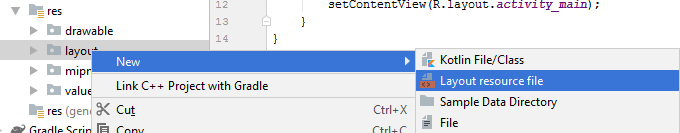
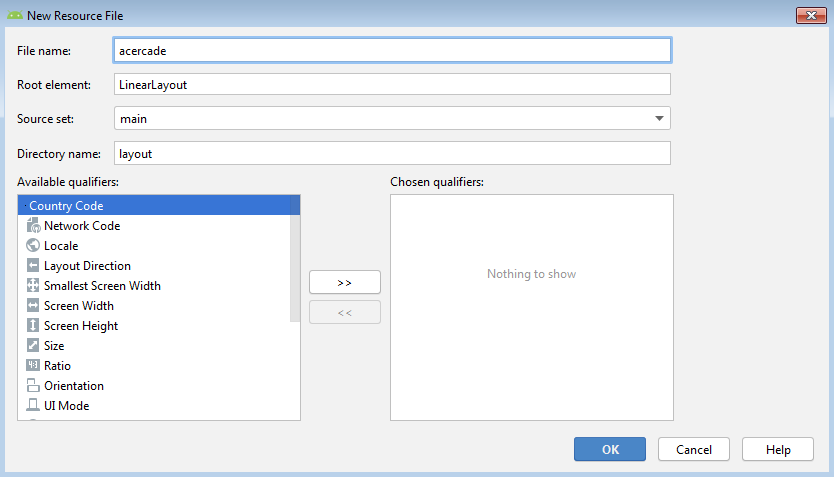
**PRACTICA 11: ASTEROIDES**

**Parte I: Implementación de una caja Acerca de**

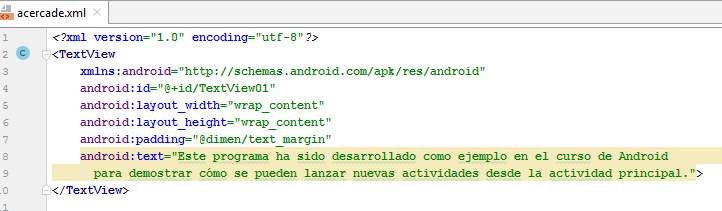
Dentro del proyecto Asteroides, vamos a crear una caja Acerca de… y visualizarla cuando se pulse el botón adecuado.

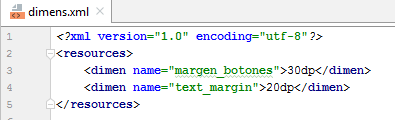
**Paso 1.** En primer lugar crea el fichero res/layout/acercade.xml. Para ello pulsa con el botón derecho sobre el explorador del proyecto en la carpeta res/layout y selecciona New > Layout resource file. Indica en File name: acercade.



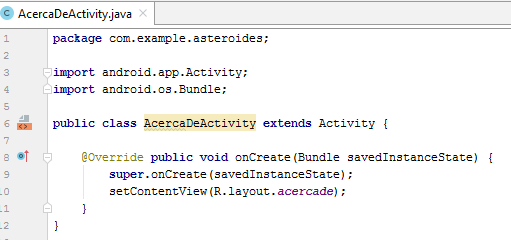


**Paso 2.** Selecciona la lengüeta Text y copia el siguiente contenido:

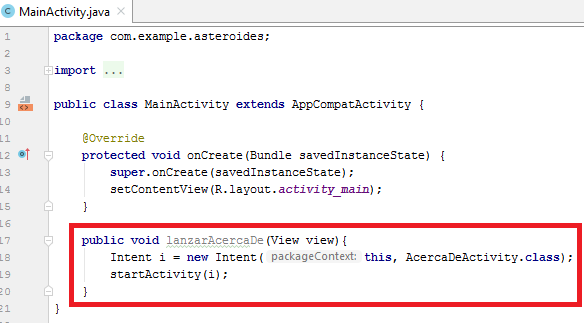
****

****

**Paso 3.** Creamos ahora una nueva actividad, que será la responsable de visualizar esta vista. Para ello crea el fichero AcercaDeActivity.java, pulsando con el botón derecho sobre el nombre del paquete de la aplicación y seleccionando New > Java Class. En el campo Name introduce AcercaDeActivity y pulsa Finish. Reemplaza el código por:

