**实 验 报 告**

**课程名称 移动应用系统**

**实验项目 移动平台数据持久化**

**实验仪器 计算机一台、Android手机一部**

**系 别 计算机学院**

**专 业 计算机科学与技术**

**班级/学号**  计科1606 / 2016010311

**学生姓名**  耿瑞

**实验日期**  2018年12月14日

**成 绩** \_\_\_\_\_\_\_\_\_\_

**指导教师** \_\_\_\_ **郝保水**\_\_\_\_ \_\_

1 实验目的

（1）了解Android数据存储的基本概念；

（2）理解并掌握SQLite数据库的使用方法;

（3）进一步熟悉与掌握Fragment使用；

（4）进一步熟悉各组件、适配器、菜单、对话框等相关概念和技术等；

（5）掌握ContentProvider相关概念和技术；

（6）掌握Handler相关概念和技术；

2 实验要求

（1）课前预习实验内容，并查找相关资料。

（2）按照实验步骤完成各个相关内容。

（3）撰写实验报告。

A） 实验报告格式必须符合学校要求（例如必须采用学校规定的实验封面）；

B） 写出实验详细步骤，包括主要采用的技术方案、相关分析和核心代码。注意：不要简单地近包括截屏和代码，完整代码可以作为附录放在实验报告结尾；

C） 总结实验中遇到的问题、分析和解决方法。

D） 写出心得体会与收获等。

3 实验步骤

主要分为几个大步骤进行实现。

（1）需求分析，调研单词类应用具有哪些功能，确定界面。

（2）系统设计，对系统进行模块划分，确定技术方案。

