**基于Android的读书笔记软件**

1. **系统的需求分析**

**1.1系统目标**

本系统旨在使个人笔记更加容易管理，同时也可以管理自己的书籍。我的读书笔记适用于个人平时喜欢看一些书籍的人，界面简洁易懂，操作简单方便，让用户的使用更加方便。

**1.2功能需求**

该读书笔记软件具有以下功能：

用户管理：管理员登陆后可以管理使用该软件的所有用户；

书籍管理：用户可以自己添加书籍，查看所添加的所有书籍信息，对书籍进行管理；

书籍分类管理：每一本书都有自己的类别，用户可以对书籍的分类进行管理；

读书笔记：用户可以看到每一本书多对应得读书笔记，可以汇总每个用户的笔记数量并且统计某一个用户的每一本书的笔记数量；

系统设置：有关软件的一些说明，系统注销和用户账号切换。

1. **系统的概要设计**

该软件是一款是基于Android开发的手机应用，简单实用，易于上手。

* 1. 运行环境

1.Android手机或平板电脑：主频在1Ghz及以上，内存为512MB以上，系统版本为Android5.0及以上。

2. AndroidStudio+SDK+Windows10+SQLite

* 1. 数据库表设计

用户表

|  |  |  |
| --- | --- | --- |
| 列名 | 数据类型 | 是否允许空 |
| UserID | integer | 不允许 |
| UserName | text | 不允许 |
| UserPwd | text | 不允许 |

书籍分类表

|  |  |  |
| --- | --- | --- |
| 列名 | 数据类型 | 是否允许空 |
| ClassifyID | integer | 不允许 |
| ClassifyName | text | 不允许 |

书籍表

|  |  |  |
| --- | --- | --- |
| 列名 | 数据类型 | 是否允许空 |
| BookID | integer | 不允许 |
| BookName | text | 不允许 |
| BookAuthor | text | 不允许 |
| BookPress | text | 不允许 |
| BookISBN | text | 不允许 |
| BookClassify | text | 允许 |

读书笔记表

|  |  |  |
| --- | --- | --- |
| 列名 | 数据类型 | 是否允许空 |
| UserID | integer | 不允许 |
| BookID | integer | 不允许 |
| NoteDate | text | 不允许 |
| NoteContent | text | 允许 |

1. **设计与实现部分**
   1. 用户登陆



关键代码：

package com.stu.mybook;

import android.app.Activity;

import android.content.ContentValues;

import android.content.Context;

import android.content.Intent;

import android.content.SharedPreferences;

import android.database.Cursor;

import android.database.sqlite.SQLiteDatabase;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.text.TextUtils;

import android.util.Log;

import android.view.View;

import android.widget.EditText;

import android.widget.TextView;

import android.widget.Toast;

import java.io.File;

import java.io.FileInputStream;

import java.io.FileOutputStream;

import java.util.Properties;

public class MainActivity extends AppCompatActivity {

// 变量 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

private DB db = null;

private EditText username = null;

private EditText pwd = null;

private TextView tvMainInfo = null;

private String name = null;

private String password = null;

private static Context context;

private SQLiteDatabase dbo;

private int userID=0;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

username = (EditText)findViewById(R.id.username);

pwd=(EditText)findViewById(R.id.pwd);

btnMainCreateDB\_Click();

oldUserLogin();

}

// 函数自定义 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// 创建数据库

public void btnMainCreateDB\_Click()

{

Log.d("EBook","调用MainActivity.btnMainCreateDB\_Click()");

db = new DB(this,"dbStudentMis.db",null,1);

Log.d("EBook","调用db.getReadableDatabase");

db.getReadableDatabase();

}

// 用户点击登陆按钮

public void btnLogin\_Click(View view)

{

dbo = db.getWritableDatabase();

String strSql = "";

name = username.getText().toString();

password = pwd.getText().toString();

strSql = "select \* from Student";

Cursor cursor;

cursor = dbo.rawQuery(strSql,null);

boolean isname=false;

String strPwd = "";

