

Extrait du site [Byron Kiourtoglou](#)

<http://examples.javacodegeeks.com/android/core/ui/gridview/android-gridview-example/>

## Android GridView Example

One of the most useful layouts in Android is the `GridView`. `GridView` organizes the items of your screen in a two-dimensional array (a grid...). In this tutorial you are going to see two examples of `GridView`. In the first part we see the normal use of `GridView`. In the second part we create our own customized `GridView`.

For this tutorial, we will use the following tools in a Windows 64-bit platform:

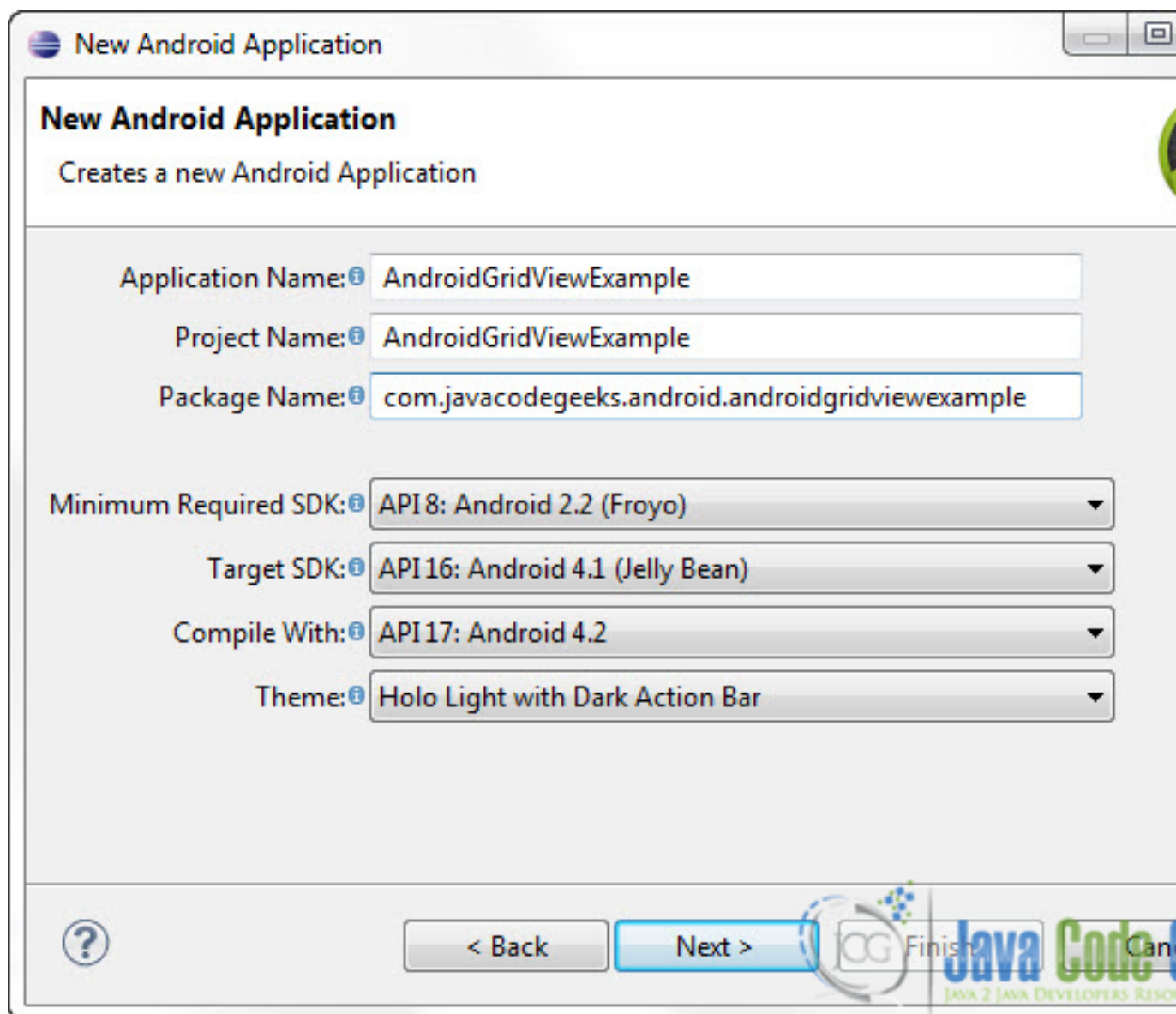
1. JDK 1.7
2. Eclipse 4.2 Juno
3. Android SDK 4.2

### **Create a new Android Project**

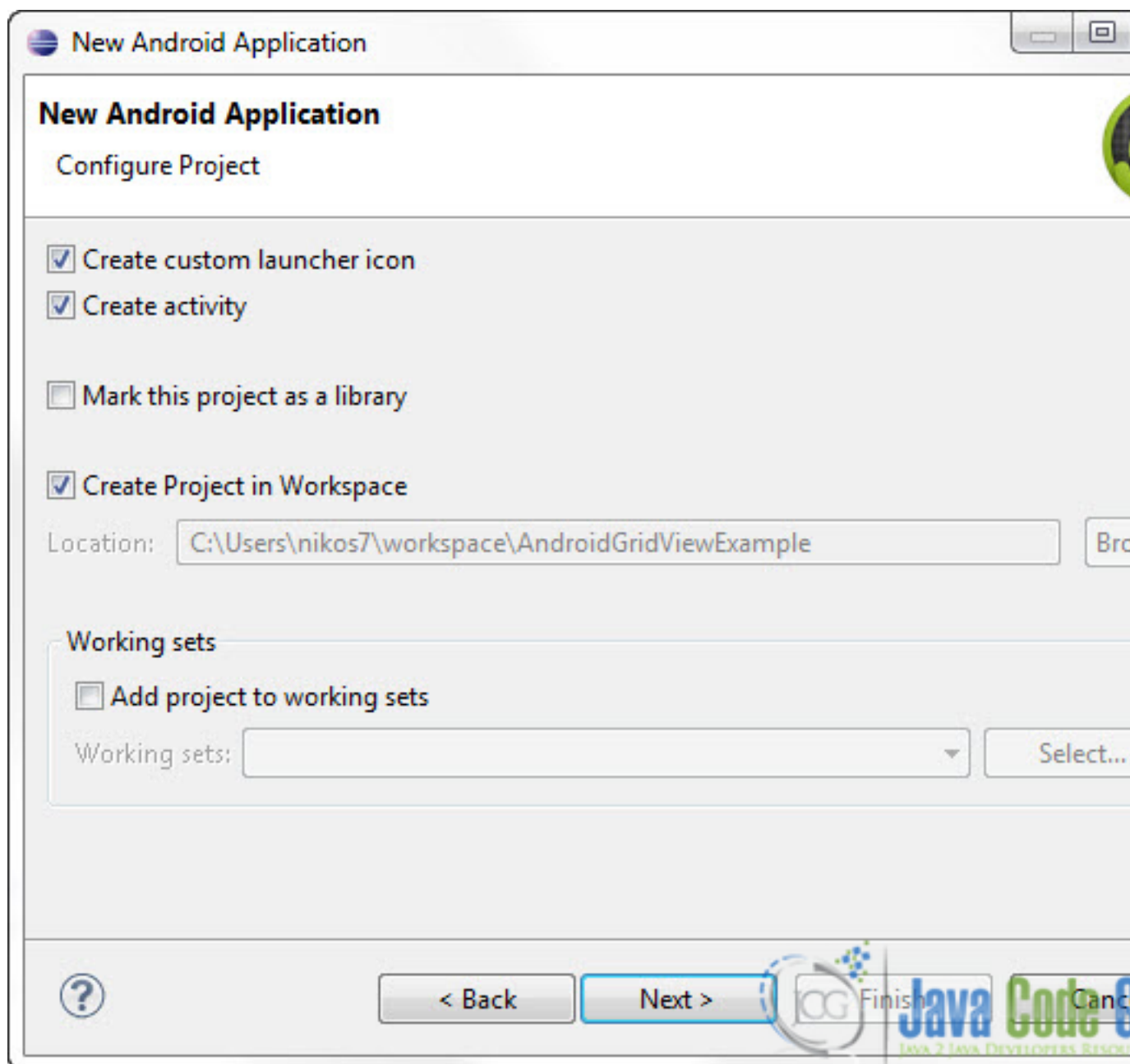
---

Open Eclipse IDE and go to File -> New -> Project -> Android -> Android Application Project and click Next.

