МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

СХІДНОУКРАЇНСЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ

імені ВОЛОДИМИРА ДАЛЯ

Звіт

з лабораторної роботи № 3,4

з теми: «**Разработка ANDROID-приложения с несколькими активностями, МЕНЮ И СУБД SQLite**»

Виконав студент:

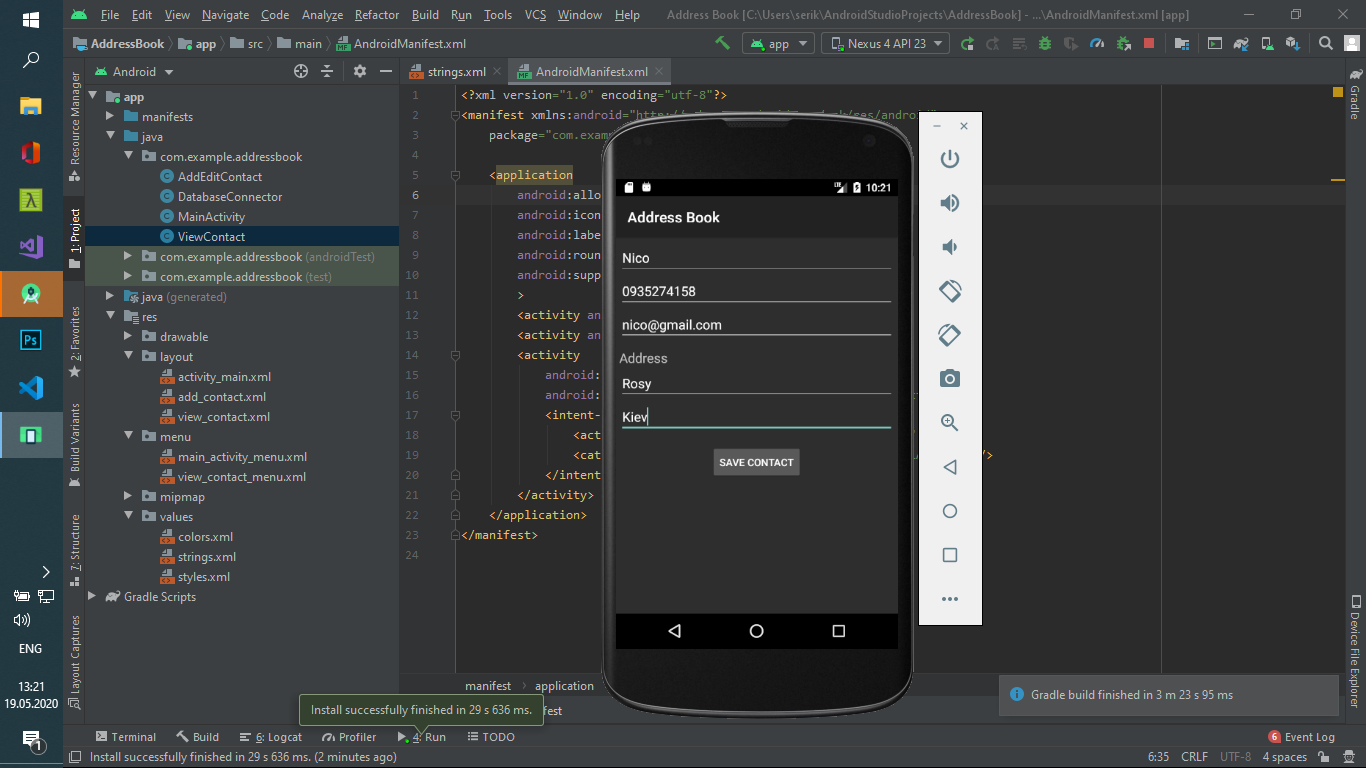
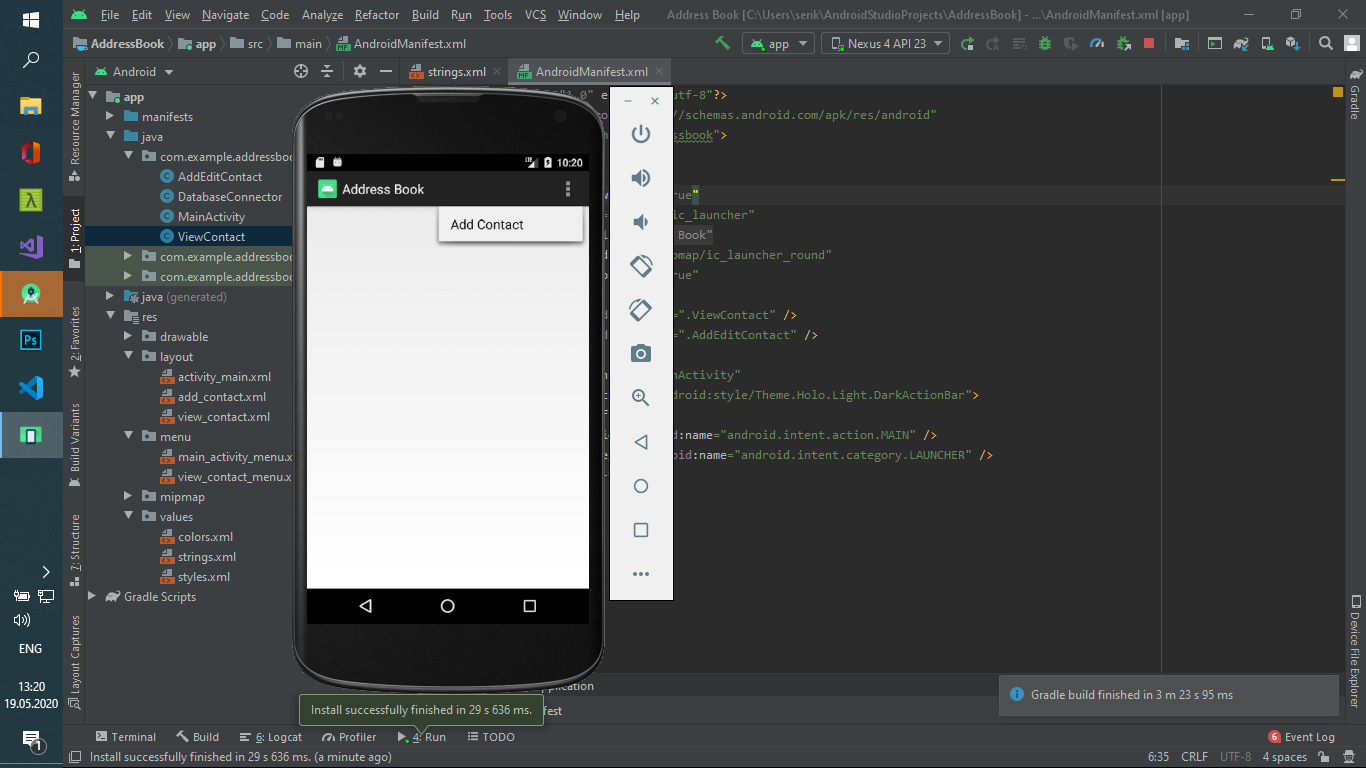
Група КН-18д

