Android UI components

Android View

- Most basic unit
- Occupy a rectangular space
- Responsible for drawing themselves
- Responsible for handling events

Rendering process

1. Measure – Get dimensions of each View
2. Layout – Positions each view
3. Draw – Draw UI using view onDraw()

View events

1. Users Interactions (Touch, keyboard)
2. Android Lifecycle changes
3. Application code

View events

- http://developer.android.com/reference/android/view/View.html

Events

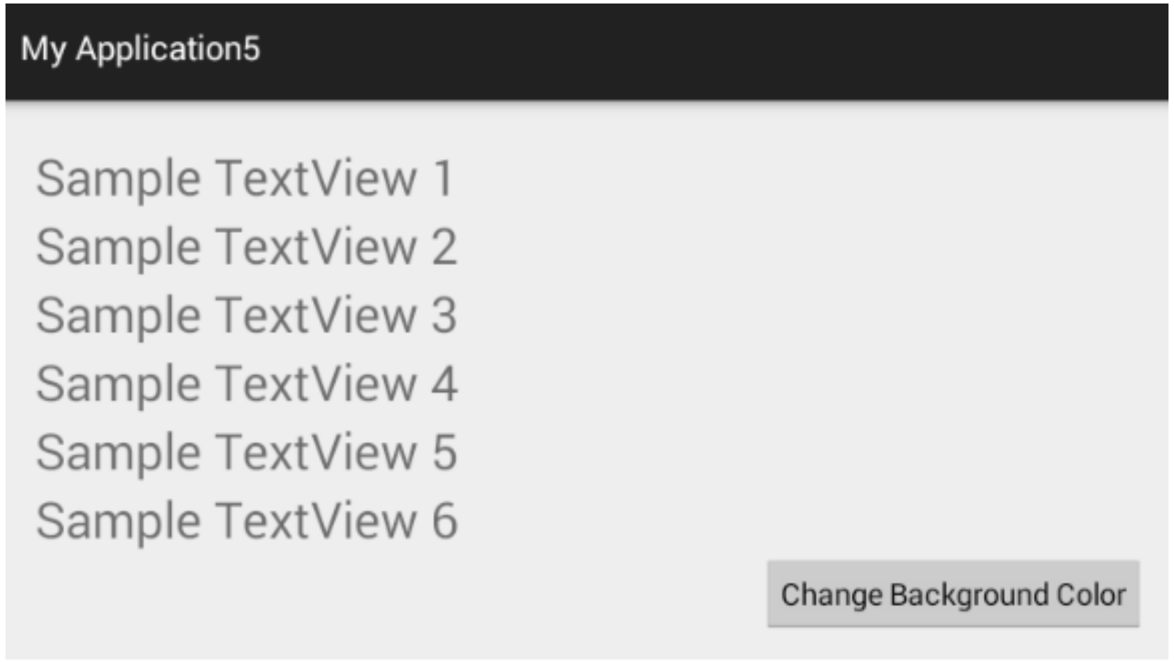
- View.OnClickListener.OnClick
- View.OnHoverListener.onHover

View common properties

- Size and Position
- Rotation
- Visibility
- Alpha
- Background
- Padding/Margin
- Clickable

TextView

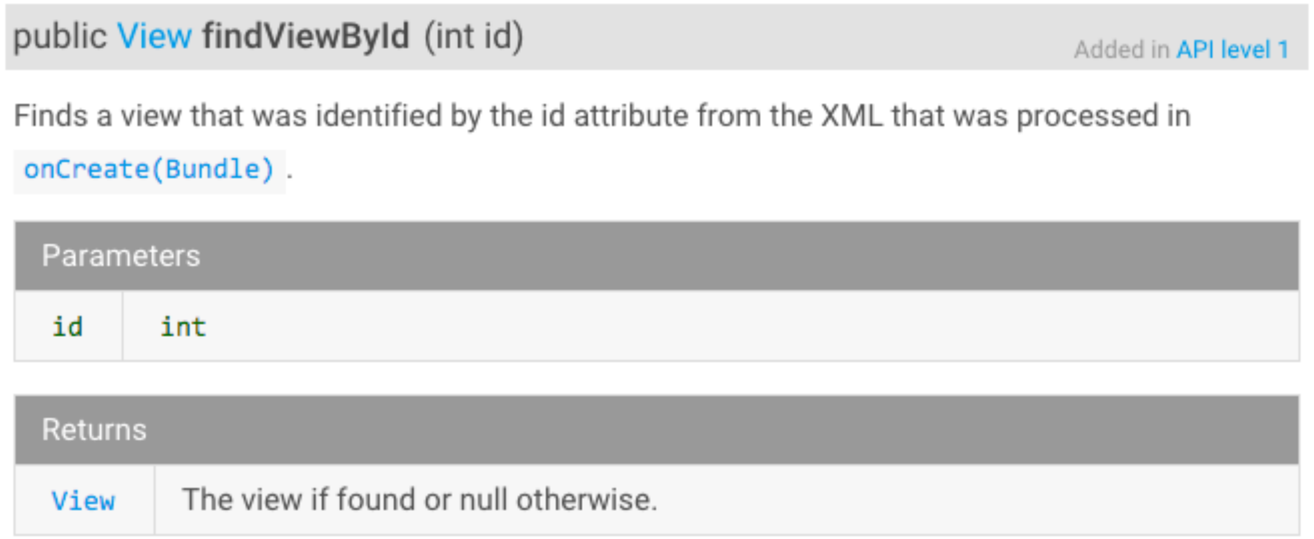
- http://developer.android.com/reference/android/widget/TextView.html



