package com.example.myapps;

import static android.content.ContentValues.TAG;

import androidx.appcompat.app.AlertDialog;

import androidx.appcompat.app.AppCompatActivity;

import android.content.ContentValues;

import android.content.DialogInterface;

import android.database.Cursor;

import android.database.sqlite.SQLiteDatabase;

import android.os.Bundle;

import android.util.Log;

import android.view.Menu;

import android.view.MenuItem;

import android.view.View;

import android.widget.ArrayAdapter;

import android.widget.EditText;

import android.widget.ListView;

import android.widget.TextView;

import com.example.myapps.db.TaskContract;

import com.example.myapps.db.TaskDbHelper;

import java.util.ArrayList;

public class MainActivity extends AppCompatActivity {

private static final String TAG = "MainActivity";

private TaskDbHelper mHelper;

private ListView mTaskListView;

private ArrayAdapter<String> mAdapter;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

mHelper = new TaskDbHelper(this);

mTaskListView = (ListView) findViewById(R.id.list\_todo);

updateUI();

}

@Override

public boolean onCreateOptionsMenu(Menu menu) {

getMenuInflater().inflate(R.menu.main\_menu, menu);

return super.onCreateOptionsMenu(menu);

}

private void updateUI() {

ArrayList<String> taskList = new ArrayList<>();

SQLiteDatabase db = mHelper.getReadableDatabase();

Cursor cursor = db.query(TaskContract.TaskEntry.TABLE,

new String[]{TaskContract.TaskEntry.\_ID, TaskContract.TaskEntry.COL\_TASK\_TITLE},

null, null, null, null, null);

while (cursor.moveToNext()) {

int idx = cursor.getColumnIndex(TaskContract.TaskEntry.COL\_TASK\_TITLE);

taskList.add(cursor.getString(idx));

}

if (mAdapter == null) {

mAdapter = new ArrayAdapter<>(this,

R.layout.item\_todo,

R.id.task\_title,

taskList);

mTaskListView.setAdapter(mAdapter);

} else {

mAdapter.clear();

mAdapter.addAll(taskList);

mAdapter.notifyDataSetChanged();

}

cursor.close();

db.close();

}

public void deleteTask(View view) {

View parent = (View) view.getParent();

TextView taskTextView = (TextView) parent.findViewById(R.id.task\_title);

String task = String.valueOf(taskTextView.getText());

SQLiteDatabase db = mHelper.getWritableDatabase();

db.delete(TaskContract.TaskEntry.TABLE,

TaskContract.TaskEntry.COL\_TASK\_TITLE + " = ?",

new String[]{task});

db.close();

updateUI();

}

@Override

public boolean onOptionsItemSelected(MenuItem item) {

switch (item.getItemId()) {

case R.id.action\_add\_task:

final EditText taskEditText = new EditText(this);

AlertDialog dialog = new AlertDialog.Builder(this)

.setTitle("Add a new task")

.setMessage("What do you want to do next?")

.setView(taskEditText)

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

@Override

public void onClick(DialogInterface dialog, int which) {

String task = String.valueOf(taskEditText.getText());

SQLiteDatabase db = mHelper.getWritableDatabase();

ContentValues values = new ContentValues();

values.put(TaskContract.TaskEntry.COL\_TASK\_TITLE, task);

db.insertWithOnConflict(TaskContract.TaskEntry.TABLE,

null,

values,

SQLiteDatabase.CONFLICT\_REPLACE);

db.close();

updateUI();

}

})

.setNegativeButton("Cancel", null)

.create();

dialog.show();

Log.d(TAG, "Add a new task");

return true;

default:

return super.onOptionsItemSelected(item);

}

}

}