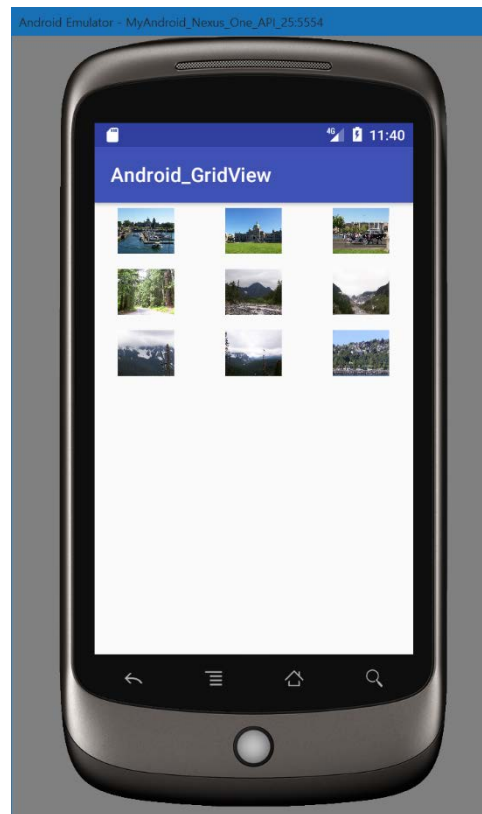




android



Android GridView 구현



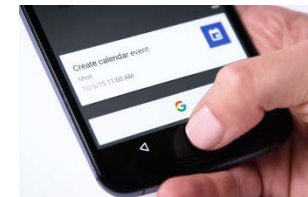
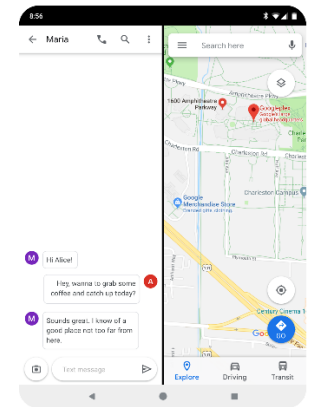
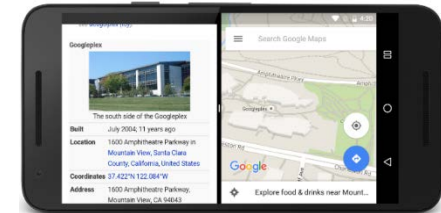


학습 목표

교육 목표

❖ GridView

❖ Gallery



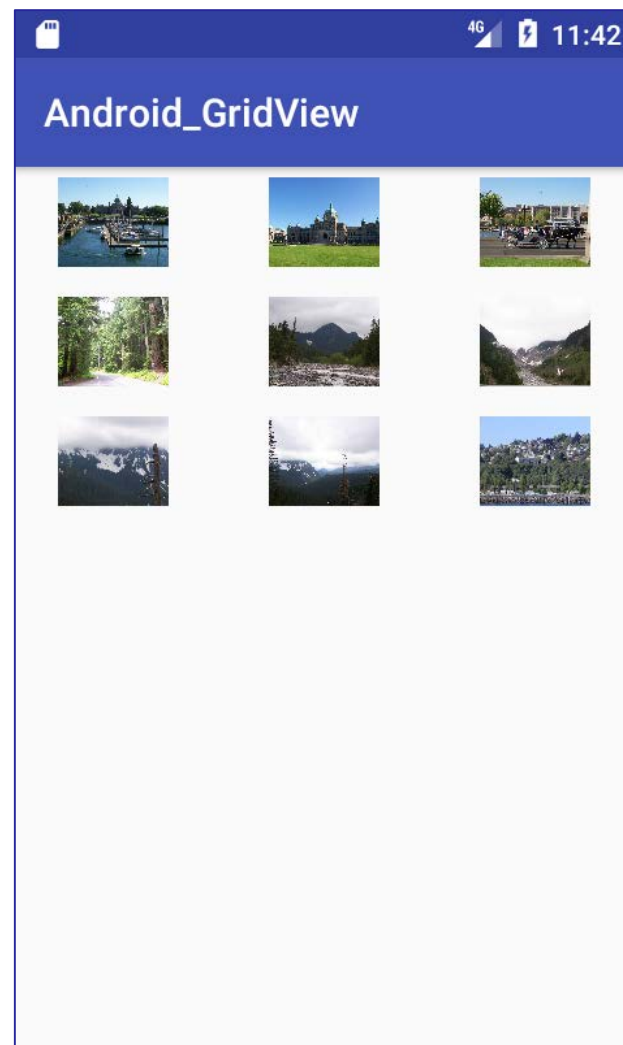


Grid View

■ Grid View

❖ 기능

- ◆ 눈금 구조를 보여주는 기능 제공
- ◆ item을 2차원 격자 형태로 출력

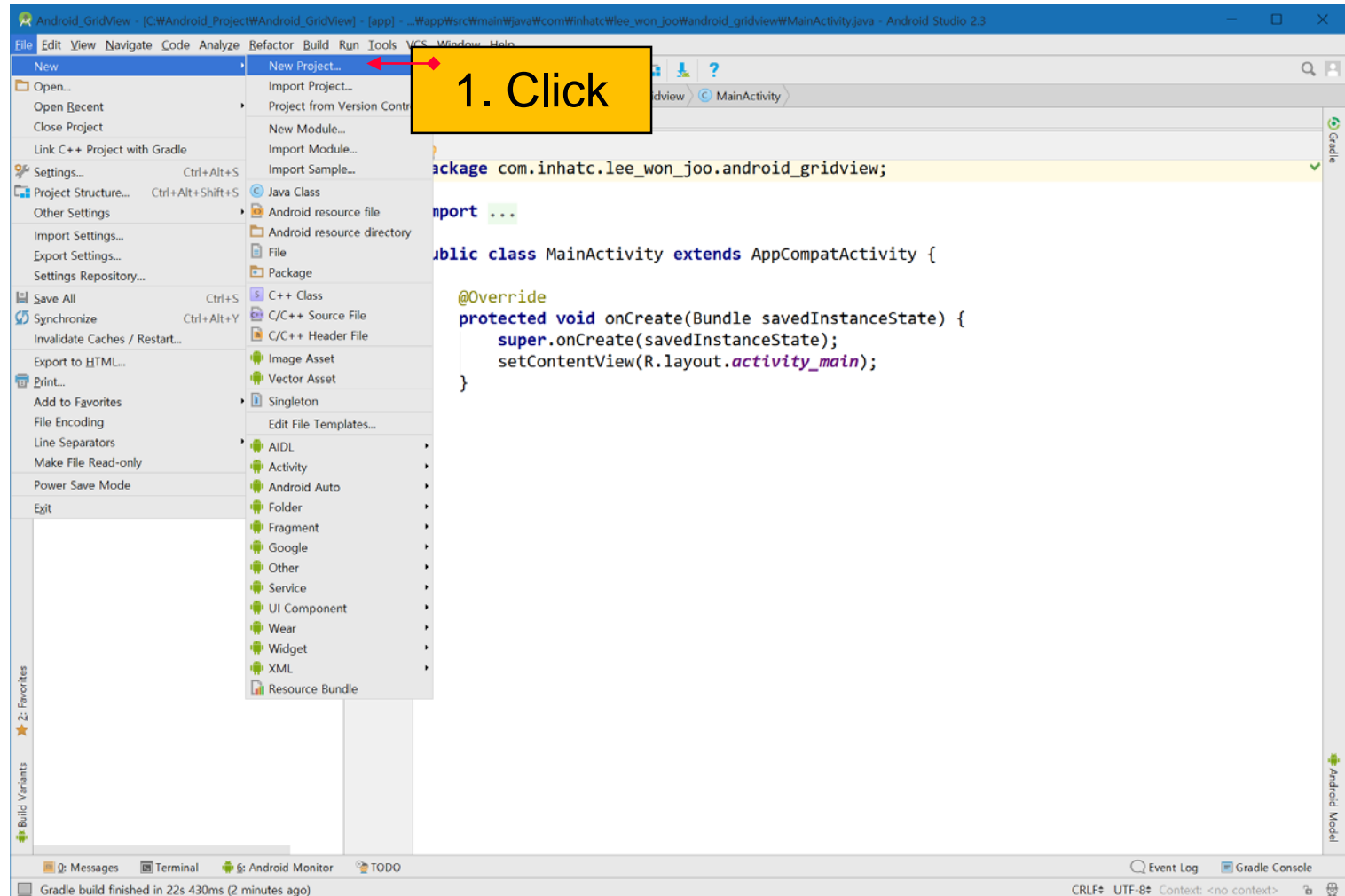




Grid View 구현 (1)