String strName = "";

while (cursor.moveToNext())

{

userID = cursor.getInt(cursor.getColumnIndex("UserID"));

strName = cursor.getString(cursor.getColumnIndex("UserName"));

strPwd = cursor.getString(cursor.getColumnIndex("UserPwd"));

if (strName.equals(name))

{

isname=true;

if(strPwd.equals(password))

{

saveProUserInfo(this,userID,name,password);

Toast.makeText(MainActivity.this, "登陆成功",Toast.LENGTH\_SHORT).show();

Intent intent = new Intent(MainActivity.this,UserActivity.class);

MainActivity.this.startActivity(intent);

finish();

break;

}

else

{

Toast.makeText(MainActivity.this, "密码错误",Toast.LENGTH\_SHORT).show();

break;

}

}

else

{

isname=false;

}

}

if(!isname)

{

btnMainInsertData\_Click();

saveProUserInfo(this,userID,name,password);

Toast.makeText(MainActivity.this, "注册成功",Toast.LENGTH\_SHORT).show();

Intent intent = new Intent(MainActivity.this,UserActivity.class);

MainActivity.this.startActivity(intent);

finish();

}

}

public void btnMainInsertData\_Click()

{

ContentValues cv = new ContentValues();

cv.put("UserName", name);

cv.put("UserPwd",password);

db.getWritableDatabase().insert("Student", null, cv);

dbo = db.getWritableDatabase();

String strSql = "select \* from Student where UserName = '"+name+"' and UserPwd = '"+password+"'";

Cursor cursor;

cursor = dbo.rawQuery(strSql,null);

if(cursor.moveToNext())

{

userID = cursor.getInt(cursor.getColumnIndex("UserID"));

}

}

/\*\*

\* 使用属性文件保存用户的信息

\*

\* @param con 上下文

\* @param userID 用户编号

\* @param username 用户名

\* @param password 密码

\* @return

\*/

public static boolean saveProUserInfo(Context con, int userID, String username,

String password) {

try {

context = con;

SharedPreferences myPreferences = con.getSharedPreferences("user\_info", Activity.MODE\_PRIVATE);

SharedPreferences.Editor editor = myPreferences.edit();

editor.putString("userID", String.valueOf(userID));

editor.putString("username", username);

editor.putString("password", password);

editor.commit();

return true;

} catch (Exception e) {

throw new RuntimeException();

}

}

public void oldUserLogin()

{

SharedPreferences myPreferences = getSharedPreferences("user\_info", Activity.MODE\_PRIVATE);

// 获取用户名或密码

if (null != myPreferences) {

String strname=myPreferences.getString("username","");

String strpwd=myPreferences.getString("password","");

// 如果获取到的用户名或密码不为空，则设置到文本框中

if (!TextUtils.isEmpty(strname) && !TextUtils.isEmpty(strpwd)) {

// 设置用户名

username.setText(strname);

// 设置密码

pwd.setText(strpwd);

Bundle bundle = this.getIntent().getExtras();

if(bundle==null)

{

Toast.makeText(MainActivity.this, "登陆成功",Toast.LENGTH\_SHORT).show();

Intent intent = new Intent(MainActivity.this,UserActivity.class);

MainActivity.this.startActivity(intent);

finish();

}

}

}

}

}

用户主界面：





关键代码：

package com.stu.mybook;

import android.annotation.SuppressLint;

import android.content.DialogInterface;

import android.content.Intent;

import android.graphics.Color;

import android.os.Bundle;

import android.provider.Settings;

import android.support.v4.app.Fragment;

import android.support.v4.app.FragmentManager;

import android.support.v4.app.FragmentTransaction;

import android.support.v7.app.ActionBarDrawerToggle;

import android.support.v7.app.AlertDialog;

import android.support.v7.app.AppCompatActivity;

import android.support.v7.widget.PopupMenu;

import android.support.v7.widget.Toolbar;

import android.view.Menu;

import android.view.MenuInflater;

import android.view.MenuItem;

import android.view.View;