（3）编码，编写代码实现各项功能。

（4）测试。

3.1 需求分析

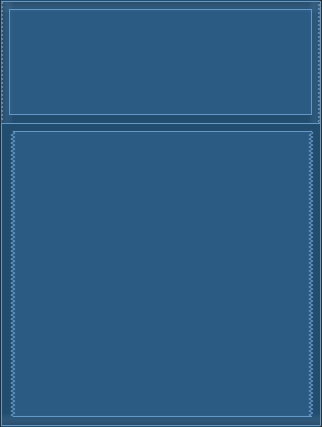
（1）主要功能

* 支持英汉词典的查询
* 支持单词本的添加与删除
* 支持中英互译

（2）界面设计



主界面



单词详情界面

3.2 系统设计

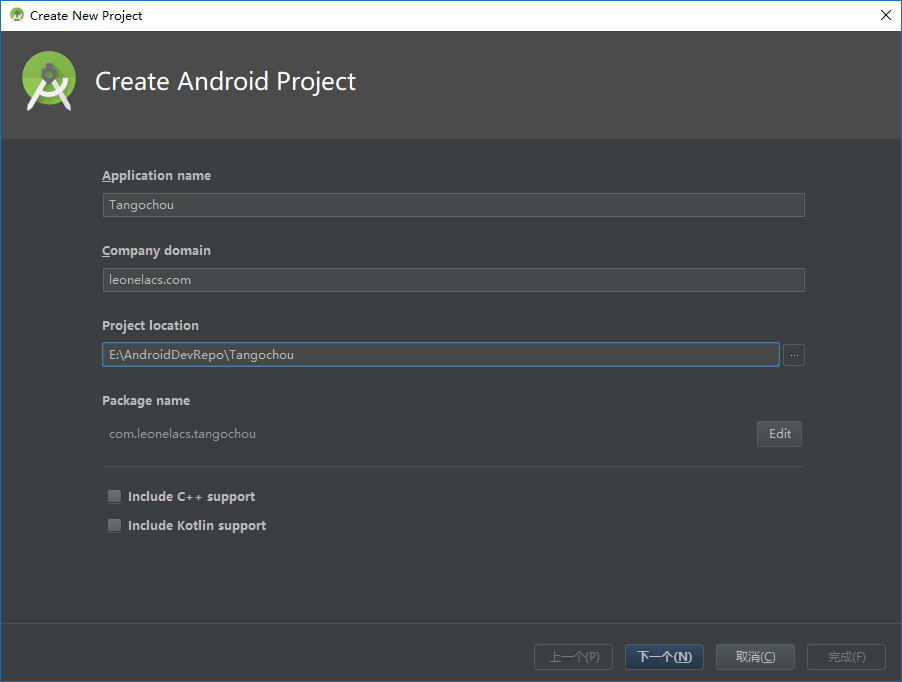
计算器应用包括以下模块：

* 词典模块
* 单词本模块
* 单词详情模块
* 翻译模块
* 词典数据库模块
* 单词本数据库模块

3.3 项目实施

主要步骤：

（1）创建工程



在Android Studio中创建Android项目。

应用名称：Tangochou

域名：leonelacs.com

包名：com.leonelacs.tangochou

最低SDK版本API 24: Android 7.0

（2）编写各界面布局、菜单等资源文件

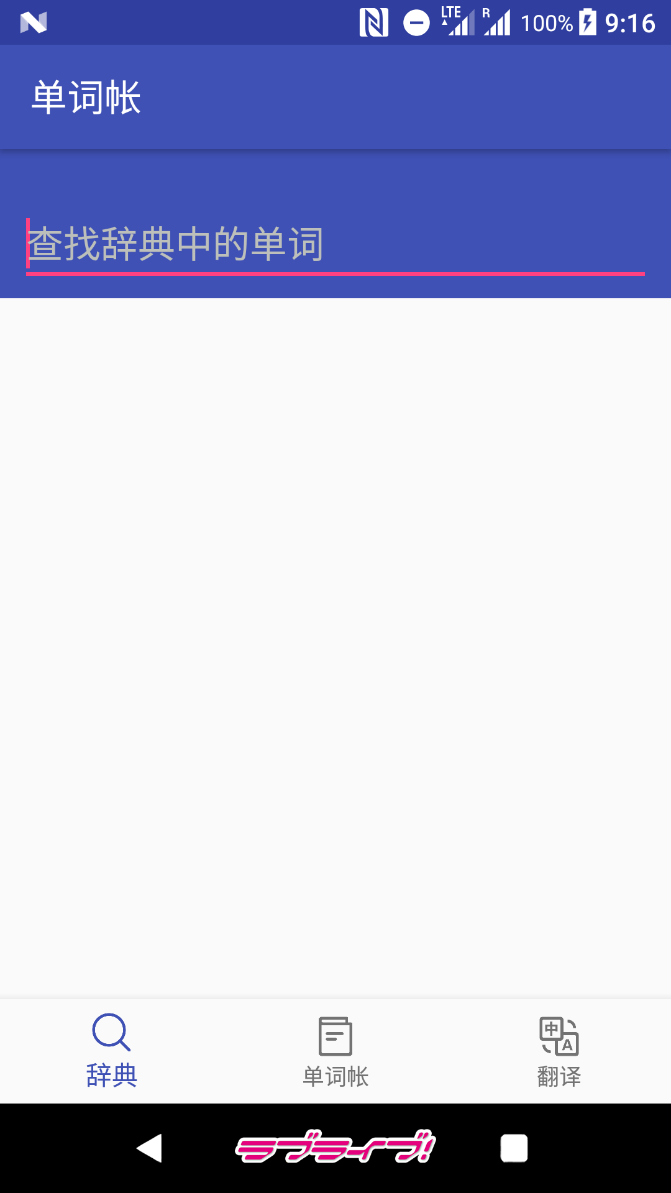
单词帐词典界面，使用LinearLayout实现，包含：

* 一个用于输入表达式的EditText：**SearchInput**；
* 一个用于显示查找结果的RecyclerView：**SearchResult**；
* 底部导航栏BottomNavigationView **navigation**

SearchInput响应软键盘回车事件，当按下软键盘的回车键时，对数据库进行查询，将查询到的单词列表经由TangoAdapter于SearchResult中显示