ImageView

```
<ImageView android:id="@+id/imageView1"
  android:layout_width="wrap_content"
  android:layout_height="wrap_content"
   android:src="@drawable/android"  />
```

Finding View in activity



public View **findViewById** (int id) Added in API level 1

Finds a view that was identified by the id attribute from the XML that was processed in `onCreate(Bundle)`.

Parameters	
`id`	`int`

Returns	
`View`	The view if found or null otherwise.

The Listener pattern

- User interface element
- Event type
- Listener interface
- Listener – Object that implements interface
- Registration method

Example - OnClickListener

Labs

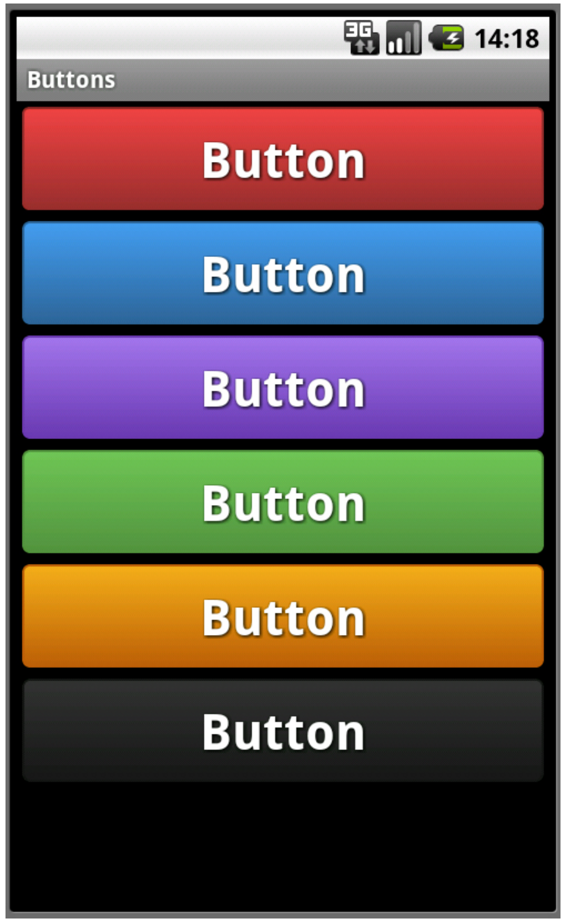
- Create both a TextView and ImageView add handlers for:
  - Onclick (Code)
  - onKeyDown(int, KeyEvent) (Xml)
  - onTouchEvent(MotionEvent) (Code, Xml)

