

Adding a Menu Page



Gill Cleeren

ARCHITECT

@gillcleeren www.snowball.be



Agenda



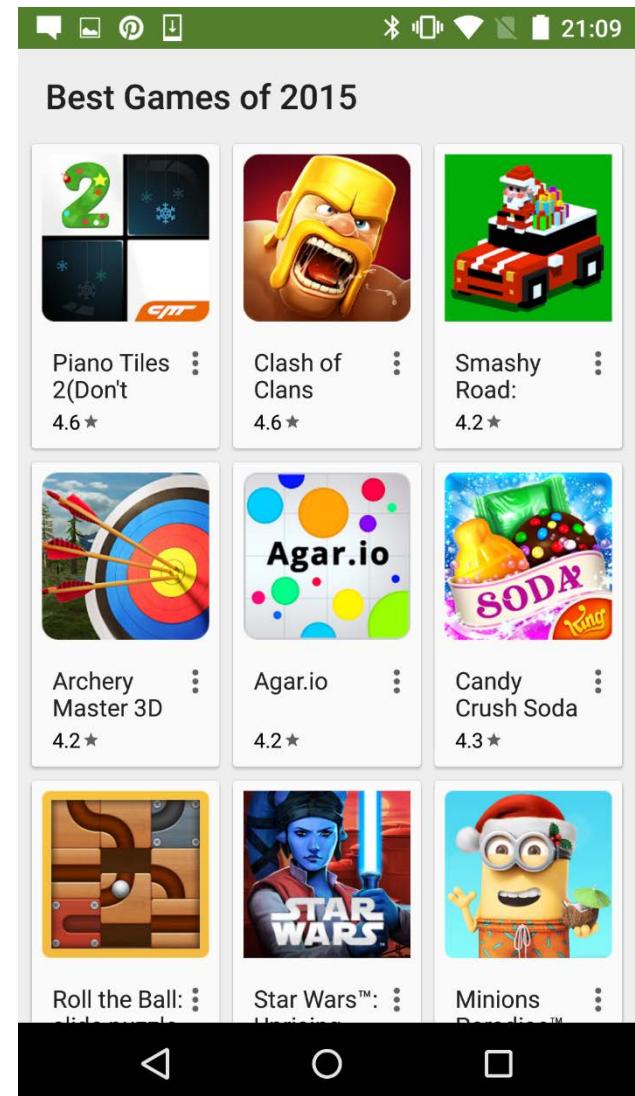
