

E0701 Canvas

Creación de una vista personalizada

A continuación se muestra un ejemplo donde se crea una vista que es dibujada por código por medio de un `Canvas`.

1. Crea un nuevo proyecto `E0701_Canvas` con los siguientes datos:

Application name: `E0701_Canvas`
Package name: `com.examples.e0701_canvas`
Minimum Required SDK: API 7 Android 2.1 (Eclair)
Activity Name: `EjemploGraficosActivity`
Layout Name: `main.xml`

2. Reemplaza el código de la actividad por:

```
public class EjemploGraficosActivity extends Activity {
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(new EjemploView(this));
    }

    public class EjemploView extends View {
        public EjemploView (Context context) {
            super(context);
        }
        @Override
        protected void onDraw(Canvas canvas) {
            //Dibujar aquí
        }
    }
}
```

Comienza con la creación de una `Activity`, pero en este caso, el objeto `View` que asociamos a la actividad mediante el método `setContentView()` no está definido mediante XML. Por el contrario, es creado mediante código a partir del constructor de la clase `EjemploView`.

La clase `EjemploView` extiende `View`, modificando solo el método `onDraw()` responsable de dibujar la vista. A lo largo del capítulo iremos viendo ejemplos de código que pueden ser escritos en este método.

3. Si ejecutas la aplicación no se observará nada. Aprenderemos a dibujar en esta `Canvas` utilizando el objeto `Paint`.

4. Escribe dentro de `OnDraw` el código siguiente:

```
Paint pincel = new Paint();  
  
pincel.setColor (Color.BLUE);  
pincel.setStrokeWidth (8);  
pincel.setStyle (Style.STROKE);  
canvas.drawCircle (150, 150, 100, pincel);
```

Ejecuta la aplicación para ver el resultado.

5. Aprovechando la opción de autocompletar de Eclipse, prueba otros valores para `Color` y `Style`.
6. Prueba otros métodos de dibujo, como `drawLine()` o `drawPoint()`.

Parte 2: Definiendo Colores

1. Modifica el ejercicio anterior, añadiendo al final de `OnDraw` el código siguiente:

```
pincel.setColor(Color.rgb(127,255,0,0));  
canvas.drawCircle(150, 250, 100, pincel);
```

Observa como el color rojo seleccionado es mezclado con el color de fondo. Prueba otros valores de alfa.

2. Reemplaza la primera línea que acabas de introducir por:

```
pincel.setColor(0x7FFF0000);
```

Observa como el resultado es idéntico.

3. Define en el fichero `res/values/colors.xml` un nuevo color utilizando el siguiente código:

```
<?xml version="1.0" encoding="utf-8"?>  
  
<resources>  
  <color name="color_circulo">#7fffff00</color>  
</resources>
```

4. Modifica el ejemplo anterior para que se utilice este color definido en los recursos:

```
pincel.setColor(getResources().getColor(R.color.color_circulo));
```