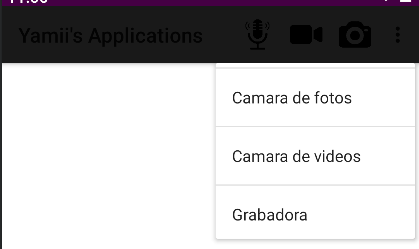
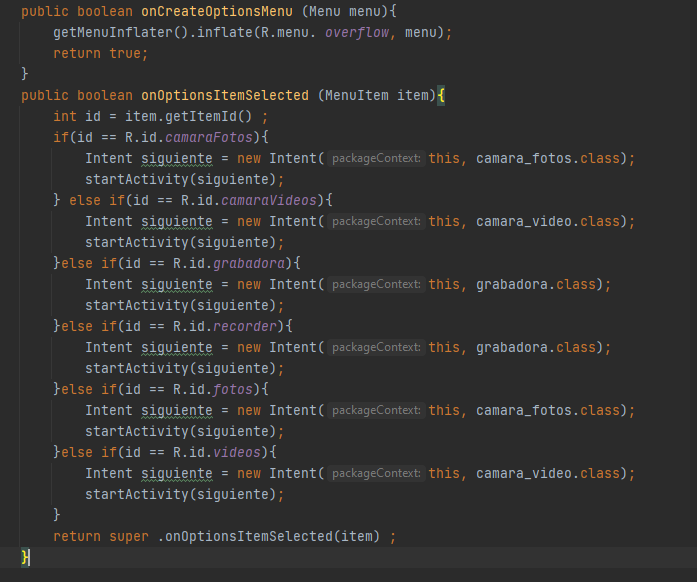
**GUIA ANDROID**

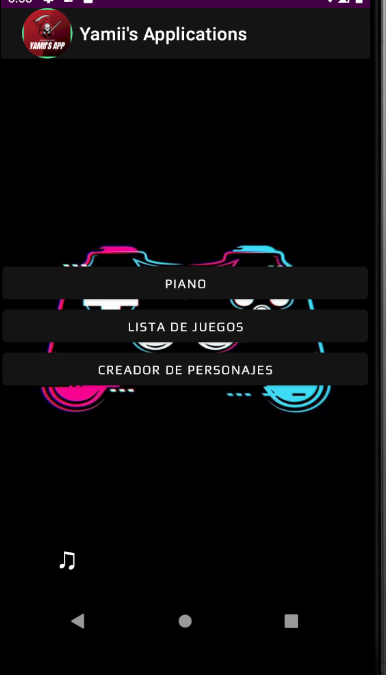
1. **Idea del Proyecto**

La idea del proyecto era desarrollar 3 tipos de apartado totalmente diferentes pero sin desvariar la temática de juegos.

Todas las clases tienen un overflow

1. **Ventana principal**



La ventana principal cuenta con un total de 4 botones.

1. PIANO

Nos envia a la aplicación del piano

1. Lista de juegos

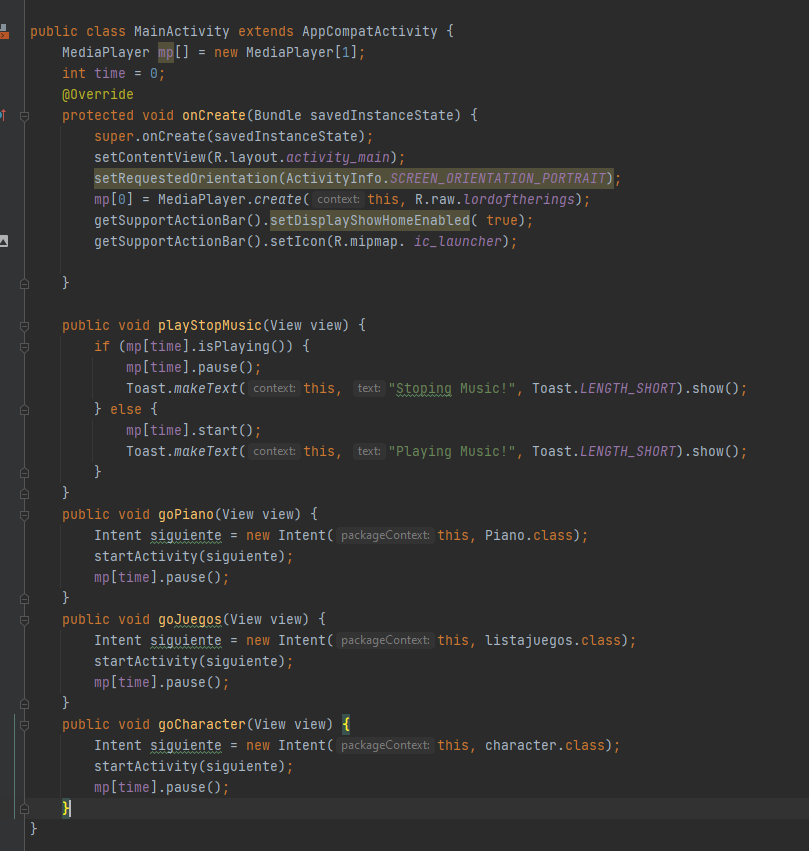
Nos envía a un buscar de juegos según su genero

