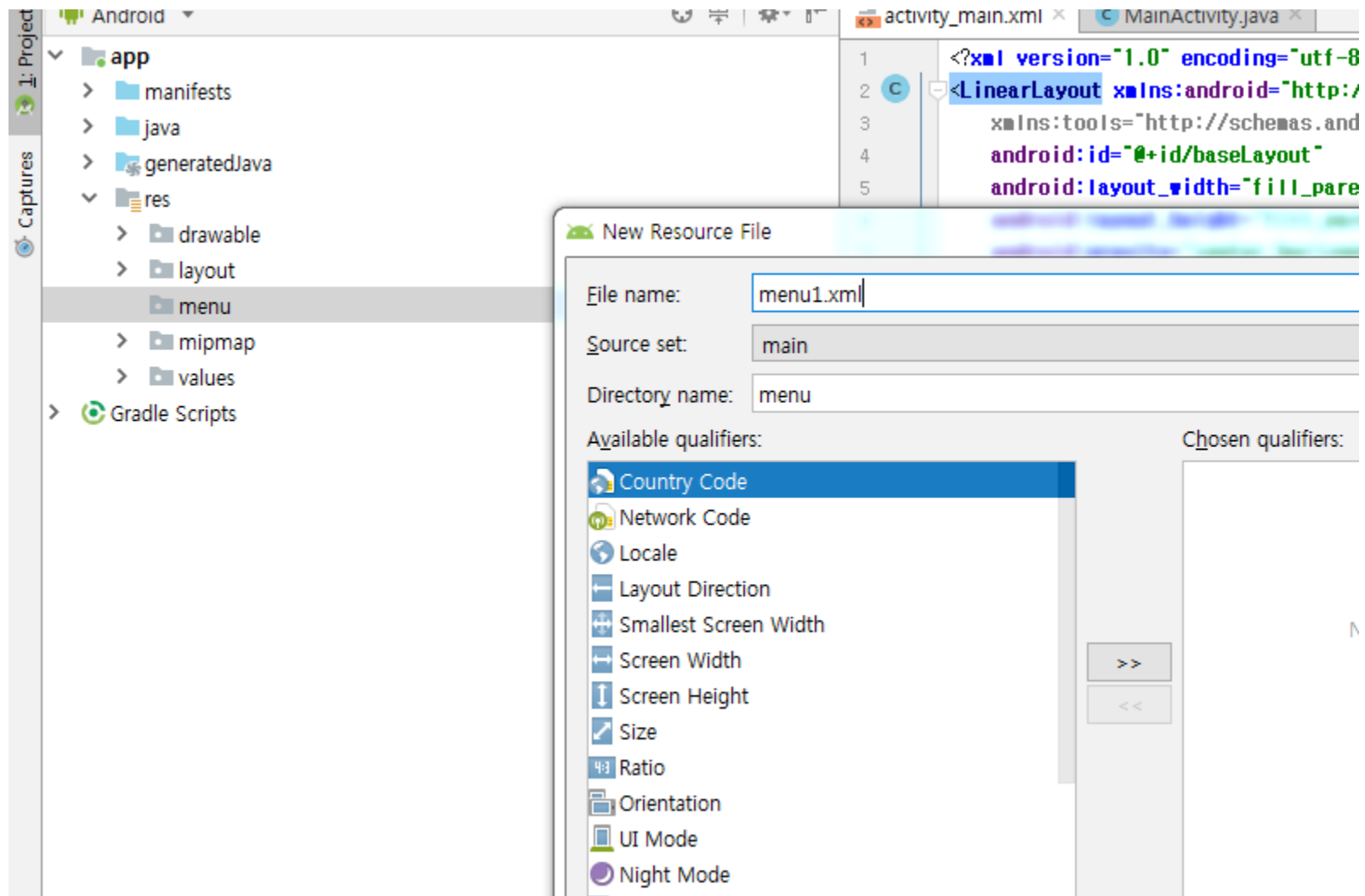
Package name
com.example.pjoon.contexttest

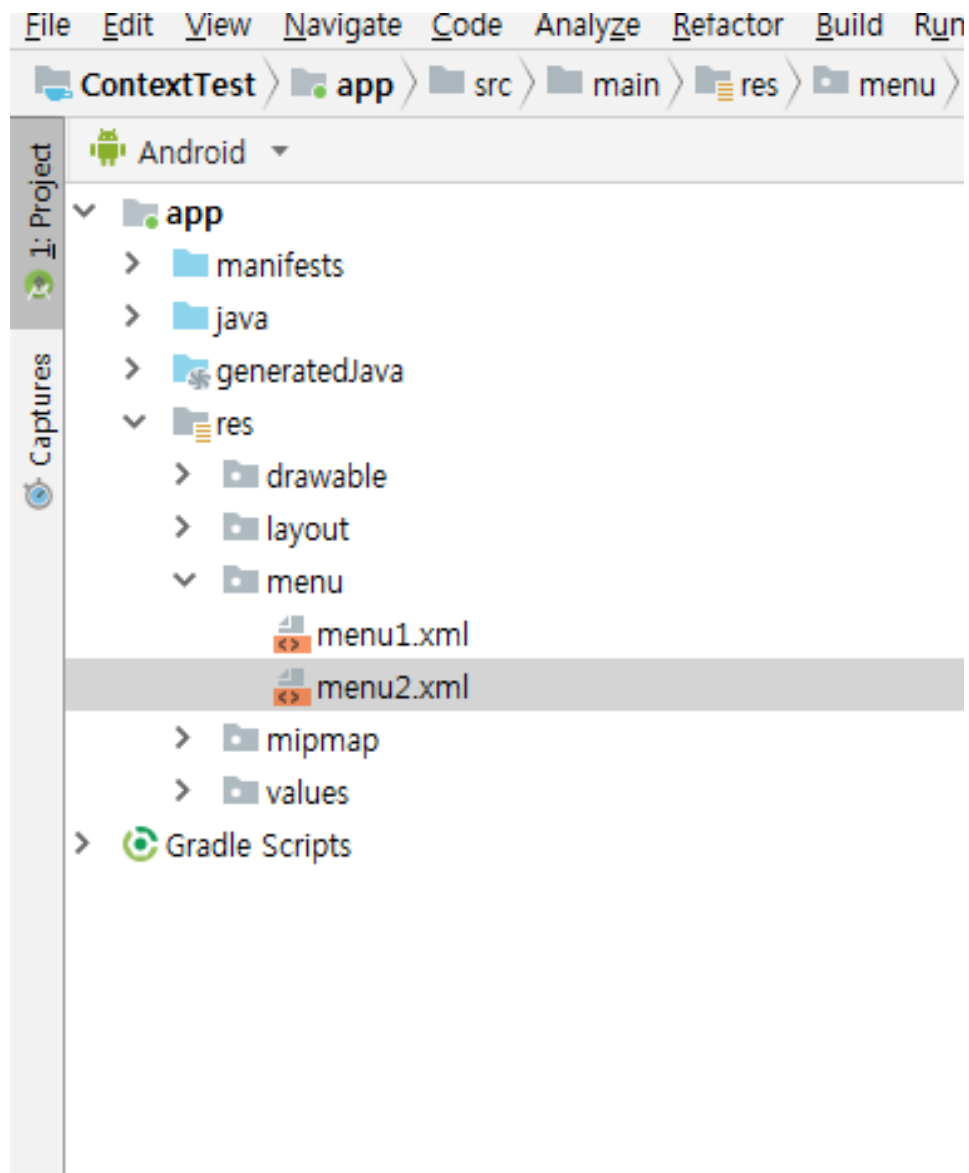
☐ Include C++ support
☐ Include Kotlin support

Previous Next Cancel Finish

```
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/baseLayout"
5   android:layout_width="fill_parent"
6   android:layout_height="fill_parent"
7   android:gravity="center_horizontal"
8   android:orientation="vertical" >
9
10  <Button
11    android:id="@+id/button1"
12    android:layout_width="wrap_content"
13    android:layout_height="wrap_content"
14    android:text="배경색 변경" />
15
16  <Button
17    android:id="@+id/button2"
18    android:layout_width="wrap_content"
19    android:layout_height="wrap_content"
20    android:text="버튼 변경" />
21
22 </LinearLayout>
```