View sub classes

- InputControls
- ViewGroup

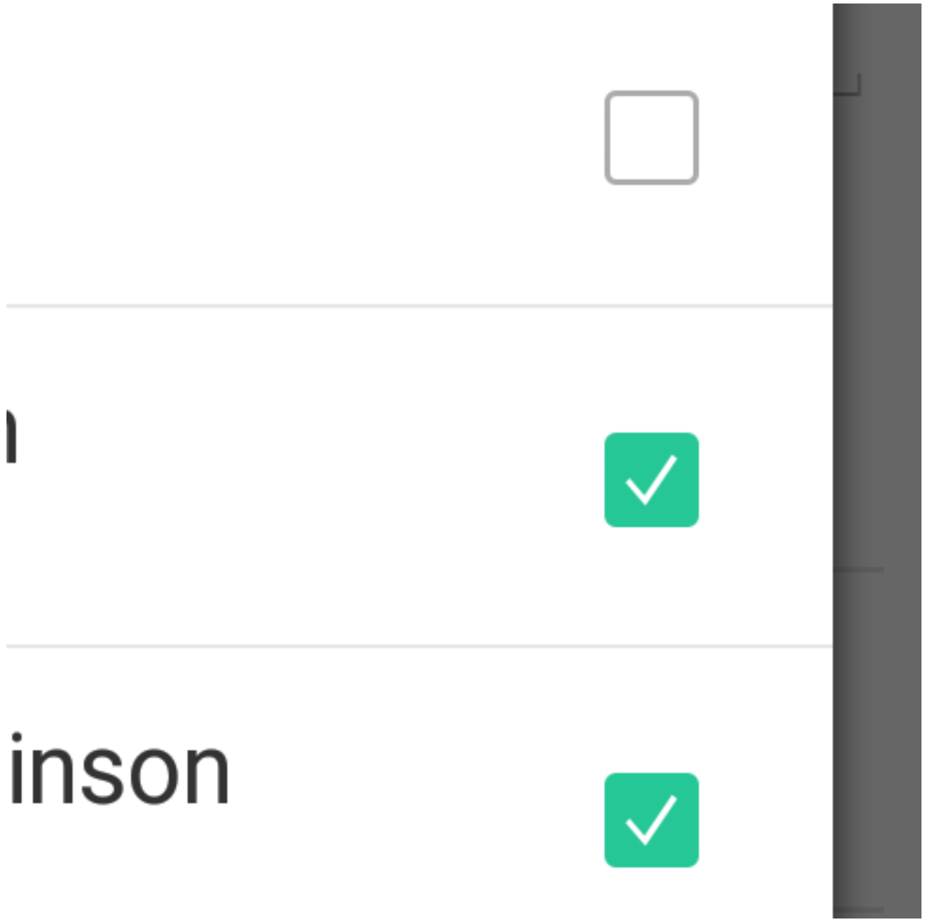
Input Controls

Input Controls - Button

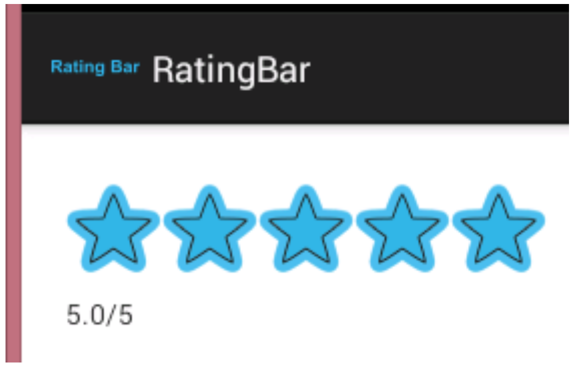


Input Controls - ToggleButton

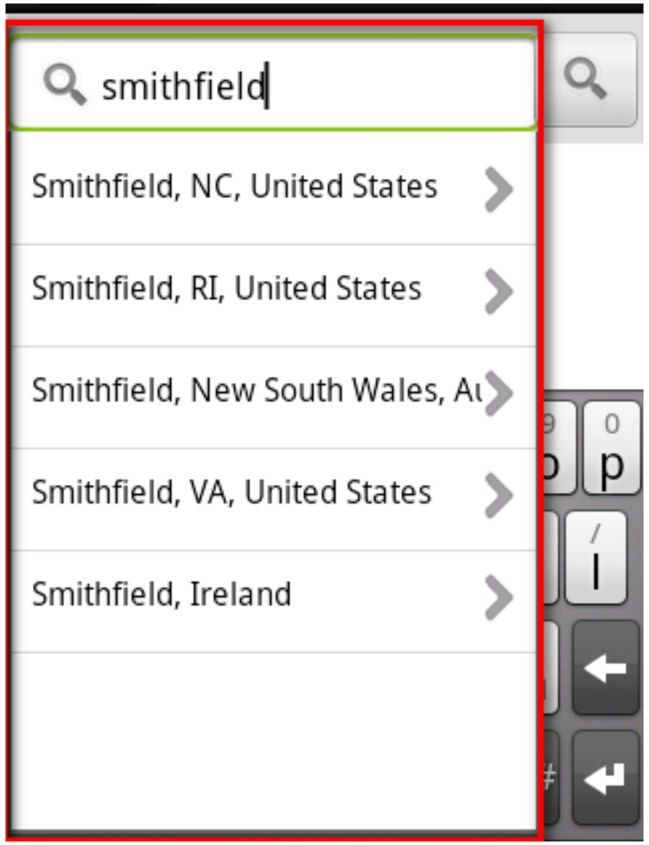
Input Controls - CheckBox



Input Control – RatingBar



Input Control - AutoCompleteTextView


smithfield

Smithfield, NC, United States

Smithfield, RI, United States

Smithfield, New South Wales, Au

Smithfield, VA, United States

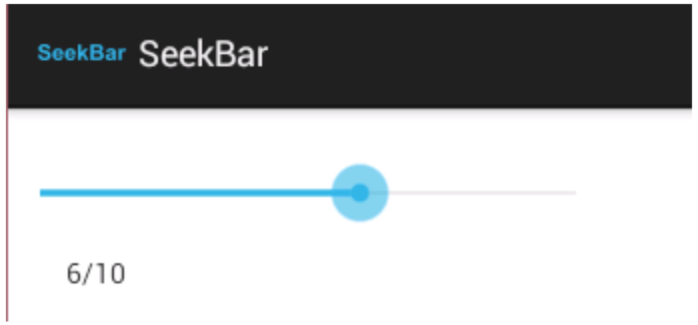
Smithfield, Ireland

```
public class CountriesActivity extends Activity
{
    protected void onCreate(Bundle icicle)
    {
        super.onCreate(icicle);
        setContentView(R.layout.countries);

        ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
                android.R.layout.simple_dropdown_item_1line, COUNTRIES);