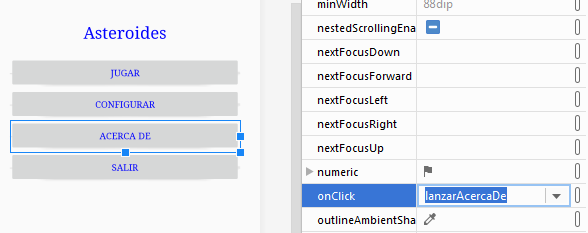


**Paso 4.** Pasemos ahora a crear un método en la clase Asteroides.java que será ejecutado cuando sea pulsado el botón Acerca de.



**Nota sobre Java:** Pulsa ***Alt-Intro***, para que automáticamente se añadan los paquetes que faltan.

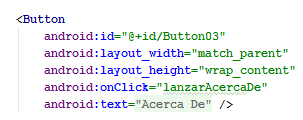
**Paso 5.** Para asociar este método al botón edita el Layout activity\_main.xml. Selecciona la lengüeta Design y pulsa sobre el botón Acerca de… y en la vista Properties busca el atributo onClick e introduce el valor lanzarAcercaDe.



Se ha de hacer esto para las el activity\_main.xml y para el activity\_main.xml (land).

**Paso 6.** Pasa a la edición xml pulsando en la lengüeta main.xml y observa como en la etiqueta <Button> correspondiente, se ha añadido el atributo:

android:onClick="lanzarAcercaDe"



**NOTA:** *En caso de que exista algún recurso alternativo para el layout repite el mismo proceso.*

**Paso 7.** Ejecuta ahora la aplicación y pulsa en el botón Acerca de. Observarás que el resultado no es satisfactorio ¿Qué ha ocurrido?