1. Creador de personajes

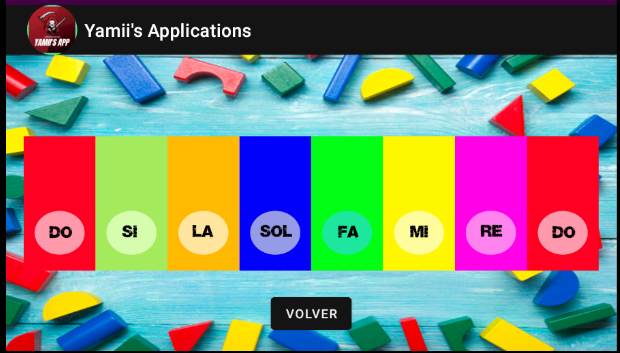
Nos envía a un creador de personaje tipo RPG sujeto en una BBDD

1. Botón Música

♫ Contiene como texto este icono. Al pulsarlo este empieza a reproducir una OST de The Lord Of The Rings, recogiendo también el valor del tiempo en el que se paro y pudiendo reproducir por el mismo minuto que se dejó. Este valor se reinicia cada vez que entramos a una activity, y también el reproductor se pausa por cada pestaña que abramos



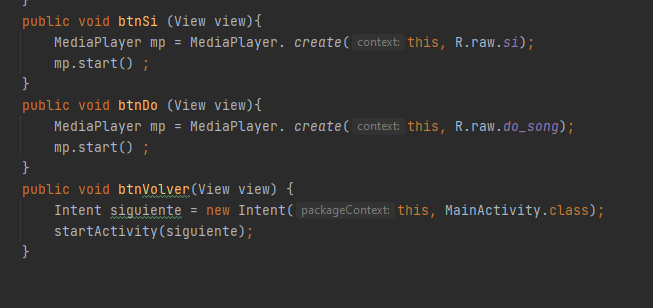
1. **Ventana Piano**



Esta ventana automáticamente se coloca en modo Landscape

Cada ImageButton tiene asociado su propio sonido de piano.

