

Android Spinner (Drop Down List) Example



Posted on November 29, 2011 , Last modified : November 7, 2012

By [mkyong](#)

In Android, you can use "[android.widget.Spinner](#)" class to render a dropdown box selection list.

Note

Spinner is a widget similar to a drop-down list for selecting items.

In this tutorial, we show you how to do the following tasks :

1. Render a Spinner in XML, and load the selection items via XML file also.
2. Render another Spinner in XML, and load the selection items via code dynamically.
3. Attach a listener on Spinner, fire when user select a value in Spinner.
4. Render and attach a listener on a normal button, fire when user click on it, and it will display selected value of Spinner.

P.S This project is developed in Eclipse 3.7, and tested with Android 2.3.3.

1. List of Items in Spinner

Open "[res/values/strings.xml](#)" file, define the list of items that will display in Spinner (dropdown list).

File : [res/values/strings.xml](#)

```
<?xml version="1.0" encoding="utf-8"?>
<resources>

    <string name="app_name">MyAndroidApp</string>
    <string name="country_prompt">Choose a country</string>

    <string-array name="country_arrays">
        <item>Malaysia</item>
        <item>United States</item>
        <item>Indonesia</item>
        <item>France</item>
        <item>Italy</item>
        <item>Singapore</item>
        <item>New Zealand</item>
        <item>India</item>
    </string-array>

</resources>
```

Free shipping
Cash on delivery*

MEN'S TEES
UNDER ₹299 + EXTRA 25% OFF

PROMO CODE: SHOP25

SHOP NOW >

2. Spinner (DropDown List)

Open "[res/layout/main.xml](#)" file, add two spinner components and a button.

1. In "spinner1", the "`android:entries`" represents the selection items in spinner.
2. In "spinner2", the selection items will be defined in code later.

LEARN **BIG DATA**
FROM THE
EXPERTS

ENROLL NOW

simplylearn

LEARN **BIG DATA**
FROM THE
EXPERTS

ENROLL NOW

simplylearn

Recent Posts

- [Log4j hello world example](#)
- [Spring Security + Hibernate Annotation Example](#)
- [Spring Security + Hibernate XML Example](#)
- [java.lang.ClassNotFoundException: org.hibernate.service.jta.platform.spi.JtaPlatform](#)
- [Spring Security : Encoded password does not look like BCrypt](#)

Popular Tutorials

- [Android Tutorial](#)
- [JSF 2.0 Tutorial](#)

File : res/layout/main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >

    <Spinner
        android:id="@+id/spinner1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:entries="@array/country_arrays"
        android:prompt="@string/country_prompt" />

    <Spinner
        android:id="@+id/spinner2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />

    <Button
        android:id="@+id/btnSubmit"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Submit" />

</LinearLayout>
```



3. Code Code

Read the code and also code's comment, it should be self-explanatory.

File : MyAndroidAppActivity.java

```
package com.mkyong.android;

import java.util.ArrayList;
import java.util.List;
import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.Spinner;
import android.widget.Toast;

public class MyAndroidAppActivity extends Activity {

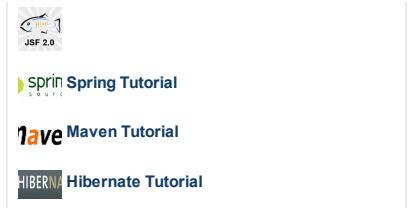
    private Spinner spinner1, spinner2;
    private Button btnSubmit;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        addItemOnSpinner2();
        addListenerOnButton();
        addListenerOnSpinnerItemSelection();
    }

    // add items into spinner dynamically
    public void addItemOnSpinner2() {

        spinner2 = (Spinner) findViewById(R.id.spinner2);
        List<String> list = new ArrayList<String>();
```




```

list.add("list 1");
list.add("list 2");
list.add("list 3");
ArrayAdapter<String> dataAdapter = new ArrayAdapter<String>(this,
    android.R.layout.simple_spinner_item, list);
dataAdapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
spinner2.setAdapter(dataAdapter);
}

public void addListenerOnSpinnerItemSelection() {
    spinner1 = (Spinner) findViewById(R.id.spinner1);
    spinner1.setOnItemSelectedListener(new CustomOnItemSelectedListener());
}