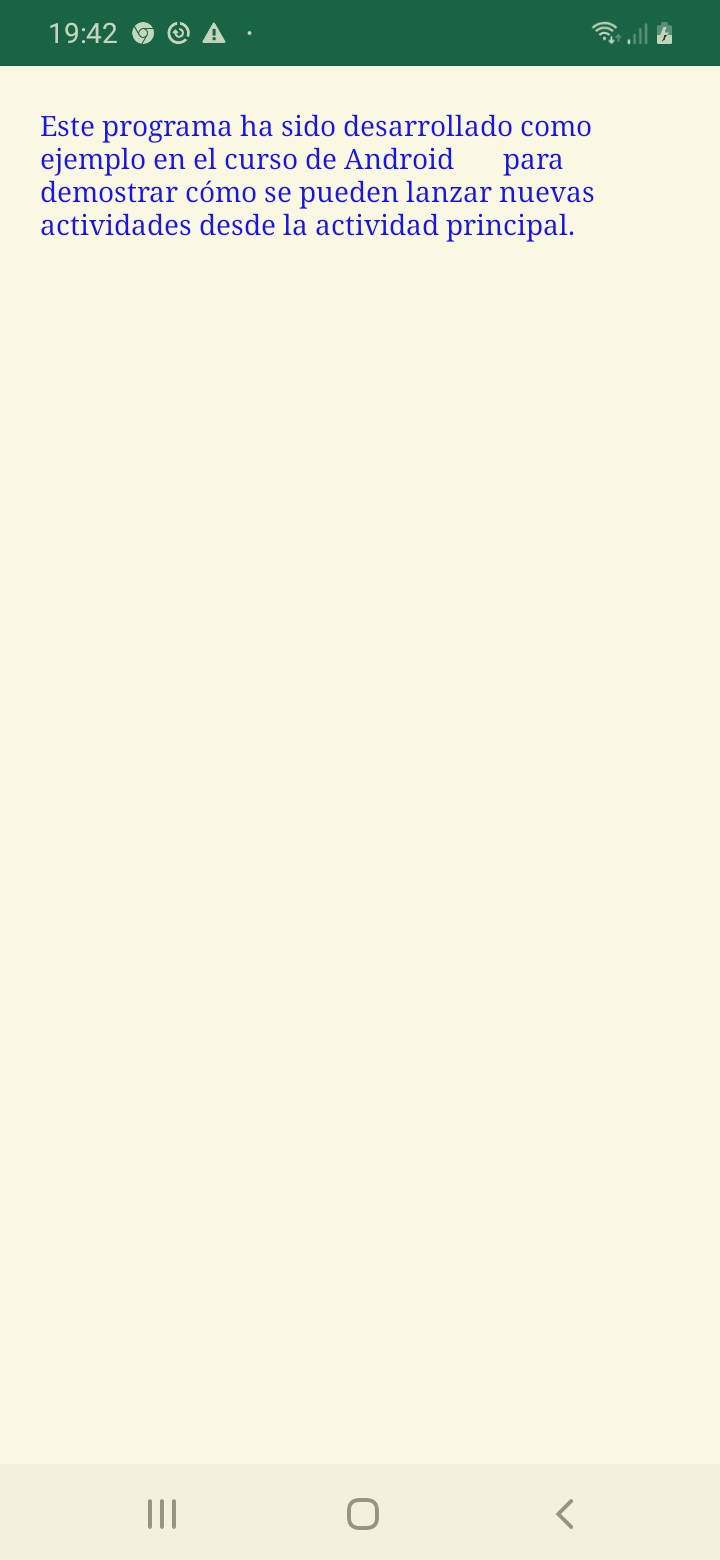
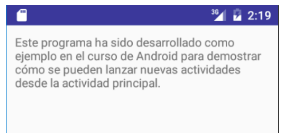


El problema es que toda actividad que ha de ser lanzada por una aplicación ha de ser registrada en el fichero AndroidManifest.xml. Para registrar la actividad, abre AndroidManifest.xml. Añade el siguiente texto dentro de la etiqueta <application …> </ application>:

<activity android:name=".AcercaDeActivity" android:label="Acerca de ..."/>



**Paso 8.** Ejecuta de nuevo el programa. El resultado ha de ser similar al mostrado a continuación:



La vista mostrada en el ejemplo anterior no parece muy atractiva. Tratemos de mejorarla aplicando un tema. Como vimos en el capítulo anterior un tema es una colección de estilos que define el aspecto de una actividad o aplicación. Puedes utilizar alguno de los temas disponibles en Android o crear el tuyo propio.

**Paso 9.** En este caso utilizaremos uno de los de Android. Para ello abre AndroidManifest.xml e introduce la línea subrayada:



**Paso 10.** Ejecuta de nuevo el programa y observa cómo mejora la apariencia:



**Parte II: Un escuchador de evento por código**

Como acabamos de ver en un layout podemos definir el atributo XML android:onClick que nos permite indicar un método que será ejecutado al hacer click en una vista. A este método se le conoce como escuchador de evento. Resulta muy sencillo y además está disponible en cualquier descendiente de la clase View. Sin embargo esta técnica presenta dos inconvenientes. Solo está disponible para el evento onClick(). La clase View tiene otros eventos (onLongClick(), onFocusChange(), onKey(),…) para los que no se han definido un atributo xml. Entonces, ¿qué hacemos si queremos definir un evento distinto de onClick(). La respuesta la encontrarás este ejercicio:

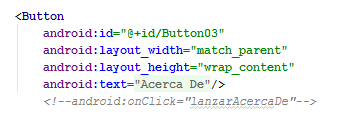
**Paso 1.** Abre la clase Asteroides.java y añade las líneas que aparecen subrayadas:



**Nota sobre Java:** Pulsa ***Alt-Intro*** para que automáticamente se añadan los paquetes que faltan. Para la clase OnClickListener selecciona android.view.View.OnClickListener.