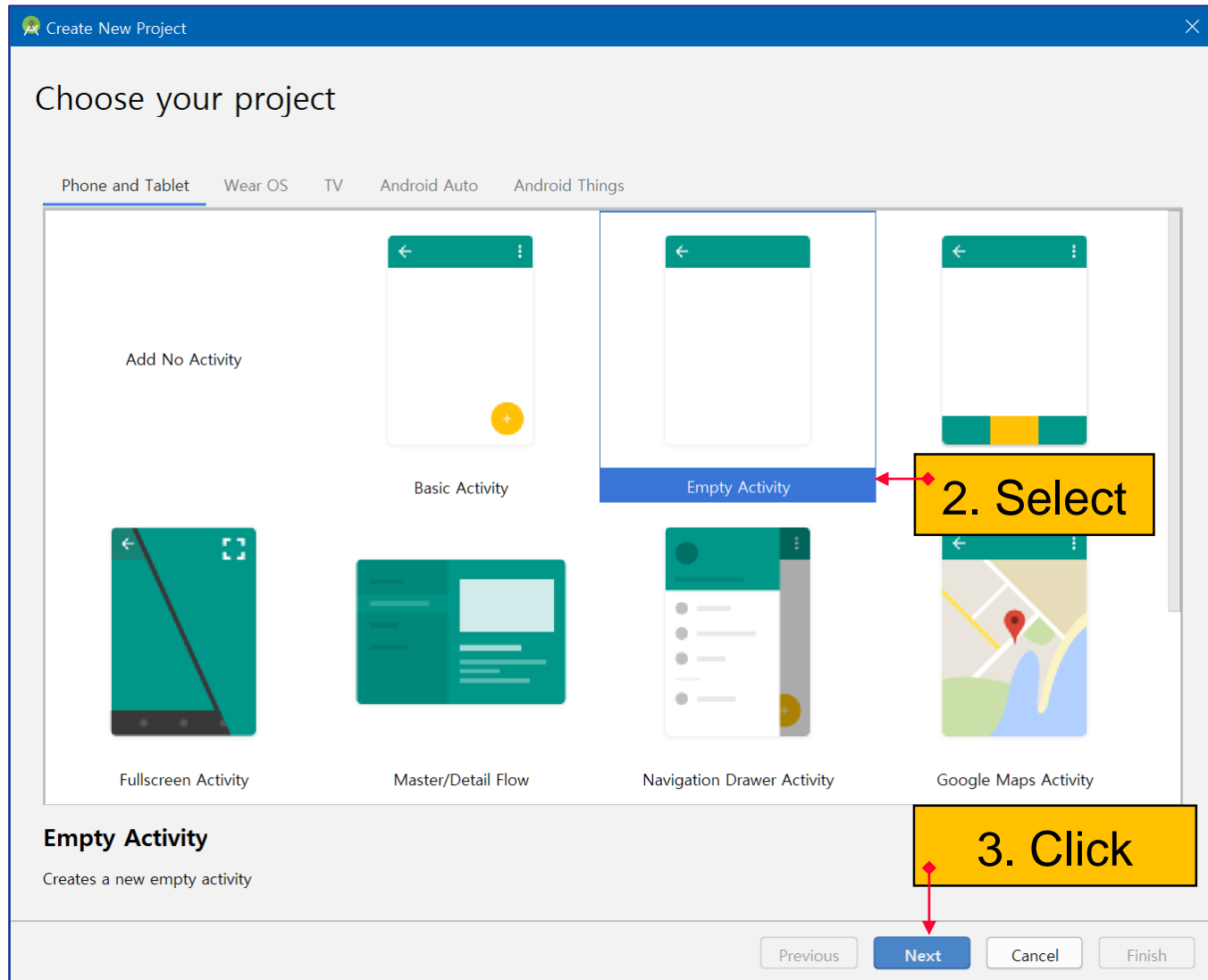
■ Android 프로젝트 생성

❖ 프로젝트 명 : Android_GridView





Grid View 구현 (2)





Grid View 구현 (3)

Create New Project

Configure your project

Name
Android_GridView

Package name
com.inhatc.android_gridview

Save location
C:\Android_Project\Android_GridView

Language
Java

Minimum API level
API 26: Android 8.0 (Oreo)

☐ This project will support instant apps

☐ Use AndroidX artifacts

Previous Next Cancel Finish

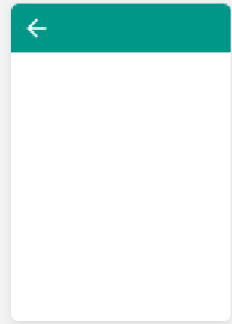
4. Input "Android_GridView"

5. Input "com.inhatc.android_gridview"

6. Select

7. Select

8. Click



Empty Activity

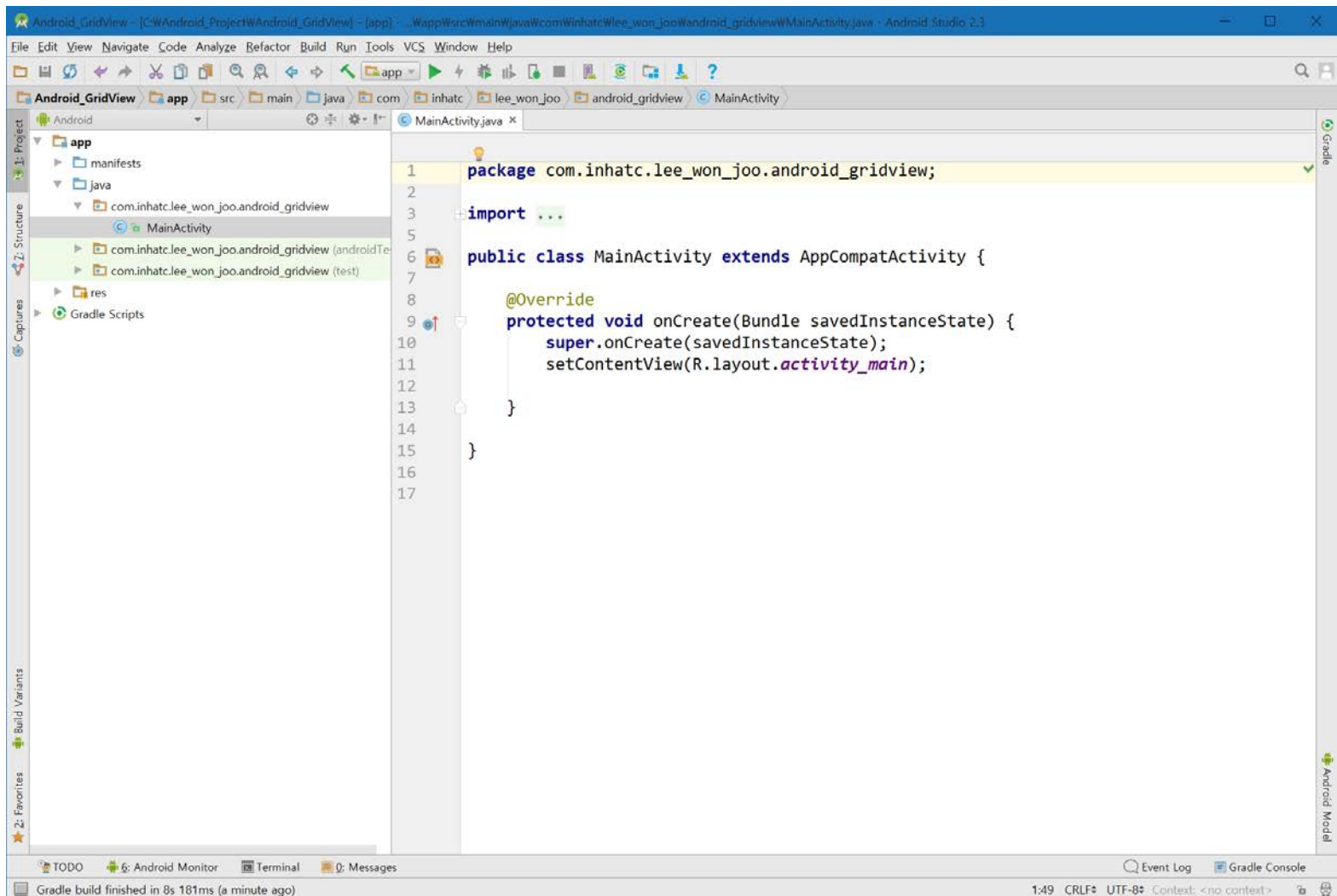
Creates a new empty activity





Grid View 구현 (4)

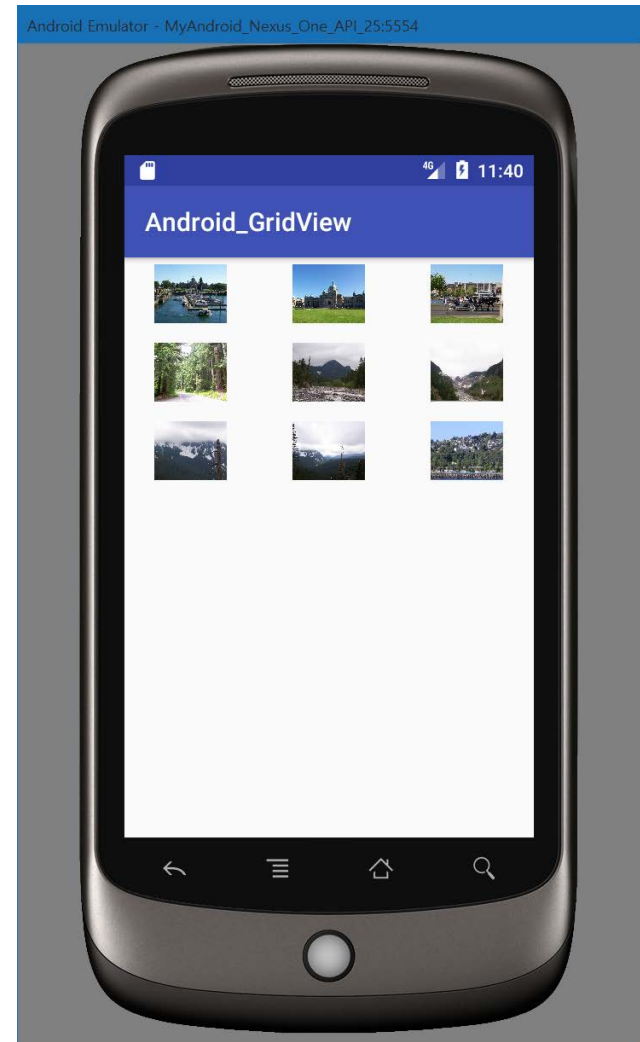
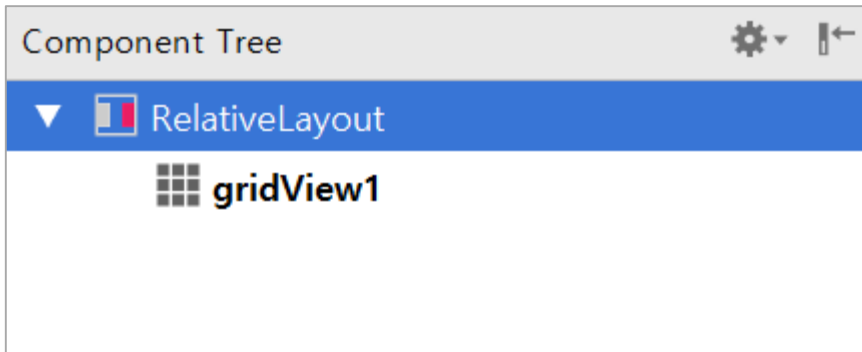
❖ Android 프로젝트 생성 완료





