CS 646 Android Mobile Application Development
Spring Semester, 2015
Doc 8 Layouts & Menus
Feb 24, 2015

Free Icons from Google

http://developer.android.com/design/downloads/index.html

http://tinyurl.com/7tp4nec

Fragment & Configuration Change

public void setRetainInstance (boolean retain)

Fragment Method

Fragment not destroyed on configuration change

View is destroyed

Does not work for fragments on the back stack

Layouts

Containers - LinearLayout

Important Properties/Concepts

- Orientation
- Fill Model
- Weight
- Gravity
- Padding

Orientation

android:orientation

horizontal
    view is a row

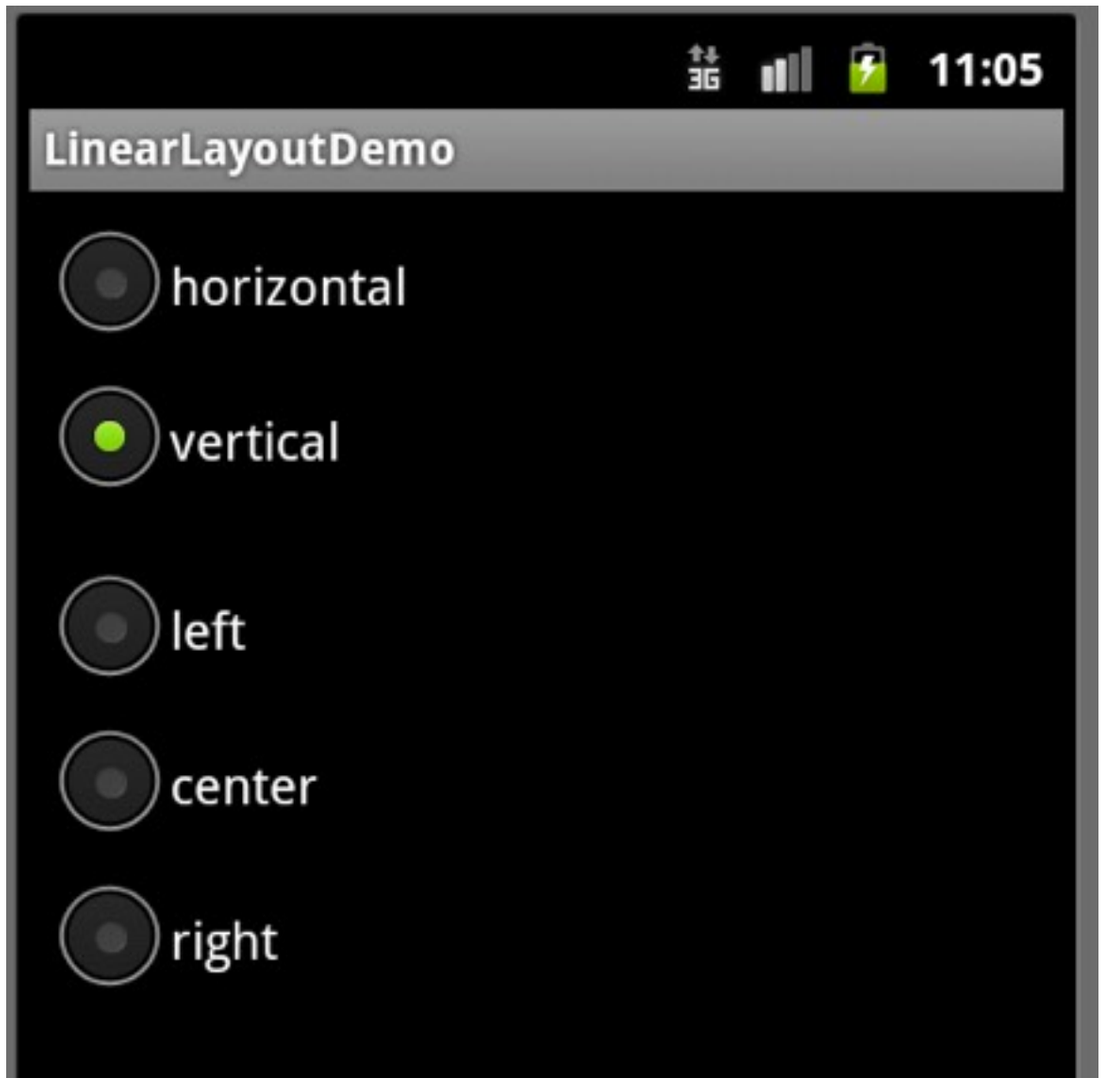
vertical
    view is a column

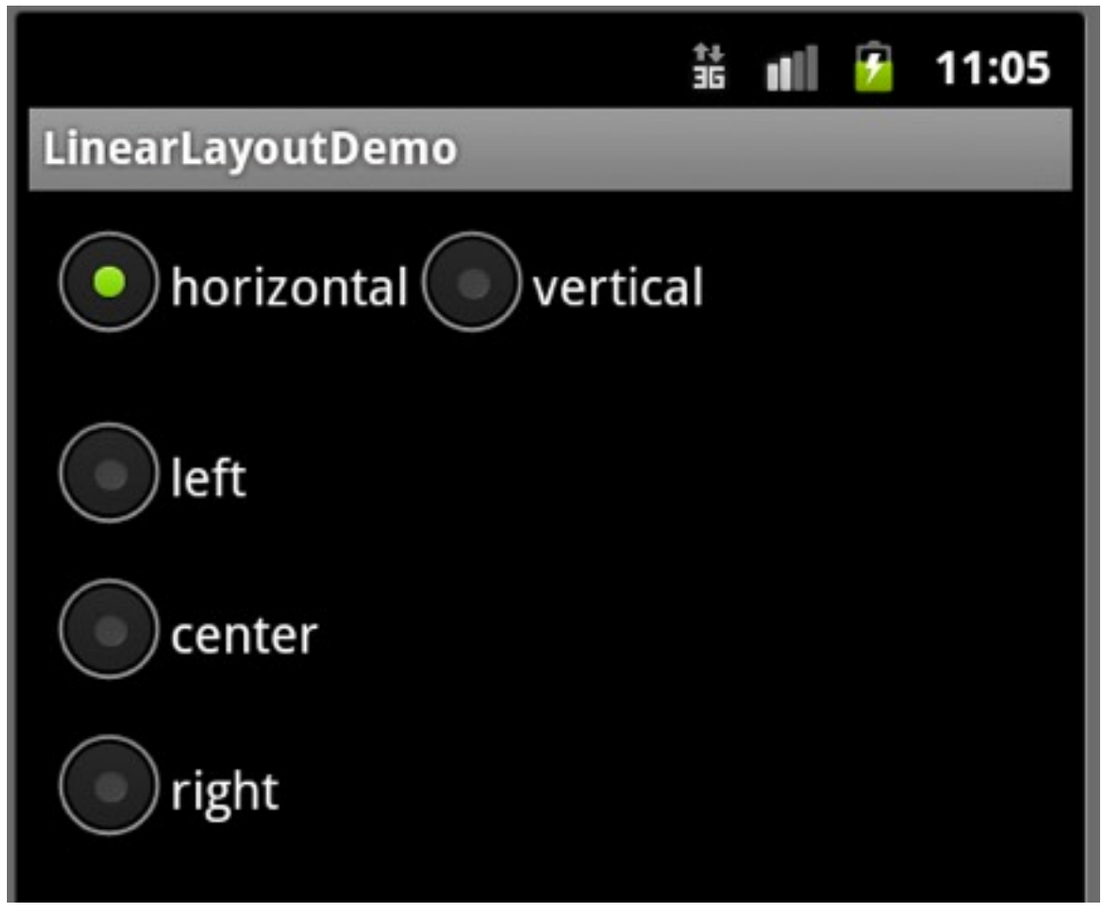
Change at runtime

setOrientation(LinearLayout.VERTICAL);

setOrientation(LinearLayout.HORIZONTAL);

Example


11:05
LinearLayoutDemo
horizontal
vertical
left
center
right


11:05
LinearLayoutDemo
horizontal
vertical
left
center
right

Gravity

android:layout_gravity
setGravity()

How do the subviews line up

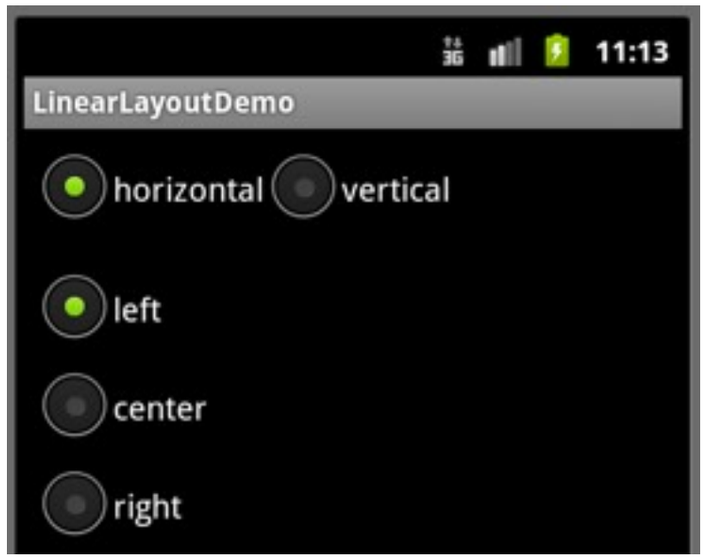
Values can be combined

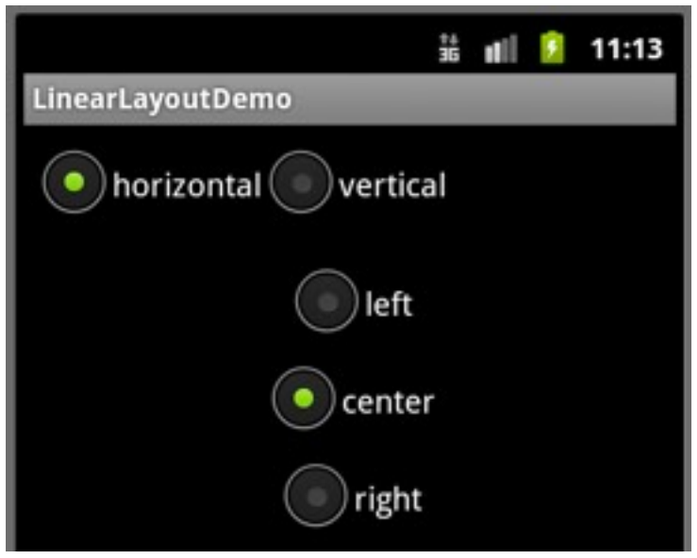
Values

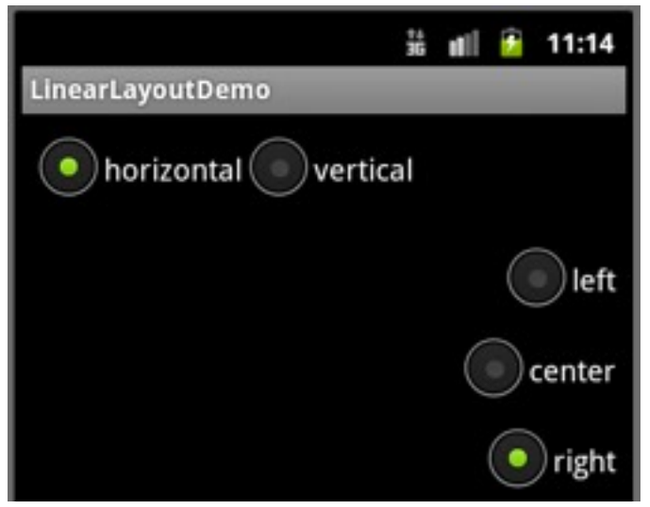
top
bottom
left
right
center_vertical
fill_vertical
center_horizontal
fill_horizontal
center
fill
clip_vertical
clip_horizontal
start
end

See http://developer.android.com/reference/android/widget/TextView.html#attr_android:gravity for descriptions of each.

