

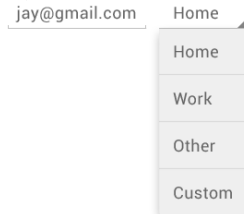


## La Clase Spinner

### CONCEPTOS

Un Spinner es un componente gráfico del tipo lista desplegable que muestra un elemento hijo a la vez y permite al usuario elegir uno de ellos. Los elementos del Spinner provienen del adaptador asociado con este componente.

En el estado predeterminado, se muestra su valor seleccionado en ese momento. Al digitar el Spinner se muestra un menú desplegable con todos los demás valores disponibles, de los cuales el usuario puede seleccionar uno nuevo.



Se puede agregar un control a la plantilla con el objeto Spinner. Para utilizarlo, se incluye en la plantilla XML una etiqueta `<Spinner>`. Por ejemplo:

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

Para rellenar el Spinner con una lista de opciones es necesario especificar un `SpinnerAdapter` en la actividad, o el código fuente de un fragmento.

### Los elementos del Spinner.

Los elementos que se agregan al Spinner pueden venir de cualquier fuente, pero deben proporcionarse a través de un `SpinnerAdapter`, por ejemplo un `ArrayAdapter` si las opciones están disponibles en un arreglo o en un `CursorAdapter` si las opciones están disponibles a partir de una consulta a una base de datos.

Si las opciones disponibles están predeterminadas, se pueden proporcionar con un arreglo de cadenas definido en el archivo `strings.xml`:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string-array name="planetas_array">
        <item>Mercurio</item>
        <item>Venus</item>
        <item>Tierra</item>
        <item>Marte</item>
        <item>Jupiter</item>
        <item>Saturno</item>
        <item>Urano</item>
        <item>Neptuno</item>
    </string-array>
</resources>
```

Con un arreglo como éste, se puede utilizar el siguiente código en la Activity o Fragment para abastecer al Spinner con el arreglo mediante una instancia de `ArrayAdapter`:

```
Spinner spinner = (Spinner) findViewById(R.id.spinner);
// Crear un ArrayAdapter usando el string array y un spinner layout dado.
ArrayAdapter<CharSequence> adapter = ArrayAdapter.createFromResource(this,
    R.array.planetas_array, android.R.layout.simple_spinner_item);
```



```
// Especificar la plantilla cuando se muestra la lista de opciones
adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
// Apply the adapter to the spinner
spinner.setAdapter(adapter);
```

El método `createFromResource()` permite crear un `ArrayAdapter` con el arreglo de cadenas. El tercer argumento del método es una plantilla que define la forma en que se muestra la opción seleccionada en el `Spinner`. La plantilla `simple_spinner_item` es proporcionado por la plataforma y es la predeterminada que se debe utilizar, a menos que se desee definir una propia plantilla para la apariencia del `Spinner`.

A continuación, se debe llamar a `setDropDownViewResource(int)` para especificar la plantilla que el adaptador debe utilizar para mostrar la lista de opciones (`simple_spinner_dropdown_item` es otra plantilla estándar definida por la plataforma). Llamar a `setAdapter()` para aplicar el adaptador al `Spinner`.

## Respuesta a las selecciones del usuario.

Cuando el usuario selecciona un elemento de la lista, el objeto `Spinner` recibe un evento `on-item-selected`.

Para definir el control del evento de selección, se implanta la interfaz `AdapterView.OnItemSelectedListener` y el correspondiente método `onItemSelected()`. Enseguida se muestra una implantación de la interfaz en una actividad:

```
public class SpinnerActivity extends Activity implements OnItemSelectedListener {
    :
    public void onItemSelected(AdapterView<?> parent, View view, int pos, long id) {
        // Se extrae el elemento seleccionado con parent.getItemAtPosition(pos)
    }
    public void onNothingSelected(AdapterView<?> parent) {
        // Otra llamada a la interface
    }
}
```

El `AdapterView.OnItemSelectedListener` necesita los métodos `onItemSelected()` y `onNothingSelected()`. Por tanto, se requiere especificar la implantación de la interfaz invocando a `setOnItemSelectedListener()`:

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

Si se implanta la interfaz `AdapterView.OnItemSelectedListener` en la `Activity` o `Fragment` (como en el ejemplo anterior), se puede pasar esto como la instancia de la interfaz.

## DESARROLLO

### EJEMPLO 1.

En este ejemplo, las opciones del `Spinner` se toman de un arreglo especificado en el archivo `strings.xml`.

**Paso 1.** Crear un proyecto `Spinner1`. En la carpeta `java/com.example.mipaquete`, abrir y modificar el archivo Java predeterminado `MainActivity.java`, con el siguiente código:

```
import android.os.Bundle;
import android.app.Activity;
import android.view.*;
import android.widget.*;
import android.widget.AdapterView.*;
public class MainActivity extends Activity{
    Spinner s;
    public void onCreate(Bundle b){
        super.onCreate(b);
```



```

setContentView(R.layout.activity_main);
s = (Spinner) findViewById(R.id.xsp);
s.setOnItemSelectedListener(new OnItemSelectedListener(){
    public void onItemSelected(AdapterView<?> av, View v, int i, long l){
        Toast.makeText(MainActivity.this, s.getSelectedItem().toString(),
                                Toast.LENGTH_LONG).show();
    }
    public void onNothingSelected(AdapterView<?> arg0){ }
});
}
}

```

**Paso 2.** En la carpeta `res/layout`, abrir y modificar el archivo XML predeterminado `activity_main.xml`, con el siguiente código:

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >
    <Spinner
        android:id="@+id/xsp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:prompt="@string/xsops"
        android:entries="@array/xops" />
</LinearLayout>

```

**Paso 3.** En la carpeta `res/values`, abrir y modificar el archivo XML predeterminado `strings.xml`, con el siguiente código:

```

<?xml version="1.0" encoding="utf-8" ?>
<resources>
    <string name="app_name">Ejemplo de Spinner </string>
    <string name="xsops">Opciones</string>
    <string name="action_settings">Ajustes</string>
    <string-array name="xops">
        <item>Opcion 1</item>
        <item>Opcion 2</item>
        <item>Opcion 3</item>
        <item>Opcion 4</item>
        <item>Opcion 5</item>
        <item>Opcion 6</item>
        <item>Opcion 7</item>
        <item>Opcion 8</item>
        <item>Opcion 9</item>
        <item>Opcion 10</item>
    </string-array>
</resources>

```

**Paso 4.** Por último, ejecutar la aplicación. Al inicio, se muestra solamente la primera opción del Spinner. Digitar sobre el primer elemento para desplegar toda la lista. Al seleccionar una opción, se muestra un Toast conteniendo el texto del elemento seleccionado. Al final, se pliega el Spinner y muestra de nuevo la primera opción.

