



# PopUp 메시지 구현





# 학습 목표

## 교육 목표

❖ Toast Class

❖ AlertDialog

◆ CostumView 미구현

◆ CostumView 구현



Navigation voice



Traffic view



Street View





# Toast Class (1)

## ■ Toast Class

### ❖ 기능

- ◆ 사용자 메시지 화면 출력 기능 제공

### ❖ 특징

- ◆ 일시적인 메시지 출력
- ◆ Focus 받지 않음

### ❖ Constants

Constants	설명
<code>int LENGTH_LONG</code>	a long period of time
<code>int LENGTH_SHORT</code>	a short period of time

```
public class
Toast
extends Object

java.lang.Object
↳ android.widget.Toast
```





# Toast Class (2)

## ❖ 생성자

Method	설 명
Toast( <a href="#">Context</a> context)	Toast 객체 생성

## ❖ Method

Method	설 명
makeText( <a href="#">Context</a> context, <a href="#">int</a> resId, <a href="#">int</a> duration)	Toast 구현
makeText( <a href="#">Context</a> context, <a href="#">CharSequence</a> text, <a href="#">int</a> duration)	Toast 구현
<a href="#">int</a> getDuration()	주기 반환
<a href="#">void</a> setDuration( <a href="#">int</a> duration)	주기 설정
<a href="#">void</a> setText( <a href="#">int</a> resId / <a href="#">CharSequence</a> text)	Text 지정
<a href="#">void</a> setView( <a href="#">View</a> v)	View 지정
<a href="#">void</a> show( )	지정 주기로 view show

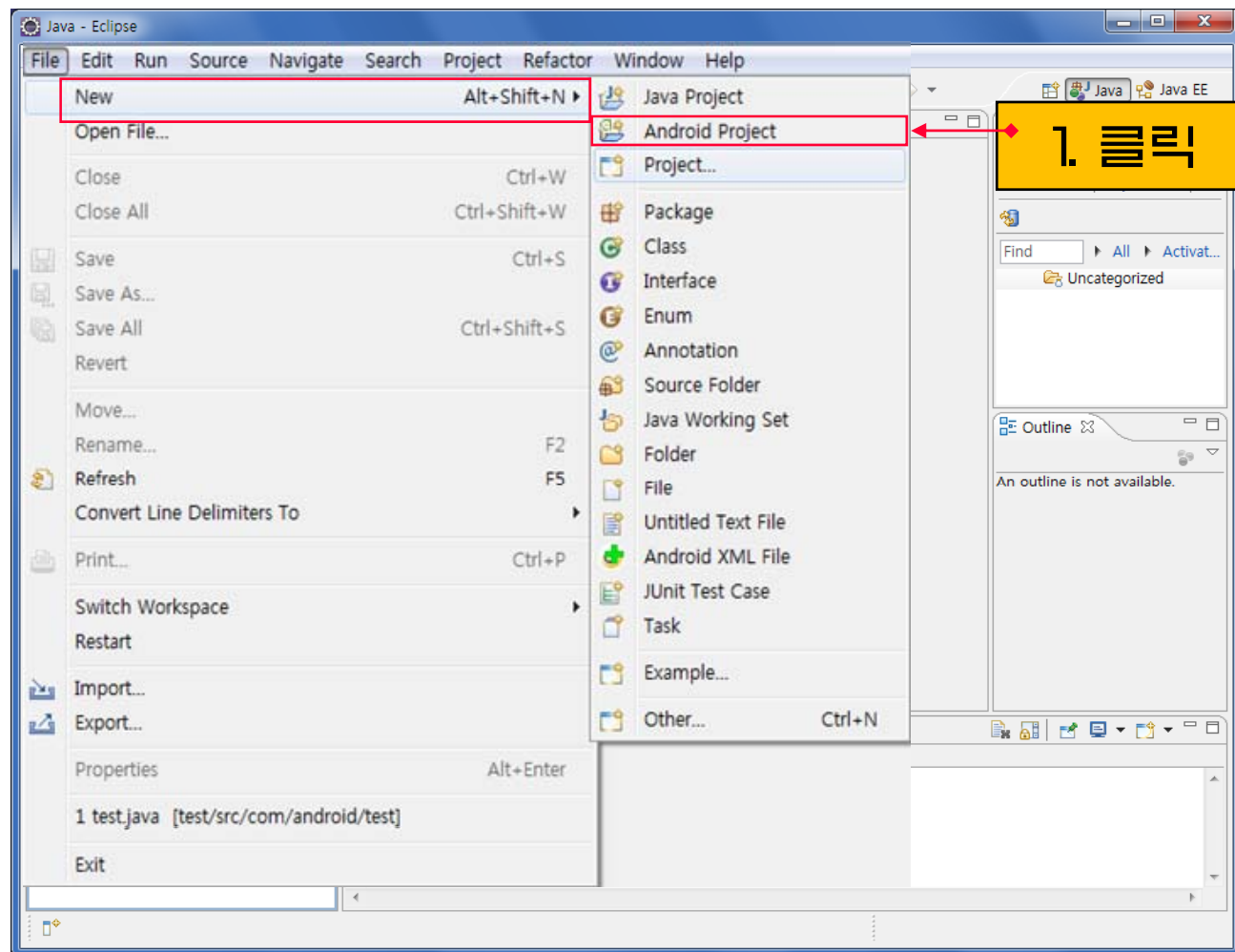




# Toast 구현 (1)

## ■ Android 프로젝트 생성

❖ 프로젝트 명 : Android\_Toast





# Toast 구현 (2)

New Android Project

Creates a new Android Project resource.

Project name:

Contents

☒ Create new project in workspace  
☐ Create project from existing source  
☒ Use default location

Location:

☐ Create project from existing sample

Samples:

Build Target

Target Name	Vendor	Platform	API ...
<input type="checkbox"/> Android 1.1	Android Open Source Project	1.1	2
<input type="checkbox"/> Android 2.0.1	Android Open Source Project	2.0.1	6
<input checked="" type="checkbox"/> Android 2.1	Android Open Source Project	2.1	7
<input type="checkbox"/> Google APIs	Google Inc.	2.0.1	6
<input type="checkbox"/> Google APIs	Google Inc.	2.1	7

Standard Android platform 2.1

Properties

Application name:

Package name:

☒ Create Activity:

Min SDK Version:

2. Android\_PopUp 입력

3. 클릭

4. Android\_PopUp 입력

5. com.inhatec.android\_PopUp 입력

6. Android\_PopUp 입력

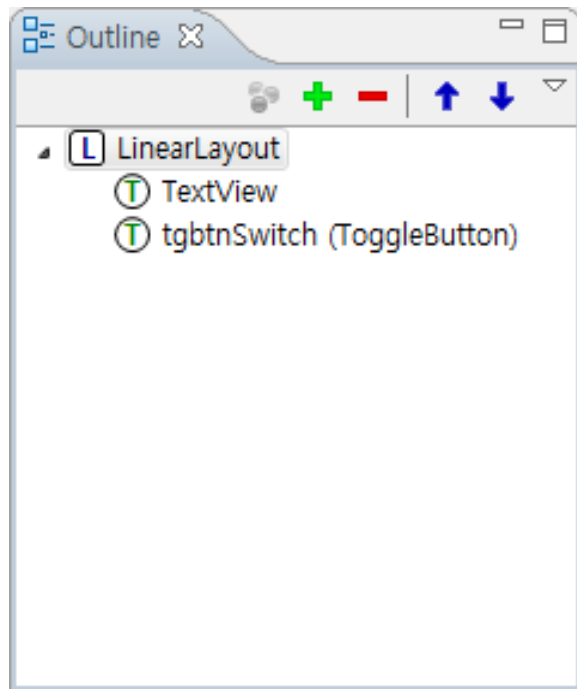
7. 클릭





# Toast 구현 (3)

## UI 설계



8. UI 설계 및 속성 지정



# Toast 구현 (4)

## ❖ ToggleButton 추가

Create a new element at the top level, in LinearLayout.

- TabWidget
- TextSwitcher
- TextView
- TimePicker
- ToggleButton**
- TwoLineListItem
- VideoView
- View
- ViewAnimator
- ViewFlipper
- ViewStub
- ViewSwitcher
- ZoomButton
- ZoomControls

android.wid...oggleButton

OK Cancel

9. 클릭

10. 클릭

11. 클릭

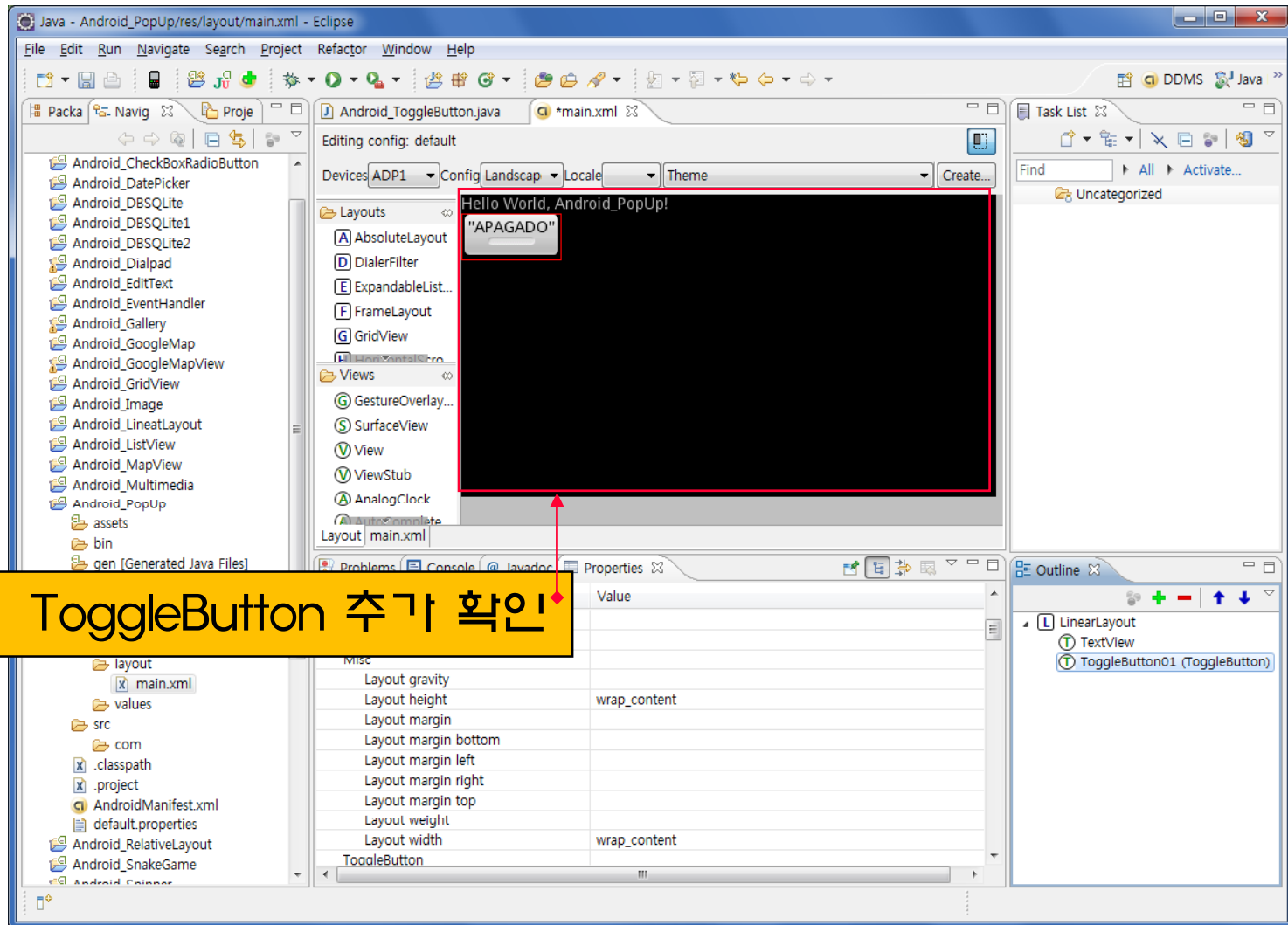
Outline: LinearLayout (1 TextView)