▶ menu.xml

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <menu xmlns:android="http://schemas.android.com/apk/res/android">
3
4      <item
5          android:id="@+id/itemRed"
6          android:title="배경색 (빨강)">
7      </item>
8      <item
9          android:id="@+id/itemGreen"
10         android:title="배경색(초록)">
11     </item>
12     <item
13         android:id="@+id/itemBlue"
14         android:title="배경색(파랑)">
15     </item>
16
17 </menu>
```

▶ menu2.xml

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <menu xmlns:android="http://schemas.android.com/apk/res/android">
3
4      <item
5          android:id="@+id/subRotate"
6          android:title="버튼 45도 회전"/>
7
8      <item
9          android:id="@+id/subSize"
10         android:title="버튼 2배 확대"/>
11 </menu>
```

```
7
8  public class MainActivity extends AppCompatActivity {
9
10     LinearLayout baseLayout;
11     Button button1, button2;
12
13     @Override
14   protected void onCreate(Bundle savedInstanceState) {
15         super.onCreate(savedInstanceState);
16         setContentView(R.layout.activity_main);
17
18         baseLayout = (LinearLayout) findViewById(R.id.baseLayout);
19         button1 = (Button) findViewById(R.id.button1);
20         registerForContextMenu(button1);
21
22         button2 = (Button) findViewById(R.id.button2);
23         registerForContextMenu(button2);
24      }
25 }
26
```



Select Methods to Override



- m onActionModeFinished(mode:ActionMode):void
- m onActionModeStarted(mode:ActionMode):void
- m onActivityReenter(resultCode:int, data:Intent):void
- m onApplyThemeResource(theme:Theme, resid:int, first:boolean):void
- m onAttachedToWindow():void
- m onAttachFragment(fragment:Fragment):void
- m onChildTitleChanged(childActivity:Activity, title:CharSequence):void
- m onContextItemSelected(item:MenuItem):boolean
- m onContextMenuClosed(menu:Menu):void
- m onCreate(savedInstanceState:Bundle, persistentState:PersistableBundle):void
- m onCreateContextMenu(menu:ContextMenu, v:View, menuInfo:ContextMenuInfo):void
- m onCreateDescription():CharSequence
- m onCreateDialog(id:int):Dialog
- m onCreateDialog(id:int, args:Bundle):Dialog
- m onCreateNavigateUpTaskStack(builder:TaskStackBuilder):void
- m onCreateOptionsMenu(menu:Menu):boolean
- m onCreatePanelView(featureId:int):View
- m onCreateThumbnail(outBitmap:Bitmap, canvas:Canvas):boolean
- m onDetachedFromWindow():void
- m onEnterAnimationComplete():void
- m onGenericMotionEvent(event:MotionEvent):boolean

☐ Copy Javadoc

☒ Insert @Override

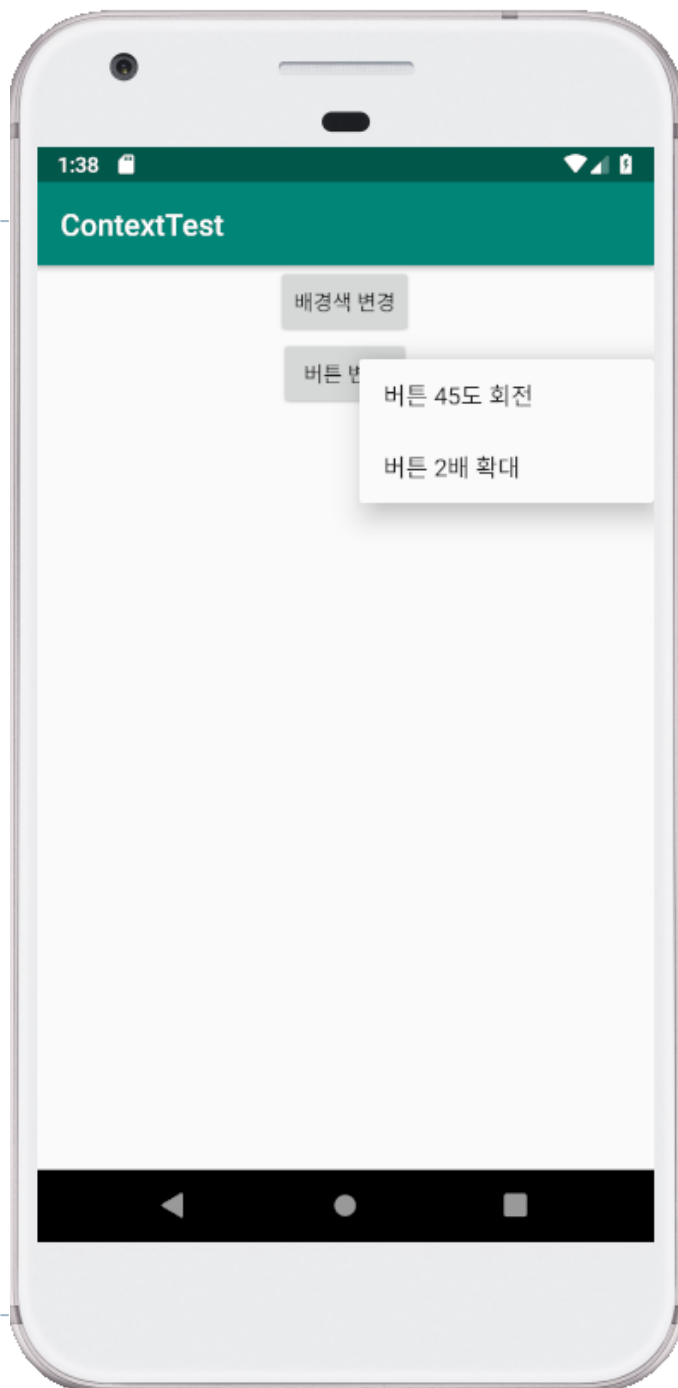
OK

Cancel

```
30  @Override
31  public void onCreateContextMenu(ContextMenu menu, View v, ContextMenu.ContextMenuInfo menuInfo) {
32
33      super.onCreateContextMenu(menu, v, menuInfo);
34
35      MenuInflater inflater = getMenuInflater();
36      if (v == button1) {
37          menu.setHeaderTitle("배경색 변경");
38
39          inflater.inflate(R.menu.menu1, menu);
40      }
41      if (v == button2) {
42          inflater.inflate(R.menu.menu2, menu);
43      }
44
45  }
```



```
49  @Override
50  public boolean onContextItemSelected(MenuItem item) {
51
52      switch (item.getItemId()) {
53          case R.id.itemRed:
54              baseLayout.setBackgroundColor(Color.RED);
55              return true;
56          case R.id.itemGreen:
57              baseLayout.setBackgroundColor(Color.GREEN);
58              return true;
59          case R.id.itemBlue:
60              baseLayout.setBackgroundColor(Color.BLUE);
61              return true;
62          case R.id.subRotate:
63              button2.setRotation(45);
64              return true;
65          case R.id.subSize:
66              button2.setScaleX(2);
67              return true;
68      }
69      return false;
70  }
```



토스트

- ▶ 사용자가 인식해야 할 작은 메시지를 줄 때 편리
- ▶ 프로그래머가 디버깅 용도로 사용하기도 함
- ▶ 일반적인 형태

```
Toast.makeText(Context context, String message, int duration).show();
```



토스트

▶ 위치 변경


- ▶ 기본적으로는 화면의 중앙 하단에 표현됨
- ▶ `setGravity()` 메소드를 사용하여 위치 변경 가능

```
Toast.setGravity(int gravity, int xOffset, int yOffset);
```



토스트

Create New Project

 Create Android Project

Application name

Company domain

Project location
 ...