SONIDOS GRABADOS POR MI

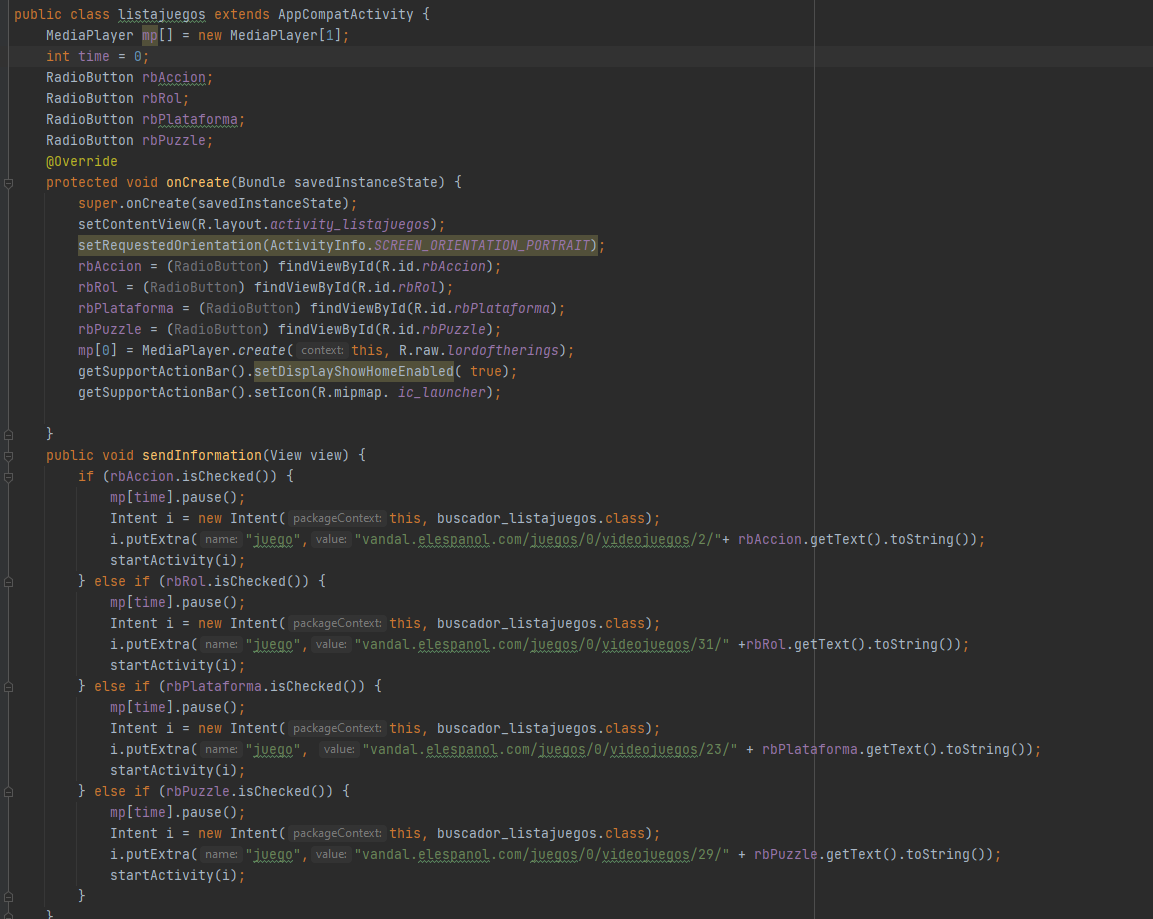


1. **Ventana lista juegos**



El usuario solo debe de Clickear cualquier radioButton y después clickear Buscar y se abrirá una WebView con los juegos del genero que haya seleccionado el usuario







1. **Ventana Creador Personaje**

La ventana esta contenida en un ScrollView por si el teléfono es mas pequeño de lo normal y distribuida en TableLayout



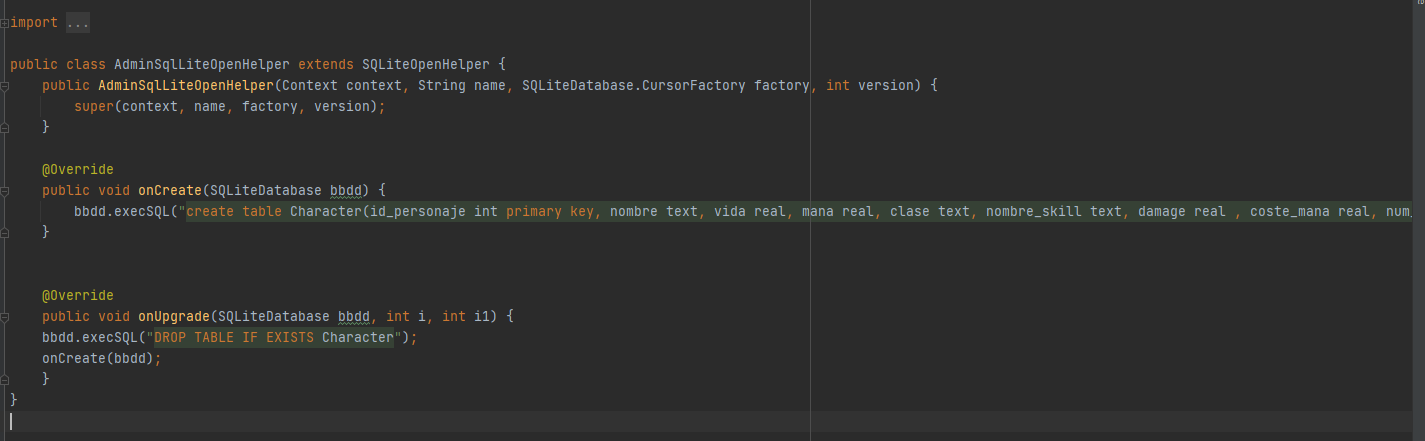


El creador de personaje consta de 9 atributos para el personaje

Codigo del personaje, su nombre, su total de vida, su total de mana, su clase, después para destacar un poco mas que tiene habilidad de “combate”, tenemos, el nombre de la habilidad, el daño que realiza, el coste de mana que supone usarla, y la cantidad de veces que podemos usarla.

Todos estos atributos se guardan en una BBDD que contiene una tabla tal que asi.