# Toast 구현 (5)

## ❖ ToggleButton 추가 결과





# Toast 구현 (6)

## ❖ TextView, ToggleButton 속성 지정

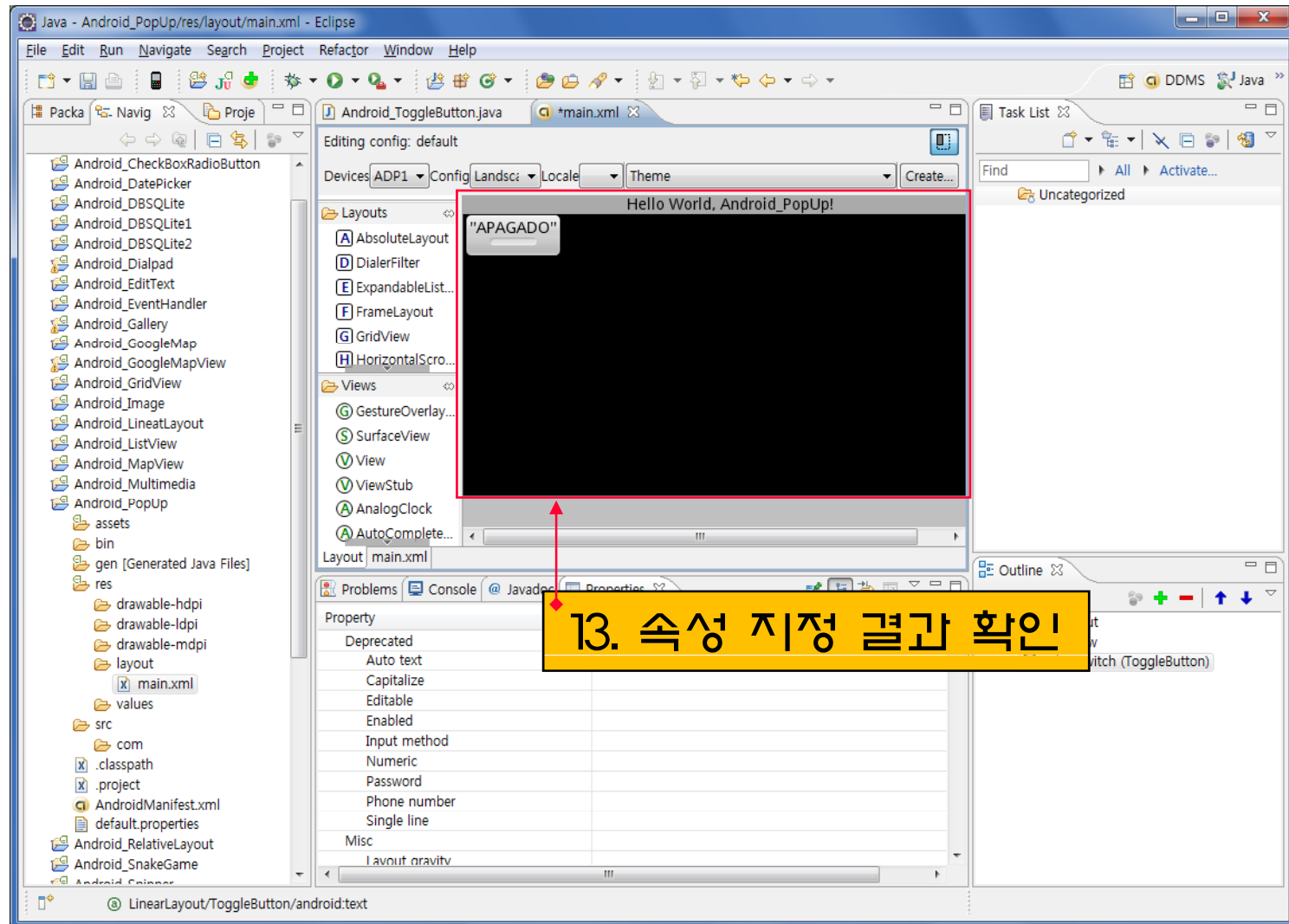
컨트롤	속성 지정
TextView	<ul style="list-style-type: none"><li>• android:layout_width="fill_parent"</li><li>• android:layout_height="wrap_content"</li><li>• android:text="@string/hello"</li><li>• android:gravity="center_horizontal"</li><li>• android:textColor="#FF000000"</li><li>• android:background="#FFaaaaaa"</li></ul>
ToggleButton	<ul style="list-style-type: none"><li>• android:layout_width="wrap_content"</li><li>• android:layout_height="wrap_content"</li><li>• android:id="@+id/tgbtnSwitch"</li><li>• android:text="Switch"</li></ul>





# Toast 구현 (7)

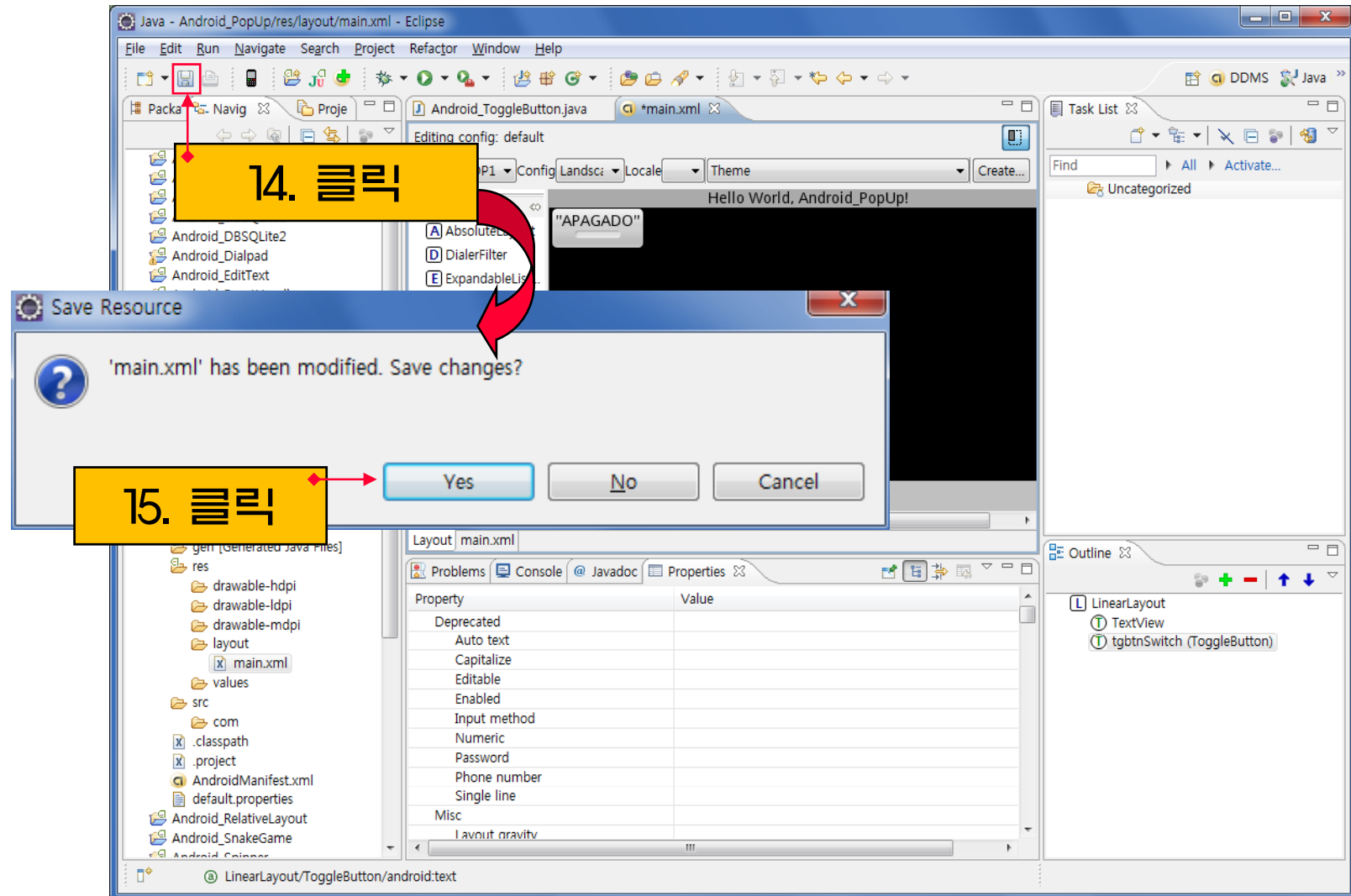
## ❖ TextView, ToggleButton 속성 지정 결과





# Toast 구현 (8)

## ❖ Main.xml 저장





# Toast 구현 (9)

## ❖ Main.xml

```
main.xml
1<?xml version="1.0" encoding="utf-8"?>
2<LinearLayout
3    xmlns:android="http://schemas.android.com/apk/res/android"
4    android:orientation="vertical"
5    android:layout_width="fill_parent"
6    android:layout_height="fill_parent"
7    >
8    <TextView
9        android:layout_width="fill_parent"
10       android:layout_height="wrap_content"
11       android:text="@string/hello"
12       android:gravity="center_horizontal"
13       android:textColor="#FF000000"
14       android:background="#FFaaaaaa"/>
15    <ToggleButton
16        android:layout_width="wrap_content"
17        android:layout_height="wrap_content"
18        android:id="@+id/tgbtnSwitch"
19        android:text="Switch">
20    </ToggleButton>
21</LinearLayout>
```

