Game Project - Database Design & Prototype

Revision 03

# Introduction

In this document, we will develop a prototype for the Hi Scores database for the game. This prototype has no UI, but the output can be seen in the logcat for the running app (in info level debug mode)

The database name is called “project3game”

The table is called “Hiscores”

The create statement suggested is:

**CREATE** **TABLE** hiscores (

score\_id INTEGER PRIMARY **KEY**,

game\_date TEXT **NOT** NULL,

player\_name TEXT **NOT** NULL,

score INTEGER **NOT** NULL

);

# The HiScore Class

**package** edu.icraig.finalgame\_db\_proto1; // replace package with your package name  
  
**public class** HiScore {  
 */\*  
 \* CREATE TABLE hiscores (  
 \* score\_id INTEGER PRIMARY KEY,  
 \* game\_date TEXT NOT NULL,  
 \* player\_name TEXT NOT NULL,  
 \* score INTEGER NOT NULL);  
 \*/* **int score\_id**; *// primary score* String **game\_date**; *// date in 02 DEC 2020* String **player\_name**; *// JOE* **int score**; *// score - should be even  
  
 /\*  
 \* Constructors  
 \*/* **public** HiScore(**int** score\_id, String game\_date, String player\_name, **int** score) {  
 **this**.**score\_id** = score\_id;  
 **this**.**game\_date** = game\_date;  
 **this**.**player\_name** = player\_name;  
 **this**.**score** = score;  
 }  
  
 **public** HiScore() {  
 }  
  
 */\*  
 \* Getter and setter methods  
 \*/* **public** HiScore(String game\_date, String player\_name, **int** score) {  
 **this**.**game\_date** = game\_date;  
 **this**.**player\_name** = player\_name;  
 **this**.**score** = score;  
 }  
  
 **public int** getScore\_id() {  
 **return score\_id**;  
 }  
  
 **public void** setScore\_id(**int** score\_id) {  
 **this**.**score\_id** = score\_id;  
 }  
  
 **public** String getGame\_date() {  
 **return game\_date**;  
 }  
  
 **public void** setGame\_date(String game\_date) {  
 **this**.**game\_date** = game\_date;  
 }  
  
 **public** String getPlayer\_name() {  
 **return player\_name**;  
 }  
  
 **public void** setPlayer\_name(String player\_name) {  
 **this**.**player\_name** = player\_name;  
 }  
  
 **public int** getScore() {  
 **return score**;  
 }  
  
 **public void** setScore(**int** score) {  
 **this**.**score** = score;  
 }  
}

# The DatabaseHandler class

This class implements the create/delete of the table(s) and the CRUD methods for the application.

**package** edu.icraig.finalgame\_db\_proto1; *// replace package with your own package name***import** android.content.ContentValues;  
**import** android.content.Context;  
**import** android.database.Cursor;  
**import** android.database.sqlite.SQLiteDatabase;  
**import** android.database.sqlite.SQLiteOpenHelper;  
  
**import** java.util.ArrayList;  
**import** java.util.List;  
  
*/\*  
 \* Class has create / drop table and CRUD ops for table  
 \*/***public class** DatabaseHandler **extends** SQLiteOpenHelper {  
 **private static final int *DATABASE\_VERSION*** = 1;  
 **private static final** String ***DATABASE\_NAME*** = **"project3game"**;  
 **private static final** String ***TABLE\_HI\_SCORES*** = **"hi\_scores"**;  
 **private static final** String ***KEY\_SCORE\_ID*** = **"score\_id"**;  
 **private static final** String ***KEY\_PLAYER\_NAME*** = **"player\_name"**;  
 **private static final** String ***KEY\_GAME\_DATE*** = **"game\_date"**;  
 **private static final** String ***KEY\_SCORE*** = **"score"**;  
  
  
 */\*  
 \* Constructor  
 \*/* **public** DatabaseHandler(Context context) {  
 **super**(context, ***DATABASE\_NAME***, **null**, ***DATABASE\_VERSION***);  
 *//3rd argument to be passed is CursorFactory instance* }  
  
 *// Creating Tables* @Override  
 **public void** onCreate(SQLiteDatabase db) {  
 String CREATE\_HISCORES\_TABLE = **"CREATE TABLE "** + ***TABLE\_HI\_SCORES*** + **"("** +  
 ***KEY\_SCORE\_ID*** + **" INTEGER PRIMARY KEY,"** +  
 ***KEY\_GAME\_DATE*** + **" TEXT NOT NULL,"** +  
 ***KEY\_PLAYER\_NAME*** + **" TEXT NOT NULL,"** +  
 ***KEY\_SCORE*** + **" INTEGER NOT NULL"** +  
 **")"**;  
 db.execSQL(CREATE\_HISCORES\_TABLE);  
 }  
  