        AutoCompleteTextView textView = (AutoCompleteTextView)
                findViewById(R.id.countries_list);
        textView.setAdapter(adapter);
    }

    private static final String[] COUNTRIES = new String[]
    {
        "Belgium", "France", "Italy", "Germany", "Spain"
    };
}
```

Input Controls - SeekBar



Lab

- Create a movie rate form
  - Movie selection with auto-complete
  - Rate Movie
  - Movie synopsis
  - Favorite actor selector

Keyboard and input handling

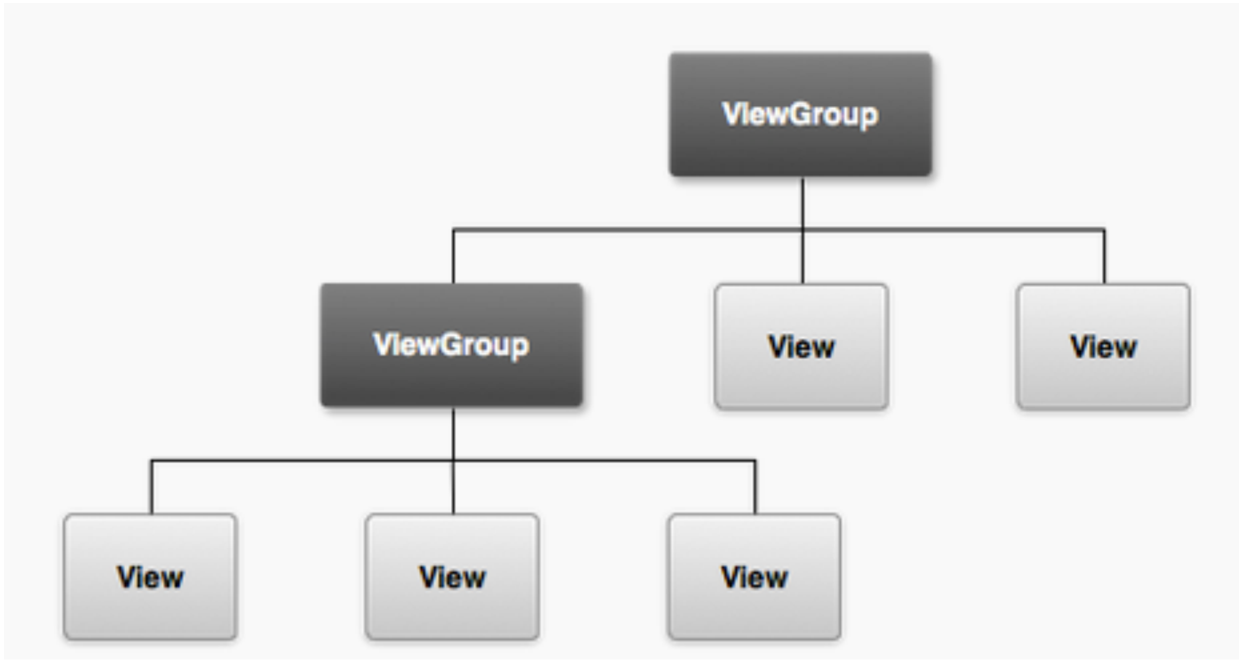
- Keyboard Behaviors
  - Autocorrect
- Keyboard Layout
  - Phone numbers, email
- Text Display Policies
- Action key
- Order Navigation

Lab Security Login

Create a login form with User/ pin - password

- Add inputType property
- Add nextFocusDown
- Add nextFocusRight
- Add setOnEditorActionLister on last field Go to second activity