SearchResult响应Item的点击事件，当点击其中的条目时，打开该条目单词的单词详情界面

Navigation响应点击事件，当点击底部导航栏按钮时显示所选页面

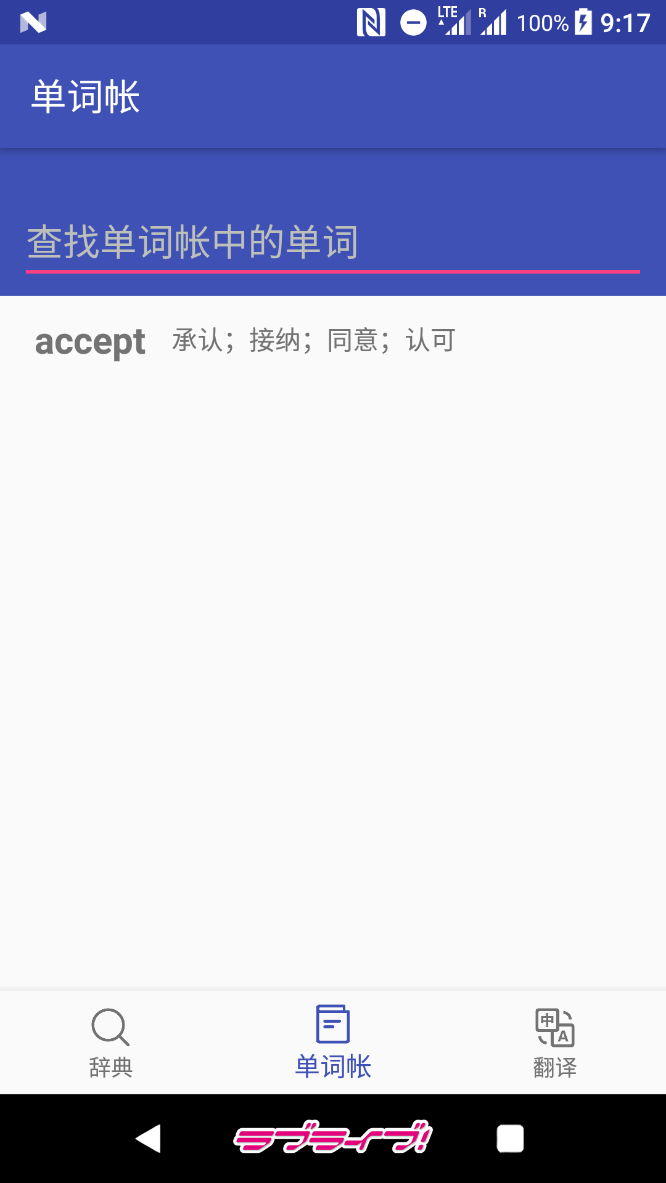


单词帐单词本界面，使用LinearLayout实现，包含：

* 一个用于输入表达式的EditText：**ChouInput**；
* 一个用于显示查找结果的RecyclerView：**ChouResult**；

ChouInput响应软键盘回车事件，当按下软键盘的回车键时，对数据库进行查询，将查询到的单词列表经由TangoAdapter于ChouResult中显示

ChouResult响应Item的点击事件，当点击其中的条目时，打开该条目单词的单词详情界面



单词详情界面，使用LinearLayout实现，包含：

* 一个用于显示单词的TextView：**ResultWord**；
* 一个用于显示单词释义的TextView：**ResultDefi**；
* 一个用于添加或删除单词本条目的FloatingActionButton：**fab**

fab响应点击事件，当点击时，根据该单词当前是否在单词本中，进行单词本数据库的删除或插入操作，并改变控件图片的颜色



词典数据库：

应用安装时，会将asset目录中的ms\_bing\_dict.db单词数据库文件复制到/data/data/com.leonelacs.tangochou/databases/目录中；

数据库的有关操作由DictOperate类实现，在Activity中通过实例化该类，使用其openDB()方法获得可以操作的SQLiteDatabase对象。

openDB()方法，包含复制数据库文件操作的实现代码如下所示：

public SQLiteDatabase openDB() {  
 File dir = new File(DB\_PATH);  
 if (!dir.exists()) {  
 dir.mkdir();  
 }  
  
 String srcDBName = DB\_PATH + SRC\_DB\_FILENAME;  
  
 try {  
 InputStream inputStream = context.getResources().getAssets().open("ms\_bing\_dict.db");  
 FileOutputStream fos = new FileOutputStream(srcDBName);  
 byte[] buf = new byte[1024 \* 8];  
 int len = 0;  
 while ((len = inputStream.read(buf)) != -1) {  
 fos.write(buf, 0, len);  
 }  
 fos.close();  
 inputStream.close();  
 SQLiteDatabase database = SQLiteDatabase.*openOrCreateDatabase*(srcDBName, null);  
 return database;  
 }  
 catch (IOException e) {  
 e.printStackTrace();  
 return null;  
 }  
}