Sample


11:13
LinearLayoutDemo
horizontal
vertical
left
center
right


11:13
LinearLayoutDemo
horizontal
vertical
left
center
right


11:14
LinearLayoutDemo
horizontal
vertical
left
center
right

Layout for example

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:orientation="vertical"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        >
        <RadioGroup android:id="@+id/orientation"
                android:orientation="horizontal"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:padding="5dip">
                <RadioButton
                        android:id="@+id/horizontal"
                        android:text="horizontal" />
                <RadioButton
                        android:id="@+id/vertical"
                        android:text="vertical" />
        </RadioGroup>
```

```
        <RadioGroup android:id="@+id/gravity"
                android:orientation="vertical"
                android:layout_width="fill_parent"
                android:layout_height="wrap_content"
                android:padding="5dip">
                <RadioButton
                        android:id="@+id/left"
                        android:text="left" />
                <RadioButton
                        android:id="@+id/center"
                        android:text="center" />
                <RadioButton
                        android:id="@+id/right"
                        android:text="right" />
        </RadioGroup>
</LinearLayout>
```

Activity source

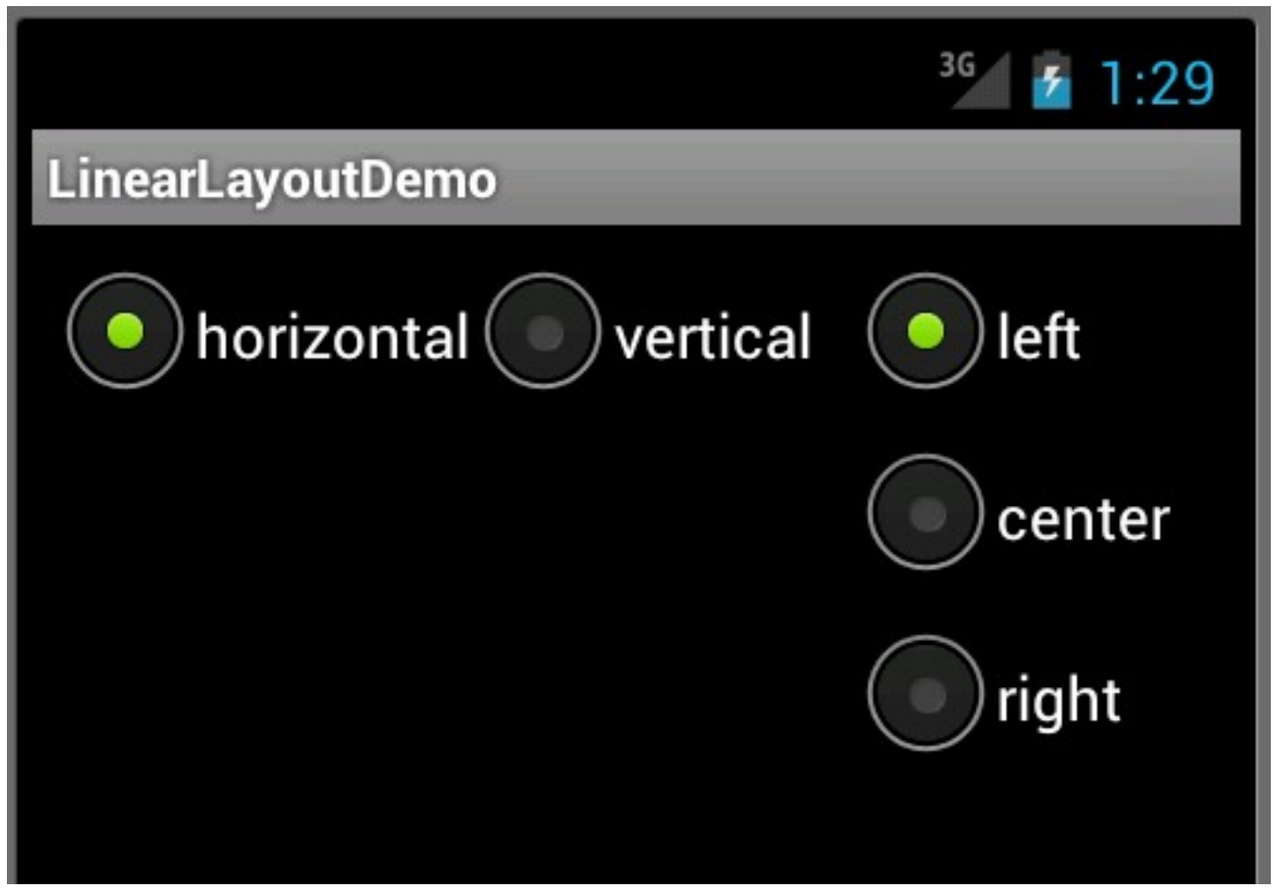
```
public class LinearLayoutDemo extends Activity
        implements RadioGroup.OnCheckedChangeListener {
        RadioGroup orientation;
        RadioGroup gravity;

        @Override
        public void onCreate(Bundle icicle) {
                super.onCreate(icicle);
                setContentView(R.layout.main);

                orientation=(RadioGroup)findViewById(R.id.orientation);
                orientation.setOnCheckedChangeListener(this);
                gravity=(RadioGroup)findViewById(R.id.gravity);
                gravity.setOnCheckedChangeListener(this);
}
```


Source: Beginning Android 4, Grant Allen, Chapter 10

Activity source

```
public void onCheckedChanged(RadioGroup group, int checkedId) {
    switch (checkedId) {
        case R.id.horizontal:
            orientation.setOrientation(LinearLayout.HORIZONTAL);
            break;
        case R.id.vertical:
            orientation.setOrientation(LinearLayout.VERTICAL);
            break;
        case R.id.left:
            gravity.setGravity(Gravity.LEFT);
            break;
        case R.id.center:
            gravity.setGravity(Gravity.CENTER_HORIZONTAL);
            break;
        case R.id.right:
            gravity.setGravity(Gravity.RIGHT);
            break;
    }
  }
}
```

Setting layout orientation




```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
      xmlns:android="http://schemas.android.com/apk/res/android"
      android:orientation="horizontal"
    etc
```

Fill Model

subviews supply
android:layout_width
android:layout_height

Specify

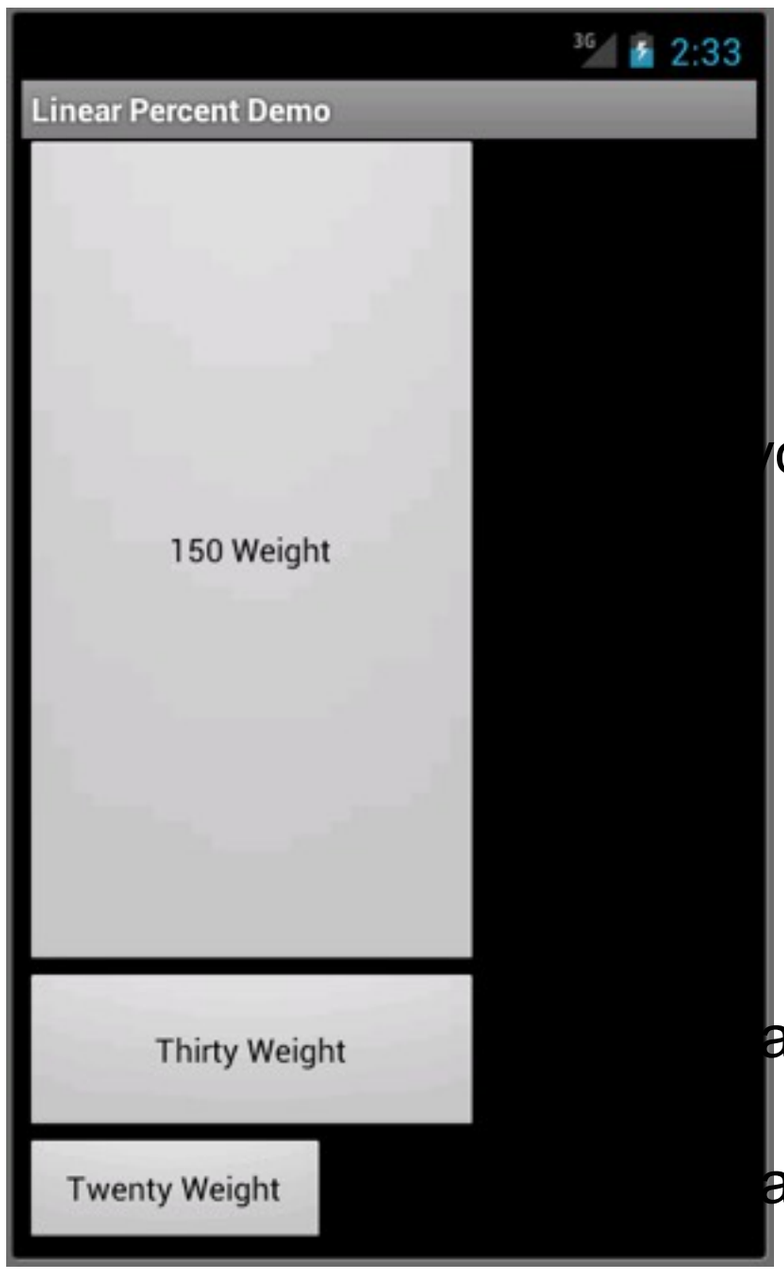
Exact number

wrap_content
Big enough to enclose content + padding

fill_parent
Big as parent minus padding
SDK 7 and earlier

match_parent
Big as parent minus padding
SDK 8 and later
Replaces fill_parent

Specifying Size of Widget



vout_width="200sp"

ayout_width="200dip"

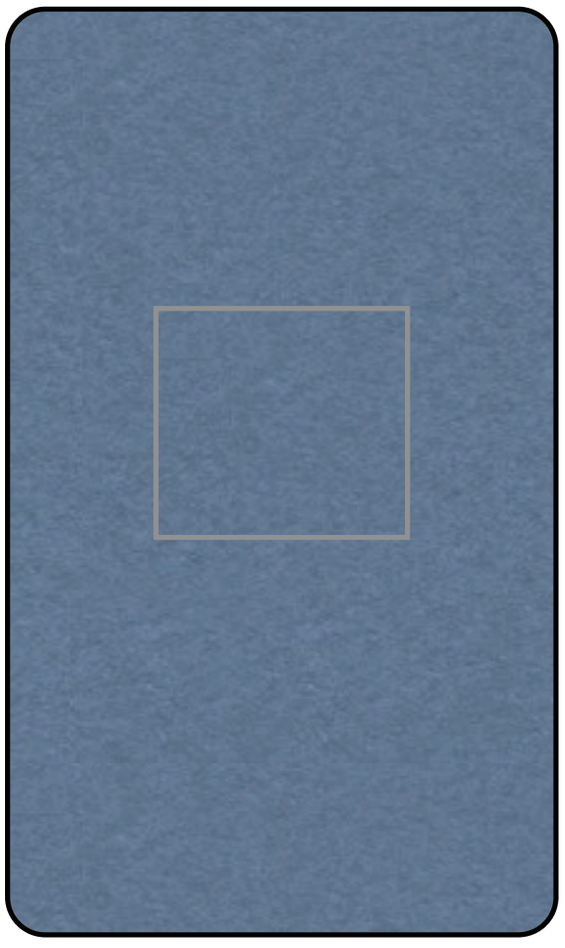
ayout_width="200px"

Units	
px	(pixels)
dp, dip	(density independent pixels)
sp	(scaled pixels)
in	(inches)
mm	(millimeters)

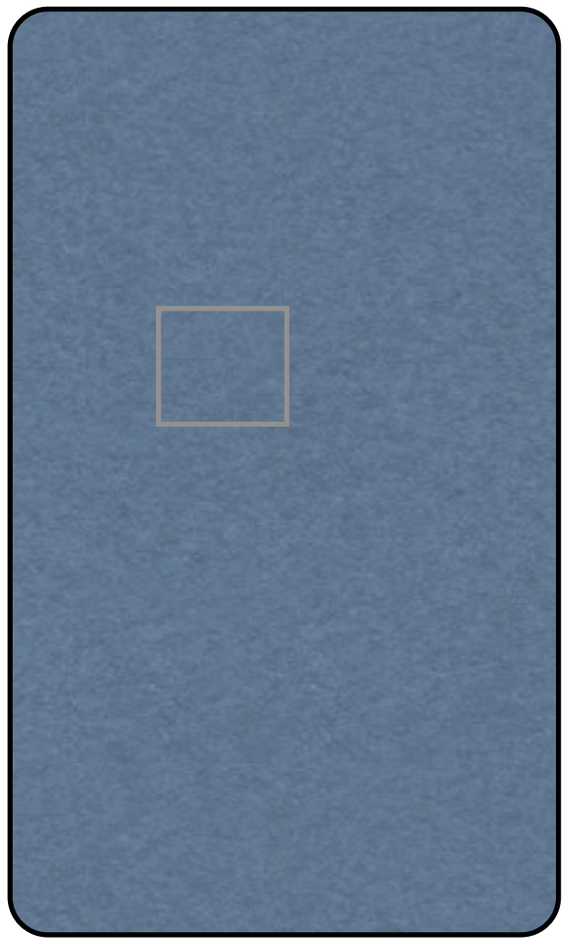
Problem with using px

n * n px box on devices with same screen size

Device with 160 dpi



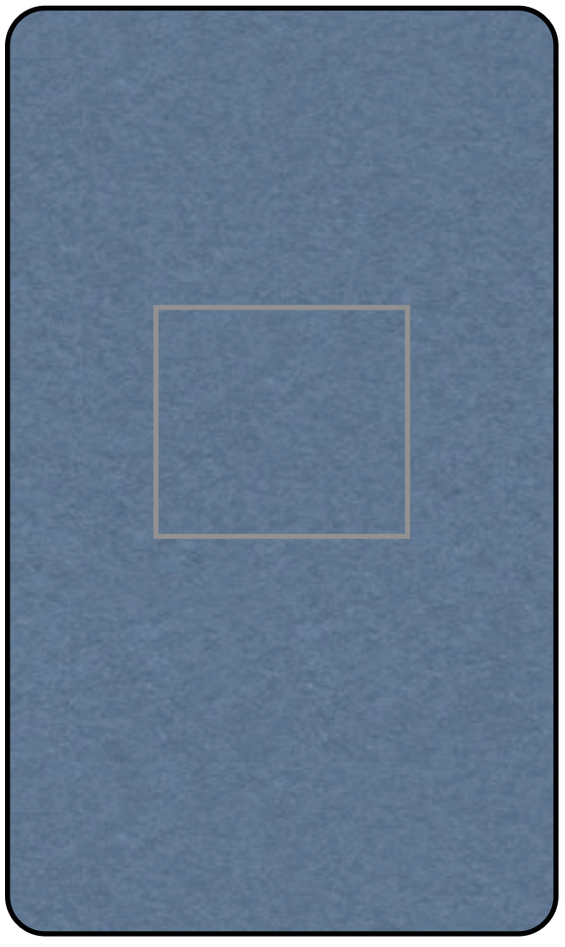
Device with 320 dpi



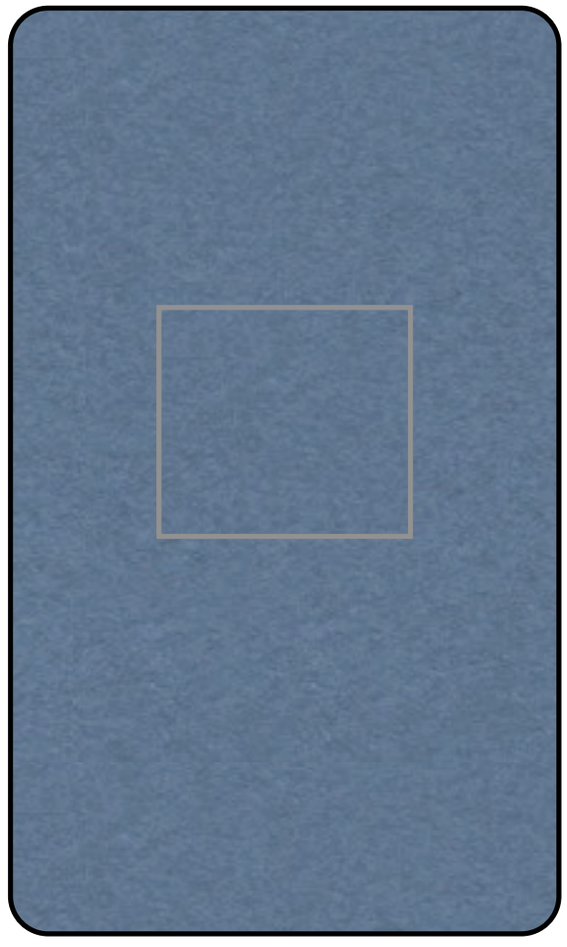
Using dp

n * n dp box on devices with same screen size

Device with 160 dpi



Device with 320 dpi



dp verses sp

dp - density independent pixels

- Size of pixel at 160 dpi
- Actual size on device is scaled using devices dpi

sp - scaled pixels

- Size of pixel at 160 dpi
- Actual size on device is scaled using devices dpi + font preference of user
- For text only

Screen Density and Bitmaps

Android will scale bitmaps (.png, .jpg, and .gif)

But bitmaps don't scale well

Provide multiple sizes for different devices

Android Density sizes

Name	Density	Scale Factor
ldpi	~120 dpi	0.75
mdpi	~160 dpi	1
hdpi	~240 dpi	1.5
xhdpi	~320 dpi	2.0
xxhdpi	~480 dpi	3.0
xxxhdpi	~640 dpi	4.0
tvdpi	~213 pdi	1.33

Provide multiple sizes for Bitmaps