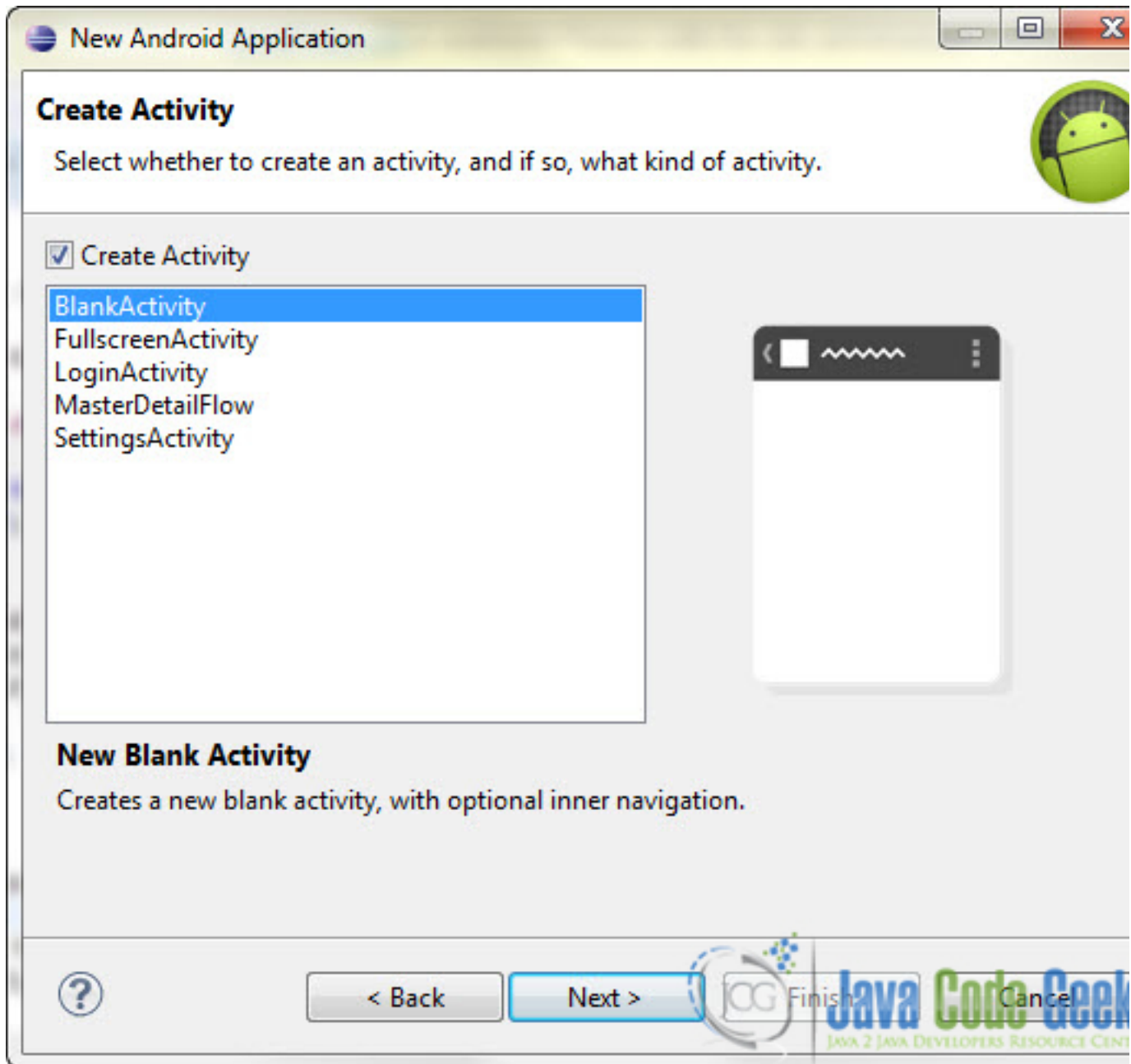
You have to specify the Application Name, the Project Name and the Package name in the appropriate text fields and then click Next.



In the next window make sure the “Create activity” option is selected in order to create a new activity for your project, and click Next. This is optional as you can create a new activity after creating the project, but you can do it all in one step.



Select "BlankActivity" and click Next.



You will be asked to specify some information about the new activity. In the Layout Name text field you have to specify the name of the file that will contain the layout description of your app. In our case the file `res/layout/main.xml` will be created. Then, click Finish.


## New Blank Activity

Creates a new blank activity, with optional inner navigation.

Activity Name

Layout Name

Navigation Type

 The name of the layout to create for the activity



< Back

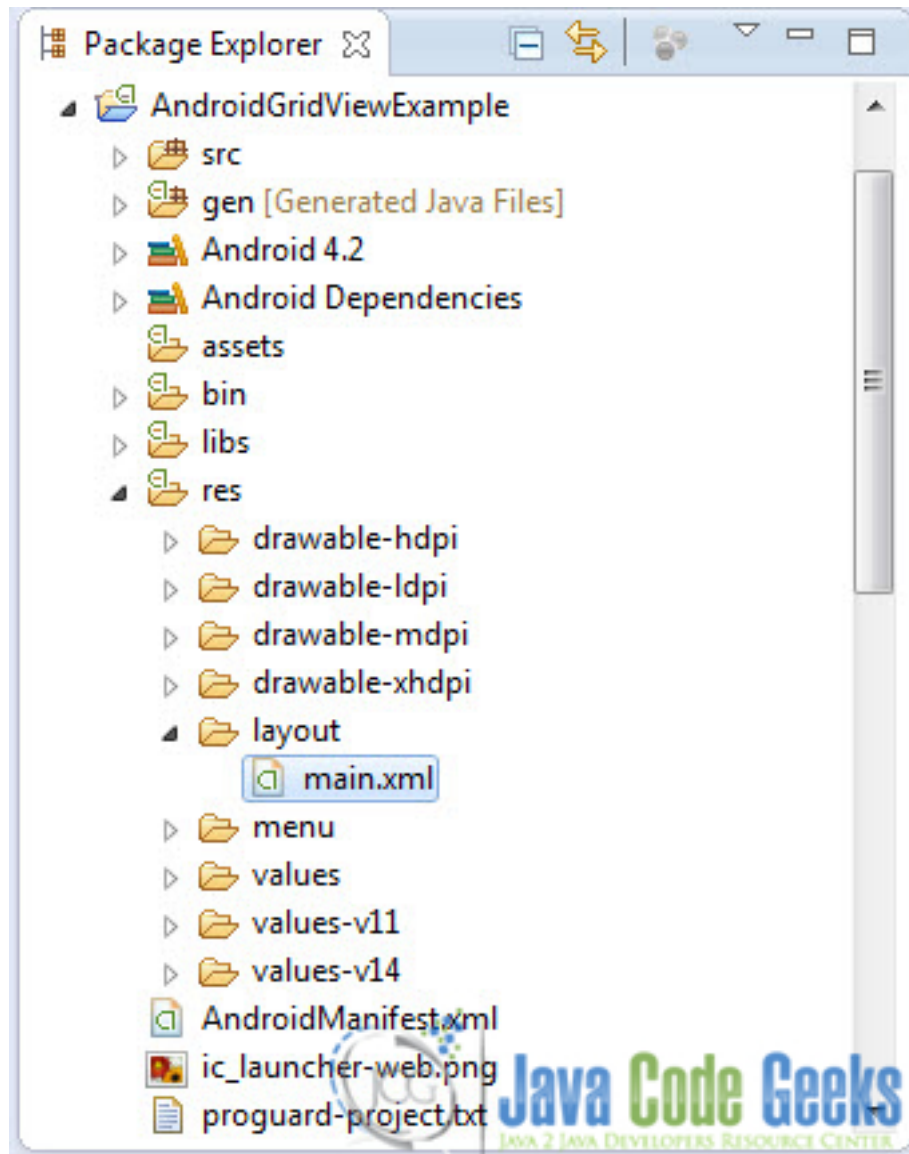
Next >



## Normal GridView Example

### 1.Create the main layout

Open `res/layout/main.xml` file :