import android.widget.AdapterView;

import android.widget.AdapterView.OnItemClickListener;

import android.widget.ArrayAdapter;

import android.widget.ListView;

import android.support.v4.widget.DrawerLayout;

import android.widget.Toast;

public class UserActivity extends AppCompatActivity

{

private Toolbar toolbar;

private ActionBarDrawerToggle mDrawerToggle;

private String[] mPlanetTitles;

private DrawerLayout mDrawerLayout;

private ListView mDrawerList;

private Fragment[] fragments={new UserManage(),new BookManage(),new BookClassifyManage(),new NotesManage()};

private FragmentManager fragmentManager;

private Fragment maincontent;

@Override

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.drawer\_layout);

ExitApplication.getInstance().addActivity(this);

findView();

setDefaultFragment();

initToolbar();

initListView();

}

@Override

public boolean onCreateOptionsMenu(Menu menu) {

getMenuInflater().inflate(R.menu.mymain, menu);

return true;

}

@Override

public boolean onOptionsItemSelected(MenuItem item) {

final int id = item.getItemId();

if (id == R.id.calendar)

{

Toast.makeText(UserActivity.this,"User",Toast.LENGTH\_SHORT).show();

return true;

}

else if(id == R.id.about)

{

String msg = "开发人：孟祥通\n当前版本：v1.0\n联系方式：15227196900";

new AlertDialog.Builder(UserActivity.this).setTitle("软件信息")//设置对话框标题

.setMessage(msg)

.setPositiveButton("确定",new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

dialog.dismiss();

}

}).show();

return true;

}

else if(id == R.id.action\_settings)

{

ExitApplication.getInstance().exit();

return true;

}

else if(id == R.id.exit)

{

new AlertDialog.Builder(UserActivity.this).setTitle("账号切换")//设置对话框标题

.setMessage("确认退出该账号?")

.setNegativeButton("取消",new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

dialog.dismiss();

}

})

.setPositiveButton("确定",new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

Intent intent = new Intent(UserActivity.this,MainActivity.class);

Bundle bundle = new Bundle();

bundle.putString("exit","0");

intent.putExtras(bundle);

startActivity(intent);

finish();

}

}).show();

return true;

}

else{}

return super.onOptionsItemSelected(item);

}

public boolean onMenuItemClick(MenuItem item) {

return true;

}

private void findView()

{

mDrawerLayout = (DrawerLayout) findViewById(R.id.drawer\_layout);

toolbar = (Toolbar) findViewById(R.id.toolbar);

fragmentManager = getSupportFragmentManager();

}

private void setDefaultFragment()

{

FragmentTransaction transaction = fragmentManager.beginTransaction();

int index;

Bundle bundle = this.getIntent().getExtras();

if(bundle==null)

{

maincontent = new Welcome();

}

else {

index = Integer.valueOf(bundle.getString("index"));

maincontent = fragments[index];

}

transaction.replace(R.id.content\_frame,maincontent);

transaction.commit();

}

private void initToolbar()

{

toolbar.setTitle("MyBook");//设置Toolbar标题

toolbar.setTitleTextColor(Color.parseColor("#ffffff")); //设置标题颜色

setSupportActionBar(toolbar);

getSupportActionBar().setHomeButtonEnabled(true); //设置返回键可用

getSupportActionBar().setDisplayHomeAsUpEnabled(true);

//创建返回键，并实现打开关/闭监听

mDrawerToggle = new ActionBarDrawerToggle(this, mDrawerLayout, toolbar, R.string.drawer\_open, R.string.drawer\_close) {

@Override

public void onDrawerOpened(View drawerView) {

super.onDrawerOpened(drawerView);

}

@Override

public void onDrawerClosed(View drawerView) {

super.onDrawerClosed(drawerView);

}

};

mDrawerToggle.syncState();

mDrawerLayout.setDrawerListener(mDrawerToggle);

}

private void initListView()