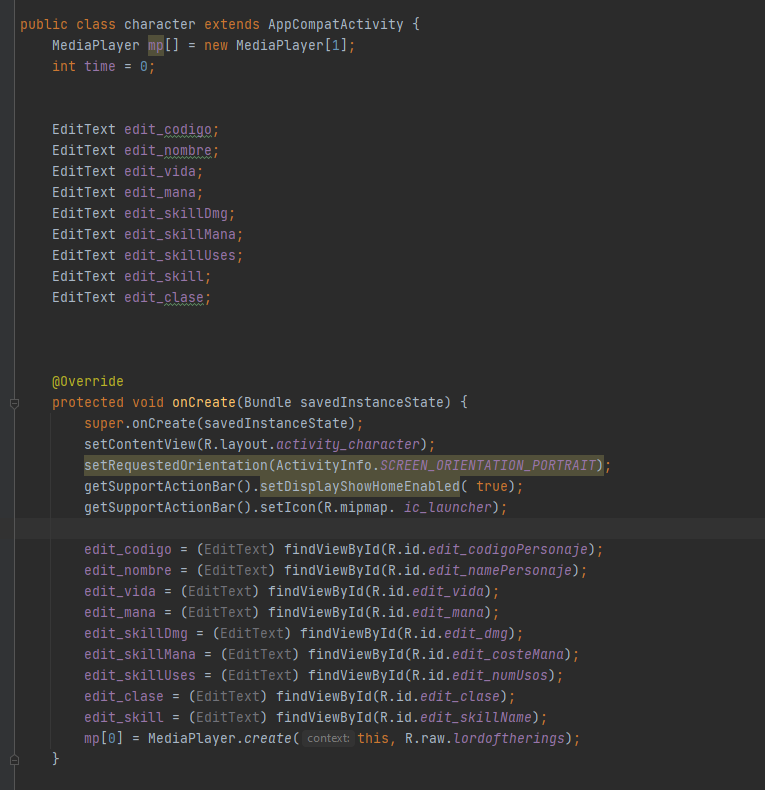
Antes de todo para usar la BBDD tenemos que generar esta clase



Y en la parte de onCreate debemos ejecutar la consulta SQL que en este caso es la creación de la tabla, que seria la siguiente

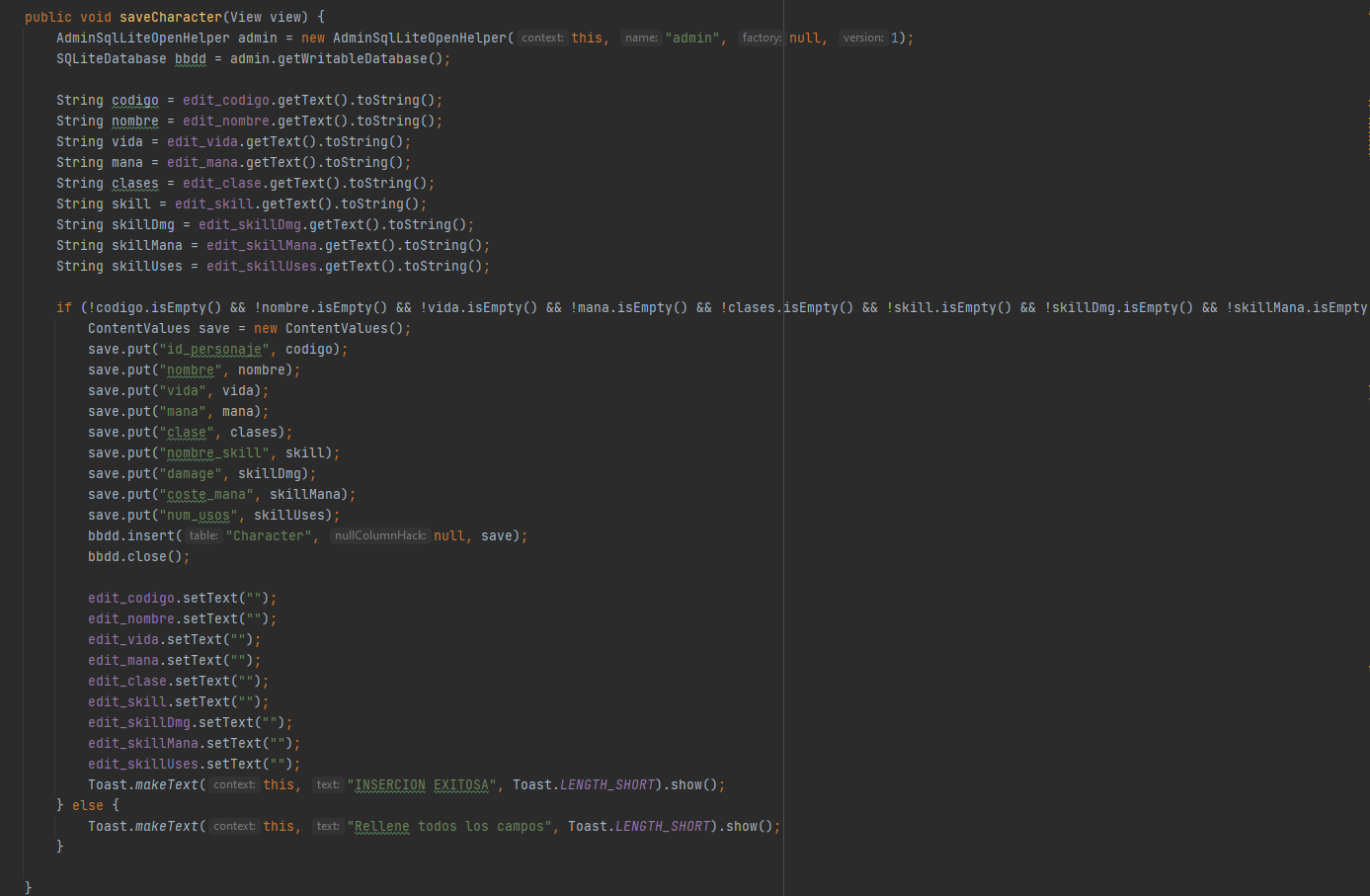
bbdd.execSQL("create table Character(id\_personaje int primary key, nombre text, vida real, mana real, clase text, nombre\_skill text, damage real , coste\_mana real, num\_usos real )");