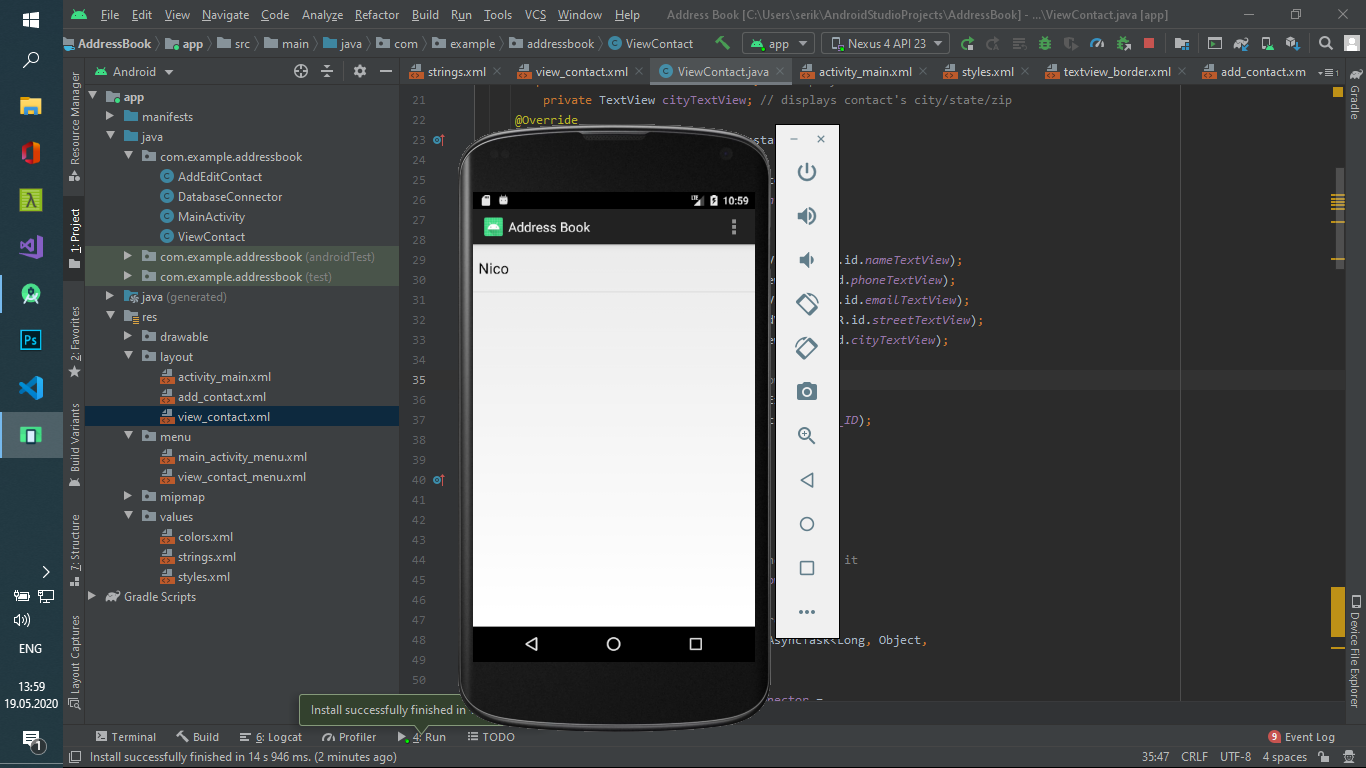
Скороход С.Г.