 *// Upgrading database* @Override  
 **public void** onUpgrade(SQLiteDatabase db, **int** oldVersion, **int** newVersion) {  
 *// Drop older table if existed* db.execSQL(**"DROP TABLE IF EXISTS "** + ***TABLE\_HI\_SCORES***);  
  
 *// Create tables again* onCreate(db);  
 }  
  
 */\*  
 \* CRUD Helper methods  
 \*/* **public void** emptyHiScores() {  
 *// Drop older table if existed* SQLiteDatabase db = **this**.getWritableDatabase();  
 db.execSQL(**"DROP TABLE IF EXISTS "** + ***TABLE\_HI\_SCORES***);  
  
 *// Create tables again* onCreate(db);  
 }  
  
 *// code to add the new hiScore* **void** addHiScore(HiScore hiScore) {  
 SQLiteDatabase db = **this**.getWritableDatabase();  
  
 ContentValues values = **new** ContentValues();  
 values.put(***KEY\_GAME\_DATE***, hiScore.getGame\_date());  
 values.put(***KEY\_PLAYER\_NAME***, hiScore.getPlayer\_name());  
 values.put(***KEY\_SCORE***, hiScore.getScore());  
  
 *// Inserting Row* db.insert(***TABLE\_HI\_SCORES***, **null**, values);  
 *//2nd argument is String containing nullColumnHack* db.close(); *// Closing database connection* }  
  
 *// code to get the single hiScore* HiScore getHiScore(**int** id) {  
 SQLiteDatabase db = **this**.getReadableDatabase();  
  
 Cursor cursor = db.query(***TABLE\_HI\_SCORES***, **new** String[]{  
 ***KEY\_SCORE\_ID***,  
 ***KEY\_GAME\_DATE***,  
 ***KEY\_PLAYER\_NAME***,  
 ***KEY\_SCORE***},  
 ***KEY\_SCORE\_ID*** + **"=?"**,  
 **new** String[]{String.*valueOf*(id)}, **null**, **null**, **null**, **null**);  
 **if** (cursor != **null**)  
 cursor.moveToFirst();  
  
 HiScore hiScore = **new** HiScore(Integer.*parseInt*(  
 cursor.getString(0)),  
 cursor.getString(1),  
 cursor.getString(2),  
 cursor.getInt(3));  
 *// return hi score* **return** hiScore;  
 }  
  
 *// code to get all hiScores in a list view* **public** List<HiScore> getAllHiScores() {  
 List<HiScore> hiScoreList = **new** ArrayList<HiScore>();  
 *// Select All Query* String selectQuery = **"SELECT \* FROM "** + ***TABLE\_HI\_SCORES***;  
  
 SQLiteDatabase db = **this**.getWritableDatabase();  
 Cursor cursor = db.rawQuery(selectQuery, **null**);  
  
 *// looping through all rows and adding to list* **if** (cursor.moveToFirst()) {  
 **do** {  
 HiScore hiScore = **new** HiScore();  
 hiScore.setScore\_id(Integer.*parseInt*(cursor.getString(0)));  
 hiScore.setGame\_date(cursor.getString(1));  
 hiScore.setPlayer\_name(cursor.getString(2));  
 hiScore.setScore(cursor.getInt(3));  
 *// Adding hi score to list* hiScoreList.add(hiScore);  
 } **while** (cursor.moveToNext());  
 }  
  
 *// return hiScore list* **return** hiScoreList;  
 }  
  
 *// code to update the single hiScore* **public int** updateHiScore(HiScore hiScore) {  
 SQLiteDatabase db = **this**.getWritableDatabase();  
  
 ContentValues values = **new** ContentValues();  
 values.put(***KEY\_PLAYER\_NAME***, hiScore.getPlayer\_name());  
 values.put(***KEY\_GAME\_DATE***, hiScore.getGame\_date());  
 values.put(***KEY\_SCORE***, hiScore.getScore());  
  
 *// updating row* **return** db.update(***TABLE\_HI\_SCORES***, values, ***KEY\_SCORE\_ID*** + **" = ?"**,  
 **new** String[]{String.*valueOf*(hiScore.getScore\_id())});  
 }  
  
 *// Deleting single hiScore* **public void** deleteHiScore(HiScore hiScore) {  
 SQLiteDatabase db = **this**.getWritableDatabase();  
 db.delete(***TABLE\_HI\_SCORES***, ***KEY\_SCORE\_ID*** + **" = ?"**,  
 **new** String[]{String.*valueOf*(hiScore.getScore\_id())});  
 db.close();  
 }  
  
 *// Getting top 5 scores* **public** List<HiScore> getTopFiveScores() {  
 List<HiScore> hiScoreList = **new** ArrayList<HiScore>();  
 *// Select All Query* String selectQuery = **"SELECT \* FROM "** + ***TABLE\_HI\_SCORES*** +  
 **" ORDER BY SCORE DESC "** +  
 **" LIMIT 5"**;  
  