METODOS:



De inicio recogemos todos los EditText y les asignamos sus respectivas Ids

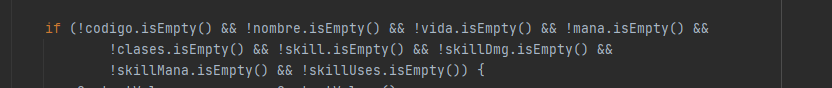
1. saveCharacter



Lo primero abrimos la conexión a nuestra base de datos.

Recogemos los valores de cada EditText.

Creamos una condición de verificación para comprobar si todos los EditText tienen contenido

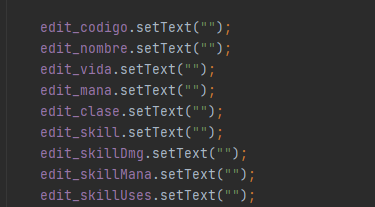


Creamos un objeto ContentValue para recoger y enviar valores a la BBDD

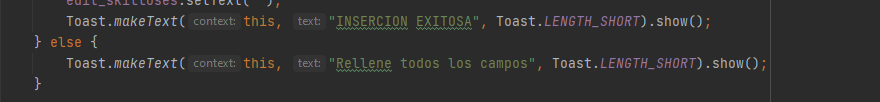
Asignamos a los nombres de la table de la BBDD sus respectivos EditText

Y después de tenerlos todos metidos en sus PUT lo Insertamos y cerramos la conexión



Para que el usuario tenga mayor facilidad para implementar otro, vaciamos todas las EditText 

Y para controlar si se a añadido correctamente o no, use unos Toast



1. searchCharacter

Iniciamos la conexión a la BBDD

Recogemos el código de la EditText

Comprobamos si el código no esta vacio.

Creamos un cursor que recorra toda la Tabla con una consulta SQL

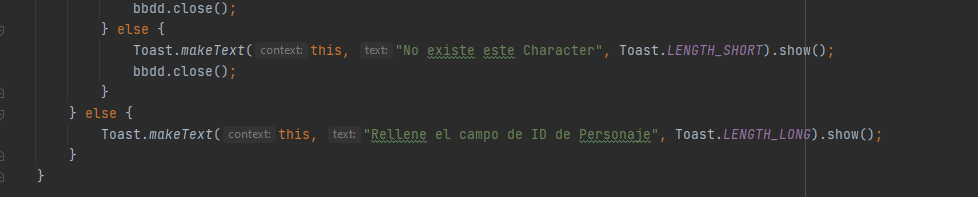
Cursor row = bbdd.rawQuery(  
 "SELECT \* FROM Character WHERE id\_personaje = " + codigo, null  
);

Creamos una condición para que se ponga en la primera posición de la tabla

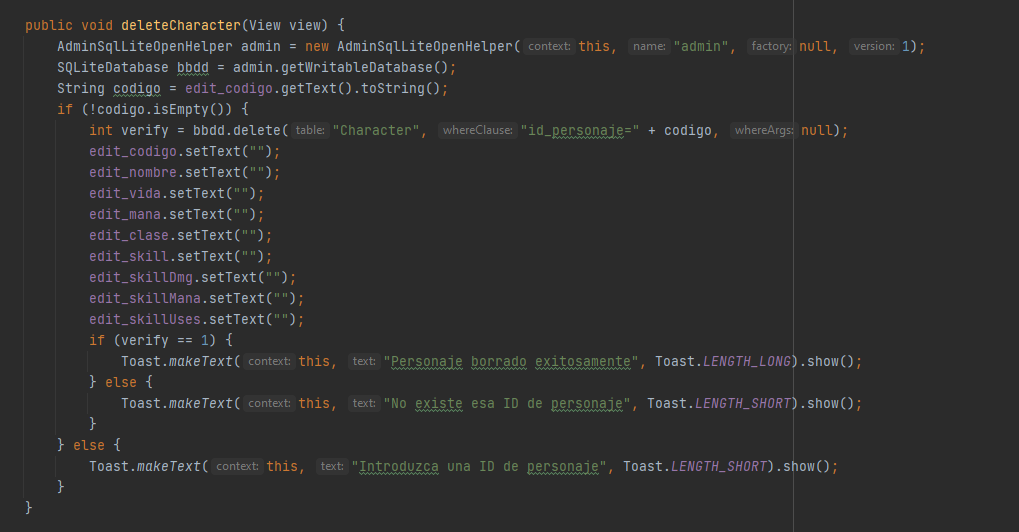
Recogemos el valor del cursor mediante posición y se lo asignamos a cada EditText

