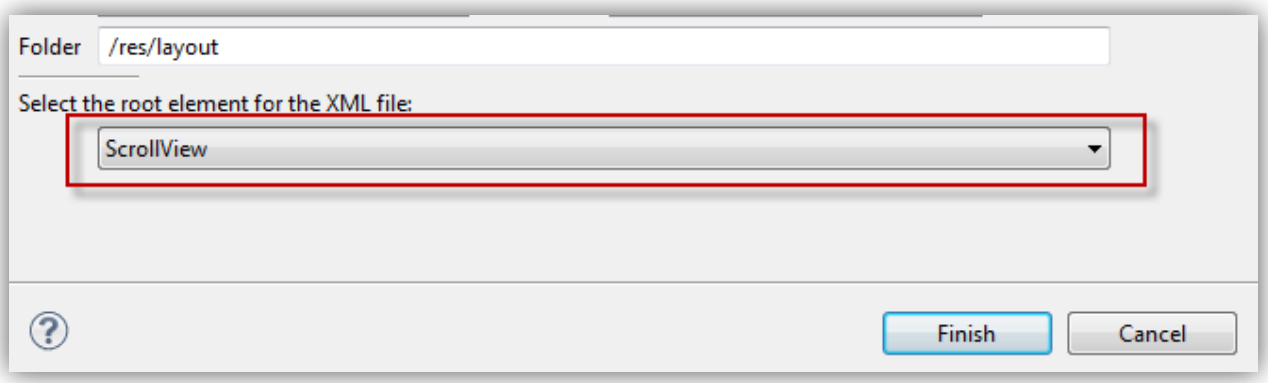
Lesson 5 – User Interfaces Part 4

This section will concentrate on the RadioButton, CheckBox, and ToggleButton controls.

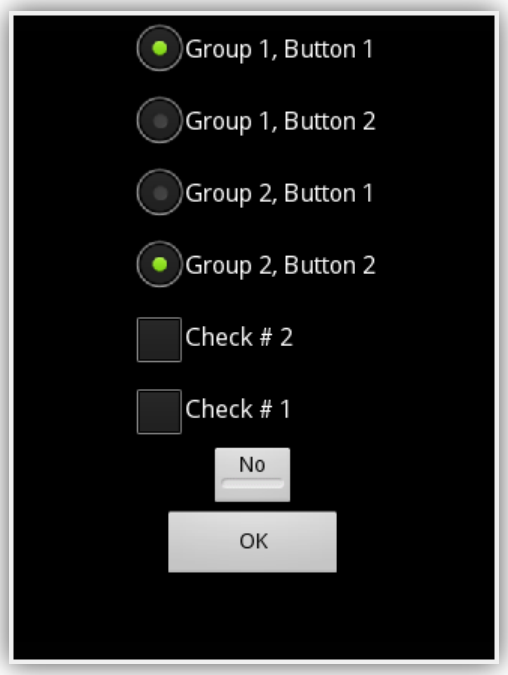
Open the existing Lesson5_UI project, and create a new Layout file (see Part 2, if you're not sure how) called **checkbox_radio_example.xml**. This one is going to use a ScrollView, so be sure to select that at the bottom of the dialog box:



Make sure that the scrollview's layout_width and height are set to either fill_parent or match_parent. Set it's layout_gravity to center_horizontal. Switch to XML mode, and add the following code:

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_height="match_parent" android:layout_width="match_parent"
        android:id="@+id/scrollView1" android:layout_gravity="center_horizontal">
        <LinearLayout android:layout_width="wrap_content"
                android:layout_height="wrap_content" android:layout_gravity="center_horizontal"
                android:orientation="vertical">
                <RadioGroup android:id="@+id/RadioGroup01"
                        android:layout_width="wrap_content" android:layout_height="wrap_content">
                        <RadioButton android:layout_width="wrap_content"
                                android:layout_height="wrap_content" android:id="@+id/rd01"
                                android:text="Group 1, Button 1" android:checked="true"></RadioButton>
                        <RadioButton android:layout_width="wrap_content"
                                android:layout_height="wrap_content" android:id="@+id/rd02"
                                android:text="Group 1, Button 2"></RadioButton>
                </RadioGroup>
                <RadioGroup android:id="@+id/RadioGroup02"
                        android:layout_width="wrap_content" android:layout_height="wrap_content">
                        <RadioButton android:layout_width="wrap_content"
                                android:layout_height="wrap_content" android:id="@+id/rd03"
                                android:text="Group 2, Button 1"></RadioButton>
                        <RadioButton android:layout_width="wrap_content"
                                android:layout_height="wrap_content" android:id="@+id/rd04"
                                android:text="Group 2, Button 2" android:checked="true"></RadioButton>
                </RadioGroup>
                <CheckBox android:layout_width="wrap_content"
                        android:layout_height="wrap_content" android:id="@+id/chk01"
                        android:text="Check # 2"></CheckBox>
                <CheckBox android:layout_width="wrap_content"
                        android:layout_height="wrap_content" android:id="@+id/chk02"
                        android:text="Check # 1"></CheckBox>
                <ToggleButton android:layout_width="wrap_content"
                        android:layout_height="wrap_content" android:layout_gravity="center_horizontal"
                        android:id="@+id/btnToggle" android:textOn="Yes"
                       android:textOff="No"></ToggleButton>
                <Button android:layout_height="wrap_content" android:id="@+id/btnCheckOK"
                        android:text="OK" android:layout_width="120dp"
                        android:layout_gravity="center_horizontal"></Button>
                <TextView android:layout_width="wrap_content"
                        android:layout_height="wrap_content" android:id="@+id/txtCheckOut"
                        android:layout_marginTop="10dp"></TextView>
        </LinearLayout>
</ScrollView>
```

Switch back to the Graphical Layout mode, and review the elements that you've just added. Your screen should look like this:



Save and close the XML file.

Open the Checkbox_Radio_Example.java file, and add the following code:

```
public class checkbox_radio_example extends Activity {
@Override
public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.checkbox_radio_example);

        final Button btn1 = (Button) this.findViewById(R.id.btnCheckOK);
        final ToggleButton btnToggle = (ToggleButton) findViewById(R.id.btnToggle);
        final CheckBox chk02 = (CheckBox) findViewById(R.id.chk02);
        final RadioButton rb01 = (RadioButton) findViewById(R.id.rd01);
        final RadioButton rb03 = (RadioButton) findViewById(R.id.rd03);
        final CheckBox chk01 = (CheckBox) findViewById(R.id.chk01);

        chk02.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View v) {
                        Toast.makeText(
                                        getApplicationContext(),
                                        "Check #2 is "
                                                        + (chk02.isChecked() ? "checked"
                                                                        : "not checked") + "!",
                                        Toast.LENGTH_SHORT).show();
                }
        });

        btn1.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View v) {

                        final TextView txtOut = (TextView) findViewById(R.id.txtCheckOut);
                        String sText = "";

                        if (rb01.isChecked()) {
                                sText = "Group 1, Button 1 is selected\n";
                        } else {
                                sText += "Group 1, Button 2 is selected\n";
                        }

                        if (rb03.isChecked()) {
                                sText += "Group 2, Button 1 is selected\n";
                        } else {
                                sText += "Group 2, Button 2 is selected\n";
                        }

                        if (chk01.isChecked()) {
                                sText += "Check # 1 is selected\n";
                        } else {
                                sText += "Check # 1 is not selected\n";
                        }

                        if (chk02.isChecked()) {
                                sText += "Check # 2 is selected\n";
                        } else {
                                sText += "Check # 2 is not selected\n";
                        }

                        if (btnToggle.isChecked()) {
                                sText += "Toggle Button is selected\n";
                        } else {
                                sText += "Toggle Button is not selected\n";
                        }

                        txtOut.setText(sText);
                }
        });

}

}
```

Run the application to see the results. Notice how the scrollview works. Vertical scrolling is available only when there's more information than can fit on the screen.

Review the code for the second checkbox. Notice that we can create an onClick Listener for a checkbox, the same way as for a button. We can do the same thing to radio buttons as well. This allows you to track selections as they happen, in case they influence other options further down on the screen. Also, note the use of the Ternary if statement assigned to the toast message within the chk02 click event. It tests the isChecked() property of the control, and returns "checked" if it's selected, and "not checked" if it's not.

Review how the two Radio Groups behave. Notice that you can only select one option from each of the groups.

Review the properties (in the XML file), code (in class file), and behavior (in emulator) of the Toggle Button. You'll notice that it behaves very similar to the checkbox control. Note where we specify the values that appear in the ToggleButton itself (Yes / No, in our case).

Review the video for this part of the lesson.