单词表数据库：

数据库的有关操作由DBOHelper类实现，该类继承自SQLiteOpenHelper。在Activity中通过实例化该类，使用其getWritableDatabase()方法获得可以操作的SQLiteDatabase对象。

3.4 测试

（1）词典

在词典页面，单击输入框弹出软键盘，可以输入单词。

在按下回车键（完成）时触发查询数据库动作，并将查询结果于中间的列表显示。

例：输入accept，列表中显示所有以accept为前缀的单词，按照词频降序排序（accept本身除外，其排在列表首位）



（2）单词本

跳转到单词本页面，自动查询单词本内保存的所有单词，并将查询结果于中间的列表显示。

单击输入框弹出软键盘，可以输入单词。

在按下回车键（完成）时触发查询数据库动作，并将查询结果于中间的列表显示

例：点击导航栏图标进入单词本页面，查看单词本保存的所有单词



例：输入capital，列表中显示单词本中保存的所有以capital为前缀的单词，按照字母顺序排序



（3）单词详情

从词典或单词表的查询结果列表中点击任意条目，跳转进入该条目单词的详情页面。

在单词详情页面可以查看该单词的详细释义。

点击右下角的按钮，可以将该单词加入单词本（或从单词本中移除）。

例：点击列表中的accept条目进入其详情页面



例：点击按钮将accept添加到单词本



例：点击按钮将accept从单词本中移除



4 实验中遇到的问题、分析和解决方法

（1）使用已有的SQLite数据库

在编写应用的过程中，从网上下载了一个必应词典的单词数据库，想要在自己的应用中使用。经过查询资料，最终通过文件流输入输出的方式，将预先放置于asset目录下的.db数据库文件复制到指定目录中供使用

（2）搜索另一单词时，RecyclerList中的内容需要滚动后才更新

在更新Adapter所绑定的List对象内容后，执行其notifyDataSetChanged()方法立即更新显示内容。

（3）在单词详情中从单词帐移除单词，返回单词本中仍有该单词

在SavedChouActivity的onStart方法中执行查询数据库操作更新Adapter所绑定的List对象，并执行notifyDataSetChanged()方法更新显示。

5 心得体会与收获

通过这次实验，我完成了一个英汉词典和单词本Android应用的开发。了解了Android数据存储的基本概念；理解并掌握了SQLite数据库的使用方法；进一步熟悉了Fragment使用，各组件、适配器、菜单、对话框等相关概念和技术等；掌握了ContentProvider相关概念和技术和Handler相关概念和技术。

在这次实验中，我选择使用比ListView更灵活的一个控件RecyclerView显示查询单词的结果，还通过设定软键盘的回车事件触发单词的查询操作，并通过查找资料实现了将已有SQLite数据库文件复制到设备指定目录供应用使用。

6 附录

代码仓库：<https://github.com/leonelacs/Tangochou>

程序完整代码：

**com.leonelacs.tangochou**

TangoItem.java

package com.leonelacs.tangochou;  
  
public class TangoItem {  
 String word;  
 String definition;  
  
 public String getWord() {  
 return word;  
 }  
  
 public void setWord(String word) {  
 this.word = word;  
 }  
  
 public String getDefinition() {  
 return definition;  
 }  
  
 public void setDefinition(String definition) {  
 this.definition = definition;  
 }  
}