Cerramos la conexión

Controlador de errores:



1. deleteCharacter

 Iniciamos la conexión a la BBDD

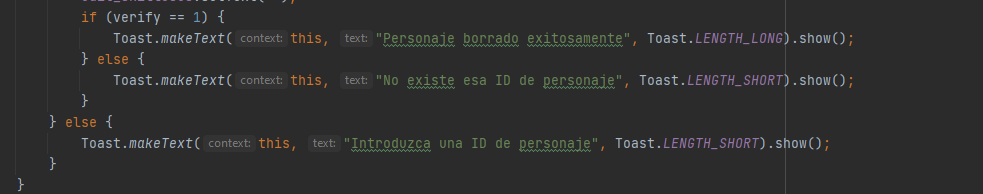
Recogemos el código de la EditText

Comprobamos si el código no esta vacio.

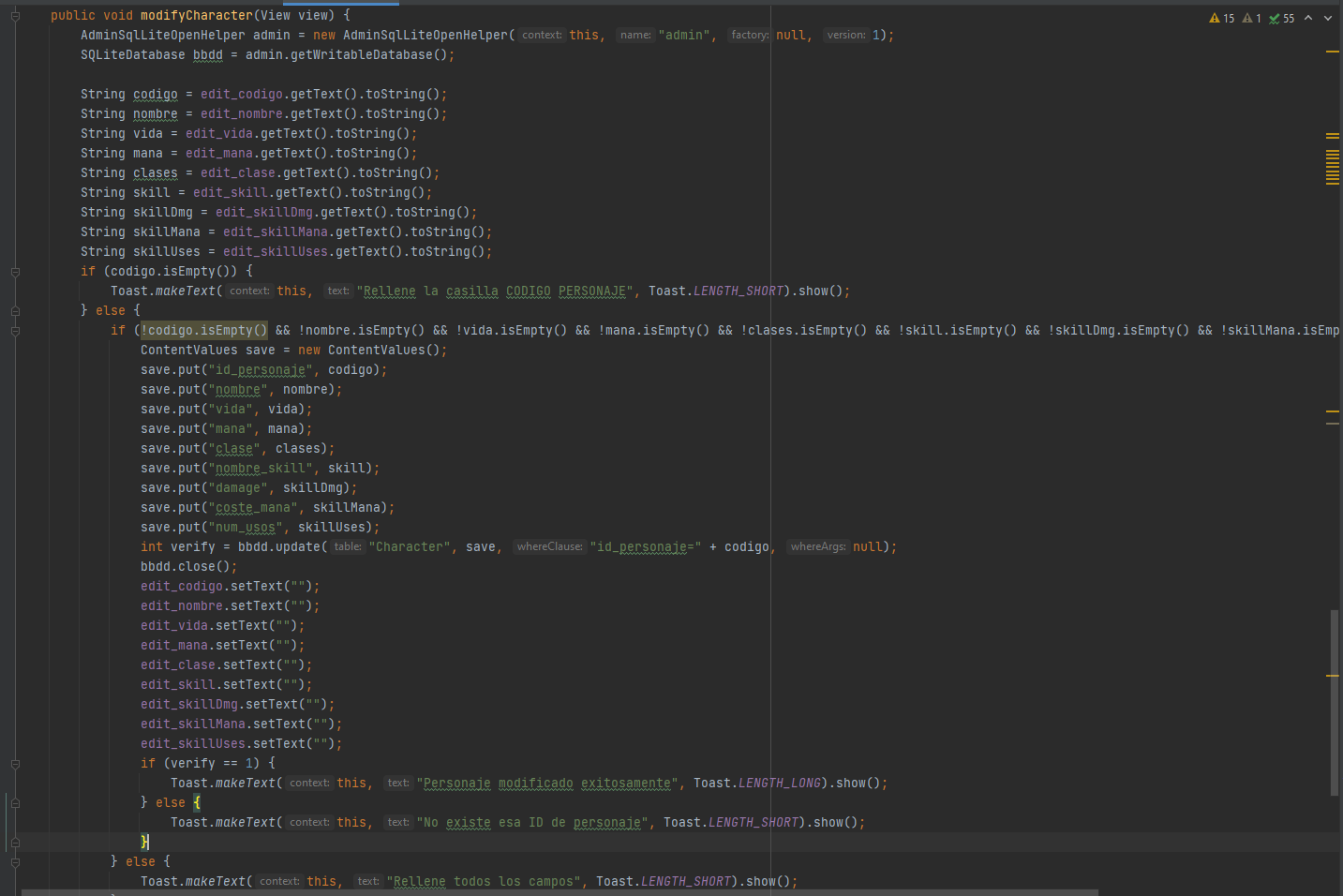
Usamos el método delete que nos facilita SQLite, le asociamos la tabla, el whereClause que en este caso queremos que borre según su ID por lo tanto id\_personaje