Package name
 Edit

☐ Include C++ support

☐ Include Kotlin support

Previous Next Cancel Finish

```
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:layout_width="fill_parent"
5   android:layout_height="fill_parent"
6   android:gravity="center"
7   android:orientation="vertical" >
8
9   <Button
10     android:id="@+id/button1"
11     android:layout_width="wrap_content"
12     android:layout_height="wrap_content"
13     android:text="메시지 출력" />
14
15 </LinearLayout>
```

```

14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44

```

```

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

    setTitle("토스트 연습");

    final Button button1 = (Button) findViewById(R.id.button1);

    button1.setOnClickListener(new View.OnClickListener() {

        public void onClick(View v) {

            Toast tMsg = Toast.makeText(context MainActivity.this, text "토스트 연습",
                Toast.LENGTH_SHORT);

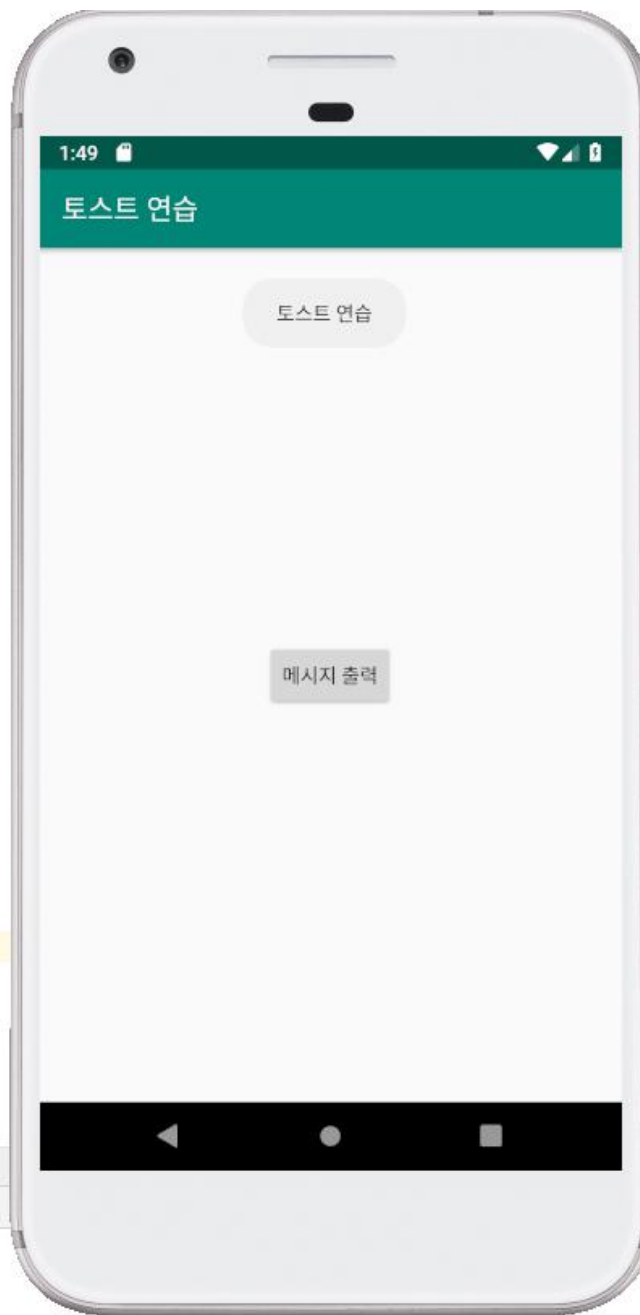
            Display display = ((WindowManager) getSystemService(WINDOW_SERVICE))
                .getDefaultDisplay();
            int xOffset = (int) (Math.random() * display.getWidth());
            int yOffset = (int) (Math.random() * display.getHeight());

            tMsg.setGravity(gravity Gravity.TOP | Gravity.LEFT, xOffset, yOffset);
            tMsg.show();

        }

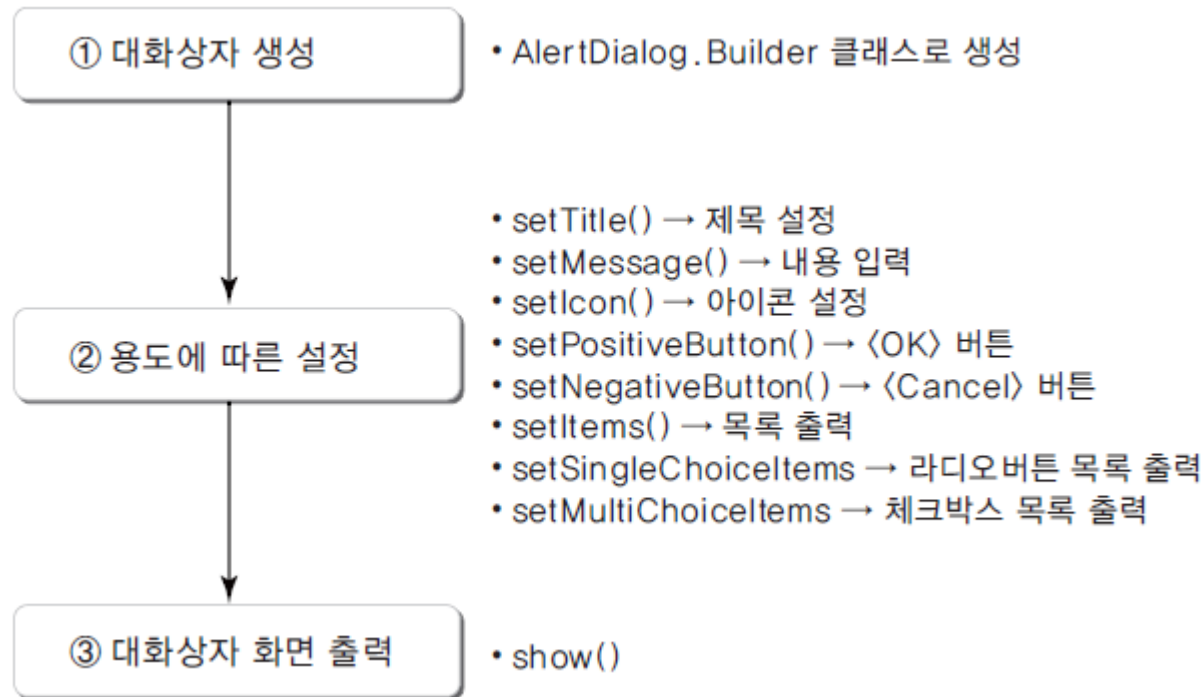
    });
}

