Android 2 RV lab assessment

Functionality:

* Create note
* Edit note
* Delete note
* Set priority as first notes (sort by priority DESC, name ASC)

Code: Database, Notes, Main activity items

Chart, table, treemap chart

Description automatically generated

MainDAO

package com.example.rvassessment.database;  
  
import static androidx.room.OnConflictStrategy.*REPLACE*;  
  
  
import androidx.room.Dao;  
import androidx.room.Delete;  
import androidx.room.Insert;  
import androidx.room.Query;  
import androidx.room.Update;  
  
import com.example.rvassessment.Notes.NoteModel;  
  
import java.util.List;  
  
@Dao  
public interface MainDAO {  
 @Insert(onConflict = *REPLACE*)  
 void insert(NoteModel notes);  
  
 @Query("SELECT \* FROM notes ORDER BY priority DESC, title ASC")  
 List<NoteModel> getAll();  
  
 @Query("UPDATE notes SET title = :title, description = :description, priority = :priority WHERE ID = :id")  
 void update (int id, String title, String description, boolean priority);  
  
 @Delete  
 void delete(NoteModel notes);  
  
 @Query("UPDATE notes SET priority = :priority WHERE ID = :id")  
 void setPriority(int id, boolean priority);  
  
 @Update  
 void update(NoteModel note);  
}

RoomDB

package com.example.rvassessment.database;  
  
import android.content.Context;  
  
import androidx.room.Database;  
import androidx.room.Room;  
import androidx.room.RoomDatabase;  
  
import com.example.rvassessment.Notes.NoteModel;  
  
@Database(entities = NoteModel.class, version = 1, exportSchema = false)  
public abstract class RoomDB extends RoomDatabase {  
 private static RoomDB *database*;  
 private static final String *DATABASE\_NAME* = "NoteApp";  
  
 public synchronized static RoomDB getInstance(Context context) {  
 if (*database* == null) {  
 *database* = Room.*databaseBuilder*(context.getApplicationContext(),  
 RoomDB.class, *DATABASE\_NAME*)  
 .allowMainThreadQueries()  
 .fallbackToDestructiveMigration()  
 .build();  
 }  
 return *database*;  
 }  
  
 static RoomDB *appDatabase*;  
  
 public static RoomDB getAppDatabase(Context context) {  
 if (*appDatabase* == null) {  
 *appDatabase* = Room.*databaseBuilder*(context, RoomDB.class, *DATABASE\_NAME*).allowMainThreadQueries().fallbackToDestructiveMigration().build();  
 }  
 return *appDatabase*;  
 }  
  
 public abstract MainDAO mainDAO();  
}

NoteModel

package com.example.rvassessment.Notes;  
  
import androidx.room.ColumnInfo;  
import androidx.room.Entity;  
import androidx.room.PrimaryKey;  
  
import java.io.Serializable;  
import java.sql.Date;  
  
@Entity(tableName = "notes")  
public class NoteModel implements Serializable{  
  
 @PrimaryKey(autoGenerate = true)  
 int ID = 0;  
 @ColumnInfo(name = "title")  
 String title = "";  
 @ColumnInfo(name = "description")  
 String description = "";  
 @ColumnInfo(name = "date")  
 String date = "";  
 @ColumnInfo(name = "priority")  
 public  
 boolean priority;  
  
  
 public int getID() {  
 return ID;  
 }  
  
 public void setID(int ID) {  
 this.ID = ID;  
 }  
  
 public String getTitle() {  
 return this.title;  
 }  
  
 public void setTitle(String title) {  
 this.title = title;  
 }  
  
 public String getDescription() {  
 return this.description;  
 }  
  
 public void setDescription(String description) {  
 this.description = description;  
 }  
  
 public String getDate() {  
 return date;  
 }  
  
 public void setDate(String date) {  
 this.date = date;  
 }  
  
 public boolean getPriority() {return this.priority; }  
  
 public void setPriority(boolean priority) {  
 this.priority = priority;  
 }  
  
}

Note onclicker

package com.example.rvassessment.Notes;  
  
  
import androidx.cardview.widget.CardView;  
  
public interface NotesClickListener {  
 void onClick(NoteModel notes);  
 void onLongClick(NoteModel notes, CardView cardView);  
}

Main Activity

package com.example.rvassessment;  
  
import androidx.annotation.Nullable;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.cardview.widget.CardView;  
import androidx.recyclerview.widget.LinearLayoutManager;  
import androidx.recyclerview.widget.RecyclerView;  
import androidx.recyclerview.widget.StaggeredGridLayoutManager;  
  
import android.view.MenuItem;  
import android.widget.PopupMenu;  
  
import android.app.Activity;  
import android.content.Intent;  
import android.os.Bundle;  
import android.widget.Button;  
import android.widget.Toast;  
  
import com.example.rvassessment.Notes.NoteModel;  
import com.example.rvassessment.Notes.NotesClickListener;  
import com.example.rvassessment.database.MainDAO;  
import com.example.rvassessment.database.RoomDB;  
  
import java.util.ArrayList;  
import java.util.List;  
  
  
public class MainActivity extends AppCompatActivity {  
 RecyclerView recyclerView;  
 Button newNote;  
 NotesAdaptor adaptor;  
 MainDAO mainDAO;  
  