Ponemos todos los EditText en nulo para facilitar el uso al usuario

Control errores:



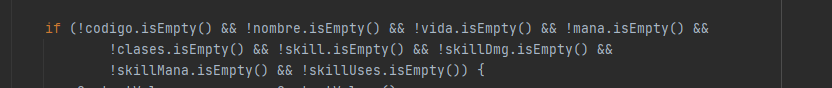
1. modifyCharacter



Lo primero abrimos la conexión a nuestra base de datos.

Recogemos los valores de cada EditText.

Creamos una condición de verificación para comprobar si todos los EditText tienen contenido



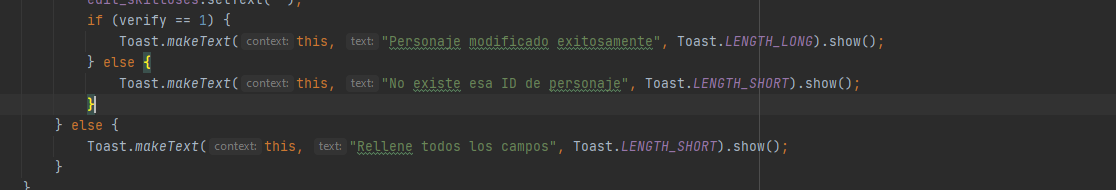
Creamos un objeto ContentValue para recoger y enviar valores a la BBDD

Asignamos a los nombres de la table de la BBDD sus respectivos EditText

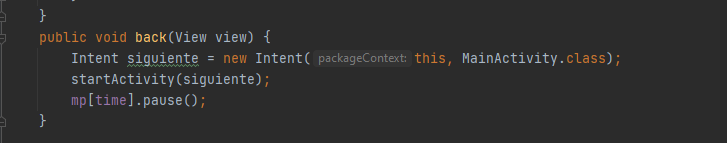
Usamos el método UPDATE que nos facilita SQLite, le asociamos la tabla, el contentvalues a insertar ,y la whereClause, que en este caso es la id\_personaje a modificar

Y después de tenerlos todos metidos en sus PUT lo Insertamos y cerramos la conexión

Control de errores:

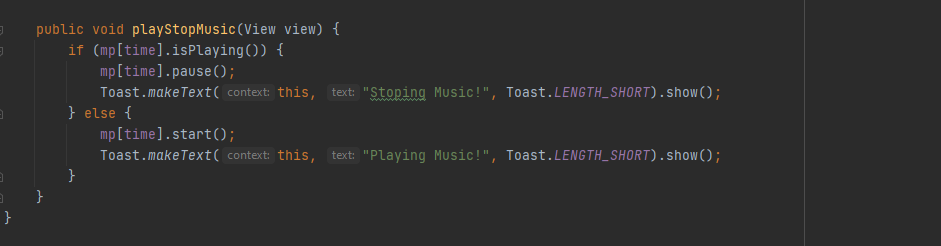


1. back



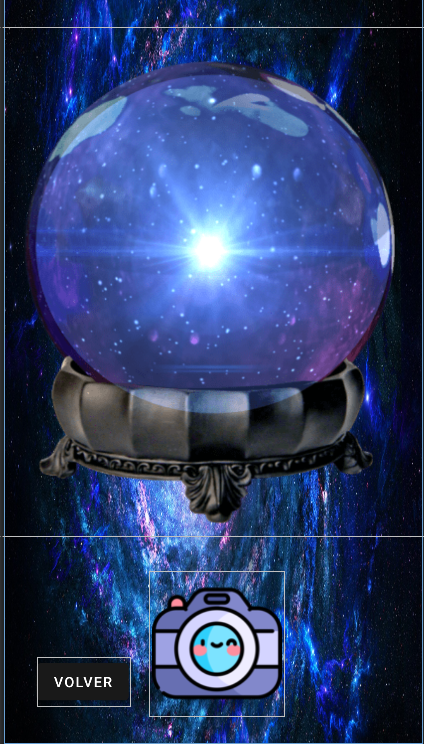
Creamos un objeto intent para poder navegar al MainActivity