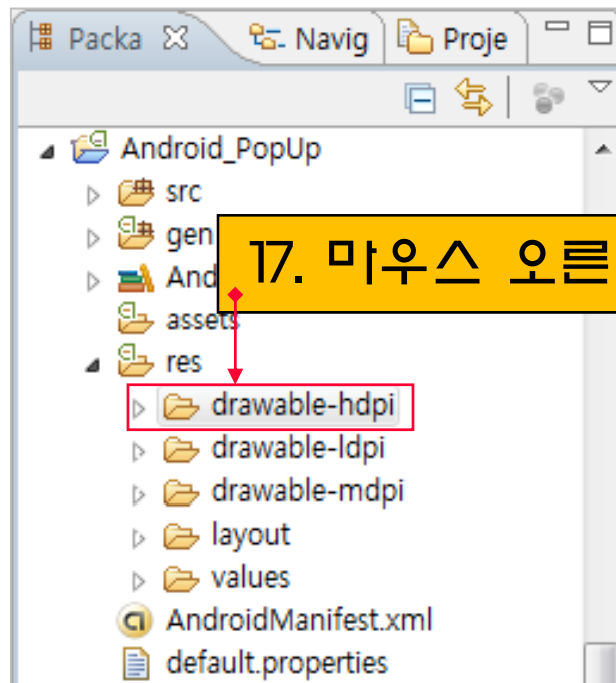
16. XML code 추가 확인



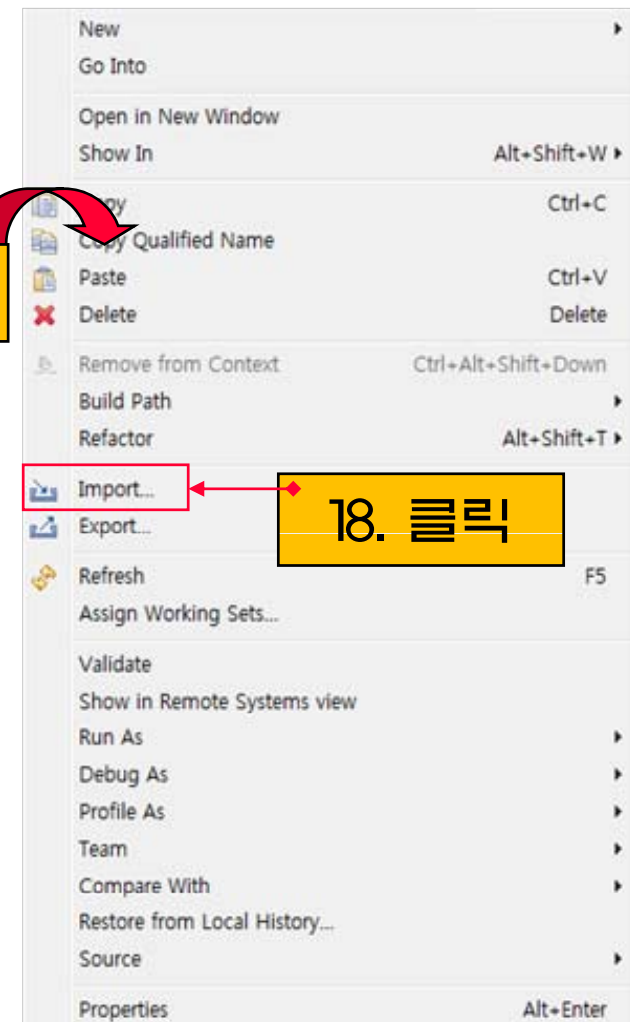


# Toast 구현 (10)

## ❖ Image import



17. 마우스 오른쪽 버튼 클릭

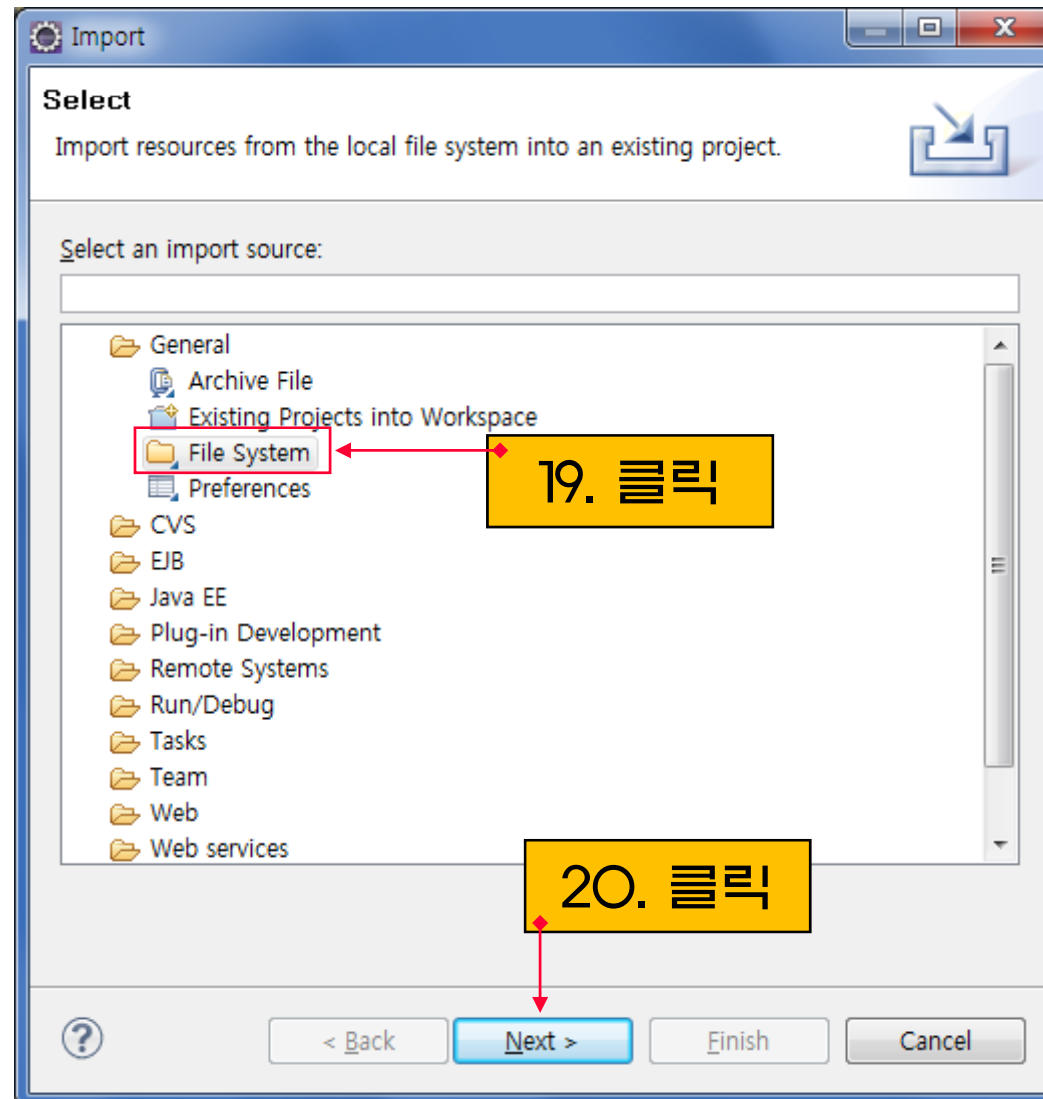


18. 클릭



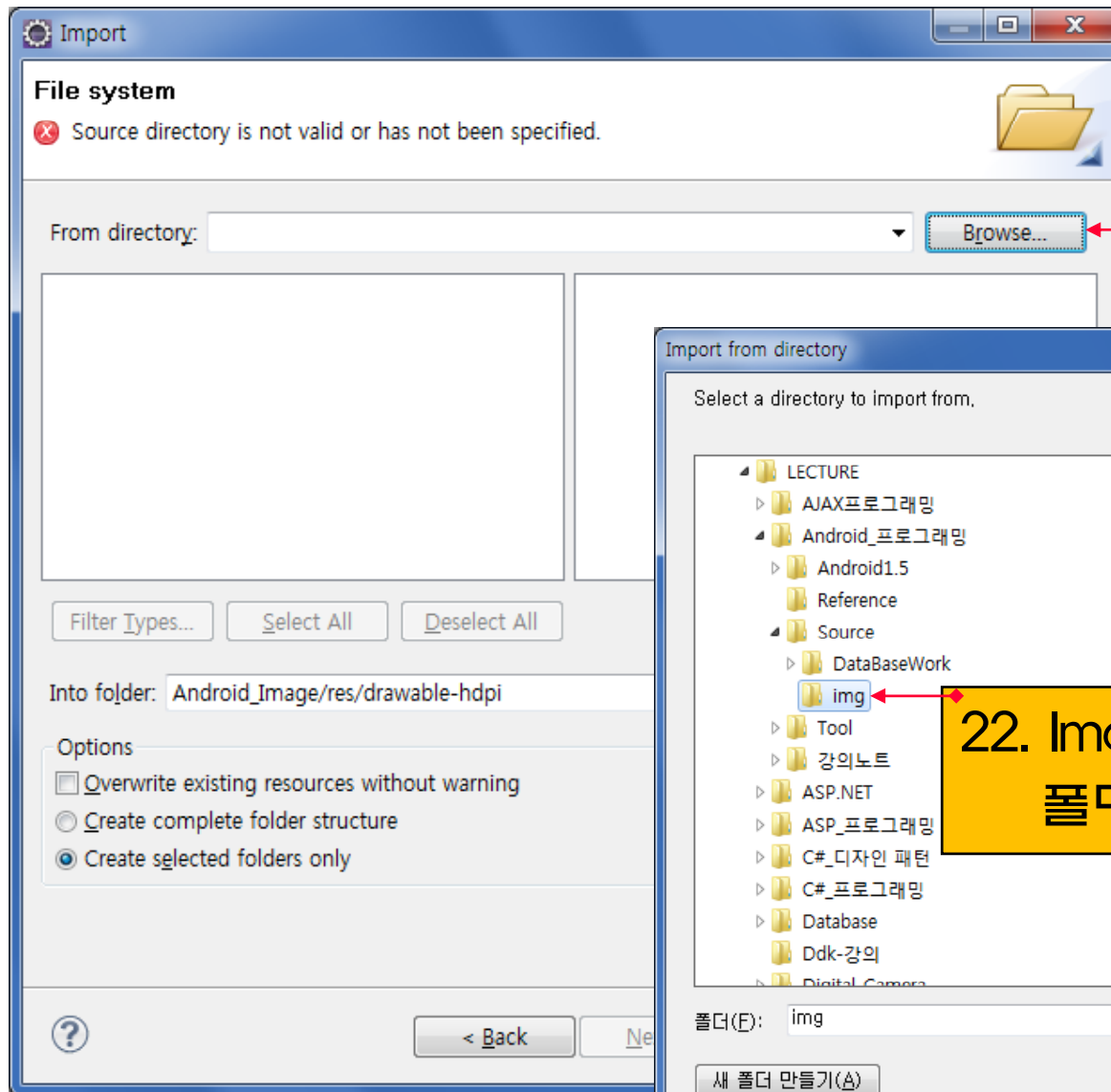


# Toast 구현 (11)

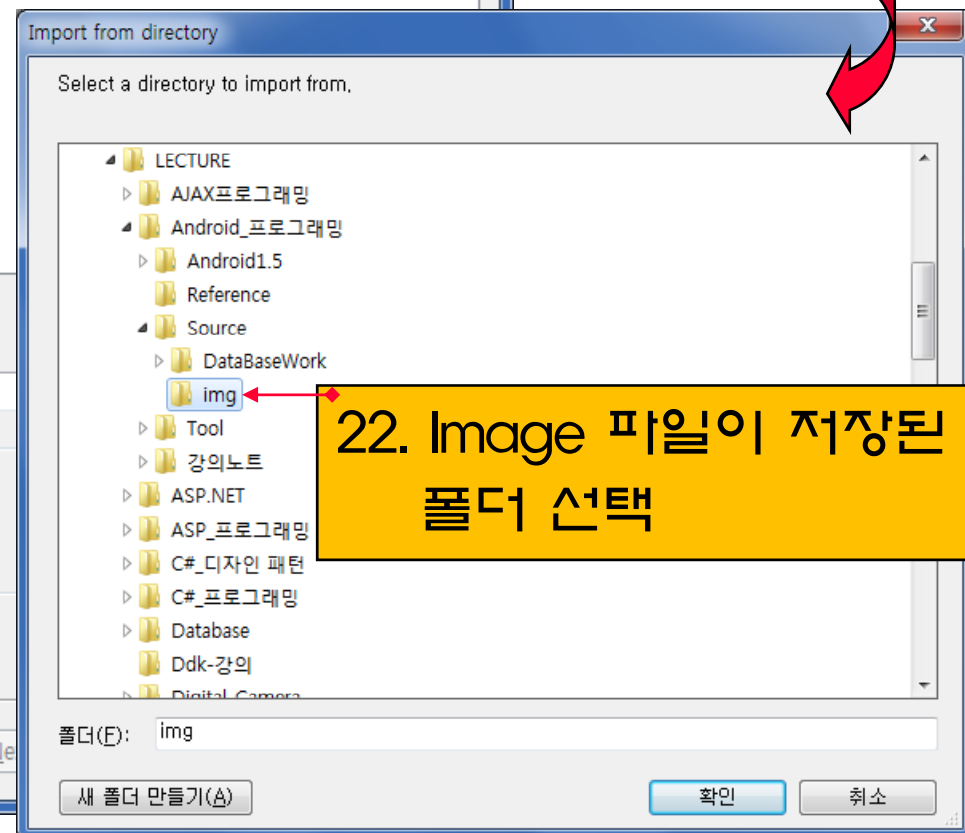




# Toast 구현 (12)



21. 클릭



22. Image 파일이 저장된  
폴더 선택



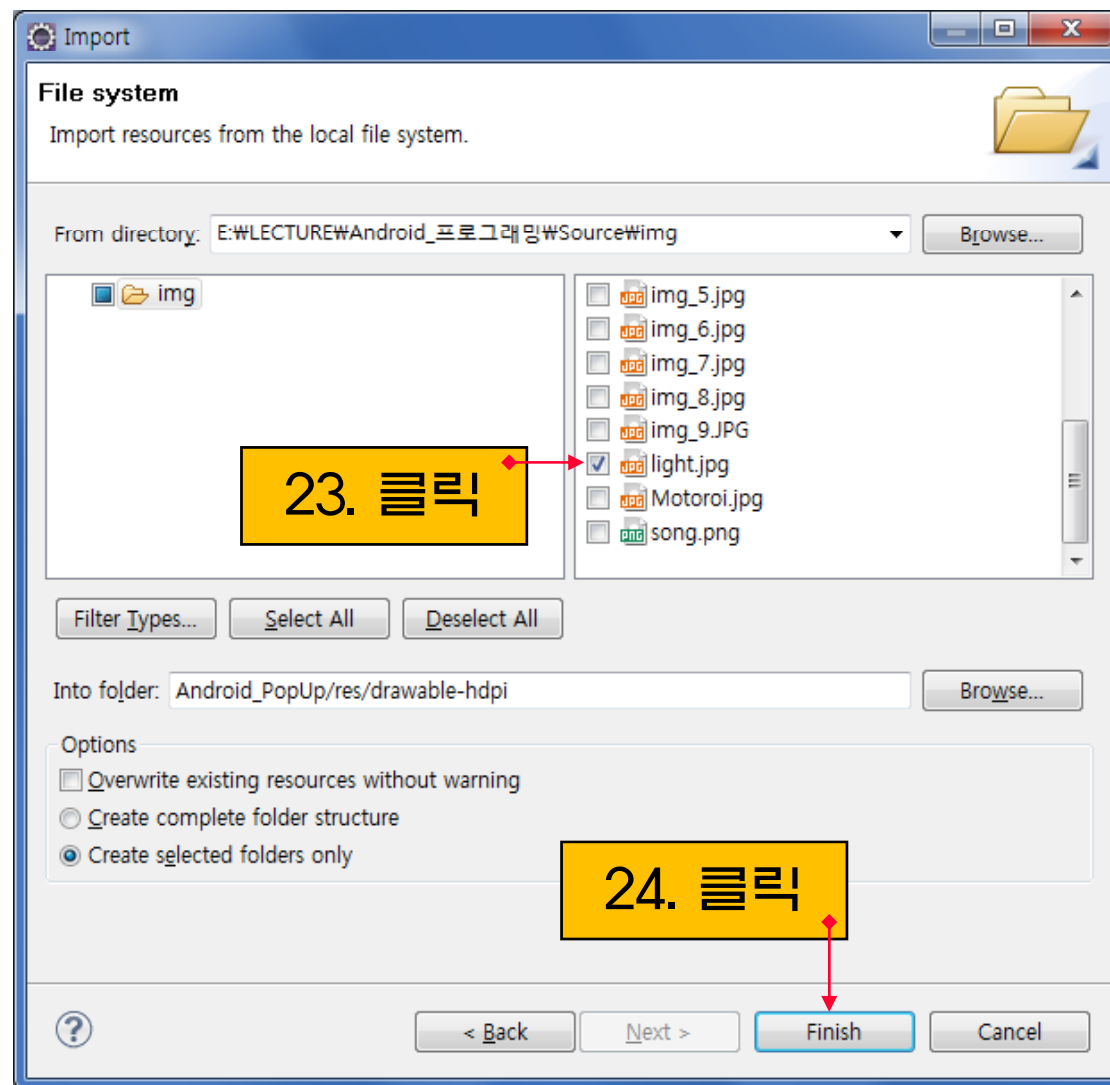




# Toast 구현 (13)

❖ Image 파일 선택

◆ light.jpg

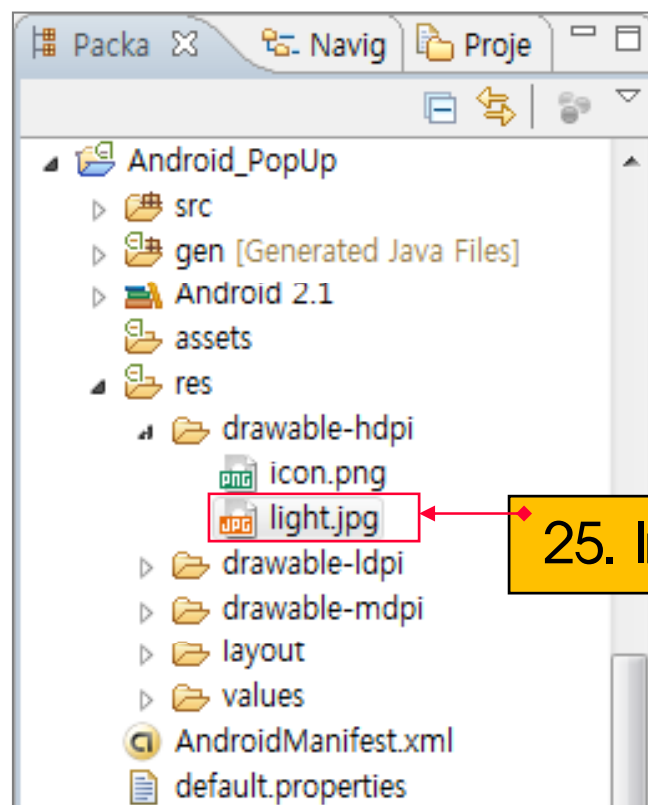




# Toast 구현 (14)

❖ Image 파일 import 확인

◆ light.jpg



25. Image 파일 import 확인





# Toast 구현 (15)

❖ R.java

```
R.java X
1+ /* AUTO-GENERATED FILE.  DO NOT MODIFY.
7
8 package com.inhatec.android_PopUp;
9
10 public final class R {
11     public static final class attr {
12     }
13     public static final class drawable {
14         public static final int icon=0x7f020000;
15         public static final int light=0x7f020001;
16     }
17     public static final class id {
18         public static final int tgbtnSwitch=0x7f050000;
19     }
20     public static final class layout {
21         public static final int main=0x7f030000;
22     }
23     public static final class string {
24         public static final int app_name=0x7f040001;
25         public static final int hello=0x7f040000;
26     }
27 }
```

26. drawable, id 클래스 변수 추가 확인





# Toast 구현 (16)

## ❖ onClickListener() Method 구현

```
*Android_PopUp.java
1 package com.inhatc.android_PopUp;
2
3 import android.app.Activity;
4 import android.os.Bundle;
5 import android.view.View;
6 import android.widget.ImageView;
7 import android.widget.LinearLayout;
8 import android.widget.TextView;
9 import android.widget.Toast;
10 import android.widget.ToggleButton;
11
12 public class Android_PopUp extends Activity
13     implements View.OnClickListener {
14     /** Called when the activity is first created. */
15     ToggleButton objTGBUTTON; //ToggleButton 객체 선언
16
17     @Override
18     public void onCreate(Bundle savedInstanceState) {
19         super.onCreate(savedInstanceState);
20         setContentView(R.layout.main);
21
22         objTGBUTTON = (ToggleButton) findViewById(R.id.tgbtnSwitch);
23         objTGBUTTON.setOnClickListener(this);
24     }
```

27. Coding



# Toast 구현 (17)

## ❖ onCreate() Method 구현

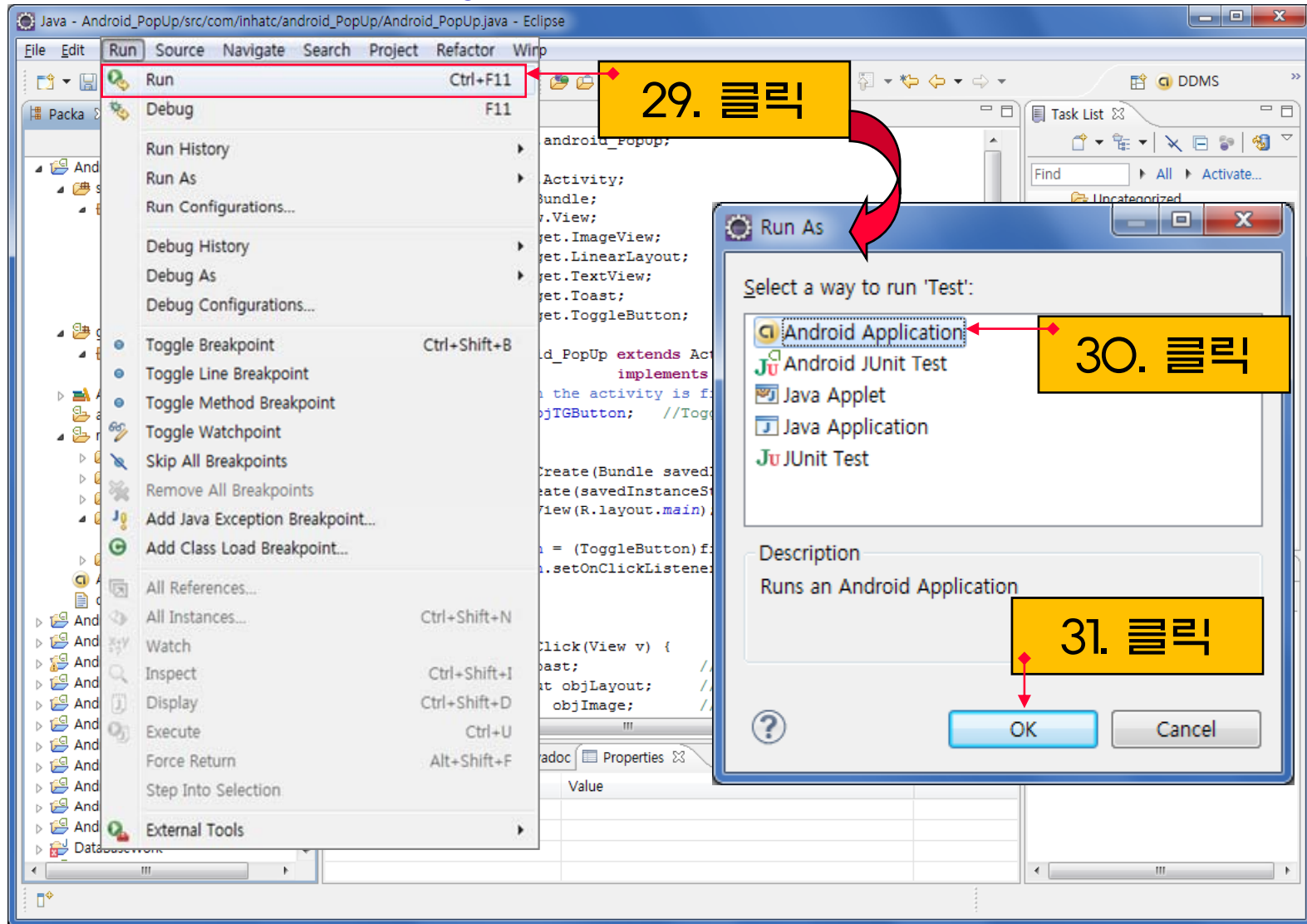
### 28. Coding

```
*Android_PopUp.java X
25
26 @Override
27 public void onClick(View v) {