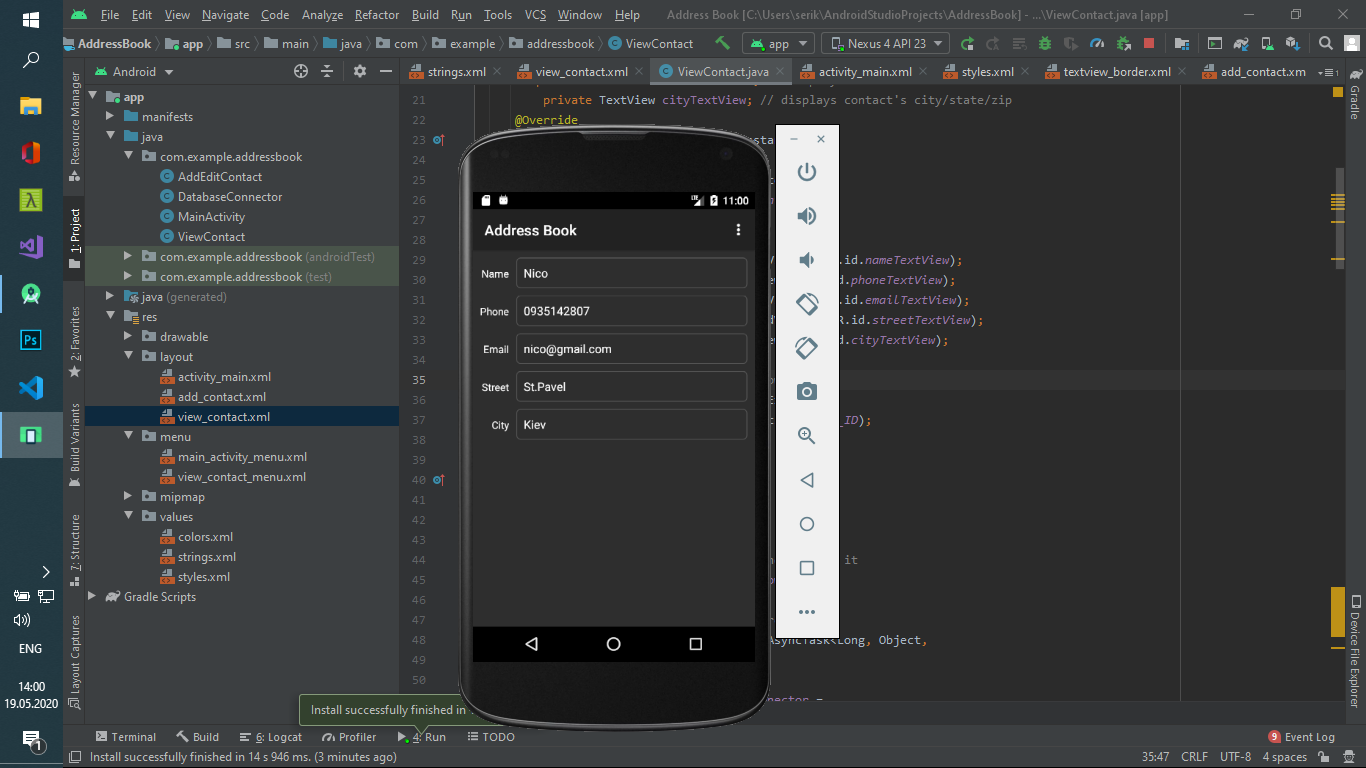
Дата 10.05.2020

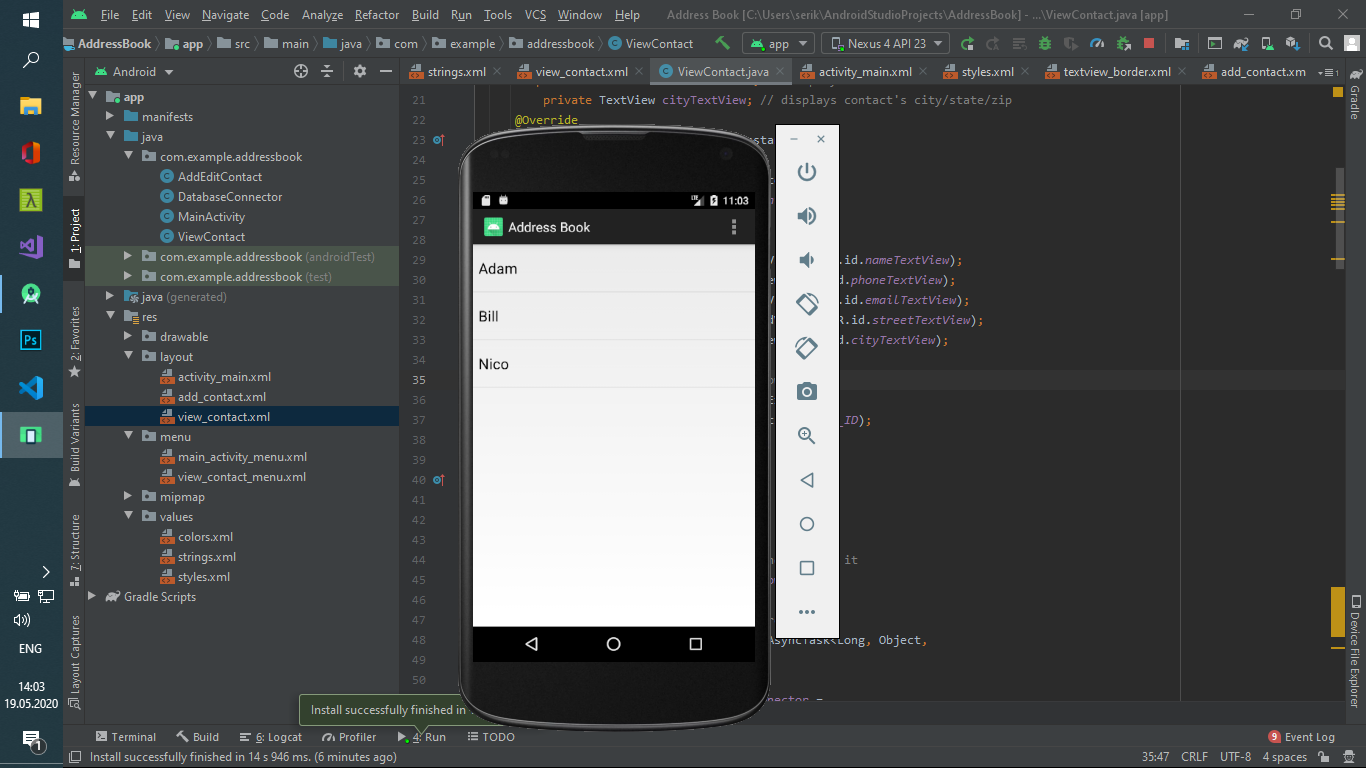
**м. Северодонецьк**

2020

****

****

****

****

**AndroidManifest.xml**

<?xml version="1.0" encoding="utf-8"?>  
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="com.example.addressbook">  
  
 <application  
 android:allowBackup="true"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:roundIcon="@mipmap/ic\_launcher\_round"  
 android:supportsRtl="true"  
 >  
 <activity android:name=".ViewContact" />  
 <activity android:name=".AddEditContact" />  
 <activity  
 android:name=".MainActivity"  
 android:theme="@android:style/Theme.Holo.Light.DarkActionBar">  
 <intent-filter>  
 <action android:name="android.intent.action.MAIN" />  
 <category android:name="android.intent.category.LAUNCHER" />  
 </intent-filter>  
 </activity>  
 </application>  
</manifest>

**styles.xml**

<resources>  
 <style name="ContactLabelTextView">  
 <item name="android:layout\_width">wrap\_content</item>  
 <item name="android:layout\_height">wrap\_content</item>  
 <item name="android:gravity">right</item>  
 <item name="android:textSize">14sp</item>  
 <item name="android:textColor">@android:color/white</item>  
 <item name="android:layout\_marginLeft">5dp</item>  
 <item name="android:layout\_marginRight">5dp</item>  
 <item name="android:layout\_marginTop">5dp</item>  
 </style>  
 <style name="ContactTextView">  
 <item name="android:layout\_width">wrap\_content</item>  
 <item name="android:layout\_height">wrap\_content</item>  
 <item name="android:textSize">16sp</item>  
 <item name="android:textColor">@android:color/white</item>  
 <item name="android:layout\_margin">5dp</item>  
 <item name="android:background">@drawable/textview\_border</item>  
 </style>  
</resources>