// get the selected dropdown list value
public void addListenerOnButton() {

    spinner1 = (Spinner) findViewById(R.id.spinner1);
    spinner2 = (Spinner) findViewById(R.id.spinner2);
    btnSubmit = (Button) findViewById(R.id.btnSubmit);

    btnSubmit.setOnClickListener(new OnClickListener() {

        @Override
        public void onClick(View v) {

            Toast.makeText(MyAndroidAppActivity.this,
                "OnClickListener : " +
                "\nSpinner 1 : " + String.valueOf(spinner1.getSelectedItem()) +
                "\nSpinner 2 : " + String.valueOf(spinner2.getSelectedItem()),
                Toast.LENGTH_SHORT).show();

        }

    });
}
}

```

File : CustomOnItemSelectedListener.java

```

package com.mkyong.android;

import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.Toast;

public class CustomOnItemSelectedListener implements OnItemSelectedListener {

    public void onItemSelected(AdapterView<?> parent, View view, int pos, long id) {
        Toast.makeText(parent.getContext(),
            "OnItemSelectedListener : " + parent.getItemAtPosition(pos).toString(),
            Toast.LENGTH_SHORT).show();
    }

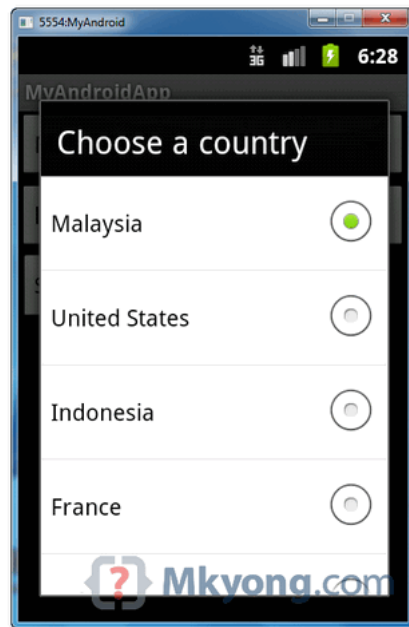
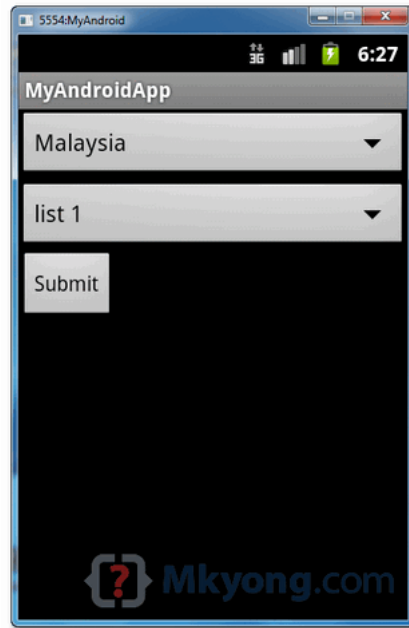
    @Override
    public void onNothingSelected(AdapterView<?> arg0) {
        // TODO Auto-generated method stub
    }
}

```

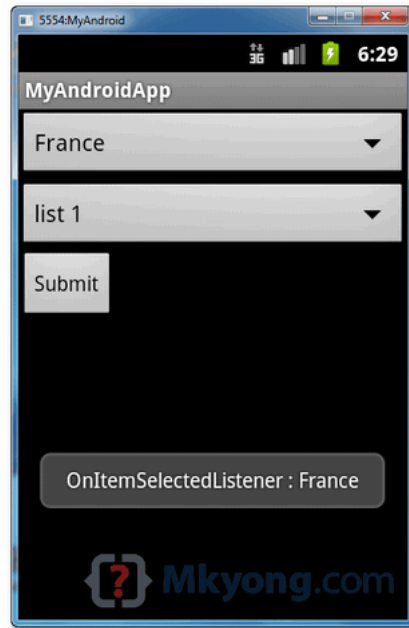
4. Demo

Run the application.

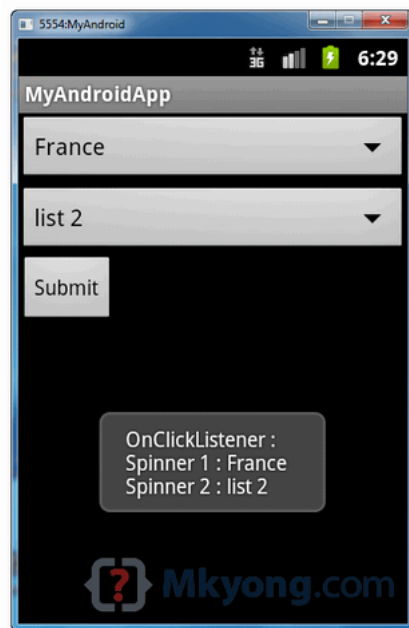
1. Result, two spinners are displayed :



2. Select "France" from spinner1, item selection listener is fired :



3. Select "list2" from spinner2, and click on the submit button :



Download Source Code

Download it – [Android-Spinner-DropDownList-Example.zip](#) (16 KB)

References

1. [Android Spinner JavaDoc](#)
2. [Android Spinner Example](#)

Tags : [android](#) [dropdown](#) [spinner](#)



mkyong

Founder of Mkyong.com, love Java and open source stuff. Follow him on [Twitter](#), or befriend him on [Facebook](#) or [Google Plus](#). If you like my tutorials, consider making a [donation to the charity](#), thanks.