28     Toast objToast;           //Toast 객체 선언
29     LinearLayout objLayout;    //LinearLayout 객체 선언
30     ImageView objImage;       //ImageView 객체 선언
31     TextView objTV;           //TextView 객체 선언
32
33     if(objTGBButton.isChecked()){
34         objToast = new Toast(this);
35         objLayout = new LinearLayout(this);
36         objLayout.setOrientation(LinearLayout.VERTICAL);
37
38         objImage = new ImageView(this);
39         objImage.setImageResource(R.drawable.light);
40         objLayout.addView(objImage);
41
42         objTV = new TextView(this);
43         objTV.setText("Turn-On Lamp");
44         objLayout.addView(objTV);
45
46         objToast.setView(objLayout);
47         objToast.setDuration(Toast.LENGTH_LONG);
48         objToast.show();
49     }else{
50         objToast = Toast.makeText(this, "Turn-Off Lamp", Toast.LENGTH_SHORT);
51         objToast.show();
52     }
53 }
54 }
```



# Toast 구현 (18)

## ❖ Android 프로젝트 실행





# Toast 구현 (19)

## ❖ 실행 결과





# Alert Dialog (1)

## ■ Alert Dialog

### ❖ 기능

◆ 사용자의 중요 메시지 화면 출력 기능

### ❖ 특징

◆ Focus 받음 (Modal Dialog)

◆ 3개 Button 출력

### ❖ 생성자

Method
AlertDialog( <a href="#">Context</a> context)
AlertDialog( <a href="#">Context</a> context, <a href="#">int</a> theme)
AlertDialog( <a href="#">Context</a> context, <a href="#">boolean</a> cancelable, <a href="#">DialogInterface.OnCancelListener</a> cancelListener)

```
public class  
AlertDialog  
extends Dialog  
implements DialogInterface
```

```
java.lang.Object  
└ android.app.Dialog  
  └ android.app.AlertDialog  
    ▶ Known Direct Subclasses  
      DatePickerDialog, ProgressDialog, TimePickerDialog
```







# Alert Dialog (2)

## ❖ Method

Method	설 명
<code>Button</code> <code>getButton(int whichButton)</code>	Dialog내 button 반환
<code>ListView</code> <code>getListView()</code>	Dialog내 ListView 반환
<code>void</code> <code>setButton(int whichButton, CharSequence text, Message msg)</code>	Dialog내 Button 클릭 시에 전송할 메시지 지정
<code>void</code> <code>setCustomTitle(view customTitleView)</code>	CustomTitle 지정
<code>void</code> <code>setTitle(CharSequence title)</code>	Title 지정
<code>void</code> <code>setText(int resId / CharSequence text)</code>	Text 지정
<code>void</code> <code>setView(View v)</code>	View 지정





# Alert Dialog (3)

## Alert Dialog 구조

❖ Icon

❖ Title

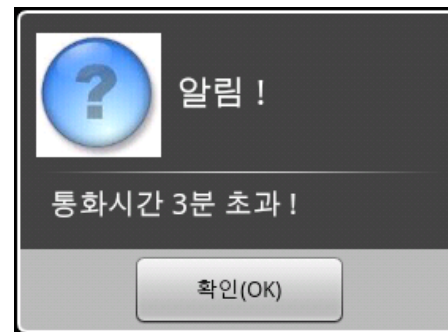
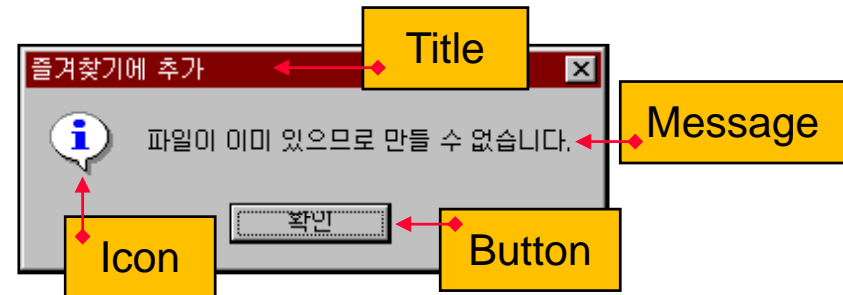
❖ Message