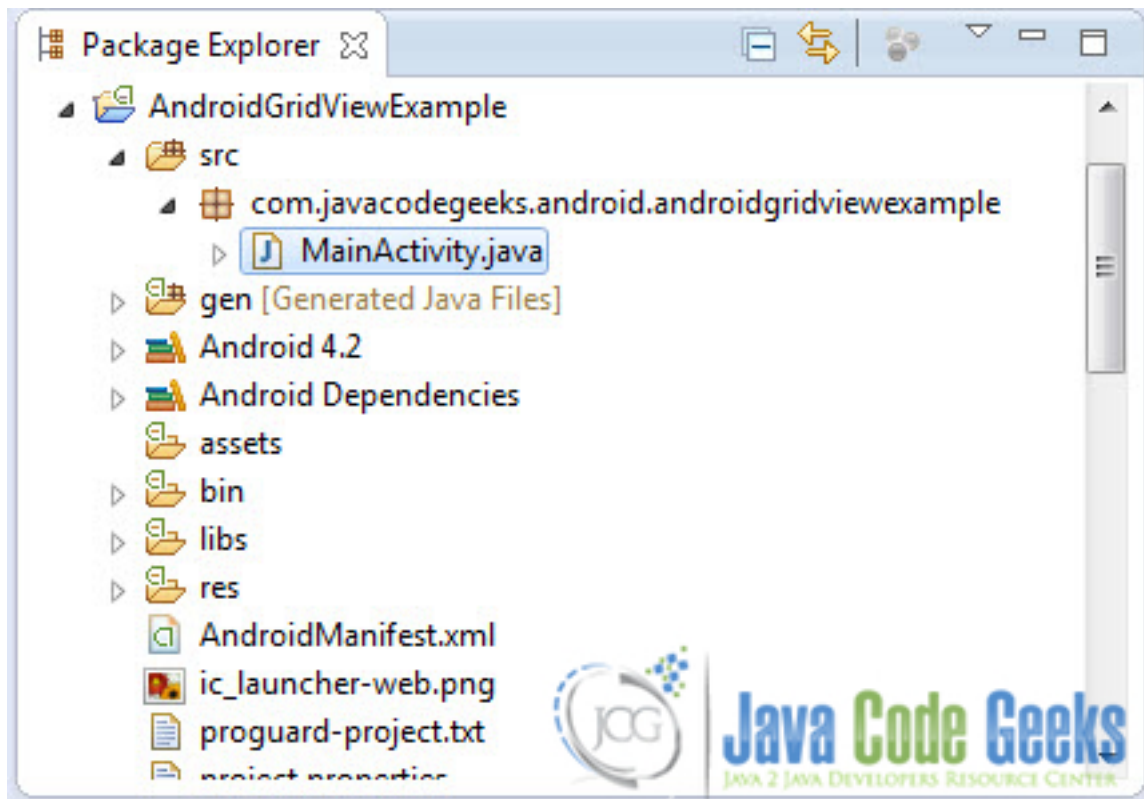


And paste the following code :

## 2. Code

```
01 <?xml version="1.0" encoding="utf-8"?>
02 <GridView
xmlns:android="http://schemas.android.com/apk/res/android"
03   android:id="@+id/gridView"
04     android:layout_width="fill_parent"
05     android:layout_height="fill_parent"
06     android:columnWidth="50dp"
07     android:gravity="center"
08     android:numColumns="auto_fit"
09     android:stretchMode="columnWidth" >
10
11 </GridView>
```

Go to the java file that contains the code of the activity you've just created:



And paste the following code:

```

01  package com.javacodegeeks.android.androidgridviewexample;
02
03  import android.app.Activity;
04  import android.os.Bundle;
05  import android.widget.AdapterView;
06  import android.widget.AdapterView.OnItemClickListener;
07  import android.widget.ArrayAdapter;
08  import android.widget.GridView;
09  import android.widget.TextView;
10  import android.widget.Toast;
11  import android.view.View;
12
13  public class MainActivity extends Activity {
14
15      GridView grid;
16
17      static final String[] letters = new String[] {
18          "A", "B", "C", "D", "E",
19          "F", "G", "H", "I", "J",
20          "K", "L", "M", "N", "O",
21          "P", "Q", "R", "S", "T",
22          "U", "V", "W", "X", "Y", "Z"};
23
24      @Override
25      public void onCreate(Bundle savedInstanceState) {
26          super.onCreate(savedInstanceState);
27
28          setContentView(R.layout.main);
29
30          grid = (GridView) findViewById(R.id.gridView);
31
32          ArrayAdapter adapter = new ArrayAdapter(this,
33  android.R.layout.simple_list_item_1, letters);
34
35          grid.setAdapter(adapter);
36
37          grid.setOnItemClickListener(new OnItemClickListener() {
38              public void onItemClick(AdapterView parent, View v,
39  int position, long id) {
40                  Toast.makeText(getApplicationContext(),
41                      ((TextView) v).getText(),
42                      Toast.LENGTH_SHORT).show();
43              }
44          });
45      }

```



### 3. Run the application

Go ahead and run the application to see how the layout looks on your emulator. This is the main screen of our application:



And when you click on a letter:



### Download Eclipse Project

This was the first part of the Android GridView Example. Download the Eclipse Project of the first part of this tutorial: [AndroidGridViewExample\\_1.zip](http://a5e2fba00d8bcb729d89839f.javacodegeeks.netdna-cdn.com/wp-content/uploads/2013/01/AndroidGridViewExample_1.zip)  
([http://a5e2fba00d8bcb729d89839f.javacodegeeks.netdna-cdn.com/wp-content/uploads/2013/01/AndroidGridViewExample\\_1.zip](http://a5e2fba00d8bcb729d89839f.javacodegeeks.netdna-cdn.com/wp-content/uploads/2013/01/AndroidGridViewExample_1.zip))

