

activity_start.xml

```
<?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=".start"

    android:background="@drawable/appstart"
">

</androidx.constraintlayout.widget.ConstraintLayout>
```

start.java

```
package
com.example.callingapp_pract_6;

import
androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;

public class start extends
AppCompatActivity {

    @Override
    protected void onCreate(Bundle
savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity_start
);
    }
    @Override
    protected void onResume() {
        super.onResume();

        new Handler().postDelayed(new
Runnable() {
            @Override
            public void run() {
                Intent intent = new
Intent(start.this,
MainActivity.class);
                startActivity(intent);
                finish();
            }
        }, 2000); // delay for 3
minutes (3 * 60 seconds * 1000
milliseconds)
    }
}
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.
com/apk/res/android"

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

    xmlns:tools="http://schemas.android.co
m/tools"

    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_alignParentStart="true"
    android:layout_alignParentTop="true"
        android:visibility="visible"
        tools:context=".MainActivity">

        <ViewSwitcher
            android:id="@+id/viewSwitcher"

            android:layout_width="match_parent"
            android:layout_height="match_parent">

            <!-- Layout for normal display -
            -->
            <LinearLayout

                android:id="@+id/normal_view"

                android:layout_width="match_parent"
                    android:layout_height="50dp"

                android:layout_above="@id/scrollView"
                android:layout_alignParentTop="true"
                android:layout_marginTop="2dp"
                android:layout_marginBottom="-2dp"
                android:background="@drawable/layout_s
hape"
                    android:minHeight="50dp"

                android:orientation="horizontal">

                <TextView

                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_marginStart="20dp"
                    android:layout_marginTop="5dp"
                        android:text="Contacts"

                    android:textColor="#000000"
                        android:textSize="22dp"
                        android:textStyle="bold"
                />

                <ImageView
```

```
                    android:id="@+id/search"

                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_marginStart="130dp"
                    android:layout_marginTop="10dp"
                    android:src="@drawable/search" />

                <ImageView
                    android:id="@+id/more"

                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_marginStart="5dp"
                    android:layout_marginTop="10dp"
                    android:src="@drawable/more" />
                </LinearLayout>
                <LinearLayout
                    android:id="@+id/searching"

                    android:layout_width="match_parent"
                        android:layout_height="50dp"

                    android:layout_above="@id/scrollView"
                    android:layout_alignParentTop="true"
                    android:layout_marginTop="2dp"
                    android:layout_marginBottom="-2dp"
                    android:background="@drawable/layout_s
hape"
                        android:minHeight="50dp"

                    android:orientation="horizontal">
                    <EditText
                        android:id="@+id/edit"

                        android:layout_width="233dp"
                        android:layout_height="wrap_content"
                        android:layout_marginStart="15dp"
                            android:hint="Search
                            here"

                        android:textColor="#000000"
                        android:textColorHint="#322F2F" />
                    <TextView
                        android:id="@+id/cancel"

                        android:layout_width="wrap_content"
                        android:layout_height="wrap_content"
                            android:text="cancel"
                            android:textSize="16dp"
                            android:textStyle="bold"
                    />
```

```
</LinearLayout>
</ViewSwitcher>

<ScrollView
    android:id="@+id/scrollView"
    android:layout_width="match_parent"
    android:layout_height="590dp"

    android:layout_alignParentStart="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentTop="false"
    android:layout_alignParentEnd="true"
    android:layout_alignParentRight="true"
    android:layout_alignParentBottom="true"
    "
        android:layout_marginTop="20dp"
        android:minHeight="500dp"
        tools:ignore="InvalidId">

<androidx.recyclerview.widget.RecyclerView
View
    android:id="@+id/recycler_view"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginStart="10dp"

    tools:listitem="@layout/item_contact"
/>
</ScrollView>
</RelativeLayout>
```