{

mDrawerList = (ListView) findViewById(R.id.left\_drawer);

mPlanetTitles = getResources().getStringArray(R.array.item\_array);

mDrawerList.setAdapter(new ArrayAdapter<>(this,

android.R.layout.simple\_list\_item\_1, mPlanetTitles));

mDrawerList.setOnItemClickListener(new OnItemClickListener()

{

@Override

public void onItemClick(AdapterView<?> parent, View view,

int position, long id)

{

FragmentTransaction fragmentTransaction =fragmentManager.beginTransaction();

fragmentTransaction.replace(R.id.content\_frame,fragments[position]);

fragmentTransaction.commit();

mDrawerLayout.closeDrawer(mDrawerList);

}

});

}

}

用户管理：









关键代码：

package com.stu.mybook;

import android.content.Intent;

import android.database.Cursor;

import android.os.Bundle;

import android.support.v4.app.Fragment;

import android.util.Log;

import android.view.ContextMenu;

import android.view.LayoutInflater;

import android.view.MenuItem;

import android.view.View;

import android.view.ViewGroup;

import android.widget.AdapterView;

import android.widget.Button;

import android.widget.ListView;

import java.util.ArrayList;

import java.util.HashMap;

import java.util.List;

import java.util.Map;

/\*\*

\* Created by Administrator on 2017/4/3 0003.

\*/

public class UserManage extends Fragment {

private ListView listView;

private List<Map<String,String>> data;

private DB db;

private Button btnAddUser;

private int MID;

private MyAdapter myadapter;

@Override

public boolean onContextItemSelected(MenuItem item) {

AdapterView.AdapterContextMenuInfo info = (AdapterView.AdapterContextMenuInfo)item.getMenuInfo();

MID = (int)info.id;

Log.d("db","onContextItemSelected");

switch (item.getItemId())

{

case 0:

Log.d("db", String.valueOf(MID));

String id=data.get(MID).get("id");

String sql="delete from Student where UserID='"+id+"'";

db.getWritableDatabase().execSQL(sql);

myadapter.remove(MID);

break;

case 1:

Log.d("db",String.valueOf(MID));

Intent intent = new Intent(getActivity().getApplicationContext(),UpdateUserActivity.class);

Bundle bundle = new Bundle();

bundle.putString("id",data.get(MID).get("id"));

bundle.putString("name",data.get(MID).get("name"));

bundle.putString("password",data.get(MID).get("password"));

intent.putExtras(bundle);

startActivity(intent);

break;

}

return super.onContextItemSelected(item);

}

@Override

public void onResume() {

super.onResume();

}

@Override

public View onCreateView(LayoutInflater inflater,ViewGroup container,Bundle savedInstanceState) {

Log.d("db","onCreateView");

db = new DB(getActivity(),"dbStudentMis.db",null,1);

db.getReadableDatabase();

View view = inflater.inflate(R.layout.view1,null);

listView = (ListView) view.findViewById(R.id.userList);

registerForContextMenu(listView);

btnAddUser = (Button) view.findViewById(R.id.btnAddUser);

btnAddUser.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

Intent intent = new Intent(getActivity(),AddUserActivity.class);

startActivity(intent);

}

});

listView.setOnItemLongClickListener(new AdapterView.OnItemLongClickListener(){

@Override

public boolean onItemLongClick(AdapterView<?> adapterView, View view, int i, long l) {

listView.setOnCreateContextMenuListener(new View.OnCreateContextMenuListener() {

@Override

public void onCreateContextMenu(ContextMenu contextMenu, View view, ContextMenu.ContextMenuInfo contextMenuInfo) {

contextMenu.add(0, 0, 0, "删除");

contextMenu.add(0, 1, 0, "修改");

}

});

return false;

}

});

return view;

}

@Override

public void onActivityCreated(Bundle savedInstanceState) {

super.onActivityCreated(savedInstanceState);

Log.d("db","onActivityCreated");

fillData();

myadapter = new MyAdapter(getActivity(),data);

listView.setAdapter(myadapter);

}

public void fillData()

{

data = new ArrayList<Map<String, String>>();