```



대화상자

- ▶ 화면에 메시지를 보낸 후 확인/취소 등의 사용자 선택을 받음
- ▶ 대화상자 설정 순서는 아래와 같음





Create Android Project

Application name

Company domain

Project location



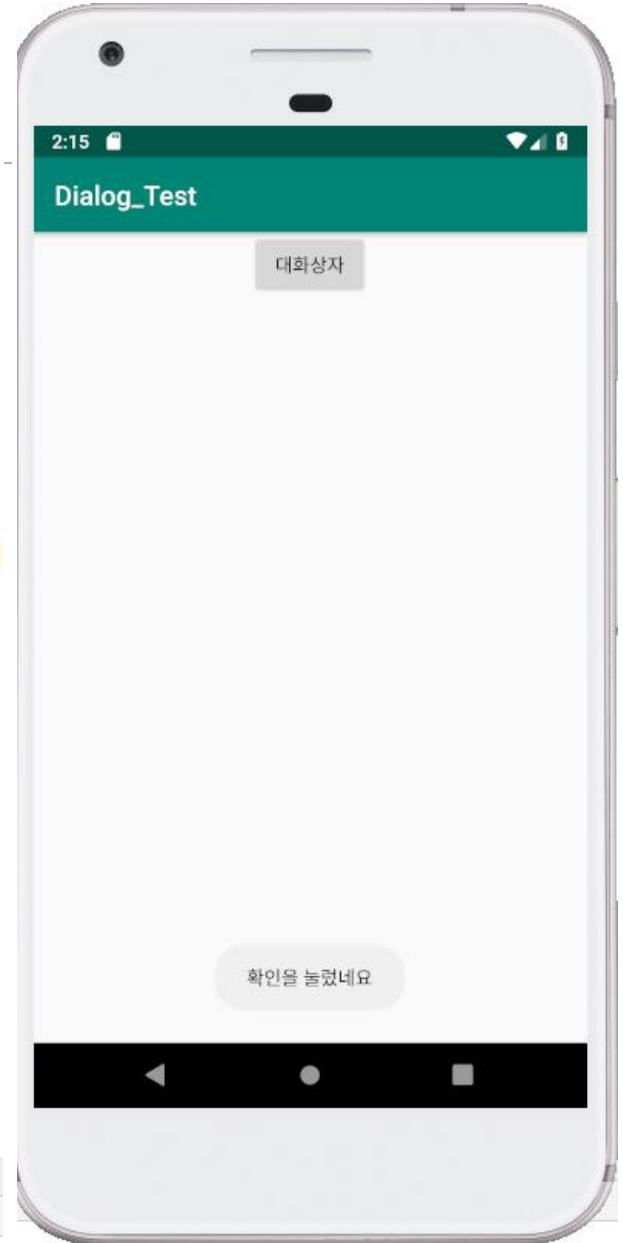
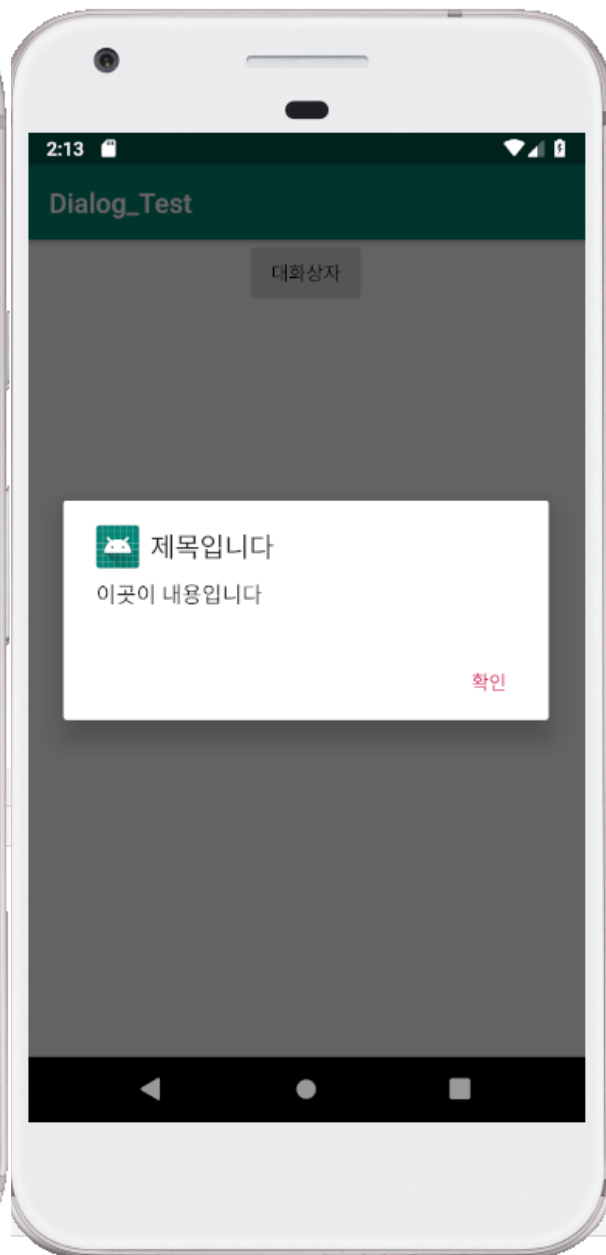
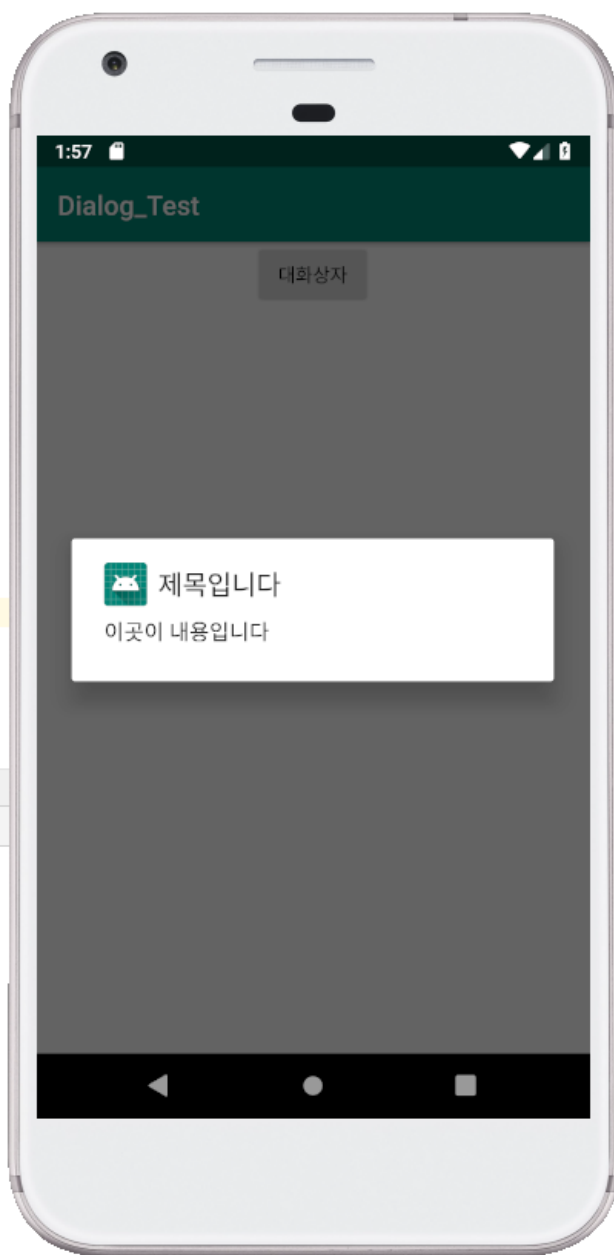
Package name

☐ Include C++ support☐ Include Kotlin support

```
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:layout_width="fill_parent"
5   android:layout_height="fill_parent"
6   android:gravity="center_horizontal"
7   android:orientation="vertical" >
8
9   <Button
10     android:id="@+id/button1"
11     android:layout_width="wrap_content"
12     android:layout_height="wrap_content"
13     android:text="대화상자" />
14
15 </LinearLayout>
```