Grid View 구현 (5)

■ UI 설계





Grid View 구현 (6)

❖ TextView 삭제



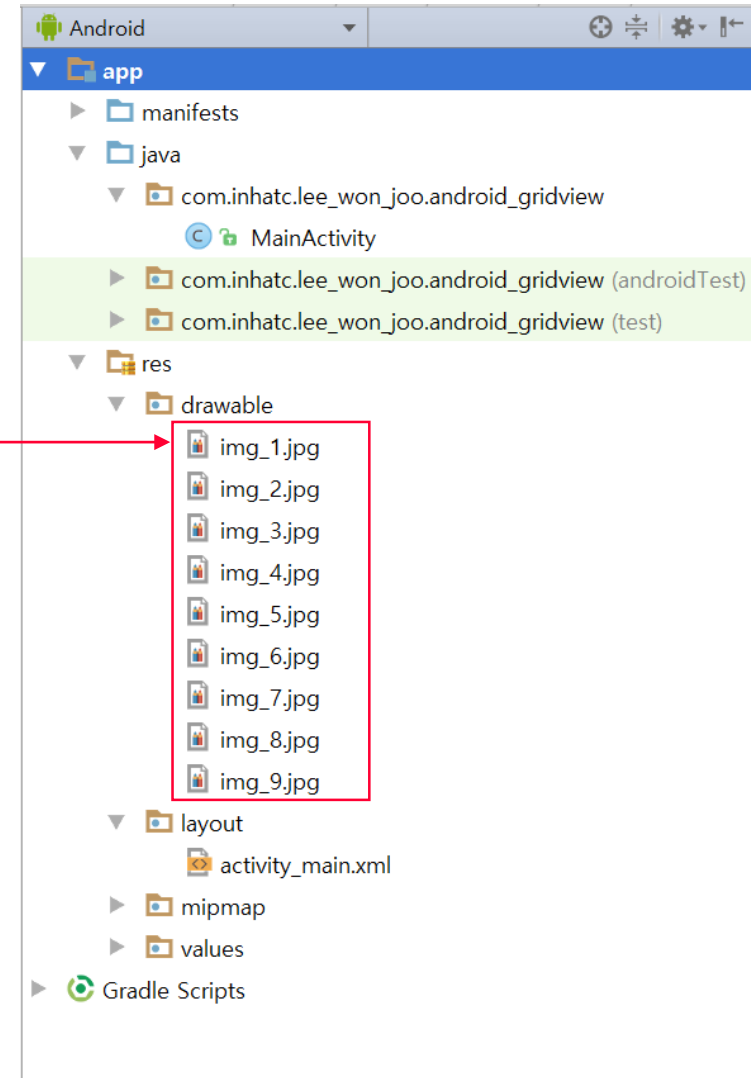
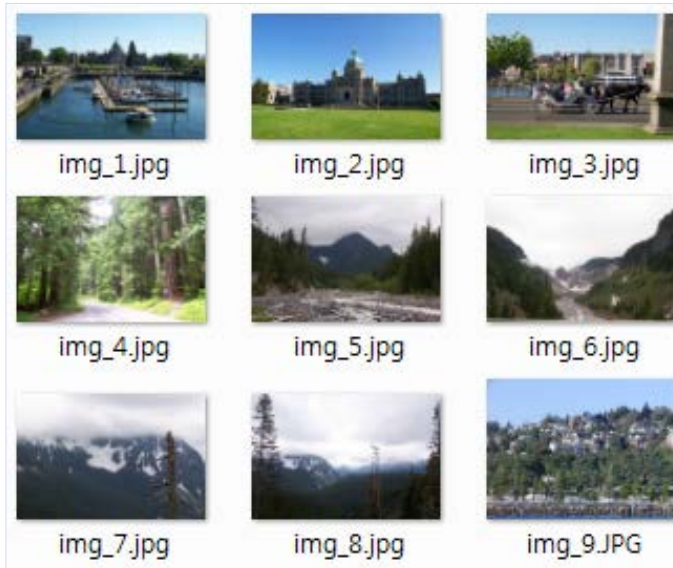


Grid View 구현 (7)

■ Image 복사

❖ .\res\drawable 폴더

11. image 복사





Grid View 구현 (8)

■ 속성 지정

❖ GridView

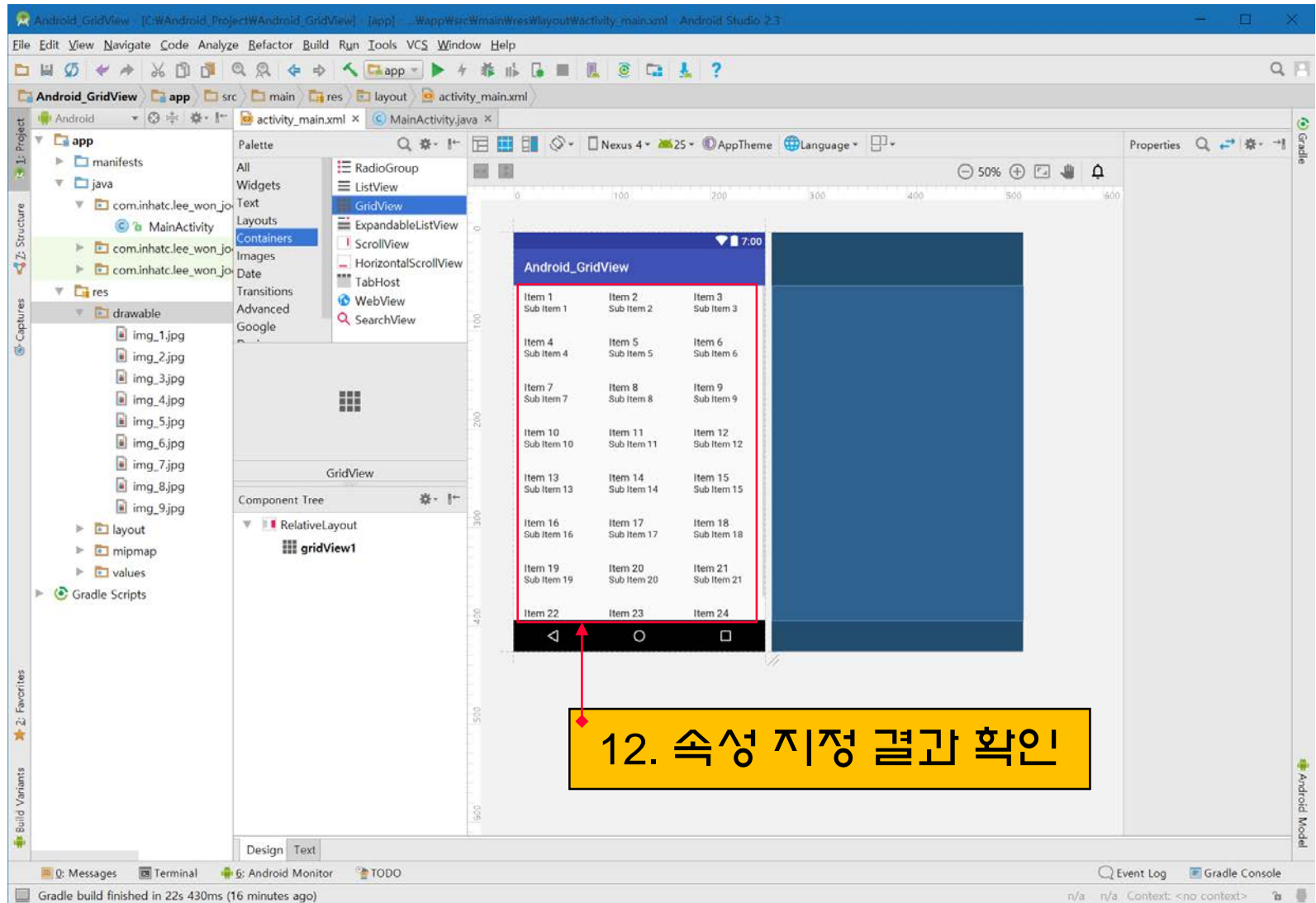
컨트롤	속성 지정
GridView	<ul style="list-style-type: none">• android:id="@+id/gridView1"• android:layout_width="match_parent"• android:layout_height="match_parent"• android:numColumns="auto_fit"• android:verticalSpacing="5dp"• android:horizontalSpacing="5dp"• android:columnWidth="100dp"• android:stretchMode="columnWidth"• android:gravity="center"