TangoAdapter.java

package com.leonelacs.tangochou;  
  
import android.content.Intent;  
import android.support.v7.widget.RecyclerView;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.TextView;  
  
import java.util.List;  
  
public class TangoAdapter extends RecyclerView.Adapter<TangoAdapter.ViewHolder> {  
  
 private List<TangoItem> tangos;  
  
 static class ViewHolder extends RecyclerView.ViewHolder {  
 View searchView;  
 TextView tangoWord, tangoDefi;  
 public ViewHolder(View view) {  
 super(view);  
 searchView = view;  
 tangoWord = (TextView) view.findViewById(R.id.*TWord*);  
 tangoDefi = (TextView) view.findViewById(R.id.*TDefinition*);  
 }  
 }  
  
 public TangoAdapter(List<TangoItem> tangoList) {  
 tangos = tangoList;  
 }  
  
 @Override  
 public ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {  
 View view = LayoutInflater.*from*(parent.getContext()).inflate(R.layout.*tango\_vector*, parent, false);  
 final ViewHolder holder = new ViewHolder(view);  
  
 holder.searchView.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 int position = holder.getAdapterPosition();  
 TangoItem tangoBeClicked = tangos.get(position);  
 Intent intent = new Intent(view.getContext(), DetailActivity.class);  
 intent.putExtra("word", tangoBeClicked.getWord());  
 intent.putExtra("definition", tangoBeClicked.getDefinition());  
 view.getContext().startActivity(intent);  
 }  
 });  
  
 return holder;  
 }  
  
 @Override  
 public void onBindViewHolder(ViewHolder holder, int position) {  
 TangoItem tangoItem = tangos.get(position);  
 holder.tangoWord.setText(tangoItem.word);  
 holder.tangoDefi.setText(tangoItem.definition);  
 }  
  
 @Override  
 public int getItemCount() {  
 return tangos.size();  
 }  
}

DictOperate.java

package com.leonelacs.tangochou;  
  
import android.content.Context;  
import android.database.sqlite.SQLiteDatabase;  
  
import java.io.File;  
import java.io.FileOutputStream;  
import java.io.IOException;  
import java.io.InputStream;  
  
public class DictOperate {  
 private final String DB\_PATH = "/data/data/com.leonelacs.tangochou/databases/";  
 private final String SRC\_DB\_FILENAME = "ms\_bing\_dict.db";  
 private Context context;  
 public DictOperate(Context context) {  
 this.context = context;  
 }  
 public SQLiteDatabase openDB() {  
 File dir = new File(DB\_PATH);  
 if (!dir.exists()) {  
 dir.mkdir();  
 }  
  
 String srcDBName = DB\_PATH + SRC\_DB\_FILENAME;  
  
 try {  
 InputStream inputStream = context.getResources().getAssets().open("ms\_bing\_dict.db");  
 FileOutputStream fos = new FileOutputStream(srcDBName);  
 byte[] buf = new byte[1024 \* 8];  
 int len = 0;  
 while ((len = inputStream.read(buf)) != -1) {  
 fos.write(buf, 0, len);  
 }  
 fos.close();  
 inputStream.close();  
 SQLiteDatabase database = SQLiteDatabase.*openOrCreateDatabase*(srcDBName, null);  
 return database;  
 }  
 catch (IOException e) {  
 e.printStackTrace();  
 return null;  
 }  
 }  
}

DBOHelper.java

package com.leonelacs.tangochou;  
  
import android.content.Context;  
import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteOpenHelper;  
import android.widget.Toast;  
  
public class DBOHelper extends SQLiteOpenHelper {  
  
 public static final String *CREATE\_CHOU* = "create table Chou ("  
 + "id integer primary key autoincrement, "  
 + "word text, "  
 + "definition text)";  
  
 private Context dboContext;  
  
 public DBOHelper(Context context, String name, SQLiteDatabase.CursorFactory factory, int version) {  
 super(context, name, factory, version);  
 dboContext = context;  
 }  
  
 @Override  
 public void onCreate(SQLiteDatabase sqLiteDatabase) {  
 sqLiteDatabase.execSQL(*CREATE\_CHOU*);  
 }  
  
 @Override  
 public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {  
 }  
}

GlanceActivity.java

package com.leonelacs.tangochou;  
  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
import android.os.Bundle;  
import android.support.annotation.NonNull;  
import android.support.design.widget.BottomNavigationView;  
import android.support.v7.app.AppCompatActivity;  
import android.support.v7.widget.LinearLayoutManager;  
import android.support.v7.widget.RecyclerView;  
import android.support.v7.widget.StaggeredGridLayoutManager;  
import android.text.Editable;  
import android.text.TextWatcher;  
import android.view.KeyEvent;  
import android.view.MenuItem;  
import android.view.View;  
import android.view.inputmethod.EditorInfo;  
import android.widget.EditText;  
import android.widget.LinearLayout;  
import android.widget.TextView;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class GlanceActivity extends AppCompatActivity {  
  