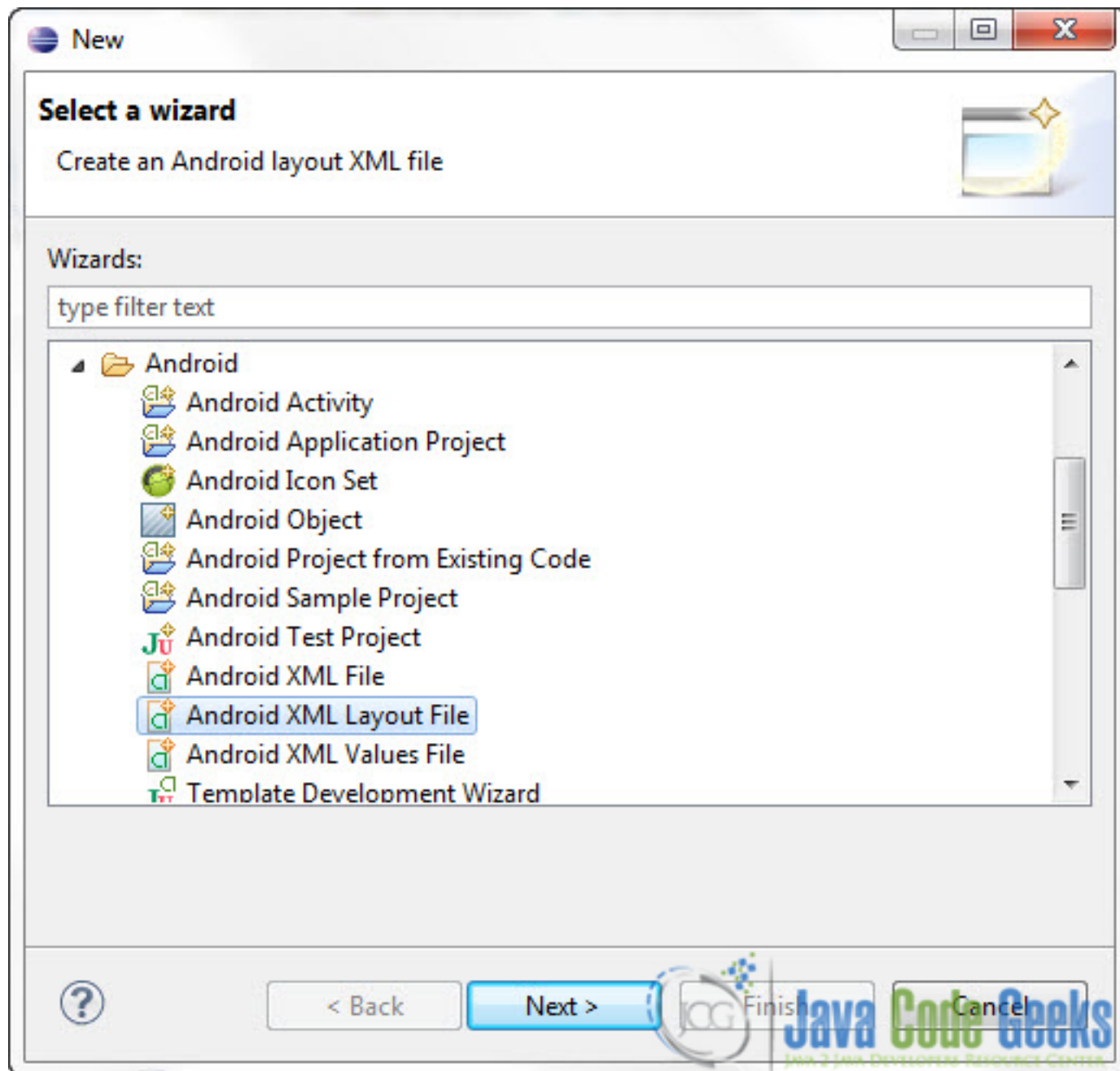
## Custom GridView Example

For this part you can create a new Android Project if you want, but I'm going to use the old one.

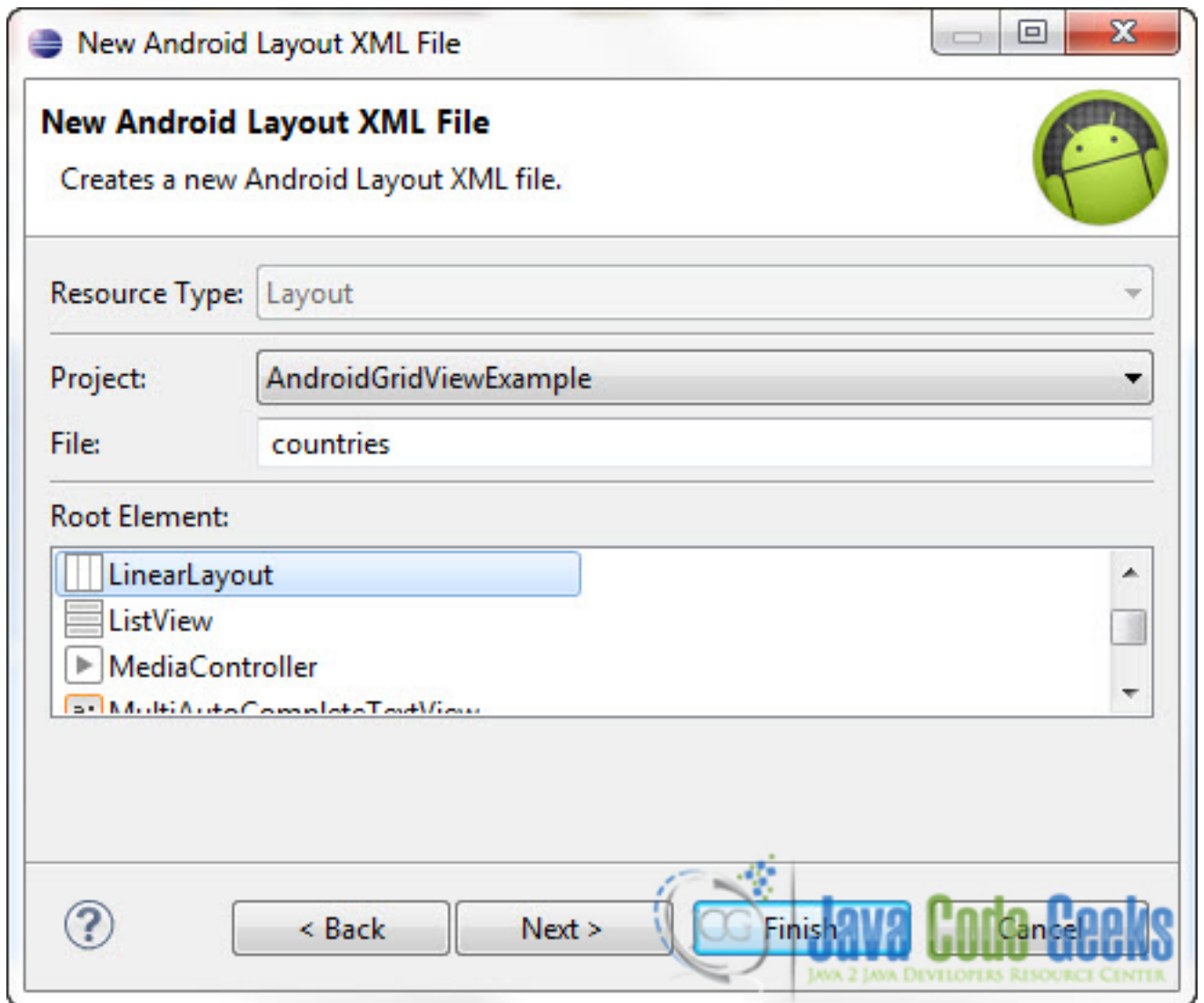
### 1. Create the custom Layout

The first thing you have to do for this part is to create a new xml Layout file that will describe the Layout of our custom GridView.