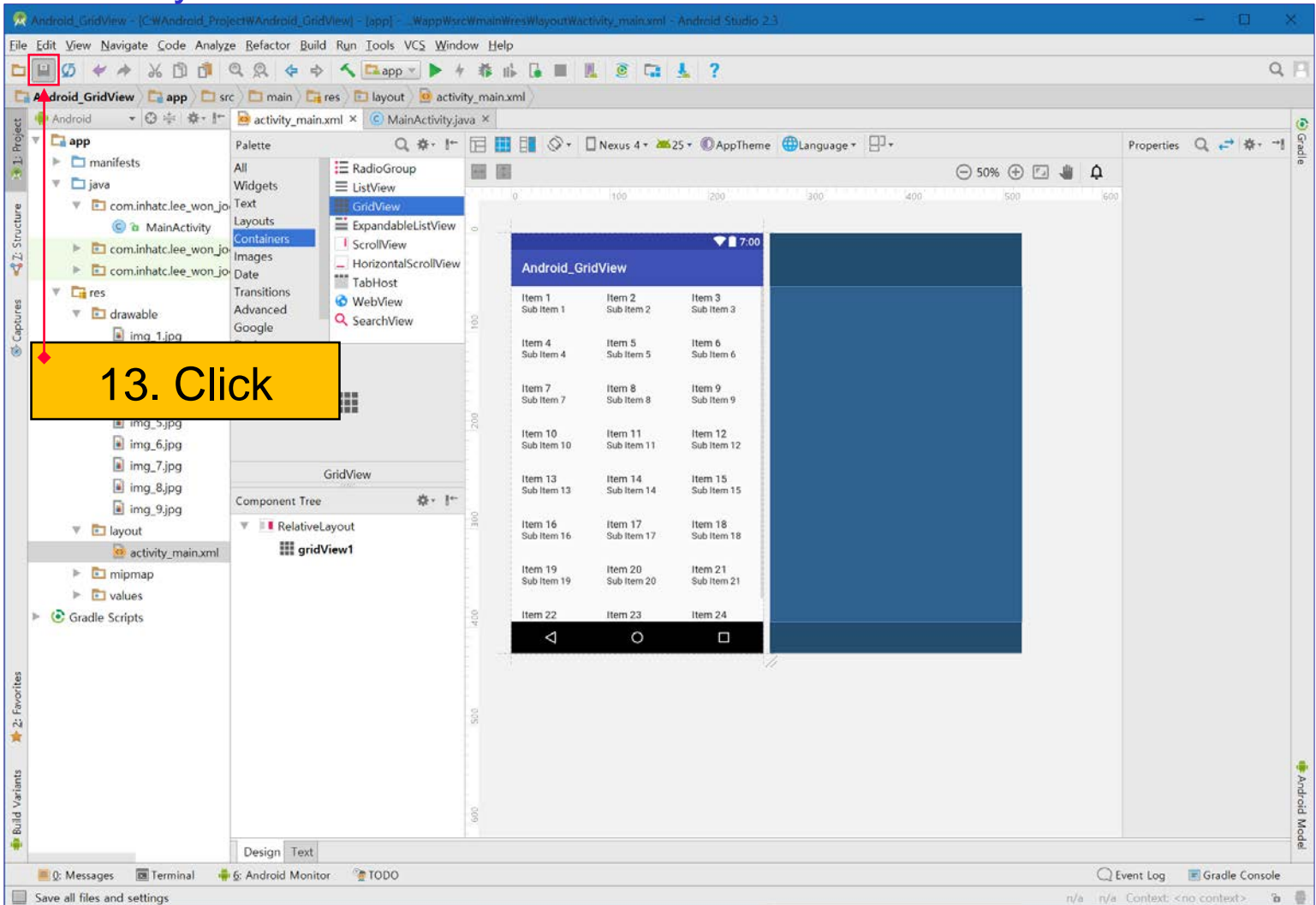
Grid View 구현 (9)

속성 지정 결과



Grid View 구현 (10)

❖ activity_main.xml 저장





Grid View 구현 (11)

❖ activity_main.xml 저장

```
activity_main.xml ×
1  <?xml version="1.0" encoding="utf-8"?>
2  <RelativeLayout 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="com.inhatc.lee_won_joo.android_gridview.MainActivity">
8
9      <GridView
10         android:id="@+id/gridView1"
11         android:layout_width="match_parent"
12         android:layout_height="match_parent"
13         android:numColumns="auto_fit"
14         android:verticalSpacing="5dp"
15         android:horizontalSpacing="5dp"
16         android:columnWidth="100dp"
17         android:stretchMode="columnWidth"
18         android:gravity="center" />
19  </RelativeLayout>
20
```

14. XML code 추가 확인



Grid View 구현 (12)

❖ Import

```
MainActivity.java x
1  package com.inhatec.lee_won_joo.android_gridview;
2
3  import android.content.Context;
4  import android.support.v7.app.AppCompatActivity;
5  import android.os.Bundle;
6  import android.view.View;
7  import android.view.ViewGroup;
8  import android.widget.BaseAdapter;
9  import android.widget.GridView;
10 import android.widget.ImageView;
11
12 public class MainActivity extends AppCompatActivity {
13
14     GridView objGridView;    //GridView Object
15
16     @Override
17     protected void onCreate(Bundle savedInstanceState) {
18         super.onCreate(savedInstanceState);
19         setContentView(R.layout.activity_main);
20
21         objGridView = (GridView)findViewById(R.id.gridView1);
22         objGridView.setAdapter(new ImageAdapter(this));
23     }
24 }
```

15. Coding



Grid View 구현 (13)

❖ ImageAdapter

```
MainActivity.java x
25 public class ImageAdapter extends BaseAdapter {
26     private Context mContext;
27     private Integer[] mThumbIds = { // 참조되는 images
28         R.drawable.img_1, R.drawable.img_2, R.drawable.img_3,
29         R.drawable.img_4, R.drawable.img_5, R.drawable.img_6,
30         R.drawable.img_7, R.drawable.img_8, R.drawable.img_9
31     };
32
33     public ImageAdapter(Context objC) { mContext = objC; }
34     public int getCount() { return mThumbIds.length; }
35     public Object getItem(int position) { return null; }
36     public long getItemId(int position) { return 0; }
37
38     // Adapter에 의해 참조되는 각 item을 위한 new ImageView 생성
39     public View getView(int position, View convertView, ViewGroup parent) {
40         ImageView objImageView;
41
42         if (convertView == null) { //initialize property
43             objImageView = new ImageView(mContext);
44             objImageView.setLayoutParams(new GridView.LayoutParams(85, 85));
45             objImageView.setScaleType(ImageView.ScaleType.CENTER_CROP);
46             objImageView.setPadding(8, 8, 8, 8);
47         } else {
48             objImageView = (ImageView) convertView;
49         }
50         objImageView.setImageResource(mThumbIds[position]);
51         return objImageView;
52     }
53 }
54 }
55 }
```




Grid View 구현 (14)

❖ Android 프로젝트 실행

The screenshot shows the Android Studio IDE with the 'Run' button highlighted in the top toolbar. A yellow box labeled '17. Click' points to the 'Run' button. Below the toolbar, the 'Run' menu is open, showing various options like 'Run', 'Debug', and 'Apply Changes'. A yellow box labeled '18. Click' points to the 'MyAndroid_Nexus One API 25' option in the 'Available Virtual Devices' list. At the bottom of the dialog, a yellow box labeled '19. Click' points to the 'OK' button. The background shows the project structure of 'Android_GridView' with files like 'activity_main.xml' and 'img_1.jpg' through 'img_9.jpg'.

17. Click

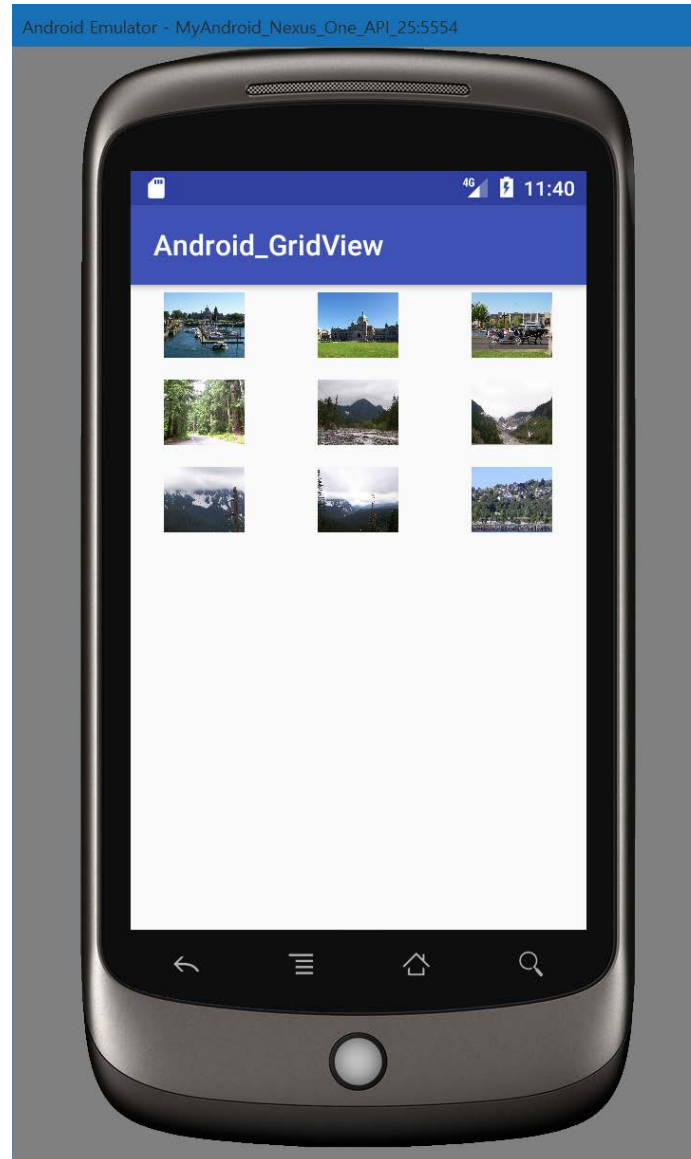
18. Click

19. Click



Grid View 구현 (15)