```
MyProject/
  res/
    drawable-xhdpi/
        awesomeimage.png
    drawable-hdpi/
        awesomeimage.png
    drawable-mdpi/
        awesomeimage.png
drawable-ldpi/
    awesomeimage.png
```

What about other densities?

Android will select closest size and scale it

Different Screen Sizes

We will cover this later

Weight

android:layout_weight

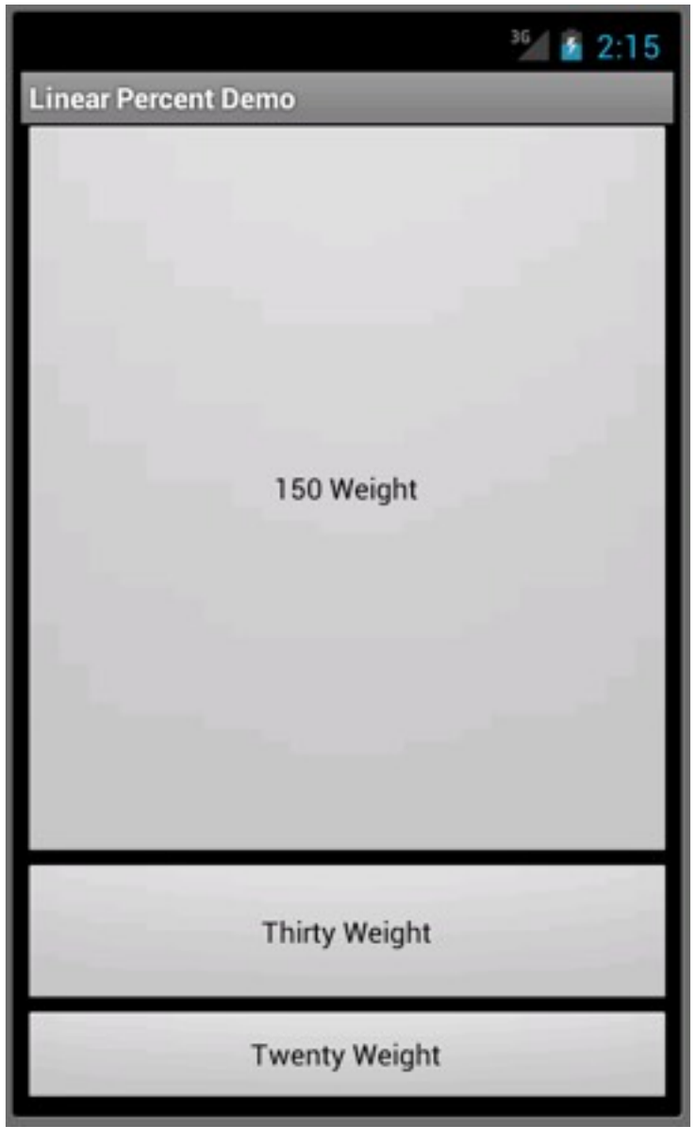
Relative weight of views to use in fill_parent

A view of twice the weight take twice the space

Example

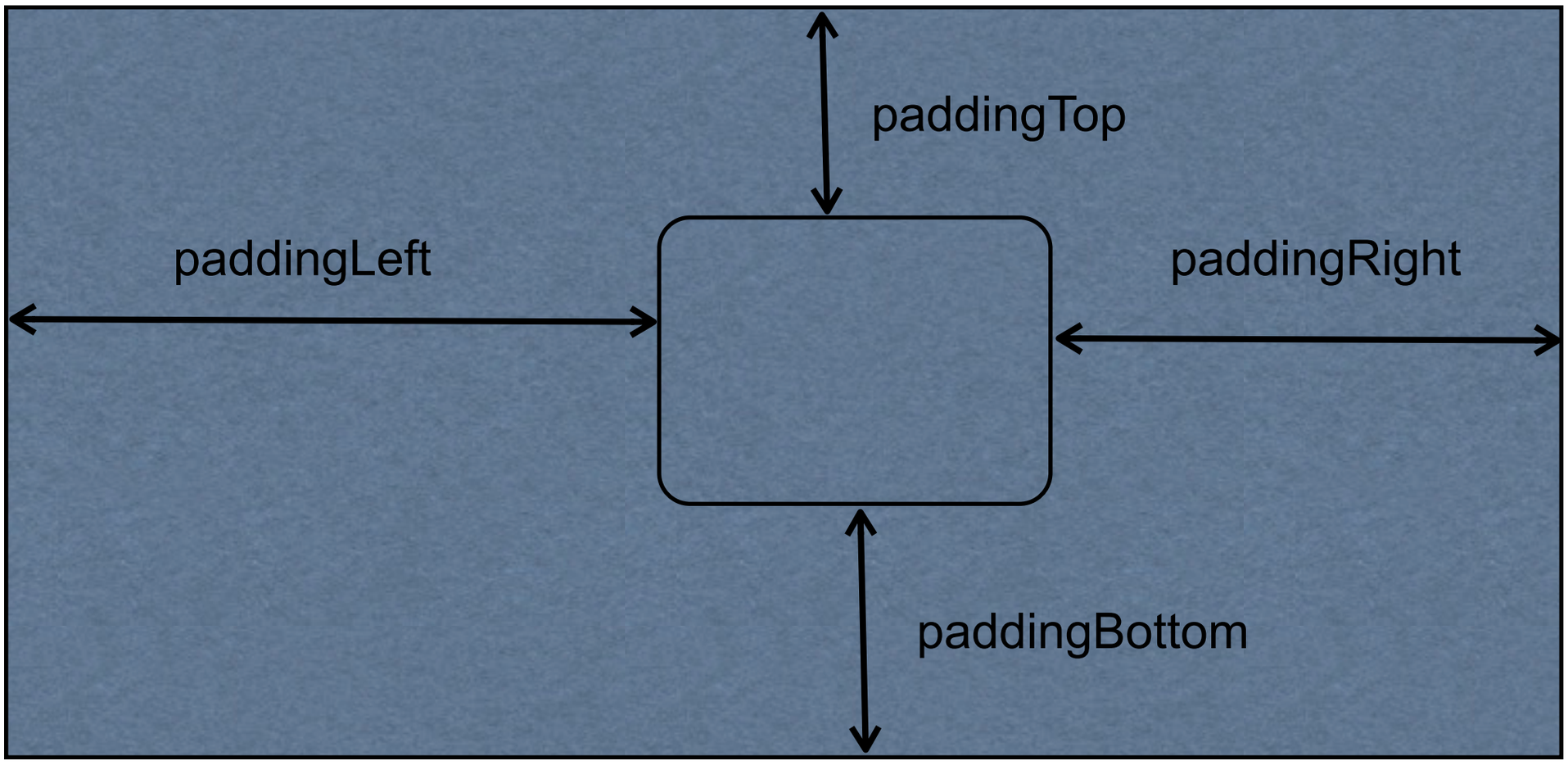

2:14
Linear Percent Demo
Fifty Weight
Thirty Weight
Twenty Weight


2:09
Linear Percent Demo
Thirty Weight
Thirty Weight
Twenty Weight


2:15
Linear Percent Demo
150 Weight
Thirty Weight
Twenty Weight

Padding

android:padding
android:paddingBottom
android:paddingLeft
etc
setPadding(int left, int top, int right, int bottom)
setPaddingLeft(int)
etc



Tuesday, February 24, 15

in the xml padding can be in px, dp, sp, in, mm
in methods it is in pixels

Relative Layout

Relative to parent

android:layout_alignParentTop:
android:layout_alignParentBottom:
android:layout_alignParentLeft:
android:layout_alignParentRight:
android:layout_centerHorizontal:
android:layout_centerVertical:

Relative Layout

Relative to other widgets

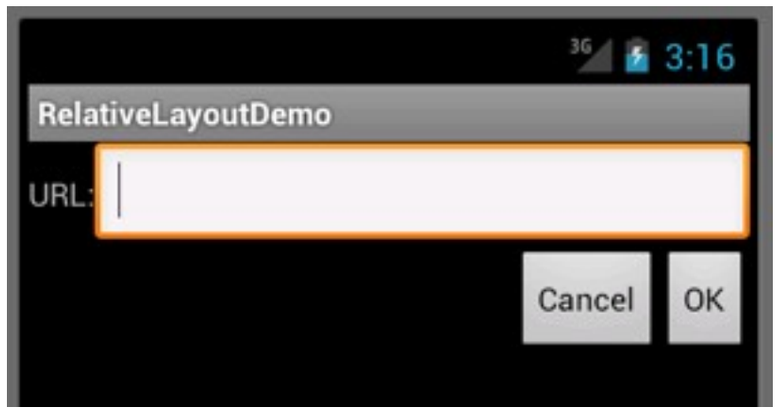
have to give widget an id

must reference the id

android:layout_above:
android:layout_below:
android:layout_toLeftOf:
android:layout_toRightOf:

android:layout_alignTop:
android:layout_alignBottom:
android:layout_alignLeft:
android:layout_alignRight:
android:layout_alignBaseline:

Example

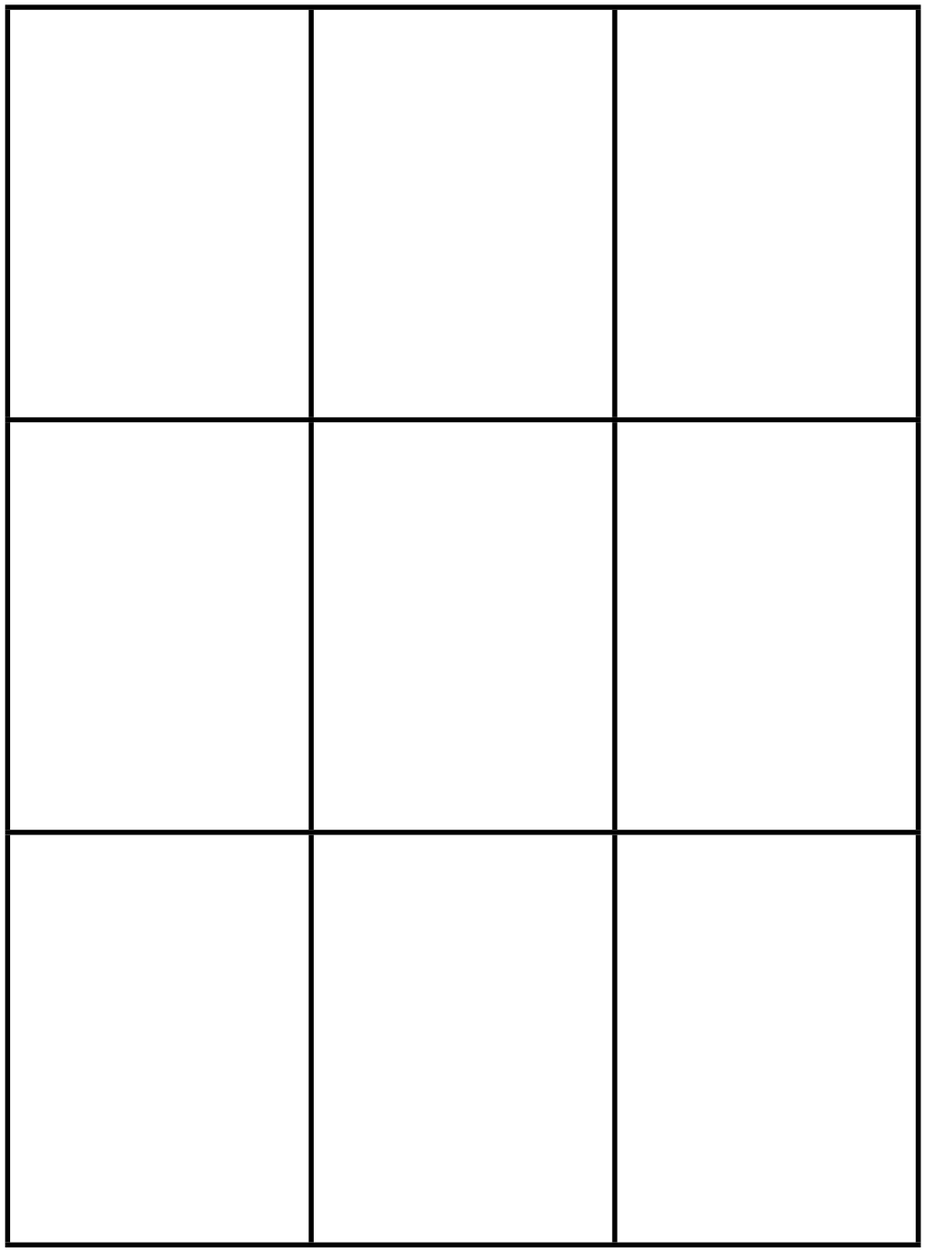