 List<TangoItem> tangoChou = new ArrayList<TangoItem>();  
  
  
 DBOHelper dbo = new DBOHelper(this, "saved\_tango.db", null, 1);  
  
  
 LinearLayoutManager layoutManagerChou = new LinearLayoutManager(this);  
 TangoAdapter adapterChou;  
  
 private BottomNavigationView.OnNavigationItemSelectedListener mOnNavigationItemSelectedListener  
 = new BottomNavigationView.OnNavigationItemSelectedListener() {  
  
 @Override  
 public boolean onNavigationItemSelected(@NonNull MenuItem item) {  
 switch (item.getItemId()) {  
 case R.id.*navigation\_home*:  
 findViewById(R.id.*SearchPage*).setVisibility(View.*VISIBLE*);  
 findViewById(R.id.*SavedChouPage*).setVisibility(View.*INVISIBLE*);  
 return true;  
 case R.id.*navigation\_dashboard*:  
 findViewById(R.id.*SearchPage*).setVisibility(View.*INVISIBLE*);  
 findViewById(R.id.*SavedChouPage*).setVisibility(View.*VISIBLE*);  
 return true;  
 case R.id.*navigation\_notifications*:  
 findViewById(R.id.*SearchPage*).setVisibility(View.*INVISIBLE*);  
 findViewById(R.id.*SavedChouPage*).setVisibility(View.*INVISIBLE*);  
 return true;  
 }  
 return false;  
 }  
 };  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_glance*);  
 EditText chouInput = (EditText)findViewById(R.id.*ChouInput*);  
 RecyclerView chouResult = (RecyclerView)findViewById(R.id.*ChouResult*);  
  
 BottomNavigationView navigation = (BottomNavigationView) findViewById(R.id.*navigation*);  
 navigation.setOnNavigationItemSelectedListener(mOnNavigationItemSelectedListener);  
  
 DictOperate dictOperate = new DictOperate(getBaseContext());  
 final SQLiteDatabase bingDict = dictOperate.openDB();  
 final EditText searchInput = (EditText)findViewById(R.id.*SearchInput*);  
 RecyclerView searchResult = (RecyclerView)findViewById(R.id.*SearchResult*);  
 final List<TangoItem> tangoList = new ArrayList<TangoItem>();  
  
  
 LinearLayoutManager layoutManager = new LinearLayoutManager(this);  
 searchResult.setLayoutManager(layoutManager);  
 final TangoAdapter adapter = new TangoAdapter(tangoList);  
 searchResult.setAdapter(adapter);  
  
 searchInChou("");  
 adapterChou = new TangoAdapter(tangoChou);  
 chouResult.setLayoutManager(layoutManagerChou);  
 chouResult.setAdapter(adapterChou);  
  
  
  
  
 searchInput.setOnEditorActionListener(new TextView.OnEditorActionListener() {  
 @Override  
 public boolean onEditorAction(TextView textView, int i, KeyEvent keyEvent) {  
 if (i == EditorInfo.*IME\_ACTION\_SEND* || i == EditorInfo.*IME\_ACTION\_DONE* || (keyEvent != null && KeyEvent.*KEYCODE\_ENTER* == keyEvent.getKeyCode() && KeyEvent.*ACTION\_DOWN* == keyEvent.getAction())) {  
 String curr = searchInput.getText().toString();  
 Cursor first = bingDict.query("Dict", new String[] {"word", "autoSugg"}, "word = ?", new String[] {curr}, null, null, null, "1");  
 Cursor cursor = bingDict.query("Dict", new String[]{"word", "autoSugg", "freq"}, "word like ? and word <> ?", new String[]{curr+"%", curr}, null, null, "freq desc");  
  
 tangoList.clear();  
  