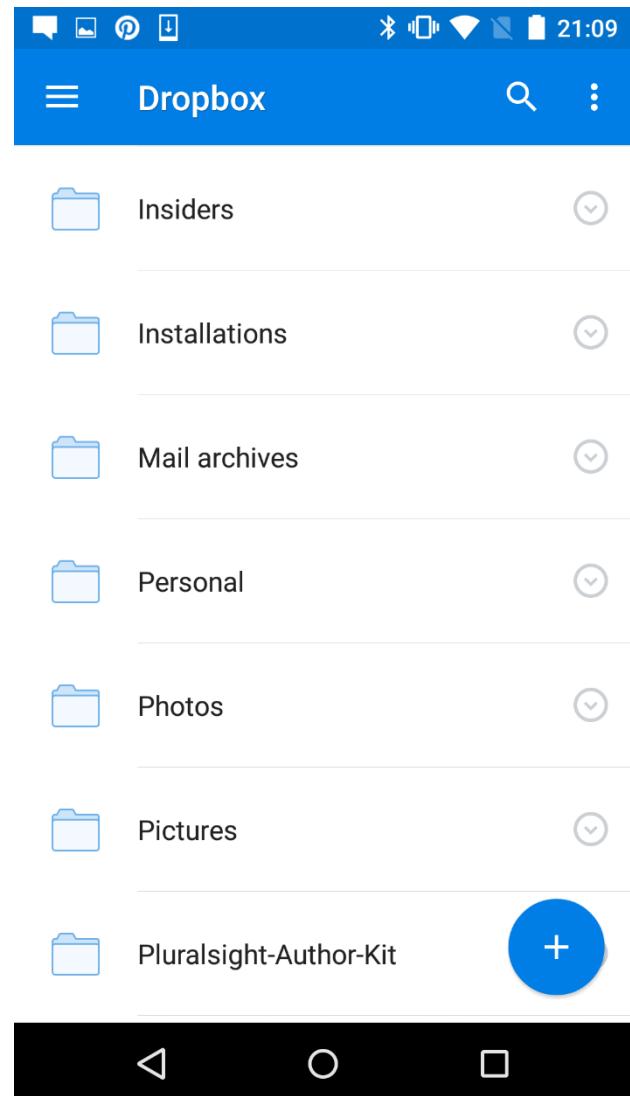
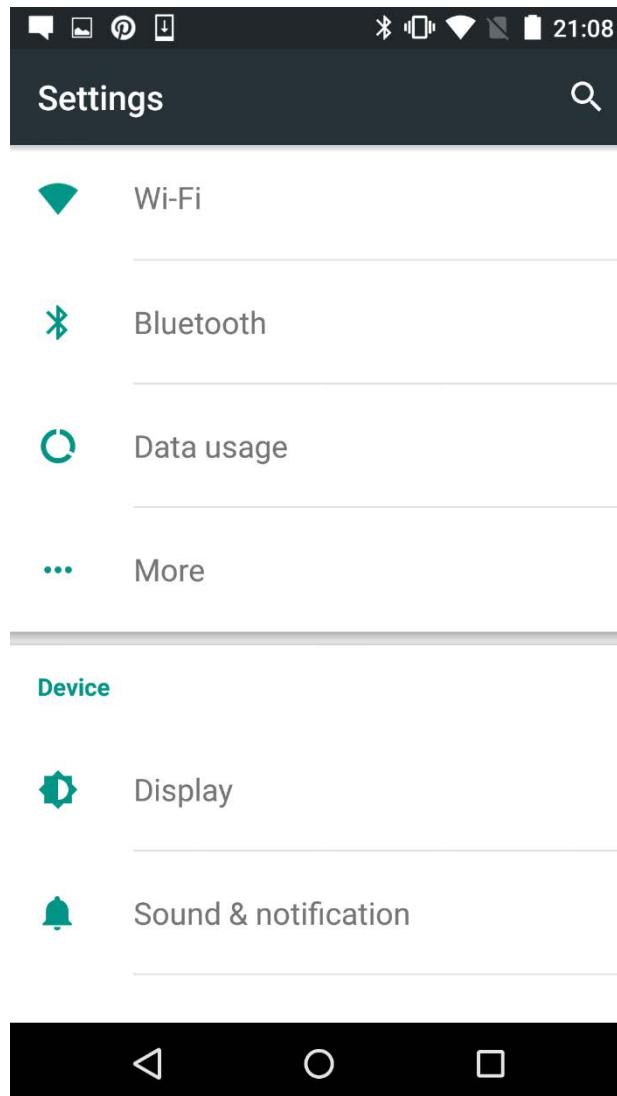
Creating the Menu Page
Changing the Row Style