String sql="select \* from Student";

try {

Cursor cursor = db.getReadableDatabase().rawQuery(sql,null);

while (cursor.moveToNext())

{

Map<String,String> user = new HashMap<String,String>();

Log.d("UserManage",cursor.getString(cursor.getColumnIndex("UserID")));

user.put("id",cursor.getString(cursor.getColumnIndex("UserID")));

user.put("name",cursor.getString(cursor.getColumnIndex("UserName")));

user.put("password",cursor.getString(cursor.getColumnIndex("UserPwd")));

data.add(user);

}

} catch (Exception e) {

e.printStackTrace();

}

}

}

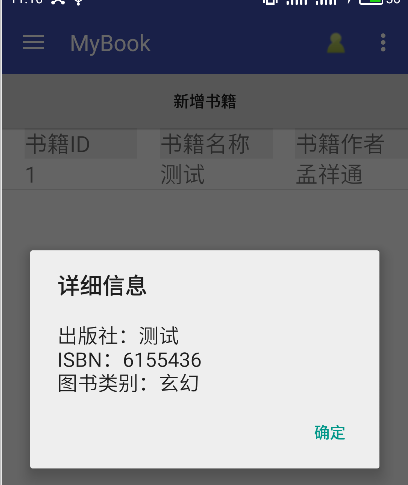
书籍管理：













关键代码：

package com.stu.mybook;

import android.content.DialogInterface;

import android.content.Intent;

import android.database.Cursor;

import android.os.Bundle;

import android.support.v4.app.Fragment;

import android.support.v7.app.AlertDialog;

import android.util.Log;

import android.view.ContextMenu;

import android.view.LayoutInflater;

import android.view.MenuItem;

import android.view.View;

import android.view.ViewGroup;

import android.widget.AdapterView;

import android.widget.Button;

import android.widget.ListView;

import java.util.ArrayList;

import java.util.HashMap;

import java.util.List;

import java.util.Map;

public class BookManage extends Fragment {

private ListView listView;

private List<Map<String,String>> data;

private DB db;

private Button btnAddBook;

private int MID;

private BookAdapter bookAdapter;

@Override

public boolean onContextItemSelected(MenuItem item) {

AdapterView.AdapterContextMenuInfo info = (AdapterView.AdapterContextMenuInfo)item.getMenuInfo();

MID = (int)info.id;

Log.d("db","onContextItemSelected");

switch (item.getItemId())

{

case 0:

Log.d("db", String.valueOf(MID));

String id=data.get(MID).get("id");

String sql="delete from Book where BookID='"+id+"'";

db.getWritableDatabase().execSQL(sql);

bookAdapter.remove(MID);

break;

case 1:

Log.d("db",String.valueOf(MID));

Intent intent = new Intent(getActivity().getApplicationContext(),UpdateBookActivity.class);

Bundle bundle = new Bundle();

bundle.putString("id",data.get(MID).get("id"));

bundle.putString("name",data.get(MID).get("name"));

bundle.putString("author",data.get(MID).get("author"));

bundle.putString("press",data.get(MID).get("press"));

bundle.putString("isbn",data.get(MID).get("isbn"));

bundle.putString("classify",data.get(MID).get("classify"));

intent.putExtras(bundle);

startActivity(intent);

break;

case 2:

Log.d("db",String.valueOf(MID));

String press = data.get(MID).get("press");

String isbn = data.get(MID).get("isbn");

String classify = data.get(MID).get("classify");

String msg = "出版社："+press+"\n"+"ISBN："+isbn+"\n"+"图书类别："+classify;

new AlertDialog.Builder(getActivity()).setTitle("详细信息")//设置对话框标题

.setMessage(msg)

.setPositiveButton("确定",new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

dialog.dismiss();

}

}).show();

break;

case 3:

Intent intent1 = new Intent(getActivity().getApplicationContext(),AddNotesActivity.class);

Bundle bundle1 = new Bundle();

bundle1.putString("bookID",data.get(MID).get("id"));

intent1.putExtras(bundle1);

startActivity(intent1);

break;