 if (first.moveToFirst()) {  
 TangoItem firstTango = new TangoItem();  
 String word = first.getString(first.getColumnIndex("word"));  
 String definition = first.getString(first.getColumnIndex("autoSugg"));  
 firstTango.setWord(word);  
 firstTango.setDefinition(definition);  
 tangoList.add(firstTango);  
 if (cursor.moveToFirst()) {  
 do {  
 word = cursor.getString(cursor.getColumnIndex("word"));  
 definition = cursor.getString(cursor.getColumnIndex("autoSugg"));  
 TangoItem temp = new TangoItem();  
 temp.setWord(word);  
 temp.setDefinition(definition);  
 tangoList.add(temp);  
 }  
 while (cursor.moveToNext());  
 }  
 }  
  
 adapter.notifyDataSetChanged();  
 cursor.close();  
 first.close();  
 return true;  
 }  
 return false;  
 }  
 });  
  
 chouInput.setOnEditorActionListener(new TextView.OnEditorActionListener() {  
 EditText chouInput = (EditText)findViewById(R.id.*ChouInput*);  
 @Override  
 public boolean onEditorAction(TextView textView, int i, KeyEvent keyEvent) {  
 if (i == EditorInfo.*IME\_ACTION\_SEND* || i == EditorInfo.*IME\_ACTION\_DONE* || (keyEvent != null && KeyEvent.*KEYCODE\_ENTER* == keyEvent.getKeyCode() && KeyEvent.*ACTION\_DOWN* == keyEvent.getAction())) {  
 String curr = chouInput.getText().toString();  
 searchInChou(curr);  
 adapterChou.notifyDataSetChanged();  
 return true;  
 }  
 return false;  
 }  
 });  
  
  
  
 }  
  
 @Override  
 protected void onStart() {  
 super.onStart();  
 EditText chouInput = (EditText)findViewById(R.id.*ChouInput*);  
 String curr = chouInput.getText().toString();  
 searchInChou(curr);  
 adapterChou.notifyDataSetChanged();  
 }  
  
 protected void searchInChou(String input) {  
 SQLiteDatabase chouDB = dbo.getWritableDatabase();  
 Cursor cursor = chouDB.query("Chou", new String[]{"word", "definition"}, "word like ?", new String[]{input+"%"}, null, null, "word");  
 String word;  
 String definition;  
 tangoChou.clear();  
 if (cursor.moveToFirst()) {  
 do {  
 word = cursor.getString(cursor.getColumnIndex("word"));  
 definition = cursor.getString(cursor.getColumnIndex("definition"));  
 TangoItem temp = new TangoItem();  
 temp.setWord(word);  
 temp.setDefinition(definition);  
 tangoChou.add(temp);  
 }  
 while (cursor.moveToNext());  
 }  
 cursor.close();  
 }  
  
}

SavedChouActivity.java

package com.leonelacs.tangochou;  
  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.support.v7.widget.LinearLayoutManager;  
import android.support.v7.widget.RecyclerView;  
import android.view.KeyEvent;  
import android.view.inputmethod.EditorInfo;  
import android.widget.EditText;  
import android.widget.TextView;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class SavedChouActivity extends AppCompatActivity {  
  
 List<TangoItem> tangoChou = new ArrayList<TangoItem>();  
 EditText chouInput = (EditText)findViewById(R.id.*ChouInput*);  
 RecyclerView chouResult = (RecyclerView)findViewById(R.id.*ChouResult*);  
 DBOHelper dbo = new DBOHelper(this, "saved\_tango.db", null, 1);  
 SQLiteDatabase chouDB = dbo.getWritableDatabase();  
  
 LinearLayoutManager layoutManager = new LinearLayoutManager(this);  
 TangoAdapter adapter = new TangoAdapter(tangoChou);  
  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_saved\_chou*);  
  
 adapter = new TangoAdapter(tangoChou);  
 chouResult.setLayoutManager(layoutManager);  
 chouResult.setAdapter(adapter);  
  
 searchInChou("");  
  
 chouInput.setOnEditorActionListener(new TextView.OnEditorActionListener() {  
 @Override  
 public boolean onEditorAction(TextView textView, int i, KeyEvent keyEvent) {  
 if (i == EditorInfo.*IME\_ACTION\_SEND* || i == EditorInfo.*IME\_ACTION\_DONE* || (keyEvent != null && KeyEvent.*KEYCODE\_ENTER* == keyEvent.getKeyCode() && KeyEvent.*ACTION\_DOWN* == keyEvent.getAction())) {  
 String curr = chouInput.getText().toString();  
 searchInChou(curr);  
 adapter.notifyDataSetChanged();  
 return true;  
 }  
 return false;  
 }  
 });  
  
