overview

This lab consists of several exercises that will comprise part of the homework assignment for Unit III. The complete assignment will be given in class on March 21, and will be due on April 3.

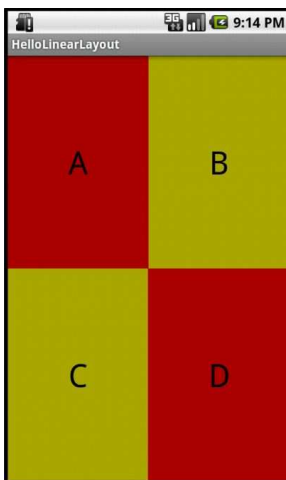Refer to the "Hello, Views" Tutorial, below:

`http://developer.android.com/resources/tutorials/views/index.html`

Exercises

Each of the following exercises is based on one of the sections of the "Hello, Views" Tutorial (URL above).

1. **Linear Layout**
   Run the example in the "Linear Layout" portion of the Tutorial.
   Modify the example so that a checkerboard is displayed, like this:
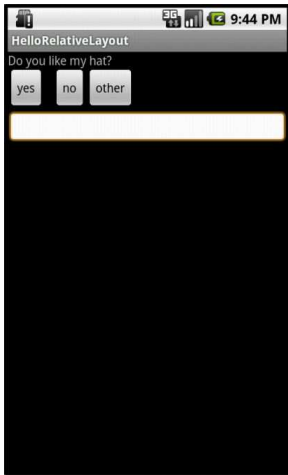

   Hints:
   - All the changes will be in `/res/layout/main.xml`.
   - To set the size of the text displayed, use the `textSize` property, like this:
     `android:textSize="15sp"`
   - To position the text in the middle of the `TextView`, both horizontally and vertically, use the `gravity` property, like this:
     `android:gravity="center_horizontal"`
   - To set the color of the text displayed, use the `textColor` property, like this:
     `android:textColor="#000000"`

2. **Relative Layout**

   Run the example in the "Relative Layout" portion of the Tutorial.
   Modify the example so that it looks like this:

   Hints:
   - As above, the changes will be in `/res/layout/main.xml`.
   - To control the position of an element in a Relative Layout, you indicate the relative position of the element you are aligning to others in the view. The following attributes might be helpful.
     These examples align the element to another element named `@id/label`:
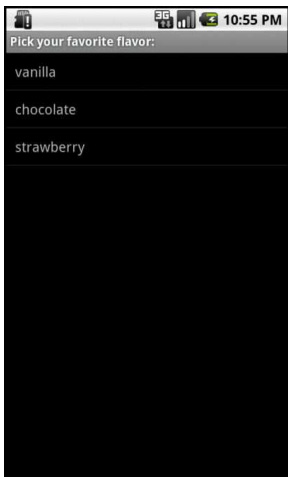   ```
     android:layout_toLeftOf="@id/label"
     android:layout_toRightOf="@id/label"
     android:layout_alignTop="@id/label"
     ```
     These examples align the element to its parent element:
     ```
     android:layout_alignParentLeft="true"
     android:layout_alignParentRight="true"
     ```

3. **List View**

   Run the example in the "List View" portion of the Tutorial.
   Modify the example so that it looks like this:

   

   *Hints:*
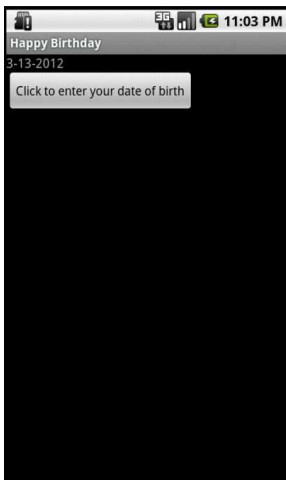   - The changes will be in two files: `/src/<your-package-name>/<your-file>.java` and `/res/values/strings.xml`.

- Note that you will need to make sure to import all the necessary Android packages in the Java source file:

```
import android.app.ListActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import android.widget.TextView;
import android.widget.Toast;
```

4. **Date Picker**

Run the example in the "Date Picker" portion of the Tutorial.

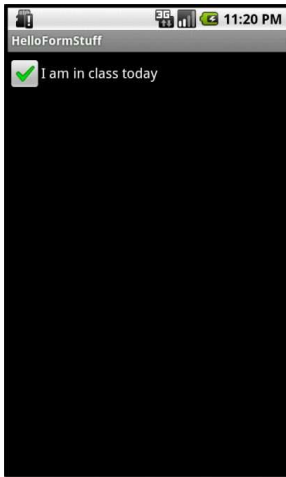Modify the example so that it looks like this:

Hints:

- The changes will be in three files: `/src/<your-package-name>/<your-file>.java` and `/res/layout/main.xml` and `/res/values/strings.xml`.
- Note that you will need to make sure to import all the necessary Android packages in the Java source file:

```
import java.util.Calendar;
import android.app.Activity;
import android.app.DatePickerDialog;
import android.app.Dialog;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.TextView;
```

5. **Form Stuff**

Run the example in the "Form Stuff" portion of the Tutorial.
Modify the example so that it looks like this:

Hints:
- The changes will be in two files: `/src/<your-package-name>/<your-file>.java` and `/res/layout/main.xml`.
- Note that you will need to make sure to import all the necessary Android packages in the Java source file:

```
import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.CheckBox;
import android.widget.Toast;
```

The above exercise uses a `Checkbox`.

Now expand your app to include the other interface widgets that are described in the Form Stuff tutorial:
- Custom Button
- Edit Text
- Radio Buttons
- Toggle Button
- Rating Bar