❖ 실행 결과



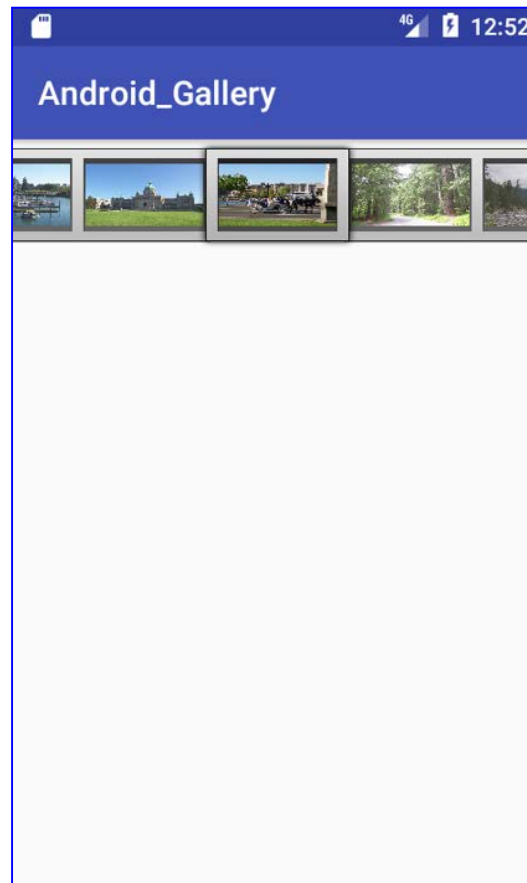


Gallery

■ Gallery

❖ 기능

- ◆ Image를 수평으로 scroll 하면서 보여주는 기능

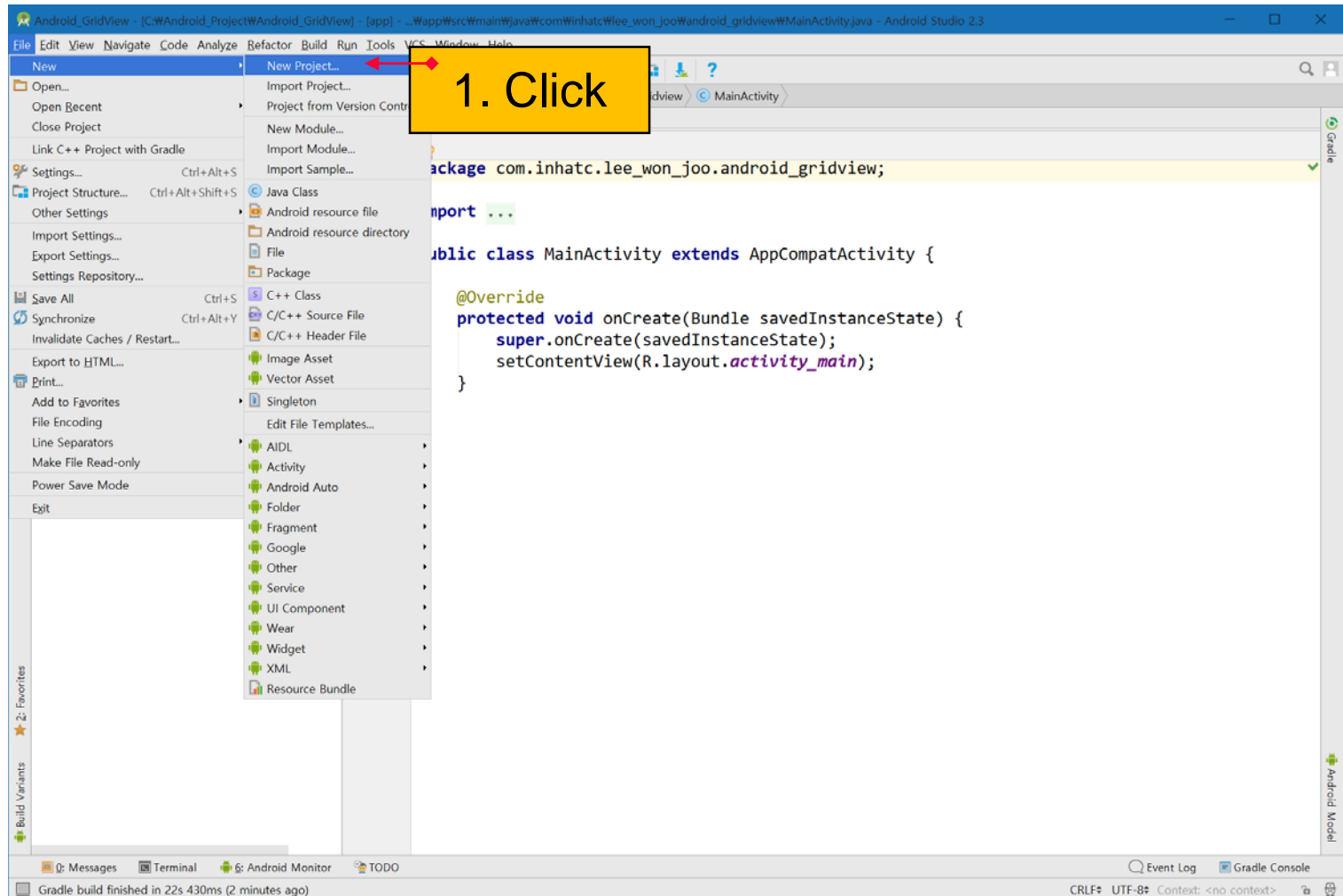




Gallery 구현 (1)

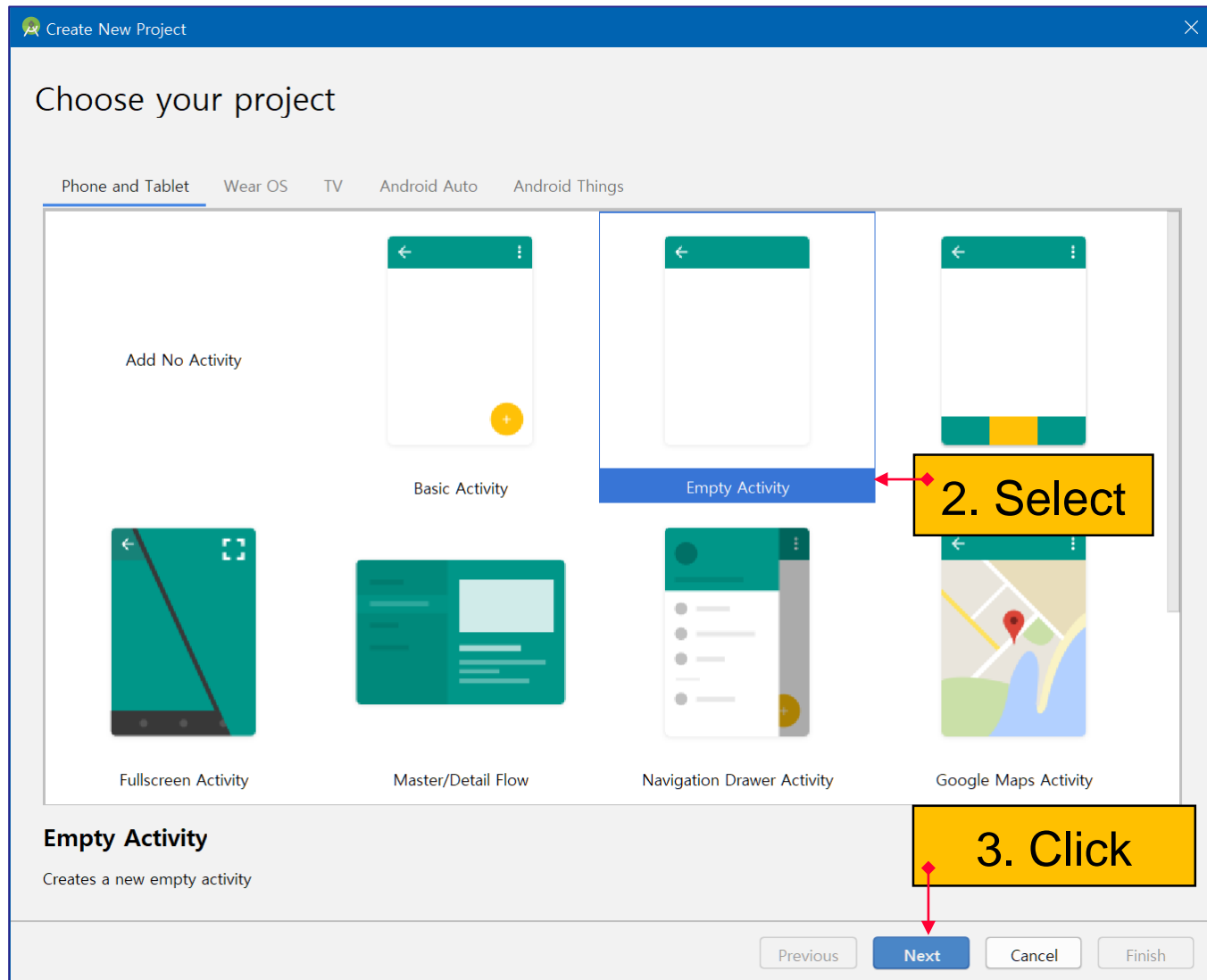
■ Android 프로젝트 생성

❖ 프로젝트 명 : Android_Gallery





Gallery 구현 (2)