◆ Message / List 중에 선택 (동시 출력 불가)

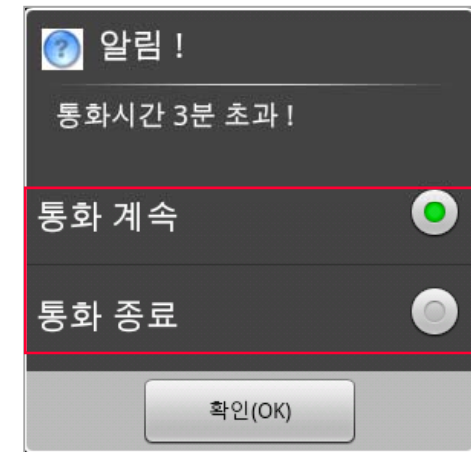
❖ CustomView

❖ Button

◆ 0~3개 가능



〈CustomView 미구현〉



〈CustomView 구현〉



# Alert Dialog (4)

## ■ Alert Dialog 생성

```
AlertDialog dlgAlert;  
dlgAlert = new AlertDialog.Builder(this)  
    .setIcon(R.drawable.icon)  
    .setTitle("타이틀 영역") // Title 지정  
    .setMessage("메시지 영역") // Message 지정  
    .setView(createCustomView()) // CustomView 지정  
    // AlertDialog의 버튼 클릭 event에 대한 Listener 객체 생성/등록  
    .setPositiveButton("버튼 영역", new DialogInterface.OnClickListener(){  
        public void onClick(DialogInterface dialog, int which) {  
            dialog.dismiss();  
        }  
    })  
    .create();
```

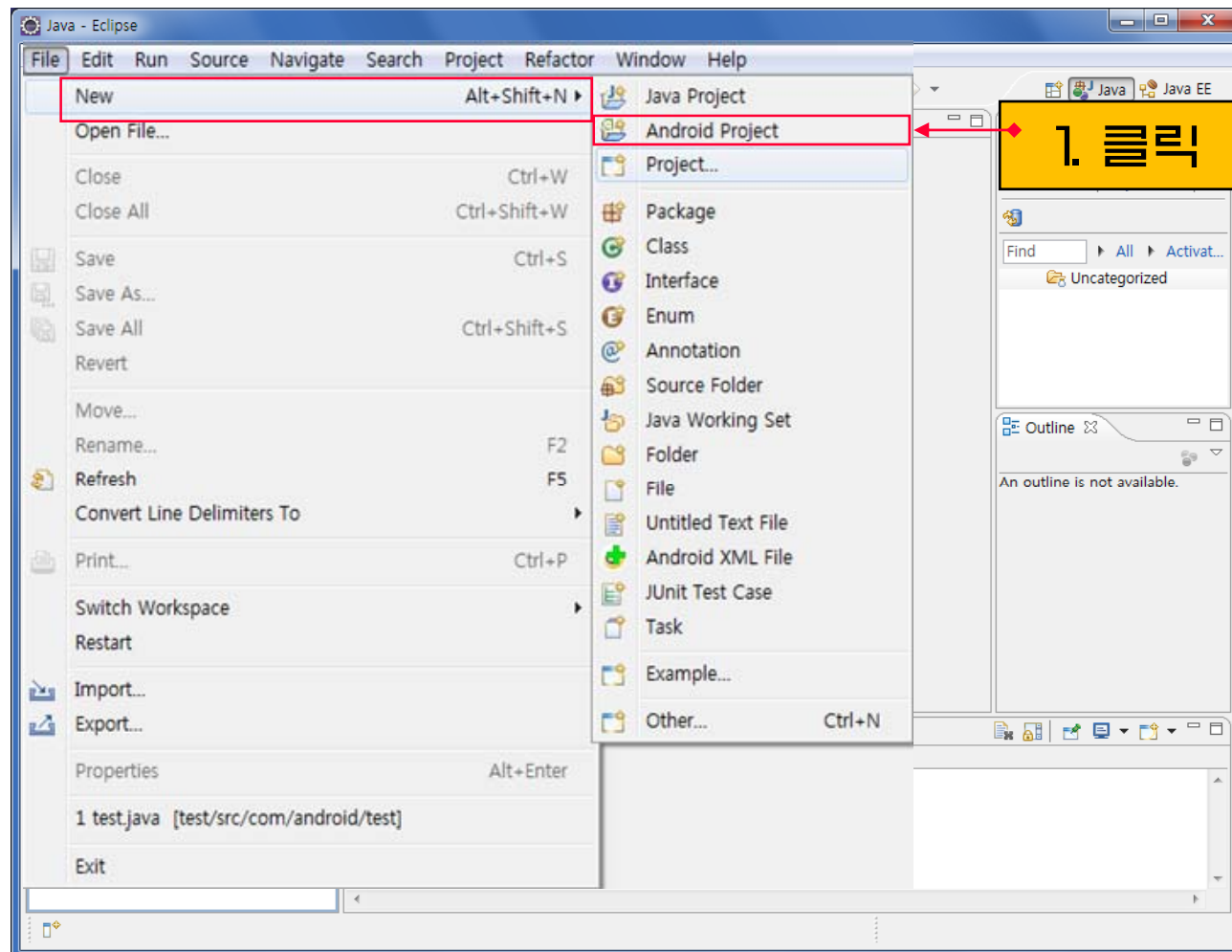




# AlertDialog 구현 (1)

## ■ Android 프로젝트 생성

❖ 프로젝트 명 : Android\_AlertDialog





# AlertDialog 구현 (2)

New Android Project

Creates a new Android Project resource.

Project name:

Contents

☒ Create new project in workspace  
☐ Create project from existing source  
☒ Use default location

Location:

☐ Create project from existing sample

Samples:

Build Target

Target Name	Vendor	Platform	API ...
<input type="checkbox"/> Android 1.1	Android Open Source Project	1.1	2
<input type="checkbox"/> Android 2.0.1	Android Open Source Project	2.0.1	6
<input checked="" type="checkbox"/> Android 2.1	Android Open Source Project	2.1	7
<input type="checkbox"/> Google APIs	Google Inc.	2.0.1	6
<input type="checkbox"/> Google APIs	Google Inc.	2.1	7

Standard Android platform 2.1

Properties

Application name:

Package name:

☒ Create Activity:

Min SDK Version:

2. Android\_AlertDialog 입력

3. 클릭

4. Android\_AlertDialog 입력

5. com.inhatc.android\_AlertDialog 입력

6. Android\_AlertDialog 입력

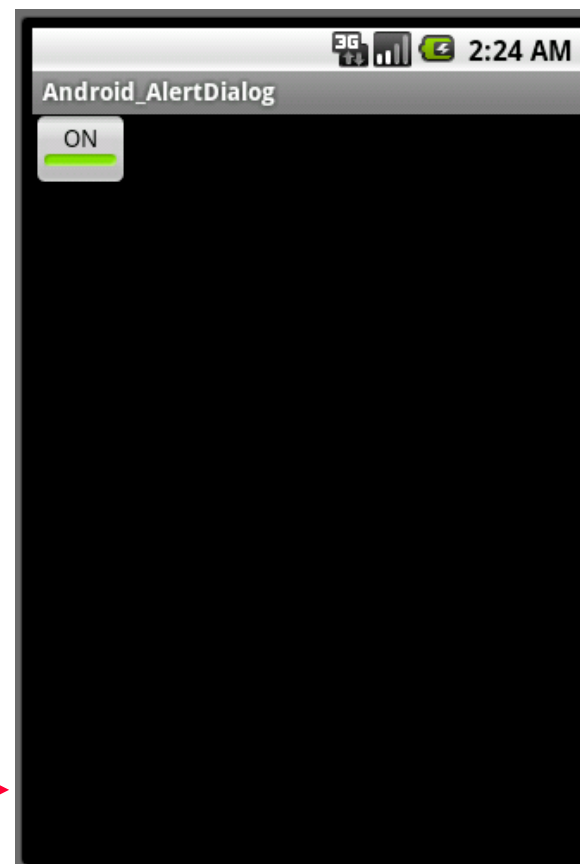
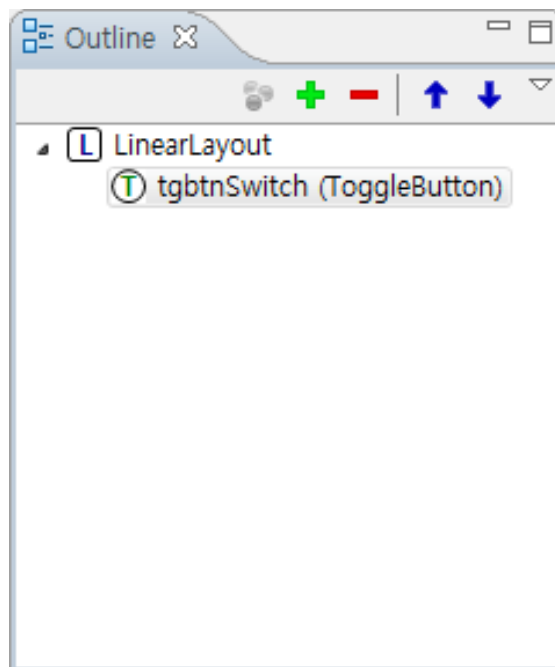
7. 클릭





# AlertDialog 구현 (3)

## ■ UI 설계



8. UI 설계 및 속성 지정



# AlertDialog 구현 (4)

## ❖ Text View 삭제

Java - Android\_AlertDialog/res/layout/main.xml - Eclipse

File Edit Run Navigate Search Project Refactor Window Help

main.xml

Editing config: default

Devices ADP1 Config Landscap Locale Theme Create...

Layouts

- A AbsoluteLayout
- D DialerFilter
- E ExpandableList...
- F FrameLayout
- G GridView
- H HorizontalScro...
- I ImageSwitcher
- L LinearLayout
- L ListView

Views

- G GestureOverlay...
- S SurfaceView
- V View

Task List

Find All Activate...

Uncategorized

Outline

- L LinearLayout
  - 1 TextView

Remove element from Android XML

Do you really want to remove - LinearLayout > TextView?

Yes No

10. 클릭

9. 클릭

11. 클릭



# AlertDialog 구현 (5)

## ❖ ToggleButton 추가

13. 클릭

12. 클릭

14. 클릭

Create a new element at the top level, in LinearLayout.

- TabWidget
- TextSwitcher
- TextView
- TimePicker
- ToggleButton**
- TwoLineListItem
- VideoView
- View
- ViewAnimator
- ViewFlipper
- ViewStub
- ViewSwitcher
- ZoomButton
- ZoomControls