```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content">
    <TextView android:id="@+id/label"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="URL:"
        android:layout_alignBaseline="@+id/entry"
        android:layout_alignParentLeft="true"/>
    <EditText
        android:id="@id/entry"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_toRightOf="@id/label"
        android:layout_alignParentTop="true"/>
    <Button
        android:id="@+id/ok"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/entry"
        android:layout_alignRight="@id/entry"
        android:text="OK" />
```


Source: Beginning Android 4, Grant Allen, Chapter 10

Table View

Screen is divided into rows and columns



Creating Rows and Columns

```
<TableRow>
    <Button android:id="@+id/A"
        android:text="A" />

    <Button android:id="@+id/B"
        android:text="B" />

<Button android:id="@+id/C"
        android:text="C" />

</TableRow>
```

Each item in a row occupies a column

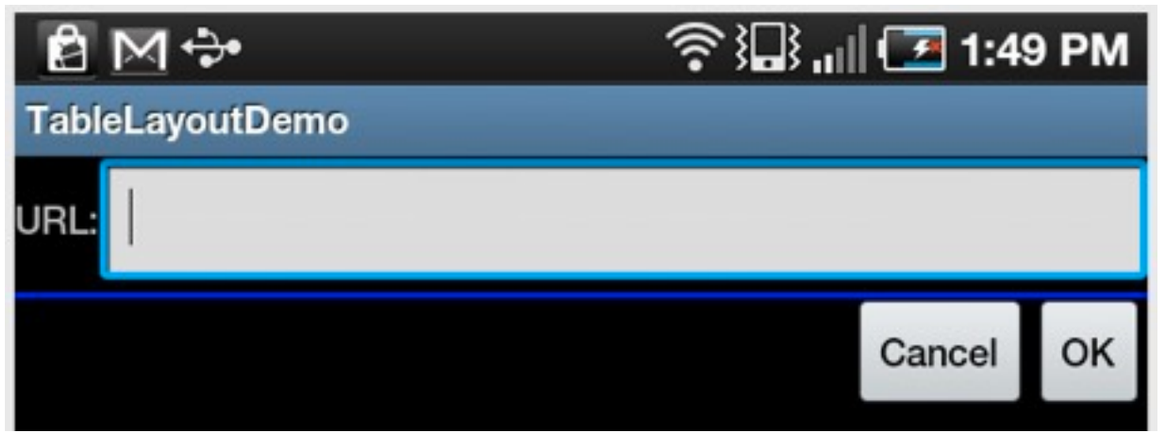
layout_span

```
<TableRow>
    <TextView
            android:text="URL:" />

    <EditText android:id="@+id/entry"          3 columns
        android:layout_span="3"/>

</TableRow>
```

Specifying the column

```
<TableRow>
    <Button android:id="@+id/cancel"
        android:layout_column="2"
        android:text="Cancel" />
    <Button android:id="@+id/ok"
        android:text="OK" />
</TableRow>
```

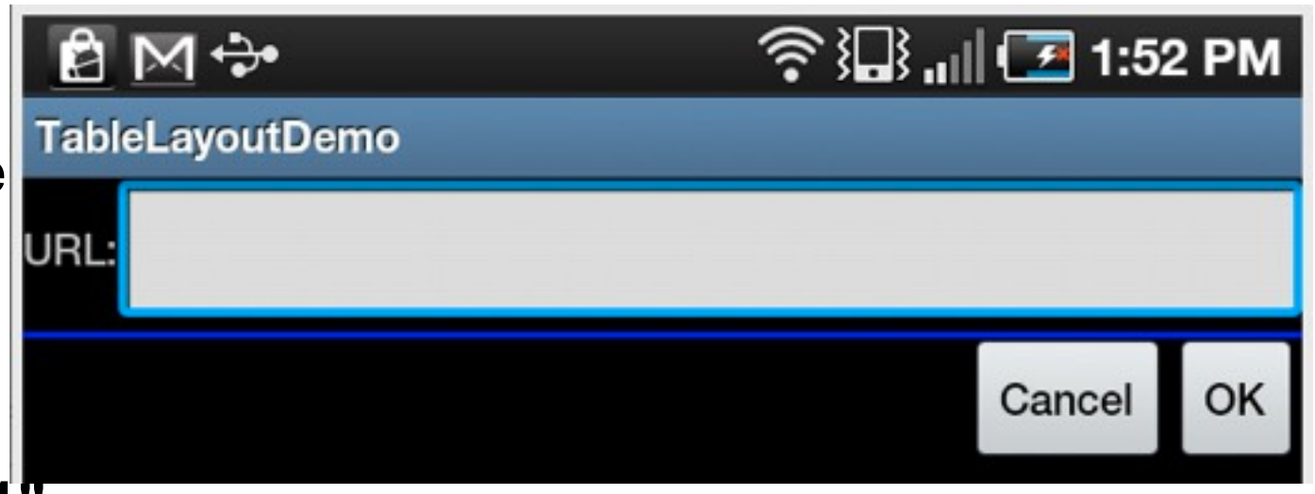
Example




```
<TableLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:stretchColumns="1">
    <TableRow>
        <TextView
                android:text="URL:" />
        <EditText android:id="@+id/entry"
            android:layout_span="3"/>
    </TableRow>
    <View
        android:layout_height="2dip"
        android:background="#0000FF" />
    <TableRow>
        <Button android:id="@+id/cancel"
            android:layout_column="2"
            android:text="Cancel" />
        <Button android:id="@+id/ok"
            android:text="OK" />
    </TableRow>
</TableLayout>
```

Later items appear in later columns

```
<TableRow>
    <Button android:id="@+id/cance
        android:layout_column="2"
        android:text="Cancel" />
    <Button android:id="@+id/ok"
        android:layout_column="1"
        android:text="OK" />
</TableRow>
```

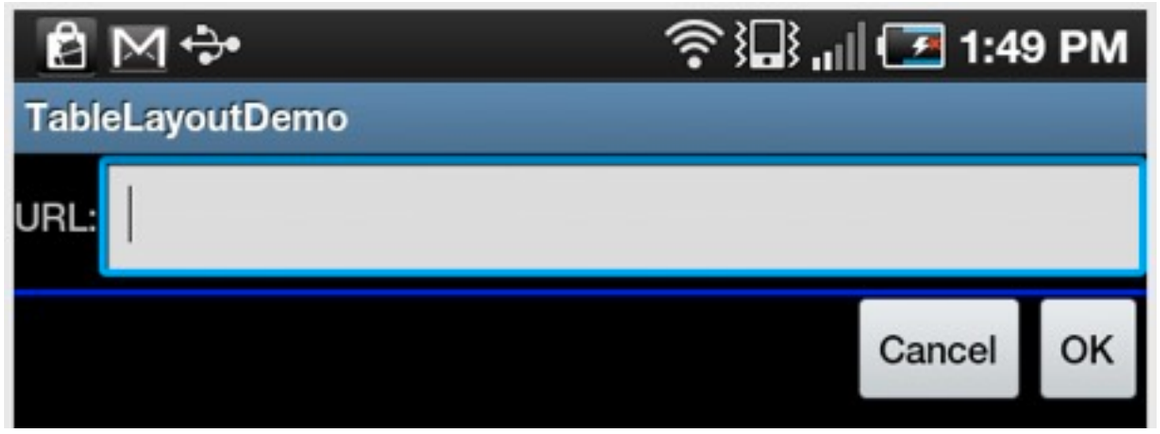


Can Skip Columns


```
<TableRow>
      <Button android:id="@+id/cancel"
            android:layout_column="0"
            android:text="Cancel" />
      <Button android:id="@+id/ok"
            android:layout_column="3"
            android:text="OK" />
</TableRow>
```

Stretch, Shrink, and Collapse

android:stretchColumns
android:shrinkColumns
android:collapseColumns

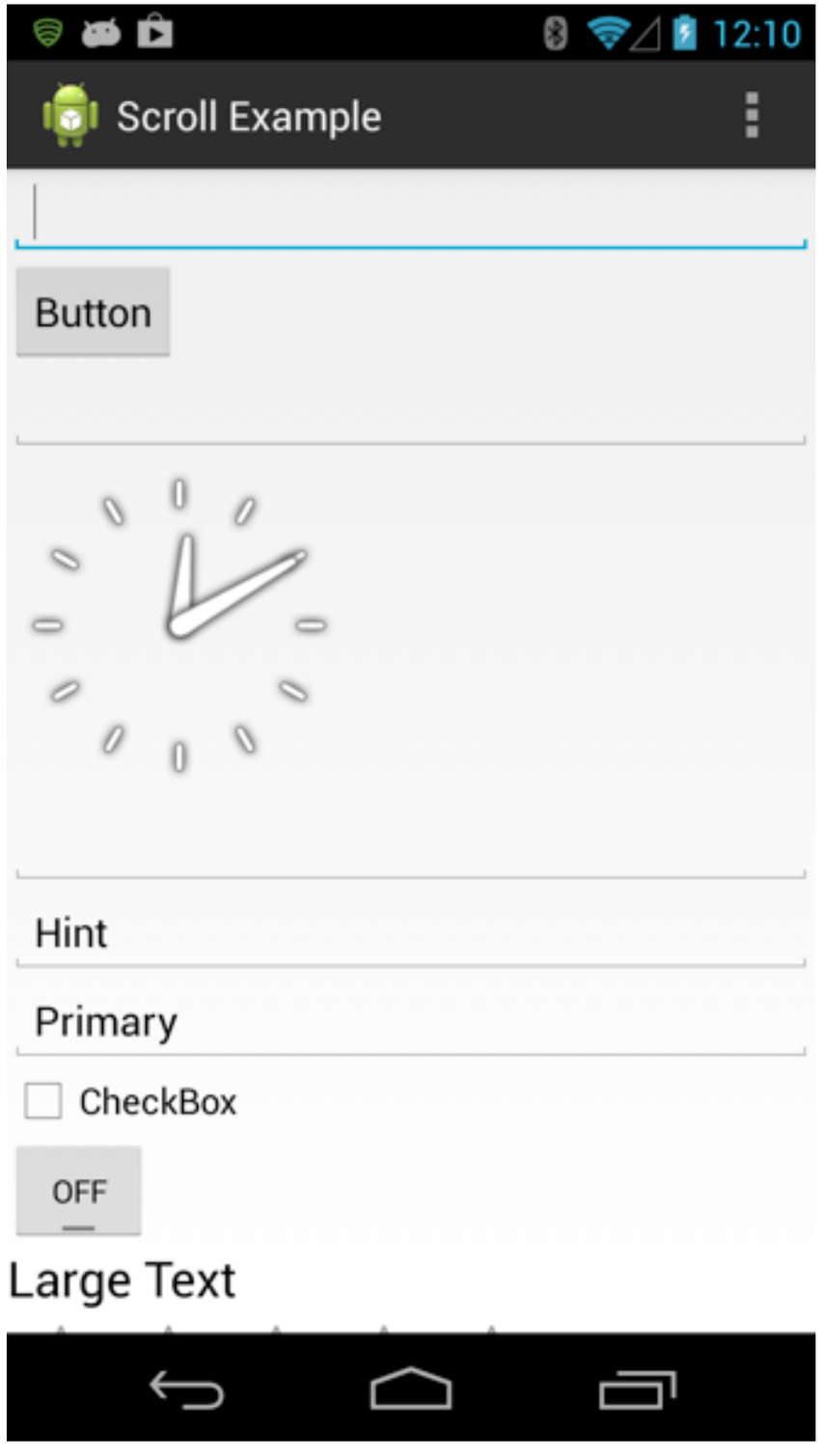



