COP4656.01 **Brandon Thompson**
Prof. Topsakal Due Date: 4/5/2020

Individual Assignment 2 Report

In this assignment we had to modify the previous project (AboutMe) to retrieve ListView data from an SQLite database instead of a string array, and pass user data with a SharedPreferences file instead of Intent.putExtra().

Table structure is as follows:

```
CREATE TABLE SONGS(_id INTEGER PRIMARY KEY AUTOINCREMENT, NAME TEXT);
CREATE TABLE COURSES(_id INTERGER PRIMARY KEY AUTOINCREMENT, NAME TEXT);
```

The tables SONG and COURSE have columns id and NAME.

We store the list of songs in the SONGS table and the list of courses in the COURSES table. Later we use the rows to fill in the ListView in the appropriate activities.

To retrieve data from the COURSE and SONG tables I implemented a getAllCourses() and getAllSongs() methods in the DatabaseHelper file (code below), this returns a list of the Course or Song class that holds the id and NAME values of the items. Then I created an ArrayList<String> from the list of Courses and Songs that works well with the ArrayAdapter class to populate the ListView

To pass the user data between Activities, I initialized a SharedPreferences object in all the Activities that need to send/receive data (MainActivity, SongActivity, CourseActivity), then depending on what CheckBoxes are clicked, and what RadioButton is selected, it will store the correct combination of name, favorite song, and favorite course with the SharedPreferencesEditor object.

In the receiving activities, you only need a SharedPrefernces object and use the getString(<key>) method to retrieve the stored data. This method call replaces the Intent.putExtra(<key>) method and because they both should retrieve Strings, the code worked without changing anything else.

```
public class DatabaseHelper extends SQLiteOpenHelper {
    private static DatabaseHelper sInstance;
    public static final int DATABASE_VERSION = 1;
    public static final String DATABASE_NAME = "data_db";
    public static final String TABLE_NAME1 = "course";
    public static final String TABLE_NAME2 = "song";
    public static final String COLUMN_ID = "id";
public static final String COLUMN_NAME = "name";
public static final String CREATE_TABLE1 =
    "CREATE TABLE " + TABLE_NAME1 + "("
    + COLUMN_ID + " INTEGER PRIMARY KEY AUTOINCREMENT,"
    + COLUMN_NAME + " TEXT"
    + ")";
public static final String CREATE_TABLE2 =
    "CREATE TABLE " + TABLE_NAME2 + "("
    + COLUMN_ID + " INTEGER PRIMARY KEY AUTOINCREMENT,"
    + COLUMN_NAME + " TEXT"
    + ")";

public static synchronized DatabaseHelper
getInstance(Context context) {
    if (sInstance == null) {
        sInstance = new DatabaseHelper(
            context.getApplicationContext());
    }
    return sInstance;
}

private DatabaseHelper(Context context) {
    super(context, DATABASE_NAME, null, DATABASE_VERSION);
}

@Override
public void onCreate(SQLiteDatabase db) {
    db.execSQL(CREATE_TABLE1);
    db.execSQL(CREATE_TABLE2);
}

@Override
public void onUpgrade(
    SQLiteDatabase db, int oldVersion, int newVersion) {
    db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME1);
    db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME2);
    onCreate(db);
}

public long insertCourse(String name) {
    SQLiteDatabase db = this.getWritableDatabase();
```

```
        ContentValues values = new ContentValues();
        values.put(COLUMN_NAME, name);
        long id = db.insert(TABLE_NAME1,null,values);
        db.close();
        return id;
    }

    public long insertSong(String name) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(COLUMN_NAME, name);
        long id = db.insert(TABLE_NAME2,null,values);
        db.close();
        return id;
    }

    public List<Course> getAllCourses() {
        List<Course> courses = new ArrayList<>();
        String selectQuery =
            "SELECT * FROM " + TABLE_NAME1 + " ORDER BY " +
                COLUMN_NAME;
        SQLiteDatabase db = this.getWritableDatabase();
        Cursor cursor = db.rawQuery(selectQuery, null);
        if (cursor.moveToFirst()) {
            do {
                Course course = new Course();
                course.setId(cursor.getInt(
                    cursor.getColumnIndex(COLUMN_ID)));
                course.setName(cursor.getString(
                    cursor.getColumnIndex(COLUMN_NAME)));
                courses.add(course);
            } while (cursor.moveToNext());
        }
        db.close();
        return courses;
    }

    public List<Song> getAllSongs() {
        List<Song> songs = new ArrayList<>();
//        String where = null;
        String selectQuery =
            "SELECT * FROM " + TABLE_NAME2 + " ORDER BY " +
                COLUMN_NAME;
        SQLiteDatabase db = this.getWritableDatabase();
        Cursor cursor = db.rawQuery(selectQuery, null);
        if (cursor.moveToFirst()) {
            do {
                Song song = new Song();
                song.setId(cursor.getInt(
                    cursor.getColumnIndex(COLUMN_ID)));
                song.setName(cursor.getString(
                    cursor.getColumnIndex(COLUMN_NAME)));
                songs.add(song);
            } while (cursor.moveToNext());
        }
        db.close();
//        return cursor;
//        Testing for returning Cursor instead of List
        return songs;
    }
}
```