@Override

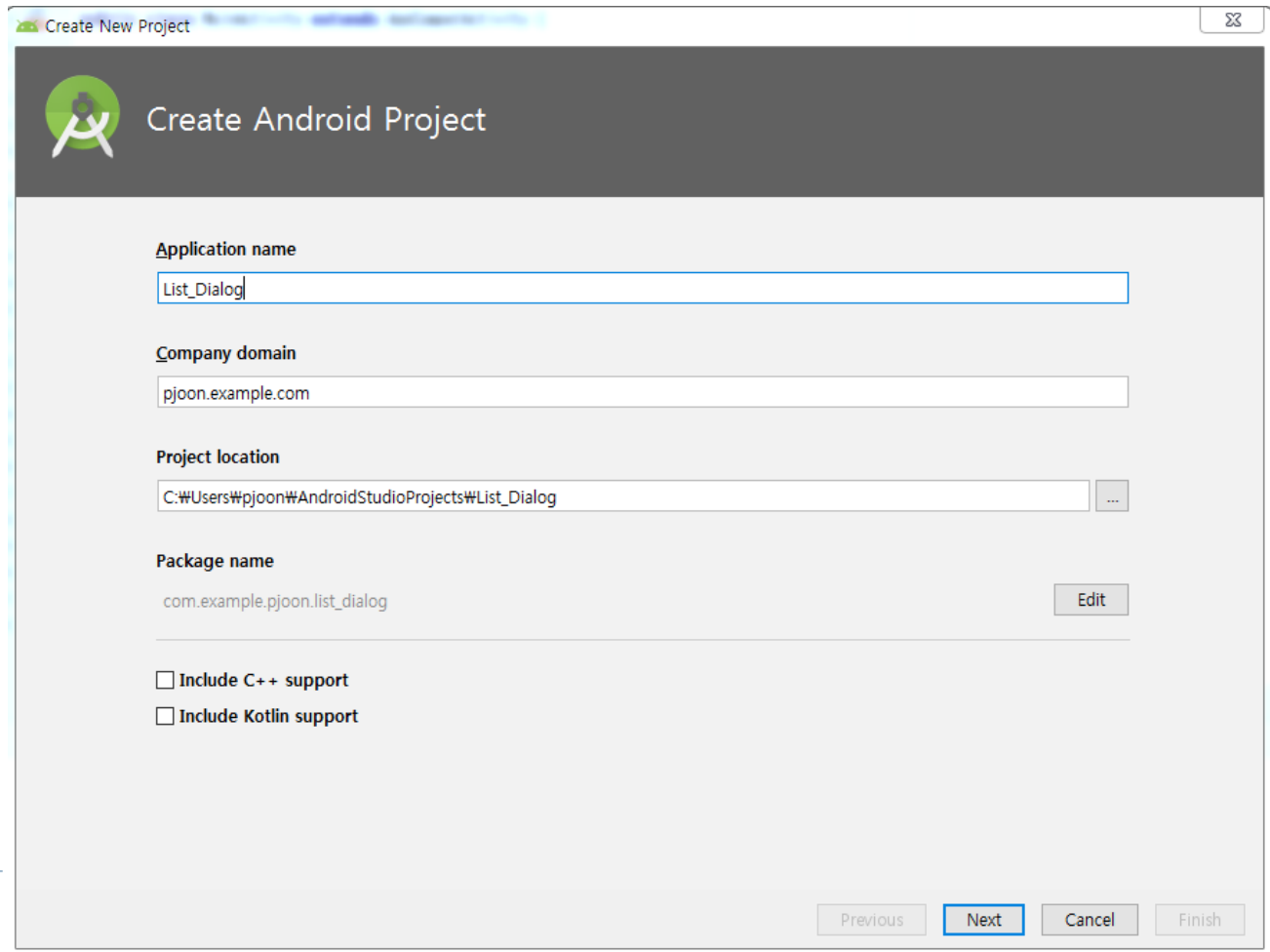
```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
  
    final Button button1 = (Button) findViewById(R.id.button1);  
    button1.setOnClickListener(new View.OnClickListener() {  
  
        public void onClick(View v) {  
  
            AlertDialog.Builder dlg = new AlertDialog.Builder(  
                context MainActivity.this);  
            dlg.setTitle("제목입니다");  
            dlg.setMessage("이곳이 내용입니다");  
            dlg.setIcon(R.mipmap.ic_launcher);  
            /*버튼이 한개인 대화 상자  
            dlg.setPositiveButton("확인", null);  
            */  
            /*버튼 클릭시 동작하는 대화상자  
            dlg.setPositiveButton("확인",  
                new DialogInterface.OnClickListener() {  
                    public void onClick(DialogInterface dialog, int which) {  
                        Toast.makeText(MainActivity.this,  
                            "확인을 눌렀네요", Toast.LENGTH_SHORT).show();  
                    }  
                });  
            */  
            dlg.show();  
        }  
    });  
}
```