 }  
  
 @Override  
 protected void onStart() {  
 super.onStart();  
 String curr = chouInput.getText().toString();  
 searchInChou(curr);  
 adapter.notifyDataSetChanged();  
 }  
  
 protected void searchInChou(String input) {  
 Cursor cursor = chouDB.query("Chou", new String[]{"word", "autoSugg"}, "word like ?", new String[]{input+"%"}, null, null, "word");  
 String word;  
 String definition;  
 TangoItem ti = new TangoItem();  
 ti.setWord("xxTest");  
 ti.setDefinition("xxTESTxx");  
 tangoChou.clear();  
 tangoChou.add(ti);  
 if (cursor.moveToFirst()) {  
 do {  
 word = cursor.getString(cursor.getColumnIndex("word"));  
 definition = cursor.getString(cursor.getColumnIndex("autoSugg"));  
 TangoItem temp = new TangoItem();  
 temp.setWord(word);  
 temp.setDefinition(definition);  
 tangoChou.add(temp);  
 }  
 while (cursor.moveToNext());  
 }  
 cursor.close();  
 }  
}

DetailActivity.java

package com.leonelacs.tangochou;  
  
import android.content.ContentValues;  
import android.content.Intent;  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
import android.os.Bundle;  
import android.support.design.widget.FloatingActionButton;  
import android.support.design.widget.Snackbar;  
import android.support.v7.app.AppCompatActivity;  
import android.support.v7.widget.Toolbar;  
import android.view.View;  
import android.widget.TextView;  
  
public class DetailActivity extends AppCompatActivity {  
  
 int count;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_detail*);  
 Toolbar toolbar = (Toolbar) findViewById(R.id.*toolbar*);  
 setSupportActionBar(toolbar);  
 final FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.*fab*);  
 TextView resultWord = (TextView)findViewById(R.id.*ResultWord*);  
 TextView resultDefi = (TextView)findViewById(R.id.*ResultDefi*);  
 Intent intent = getIntent();  
 final String word = intent.getStringExtra("word");  
 final String definition = intent.getStringExtra("definition");  
 resultWord.setText(word);  
 resultDefi.setText(definition);  
 count = 0;  
 DBOHelper dbo = new DBOHelper(this, "saved\_tango.db", null, 1);  
 final SQLiteDatabase chouDB = dbo.getWritableDatabase();  
 Cursor cursor = chouDB.query("Chou", new String[] {"count(\*) as cnt"}, "word=?", new String[] {word}, null, null, null, "1");  
 if (cursor.moveToFirst()) {  
 count = cursor.getInt(cursor.getColumnIndex("cnt"));  
 }  
 cursor.close();  
 if (count != 1) {  
 fab.setImageResource(R.drawable.*big\_penta\_w*);  
 }  
 else {  
 fab.setImageResource(R.drawable.*big\_penta\_y*);  
 }  
  
 fab.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 String showOnSnackbar = "已将此单词";  
 if (count != 1) {  
 ContentValues contentValues = new ContentValues();  
 contentValues.put("word", word);  
 contentValues.put("definition", definition);  
 chouDB.insert("Chou", null, contentValues);  
 count = 1;  
 fab.setImageResource(R.drawable.*big\_penta\_y*);  
 showOnSnackbar = showOnSnackbar + "加入单词帐";  
 }  
 else {  
 chouDB.delete("Chou", "word=?", new String[] {word});  
 count = 0;  
 fab.setImageResource(R.drawable.*big\_penta\_w*);  
 showOnSnackbar = showOnSnackbar + "从单词帐移除";  
 }  
 Snackbar.*make*(view, showOnSnackbar, Snackbar.*LENGTH\_LONG*)  
 .setAction("Action", null).show();  
 }  
 });  
 }  
  
}