View – View Group



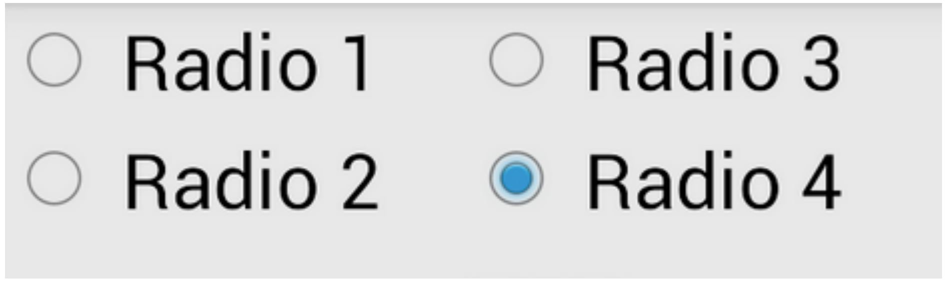
View Group

- Layouts
- RadioGroup
- TimePicker
- DatePicker
- WebView
- MapView

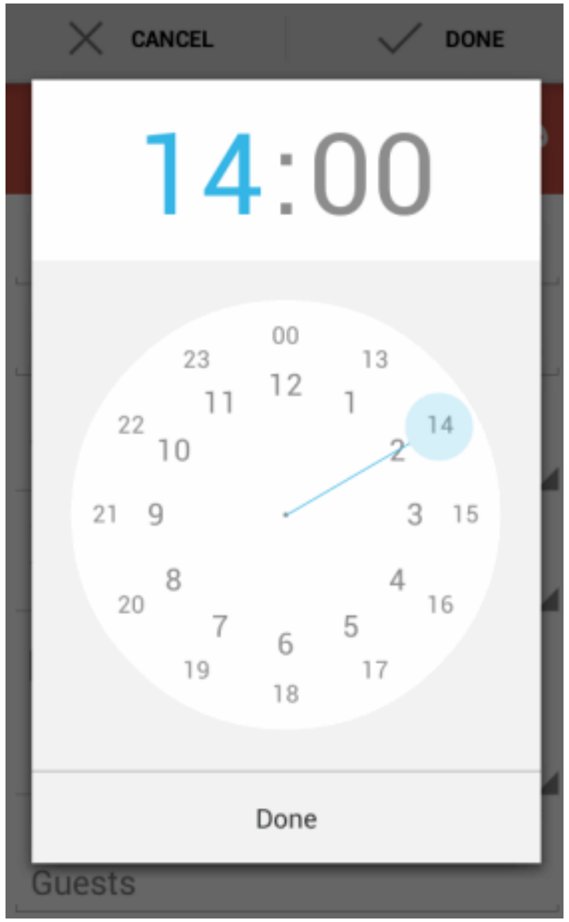
Layouts

- View Group is the invisible container. it holds View and View Group
- Example LineraLayout is the ViewGroup it contains Button(View),Other Layouts also.
- ViewGroup is the base class for Layouts.