 List<NoteModel> notes = new ArrayList<>();  
 RoomDB database;  
 NoteModel selectedNote;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 recyclerView = findViewById(R.id.*recyclerNotebook*);  
 newNote = findViewById(R.id.*newNote*);  
 mainDAO = RoomDB.*getAppDatabase*(this).mainDAO();  
 database = RoomDB.*getInstance*(MainActivity.this);  
 notes = database.mainDAO().getAll();  
  
 updateRecycler(notes);  
  
 newNote.setOnClickListener(view -> {  
 Intent intent = new Intent(MainActivity.this, NotePad2.class);  
 startActivityForResult(intent, 101);  
 });  
 }  
  
  
 @Override  
 protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {  
 super.onActivityResult(requestCode, resultCode, data);  
  
 if (requestCode == 101) {  
 if (resultCode == Activity.*RESULT\_OK*) {  
 NoteModel new\_notes = (NoteModel) data.getSerializableExtra("note");  
 database.mainDAO().insert(new\_notes);  
 notes.clear();  
 notes.addAll(database.mainDAO().getAll());  
 }  
 }else if (requestCode == 102) {  
 if (resultCode==Activity.*RESULT\_OK*){  
 NoteModel new\_notes = (NoteModel) data.getSerializableExtra("note");  
 database.mainDAO().update(new\_notes.getID(), new\_notes.getTitle(), new\_notes.getDescription(), new\_notes.priority);  
 notes.clear();  
 notes.addAll(database.mainDAO().getAll());  
 }  
 }  
 updateRecycler(notes);  
 }  
  
 private void updateRecycler(List<NoteModel> notes) {  
 recyclerView.setHasFixedSize(true);  
 recyclerView.setLayoutManager(new StaggeredGridLayoutManager(2, LinearLayoutManager.*VERTICAL*));  
 adaptor = new NotesAdaptor(MainActivity.this, notes, notesClickListener, mainDAO);  
 recyclerView.setAdapter(adaptor);  
 }  
  
 private final NotesClickListener notesClickListener = new NotesClickListener() {  
 @Override  
 public void onClick(NoteModel notes) {  
 Intent intent = new Intent(MainActivity.this, NotePad2.class);  
 intent.putExtra("oldNote", notes);  
 startActivityForResult(intent, 102);  
  
 }  
  
 @Override  
 public void onLongClick(NoteModel notes, CardView cardView) {  
 selectedNote = new NoteModel();  
 selectedNote = notes;  
 showPopup(cardView);  
 }  
 };  
  
 private void showPopup(CardView cardView) {  
 PopupMenu popupMenu = new PopupMenu(MainActivity.this, cardView);  
 popupMenu.inflate(R.menu.*popup\_menu*);  
 popupMenu.setOnMenuItemClickListener(menuItem -> {  
 if (menuItem.getItemId() == R.id.*delete*) {  
 database.mainDAO().delete(selectedNote);  
 notes.remove(selectedNote);  
 notes = mainDAO.getAll();  
 //adaptor.notifyDataSetChanged();  
 Toast.*makeText*(MainActivity.this, "Note Deleted!", Toast.*LENGTH\_SHORT*).show();  
 updateRecycler(notes);  
 }  
 return true;  
 });  
 popupMenu.show();  
 }  
}

NotePad

package com.example.rvassessment;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.app.Activity;  
import android.content.Intent;  
import android.os.Bundle;  
import android.util.Log;  
import android.view.View;  
import android.widget.Button;  
import android.widget.CheckBox;  
import android.widget.EditText;  
import android.widget.Toast;  
  
import com.example.rvassessment.Notes.NoteModel;  
  
import java.text.SimpleDateFormat;  
import java.util.Date;  
  
public class NotePad2 extends AppCompatActivity {  
  
 Button saveNote;  
 EditText newNoteTitle, newNoteDescription;  
 CheckBox newNotePriority;  
 NoteModel notes;  
 boolean oldNote = false;  
  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*notepad*);  
 saveNote = findViewById(R.id.*saveNote*);  
 newNoteTitle = findViewById(R.id.*newNoteTitle*);  
 newNoteDescription = findViewById(R.id.*newNoteDescription*);  
 newNotePriority = findViewById(R.id.*newNotePriority*);  
  
 try {  
 notes = (NoteModel) getIntent().getSerializableExtra("oldNote");  
 newNoteTitle.setText(notes.getTitle());  
 newNoteDescription.setText(notes.getDescription());  
 if (notes.getPriority()) {  
 newNotePriority.setChecked(true);  
 }else {  
 newNotePriority.setChecked(false);  
 }  
 oldNote = true;  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
  
 saveNote.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 String title = newNoteTitle.getText().toString();  
 String description = newNoteDescription.getText().toString();  
  
 if (title.isEmpty()) {  
 Toast.*makeText*(NotePad2.this, "Please add a title", Toast.*LENGTH\_SHORT*).show();  
 return;  
 }  
 if (description.isEmpty()) {  
 Toast.*makeText*(NotePad2.this, "Please add a description", Toast.*LENGTH\_SHORT*).show();  
 return;  
 }  
  
