```
package com.internshala.echo.databases
import android.content.ContentValues
import android.content.Context
import android.database.sqlite.SQLiteDatabase
import android.database.sqlite.SQLiteOpenHelper
/**
 * Created by Harsh Deep Singh on 2/20/2018.
 */

/*This class is created for managing the database for our application
* In Android we use SQLite database for storing the data
* We create this table for keeping the data which is used even after the app is closed
* Now you may think how it is different from SharedPreferences?
* Shared preferences can store very small amount of data only whereas SQLite has the ability
to store huge amounts of data
* Therefore, in our application we will be using the SQLite database to store the favorite
tracks*/
class EchoDatabase : SQLiteOpenHelper {

    /*Let's define some params for our database
    * All the below params are case-sensitive and should be used with the same case*/
    /*Database Name*/
    val DB_NAME = "FavoriteDatabase"

    /*Name of the table*/
    val TABLE_NAME = "FavoriteTable"

    /*Name of column 1*/
    val COLUMN_ID = "SongID"

    /*Name of column 2*/
    val COLUMN_SONG_TITLE = "SongTitle"

    /*Name of column 3*/
    val COLUMN_SONG_ARTIST = "SongArtist"

    /*Name of column 4*/
    val COLUMN_SONG_PATH = "SongPath"

    /*This function is called when the application first creates the database
    * If already a table with the same name is present in the database, then this method is
skipped*/
    override fun onCreate(db: SQLiteDatabase?) {

        /*In-order to perform any function in our database we use SQL queries
        * These queries are pre-defined statements similar to english statements which
perform any action in the database*/
        /*The query here is used to create the table
        * The query is :
        * "CREATE TABLE FavoriteTable (
        *  SongsID INTEGER,
        *  SongArtist STRING,
        *  SongTitle STRING,
        *  SongPath STRING);"  */
        db?.execSQL("CREATE TABLE " + TABLE_NAME + "( " + COLUMN_ID +
                " INTEGER," + COLUMN_SONG_ARTIST + " STRING," + COLUMN_SONG_TITLE + "
STRING,"
                + COLUMN_SONG_PATH + " STRING);")
    }

    override fun onUpgrade(db: SQLiteDatabase?, oldVersion: Int, newVersion: Int) {
```

```
        TODO("not implemented") //To change body of created functions use File | Settings |
File Templates.
    }

    constructor(context: Context?, name: String?, factory: SQLiteDatabase.CursorFactory?,
version: Int) : super(context, name, factory, version) {}

    /*As the name suggests, this function is used to store the songs as favorites*/
    fun storeAsFavorite(id: Int?, artist: String?, songTitle: String?, path: String?) {

        /*The function writableDatabase is used to open the db for editing so that changes
can be made to the database*/
        val db = this.writableDatabase

        /*Here we define the Content Values which store the values which are pushed into
the database*/
        val contentValues = ContentValues()

        /*The .put() function is used for adding the values to the content values object*/
        contentValues.put(COLUMN_ID, id)
        contentValues.put(COLUMN_SONG_ARTIST, artist)
        contentValues.put(COLUMN_SONG_TITLE, songTitle)
        contentValues.put(COLUMN_SONG_PATH, path)

        /*Here we use the insert function to insert the values into the table whose name is
specified using the TABLE_NAME
        * and the values which are added are the content values*/
        db.insert(TABLE_NAME, null, contentValues)

        /*After performing the db actions, we are required to close the database in-order
to prevent any data leakage and saving the resources*/
        db.close()
    }
}
```