**strings.xml**

<resources>  
 <string name="app\_name">Address Book</string>  
 <string name="label\_email">Email</string>  
 <string name="label\_name">Name</string>  
 <string name="label\_phone">Phone</string>  
 <string name="label\_street">Street</string>  
 <string name="label\_city">City</string>  
 <string name="hint\_name">Name</string>  
 <string name="hint\_phone">Phone</string>  
 <string name="hint\_email">Email</string>  
 <string name="address">Address</string>  
 <string name="hint\_street">Street</string>  
 <string name="hint\_city">City</string>  
 <string name="button\_save\_contact">Save Contact</string>  
 <string name="menuitem\_add\_contact">Add Contact</string>  
 <string name="confirmTitle" />  
 <string name="confirmMessage" />  
 <string name="Delete">Delete</string>  
 <string name="button\_delete" >Delete</string>  
 <string name="button\_cancel" >Cancel</string>  
 <string name="errorTitle">Error</string>  
 <string name="errorMessage">Error</string>  
 <string name="errorButton" />  
 <string name="menuitem\_edit\_contact">Edit Contect</string>  
 <string name="menuitem\_delete\_contact">Delete Contact</string>  
</resources>

**view\_contact\_menu.xml**

<?xml version="1.0" encoding="utf-8"?>  
 <menu xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_height="40dp"  
 android:layout\_width="40dp">  
 <item android:id="@+id/editItem"  
 android:title="@string/menuitem\_edit\_contact"  
 android:orderInCategory="1" android:alphabeticShortcut="e"  
 android:titleCondensed="@string/menuitem\_edit\_contact"  
 android:icon="@android:drawable/ic\_menu\_edit"></item>  
 <item android:id="@+id/deleteItem"  
 android:title="@string/menuitem\_delete\_contact"  
 android:orderInCategory="2" android:alphabeticShortcut="d"  
 android:titleCondensed="@string/menuitem\_delete\_contact"  
 android:icon="@android:drawable/ic\_delete"></item>  
 </menu>

**main\_activity\_menu.xml**

<?xml version="1.0" encoding="utf-8"?>  
<menu xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_height="40dp"  
 android:layout\_width="40dp">  
 <item android:id="@+id/addContactItem"  
 android:title="@string/menuitem\_add\_contact"  
 android:icon="@android:drawable/ic\_menu\_add"  
 android:titleCondensed="@string/menuitem\_add\_contact"  
 android:alphabeticShortcut="e">  
  
 </item>  
</menu>

**view\_contact.xml**

<?xml version="1.0" encoding="utf-8"?>  
 <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content" android:layout\_weight="1">  
 <TableLayout android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:stretchColumns="1" android:layout\_margin="5dp">  
 <TableRow android:id="@+id/nameTableRow"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content">  
 <TextView android:id="@+id/nameLabelTextView"  
 style="@style/ContactLabelTextView"  
 android:text="@string/label\_name">  
  
 </TextView>  
 <TextView android:id="@+id/nameTextView"  
 style="@style/ContactTextView">  
  
 </TextView>  
 </TableRow>  
  
 <TableRow android:id="@+id/phoneTableRow"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content">  
 <TextView android:id="@+id/phoneLabelTextView"  
 style="@style/ContactLabelTextView"  
 android:text="@string/label\_phone">  
  
 </TextView>  
 <TextView android:id="@+id/phoneTextView"  
 style="@style/ContactTextView">  
  
 </TextView>  
 </TableRow>  
  
 <TableRow android:id="@+id/emailTableRow"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content">  
  
 <TextView android:id="@+id/emailLabelTextViews"  
 style="@style/ContactLabelTextView"  
 android:text="@string/label\_email">  
 </TextView>  
 <TextView android:id="@+id/emailTextView"  
 style="@style/ContactTextView">  
 </TextView>  
 </TableRow>  
  
 <TableRow android:id="@+id/streetTableRow"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content">  
 <TextView android:id="@+id/streetLabelTextView"  
 style="@style/ContactLabelTextView"  
 android:text="@string/label\_street">  
  
 </TextView>  
 <TextView android:id="@+id/streetTextView"  
 style="@style/ContactTextView">  
  
 </TextView>  
 </TableRow>  
  
 <TableRow android:id="@+id/cityTableRow"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content">  
 <TextView android:id="@+id/cityLabelTextView"  
 style="@style/ContactLabelTextView"  
 android:text="@string/label\_city">  
  
 </TextView>  
 <TextView android:id="@+id/cityTextView"  
 style="@style/ContactTextView">  
  