MainActivity.java

```
package
com.example.callingapp_pract_6;

import static
androidx.core.content.ContextCompat.startActivity;

import
android.annotation.SuppressLint;

import android.content.Intent;
import
android.content.pm.PackageManager;
import android.database.Cursor;
import android.net.Uri;
import android.os.Bundle;
import
android.provider.ContactsContract;
import android.text.Editable;
import android.text.TextWatcher;
import android.view.View;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.LinearLayout;
import android.widget.TextView;
import android.widget.Toast;

import androidx.annotation.NonNull;
import
androidx.appcompat.app.AppCompatActivity;
import
androidx.core.app.ActivityCompat;
import
androidx.core.content.ContextCompat;
import
androidx.recyclerview.widget.LinearLayoutManager;
import
androidx.recyclerview.widget.RecyclerView;

import java.util.ArrayList;
import android.Manifest;
import android.widget.ViewSwitcher;

public class MainActivity extends
AppCompatActivity {

    RecyclerView recyclerView;
    LinearLayout normal , search;
    ViewSwitcher viewSwitcher;
    ImageView imageView;
    TextView c;
    EditText e;
    ArrayList<ContactModel> arrayList
= new ArrayList<ContactModel>();

    MainAdapter adapter;
    private static final int
REQUEST_CALL = 1;

    @SuppressWarnings("MissingInflatedId")
    @Override
    protected void onCreate(Bundle

savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main)
        ;

        recyclerView =
findViewById(R.id.recycler_view);
        imageView =
findViewById(R.id.search);
        viewSwitcher =
findViewById(R.id.viewSwitcher);
        normal =
findViewById(R.id.normal_view);
        c = findViewById(R.id.cancel);
        search =
findViewById(R.id.searching);
        e = findViewById(R.id.edit);
        checkPermission();

        viewSwitcher.setDisplayedChild(0);

        if(ContextCompat.checkSelfPermission(M
ainActivity.this,

Manifest.permission.READ_PHONE_STATE)
!= PackageManager.PERMISSION_GRANTED) {

            if(ActivityCompat.shouldShowRequestPer
missionRationale(MainActivity.this,Man
ifest.permission.READ_PHONE_STATE)){

                ActivityCompat.requestPermissions(Main
Activity.this,new String[]
{Manifest.permission.READ_PHONE_STATE}
,1);

            }else{

                ActivityCompat.requestPermissions(Main
Activity.this,new String[]
{Manifest.permission.READ_PHONE_STATE}
,1);

            }
        }else{

        }

        imageView.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View
view) {

                Toast.makeText(MainActivity.this,
"clicked", Toast.LENGTH_SHORT).show();

                viewSwitcher.setDisplayedChild(1);

                // adapter = new
MainAdapter();
            }
        });

        c.setOnClickListener(new
View.OnClickListener() {
```

```

        @Override
        public void onClick(View
view) {

viewSwitcher.setDisplayedChild(0);
    }
    });

    e.addTextChangedListener(new
TextWatcher() {
        @Override
        public void
beforeTextChanged(CharSequence s, int
start, int count, int after) {
            // Not used
        }

        @Override
        public void
onTextChanged(CharSequence s, int
start, int before, int count) {
            // Call the filter
method on the adapter
            if (s.length() == 0) {
                // Show the normal
list if the search box is empty

adapter.setArrayList();
            } else {
                // Filter the
adapter based on the search query

adapter.filter(s.toString());
            }
        }
        @Override
        public void
afterTextChanged(Editable s) {
            // Not used
        }
    });

}

@Override
public void
onRequestPermissionsResult(int
requestCode, @NonNull String[]
permissions, @NonNull int[]
grantResults) {

super.onRequestPermissionsResult(reque
stCode, permissions, grantResults);
    if (requestCode ==
REQUEST_CALL) {
        if (grantResults.length >
0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {

            } else {
                Toast.makeText(this,
"PERMISSION DENIED",

```

```

Toast.LENGTH_SHORT).show();
            }
        }
        if(requestCode == 1) {
            if (grantResults.length >
0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
                if
(ContextCompat.checkSelfPermission(Mai
nActivity.this,
Manifest.permission.READ_PHONE_STATE)
== PackageManager.PERMISSION_GRANTED)
{

                } else {

Toast.makeText(this, "PERMISSION
DENIED", Toast.LENGTH_SHORT).show();
                }
            }
        }

        private void checkPermission(){

if(ContextCompat.checkSelfPermission(M
ainActivity.this,Manifest.permission.R
EAD_CONTACTS) !=
PackageManager.PERMISSION_GRANTED) {

ActivityCompat.requestPermissions(Main
Activity.this,

                new
String[]{Manifest.permission.READ_CONT
ACTS},

                100);
            }else{
                getContactList();
                adapter = new
MainAdapter(this, arrayList);

recyclerView.setAdapter(adapter);

recyclerView.setLayoutManager(new
LinearLayoutManager(this));
            }

            private void getContactList() {
                Uri uri =
ContactsContract.Contacts.CONTENT_URI;
                String sort =
ContactsContract.CommonDataKinds.Phone
.DISPLAY_NAME + " ASC";

                try (Cursor cursor =
getContentResolver().query(uri, null,
null, null, sort)) {
                    if (cursor != null &&
cursor.getCount() > 0) {
                        while
(cursor.moveToNext()) {

@SuppressLint("Range") String id =
cursor.getString(cursor.getColumnIndex
(ContactsContract.Contacts._ID));

@SuppressLint("Range") String name =