Gallery 구현 (3)

Create New Project

Configure your project

Name
Android_Gallery

Package name
com.inhatc.android_gallery

Save location
C:\Android_Project\Android_Gallery

Language
Java

Minimum API level
API 26: Android 8.0 (Oreo)

☐ Your app will run on approximately 6.0% of devices.
[Help me choose](#)

☐ This project will support instant apps

☐ Use AndroidX artifacts

Empty Activity

Creates a new empty activity

Previous Next Cancel Finish

4. Input "Android_Gallery"

5. Input "com.inhatc.android_gallery"

6. Select

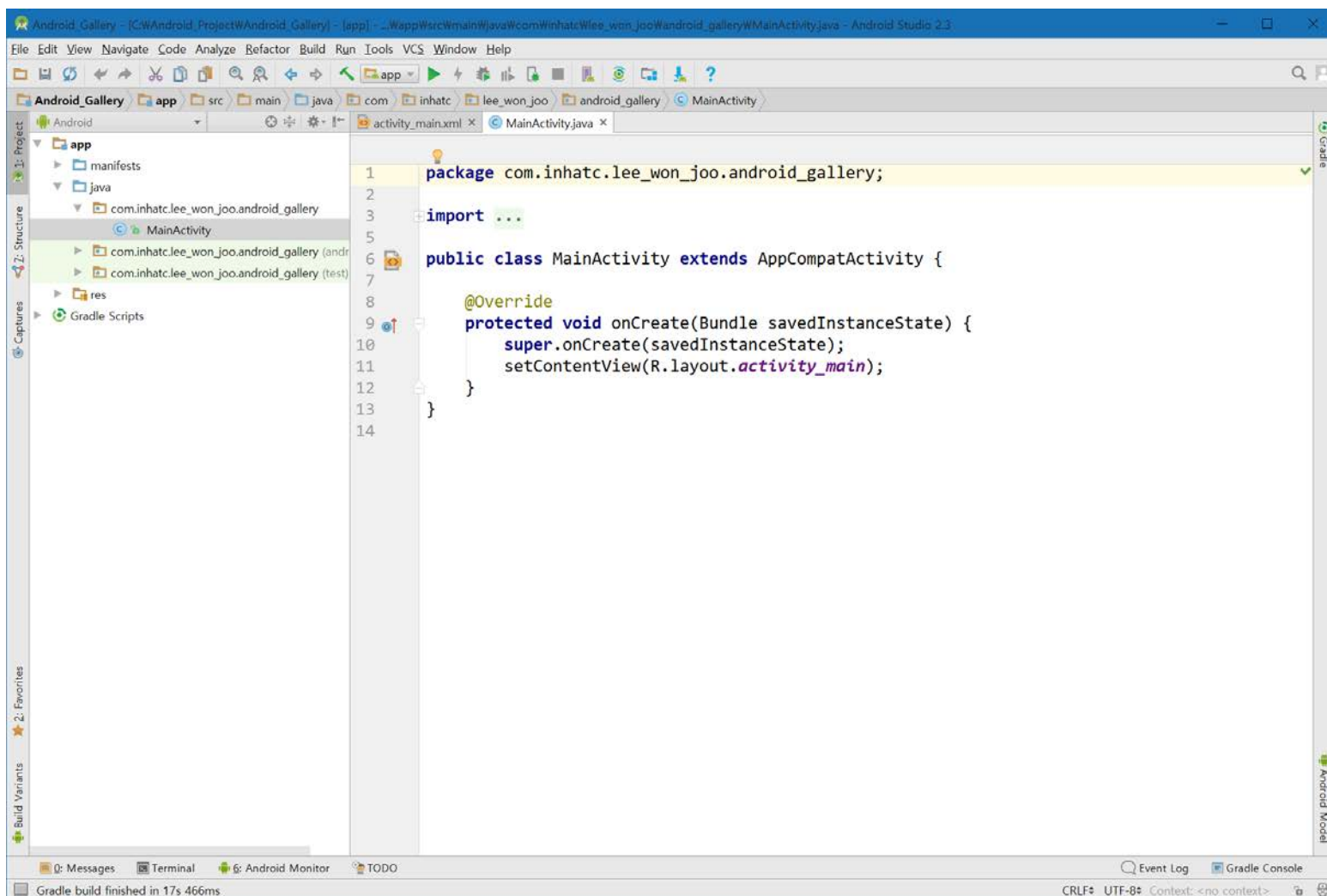
7. Select

8. Click



Gallery 구현 (4)

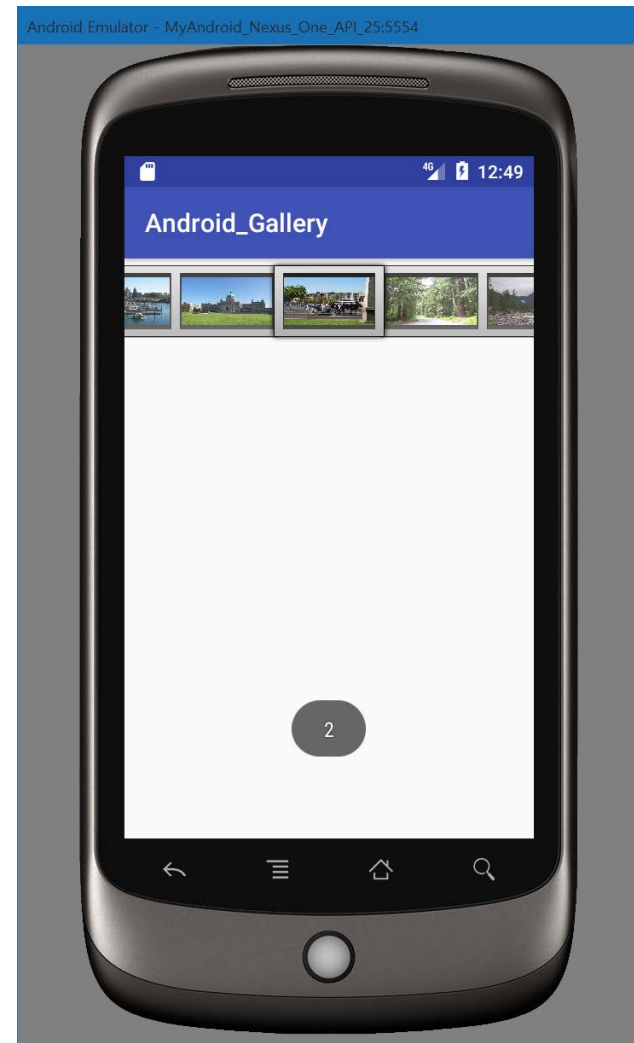
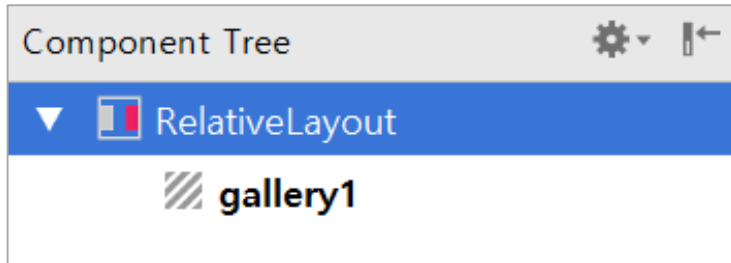
❖ Android 프로젝트 생성 완료





Gallery 구현 (5)

■ UI 설계





Gallery 구현 (6)