```
<TableLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:stretchColumns="1">
```

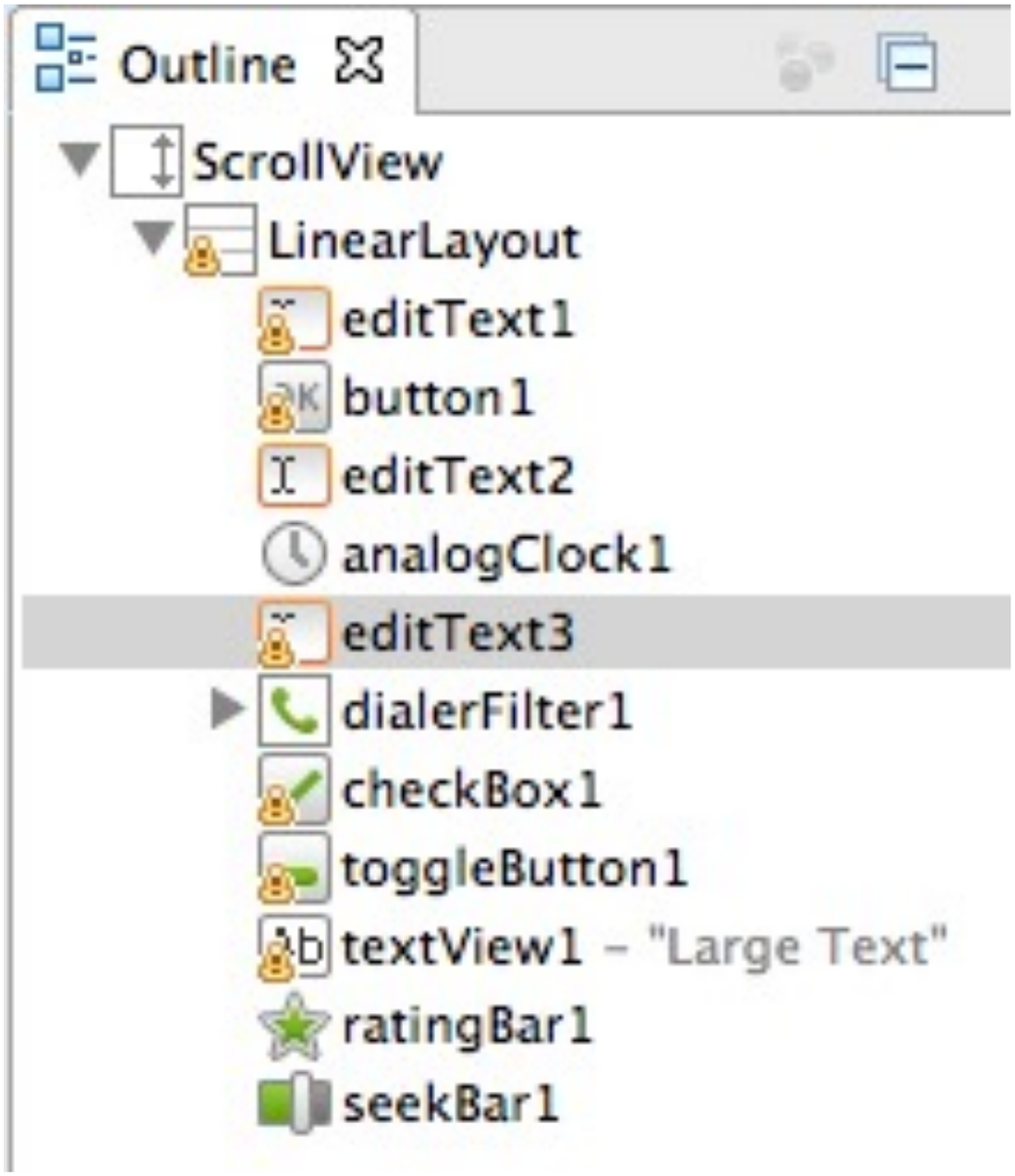
ScrollView, HorizontalScrollView

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content">
    <TableLayout
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:stretchColumns="0">
        <TableRow>
            <View
                android:layout_height="80dip"
                android:background="#000000"/>
            <TextView android:text="#000000"
                android:paddingLeft="4dip"
                android:layout_gravity="center_vertical" />
        </TableRow>
    </TableLayout>
</ScrollView>
```

ScrollView, HorizontalScrollView



ScrollView


Outline
ScrollView
LinearLayout
editText1
button1
editText2
analogClock1
editText3
dialerFilter1
checkBox1
toggleButton1
textView1 - "Large Text"
ratingBar1
seekBar1

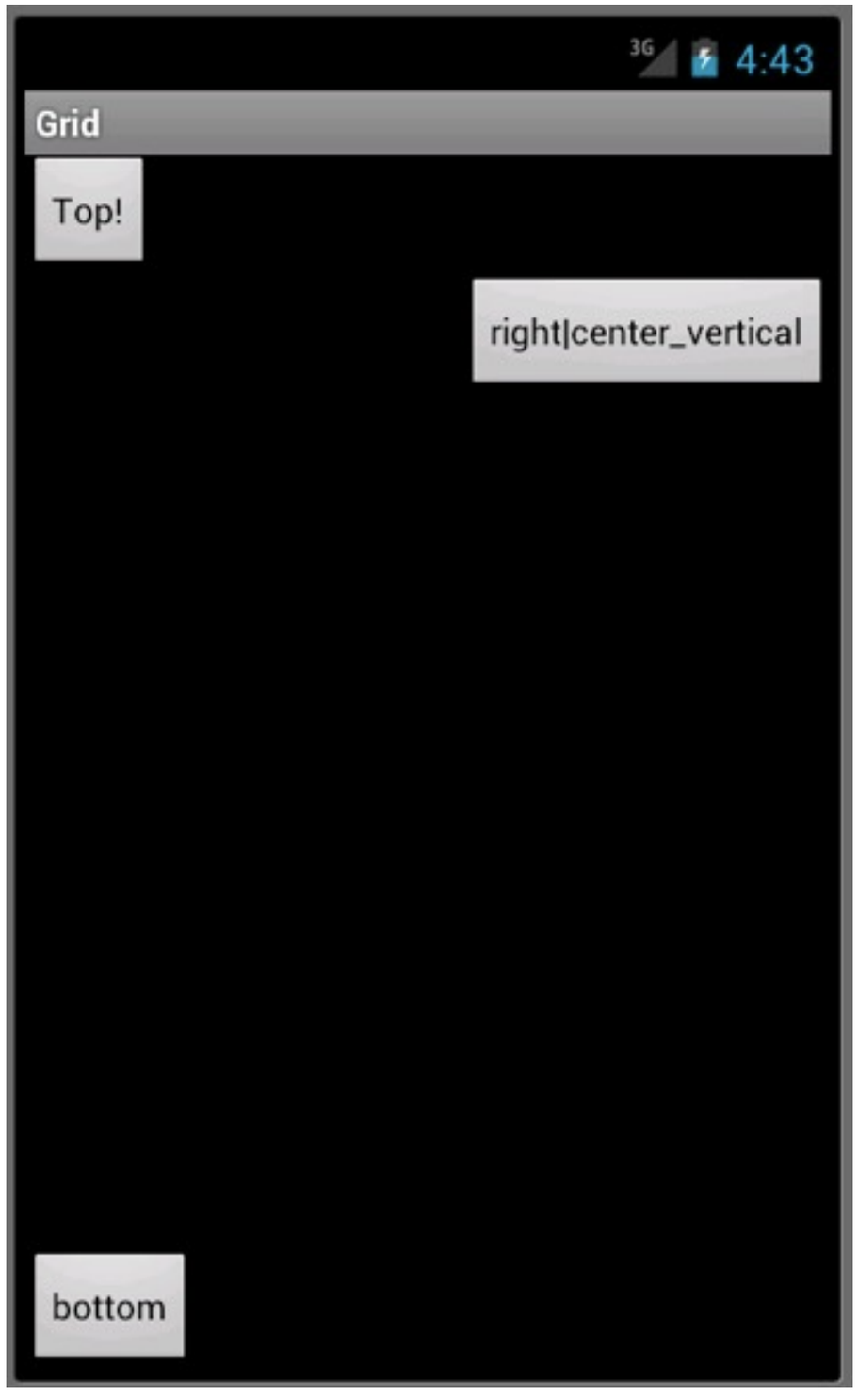
GridLayout

New in Android 4.0

Allows any number of rows and columns

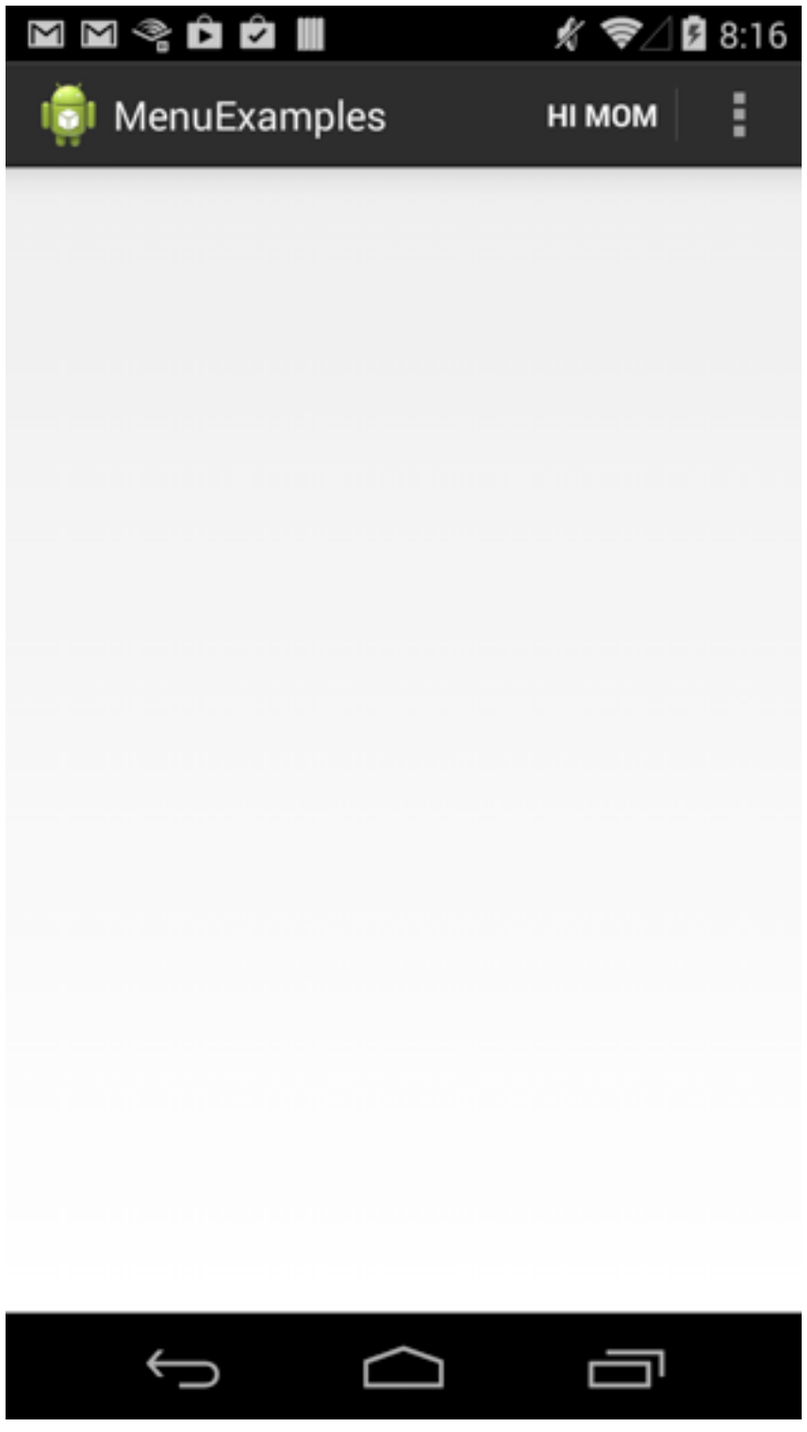
Using Default Rows, Columns

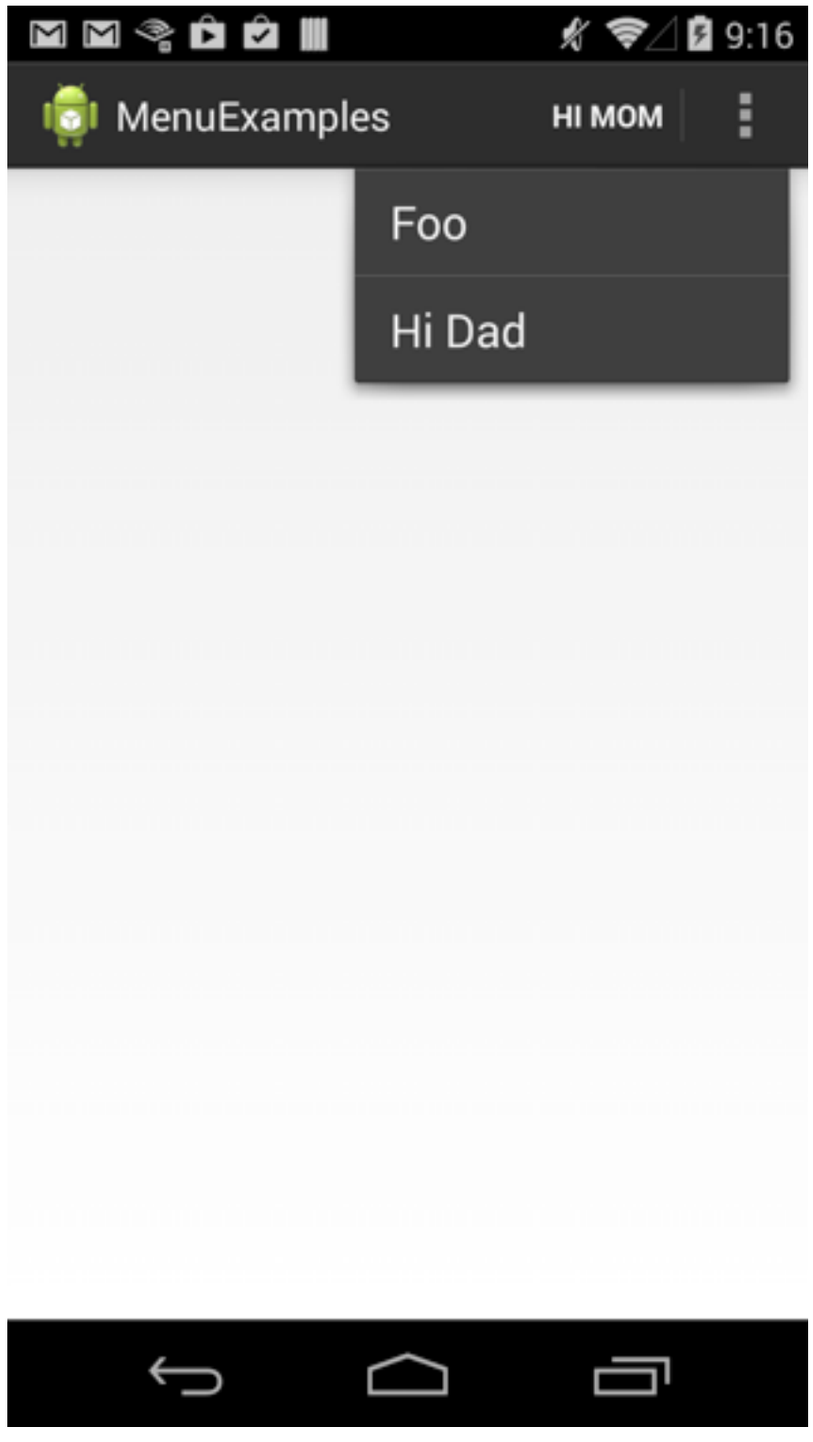
```
<?xml version="1.0" encoding="utf-8"?>
<GridLayout
 xmlns:android="http://schemas.android.com/apk/res/android"
 android:orientation="vertical"
 android:layout_width="fill_parent"
 android:layout_height="fill_parent"
>
  <Button
    android:text="Top!"
    android:layout_gravity="top"
  />
  <Button
    android:text="right|center_vertical"
    android:layout_gravity="right|center_vertical"
  />
  <Button
    android:text="bottom"
    android:layout_gravity="bottom"
  />
</GridLayout>
```