}

return super.onContextItemSelected(item);

}

@Override

public void onResume() {

super.onResume();

}

@Override

public View onCreateView(LayoutInflater inflater,ViewGroup container,Bundle savedInstanceState) {

Log.d("db","onCreateView");

db = new DB(getActivity(),"dbStudentMis.db",null,1);

db.getReadableDatabase();

View view = inflater.inflate(R.layout.view2,null);

listView = (ListView) view.findViewById(R.id.bookList);

registerForContextMenu(listView);

btnAddBook = (Button) view.findViewById(R.id.btnAddBook);

btnAddBook.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

Intent intent = new Intent(getActivity(),AddBookActivity.class);

startActivity(intent);

}

});

listView.setOnItemLongClickListener(new AdapterView.OnItemLongClickListener(){

@Override

public boolean onItemLongClick(AdapterView<?> adapterView, View view, int i, long l) {

listView.setOnCreateContextMenuListener(new View.OnCreateContextMenuListener() {

@Override

public void onCreateContextMenu(ContextMenu contextMenu, View view, ContextMenu.ContextMenuInfo contextMenuInfo) {

contextMenu.add(0, 0, 0, "删除");

contextMenu.add(0, 1, 0, "修改");

contextMenu.add(0, 2, 0, "详情");

contextMenu.add(0, 3, 0, "添加笔记");

}

});

return false;

}

});

return view;

}

@Override

public void onActivityCreated(Bundle savedInstanceState) {

super.onActivityCreated(savedInstanceState);

Log.d("db","onActivityCreated");

fillData();

bookAdapter = new BookAdapter(getActivity(),data);

listView.setAdapter(bookAdapter);

}

public void fillData()

{

data = new ArrayList<Map<String, String>>();

String sql="select \* from Book";

try {

Cursor cursor = db.getReadableDatabase().rawQuery(sql,null);

while (cursor.moveToNext())

{

Map<String,String> book = new HashMap<String,String>();

Log.d("BookManage",cursor.getString(cursor.getColumnIndex("BookID")));

book.put("id",cursor.getString(cursor.getColumnIndex("BookID")));

book.put("name",cursor.getString(cursor.getColumnIndex("BookName")));

book.put("author",cursor.getString(cursor.getColumnIndex("BookAuthor")));

book.put("press",cursor.getString(cursor.getColumnIndex("BookPress")));

book.put("isbn",cursor.getString(cursor.getColumnIndex("BookISBN")));

book.put("classify",cursor.getString(cursor.getColumnIndex("BookClassify")));

data.add(book);

}

} catch (Exception e) {

e.printStackTrace();

}

}

}

书籍分类管理：







关键代码：

package com.stu.mybook;

import android.content.DialogInterface;

import android.content.Intent;

import android.database.Cursor;

import android.os.Bundle;

import android.support.v4.app.Fragment;

import android.support.v7.app.AlertDialog;

import android.util.Log;

import android.view.ContextMenu;

import android.view.LayoutInflater;

import android.view.MenuItem;

import android.view.View;

import android.view.ViewGroup;

import android.widget.AdapterView;

import android.widget.Button;

import android.widget.ListView;

import java.util.ArrayList;

import java.util.HashMap;

import java.util.List;

import java.util.Map;

/\*\*

\* Created by Administrator on 2017/4/3 0003.

\*/

public class BookClassifyManage extends Fragment {

private ListView listView;

private List<Map<String,String>> data;

private DB db;

private Button btnAddClassify;

private int MID;

private ClassifyAdapter classifyAdapter;

public boolean onContextItemSelected(MenuItem item) {

AdapterView.AdapterContextMenuInfo info = (AdapterView.AdapterContextMenuInfo)item.getMenuInfo();

MID = (int)info.id;

Log.d("db","onContextItemSelected");

switch (item.getItemId())

{

case 0:

Log.d("db", String.valueOf(MID));

String id=data.get(MID).get("id");

String name=data.get(MID).get("name");