❖ TextView 삭제

9. 마우스
오른쪽 버튼 Click

10. Click

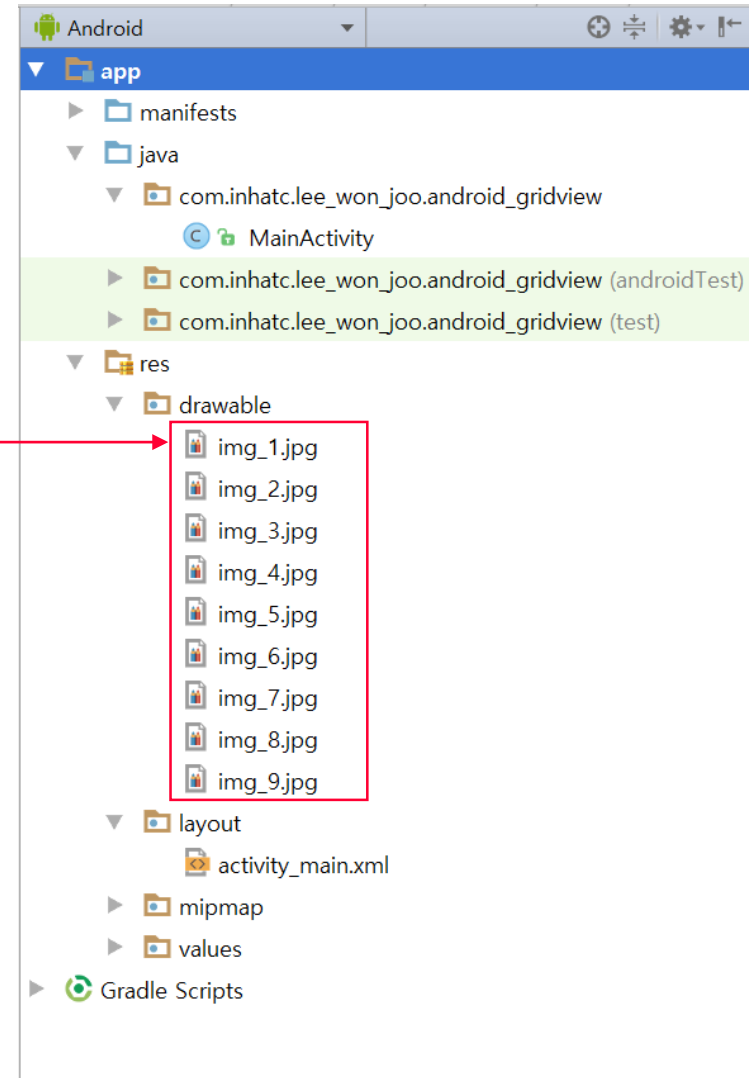
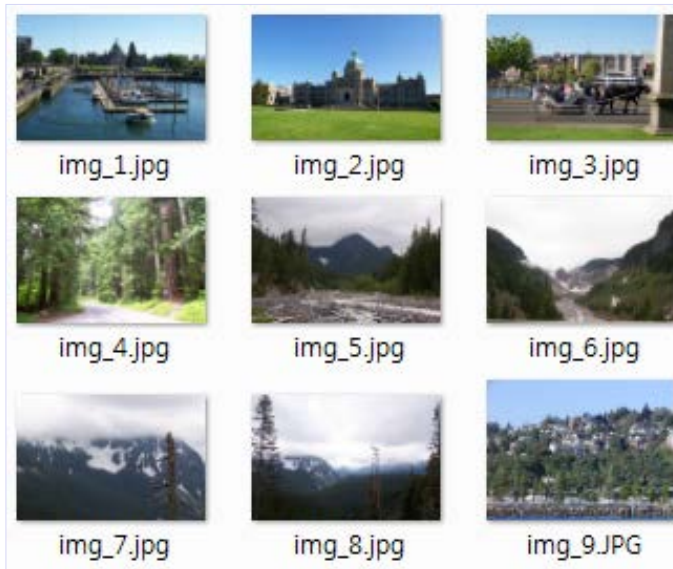


Gallery 구현 (7)

■ Image 복사

❖ .\res\drawable 폴더

11. image 복사





Gallery 구현 (8)

❖ Gallery 속성 지정

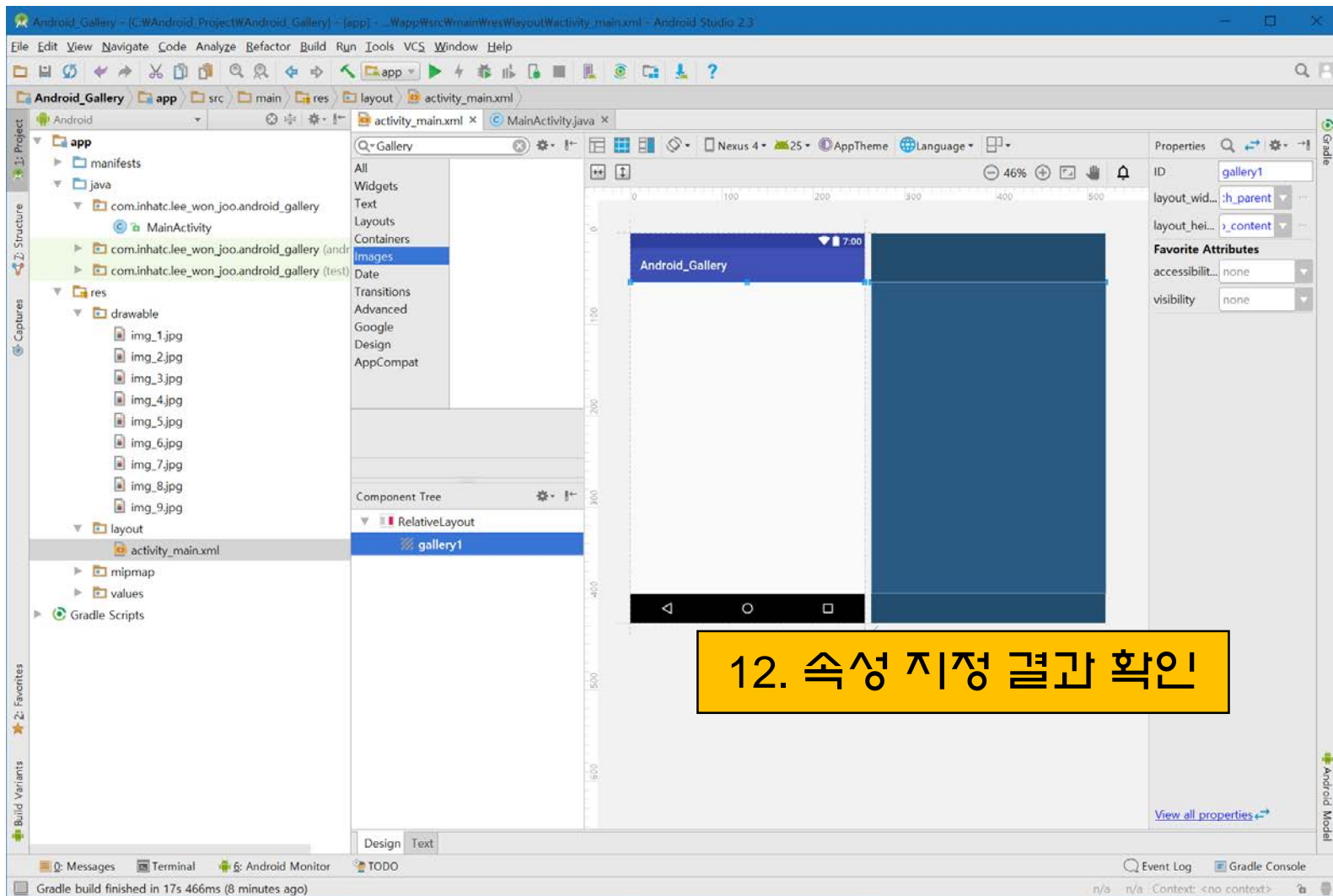
컨트롤	속성 지정
Gallery	<ul style="list-style-type: none">• android:id="@+id/gallery1"• android:layout_width="match_parent"• android:layout_height="wrap_content"





Gallery 구현 (9)

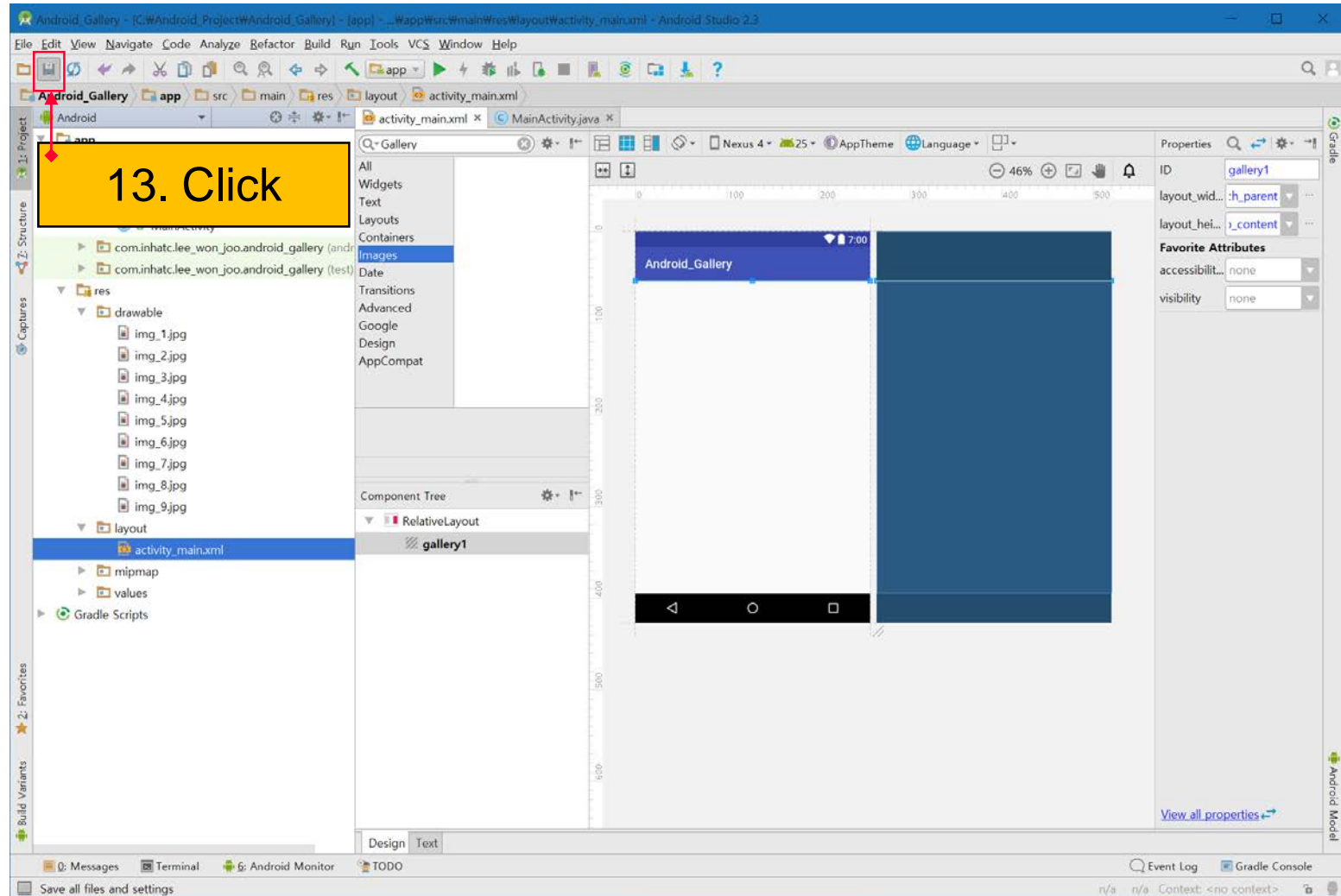
❖ Gallery 속성 지정 결과





Gallery 구현 (10)