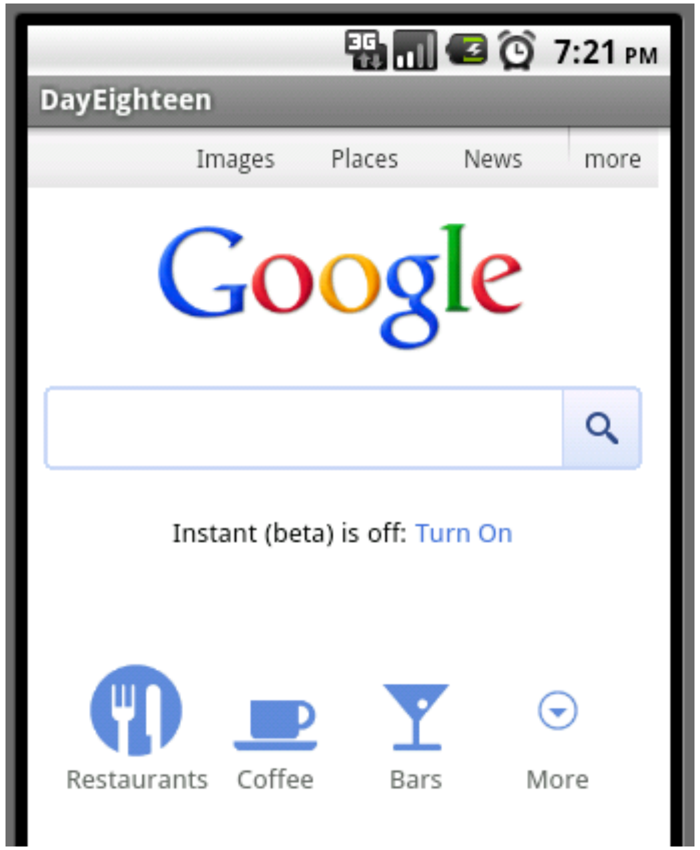
Radio Group



TimePicker

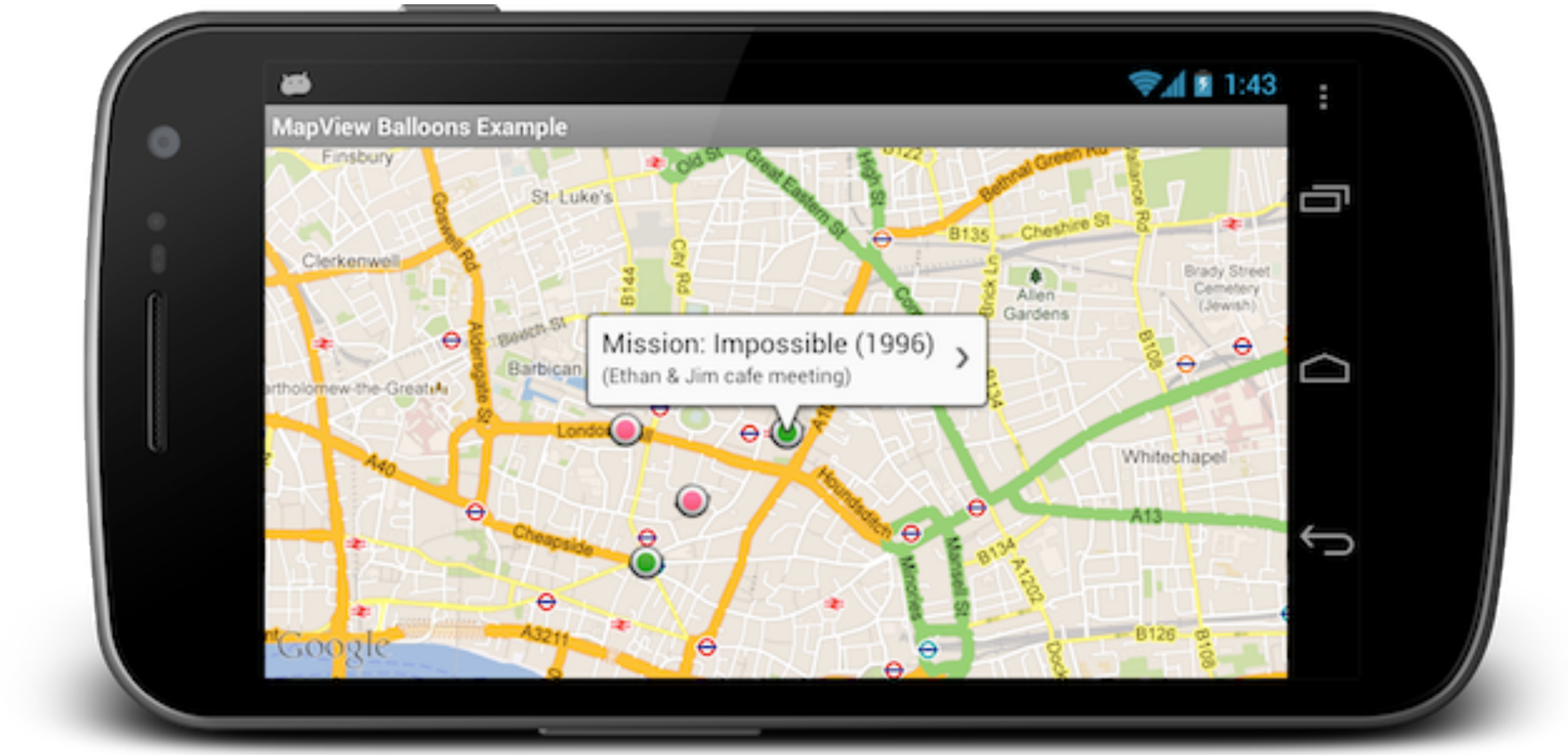

CANCEL
DONE
14:00
00
13
14
15
16
17
18
19
20
21
22
23
12
1
2
3
4
5
6
7
8
9
10
11
Done
Guests

WebView



```
webview.setWebViewClient(new WebViewClient());
webview.loadUrl("http://www.google.com");
```

MapView


1:43
MapView Balloons Example
Mission: Impossible (1996)
(Ethan & Jim cafe meeting)
Finsbury
St Luke's
Clerkenwell
Barbican
Whitechapel
Allen Gardens
Brady Street Cemetery (Jewish)
Cheapside
Houndsditch
Google