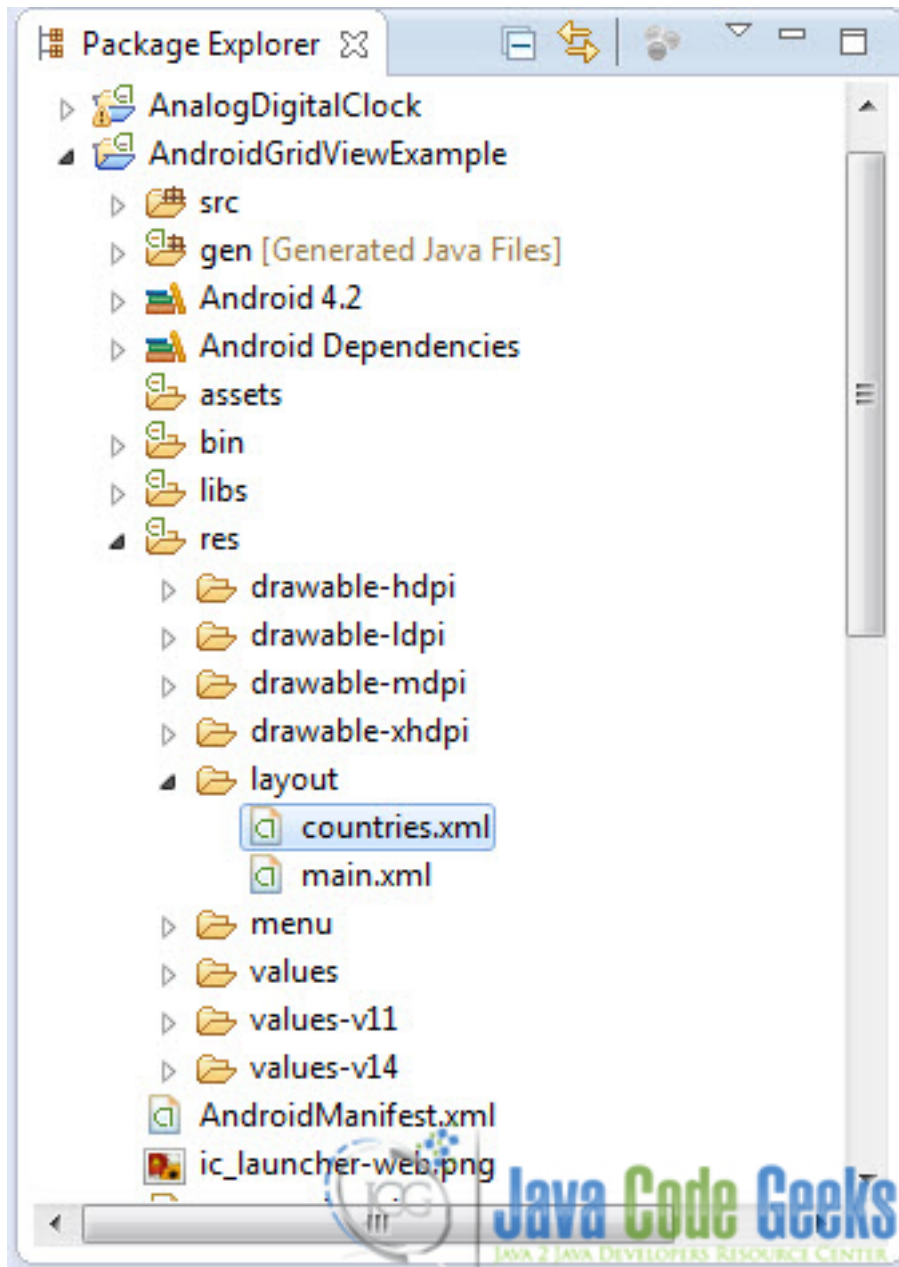
Go to the Package Explorer and right click on the `res/layout` folder. Select New -> Other -> Android -> Android XML Layout File. And click Next:



Then specify the name of the file and the Layout type and click Finish:



As you will see in the Package Explorer the new `/res/layout/countries.xml` file has been created:



Open that file and paste the following code :

```

01  <?xml version="1.0" encoding="utf-8"?>
02  <LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
03      android:layout_width="wrap_content"
04      android:layout_height="wrap_content"
05      android:padding="5dp" >
06
07      <ImageView
08          android:id="@+id/flag"
09          android:layout_width="50sp"
10          android:layout_height="50sp"
11          android:layout_marginRight="10sp">
12      </ImageView>
13
14      <TextView
15          android:id="@+id/label"
16          android:layout_width="wrap_content"
17          android:layout_height="wrap_content"
18          android:text="@+id/label"
19          android:layout_marginTop="15sp"
20          android:textSize="15sp" >
21      </TextView>
22
23  </LinearLayout>

```

**Now, go back to the /res/layout/main.xml and paste the following code :**

```

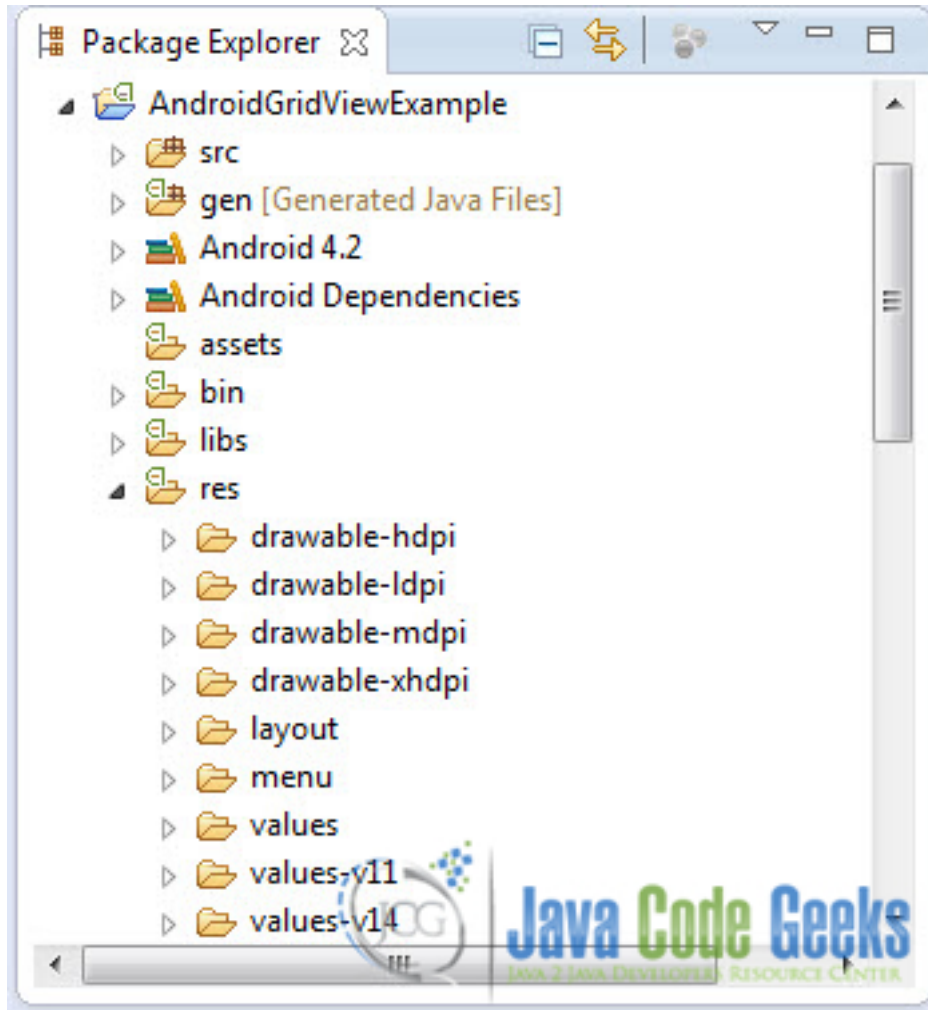
01  <?xml version="1.0" encoding="utf-8"?>
02  <GridView
xmlns:android="http://schemas.android.com/apk/res/android"
03      android:id="@+id/gridView"
04      android:numColumns="auto_fit"
05      android:gravity="center"
06      android:columnWidth="160dp"
07      android:stretchMode="columnWidth"
08      android:layout_width="fill_parent"
09      android:layout_height="fill_parent" >
10
11  </GridView>