❖ activity_main.xml 저장





Gallery 구현 (11)

❖ activity_main.xml

```
activity_main.xml x
1  <?xml version="1.0" encoding="utf-8"?>
2  <RelativeLayout 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="com.inhatc.lee_won_joo.android_gallery.MainActivity">
8
9      <Gallery
10         android:id="@+id/gallery1"
11         android:layout_width="match_parent"
12         android:layout_height="wrap_content" />
13
14  </RelativeLayout>
15
```

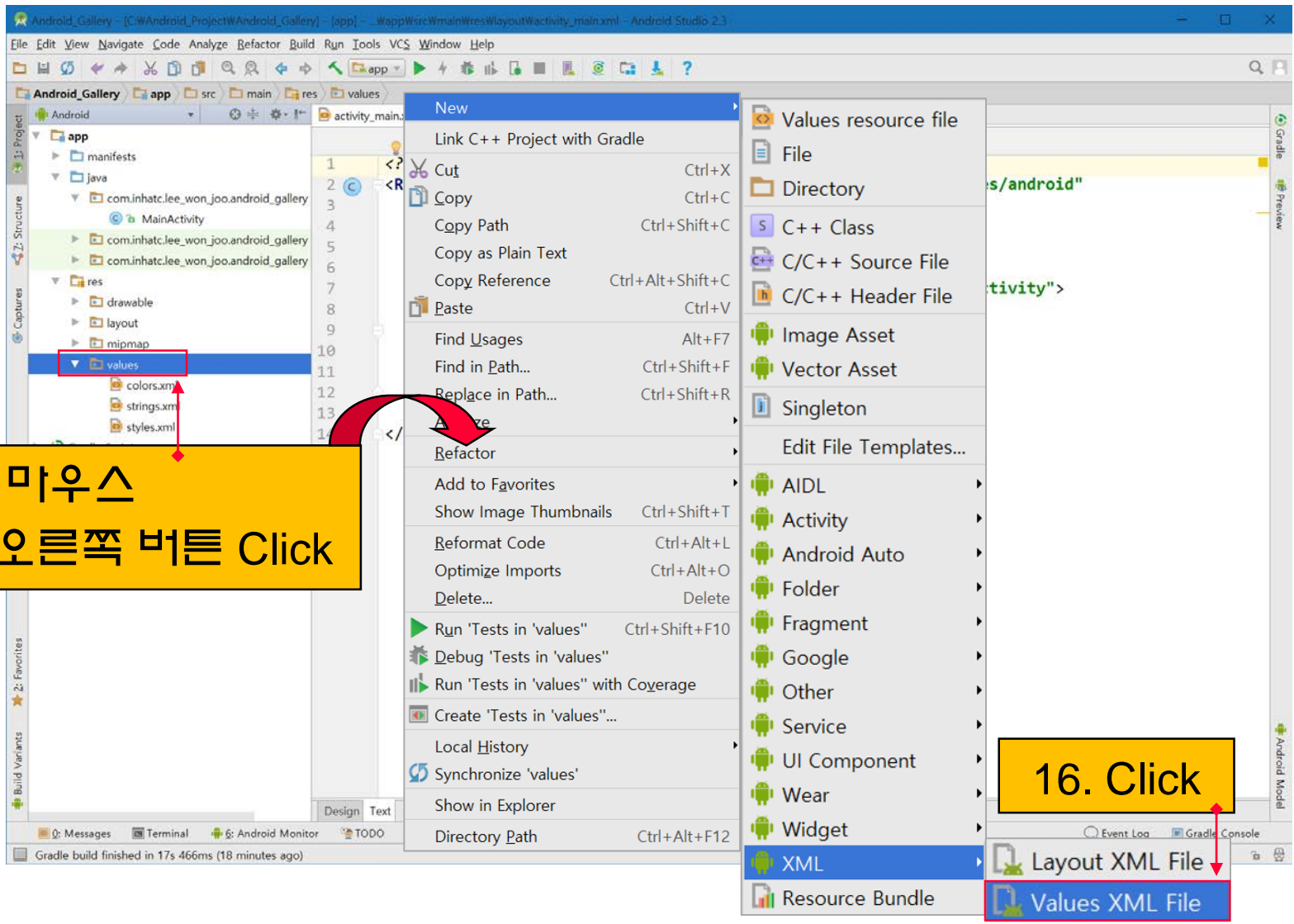
14. XML code 추가 확인



Gallery 구현 (12)

❖ Create gallery_attr.xml

15. 마우스
오른쪽 버튼 Click



16. Click



Gallery 구현 (13)

New Android Component

Configure Component
Android Studio

Creates a new XML values file.

Values File Name:

Name of the values XML file.

Previous Next Cancel Finish

17. "gallery_attr.xml" 입력

18. Click



Gallery 구현 (14)

❖ gallery_attr.xml

```
gallery_attr.xml x
1  <?xml version="1.0" encoding="utf-8"?>
2  <resources>
3      <declare-styleable name="gallery1">
4          <attr name="android:galleryItemBackground" />
5      </declare-styleable></resources>
6
```

19. XML code 추가 확인





Gallery 구현 (15)

❖ MainActivity.java

```
MainActivity.java x
1 package com.inhatc.lee_won_joo.android_gallery;
2
3 import android.content.Context;
4 import android.content.res.TypedArray;
5 import android.support.v7.app.AppCompatActivity;
6 import android.os.Bundle;
7 import android.view.View;
8 import android.view.ViewGroup;
9 import android.widget.AdapterView;
10 import android.widget.BaseAdapter;
11 import android.widget.Gallery;
12 import android.widget.ImageView;
13 import android.widget.Toast;
14
15 public class MainActivity extends AppCompatActivity {
16
17     Gallery objGallery; //Gallery object
18
19     @Override
20     protected void onCreate(Bundle savedInstanceState) {
21         super.onCreate(savedInstanceState);
22         setContentView(R.layout.activity_main);
23
24         objGallery = (Gallery)findViewById(R.id.gallery1);
25         objGallery.setAdapter(new ImageAdapter(this));
26         objGallery.setOnItemClickListener( new AdapterView.OnItemClickListener() {
27             public void onItemClick(AdapterView parent, View v, int position, long id) {
28                 Toast.makeText(MainActivity.this, "" + position, Toast.LENGTH_LONG).show();
29             }
30         });
31     }
}
```

20. Coding



Gallery 구현 (16)

```
MainActivity.java ×

33 public class ImageAdapter extends BaseAdapter {
34     int mGalleryItemBackground;
35     private Context mContext;
36     private Integer[] mThumbIds = { // Reference images
37         R.drawable.img_1, R.drawable.img_2, R.drawable.img_3,
38         R.drawable.img_4, R.drawable.img_5, R.drawable.img_6,
39         R.drawable.img_7, R.drawable.img_8, R.drawable.img_9
40     };
41
42     public ImageAdapter(Context objContxt) {
43         mContext = objContxt;
44         TypedArray objArray = obtainStyledAttributes(R.styleable.gallery1);
45         mGalleryItemBackground = objArray.getResourceId(
46             R.styleable.gallery1_android_galleryItemBackground, 0);
47         objArray.recycle();
48     }
49     public int getCount() { return mThumbIds.length; }
50     public Object getItem(int position) { return position; }
51     public long getItemId(int position) { return position; }
52
53     // Adapter에 의해 참조되는 각 item을 new ImageView 생성
54     public View getView(int position, View convertView, ViewGroup parent) {
55         ImageView objImageView; //ImageView object
56
57         objImageView = new ImageView(mContext);
58         objImageView.setImageResource(mThumbIds[position]);
59         objImageView.setLayoutParams(new Gallery.LayoutParams(150, 100));
60         objImageView.setScaleType(ImageView.ScaleType.FIT_XY);
61         objImageView.setBackgroundResource(mGalleryItemBackground);
62         return objImageView;
63     }
64 }
65 }
```

21. Coding





Gallery 구현 (17)

❖ Android 프로젝트 실행

The screenshot shows the Android Studio 2.3 interface. The 'Run' menu is open, and the 'Run' option is highlighted. A yellow box with the text '22. Click' and a red arrow points to the 'Run' menu item. Below the 'Run' menu, the 'Select Deployment Target' dialog is open. In the 'Available Virtual Devices' section, 'MyAndroid_Nexus One API 25' is selected. A yellow box with the text '23. Click' and a red arrow points to this device. At the bottom of the dialog, the 'OK' button is highlighted. A yellow box with the text '24. Click' and a red arrow points to the 'OK' button. The background shows the project structure of 'Android_Gallery' with files like 'MainActivity.java', 'colors.xml', 'gallery_attr.xml.xml', 'strings.xml', and 'styles.xml'.

22. Click

23. Click

24. Click

Gallery 구현 (18)

❖ 실행 결과



25. 실행 결과 확인





학습 요약

■ Grid View

■ Gallery



open handset alliance