String strsql = "select COUNT(\*) from Book where BookClassify = '"+name+"'";

Cursor cursor = db.getWritableDatabase().rawQuery(strsql,null);

if(cursor.moveToNext()) {

String msg = "该分类下已有图书，无法删除！";

new AlertDialog.Builder(getActivity()).setTitle("警告")//设置对话框标题

.setMessage(msg)

.setPositiveButton("确定",new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

dialog.dismiss();

}

}).show();

}

else {

String sql = "delete from BookClassify where ClassifyID='" + id + "'";

db.getWritableDatabase().execSQL(sql);

classifyAdapter.remove(MID);

}

break;

case 1:

Log.d("db",String.valueOf(MID));

Intent intent = new Intent(getActivity().getApplicationContext(),UpdateClassifyActivity.class);

Bundle bundle = new Bundle();

bundle.putString("id",data.get(MID).get("id"));

bundle.putString("name",data.get(MID).get("name"));

intent.putExtras(bundle);

startActivity(intent);

break;

}

return super.onContextItemSelected(item);

}

@Override

public void onResume() {

super.onResume();

}

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {

Log.d("db","onCreateView");

db = new DB(getActivity(),"dbStudentMis.db",null,1);

db.getReadableDatabase();

View view = inflater.inflate(R.layout.view3,null);

listView = (ListView) view.findViewById(R.id.classifyList);

registerForContextMenu(listView);

btnAddClassify = (Button) view.findViewById(R.id.btnAddBookClassify);

btnAddClassify.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

Intent intent = new Intent(getActivity(),AddClassifyActivity.class);

startActivity(intent);

}

});

listView.setOnItemLongClickListener(new AdapterView.OnItemLongClickListener(){

@Override

public boolean onItemLongClick(AdapterView<?> adapterView, View view, int i, long l) {

listView.setOnCreateContextMenuListener(new View.OnCreateContextMenuListener() {

@Override

public void onCreateContextMenu(ContextMenu contextMenu, View view, ContextMenu.ContextMenuInfo contextMenuInfo) {

contextMenu.add(0, 0, 0, "删除");

contextMenu.add(0, 1, 0, "修改");

}

});

return false;

}

});

return view;

}

@Override

public void onActivityCreated(Bundle savedInstanceState) {

super.onActivityCreated(savedInstanceState);

Log.d("db","onActivityCreated");

fillData();

classifyAdapter = new ClassifyAdapter(getActivity(),data);

listView.setAdapter(classifyAdapter);

}

public void fillData()

{

data = new ArrayList<Map<String, String>>();

String sql="select \* from BookClassify";

try {

Cursor cursor = db.getReadableDatabase().rawQuery(sql,null);

while (cursor.moveToNext())

{

Map<String,String> user = new HashMap<String,String>();

user.put("id",cursor.getString(cursor.getColumnIndex("ClassifyID")));

user.put("name",cursor.getString(cursor.getColumnIndex("ClassifyName")));

data.add(user);

}

} catch (Exception e) {

e.printStackTrace();

}

}

}

读书笔记：







关键代码：

package com.stu.mybook;

import android.app.Activity;

import android.content.Intent;

import android.content.SharedPreferences;

import android.database.Cursor;

import android.os.Bundle;

import android.support.v4.app.Fragment;

import android.util.Log;

import android.view.ContextMenu;

import android.view.LayoutInflater;

import android.view.MenuItem;

import android.view.View;

import android.view.ViewGroup;

import android.widget.AdapterView;

import android.widget.Button;

import android.widget.ListView;

import android.widget.Toast;

import java.util.ArrayList;

import java.util.HashMap;

import java.util.List;

import java.util.Map;

public class NotesManage extends Fragment {

private ListView listView;

private List<Map<String,String>> data;

private DB db;

private String userID;

private Button btnUserNotes;

private int MID;

private NotesAdapter notesAdapter;

@Override

public boolean onContextItemSelected(MenuItem item) {

AdapterView.AdapterContextMenuInfo info = (AdapterView.AdapterContextMenuInfo)item.getMenuInfo();