 </TextView>  
 </TableRow>  
 </TableLayout>  
 </ScrollView>

**add\_contact.xml**

<?xml version="1.0" encoding="utf-8"?>  
 <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content" android:layout\_weight="1">  
 <LinearLayout android:id="@+id/linearLayout"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical" android:padding="5dp">  
 <EditText android:id="@+id/nameEditText"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:imeOptions="actionNext" android:hint="@string/hint\_name"  
 android:inputType="textPersonName|textCapWords"></EditText>  
 <EditText android:id="@+id/phoneEditText"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:imeOptions="actionNext" android:hint="@string/hint\_phone"  
 android:inputType="phone"></EditText>  
 <EditText android:id="@+id/emailEditText"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:imeOptions="actionNext" android:hint="@string/hint\_email"  
 android:inputType="textEmailAddress"></EditText>  
 <TextView android:id="@+id/addressTextView"  
 android:color="@android:color/black" android:textSize="18sp"  
 android:paddingTop="10dp" android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:imeOptions="actionNext"  
 android:text="@string/address"></TextView>  
 <EditText android:id="@+id/streetEditText"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:imeOptions="actionNext" android:hint="@string/hint\_street"  
 android:inputType="textPostalAddress|textCapWords"></EditText>  
 <EditText android:id="@+id/cityEditText"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:imeOptions="actionDone" android:hint="@string/hint\_city"  
 android:inputType="textPostalAddress|textCapWords"></EditText>  
 <Button android:id="@+id/saveContactButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="15dp"  
 android:layout\_gravity="center\_horizontal"  
 android:text="@string/button\_save\_contact"></Button>  
 </LinearLayout>  
</ScrollView>

**activity\_main.xml**

<?xml version="1.0" encoding="utf-8"?>  
 <TextView xmlns:android="http://schemas.android.com/apk/res/android"  
 android:id="@+id/contactTextView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content" android:padding="8dp"  
 android:textSize="20sp" android:textColor="@android:color/black"  
 android:minHeight="?android:attr/listPreferredItemHeight"  
 android:gravity="center\_vertical">  
 </TextView>

**textview\_border.xml**

<?xml version="1.0" encoding="utf-8"?>  
 <shape xmlns:android="http://schemas.android.com/apk/res/android"  
 android:shape="rectangle" >  
 <corners android:radius="5dp"/>  
 <stroke android:width="1dp" android:color="#555"/>  
 <padding android:top="10dp" android:left="10dp" android:bottom="10dp" android:right="10dp"/>  
 </shape>

**ViewContact.java**

package com.example.addressbook;  
import android.app.Activity;  
import android.app.AlertDialog;  
import android.content.DialogInterface;  
import android.content.Intent;  
import android.database.Cursor;  
import android.os.AsyncTask;  
import android.os.Bundle;  
import android.view.Menu;  
import android.view.MenuInflater;  
import android.view.MenuItem;  
import android.widget.TextView;  
  
public class ViewContact extends Activity  
 {  
 private long rowID; // selected contact's name  
 private TextView nameTextView; // displays contact's name  
 private TextView phoneTextView; // displays contact's phone  
 private TextView emailTextView; // displays contact's email  
 private TextView streetTextView; // displays contact's street  
 private TextView cityTextView; // displays contact's city/state/zip  
 @Override  
 public void onCreate(Bundle savedInstanceState)  
 {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*view\_contact*);  
  
 // get the EditTexts  
 nameTextView = (TextView) findViewById(R.id.*nameTextView*);  
 phoneTextView = (TextView) findViewById(R.id.*phoneTextView*);  
 emailTextView = (TextView) findViewById(R.id.*emailTextView*);  
 streetTextView = (TextView) findViewById(R.id.*streetTextView*);  
 cityTextView = (TextView) findViewById(R.id.*cityTextView*);  
  
 // get the selected contact's row ID  
 Bundle extras = getIntent().getExtras();  
 rowID = extras.getLong(MainActivity.*ROW\_ID*);  
 } // end method onCreate  
 @Override  
 protected void onResume()  
 {  
 super.onResume();  
  
 // create new LoadContactTask and execute it  
 new LoadContactTask().execute(rowID);  
 } // end method onResume  
// performs database query outside GUI thread  
 private class LoadContactTask extends AsyncTask<Long, Object,  
 Cursor>  
 {  
 DatabaseConnector databaseConnector =  
 new DatabaseConnector(ViewContact.this);  
  
 // perform the database access  
 @Override  
 protected Cursor doInBackground(Long... params)  
 {  
 databaseConnector.open();  
  
 // get a cursor containing all data on given entry  
 return databaseConnector.getOneContact(params[0]);  
 } // end method doInBackground  
 @Override  
 protected void onPostExecute(Cursor result)  
 {  
 super.onPostExecute(result);  
  
 result.moveToFirst(); // move to the first item  
  
 // get the column index for each data item  
 int nameIndex = result.getColumnIndex("name");  
 int phoneIndex = result.getColumnIndex("phone");  
 int emailIndex = result.getColumnIndex("email");  
 int streetIndex = result.getColumnIndex("street");  
 int cityIndex = result.getColumnIndex("city");  
  