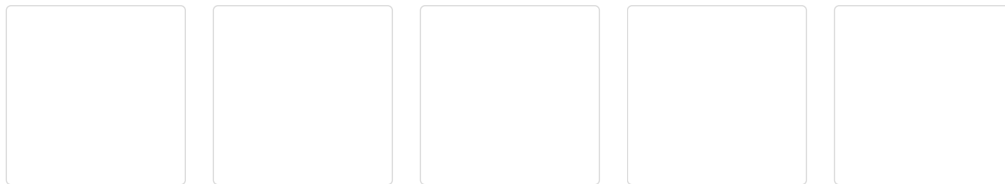
Related Posts

- ▶ [Android Tutorial](#)
- ▶ [How to send Email in Android](#)
- ▶ [How to send SMS message in Android](#)
- ▶ [Android : how to check if device has camera](#)
- ▶ [How to turn on/off camera LED / flashlight in Android](#)

Popular Posts

- ▶ [Top 8 Java People You Should Know](#)
- ▶ [Top 20 Java Websites](#)
- ▶ [Top 5 Free Java eBooks](#)
- ▶ [Top 10 Java Regular Expression Examples](#)
- ▶ [Top 5 Open Source Q&A Systems](#)

You might also like following tutorials :



All Available Tutorials

Java Core Technologies :
 Java I/O, Java RegEx, Java XML, Java JSON, JDBC, Java Misc
 J2EE Frameworks :
 Hibernate, JSF 2.0, Spring Core, Spring MVC, Spring Security, Spring MongoDB, Spring BatchApache Wicket, Struts 1.x, Struts 2.x
 Web Service :
 JAX-WS (SOAP), JAX-RS (REST)
 Build Tools :
 Maven, Archiva
 Unit Test Frameworks :
 JUnit, TestNG
 Others :
 Android, Google App Engine, jQuery, Java MongoDB, Quartz Scheduler

Favorites Links

Android Getting Started
 Google App Engine - Java
 DZone - Fresh Links
 Official Java EE 5 Tutorial
 Official Java EE 6 Tutorial
 Official Java EE 7 Tutorial
 Spring 2.5.x documentation
 Spring 3.2.x documentation
 Spring Security 3.2.x documentation
 Hibernate core 4.3 documentation
 Java SE 6.0 API documentation
 JSP home page
 JSF home page
 Eclipse IDE for Java developer
 Struts 1.3.x documentation
 Struts 2.3.x documentation
 Maven home page
 Maven central repository Search
 Ant home page

Friends & Links

Java Code Geeks
 PHP Tutorials
 TenthOfMarch
 Web Security Blog
 Web Development
 Cédric Beust (TestNG)

About Us

Mkyong.com is a weblog dedicated to Java/J2EE developers and Web Developers. We constantly publish useful tricks, tutorials on J2EE or web development.

All examples are simple, easy to read, and full source code available, and of course well tested in our development environment.

We're Social

Twitter - [Follow Me](#)
 Facebook - [Like Me](#)
 Google Plus - [Add Me](#)
 RSS - [Subscribe Me](#)