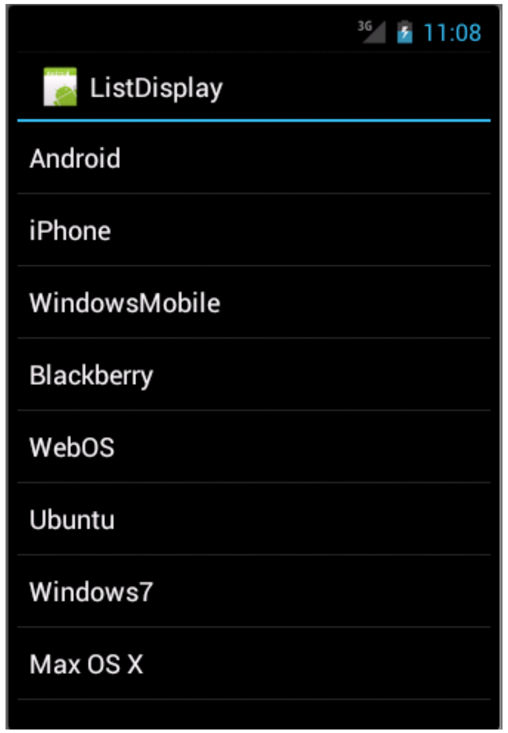
Lab

1. My first fake Android application
2. Activity for select a coordinates

Adapters Views

- Used for dynamic data
- Subclasses of ViewGroup
- Adapter is an interface for managing data
- Adapter view display the views provider by an Adapter