Creating the Menu Page



Lists in Android



Building Blocks



ListView



Rows



Adapter



How Things Work

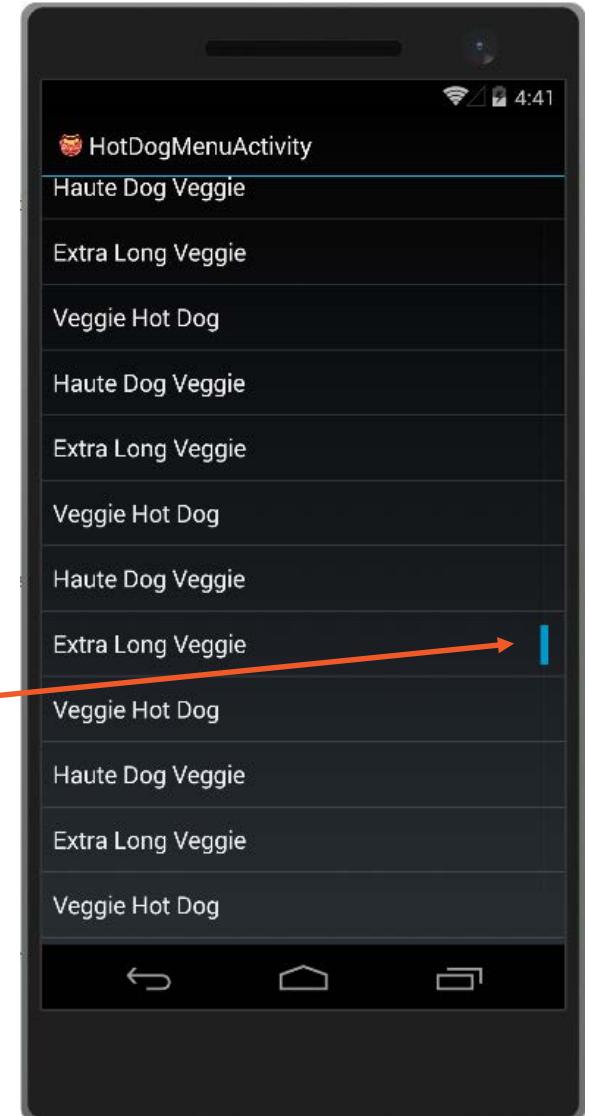


ListView Visuals



Rows

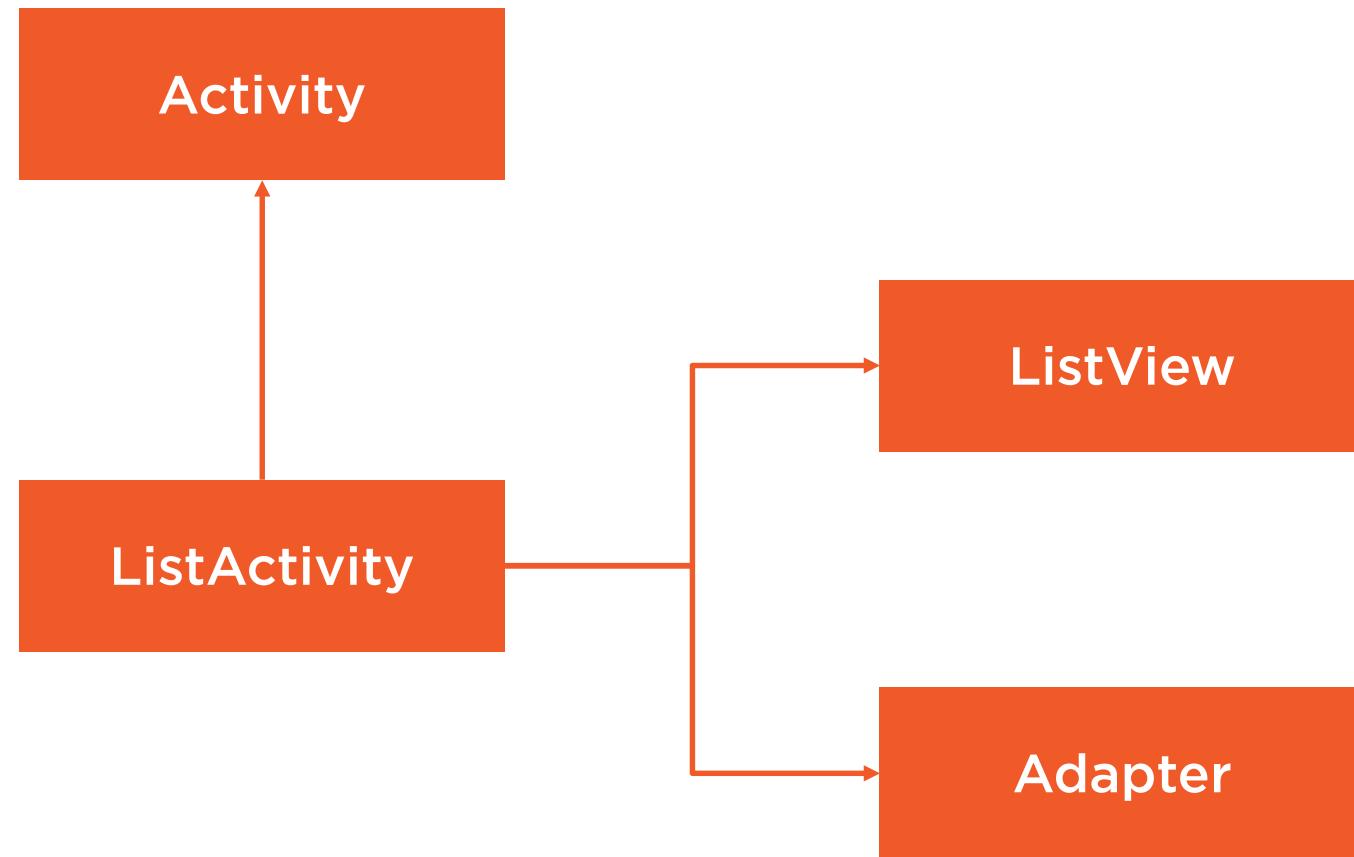
Fast scrolling



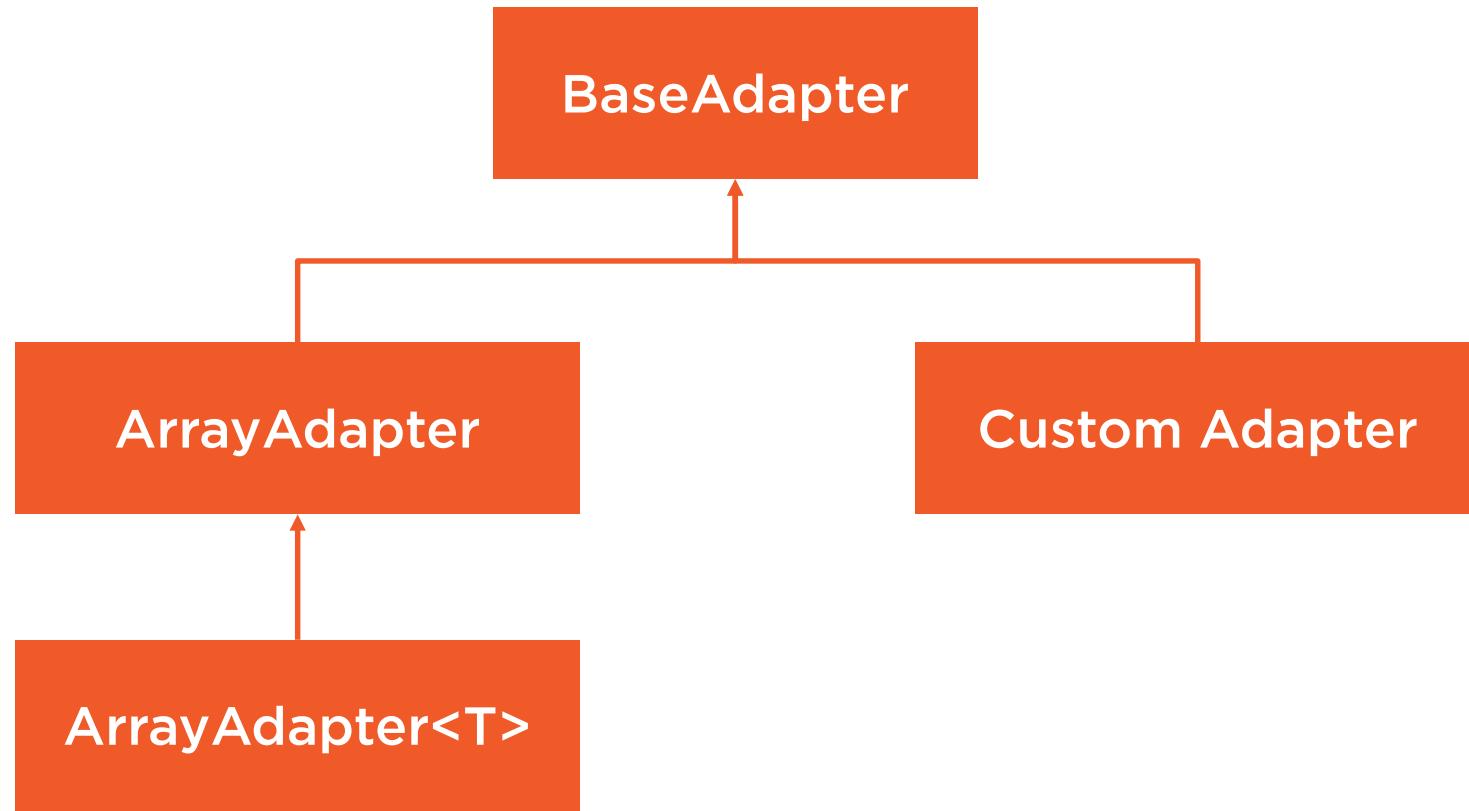
Participating Classes



Participating Classes



Participating Classes



Simple ListActivity

```
[Activity(Label = "BasicTable", MainLauncher = true, Icon = "@drawable/icon")]
public class HotDogListActivity : ListActivity
{
    string[] items;

    protected override void OnCreate(Bundle bundle)
    {
        base.OnCreate(bundle);

        items = new string[]
        { "Veggie Hot Dog", "Extra Long", "Regular Hot Dog" };

        ListAdapter = new  ArrayAdapter<String>(this,
            Android.Resource.Layout.SimpleListItem1, items);
    }

}
```



```
<LinearLayout  
    android:orientation="vertical"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:minWidth="25px"  
    android:minHeight="25px">  
    <ListView  
        android:minWidth="25px"  
        android:minHeight="25px"  
        android:layout_width="match_parent"  
        android:layout_height="match_parent"  
        android:id="@+id/hotDogListView" />  
</LinearLayout>
```