 // fill TextViews with the retrieved data  
 nameTextView.setText(result.getString(nameIndex));  
 phoneTextView.setText(result.getString(phoneIndex));  
 emailTextView.setText(result.getString(emailIndex));  
 streetTextView.setText(result.getString(streetIndex));  
 cityTextView.setText(result.getString(cityIndex));  
 result.close(); // close the result cursor  
 databaseConnector.close(); // close database connection  
 } // end method onPostExecute  
 } // end class LoadContactTask  
 @Override  
 public boolean onCreateOptionsMenu(Menu menu)  
 {  
 super.onCreateOptionsMenu(menu);  
 MenuInflater inflater = getMenuInflater();  
 inflater.inflate(R.menu.*view\_contact\_menu*, menu);return true;  
 } // end method onCreateOptionsMenu  
  
 // handle choice from options menu  
 @Override  
 public boolean onOptionsItemSelected(MenuItem item)  
 {  
 switch (item.getItemId()) // switch based on selected  
 // MenuItem's ID  
 {  
 case R.id.*editItem*:  
 // create an Intent to launch the AddEditContact Activity  
 Intent addEditContact =  
 new Intent(this, AddEditContact.class);  
  
 // pass the selected contact's data as extras  
 // with the Intent  
 addEditContact.putExtra(MainActivity.*ROW\_ID*, rowID);  
 addEditContact.putExtra("name", nameTextView.getText());  
 addEditContact.putExtra("phone",  
 phoneTextView.getText());  
 addEditContact.putExtra("email",  
 emailTextView.getText());  
 addEditContact.putExtra("street",  
 streetTextView.getText());  
 addEditContact.putExtra("city",  
 cityTextView.getText());  
 startActivity(addEditContact); // start the Activity  
 return true;  
 case R.id.*deleteItem*:  
 deleteContact(); // delete the displayed contact  
 return true;  
 default:  
 return super.onOptionsItemSelected(item);  
 } // end switch  
 } // end method onOptionsItemSelected  
 private void deleteContact()  
 {  
 AlertDialog.Builder builder =  
 new AlertDialog.Builder(ViewContact.this);  
  
 builder.setTitle(R.string.*confirmTitle*); // title bar string  
 builder.setMessage(R.string.*confirmMessage*); // message to  
 // display  
  
 // provide an OK button that simply dismisses the dialog  
 builder.setPositiveButton(R.string.*button\_delete*,  
 new DialogInterface.OnClickListener()  
 {  
 @Override  
 public void onClick(DialogInterface dialog, int button) {  
 final DatabaseConnector databaseConnector =  
 new DatabaseConnector(ViewContact.this);  
 // create an AsyncTask that deletes the contact in  
 // another  
 // thread, then calls finish after the deletion  
 AsyncTask<Long, Object, Object> deleteTask =  
 new AsyncTask<Long, Object, Object>()  
 {  
 @Override  
 protected Object doInBackground(Long... params)  
 {  
 databaseConnector.deleteContact(params[0]);  
 return null;  
 } // end method doInBackground  
  
 @Override  
 protected void onPostExecute(Object result)  
 {  
 finish(); // return to the MainActivity  
 // Activity  
 } // end method onPostExecute  
 }; // end new AsyncTask  
  
 // execute the AsyncTask to delete contact at rowID  
 deleteTask.execute(new Long[] { rowID });  
 } // end method onClick  
 } // end anonymous inner class  
 ); // end call to method setPositiveButton  
  
 builder.setNegativeButton(R.string.*button\_cancel*, null);  
 builder.show(); // display the Dialog  
 } // end method deleteContact  
 }

**DatabaseConnector.java**

package com.example.addressbook;  
 import android.content.ContentValues;  
 import android.content.Context;  
 import android.database.Cursor;  
 import android.database.SQLException;  
 import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteOpenHelper;  
 import android.database.sqlite.SQLiteDatabase.CursorFactory;  
  
 public class DatabaseConnector  
 {  
 // database name  
 private static final String *DATABASE\_NAME* = "UserContacts";  
 private SQLiteDatabase database; // database object  
 private DatabaseOpenHelper databaseOpenHelper; // database helper  
 public DatabaseConnector(Context context)  
 {  
 // create a new DatabaseOpenHelper  
 databaseOpenHelper =  
 new DatabaseOpenHelper(context, *DATABASE\_NAME*, null, 1);  
 } // end DatabaseConnector constructor  
  
 // open the database connection  
 public void open() throws SQLException  
 {  
 // create or open a database for reading/writing  
 database = databaseOpenHelper.getWritableDatabase();  
 } // end method open  
  
