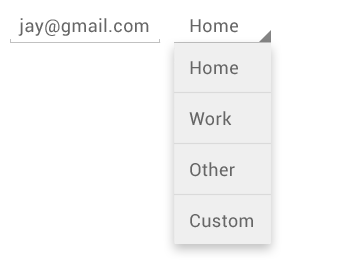
Spinners

Spinners provide a quick way to select one value from a set. In the default state, a spinner shows its currently selected value. Touching the spinner displays a dropdown menu with all other available values, from which the user can select a new one.



You can add a spinner to your layout with the [Spinner](file:///G:\Studio\SDK\docs\reference\android\widget\Spinner.html) object. You should usually do so in your XML layout with a <Spinner> element. For example:

<Spinner  
    android:id="@+id/planets\_spinner"  
    android:layout\_width="fill\_parent"  
    android:layout\_height="wrap\_content" />

To populate the spinner with a list of choices, you then need to specify a [SpinnerAdapter](file:///G:\\Studio\\SDK\\docs\\reference\\android\\widget\\SpinnerAdapter.html) in your[Activity](file:///G:\Studio\SDK\docs\reference\android\app\Activity.html) or [Fragment](file:///G:\Studio\SDK\docs\reference\android\app\Fragment.html) source code.

## Populate the Spinner with User Choices

The choices you provide for the spinner can come from any source, but must be provided through an[SpinnerAdapter](file:///G:\Studio\SDK\docs\reference\android\widget\SpinnerAdapter.html), such as an [ArrayAdapter](file:///G:\\Studio\\SDK\\docs\\reference\\android\\widget\\ArrayAdapter.html) if the choices are available in an array or a[CursorAdapter](file:///G:\Studio\SDK\docs\reference\android\widget\CursorAdapter.html) if the choices are available from a database query.

For instance, if the available choices for your spinner are pre-determined, you can provide them with a string array defined in a [string resource file](file:///G:\\Studio\\SDK\\docs\\guide\\topics\\resources\\string-resource.html):

<?xml version="1.0" encoding="utf-8"?>  
<resources>  
    <string-array name="planets\_array">  
        <item>Mercury</item>  
        <item>Venus</item>  
        <item>Earth</item>  
        <item>Mars</item>  
        <item>Jupiter</item>  
        <item>Saturn</item>  
        <item>Uranus</item>  
        <item>Neptune</item>  
    </string-array>  
</resources>

With an array such as this one, you can use the following code in your [Activity](file:///G:\Studio\SDK\docs\reference\android\app\Activity.html) or [Fragment](file:///G:\Studio\SDK\docs\reference\android\app\Fragment.html) to supply the spinner with the array using an instance of [ArrayAdapter](file:///G:\\Studio\\SDK\\docs\\reference\\android\\widget\\ArrayAdapter.html):

Spinner spinner = (Spinner) findViewById(R.id.spinner);  
// Create an ArrayAdapter using the string array and a default spinner layout  
ArrayAdapter<CharSequence> adapter = ArrayAdapter.createFromResource(this,  
        R.array.planets\_array, android.R.layout.simple\_spinner\_item);  
// Specify the layout to use when the list of choices appears  
adapter.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);  
// Apply the adapter to the spinner  
spinner.setAdapter(adapter);

method. For example, here's an implementation of the interface in an [Activity](file:///G:\Studio\SDK\docs\reference\android\app\Activity.html):

public class SpinnerActivity extends Activity implements OnItemSelectedListener {  
    ...  
      
    public void onItemSelected(AdapterView<?> parent, View view,   
            int pos, long id) {  
        // An item was selected. You can retrieve the selected item using  
        // parent.getItemAtPosition(pos)  
    }  
  
    public void onNothingSelected(AdapterView<?> parent) {  
        // Another interface callback  
    }  
}

The [AdapterView.OnItemSelectedListener](file:///G:\\Studio\\SDK\\docs\\reference\\android\\widget\\AdapterView.OnItemSelectedListener.html) requires the [onItemSelected()](file:///G:\\Studio\\SDK\\docs\\reference\\android\\widget\\AdapterView.OnItemSelectedListener.html" \l "onItemSelected(android.widget.AdapterView<?>, android.view.View, int, long)) and [onNothingSelected()](file:///G:\\Studio\\SDK\\docs\\reference\\android\\widget\\AdapterView.OnItemSelectedListener.html" \l "onNothingSelected(android.widget.AdapterView<?>))callback methods.

Then you need to specify the interface implementation by calling [setOnItemSelectedListener()](file:///G:\\Studio\\SDK\\docs\\reference\\android\\widget\\AdapterView.html" \l "setOnItemSelectedListener(android.widget.AdapterView.OnItemSelectedListener)):

Spinner spinner = (Spinner) findViewById(R.id.spinner);  
spinner.setOnItemSelectedListener(this);

If you implement the [AdapterView.OnItemSelectedListener](file:///G:\\Studio\\SDK\\docs\\reference\\android\\widget\\AdapterView.OnItemSelectedListener.html) interface with your [Activity](file:///G:\Studio\SDK\docs\reference\android\app\Activity.html) or[Fragment](file:///G:\Studio\SDK\docs\reference\android\app\Fragment.html) (such as in the example above), you can pass this as the interface instance.