**Assigment: 4**

**Name:Muhammad Husnain**

**AridNo: 21-Arid-622**

**Section: BSCS 6B**

**Submitted to: Sir Azhar**

**Create Layout Files**

**1. activity\_main.xml**

This file will include a FragmentContainerView to host the fragments:

xml

Copy code

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".MainActivity">

<androidx.fragment.app.FragmentContainerView

android:id="@+id/fragment\_container\_view"

android:layout\_width="0dp"

android:layout\_height="0dp"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

tools:layout="@layout/fragment\_contacts" />

</androidx.constraintlayout.widget.ConstraintLayout>

**2. fragment\_contacts.xml**

This layout will include a RecyclerView to display the list of contacts:

xml

Copy code

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".ContactsFragment">

<androidx.recyclerview.widget.RecyclerView

android:id="@+id/recycler\_view"

android:layout\_width="0dp"

android:layout\_height="0dp"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

tools:listitem="@layout/item\_contact" />

</androidx.constraintlayout.widget.ConstraintLayout>

**3. item\_contact.xml**

This layout will define the item view in the RecyclerView:

xml

Copy code

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:orientation="vertical"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:padding="16dp">

<TextView

android:id="@+id/contact\_name"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textSize="18sp" />

<TextView

android:id="@+id/contact\_number"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textSize="14sp"

android:textColor="#888888" />

</LinearLayout>

**4. fragment\_contact\_detail.xml**

This layout will show the details of a selected contact:

xml

Copy code

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".ContactDetailFragment">

<TextView

android:id="@+id/contact\_detail\_name"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:textSize="24sp"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

android:padding="16dp"

tools:text="Contact Name" />

<TextView

android:id="@+id/contact\_detail\_number"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:textSize="18sp"

app:layout\_constraintTop\_toBottomOf="@id/contact\_detail\_name"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

android:padding="16dp"

tools:text="Contact Number" />

</androidx.constraintlayout.widget.ConstraintLayout>

**Step 4: Create Java Classes**

**1. MainActivity.java**

This class will set up the initial fragment:

java

Copy code

package com.example.multifragmentapp;

import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

if (savedInstanceState == null) {

getSupportFragmentManager().beginTransaction()

.replace(R.id.fragment\_container\_view, new ContactsFragment())

.commit();

}

}

}

**2. ContactsFragment.java**

This fragment will display a list of contacts using a RecyclerView:

java

Copy code

package com.example.multifragmentapp;

import android.database.Cursor;

import android.os.Bundle;

import android.provider.ContactsContract;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.TextView;

import androidx.annotation.NonNull;

import androidx.annotation.Nullable;

import androidx.fragment.app.Fragment;

import androidx.recyclerview.widget.LinearLayoutManager;

import androidx.recyclerview.widget.RecyclerView;

import java.util.ArrayList;

import java.util.List;

public class ContactsFragment extends Fragment {

private RecyclerView recyclerView;

private ContactsAdapter adapter;

@Nullable

@Override

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

return inflater.inflate(R.layout.fragment\_contacts, container, false);

}

@Override

public void onViewCreated(@NonNull View view, @Nullable Bundle savedInstanceState) {

super.onViewCreated(view, savedInstanceState);

recyclerView = view.findViewById(R.id.recycler\_view);

recyclerView.setLayoutManager(new LinearLayoutManager(getContext()));

List<Contact> contacts = loadContacts();

adapter = new ContactsAdapter(contacts, contact -> {

// Handle item click

Bundle bundle = new Bundle();

bundle.putString("name", contact.getName());

bundle.putString("number", contact.getNumber());

ContactDetailFragment detailFragment = new ContactDetailFragment();

detailFragment.setArguments(bundle);

getParentFragmentManager().beginTransaction()

.replace(R.id.fragment\_container\_view, detailFragment)

.addToBackStack(null)

.commit();

});

recyclerView.setAdapter(adapter);

}

private List<Contact> loadContacts() {

List<Contact> contacts = new ArrayList<>();

Cursor cursor = getActivity().getContentResolver().query(ContactsContract.CommonDataKinds.Phone.CONTENT\_URI, null, null, null, null);

if (cursor != null) {

while (cursor.moveToNext()) {

String name = cursor.getString(cursor.getColumnIndex(ContactsContract.CommonDataKinds.Phone.DISPLAY\_NAME));

String number = cursor.getString(cursor.getColumnIndex(ContactsContract.CommonDataKinds.Phone.NUMBER));

contacts.add(new Contact(name, number));

}

cursor.close();

}

return contacts;

}

}

**3. ContactDetailFragment.java**

This fragment will show the details of the selected contact:

java

Copy code

package com.example.multifragmentapp;

import android.os.Bundle;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.TextView;

import androidx.annotation.NonNull;

import androidx.annotation.Nullable;

import androidx.fragment.app.Fragment;

public class ContactDetailFragment extends Fragment {

@Nullable

@Override

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

return inflater.inflate(R.layout.fragment\_contact\_detail, container, false);

}

@Override

public void onViewCreated(@NonNull View view, @Nullable Bundle savedInstanceState) {

super.onViewCreated(view, savedInstanceState);

TextView nameTextView = view.findViewById(R.id.contact\_detail\_name);

TextView numberTextView = view.findViewById(R.id.contact\_detail\_number);

if (getArguments() != null) {

String name = getArguments().getString("name");

String number = getArguments().getString("number");

nameTextView.setText(name);

numberTextView.setText(number);

}

}

}

**4. ContactsAdapter.java**

This adapter will bind the contacts data to the RecyclerView:

java

Copy code

package com.example.multifragmentapp;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.TextView;

import androidx.annotation.NonNull;

import androidx.recyclerview.widget.RecyclerView;

import java.util.List;

public class ContactsAdapter extends RecyclerView.Adapter<ContactsAdapter.ContactViewHolder> {

private List<Contact> contacts;

private OnItemClickListener listener;

public interface OnItemClickListener {

void onItemClick(Contact contact);

}

public ContactsAdapter(List<Contact> contacts, OnItemClickListener listener) {

this.contacts = contacts;

this.listener = listener;

}

@NonNull

@Override

public ContactViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {

View view = LayoutInflater.from(parent.getContext()).inflate(R.layout.item\_contact, parent, false);

return new ContactViewHolder(view);

}

@Override

public void onBindViewHolder(@NonNull ContactViewHolder holder, int position) {

Contact contact = contacts.get(position);

holder.bind(contact, listener);

}

@Override

public int getItemCount() {

return contacts.size();

}

public static class ContactViewHolder extends RecyclerView.ViewHolder {

private TextView nameTextView;

private TextView numberTextView;

public ContactViewHolder(@NonNull View itemView) {

super(itemView);

nameTextView = itemView.findViewById(R.id.contact\_name);

numberTextView = itemView.findViewById(R.id.contact\_number);

}

public void bind(Contact contact, OnItemClickListener listener) {

nameTextView.setText(contact.getName());

numberTextView.setText(contact.getNumber());

itemView.setOnClickListener(v -> listener.onItemClick(contact));

}

}

}

**5. Contact.java**

This is a simple model class for contacts:

java

Copy code

package com.example.multifragmentapp;

public class Contact {

private String name;

private String number;

public Contact(String name, String number) {

this.name = name;

this.number = number;

}

public String getName() {

return name;

}

public String getNumber() {

return number;

}

}