대화상자

▶ 목록 대화상자

- ▶ 대화상자에 리스트 형태의 목록을 출력하고 선택



Create New Project

Create Android Project

Application name
List_Dialog

Company domain
pjoon.example.com

Project location
C:\Users\pjoon\AndroidStudioProjects\List_Dialog

Package name
com.example.pjoon.list_dialog

☐ Include C++ support

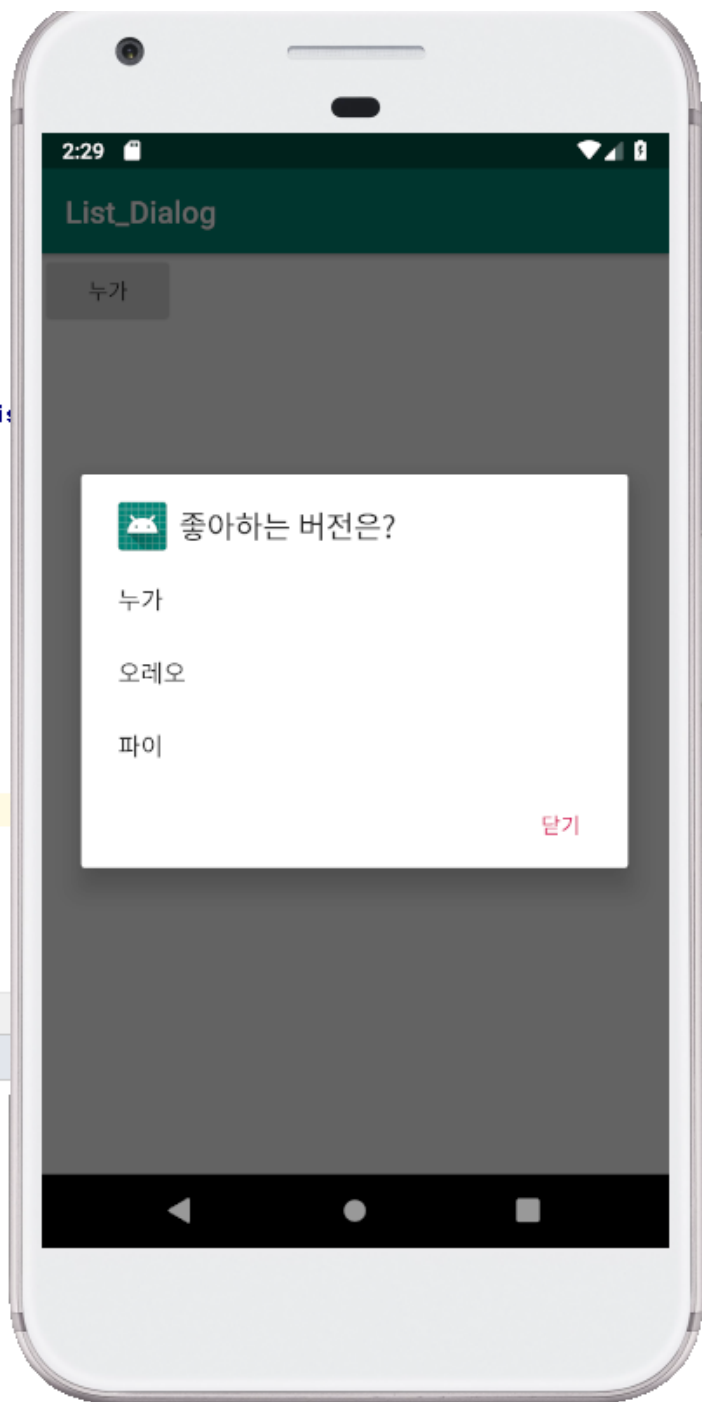
☐ Include Kotlin support

Previous Next Cancel Finish

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     tools:context=".MainActivity">
8
9     <Button
10         android:id="@+id/button1"
11         android:layout_width="wrap_content"
12         android:layout_height="wrap_content"
13         android:text="클릭"/>
14
15 </android.support.constraint.ConstraintLayout>
```

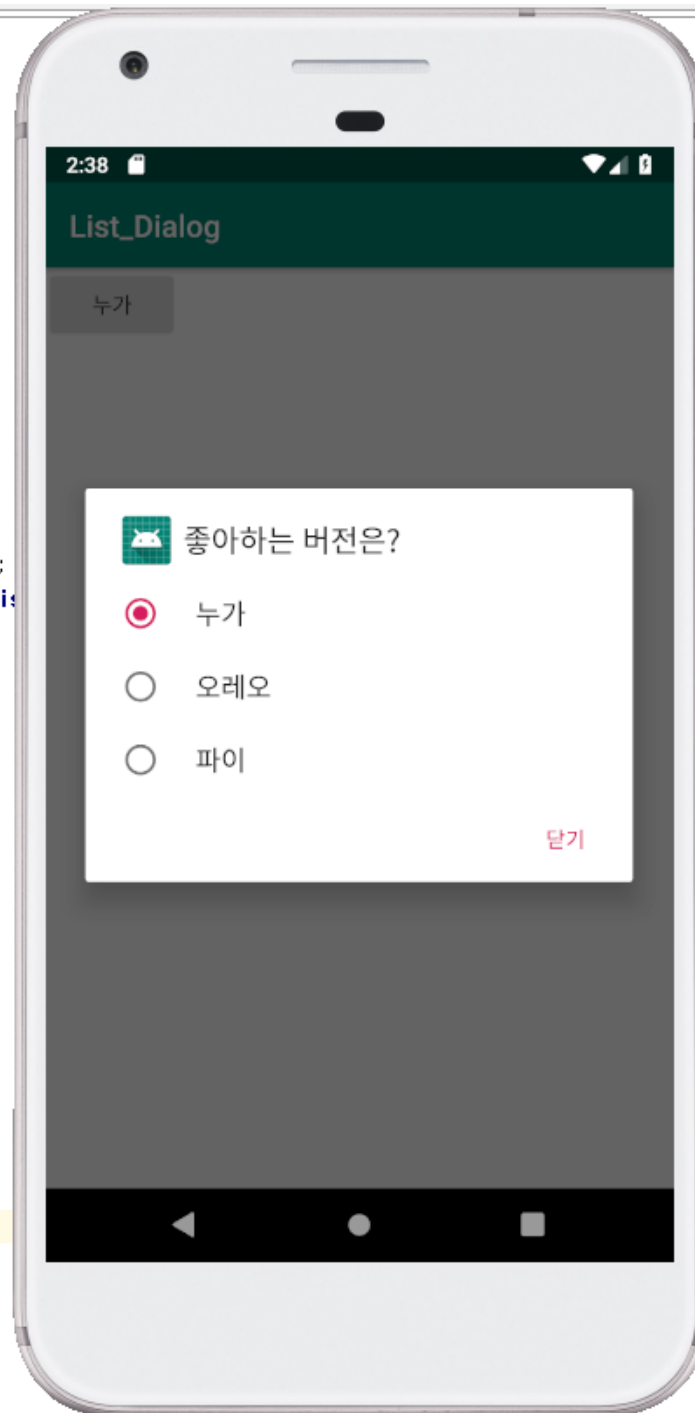
override

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
  
    final Button button1 = (Button) findViewById(R.id.button1);  
    button1.setOnClickListener(new View.OnClickListener() {  
  
        public void onClick(View v) {  
  
            final String[] versionArray = new String[] { "누가", "오레오", "파이" };  
            AlertDialog.Builder dlg = new AlertDialog.Builder(context MainActivity.this);  
            dlg.setTitle("좋아하는 버전은?");  
            dlg.setIcon(R.mipmap.ic_launcher);  
            dlg.setItems(versionArray,  
                new DialogInterface.OnClickListener() {  
                    public void onClick(DialogInterface dialog, int which) {  
                        button1.setText(versionArray[which]);  
                    }  
                });  
            dlg.setPositiveButton(text "닫기", listener: null);  
            dlg.show();  
        }  
    });  
}
```



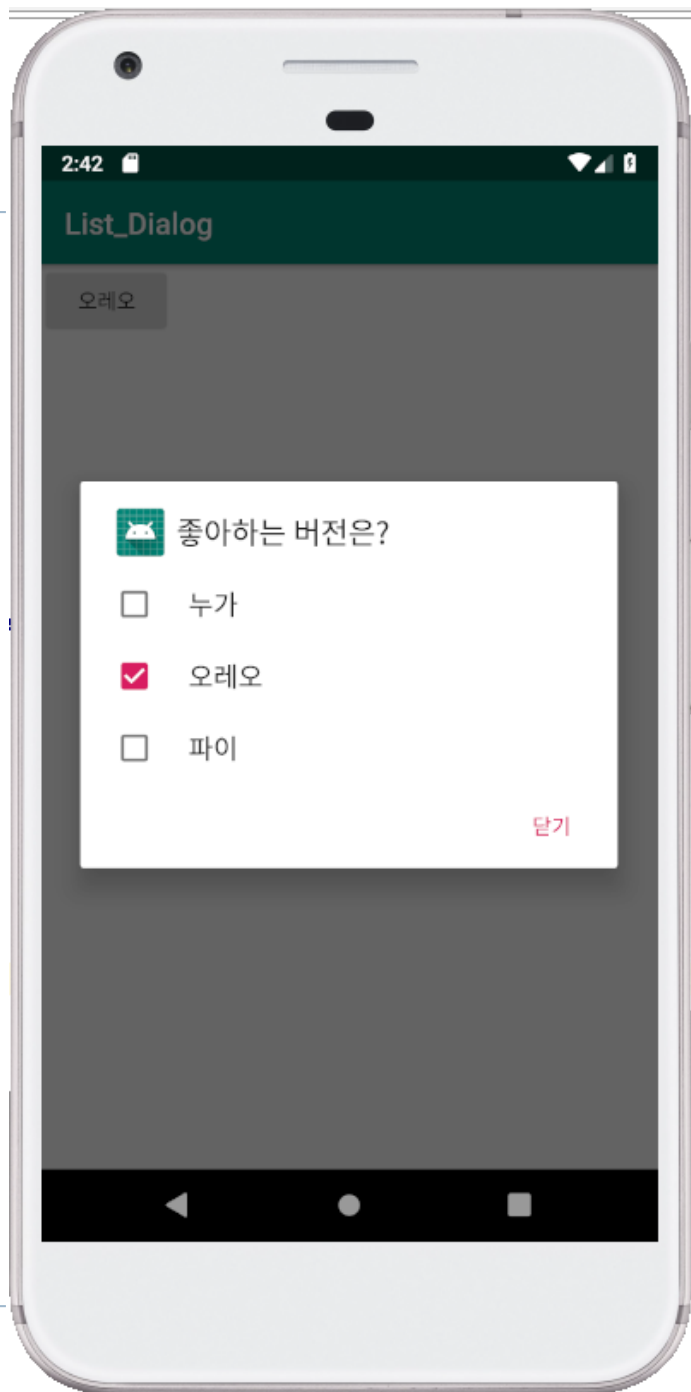
대화상자 – 라디오 버튼

```
10 public class MainActivity extends AppCompatActivity {  
11  
12     @Override  
13     protected void onCreate(Bundle savedInstanceState) {  
14         super.onCreate(savedInstanceState);  
15         setContentView(R.layout.activity_main);  
16  
17         final Button button1 = (Button) findViewById(R.id.button1);  
18         button1.setOnClickListener(new View.OnClickListener() {  
19  
20             public void onClick(View v) {  
21  
22                 final String[] versionArray = new String[] { "누가", "오레오", "파이" };  
23                 AlertDialog.Builder dlg = new AlertDialog.Builder(context MainActivity.this);  
24                 dlg.setTitle("좋아하는 버전은?");  
25                 dlg.setIcon(R.mipmap.ic_launcher);  
26                 //라디오 버튼  
27                 dlg.setSingleChoiceItems(versionArray, checkedItem: 0,  
28                     new DialogInterface.OnClickListener() {  
29                         public void onClick(DialogInterface dialog, int which) {  
30                             button1.setText(versionArray[which]);  
31                         }  
32                     });  
33  
34                 dlg.setPositiveButton(text: "닫기", listener: null);  
35                 dlg.show();  
36             }  
37         });  
38  
39     }  
40 }  
41  
42
```