Menus

Menus





Type Of Menus

Normal
Action Bar Items
Popup menus
Contextual menus

Menus

3 methods

public boolean onCreateOptionsMenu(Menu menu)

Called once

public boolean onPrepareOptionsMenu(Menu menu)

Called before menu is shown, every time menu is shown

public boolean onOptionsItemSelected(MenuItem item)

Called to handle selection

Adding The Menu - In code

```
public class MainActivity extends Activity {
    private static final int DAD_ID = Menu.FIRST;
    private static final int MOM_ID = Menu.FIRST + 1;

        @Override
     public boolean onCreateOptionsMenu(Menu menu) {
     super.onCreateOptionsMenu(menu);

     menu.add(Menu.NONE, DAD_ID, 2, R.string.menu_dad).setShortcut('0', 'd');

     menu.add("Foo");

     MenuItem mom = menu.add(Menu.NONE, MOM_ID, Menu.NONE,
                                 R.string.menu_mom);
     mom.setShowAsAction(MenuItem.SHOW_AS_ACTION_ALWAYS);
     return true;
     }
}
```

Add methods

add(int groupId, int itemId, int order, CharSequence title)

add(CharSequence title)

add(int titleRes)

ShortCuts

setAlphabeticShortcut(char alphaChar)

setNumericShortcut(char numericChar)

setShortcut(char numericChar, char alphaChar)

setShowAsAction

SHOW_AS_ACTION_ALWAYS
SHOW_AS_ACTION_COLLAPSE_ACTION_VIEW
SHOW_AS_ACTION_IF_ROOM
SHOW_AS_ACTION_NEVER
SHOW_AS_ACTION_WITH_TEXT

boolean onCreateOptionsMenu

Called once

First time menu is to be displayed

Return true to have menu displayed

Handling the Menu

```
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
    case DAD_ID:
            Log.i("rew","Dad selected");
        return true;
    case MOM_ID:
            Log.i("rew","Mom selected");
        return true;
    }
    if ( item.getTitle() == "Foo") {
          Log.i("rew", "Menu Foo selected");
       return true;
       }
    return super.onOptionsItemSelected(item);
}
```

Menu from XML Resource

Better to use Menu resource

Multiple languages

Different screen size & density

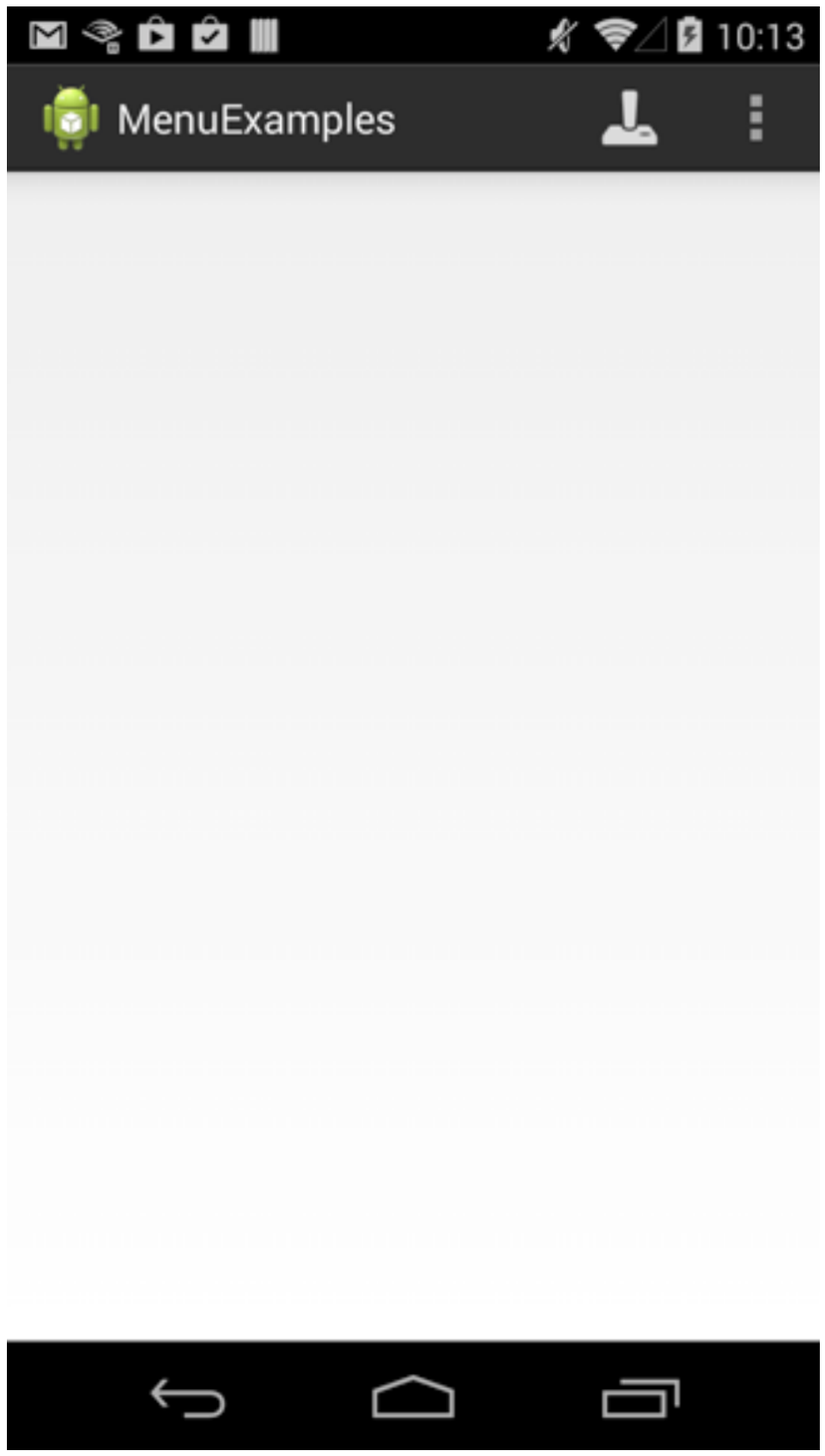
res/menu/game_menu.xml

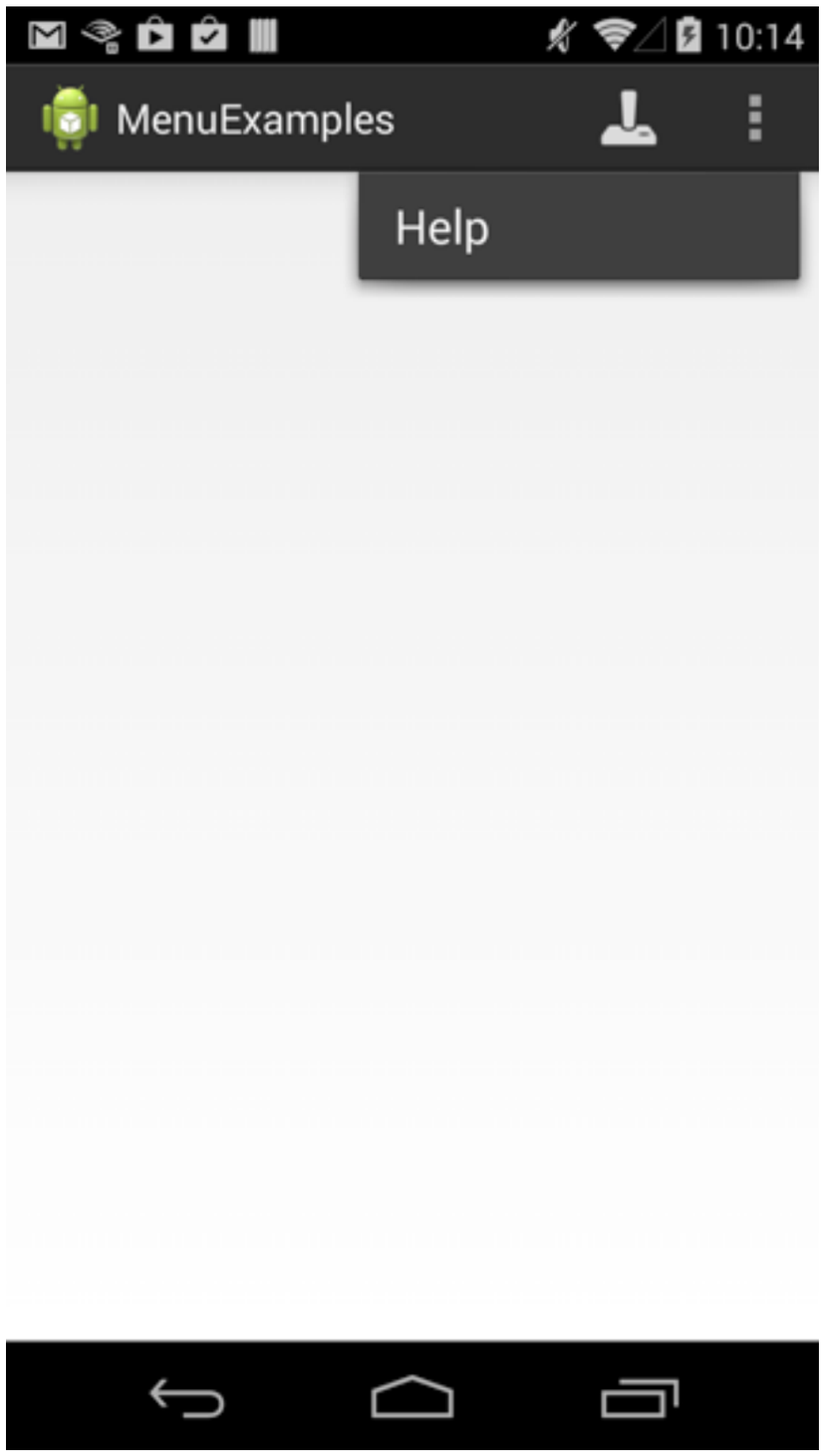
```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:id="@+id/new_game"
          android:icon="@drawable/ic_action_gamepad"
          android:title="@string/new_game"
          android:showAsAction="ifRoom"/>
    <item android:id="@+id/help"
          android:icon="@drawable/ic_action_help"
          android:title="@string/help" />
</menu>
```

Activity Menu methods

```
public class MainActivity extends Activity {

    public boolean onCreateOptionsMenu(Menu menu) {
        MenuInflater inflater = getMenuInflater();
        inflater.inflate(R.menu.game_menu, menu);
        return true;
    }

    public boolean onOptionsItemSelected(MenuItem item) {
        switch (item.getItemId()) {
        case R.id.new_game:
            Log.i("rew","new game");
            return true;
        case R.id.help:
            Log.i("rew","help selected");
            return true;
        }
        return super.onOptionsItemSelected(item);
    }
```

The Menu





OnClick for Menus

One can specify a method to call when a menu is selected

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:id="@+id/new_game"
        android:icon="@drawable/ic_action_gamepad"
        android:title="@string/new_game"
        android:onClick="newGame"
        android:showAsAction="ifRoom"/>
    <item android:id="@+id/help"
        android:icon="@drawable/ic_action_help"
        android:title="@string/help" />
</menu>
```

The Method