```

```

cursor.getString(cursor.getColumnIndex
(ContactsContract.Contacts.DISPLAY_NAM
E));

        Uri uriPhone =
ContactsContract.CommonDataKinds.Phone
.CONTENT_URI;

        String selection =
ContactsContract.CommonDataKinds.Phone
.CONTACT_ID + "=?";

        try (Cursor
phoneCursor =
getContentResolver().query(uriPhone,
null, selection, new String[]{id},
null)) {

            while
(phoneCursor != null &&
phoneCursor.moveToNext()) {

                @SuppressWarnings("Range") String number =
phoneCursor.getString(phoneCursor.getCo
lumnIndex(ContactsContract.CommonData
Kinds.Phone.NUMBER));

                ContactModel model = new
ContactModel();

                model.setName(name);

                model.setNumber(number);

                arrayList.add(model);
            }
        }
    } catch (Exception e) {
        e.printStackTrace();
    }

    //
recyclerView.setLayoutManager(new
LinearLayoutManager(this));
    adapter = new
MainAdapter(this, arrayList);

    recyclerView.setAdapter(adapter);

    recyclerView.setLayoutManager(new
LinearLayoutManager(this));

    adapter.setOnItemClickListener(new
MainAdapter.OnItemClickListener() {
        @Override
        public void
onItemClick(ContactModel contact) {

            Toast.makeText(MainActivity.this,
contact.getName().toString(),
Toast.LENGTH_SHORT).show();

            if(ContextCompat.checkSelfPermission(M
ainActivity.this,
Manifest.permission.CALL_PHONE) !=
PackageManager.PERMISSION_GRANTED) {

```

```

ActivityCompat.requestPermissions(Main
Activity.this, new String[]
{Manifest.permission.CALL_PHONE},
REQUEST_CALL);
        //
            }else{
                String phone =
contact.getNumber();
                Intent i = new
Intent(Intent.ACTION_CALL);

                i.setData(Uri.parse("tel:"+
phone));
                startActivity(i);
            }
        }
    });
}
}

```

item_contact.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"

    android:layout_width="match_parent"

    android:layout_height="wrap_content"

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

    <ImageView
        android:id="@+id/image"
        android:layout_width="80dp"
        android:layout_height="80dp"
        android:src="@drawable/person"

        app:layout_constraintBottom_toBottomOf="parent"

        app:layout_constraintStart_toStartOf="parent"

        app:layout_constraintTop_toTopOf="parent" />

    <LinearLayout
        android:layout_width="0dp"

        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:layout_marginEnd="8dp"

        android:layout_marginStart="10dp"

        app:layout_constraintStart_toEndOf="@id/image"

        app:layout_constraintTop_toTopOf="parent"

        app:layout_constraintBottom_toBottomOf="parent"

        app:layout_constraintEnd_toEndOf="parent">

        <TextView
            android:layout_width="match_parent"

            android:layout_height="wrap_content"
            android:id="@+id/name"
            android:textSize="18dp"

            android:textColor="@color/black"
            android:paddingTop="4dp"
            android:textStyle="bold"
            android:text="name"

            android:paddingBottom="8dp" />

        <TextView
            android:layout_width="match_parent"

            android:layout_height="wrap_content"
```

```
        android:id="@+id/number"
        android:text="9999999999"
        android:textSize="14dp"

        android:textColor="@color/black" />
    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```