android.wid...oggleButton

OK Cancel

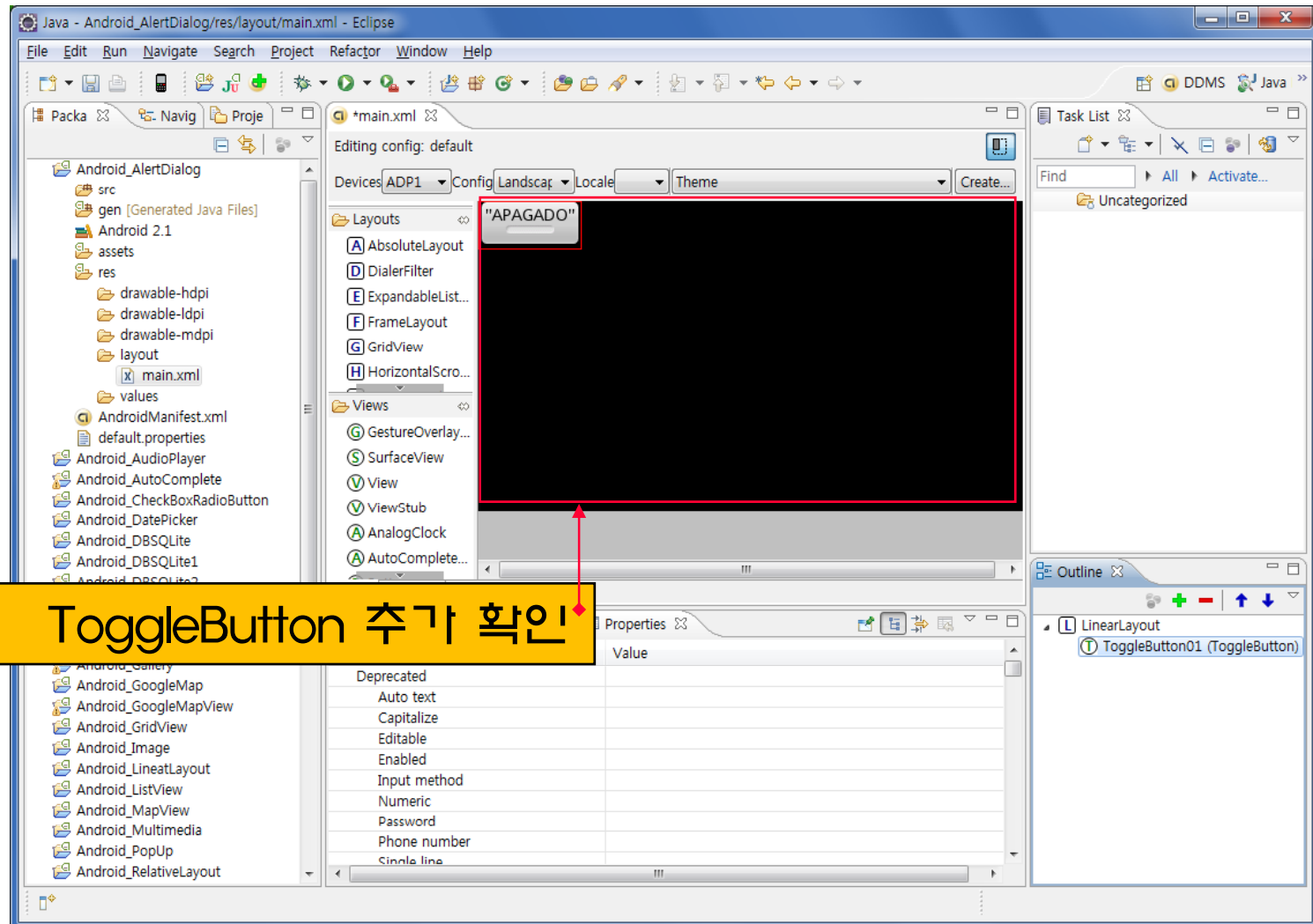
Outline: LinearLayout





# AlertDialog 구현 (6)

## ❖ ToggleButton 추가 결과





# AlertDialog 구현 (7)

## ❖ ToggleButton 속성 지정

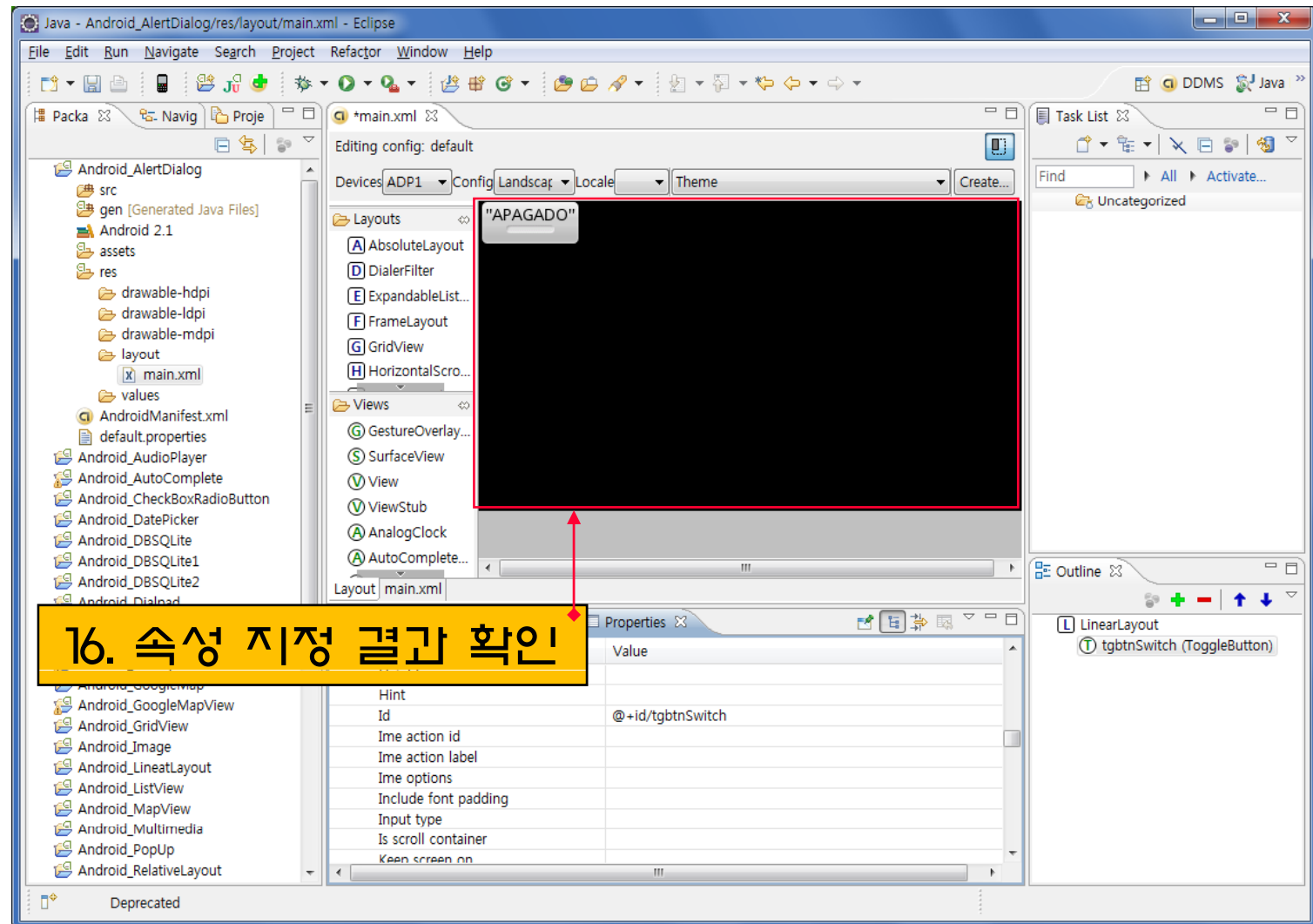
컨트롤	속성 지정
ToggleButton	<ul style="list-style-type: none"><li>• android:layout_width="wrap_content"</li><li>• android:layout_height="wrap_content"</li><li>• android:id="@+id/tgbtnSwitch"</li><li>• android:text="Switch"</li></ul>





# AlertDialog 구현 (8)

## ❖ ToggleButton 속성 지정 결과





# AlertDialog 구현 (9)

## ❖ Main.xml 저장

17. 클릭

18. 클릭

Save Resource

'main.xml' has been modified. Save changes?

Yes No Cancel

Property Value

Property	Value
Height	
Hint	
Id	@+id/tgbtnSwitch
Ime action id	
Ime action label	
Ime options	
Include font padding	
Input type	
Is scroll container	
Keep screen on	

Layout main.xml

Problems Console Javadoc Properties

Outline

LinearLayout

tgbtnSwitch (ToggleButton)



# AlertDialog 구현 (10)

## ❖ Main.xml

```
*main.xml x
1<?xml version="1.0" encoding="utf-8"?>
2<LinearLayout
3    xmlns:android="http://schemas.android.com/apk/res/android"
4    android:orientation="vertical"
5    android:layout_width="fill_parent"
6    android:layout_height="fill_parent"
7    >
8    <ToggleButton
9        android:layout_width="wrap_content"
10       android:layout_height="wrap_content"
11       android:text="Switch"
12       android:id="@+id/tgbtnSwitch">
13    </ToggleButton>
14</LinearLayout>
```