MID = (int)info.id;

Log.d("db","onContextItemSelected");

switch (item.getItemId())

{

case 0:

Log.d("db", String.valueOf(MID));

String userID=data.get(MID).get("userID");

String bookID=data.get(MID).get("bookID");

String sql="delete from Notes where UserID='"+userID+"' and BookID='"+bookID+"'";

db.getWritableDatabase().execSQL(sql);

notesAdapter.remove(MID);

break;

}

return super.onContextItemSelected(item);

}

@Override

public void onResume() {

super.onResume();

}

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {

Log.d("db","onCreateView");

db = new DB(getActivity(),"dbStudentMis.db",null,1);

db.getReadableDatabase();

View view = inflater.inflate(R.layout.view4,null);

listView = (ListView) view.findViewById(R.id.notesList);

registerForContextMenu(listView);

btnUserNotes = (Button) view.findViewById(R.id.btnUserNotes);

btnUserNotes.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

Intent intent = new Intent(getActivity(),UserNotesActivity.class);

startActivity(intent);

}

});

listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {

@Override

public void onItemClick(AdapterView<?> parent, View view, int position, long id) {

String userID = data.get(position).get("userID");

String bookID = data.get(position).get("bookID");

Intent intent = new Intent(getActivity().getApplicationContext(),LookNotesActivity.class);

Bundle bundle = new Bundle();

bundle.putString("userID",userID);

bundle.putString("bookID",bookID);

intent.putExtras(bundle);

startActivity(intent);

}

});

listView.setOnItemLongClickListener(new AdapterView.OnItemLongClickListener(){

@Override

public boolean onItemLongClick(AdapterView<?> adapterView, View view, int i, long l) {

listView.setOnCreateContextMenuListener(new View.OnCreateContextMenuListener() {

@Override

public void onCreateContextMenu(ContextMenu contextMenu, View view, ContextMenu.ContextMenuInfo contextMenuInfo) {

contextMenu.add(0, 0, 0, "删除");

}

});

return false;

}

});

return view;

}

@Override

public void onActivityCreated(Bundle savedInstanceState) {

super.onActivityCreated(savedInstanceState);

Log.d("db","onActivityCreated");

SharedPreferences myPreferences = getActivity().getSharedPreferences("user\_info", Activity.MODE\_PRIVATE);

// 获取用户名或密码

if (null != myPreferences) {

userID = myPreferences.getString("userID","");

}

fillData();

notesAdapter = new NotesAdapter(getActivity(),data);

listView.setAdapter(notesAdapter);

}

public void fillData()

{

data = new ArrayList<Map<String, String>>();

String sql="select BookID, COUNT(\*) as notes from Notes where UserID = "+ userID + " group by BookID";

try {

Cursor cursor = db.getReadableDatabase().rawQuery(sql,null);

while (cursor.moveToNext())

{

Map<String,String> note = new HashMap<String,String>();

note.put("userID",userID);

String bookID = cursor.getString(cursor.getColumnIndex("BookID"));

note.put("bookID",bookID);

String bookName = GetBookName(bookID);

note.put("bookName",bookName);

note.put("bookNotes",cursor.getString(cursor.getColumnIndex("notes")));

data.add(note);

}

} catch (Exception e) {

e.printStackTrace();

}

}

public String GetBookName(String bookID)

{

String sql="select BookName from Book where BookID = " + bookID;

String bookName="";

try {

Cursor cursor = db.getReadableDatabase().rawQuery(sql,null);

if (cursor.moveToNext())

{

bookName = cursor.getString(cursor.getColumnIndex("BookName"));

}

} catch (Exception e) {

e.printStackTrace();

}

return bookName;

}

}

1. **心得体会**

每当学习新的技术，总是会有意想不到的困难出现，可也正是这些困难才使我真正学到有用的知识，Android学习了这么长时间，也有了不小的收获，一些常用的控件也都用过，也学习了设配器的使用，想说的是在某些地方android实现起来很不方便，很多时候要借用设配器来完成。