MainAdapter.java

```
package
com.example.callingapp_pract_6;

import
android.annotation.SuppressLint;
import android.app.Activity;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import
android.widget.AdapterView.OnItemClickListener;
Listener;
import android.widget.TextView;

import androidx.annotation.NonNull;
import
androidx.recyclerview.widget.RecyclerView;

import java.util.ArrayList;

public class MainAdapter extends
RecyclerView.Adapter<MainAdapter.viewH
older> {
    Activity activity;
    ArrayList<ContactModel> arrayList;
    ArrayList<ContactModel>
arrayList2;
    int selectedItem = -1;
    private static OnItemClickListener
listener;

    public void filter(String query) {
        ArrayList<ContactModel>
filteredList = new ArrayList<>();

        for (ContactModel contact :
arrayList) {
            if
(contact.getName().toLowerCase().conta
ins(query.toLowerCase())) {

                filteredList.add(contact);
            }
        }

        arrayList = filteredList;
        notifyDataSetChanged();
    }

    @SuppressWarnings("NotifyDataSetChanged")
    public void setArrayList() {
        arrayList = new
ArrayList<>(arrayList2);
        notifyDataSetChanged();
    }

    public interface
OnItemClickListener {
        void onItemClick(ContactModel
contact);
    }

    public void
setOnItemClickListener(OnItemClickListener
listener) {
        MainAdapter.listener =
```

```
listener;
    }

    @SuppressWarnings("NotifyDataSetChanged")
    public MainAdapter(Activity
activity, ArrayList<ContactModel>
arrayList) {
        this.activity = activity;
        this.arrayList = arrayList;
        this.arrayList2 = new
ArrayList<>(arrayList);
        notifyDataSetChanged();

    }

    @NonNull
    @Override
    public ViewHolder
onCreateViewHolder(@NonNull ViewGroup
parent, int viewType) {
        View view =
LayoutInflater.from(parent.getContext(
)).inflate(R.layout.item_contact, paren
t, false);
        return new ViewHolder(view);
    }

    @SuppressWarnings("ResourceAsColor")
    @Override
    public void
onBindViewHolder(@NonNull ViewHolder
holder, @SuppressWarnings("RecyclerView")
int position) {
        ContactModel model =
arrayList.get(position);

        holder.t_name.setText(model.getName())
;

        holder.t_number.setText(model.getNumb
er());

        holder.itemView.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View
view) {
                // int p =
arrayList.get(position);
                if (position !=
RecyclerView.NO_POSITION && listener
!= null) {

                    listener.onItemClick(arrayList.get(pos
ition));

                }
            }
        });
    }

    @Override
    public int getItemCount() {
        return arrayList.size();
    }
}
```

```

        public class viewHolder extends
RecyclerView.ViewHolder {
            TextView t_name , t_number;

@SuppressLint("ResourceAsColor")
        public viewHolder(@NonNull
View itemView) {
            super(itemView);
            t_name =
itemView.findViewById(R.id.name);
            t_number =
itemView.findViewById(R.id.number);
        }
    }
}

```

ContactModel.java

```

package
com.example.callingapp_pract_6;

public class ContactModel {
    String name, number;

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getNumber() {
        return number;
    }

    public void setNumber(String
number) {
        this.number = number;
    }
}

```

Call.java

```
package
com.example.callingapp_pract_6;

import static
androidx.core.content.ContextCompat.ge
tSystemService;

import
android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import
android.content.pm.PackageManager;
import android.os.Build;
import android.telecom.TelecomManager;
import
android.telephony.TelephonyManager;
import android.view.KeyEvent;
import android.view.WindowManager;
import android.widget.PopupWindow;
import android.widget.Toast;

import
androidx.core.app.ActivityCompat;

public class Call extends
BroadcastReceiver {

    private WindowManager
windowManager;
    private PopupWindow popupWindow;

    @Override
    public void onReceive(Context
context, Intent intent) {
        if
(intent.getAction().equals(TelephonyMa
nager.ACTION_PHONE_STATE_CHANGED)) {
            String state =
intent.getStringExtra(TelephonyManager
.EXTRA_STATE);
            if
(state.equals(TelephonyManager.CALL_ST
ATE_RINGING)) {

                Toast.makeText(context, "Ringing",
                Toast.LENGTH_SHORT).show();
            } else if
(state.equals(TelephonyManager.CALL_ST
ATE_IDLE)) {

                Toast.makeText(context,
                "Disconnected",
                Toast.LENGTH_SHORT).show();
            } else if
(state.equals(TelephonyManager.CALL_ST
ATE_OFFHOOK)) {

                Toast.makeText(context, "Answered",
                Toast.LENGTH_SHORT).show();
            }
        }
    }
}
```

OUTPUT