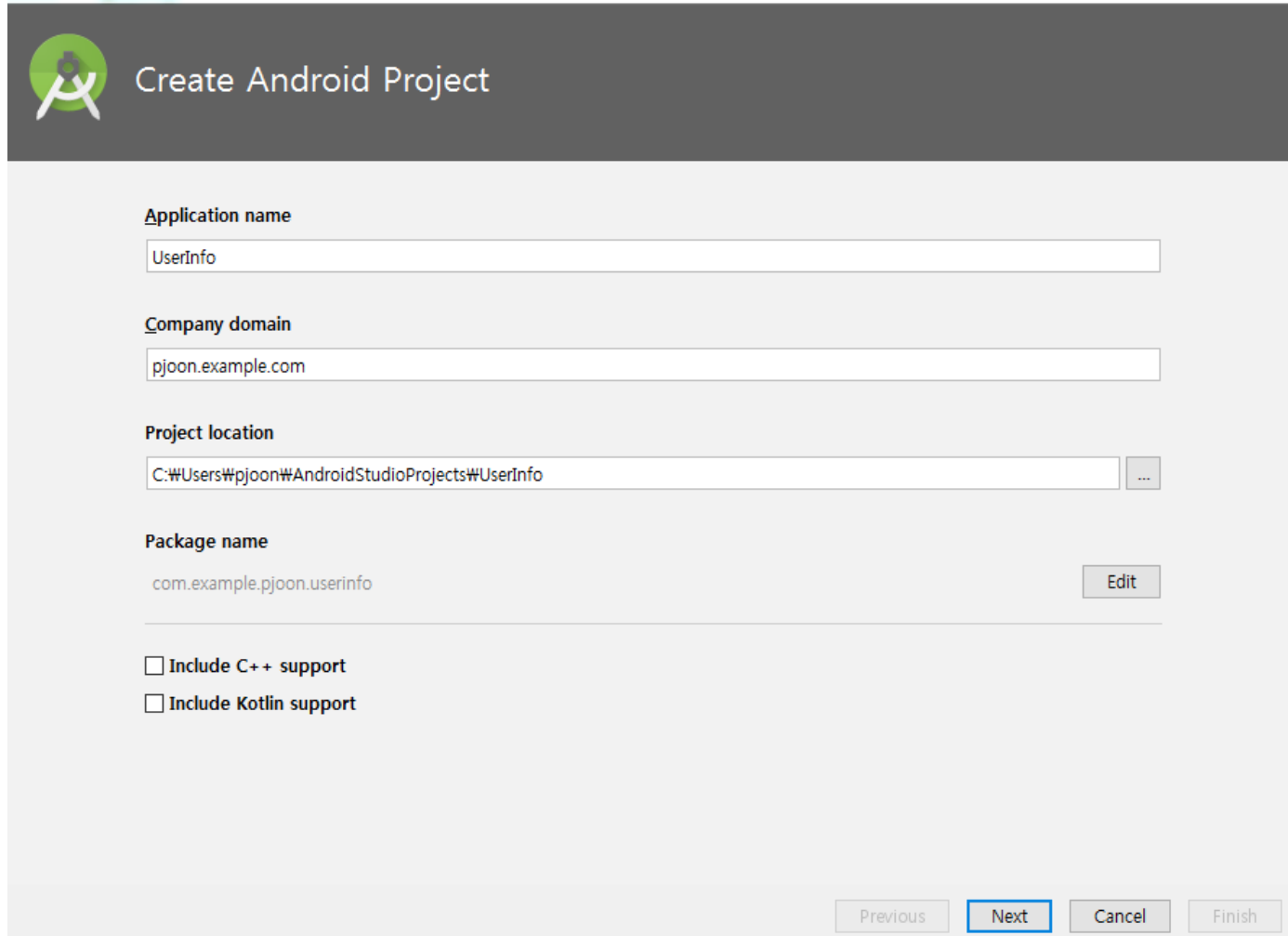


대화상자 – 체크박스

```
12  @Override
13  protected void onCreate(Bundle savedInstanceState) {
14      super.onCreate(savedInstanceState);
15      setContentView(R.layout.activity_main);
16
17      final Button button1 = (Button) findViewById(R.id.button1);
18      button1.setOnClickListener(new View.OnClickListener() {
19
20          public void onClick(View v) {
21
22              final String[] versionArray = new String[] { "누가", "오래오", "파미" };
23              final boolean[] checkArray = new boolean[] { true, false, false };
24              AlertDialog.Builder dlg = new AlertDialog.Builder( context MainActivity.this);
25              dlg.setTitle("좋아하는 버전은?");
26              dlg.setIcon(R.mipmap.ic_launcher);
27              //체크박스 버튼
28              dlg.setMultiChoiceItems(versionArray, checkArray,
29                  new DialogInterface.OnMultiChoiceClickListener() {
30                      public void onClick(DialogInterface dialog,
31                          int which, boolean isChecked) {
32                          button1.setText(versionArray[which]);
33                      }
34                  });
35
36              dlg.setPositiveButton( text "닫기", listener: null);
37              dlg.show();
38          }
39      });
40
41  }
```



사용자 정보입력 앱



The image shows the 'Create Android Project' dialog in Android Studio. It has a dark grey header with the Android logo and the title 'Create Android Project'. The main area is light grey and contains several input fields and checkboxes. The 'Application name' field contains 'UserInfo'. The 'Company domain' field contains 'pjoon.example.com'. The 'Project location' field contains 'C:\Users\pjoon\AndroidStudioProjects\UserInfo' and has a browse button (...). The 'Package name' field contains 'com.example.pjoon.userinfo' and has an 'Edit' button. There are two checkboxes: 'Include C++ support' and 'Include Kotlin support', both of which are unchecked. At the bottom, there are four buttons: 'Previous', 'Next' (which is highlighted with a blue border), 'Cancel', and 'Finish'.