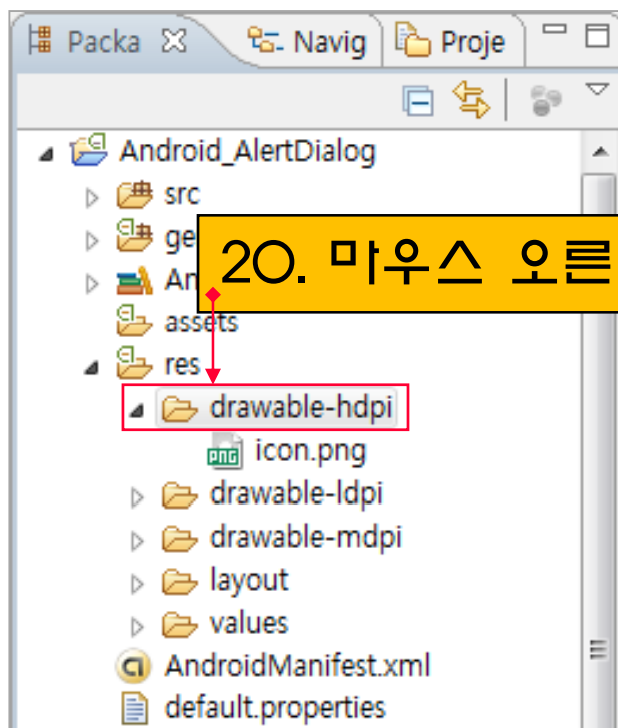
19. XML code 추가 확인



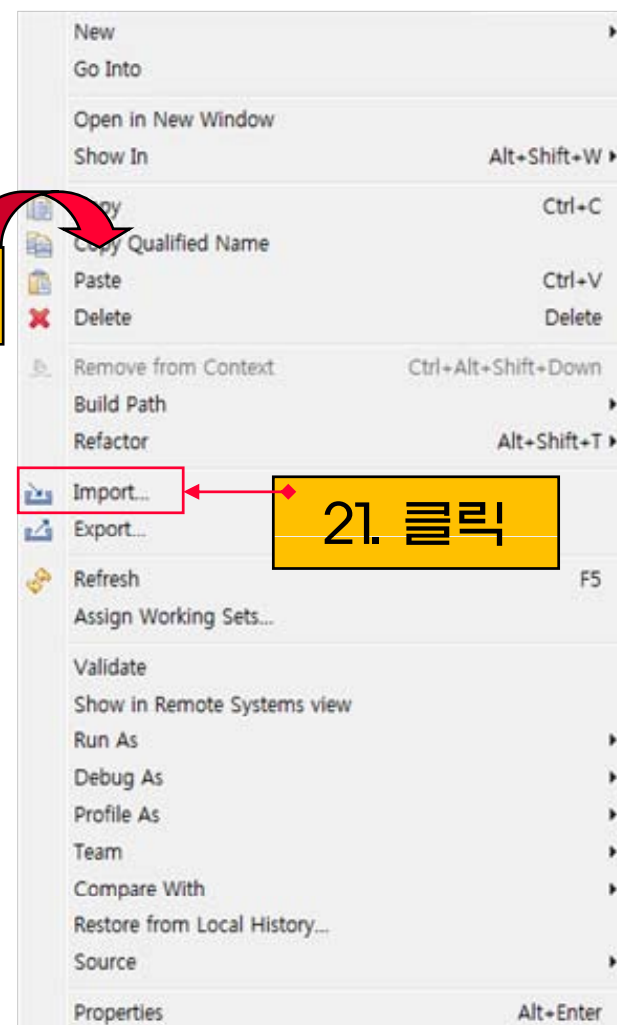


# AlertDialog 구현 (11)

## ❖ Image import



20. 마우스 오른쪽 버튼 클릭

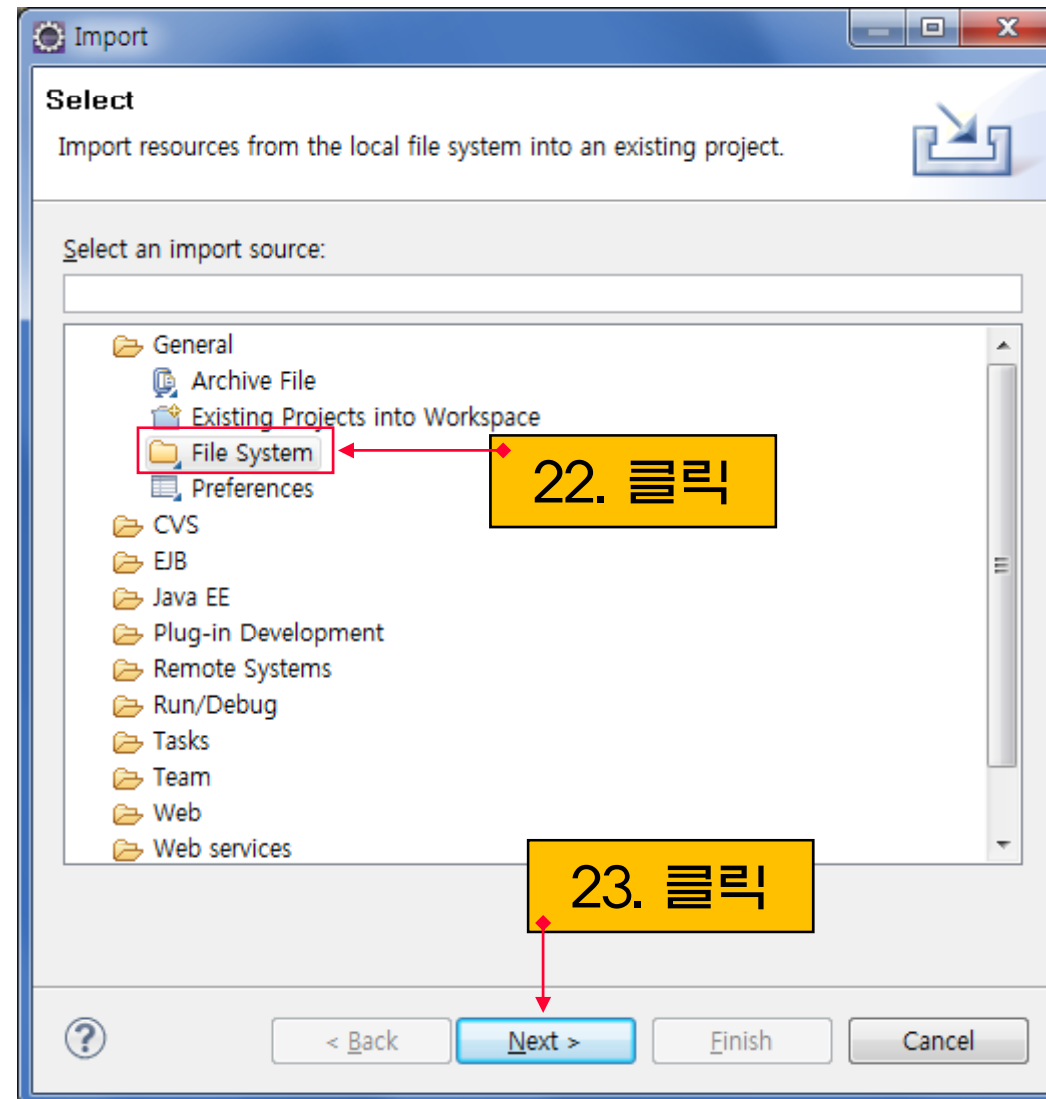


21. 클릭



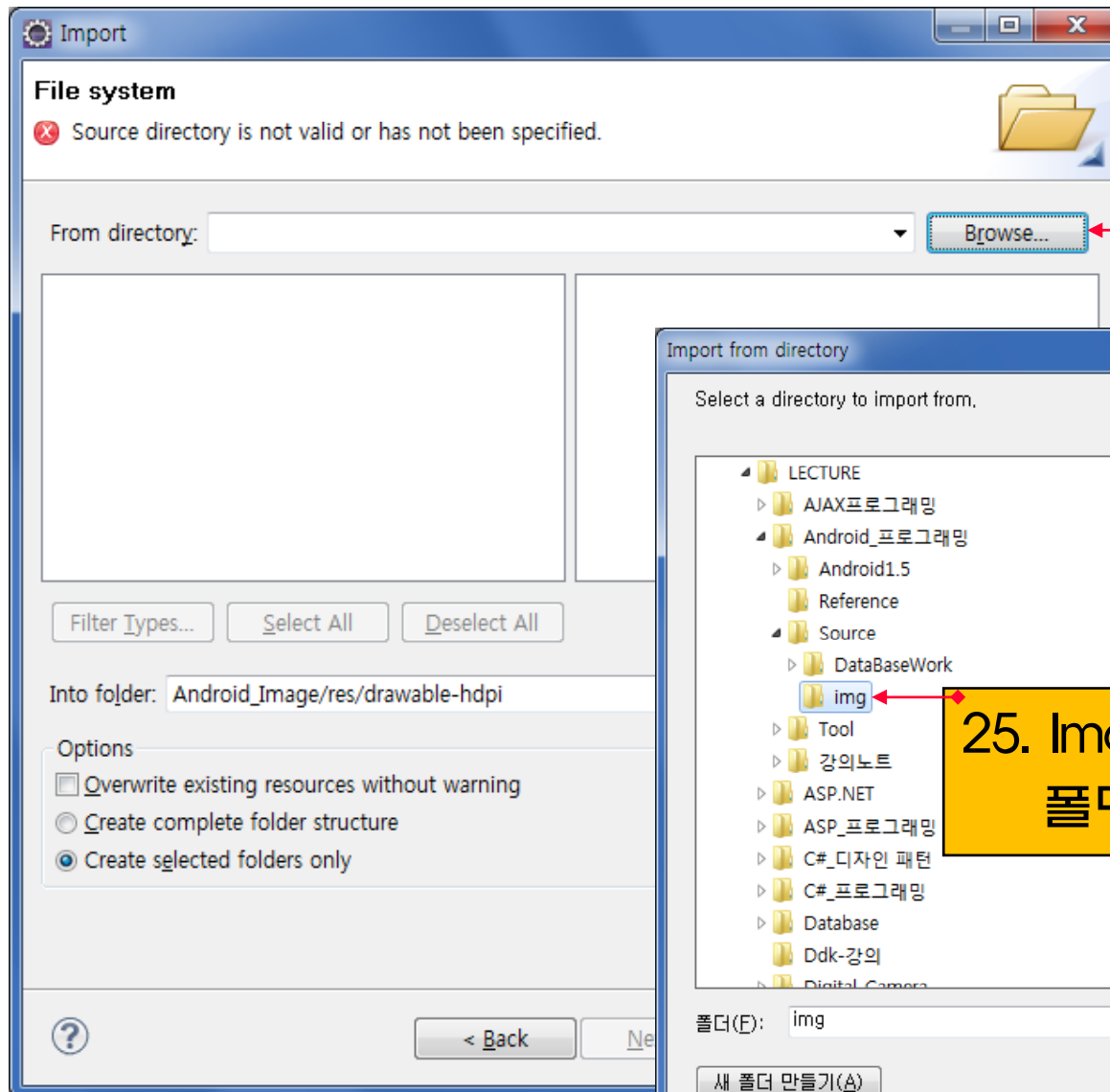


# AlertDialog 구현 (12)

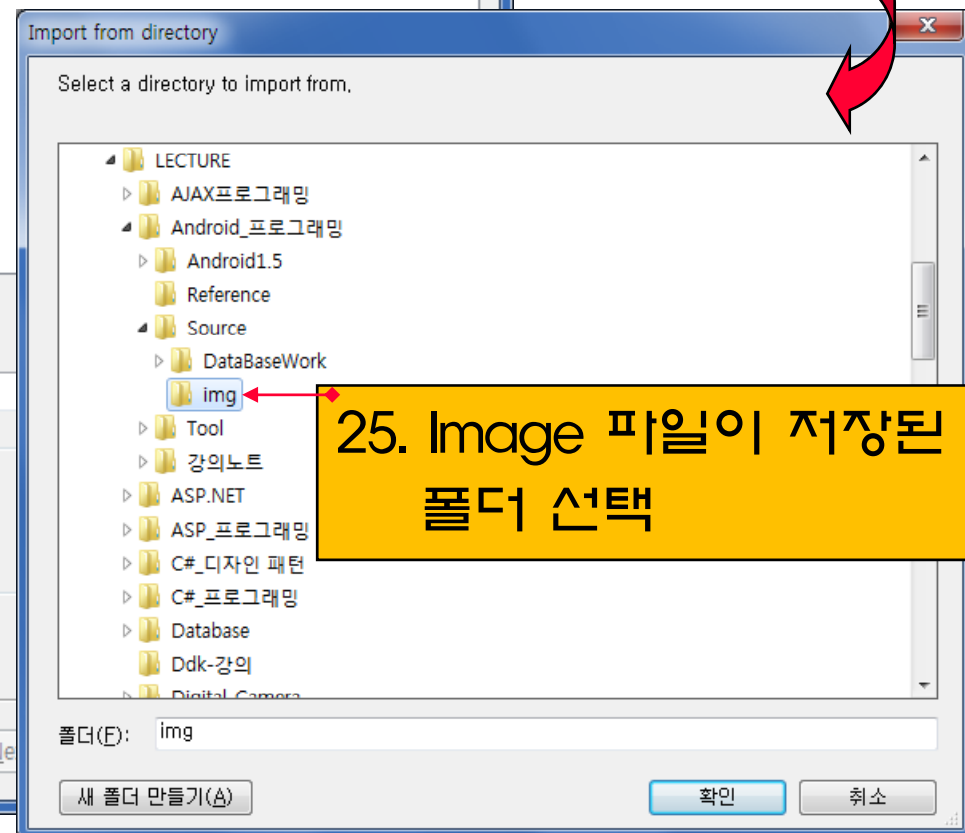




# AlertDialog 구현 (13)



24. 클릭



25. Image 파일이 저장된  
폴더 선택



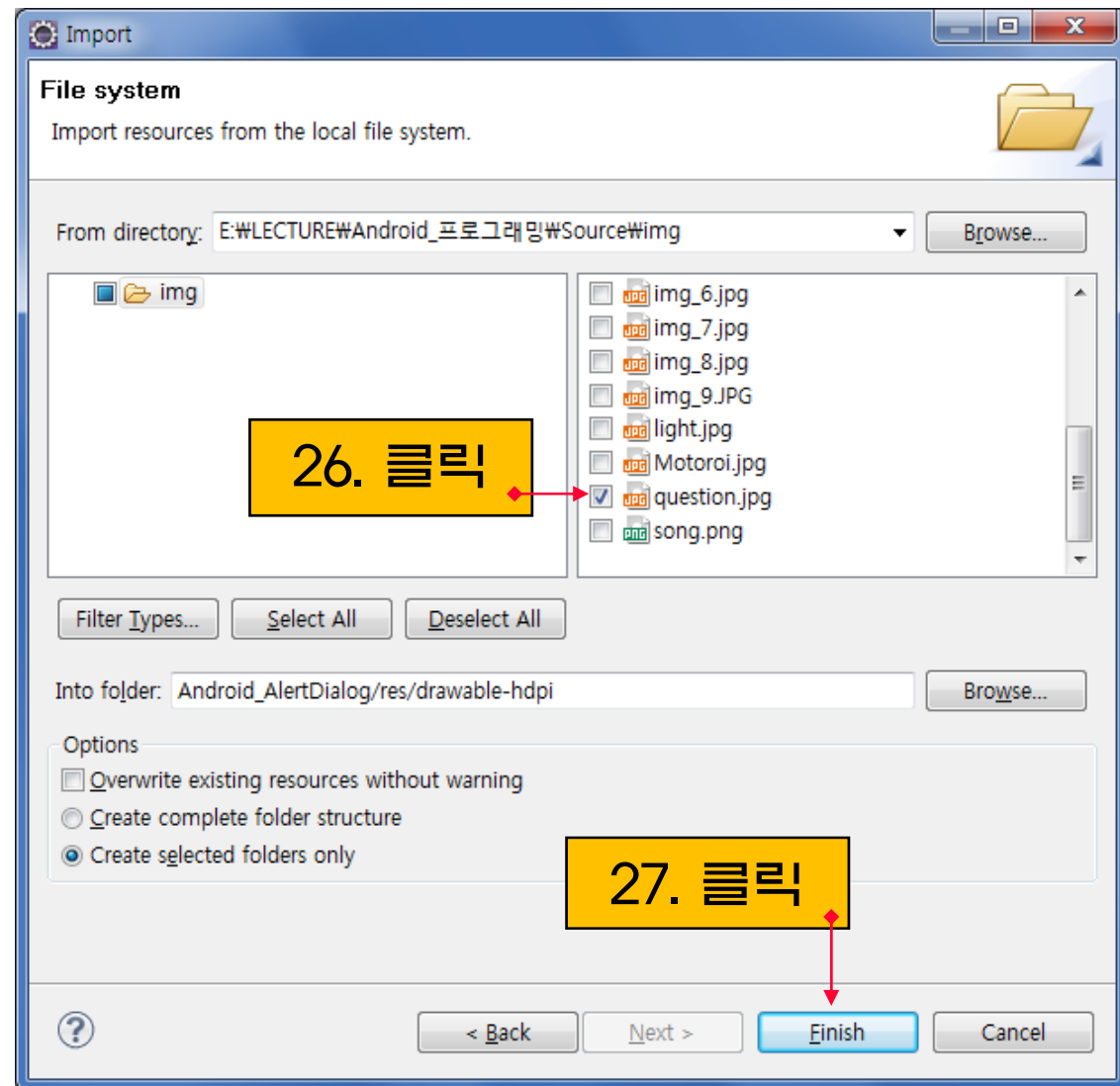




# AlertDialog 구현 (14)

❖ Image 파일 선택

◆ question.jpg

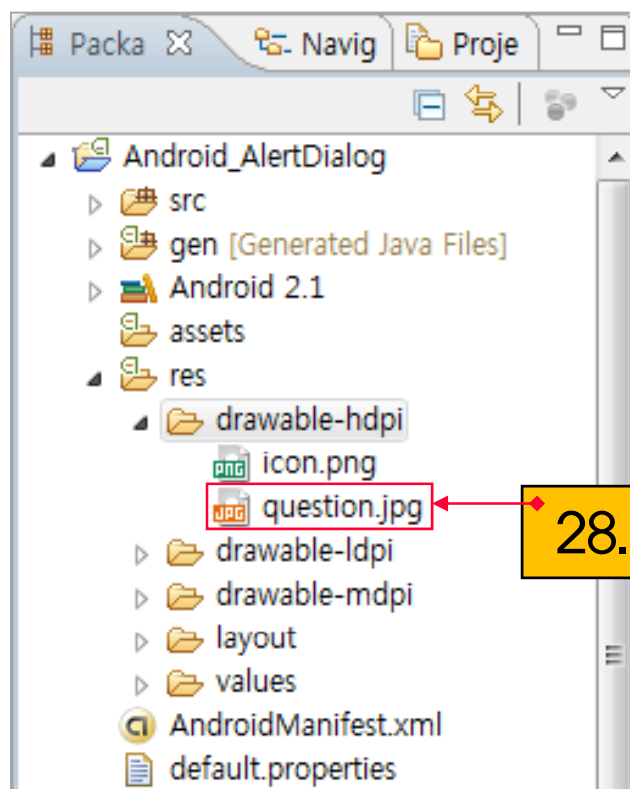




# AlertDialog 구현 (15)

❖ Image 파일 import 확인

◆ question.jpg



28. Image 파일 import 확인





# AlertDialog 구현 (16)

❖ R.java

```
R.java X
1  /* AUTO-GENERATED FILE.  DO NOT MODIFY.
7
8  package com.inhatc.android_AlertDialog;
9
10 public final class R {
11     public static final class attr {
12     }
13     public static final class drawable {
14         public static final int icon=0x7f020000;
15         public static final int question=0x7f020001;
16     }
17     public static final class id {
18         public static final int tgbtnSwitch=0x7f050000;
19     }
20     public static final class layout {
21         public static final int main=0x7f030000;
22     }
23     public static final class string {
24         public static final int app_name=0x7f040001;
25         public static final int hello=0x7f040000;
26     }
27 }
```

29. drawable, id 클래스 변수 추가 확인





# AlertDialog 구현 (17)

## ❖ onClickListener() Method 구현

```
Android_AlertDialog.java
1 package com.inhatec.android_AlertDialog;
2
3 import android.app.Activity;
4 import android.app.AlertDialog;
5 import android.app.Dialog;
6 import android.content.DialogInterface;
7 import android.os.Bundle;
8 import android.view.View;
9 import android.widget.ToggleButton;
10
11 public class Android_AlertDialog extends Activity
12     implements View.OnClickListener {
13     /** Called when the activity is first created. */
14     ToggleButton objTGBButton; //ToggleButton 객체 선언
15
16     @Override
17     public void onCreate(Bundle savedInstanceState) {
18         super.onCreate(savedInstanceState);
19         setContentView(R.layout.main);
20
21         objTGBButton = (ToggleButton) findViewById(R.id.tgbtnSwitch);
22         objTGBButton.setOnClickListener(this);
23     }
24
25     @Override
26     public void onClick(View arg0) {
27         // onCreateDialog(int), onPrepareDialog(int) 호출
28         showDialog(0);
29     }
30 }
```

30. Coding



# AlertDialog 구현 (18)

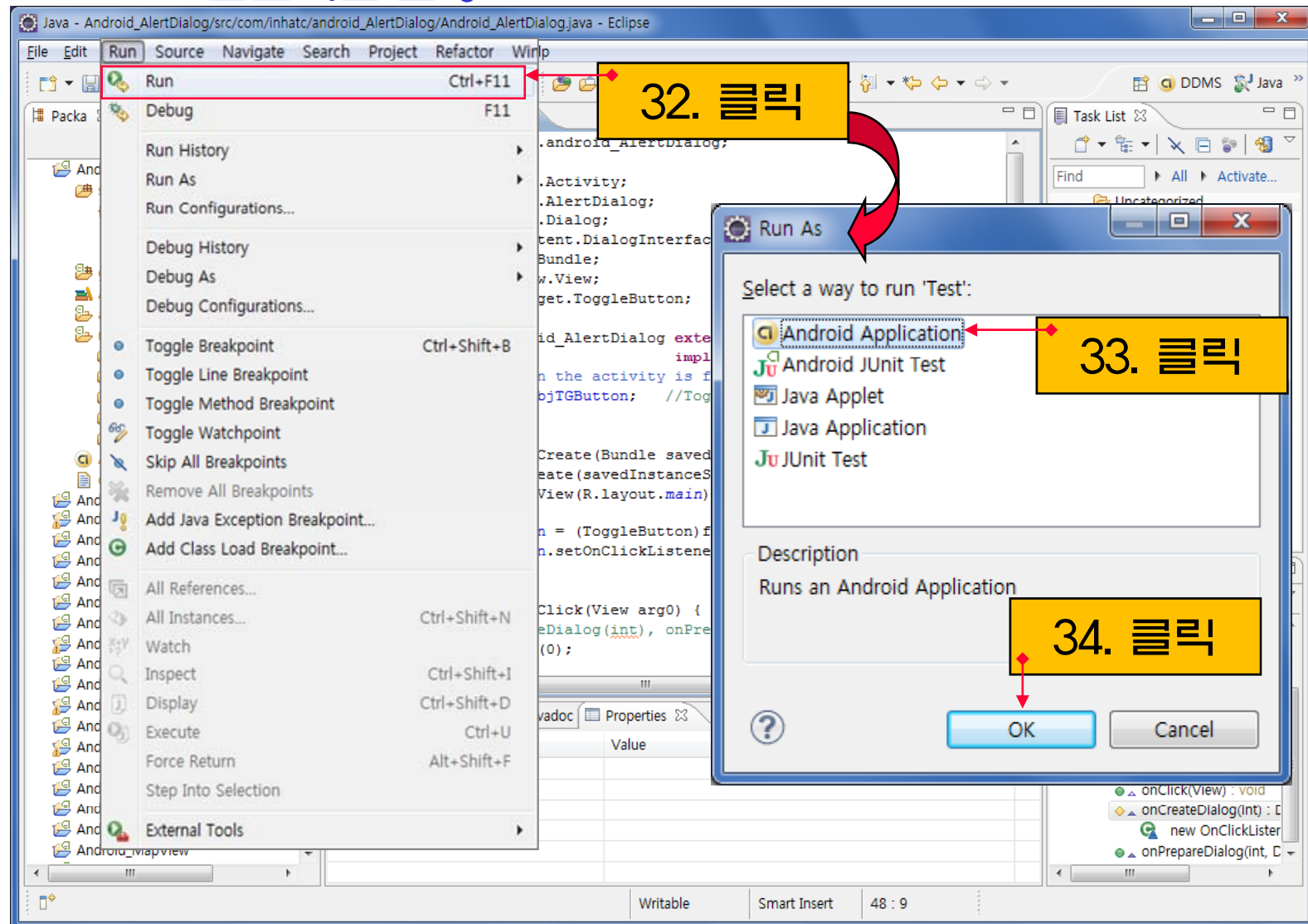
## ❖ onCreateDialog() / onPrepareDialog() Method 구현

```
*Android_AlertDialog.java
29
30
31 @Override // AlertDialog 객체 생성시 한번만 호출
protected Dialog onCreateDialog(int id) {
32     super.onCreateDialog(id);
33     AlertDialog dlgAlert; // AlertDialog 객체 선언
34
35     dlgAlert = new AlertDialog.Builder(this)
36         .setIcon(R.drawable.question)
37         .setTitle("알림 !") // Title 지정
38         .setMessage("통화시간 3분 초과 !") // Message 지정
39         .setView(null) // CustomView 지정
40
41     // AlertDialog의 버튼 클릭 event에 대한 Listener 객체 생성/등록
42     .setPositiveButton("확인 (OK)", new DialogInterface.OnClickListener() {
43         public void onClick(DialogInterface dialog, int which) {