```
public void newGame(MenuItem selected) {
        Log.i("rew", "New game method");
    }

 public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
    case R.id.new_game:
          Log.i("rew","new game");
        return true;
    case R.id.help:
          Log.i("rew","help selected");
        return true;
    }
    return super.onOptionsItemSelected(item);
}
```

onOptionsItemSelected is not called when new game item is selected

Changing The Menu

onCreateOptionsMenu

Called only once

public boolean onOptionsItemSelected(MenuItem item)

Called before each time menu is displayed

Make changes here

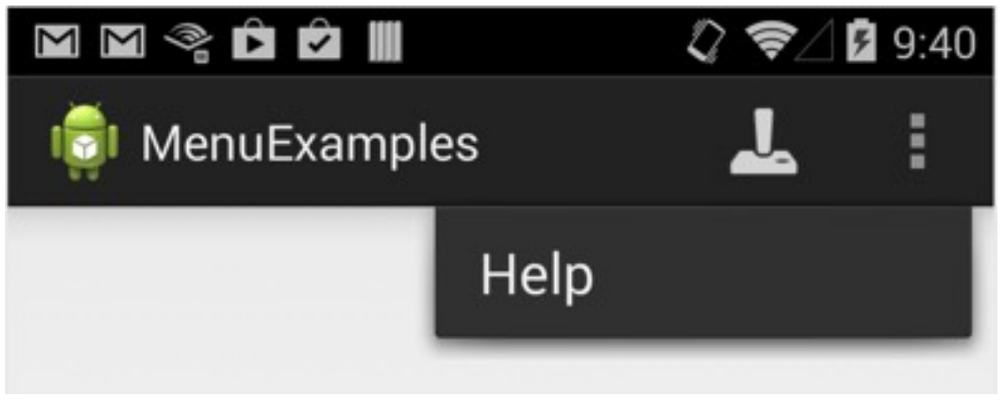
Add/remove item

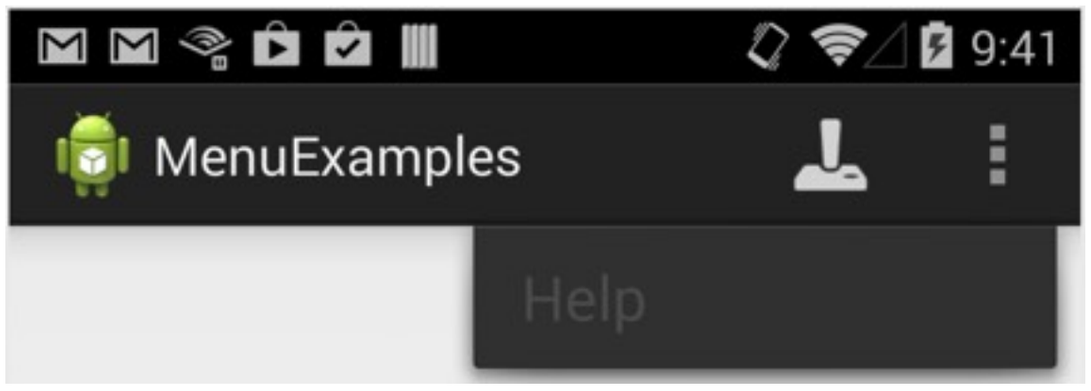
Enable/Disable

Hide/show

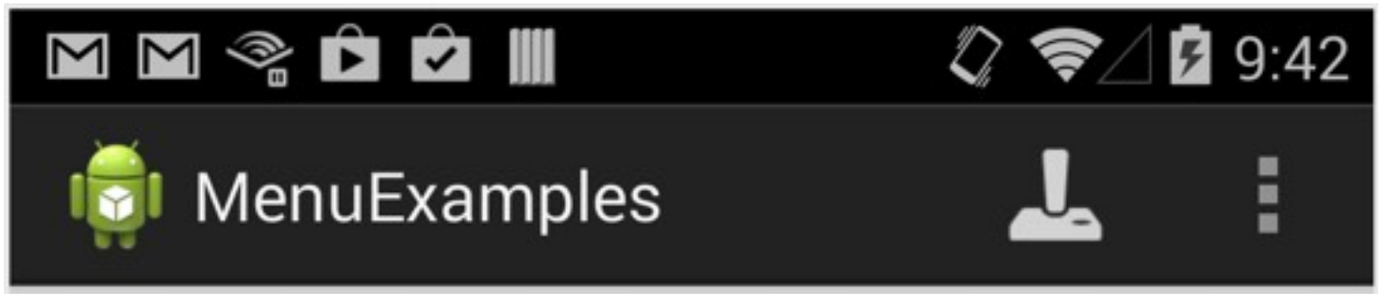
Example - Enable/Disable

```
public boolean onPrepareOptionsMenu (Menu menu) {
        MenuItem help = menu.findItem(R.id.help);
        if (help.isEnabled())
                help.setEnabled(false);
        else
                help.setEnabled(true);
        return true;
}
```





Android 3.0+ & Action Bar

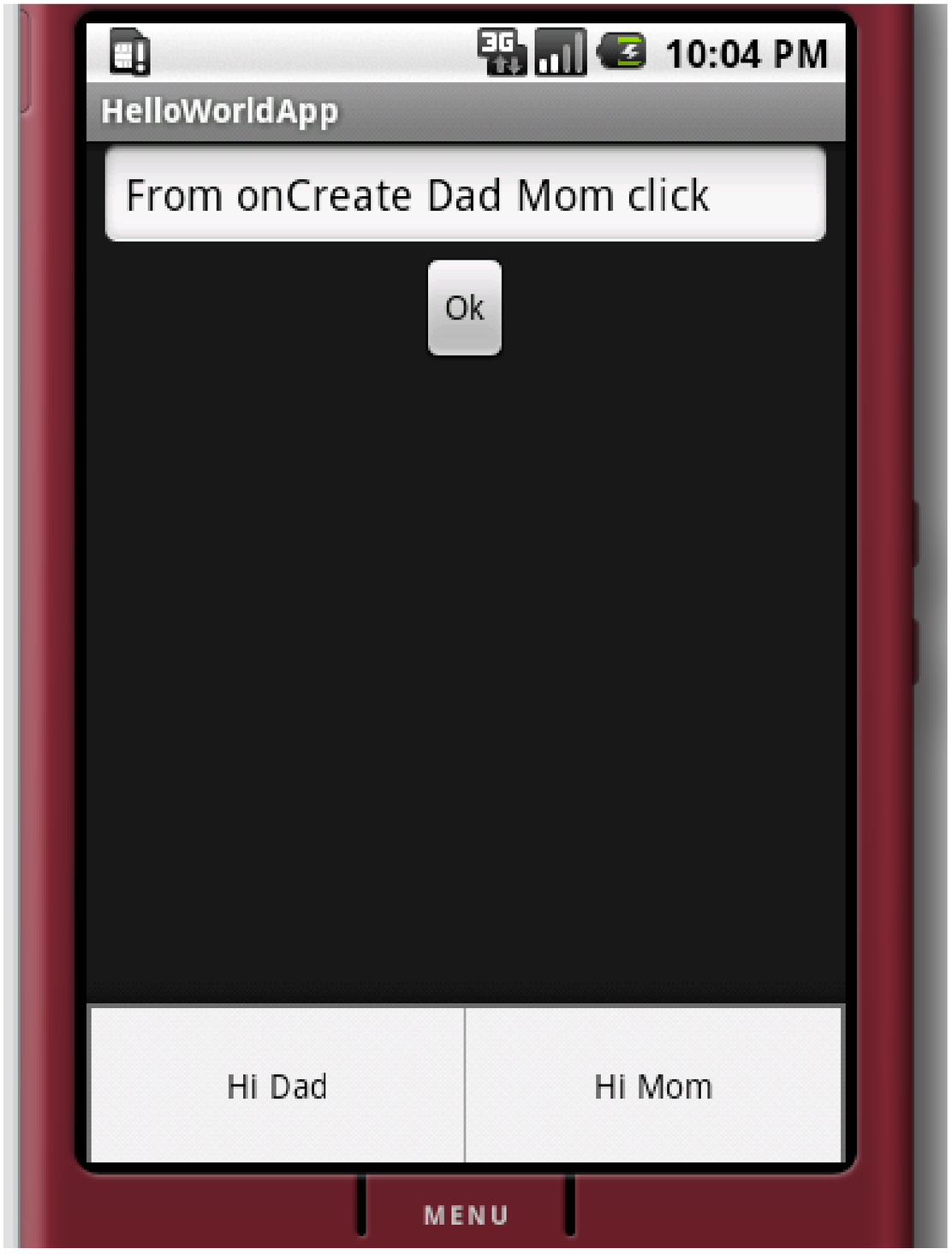


Menu items in the action bar are always shown

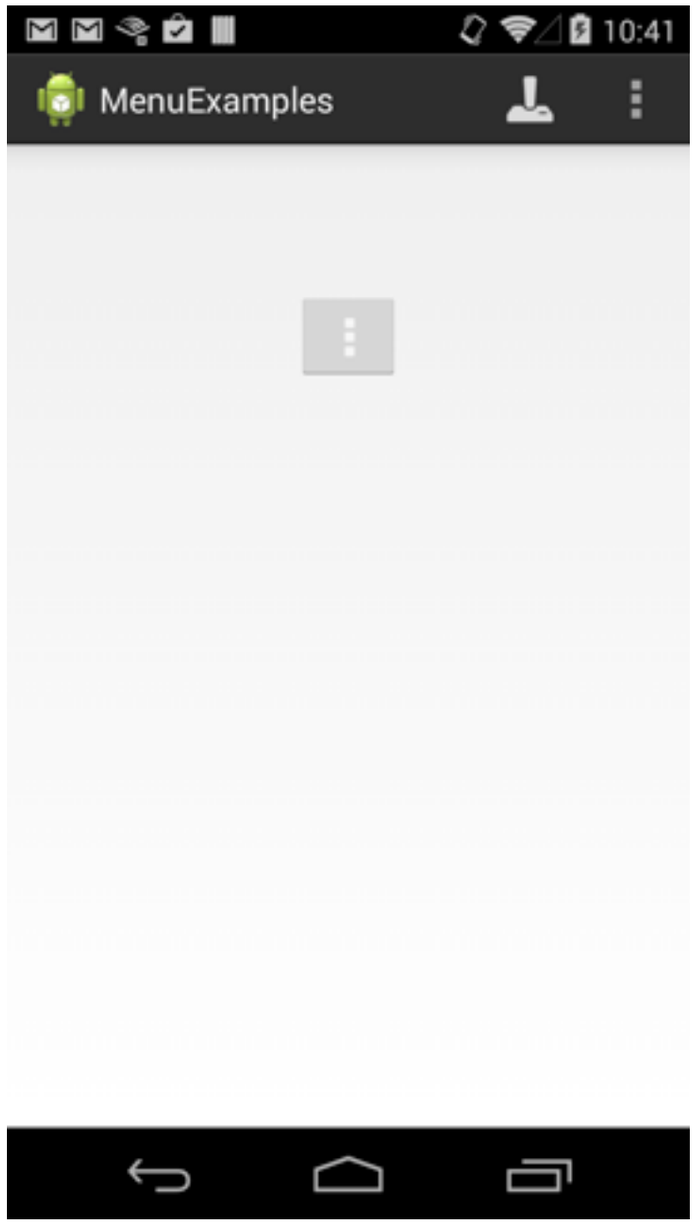
If you want to modify them you need to force onPrepareOptions to be called

Call invalidateOptionsMenu()

Pre Android 3 emulator



Pop Up Menus



Layout

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity" >

<ImageButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="74dp"
    android:contentDescription="@string/description_overflow_button"
    android:onClick="showPopup"
    android:src="@drawable/ic_action_overflow" />

</RelativeLayout>
```

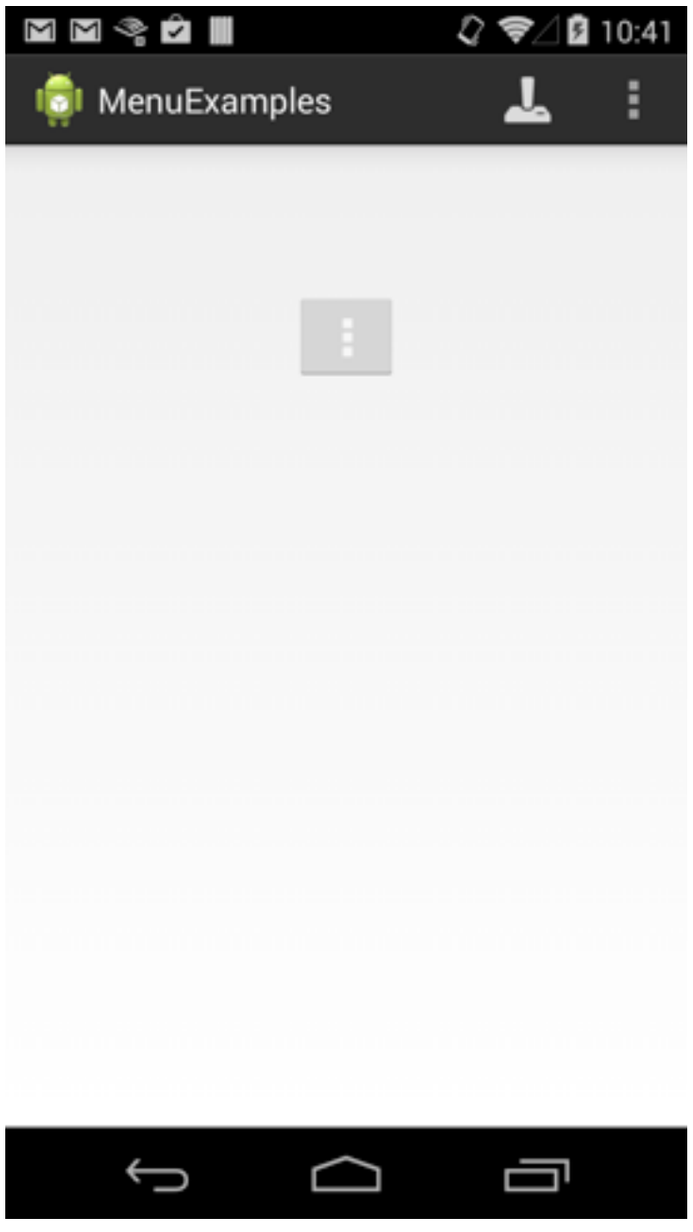
showPopup method

```
@TargetApi(Build.VERSION_CODES.HONEYCOMB)
public void showPopup(View v) {
    PopupMenu popup = new PopupMenu(this, v);
    MenuInflater inflater = popup.getMenuInflater();
    inflater.inflate(R.menu.actions, popup.getMenu());
    popup.show();
}
```