**Paso 2.** Elimina el atributo añadido al botón:

android:onClick="lanzarAcercaDe"



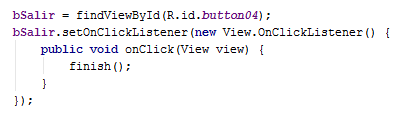
**Paso 3.** Ejecuta la aplicación. El resultado ha de ser idéntico al anterior.

**NOTA:** *Más adelante se estudiará con más detalle los escuchadores de evento.*

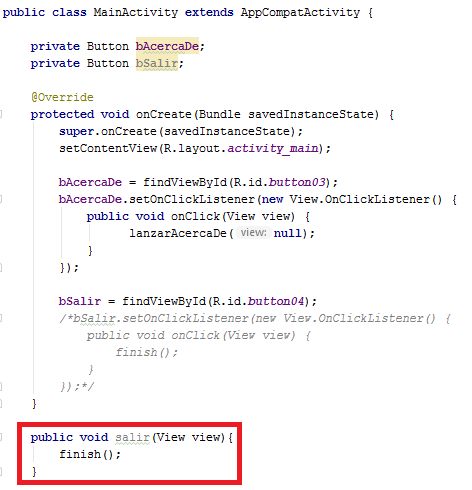
**Parte III: El botón Salir**

En el layout activity\_main.xml (o content\_main.xml en Mis Lugares) hemos introducido un botón con texto “Salir”. Queremos que cuando se pulse este botón se cierre la actividad. Para cerrar una actividad puedes llamar al método finish(); Llamar a este método es equivalente a cuando un usuario pulsa la tecla «retorno».

**Paso 1.** Realiza este trabajo utilizando un escuchador de evento por código. Para resolverlo añade en el método onCreate() de la clase MainActivity el siguiente código:



**Paso 2.** Hazlo ahora con el atributo xml android:onClick. Para resolverlo con el atributo onClick, añade en MainActivity el método salir(). Y añade el siguiente atributo al botón “salir” en el layout activity\_main.xml: android:onClick="salir"

****

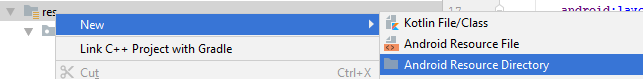
**Paso 3.** Verifica que el resultado es el mismo en ambos casos.

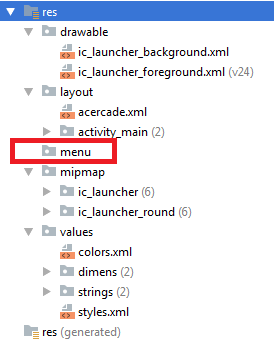
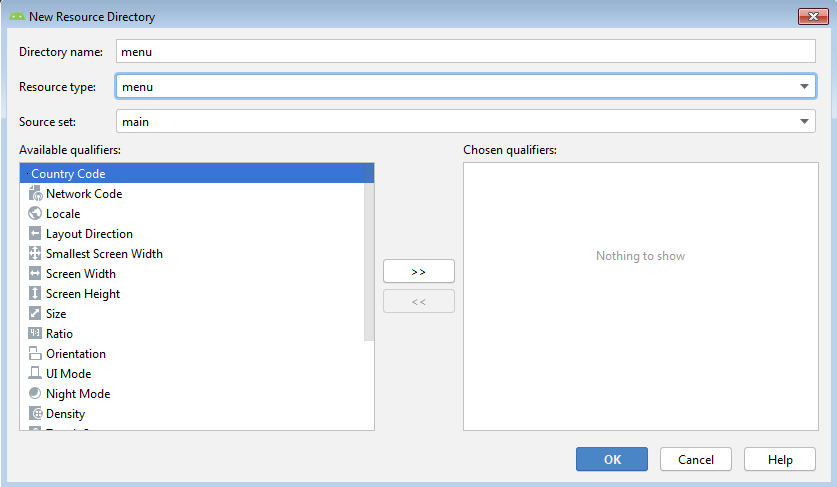
**NOTA:** *No es conveniente que en tus actividades incluyas un botón para cerrarlas. Un dispositivo Android siempre dispone de la tecla «retorno», que tiene la misma función.*

**Parte IV: Añadiendo un menú a una actividad**

Podemos asignar un menú a nuestra actividad de forma muy sencilla.