1. playStopMusic



Nos permite activar y desactivar la música de fondo

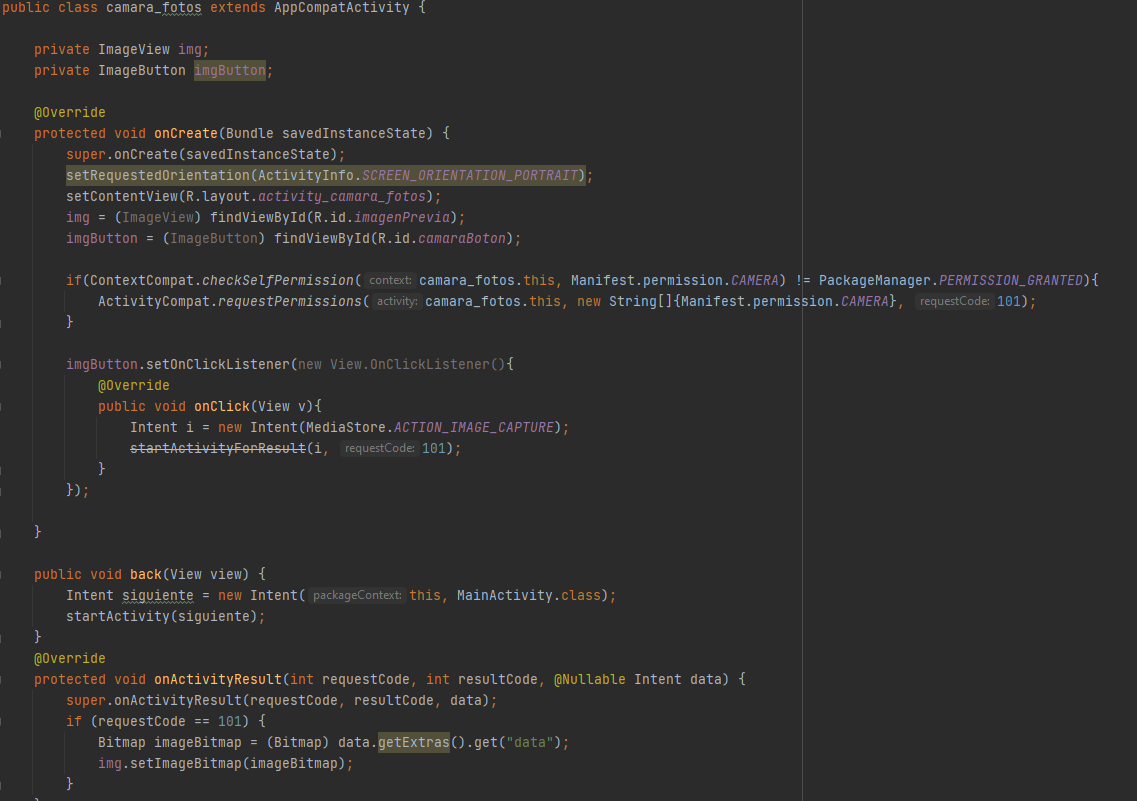
1. **Ventana Cámara de fotos**

****

Esta clase contiene un ImageView, un ImageBotton y un botón para volver.

El ImageView va a mostrar la imagen que, pulsando el ImageButton vamos a realizar.

A esta clase se le debe de dar ciertos permisos para poder realizar la acción.



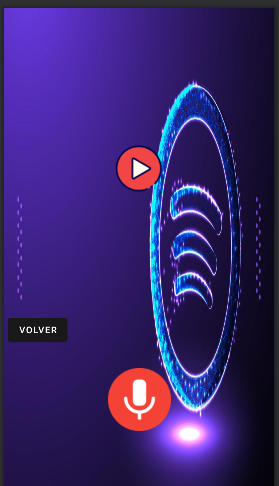
1. **Ventana Cámara de video**

****

Esta ventana cuenta con un ImageButton de una cámara de video centrada en medio de la pantalla, un TextView el cual te da la indicación de como funciona la pantalla, que simplemente es pulsar la imagen de la cámara de videos y un botón para volver atrás.

Para que tu APP permita grabar videos debemos de darle ciertos permisos. ****

1. **Ventana Grabadora**

****

Consta de dos ImageButtons, el microfono para grabar, y el reanudar para escuchar la grabación.

Para que tu aplicación permita grabar hay que darle permisos específicos.