Create Android Project

Application name
UserInfo

Company domain
pjoon.example.com

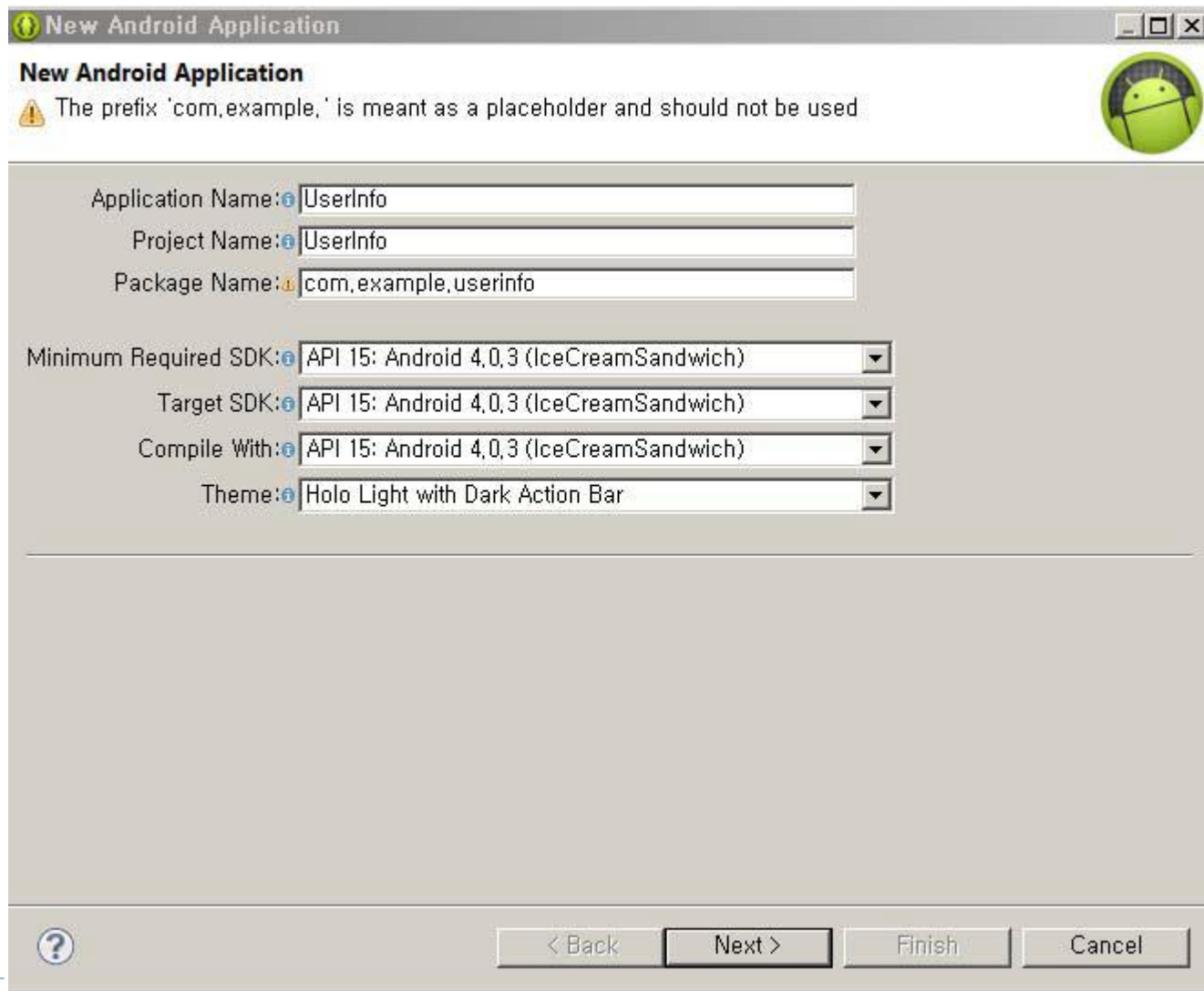
Project location
C:\Users\pjoon\AndroidStudioProjects\UserInfo ...

Package name
com.example.pjoon.userinfo Edit

☐ Include C++ support
☐ Include Kotlin support

Previous Next Cancel Finish

사용자 정보 입력 앱



New Android Application

The prefix 'com.example,' is meant as a placeholder and should not be used

Application Name:

Project Name:

Package Name:

Minimum Required SDK:

Target SDK:

Compile With:

Theme:

```
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:layout_width="fill_parent"
5   android:layout_height="fill_parent"
6   android:gravity="center_horizontal"
7   android:orientation="vertical" >
8
9   <TextView
10     android:id="@+id/tvName"
11     android:layout_width="wrap_content"
12     android:layout_height="wrap_content"
13     android:text="사용자 이름"
14     android:textSize="20dp" />
15
16   <TextView
17     android:id="@+id/tvEmail"
18     android:layout_width="wrap_content"
19     android:layout_height="wrap_content"
20     android:text="이메일"
21     android:textSize="20dp" />
22
23   <Button
24     android:id="@+id/button1"
25     android:layout_width="wrap_content"
26     android:layout_height="wrap_content"
27     android:text="여기를 클릭" />
28
29 </LinearLayout>
```

C:\Users#pjoon#AndroidStudioProjects#UserInfo] - ...#app#src#main#res#layout#activity_main.xml [app] - Android Studio

File View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

app > src > main > res > layout >

Android

app

manifests

java

generatedJava

res

drawable

layout

m

va

ad

Sc

ad

Sc

ad

Sc

ad

Sc

ad

Sc

ad

Sc

ad

Sc

ad

Sc

ad

Sc

ad

Sc

ad

activity_main.xml x MainActivity.java x

```
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:layout_width="fill_parent"
5     android:layout_height="fill_parent"
6     android:gravity="center_horizontal"
7     android:orientation="vertical" >
```

New > Kotlin File/Class
Layout resource file
Sample Data Directory
File
Scratch File Ctrl+Alt+Shift+Insert
Directory
C++ Class
C/C++ Source File
C/C++ Header File
Image Asset
Vector Asset
Kotlin Script

```
"@+id/tvName"
out_width="wrap_content"
out_height="wrap_content"
t="사용자 이름"
tSize="20dp" />

"@+id/tvEmail"
out_width="wrap_content"
out_height="wrap_content"
t="이메일"
tSize="20dp" />
```

New Resource File



File name: dialog1.xml

Root element: LinearLayout

Source set: main

Directory name: layout

Available qualifiers:

- Country Code
- Network Code
- Locale
- Layout Direction
- Smallest Screen Width
- Screen Width
- Screen Height
- Size
- Ratio
- Orientation
- UI Mode
- Night Mode

Chosen qualifiers:

Nothing to show

>>

<<



OK

Cancel

```
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
7     <TextView
8         android:layout_width="wrap_content"
9         android:layout_height="wrap_content"
10        android:text="사용자 이름"
11        android:textSize="20dp" />
12
13    <EditText
14        android:id="@+id/dlgEdt1"
15        android:layout_width="match_parent"
16        android:layout_height="wrap_content" />
17
18    <TextView
19        android:layout_width="wrap_content"
20        android:layout_height="wrap_content"
21        android:text="이메일"
22        android:textSize="20dp" />
23
24    <EditText
25        android:id="@+id/dlgEdt2"
26        android:layout_width="match_parent"
27        android:layout_height="wrap_content" />
28
29 </LinearLayout>
```


File name: toast1.xml

Root element: LinearLayout

Source set: main

Directory name: layout

Available qualifiers:

- Country Code
- Network Code
- Locale
- Layout Direction
- Smallest Screen Width
- Screen Width
- Screen Height
- Size
- Ratio
- Orientation
- UI Mode
- Night Mode

Chosen qualifiers:

Nothing to show



OK

Cancel

```

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

```



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

```
button1.setOnClickListener(new View.OnClickListener() {  
  
    public void onClick(View v) {  
        dialogView = (View) View.inflate( context MainActivity.this,  
            R.layout.dialog1, root null);  
  
        AlertDialog.Builder dlg = new AlertDialog.Builder(  
            context MainActivity.this);  
        dlg.setTitle("사용자 정보 입력");  
        dlg.setIcon(R.drawable.ic_menu_allfriends);  
        dlg.setView(dialogView);  
        dlg.setPositiveButton( text "확인", listener null);  
  
        dlg.setNegativeButton( text "취소", listener null);  
        dlg.show();  
    }  
});
```



대화상자의 <확인> 버튼을 클릭했을 때, 대화상자에서 입력한 사용자 이름과 이메일이 본 화면의 텍스트뷰에 쓰지도록 코딩

```
41 //dlg.setPositiveButton("확인", null);
42 dlg.setPositiveButton( text "확인",
43     new DialogInterface.OnClickListener() {
44         public void onClick(DialogInterface dialog, int which) {
45             dlgEdtName = (EditText) dialogView
46                 .findViewById(R.id.dlgEdt1);
47             dlgEdtEmail = (EditText) dialogView
48                 .findViewById(R.id.dlgEdt2);
49
50             tvName.setText(dlgEdtName.getText().toString());
51             tvEmail.setText(dlgEdtEmail.getText()
52                 .toString());
53         }
54     });
55
```

<취소> 버튼을 클릭했을 때, toast1.xml이 토스트 메시지로 나오도록 설정

```
57 // dlg.setNegativeButton("취소", null);
58
59 dlg.setNegativeButton( text "취소",
60     new DialogInterface.OnClickListener() {
61         public void onClick(DialogInterface dialog,
62             int which) {
63             Toast toast = new Toast( context MainActivity.this);
64
65             toastView = (View) View.inflate(
66                 context MainActivity.this,
67                 R.layout.toast1, root null);
68             toastText = (TextView) toastView
69                 .findViewById(R.id.toastText1);
70             toastText.setText("취소했습니다");
71             toast.setView(toastView);
72             toast.show();
73         }
74     });
75
76 dlg.show();
```