Adapter View – ListView



11:08

ListDisplay

Android

iPhone

WindowsMobile

Blackberry

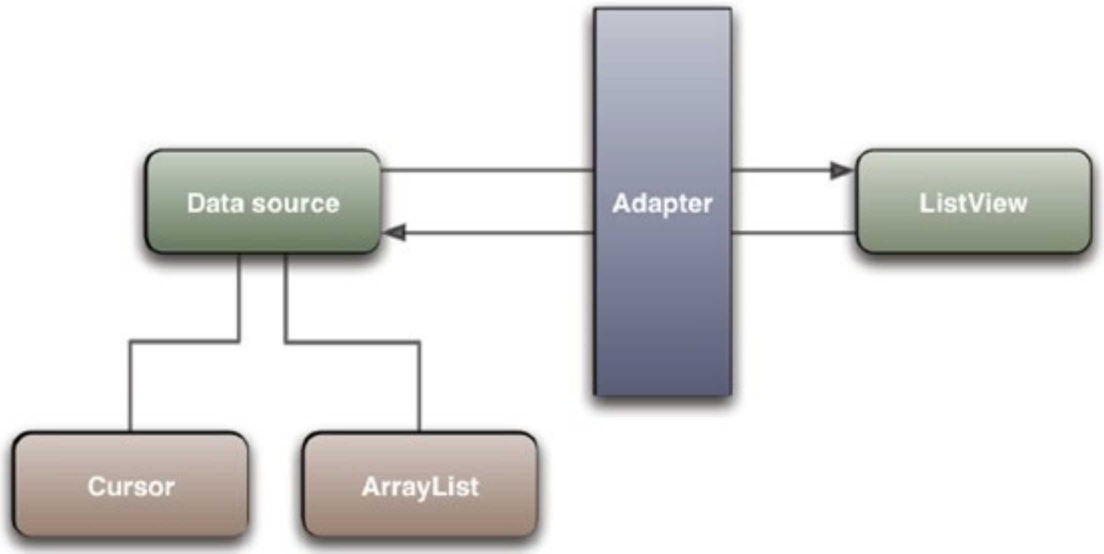
WebOS

Ubuntu

Windows7

Max OS X

Adapter View – ListView



Adapter View – ListView

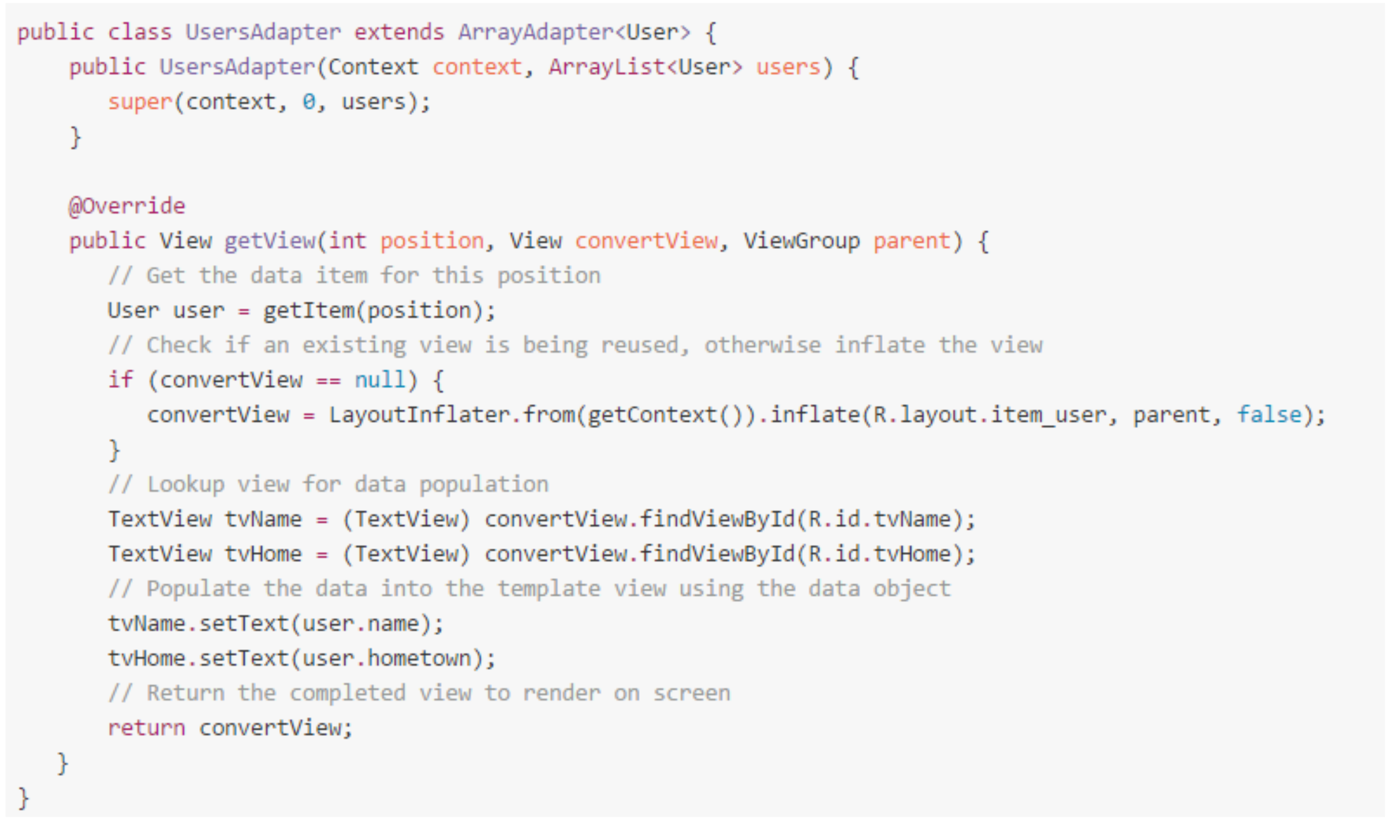
- ListActivity

```
ListViewAdapter adapter = new ListViewAdapter(context, layout, values);
setListAdapter(adapter);
```

Lab

- Create simple list view using List activity and ListAdapter

Custom Adapter

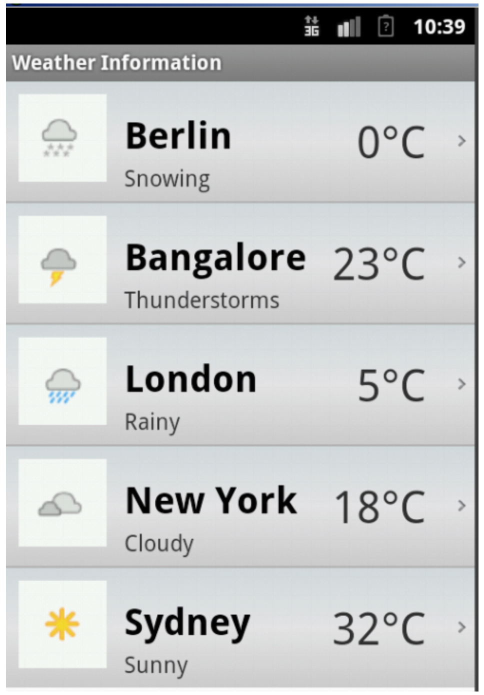


```java
public class UsersAdapter extends ArrayAdapter<User> {
    public UsersAdapter(Context context, ArrayList<User> users) {
       super(context, 0, users);
    }

    @Override
    public View getView(int position, View convertView, ViewGroup parent) {
       // Get the data item for this position
       User user = getItem(position);
       // Check if an existing view is being reused, otherwise inflate the view
       if (convertView == null) {
          convertView = LayoutInflater.from(getContext()).inflate(R.layout.item_user, parent, false);
       }
       // Lookup view for data population
       TextView tvName = (TextView) convertView.findViewById(R.id.tvName);
       TextView tvHome = (TextView) convertView.findViewById(R.id.tvHome);
       // Populate the data into the template view using the data object
       tvName.setText(user.name);
       tvHome.setText(user.hometown);
       // Return the completed view to render on screen
       return convertView;
   }
}
```

Lab

- Create a list view with the CityWeatherInformation data structure.



Weather Information

Berlin	Snowing	0°C
Bangalore	Thunderstorms	23°C
London	Rainy	5°C
New York	Cloudy	18°C
Sydney	Sunny	32°C