

Ejercicio 1. Personalización de aspecto

- Crear una app llamada Drawables con un único TextView que ocupe todo el ancho de pantalla y tenga el texto de color negro. Asignar como fondo del TextView un elemento drawable que dibuje un marco de 5dp de grosor, en color azul y con esquinas redondeadas. Añade al drawable un gradiente lineal para el fondo, desde el color gris claro al blanco.

Ejercicio 2. Personalización de estados

- Añade un botón a la app anterior debajo del TextView y haz que ocupe todo el ancho. Para personalizar su aspecto crea un drawable tipo lista de estados, que asigne una imagen distinta a cada uno de los estados del botón. Busca una imagen para el fondo del botón que consista en un rectángulo alargado con los bordes redondeados. Modifica esta imagen para crear las tres que necesitamos: botón_normal.png con el fondo gris claro, botón_pressed.png con el fondo gris oscuro y botón_focused.png con el fondo amarillo. Asigna estas imágenes a cada uno de los estados y establécelo como fondo del botón.

Ejercicio 3. Animación por fotogramas

- Añade una vista de tipo ProgressBar centrada en horizontal que muestre un progreso indeterminado(el circular). A continuación, personaliza la animación de esta vista para que muestre un contador del 1 al 5.
- Crea cinco imágenes y dibuja en cada una de ellas el numero correspondiente. Crea un animation-list drawable a partir de dichas imágenes y configúralo para que se reproduzca de forma continuada cambiando de fotograma cada 1000ms. Reemplaza la animación del progressBar por la que acabamos de crear.(No es necesario escribir ni una línea de código java, sino que lo puedes hacer desde el layout mediante el atributo indeterminateDrawable).

Ejercicio 4. 9Patch

- Añade una nueva actividad llamada 9Patch, donde aparezcan seis botones, dos cuadrados, dos alargados y dos achatados. La mitad de ellos tendrán de fondo la imagen normal y los otros imágenes 9patch.
- Busca tres imágenes diferentes para cada uno de los botones