 SimpleDateFormat formatter = new SimpleDateFormat("MM dd, yyyy");  
 Date date = new Date();  
  
 if (!oldNote) {  
 notes = new NoteModel();  
 }  
 notes.setTitle(title);  
 notes.setDescription(description);  
 notes.setDate(formatter.format(date));  
  
 if (newNotePriority.isChecked()) {  
 notes.setPriority(true);  
 newNotePriority.setChecked(true);  
 } else {  
 notes.setPriority(false);  
 newNotePriority.setChecked(false);  
 }  
  
 Intent intent = new Intent();  
 intent.putExtra("note", notes);  
 setResult(Activity.*RESULT\_OK*, intent);  
 finish();  
 }  
 });  
 }  
}

Adaptor

package com.example.rvassessment;  
  
import android.annotation.SuppressLint;  
import android.content.Context;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.CheckBox;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import androidx.annotation.NonNull;  
import androidx.cardview.widget.CardView;  
import androidx.recyclerview.widget.RecyclerView;  
  
  
import com.example.rvassessment.Notes.NoteModel;  
import com.example.rvassessment.Notes.NotesClickListener;  
import com.example.rvassessment.database.MainDAO;  
  
import java.text.SimpleDateFormat;  
import java.util.Date;  
import java.util.List;  
  
public class NotesAdaptor extends RecyclerView.Adapter<NotesAdaptor.ViewHolder> {  
 Context context;  
 List<NoteModel> notebook;  
 NotesClickListener listener;  
 MainDAO mainDAO;  
  
  
 public NotesAdaptor(Context context, List<NoteModel> list, NotesClickListener listener, MainDAO mainDAO) {  
 this.context = context;  
 this.notebook = list;  
 this.listener = listener;  
 this.mainDAO = mainDAO;  
 }  
  
 @Override  
 public NotesAdaptor.ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {  
 return new ViewHolder(LayoutInflater.*from*(context).inflate(R.layout.*noteblock*, parent, false));  
 }  
  
 @Override  
 public void onBindViewHolder(@NonNull NotesAdaptor.ViewHolder holder, @SuppressLint("RecyclerView") int position) {  
 NoteModel note = notebook.get(position);  
 holder.title.setText(notebook.get(position).getTitle());  
 //holder.title.setSelected(true);  
 holder.description.setText(notebook.get(position).getDescription());  
 SimpleDateFormat format = new SimpleDateFormat("MM dd, yyyy");  
 holder.date.setText(format.format(new Date()));  
 holder.priority.setChecked(notebook.get(position).priority);  
  
  
 holder.getPriority().setOnClickListener(view -> {  
 note.setPriority(holder.getPriority().isChecked());  
 note.setDate(String.*valueOf*(new Date()));  
 mainDAO.update(note);  
 notebook = mainDAO.getAll();  
 notifyDataSetChanged();  
 String text;  
 if (note.getPriority()) {  
 text = "high";  
 } else {  
 text = "low";  
 }  
 Toast.*makeText*(view.getContext(),  
 notebook.get(position).getTitle() + " is " + text + " priority",  
 Toast.*LENGTH\_SHORT*).show();  
 });  
  
 holder.itemView.setOnClickListener(view -> listener.onClick(notebook.get(holder.getAdapterPosition())));  
  
 holder.notes\_container.setOnLongClickListener(view -> {  
 listener.onLongClick(notebook.get(holder.getAdapterPosition()), holder.notes\_container);  
 return true;  
 });  
 }  
 @Override  
 public int getItemCount() {  
 return notebook.size();  
 }  
  
 public class ViewHolder extends RecyclerView.ViewHolder {  
 private TextView title;  
 private TextView description;  
 private TextView date;  
 private CheckBox priority;  
 private CardView notes\_container;  
  
 public ViewHolder(@NonNull View view) {  
 super(view);  
 notes\_container = view.findViewById(R.id.*notes\_container*);  
 title = view.findViewById(R.id.*noteTitle*);  
 description = view.findViewById(R.id.*description*);  
 date = view.findViewById(R.id.*date*);  
 priority = view.findViewById(R.id.*priority*);  
 }  
  
 public CheckBox getPriority() {  
 return priority;  
 }  
  
 }  
  
  
  
  
}