

Android

Interfaz de usuario

Enrique López-Mañas

Contenido

Entendiendo la interfaz de usuario

Diseño de la interfaz

Capturar eventos de la interfaz

Responder a cambios de orientación

Notificaciones al usuario

Entendiendo la UI

Dos maneras de definir la IU:

XML: (p.ej., main.xml en la carpeta res/layout)

En tiempo de ejecución

Interfaz contiene Views y ViewGroups

Todos los Views deben colocarse dentro de un ViewGroup

Unidades de medida

Para especificar el tamaño de un elemento

dp (píxel independiente de la densidad. 160dp equivalen a 1 pulgada de pantalla física)

sp (píxel independiente de la escala, se usa en tamaño de fuente de letra)

pt (punto. Equivale a $1/72$ pulgadas, basada en el tamaño de la pantalla)

px (píxel. Equivale a los pixeles de la pantalla, no recomendable ya que no renderizará correctamente)

Views y ViewGroups

Tipos de ViewGroups

LinearLayout

TableLayout

RelativeLayout

FrameLayout

ScrollView

Linear Layout

Coloca los elementos en una columna o en una fila.

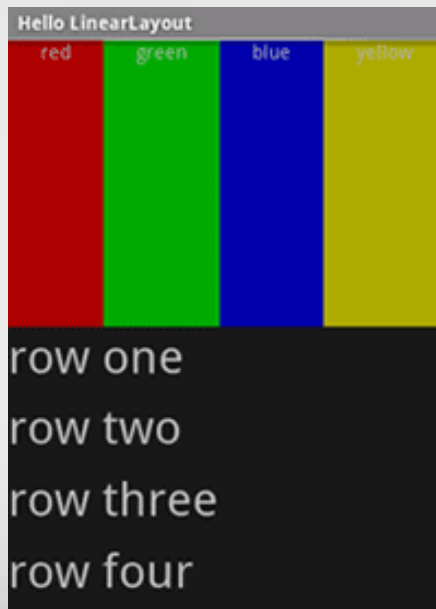
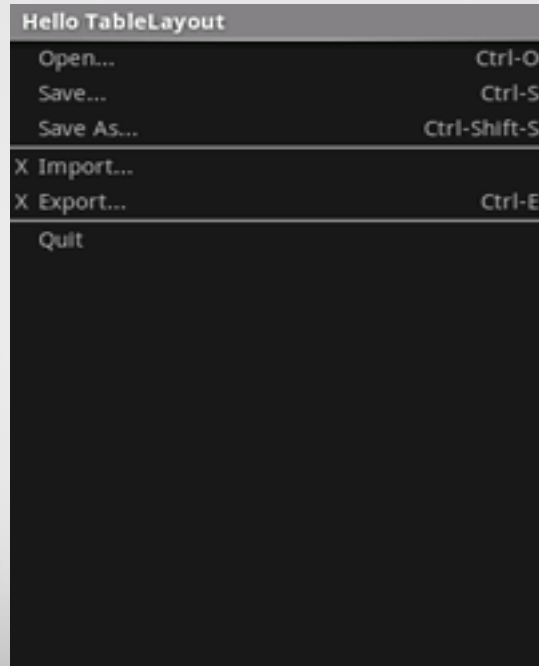


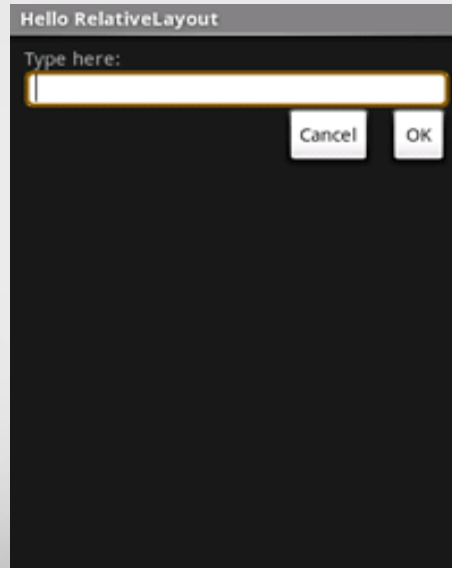
Table Layout

Agrupar Views en filas y columnas



RelativeLayout

Permite especificar la posición relativa entre las Views



Diseño de la UI

Mediante archivos xml

```
<?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">
    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="@string/hello"/>
</LinearLayout>
```

En tiempo de ejecución

Ejemplo:

Creación de la interfaz en tiempo de ejecución

Capturar Eventos de la UI:

Hay más de una manera de interceptar los eventos de interacción del usuario

Extender la clase View y sobrecargar el método

EventListeners, relativos a cada View
onClick, onLongClick, onTouch, etc...

EventHandler, eventos del dispositivo
onKeyDown, onKeyUp, onTouchEvent

Extender Class View

```
public class ExampleActivity extends Activity implements OnClickListener {  
    protected void onCreate(Bundle savedInstanceState) {  
        ...  
        Button button = (Button)findViewById(R.id.corky);  
        button.setOnClickListener(this);  
    }  
  
    // Implement the OnClickListener callback  
    public void onClick(View v) {  
        // do something when the button is clicked  
    }  
    ...  
}
```

EventListener

```
// Create an anonymous implementation of OnClickListener
private OnClickListener mCorkyListener = new OnClickListener() {
    public void onClick(View v) {
        // do something when the button is clicked
    }
};

protected void onCreate(Bundle savedInstanceState) {
    ...
    // Capture our button from layout
    Button button = (Button)findViewById(R.id.corky);
    // Register the onClick listener with the implementation above
    button.setOnClickListener(mCorkyListener);
    ...
}
```

Cambios de Orientación

Cuando cambia la orientación del dispositivo,
la Activity visible se destruye y se vuelve a crear

Es necesario adaptarse a esos cambios

- Anclar Views

- Reposicionar y cambiar el tamaño de los Views

Ejemplo

Layout que responde a los cambios de orientación

Cómo anclar Views

Cómo reposicionar y cambiar el tamaño

res/layout

res/layout-land

Notificaciones al usuario

Diálogos

AlertDialog

ProgressDialog

DatePickerDialog

TimePickerDialog

Menús

OptionsMenu

ContextMenu

Barra de notificaciones

Activity como Dialog

Ejemplo

Creación de diálogos

Utilización de menús

Notificaciones

Activity como Dialog

Aplicar estilos