actions menu resource

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android" >
    <item android:id="@+id/run"
          android:titleCondensed="@string/run_condensed"
          android:title="@string/run"></item>
    <item android:id="@+id/jump" android:title="@string/jump">
        <menu>
            <item android:id="@+id/left" android:title="@string/left"/>
            <item android:id="@+id/right" android:title="@string/right"/>
        </menu>
    </item>
</menu>
```

Submenus


10:41
MenuExamples


10:42
MenuExamples
Run
Jump


10:50
MenuExamples
Left
Right

Contextual Menus

Menu for a specific item or view

Mainly used for ListViews or GridViews

Two types

Floating Context menu

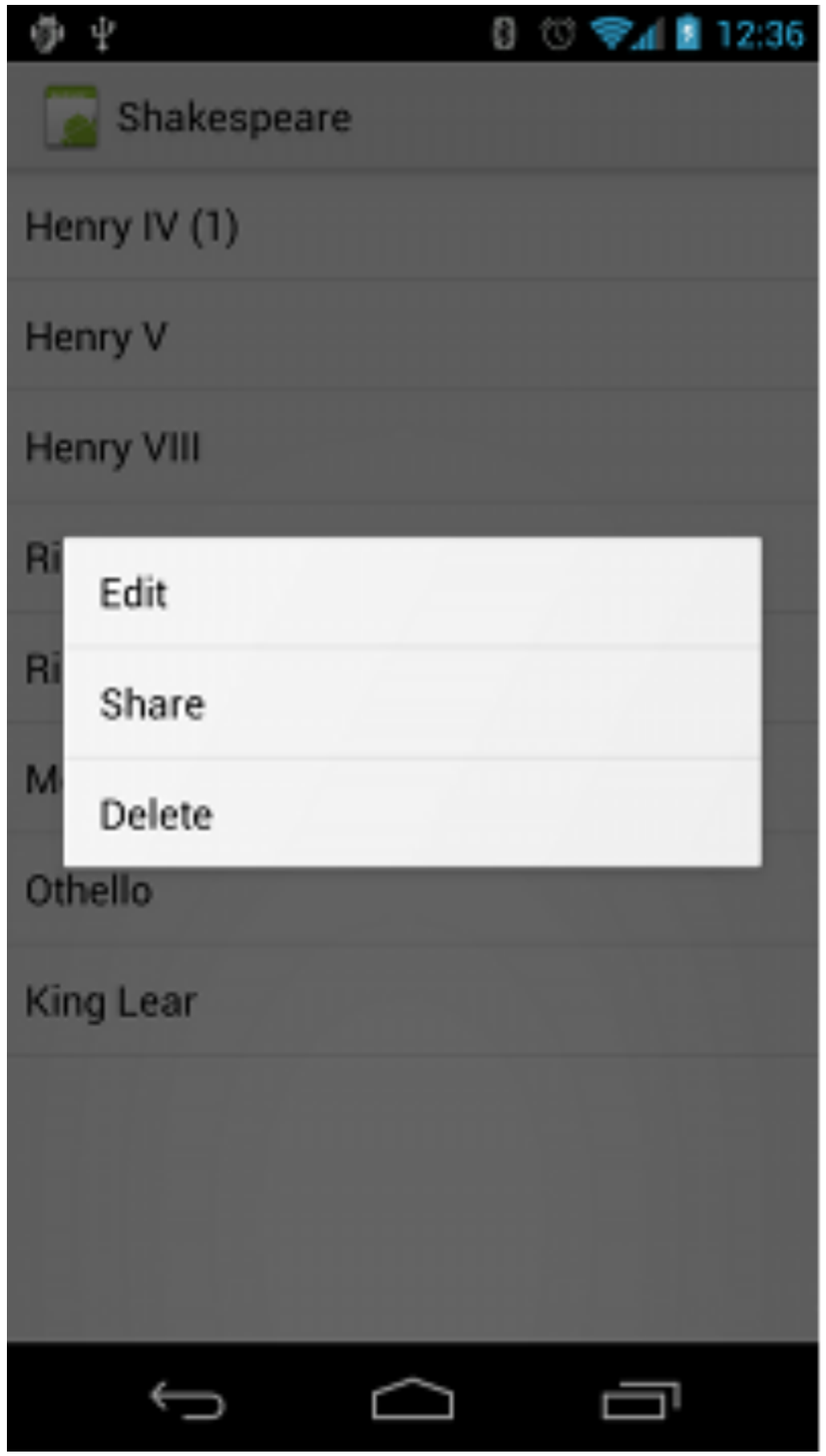
One item at a time

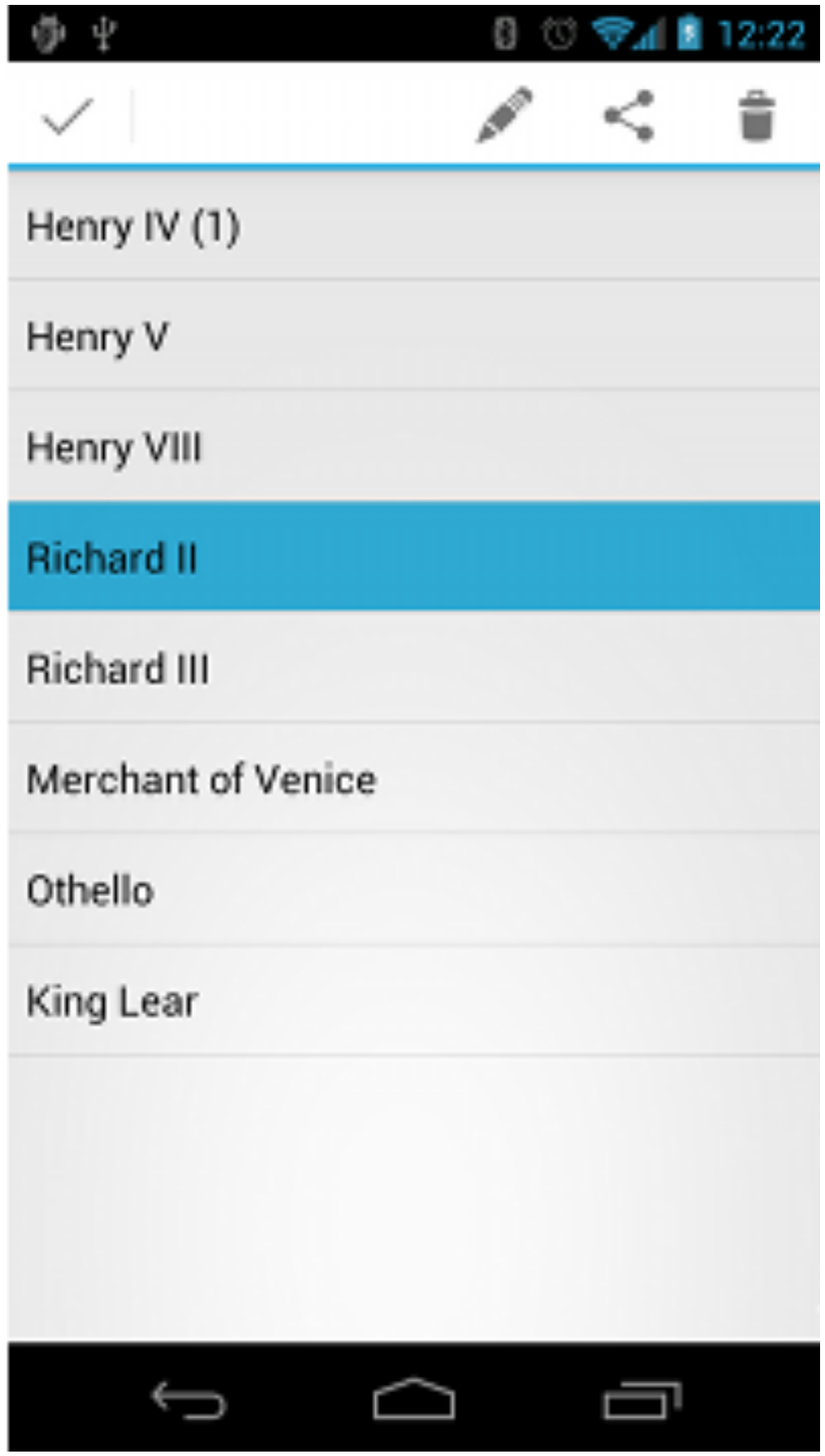
Contextual action mode

Can handle multiple times

Android 3.0 +

Floating menu vs Contextual action bar





Contextual Menus - How to

http://developer.android.com/guide/topics/ui/menus.html

http://tinyurl.com/yz7cytu

Menu Groups

Collection of menu items that share certain traits

Show or hide all items with setGroupVisible()
Enable or disable all items with setGroupEnabled()
Specify whether all items are checkable with setGroupCheckable()

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:id="@+id/menu_save"
        android:icon="@drawable/menu_save"
        android:title="@string/menu_save" />
    <!-- menu group -->
    <group android:id="@+id/group_delete">
      <item android:id="@+id/menu_archive"
          android:title="@string/menu_archive" />
      <item android:id="@+id/menu_delete"
          android:title="@string/menu_delete" />
    </group>
```

Menu to Start Activities from Other Apps

Menu actions can start new activities

Can start activities from other apps

What if device does not have the other app?

Can add Menu Items Based on an Intent

http://developer.android.com/guide/topics/ui/menus.html

http://tinyurl.com/yz7cytu

Fragments & Menus

Fragments can add menus

In onCreateView call
setHasOptionsMenu(true);

public void onCreateOptionsMenu(Menu menu, MenuInflater inflater)
Create menu

public boolean onOptionsItemSelected(MenuItem item)
handle menu

public void onPrepareOptionsMenu (Menu menu)
Modify menu

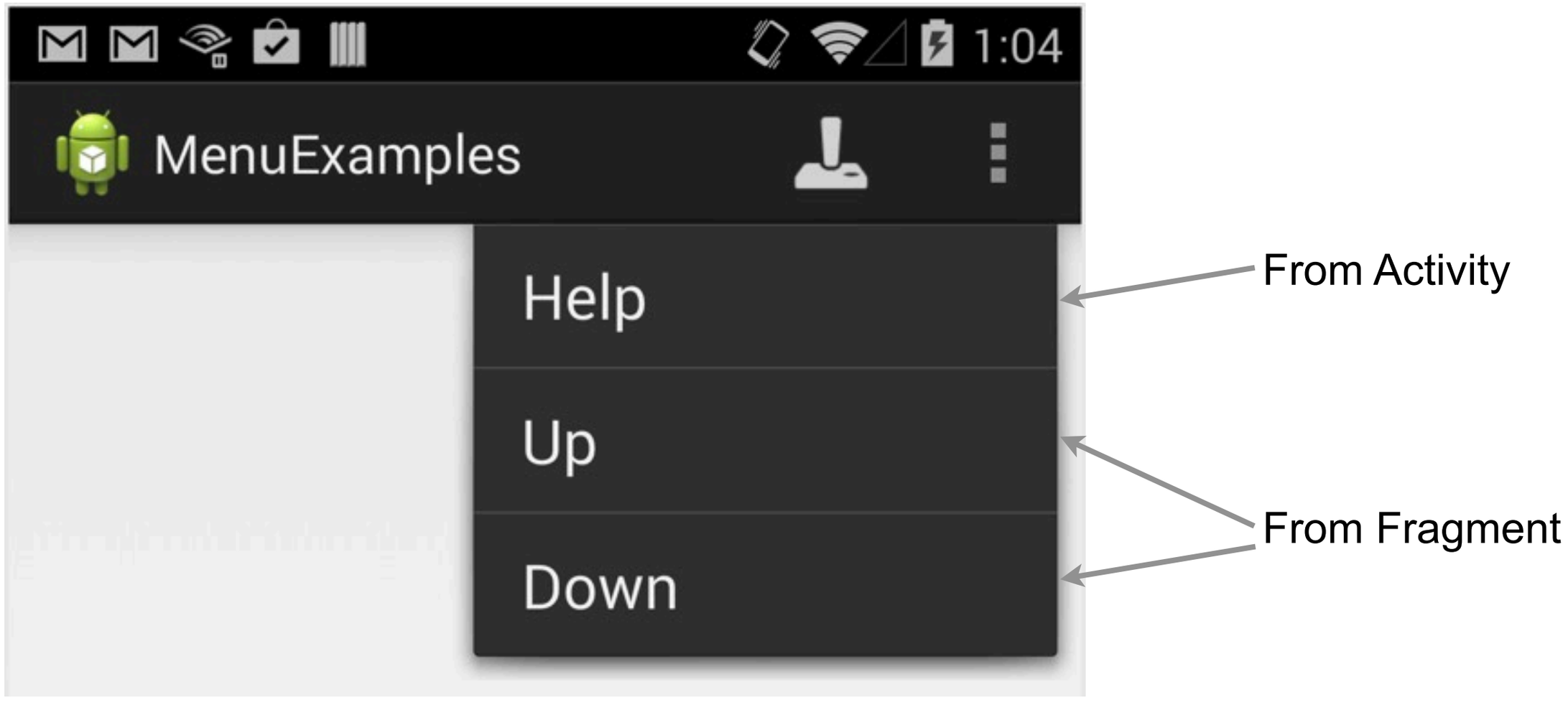
Example

```
public class MenuFragment extends Fragment {
      public View onCreateView(LayoutInflater inflater, ViewGroup container,
                  Bundle savedInstanceState) {
            View resultView =  inflater.inflate(R.layout.menu_fragment, container, false);
            setHasOptionsMenu(true);
            return resultView;
      }
```

```
public void onCreateOptionsMenu(Menu menu, MenuInflater inflater) {
      Log.i("rew","on onCreateOptionsMenu fragment");
   inflater.inflate(R.menu.menu_for_fragment, menu);
}
```

Fragment & Activity Menus

Fragment Option menus come after Activity Option menus



Fragment & Activity Methods

Fragment menu methods are called after the Activity methods

Fragment has access to menu item from Activity
But don't

Activity onCreateOptionsMenu
Fragment onCreateOptionsMenu
Activity onPrepareOptionsMenu
Fragment onPrepareOptionsMenu
Activity onOptionsItemSelected
Fragment onOptionsItemSelected

Fragment onClick in Menu

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android" >
    <item android:id="@+id/up"
        android:title="@string/up"
       ></item>
    <item android:id="@+id/down"
        android:title="@string/down"
        android:onClick="down"></item>
</menu>
```

Calls the method in the Activity :(