**Paso 1.** Abre el proyecto Asteroides. Crea una carpeta de recursos Android llamada menú

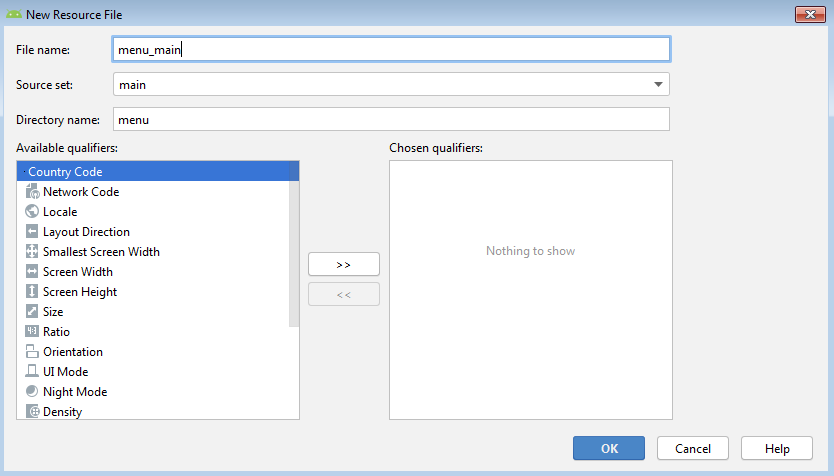




**Paso 2.** Crea un nuevo menú, usa File > New > Android resource file. En el campo File name: selecciona menu\_main y en el campo Resource type: selecciona Menu.

Si desarrollas Mis Lugares abre el fichero res / menu / menu\_main.xml.

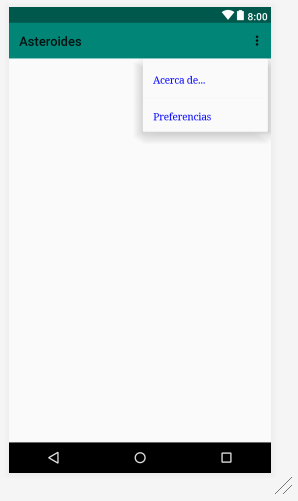




**NOTA:** *El fichero de menú se crea automáticamente si seleccionas una actividad de tipo: Basic Activity o Scrolling Activity.*

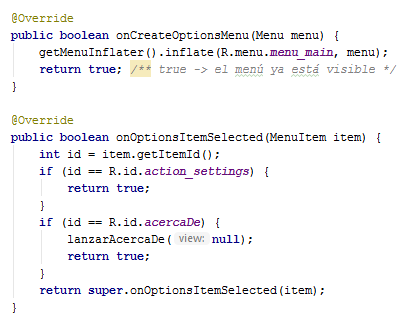
**Paso 3.** Reemplaza el contenido que se muestran a continuación:



Como puedes ver cada ítem de menú tiene tres atributos principales: id que permite identificarlo desde el código, title, para asociarle un texto e icon, para asociarle un icono. Los otros dos atributos se aplican cuando el menú actúa como una barra de acciones y serán explicados en el próximo punto.

**Paso 4.** A continuación se muestra la apariencia de este menú. Esta apariencia puede cambiar dependiendo de la versión de Android.

**Paso 5.** Para activar el menú, has de introducir el siguiente código en *l*a actividad que muestra el menu. Posiblemente solo tengas que incluir el texto subrayado.



**Paso 6.** Ejecuta el programa y pulsa en el icono del menú en la esquina superior derecha. Han de aparecer los dos ítems de menú. Selecciona Acerca de… para pasar a la actividad correspondiente.