44             dialog.dismiss();
45         }
46     })
47     .create(); //dlgAlert에 AlertDialog return
48
49     return dlgAlert;
50 }
51
52 @Override // AlertDialog 객체 화면 출력시 마다 항상 호출
53 public void onPrepareDialog(int id, Dialog dlg) {
54     super.onPrepareDialog(id, dlg);
55 }
56 }
```



# AlertDialog 구현 (19)

## ❖ Android 프로젝트 실행





# AlertDialog 구현 (20)

## ❖ 실행 결과







# 실습 I : AlertDialog에 CustomView 구현

## ■ Android\_AlertDialog (실습 시간 : 30분)

❖ 아래 그림과 같이 AlertDialog의 CustomView를 ListView로 구현 하시오.

◆ CustomView에 ListView 구현







# AlertDialog에 CustomView : ListView 구현 (1)

```
Android_AlertDialog.java
37 @Override // AlertDialog 객체 생성시 한번만 호출
38 protected Dialog onCreateDialog(int id) {
39     super.onCreateDialog(id);
40     AlertDialog dlgAlert; // AlertDialog 객체 선언
41
42     dlgAlert = new AlertDialog.Builder(this)
43         .setIcon(R.drawable.question)
44         .setTitle("알림 !") // Title 지정
45         .setMessage("통화시간 3분 초과 !") // Message 지정
46         .setView(createCustomView()) // CustomView 지정
47
48     // AlertDialog의 버튼 클릭 event에 대한 Listener 객체 생성/등록
49     .setPositiveButton("확인 (OK)", new DialogInterface.OnClickListener() {
50         public void onClick(DialogInterface dialog, int which) {
51             dialog.dismiss();
52         }
53     })
54     .create(); //dlgAlert에 AlertDialog return
55
56     return dlgAlert;
57 }
58
59 @Override // AlertDialog 객체 화면 출력시 마다 항상 호출
60 public void onPrepareDialog(int id, Dialog dlg) {
61     super.onPrepareDialog(id, dlg);
62 }
```

Coding





## AlertDialog에 CustomView : ListView 구현 (2)

```
*Android_AlertDialog.java 33

64 // AlertDialog의 CustomView 생성
65 private View createCustomView() {
66     LinearLayout objLayoutView;           //LinearLayout 객체 선언
67     ListView lstViewCalling;              //ListView 객체 선언
68     ArrayList<String> lstItems;           //ArrayList 객체선언
69     ArrayAdapter<String> aryADTItems;     //ArrayAdapter 객체선언
70
71     objLayoutView = new LinearLayout(this);
72     lstViewCalling = new ListView(this);
73     lstItems = new ArrayList<String>();
74     lstItems.add("통화 계속");
75     lstItems.add("통화 종료");
76
77     aryADTItems = new ArrayAdapter<String> (this,
78         android.R.layout.simple_list_item_single_choice, lstItems);
79
80     lstViewCalling.setAdapter(aryADTItems);
81     lstViewCalling.setChoiceMode(ListView.CHOICE_MODE_SINGLE);
82     lstViewCalling.setOnItemClickListener(new AdapterView.OnItemClickListener() {
83         public void onItemClick(AdapterView<?> parent, View view, int index, long id) {
84             String strData = null;
85             if (index == 0) strData = "통화중.....";
86             else strData = "통화를 종료합니다.";
87             Toast.makeText(Android_AlertDialog.this, strData, Toast.LENGTH_SHORT).show();
88         }
89     });
90     objLayoutView.setOrientation(LinearLayout.VERTICAL);
91     objLayoutView.addView(lstViewCalling);
92     return objLayoutView;
93 }
94 }
```

Coding



## 실습 II : AlertDialog에 CustomView 구현

### ■ Android\_AlertDialog (실습 시간 : 30분)

❖ 아래 그림과 같이 AlertDialog의 CustomView를 RadioButton으로 구현하시오.

◆ CustomView에 RadioButton 구현





# 학습 요약

■ Toast Class

■ AlertDialog

❖ CostumView 미구현

❖ CostumView 구현



open handset alliance