```

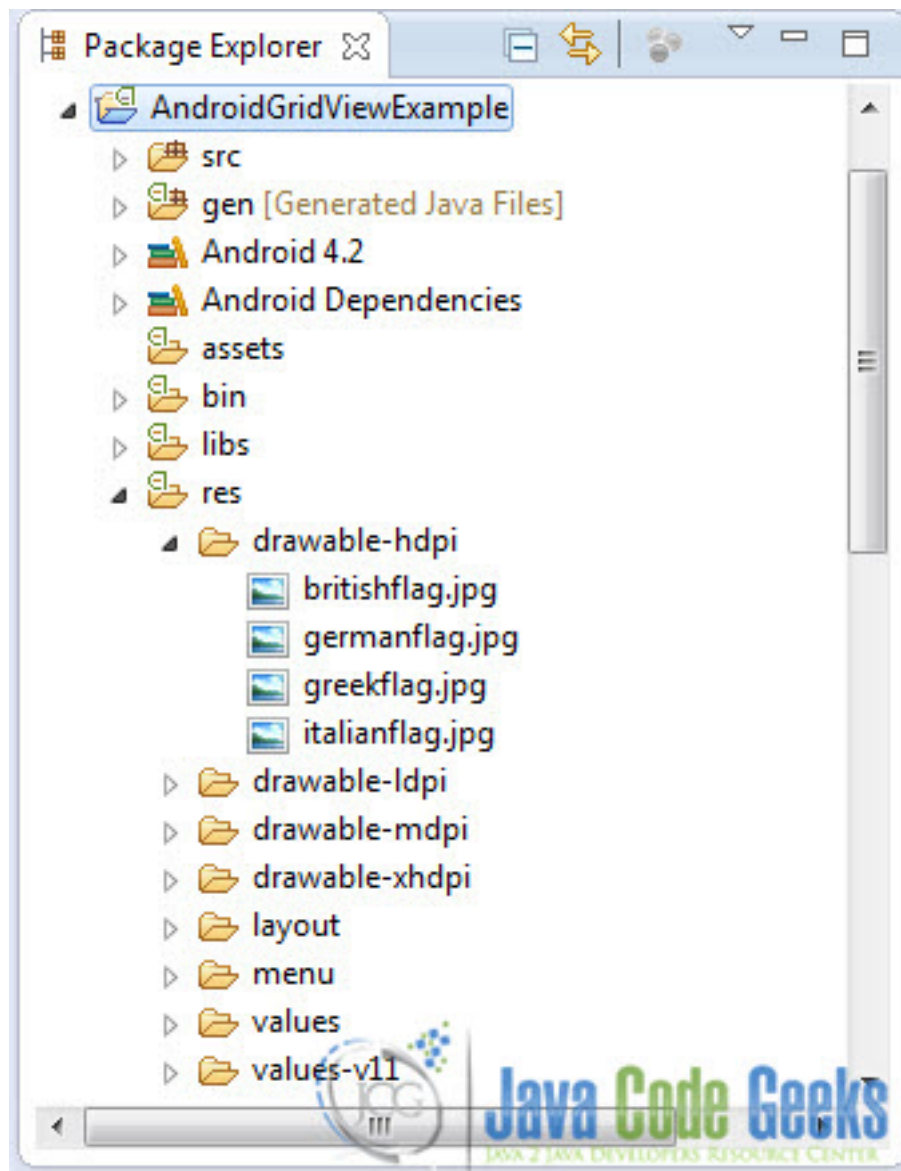
## 2. Adding pictures in the appropriate project folder

As you would have noticed in the Package explorer, there are 4 folders that contain the image resources of the project :

1. `res/drawable-hdpi`
2. `res/drawable-ldpi`
3. `res/drawable-mdpi`
4. `res/drawable-xhdpi`



You can copy the images you want to put in your application in any one of these folders. Android SDK will automatically recognize any images you put on any one of these folders as drawable resources. So, copy the images in the folder you want. If the image does not appear in the Package Explorer under the folder you've copied it into, right click on the project name and select Refresh. Now the image should be under the correct folder. As you can see, I have four images in `res/drawable-hdpi`:

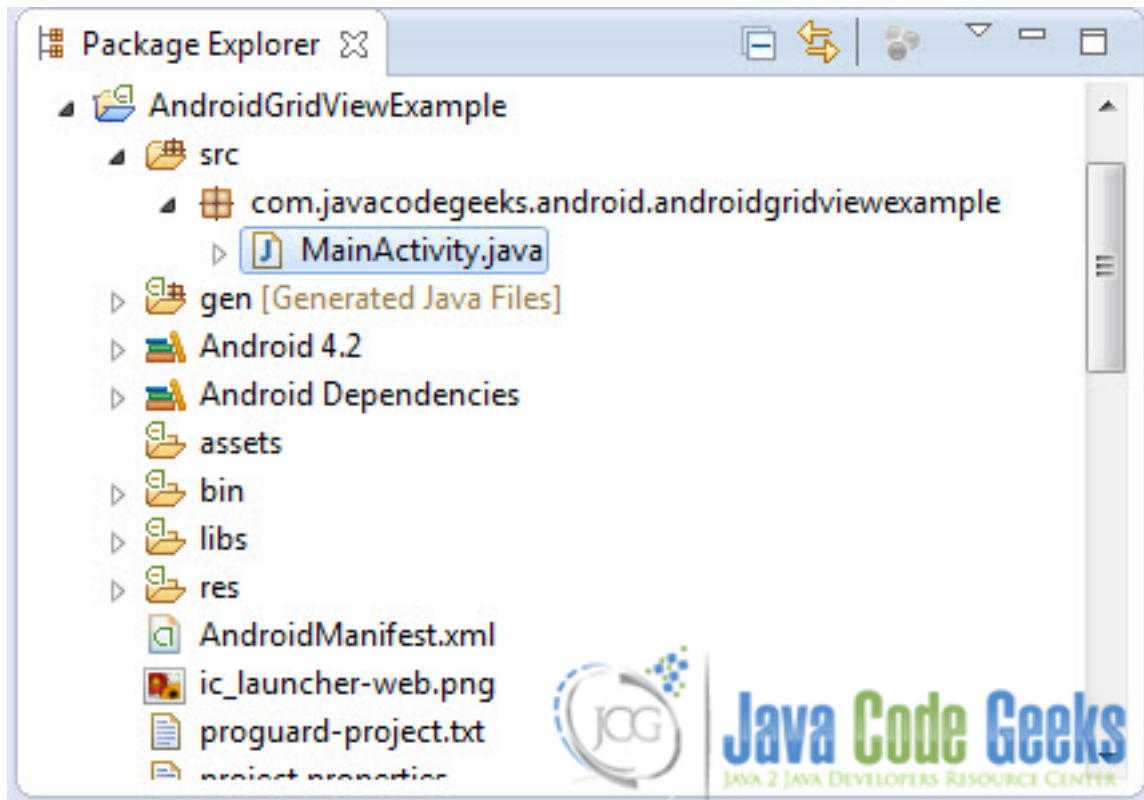


### 3. Create a Custom ArrayAdapter

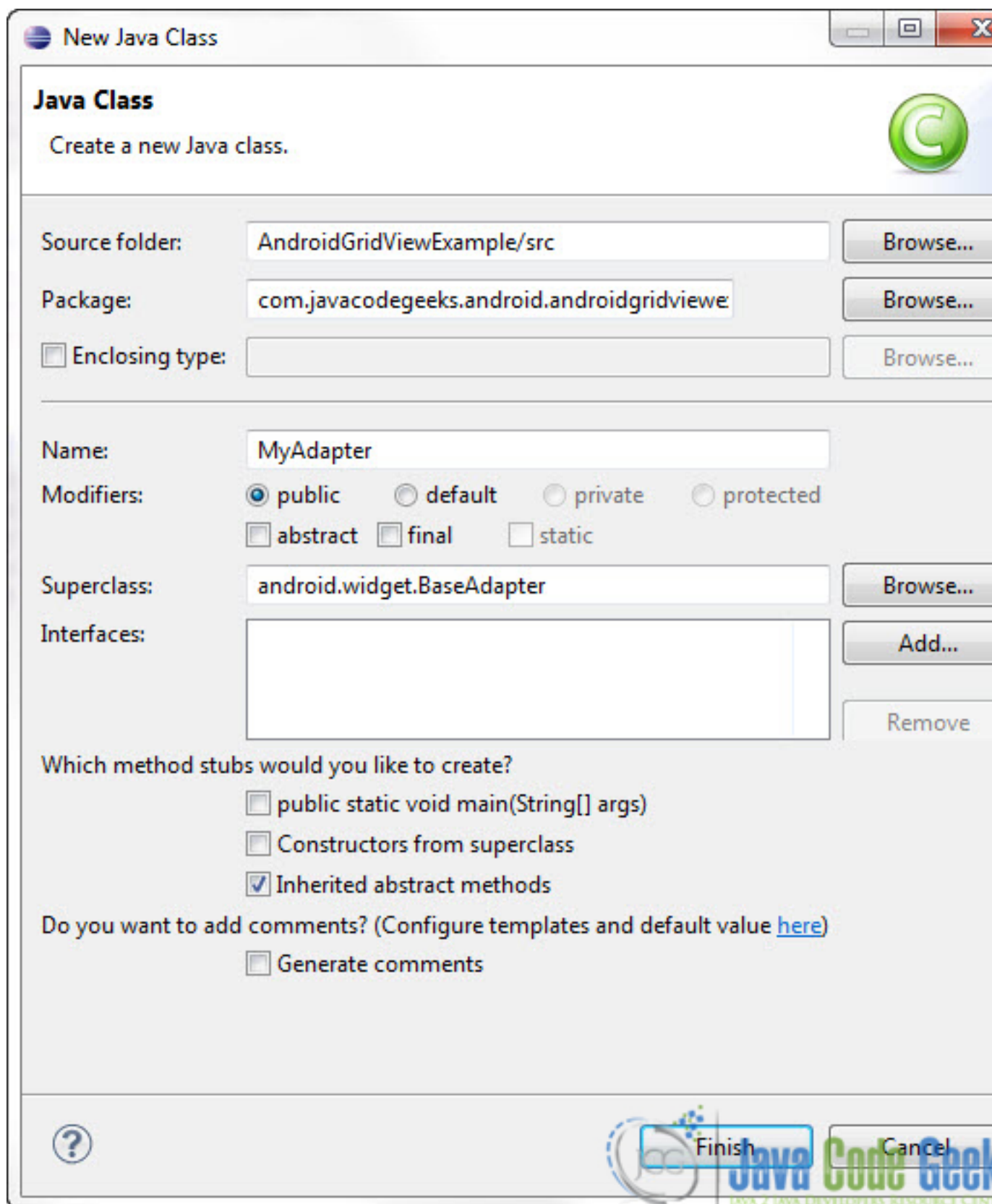
To create a custom `GridView` you have to make your own `BaseAdapter`. To do that you have to create a new class that will extend `BaseAdapter`.

Go to the Package Explorer and right click on the package (in our case `com.javacodegeeks.android.androidgridvieexample`):

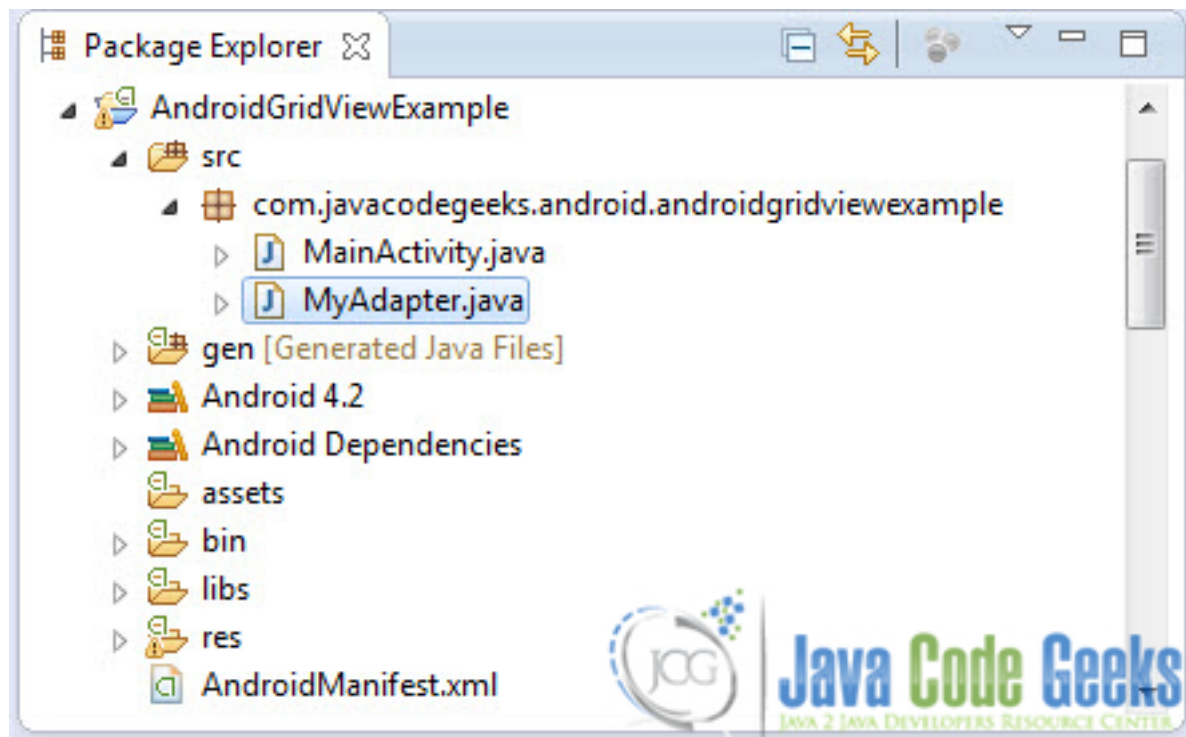




Select New -> Class. Put the appropriate attributes as shown in the picture below:



As you will see a new java file has been created:



Open that file and paste the following code :

```
01 package com.javacodegeeks.android.androidgridviewexample;
02
03 import android.content.Context;
04 import android.view.LayoutInflater;
05 import android.view.View;
06 import android.view.ViewGroup;
07 import android.widget.BaseAdapter;
08 import android.widget.ImageView;
09 import android.widget.TextView;
10
11 public class MyAdapter extends BaseAdapter {
12     private Context context;
13     private final String[] countries;
14
15     public MyAdapter(Context context, String[] countries)
16     {
17         this.context = context;
18         this.countries = countries;
19     }
20     public View getView(int position, View convertView,
21     ViewGroup parent) {
22         LayoutInflater inflater = (LayoutInflater)
23     context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
```

```

23
24     View gridView;
25
26     if (convertView == null) {
27
28         gridView = new View(context);
29
30         gridView =
inflater.inflate(R.layout.countries, null);
31
32         TextView textView = (TextView)
gridView.findViewById(R.id.label);
33
34         textView.setText(countries[position]);
35
36         ImageView flag = (ImageView) gridView
.findViewById(R.id.flag);
37
38         String mobile = countries[position];
39
40         if (mobile.equals("Greece")) {
41
42             flag.setImageResource(R.drawable.greekflag);
43         } else if (mobile.equals("Germany")) {
44
45             flag.setImageResource(R.drawable.germanflag);
46         } else if (mobile.equals("Italy")) {
47
48             flag.setImageResource(R.drawable.italianflag);
49         } else {
50
51             flag.setImageResource(R.drawable.britishflag);
52         }
53
54         return gridView;
55     }
56
57     @Override
58     public int getCount() {
59         return countries.length;
60     }
61
62     @Override
63     public Object getItem(int position) {
64         return null;
65     }

```

```

66
67     @Override
68     public long getItemId(int position) {
69         return 0;
70     }
71
72 }

