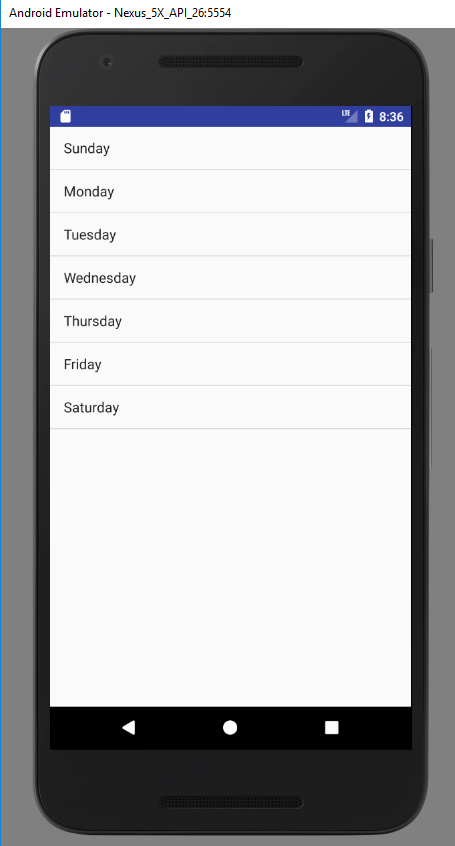
***Avoid cutting and pasting code from my lecture notes – better to retype it to avoid any issues with hidden characters or case issues.***

1. **Simple 1 column list**

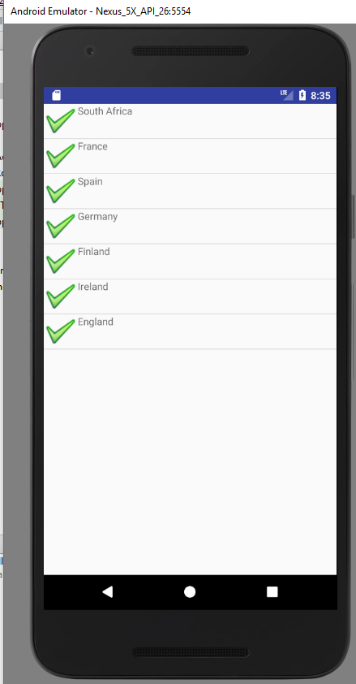
****

Create an app that displays a one column list of countries. When an item is clicked, it should display the country in a Toast message. This will involve

* An activity that extends ListActivity class
* A layout with a <Listview> attribute
* An array in your activity that contains the countries.
* An adapter (ArrayAdapter) that you create, using the Android Simple List Item 1 template for the row layout, as covered in class.
* to handle clicks – using the onListItemClick() callback as per lecture

Array contents: {"South Africa", "France", "Spain", “Samoa”, "Germany", "Finland", "Ireland", "England"}

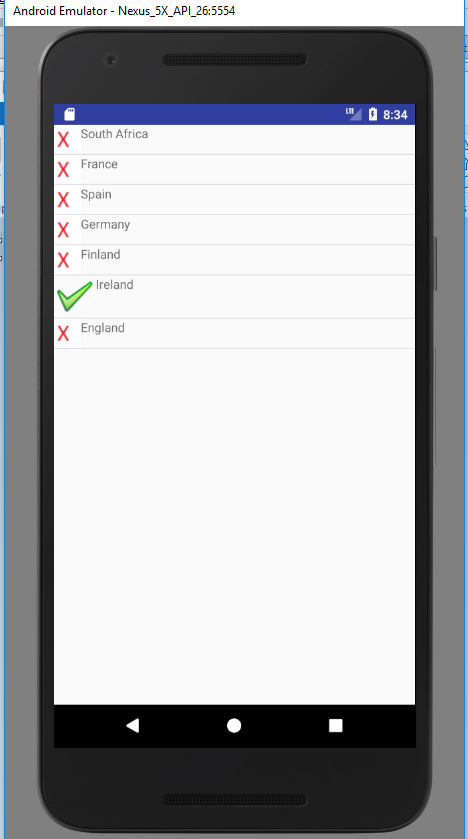
1. **Custom row layouts**



Create a list that has the list of countries showing, and with a a Tickbox on each row as well. The tickbox doesn’t actually do anything.. it just an illustrative second column. The tickbox is an image you can just paste into your res/drawable directory to store

* Use the image ok.png on webcourses
* To create your own row layout, add a second layout file – by convention called row.xml – but doesn’t have to be called that.
* Images can be put in XML layouts – as per example in the List lecture.
* hint: imageview tag used for mages
* hint 2: images (i.e. .png files) are held in the res/drawable directory.
* hint 3: the “android:src” attribute of imageview tag stores the name/ location of the image file.

**3. Custom adapters - dynamic row inflation**

****

So far, you’ve used an off the shelf adapter – ArrayAdapter in this case – to “inflate” each of the rows based on the data source. A *custom adapter* class is needed when you want to do anything dynamic in your rows, or that requires more than 2 columns.

Implement a custom adaptor to put the **Android icon** (already in your resources directory by default) on all rows, *except* the row that says Ireland, which will display a Tick icon.

* Create your custom adapter class (call it what you want) – It should inherit from ArrayAdapter (as this is the most “similar” adapter to inherit from)
* Create a constructor in your custom adapter class:

public WhateverYourAdapterIsCalled (Context context, int textViewResourceId, String[] objects)

{  
super(context, textViewResourceId, objects);

}

* Now override the getView() method which runs once for each row – it is responsible for “inflating” each row:

public View getView(int position, View convertView, ViewGroup parent)

{

LayoutInflater inflater=getLayoutInflater();

View row=inflater.inflate(R.layout.row, parent, false);

// row formatting:

// Retrieve your textView from your Row.xml using:

TextView label = (TextView)row.findViewById(your text view ID in your row layout);

// Retrieve your imageview from your Row.xml using:

ImageView myImage = (ImageView)row.findViewById(your image view ID from your row layout);

// Now format the textview with the current value of the array, and the image with the right image depending on whether Ireland is the current value or not

return row;

} // get View