A View with a ListView



Implementing a Custom Adapter

```
public class HotDogListAdapter: BaseAdapter<HotDog>
{
    public HotDogListAdapter(Activity context, List<HotDog> items): base()
    {
    }

    public override long GetItemId(int position)
    {
    }

    public override HotDog this[int position]
    {
    }

    public override int Count
    {
    }

    public override View GetView(int position, View convertView, ViewGroup parent)
    {
    }
}
```



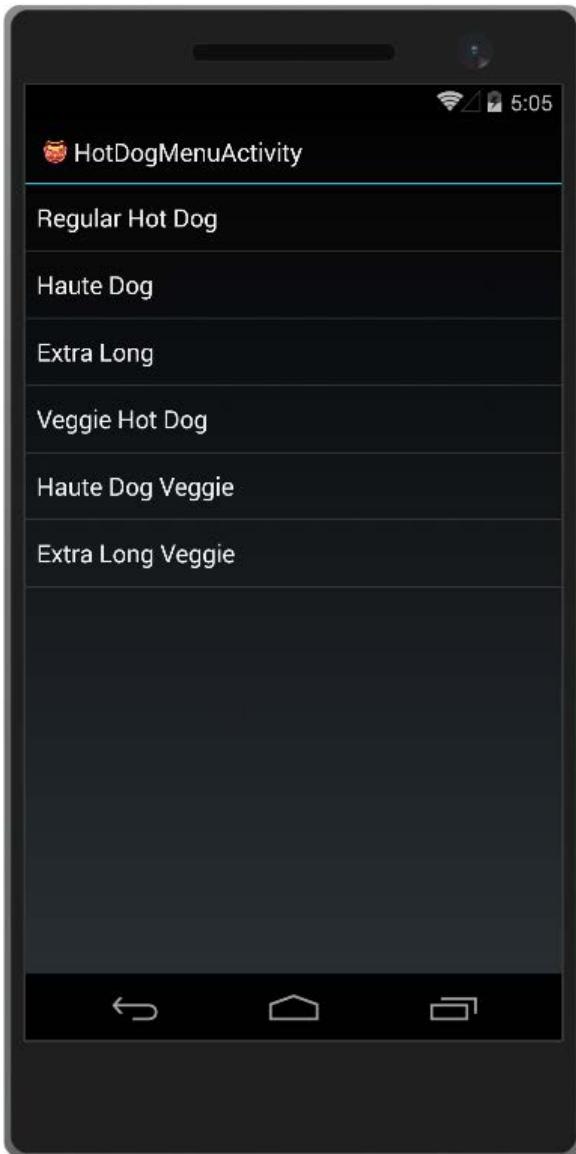
Linking the ListView and the Adapter

```
public class HotDogMenuActivity : Activity
{
    protected override void OnCreate(Bundle savedInstanceState)
    {
        base.OnCreate(savedInstanceState);
        SetContentView(Resource.Layout.HotDogMenuView);
        hotDogListView = FindViewById<ListView>(Resource.Id.hotDogListView);
        hotDogDataService = new HotDogDataService();
        allHotDogs = hotDogDataService.GetAllHotDogs();

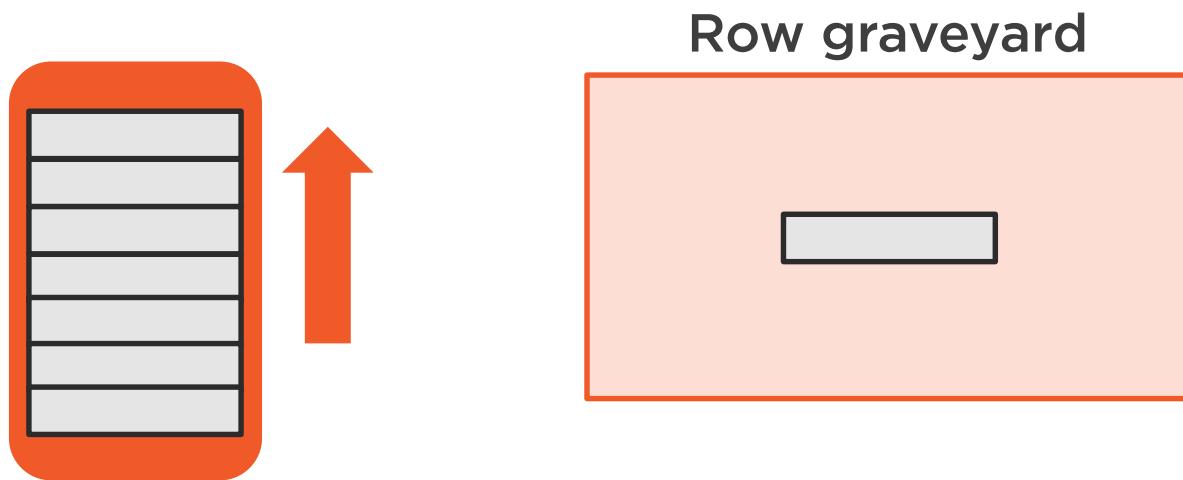
        hotDogListView.Adapter = new HotDogListAdapter(this, allHotDogs);
    }
}
```