 SQLiteDatabase db = **this**.getWritableDatabase();  
 Cursor cursor = db.rawQuery(selectQuery, **null**);  
  
 *// looping through all rows and adding to list* **if** (cursor.moveToFirst()) {  
 **do** {  
 HiScore hiScore = **new** HiScore();  
 hiScore.setScore\_id(Integer.*parseInt*(cursor.getString(0)));  
 hiScore.setGame\_date(cursor.getString(1));  
 hiScore.setPlayer\_name(cursor.getString(2));  
 hiScore.setScore(cursor.getInt(3));  
 *// Adding hi score to list* hiScoreList.add(hiScore);  
 } **while** (cursor.moveToNext());  
 }  
  
 *// return hi score list* **return** hiScoreList;  
 }  
  
}

# MainActivity Class

Here we put our test code to exercise our database

**package** edu.icraig.finalgame\_db\_proto1; *// replace package with your own package name*  
  
**import** androidx.appcompat.app.AppCompatActivity;  
  
**import** android.os.Bundle;  
**import** android.util.Log;  
  
**import** java.util.List;  
  
**public class** MainActivity **extends** AppCompatActivity {  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
  
 DatabaseHandler db = **new** DatabaseHandler(**this**);  
  
 db.emptyHiScores(); *// empty table if required  
  
 // Inserting hi scores* Log.*i*(**"Insert: "**, **"Inserting .."**);  
 db.addHiScore(**new** HiScore(**"20 OCT 2020"**, **"Frodo"**, 12));  
 db.addHiScore(**new** HiScore(**"28 OCT 2020"**, **"Dobby"**, 16));  
 db.addHiScore(**new** HiScore(**"20 NOV 2020"**, **"DarthV"**, 20));  
 db.addHiScore(**new** HiScore(**"20 NOV 2020"**, **"Bob"**, 18));  
 db.addHiScore(**new** HiScore(**"22 NOV 2020"**, **"Gemma"**, 22));  
 db.addHiScore(**new** HiScore(**"30 NOV 2020"**, **"Joe"**, 30));  
 db.addHiScore(**new** HiScore(**"01 DEC 2020"**, **"DarthV"**, 22));  
 db.addHiScore(**new** HiScore(**"02 DEC 2020"**, **"Gandalf"**, 132));  
  
  
 *// Reading all scores* Log.*i*(**"Reading: "**, **"Reading all scores.."**);  
 List<HiScore> hiScores = db.getAllHiScores();  
  
  
 **for** (HiScore hs : hiScores) {  
 String log =  
 **"Id: "** + hs.getScore\_id() +  
 **", Date: "** + hs.getGame\_date() +  
 **" , Player: "** + hs.getPlayer\_name() +  
 **" , Score: "** + hs.getScore();  
  
 *// Writing HiScore to log* Log.*i*(**"Score: "**, log);  
 }  
  
 Log.*i*(**"divider"**, **"===================="**);  
  
 HiScore singleScore = db.getHiScore(5);  
 Log.*i*(**"High Score 5 is by "**, singleScore.getPlayer\_name() + **" with a score of "** +  
 singleScore.getScore());  
  
 Log.*i*(**"divider"**, **"===================="**);  
  
 *// Calling SQL statement* List<HiScore> top5HiScores = db.getTopFiveScores();  
  
 **for** (HiScore hs : top5HiScores) {  
 String log =  
 **"Id: "** + hs.getScore\_id() +  
 **", Date: "** + hs.getGame\_date() +  
 **" , Player: "** + hs.getPlayer\_name() +  
 **" , Score: "** + hs.getScore();  
  
 *// Writing HiScore to log* Log.*i*(**"Score: "**, log);  
 }  
 Log.*i*(**"divider"**, **"===================="**);  
  
 HiScore hiScore = top5HiScores.get(top5HiScores.size() - 1);  
 *// hiScore contains the 5th highest score* Log.*i*(**"fifth Highest score: "**, String.*valueOf*(hiScore.getScore()) );  
  
 *// simple test to add a hi score* **int** myCurrentScore = 40;  
 *// if 5th highest score < myCurrentScore, then insert new score* **if** (hiScore.getScore() < myCurrentScore) {  
 db.addHiScore(**new** HiScore(**"08 DEC 2020"**, **"Elrond"**, 40));  
 }  
  
 Log.*i*(**"divider"**, **"===================="**);  
  
 *// Calling SQL statement* top5HiScores = db.getTopFiveScores();  
  
 **for** (HiScore hs : top5HiScores) {  
 String log =  
 **"Id: "** + hs.getScore\_id() +  
 **", Date: "** + hs.getGame\_date() +  
 **" , Player: "** + hs.getPlayer\_name() +  
 **" , Score: "** + hs.getScore();  
  
 *// Writing HiScore to log* Log.*i*(**"Score: "**, log);  
 }  
  
 }  
}