```

Now, go back to MainActivity.java and paste the following code :

```

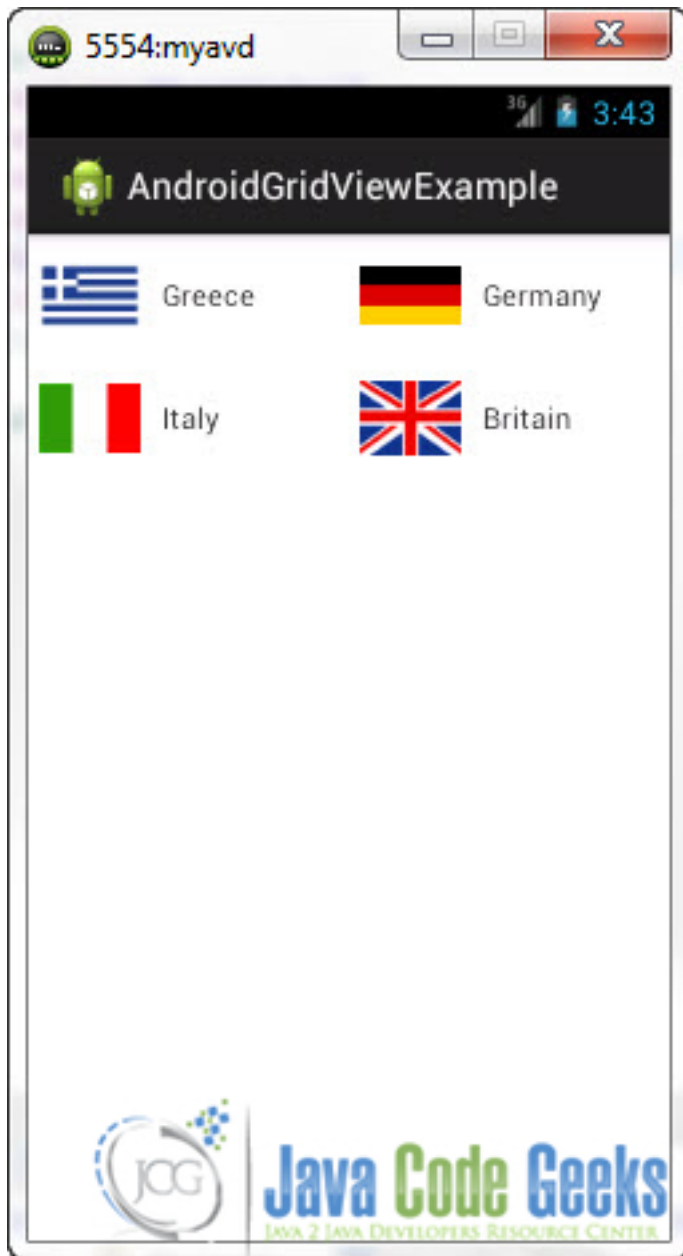
01  package com.javacodegeeks.android.androidgridviewexample;
02
03  import android.app.Activity;
04  import android.os.Bundle;
05  import android.widget.AdapterView;
06  import android.widget.GridView;
07  import android.widget.TextView;
08  import android.widget.Toast;
09  import android.view.View;
10  import android.widget.AdapterView.OnItemClickListener;
11
12  public class MainActivity extends Activity {
13
14      GridView gridView;
15
16      static final String[] MOBILE_OS = new String[] {
17  "Greece", "Germany", "Italy", "Britain" };
18
19      @Override
20      public void onCreate(Bundle savedInstanceState) {
21
22          super.onCreate(savedInstanceState);
23          setContentView(R.layout.main);
24
25          gridView = (GridView) findViewById(R.id.gridView);
26
27          gridView.setAdapter(new MyAdapter(this,
28  MOBILE_OS));
29
30          gridView.setOnItemClickListener(new
31  OnItemClickListener() {
32      public void onItemClick(AdapterView parent,
33  View v,
34      int position, long id) {
35          Toast.makeText(
36              getApplicationContext(),

```

```
33                                     ((TextView)
v.findViewById(R.id.label)).getText(),
Toast.LENGTH_SHORT).show();
34
35                                     }
36                                     });
37
38     }
39
40 }
```

#### 4. Run the application

This is the main screen of our Application:



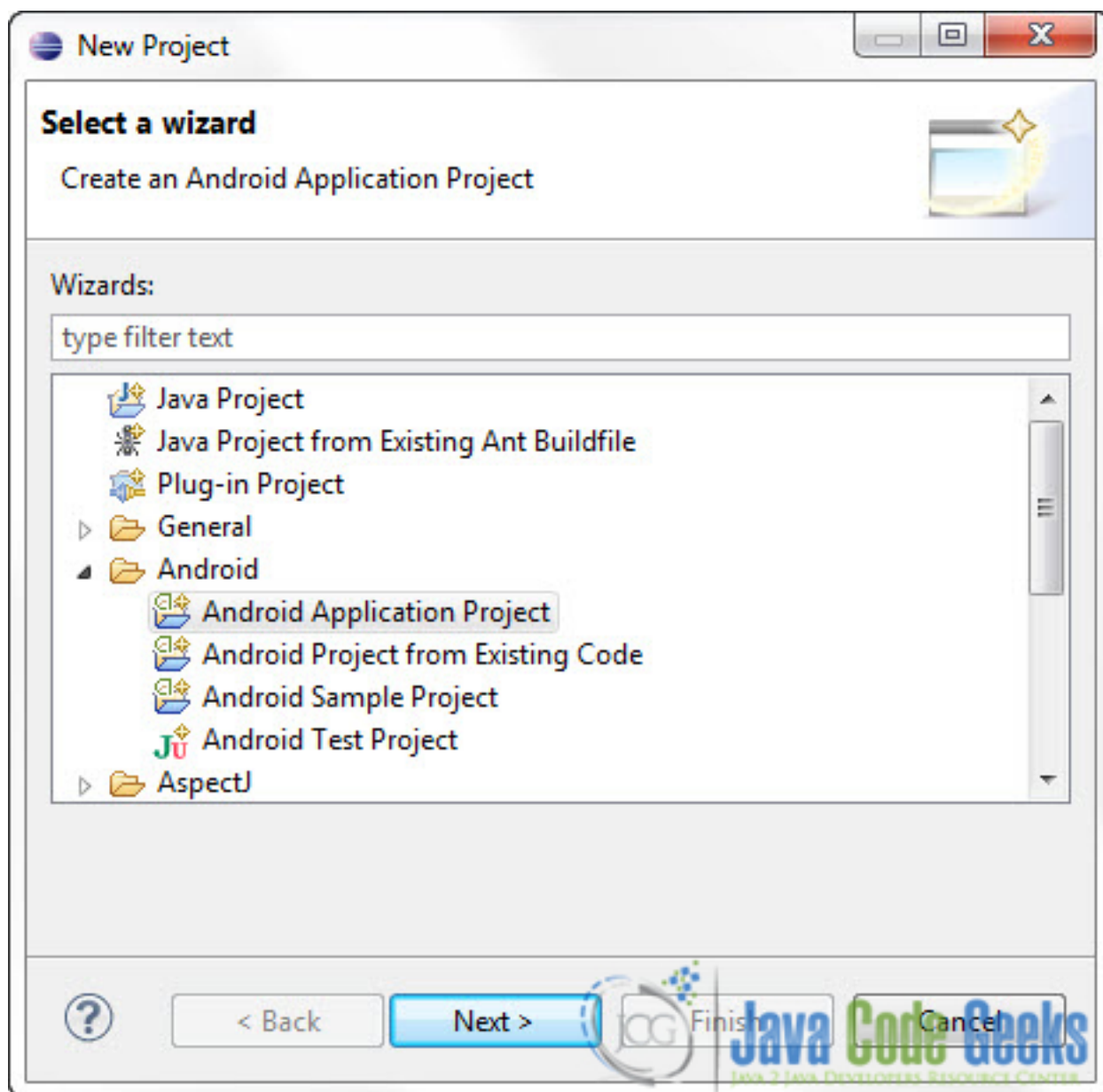
Now, when you click on an item:



### Download Eclipse Project

This was the second part of the Android GridView Example. Download the Eclipse Project of the second part of this tutorial: [AndroidGridViewExample\\_2.zip](http://a5e2fba00d8bcb729d89839f.javacodegeeks.netdna-cdn.com/wp-content/uploads/2013/01/AndroidGridViewExample_2.zip) ([http://a5e2fba00d8bcb729d89839f.javacodegeeks.netdna-cdn.com/wp-content/uploads/2013/01/AndroidGridViewExample\\_2.zip](http://a5e2fba00d8bcb729d89839f.javacodegeeks.netdna-cdn.com/wp-content/uploads/2013/01/AndroidGridViewExample_2.zip))





You have to specify the Application Name, the Project Name and the Package name in the appropriate text fields and then click Next.