The Result



Row Reuse



Row Reuse

```
public override View GetView(int position, View convertView, ViewGroup parent)
{
    View view = convertView; // re-use an existing view, if one is supplied
    if (view == null) // otherwise create a new one
        view =
            context.LayoutInflater.Inflate(Android.Resource.Layout.SimpleListItem1, null);

    view.FindViewById<TextView>(Android.Resource.Id.Text1).Text = items[position];

    return view;
}
```



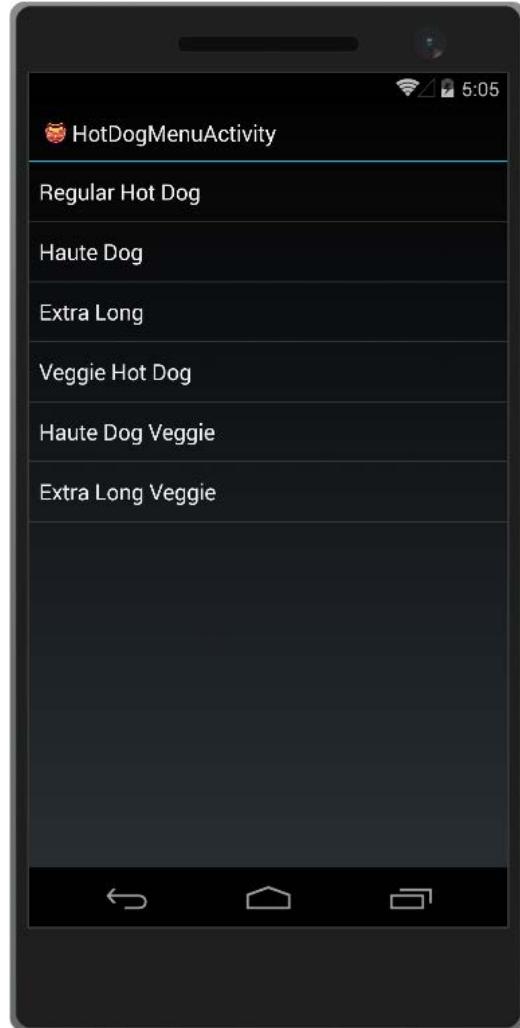
Demo Creating the Menu View



Changing the Row Style



Built-in Styles



SimpleList Item



TestList Item



ActivityList Item



Using a Built-in Row Style

```
if (convertView == null)
{
    convertView = contextLayoutInflater.Inflate
        (Android.Resource.Layout.SimpleListItem1, null);
}

convertView.FindViewById<TextView>
    (Android.Resource.Id.Text1).Text = item.Name;

return convertView;
```



Demo Using the Built-in Styles



Using a Custom Row Style

```
if (convertView == null)
{
    convertView = contextLayoutInflater.Inflate(Resource.Layout.HotDogRowView, null);
}

convertView.FindViewById<TextView>(Resource.Id.hotDogNameTextView).Text =
    item.Name;

convertView.FindViewById<TextView>(Resource.Id.shortDescriptionTextView).Text =
    item.ShortDescription;

convertView.FindViewById<TextView>(Resource.Id.priceTextView).Text =
    "$ " + item.Price;

convertView.FindViewById<ImageView>(Resource.Id.hotDogImageView)
    .SetImageBitmap(imageBitmap);
```



Demo Using a Custom Row Style



Summary



ListViews are the key to working with lists in Android

Require an Adapter to work

Custom row styles are key to customization

