Program 6b:

Activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
  
 <Button  
 android:id="@+id/button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/button"  
 tools:layout\_editor\_absoluteX="33dp"  
 tools:layout\_editor\_absoluteY="40dp"  
 tools:ignore="MissingConstraints" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

Main\_activity.java

package com.example.sixthbapp;  
  
import android.Manifest;  
import android.annotation.SuppressLint;  
import android.app.Activity;  
import android.content.Intent;  
import android.database.Cursor;  
import android.net.Uri;  
import android.os.Bundle;  
import android.provider.ContactsContract;  
import android.widget.Button;  
import android.widget.Toast;  
  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.app.ActivityCompat;  
  
@SuppressWarnings("ALL")  
public class MainActivity extends AppCompatActivity {  
  
 public static final int *RequestPermissionCode* = 1 ;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 Button button = findViewById(R.id.*button*);  
 EnableRuntimePermission();  
 button.setOnClickListener(v -> {  
 Intent intent = new Intent(Intent.*ACTION\_PICK*, ContactsContract.Contacts.*CONTENT\_URI*);  
 startActivityForResult(intent, 1);  
 });  
 }  
  
 public void EnableRuntimePermission(){  
  
 if (ActivityCompat.*shouldShowRequestPermissionRationale*(MainActivity.this, Manifest.permission.*READ\_CONTACTS*))  
 {  
 Toast.*makeText*(MainActivity.this,"CONTACTS permission allows us to Access CONTACTS app", Toast.*LENGTH\_LONG*).show();  
 } else {  
 ActivityCompat.*requestPermissions*(MainActivity.this,new String[]{ Manifest.permission.*READ\_CONTACTS*}, *RequestPermissionCode*);  
 }  
 }  
  
 @Override  
 public void onActivityResult(int reqCode, int resultCode, Intent data){  
 super.onActivityResult(reqCode, resultCode, data);  
 if (reqCode == 1) {  
 if (resultCode == Activity.*RESULT\_OK*) {  
 Uri contactData = data.getData();  
 try (Cursor c = managedQuery(contactData, null, null, null, null)) {  
 if (c.moveToFirst()) {  
 String id = c.getString(c.getColumnIndexOrThrow(ContactsContract.Contacts.*\_ID*));  
 @SuppressLint("Range")  
 String hasPhone = c.getString(c.getColumnIndex(ContactsContract.Contacts.*HAS\_PHONE\_NUMBER*));  
 if (hasPhone.equalsIgnoreCase("1")) {  
 Cursor phones = getContentResolver().query(ContactsContract.CommonDataKinds.Phone.*CONTENT\_URI*, null, ContactsContract.CommonDataKinds.Phone.*CONTACT\_ID* + " = " + id, null, null);  
 phones.moveToFirst();  
 @SuppressLint("Range")  
 String cNumber = phones.getString(phones.getColumnIndex(ContactsContract.CommonDataKinds.Phone.*NUMBER*));  
 Toast.*makeText*(getApplicationContext(), cNumber, Toast.*LENGTH\_SHORT*).show();  
 phones.close();  
 }  
 }  
 }  
 catch(Exception e)  
 {  
 e.printStackTrace();  
 }  
 }  
 }  
 }  
}