 // close the database connection  
 public void close()  
 {  
 if (database != null)  
 database.close(); // close the database connection  
 } // end method close  
 public void insertContact(String name, String email, String phone,  
 String state, String city)  
 {  
 ContentValues newContact = new ContentValues();  
 newContact.put("name", name);  
 newContact.put("email", email);  
 newContact.put("phone", phone);  
 newContact.put("street", state);  
 newContact.put("city", city);  
  
 open(); // open the database  
 database.insert("contacts", null, newContact);  
 close(); // close the database  
 } // end method insertContact  
  
 // inserts a new contact in the database  
 public void updateContact(long id, String name, String email,  
 String phone, String state, String city)  
 {  
 ContentValues editContact = new ContentValues();  
 editContact.put("name", name);  
 editContact.put("email", email);  
 editContact.put("phone", phone);  
 editContact.put("street", state);  
 editContact.put("city", city);  
  
 open(); // open the database  
 database.update("contacts", editContact, "\_id=" + id, null);  
 close(); // close the database  
 } // end method updateContact  
  
 // return a Cursor with all contact information in the database  
 public Cursor getAllContacts()  
 {  
 return database.query("contacts", new String[] {"\_id", "name"},  
 null, null, null, null, "name");  
 } // end method getAllContacts  
  
 // get a Cursor containing all information about the contact specified  
 // by the given id  
 public Cursor getOneContact(long id)  
 {  
 return database.query(  
 "contacts", null, "\_id=" + id, null, null, null, null);  
 } // end method getOnContact  
  
 // delete the contact specified by the given String name  
 public void deleteContact(long id)  
 {  
 open(); // open the database  
 database.delete("contacts", "\_id=" + id, null);  
 close(); // close the database  
 } // end method deleteContact  
 private class DatabaseOpenHelper extends SQLiteOpenHelper  
 {  
 // public constructor  
 public DatabaseOpenHelper(Context context, String name,  
 CursorFactory factory, int version)  
 {  
 super(context, name, factory, version);  
 } // end DatabaseOpenHelper constructor  
  
 // creates the contacts table when the database is created  
 @Override  
 public void onCreate(SQLiteDatabase db)  
 {  
 // query to create a new table named contacts  
 String createQuery = "CREATE TABLE contacts" +  
 "(\_id integer primary key autoincrement," +  
 "name TEXT, email TEXT, phone TEXT," +  
 "street TEXT, city TEXT);";  
 db.execSQL(createQuery); // execute the query  
 } // end method onCreate  
  
 @Override  
 public void onUpgrade(SQLiteDatabase db, int oldVersion,  
 int newVersion)  
 {  
 } // end method onUpgrade  
 } // end class DatabaseOpenHelper  
 } // end class DatabaseConnector

**AddEditContact.java**

package com.example.addressbook;  
import com.example.addressbook.R;  
import android.app.Activity;  
import android.app.AlertDialog;  
import android.os.AsyncTask;  
 import android.os.Bundle;  
import android.view.View;  
import android.view.View.OnClickListener;  
import android.widget.Button;  
import android.widget.EditText;  
public class AddEditContact extends Activity  
{  
 private long rowID; // id of contact being edited, if any  
 private EditText nameEditText;  
 private EditText phoneEditText;  
 private EditText emailEditText;  
 private EditText streetEditText;  
 private EditText cityEditText;  
 @Override  
 public void onCreate(Bundle savedInstanceState)  
 { super.onCreate(savedInstanceState); // call super's onCreate  
 setContentView(R.layout.*add\_contact*); // inflate the UI  
 nameEditText = (EditText) findViewById(R.id.*nameEditText*);  
 emailEditText = (EditText) findViewById(R.id.*emailEditText*);  
 phoneEditText = (EditText) findViewById(R.id.*phoneEditText*);  
 streetEditText = (EditText) findViewById(R.id.*streetEditText*);  
 cityEditText = (EditText) findViewById(R.id.*cityEditText*);  
  
 Bundle extras = getIntent().getExtras(); // get Bundle of extras  
  
 // if there are extras, use them to populate the EditTexts  
 if (extras != null)  
 {  
 rowID = extras.getLong("row\_id");  
 nameEditText.setText(extras.getString("name"));  
 emailEditText.setText(extras.getString("email"));  
 phoneEditText.setText(extras.getString("phone"));  
 streetEditText.setText(extras.getString("street"));  
 cityEditText.setText(extras.getString("city"));  
 } // end if  
  
 // set event listener for the Save Contact Button  
 Button saveContactButton =  
 (Button) findViewById(R.id.*saveContactButton*);  
 saveContactButton.setOnClickListener(saveContactButtonClicked);  
 } // end method onCreate  
 OnClickListener saveContactButtonClicked = new OnClickListener()  
 {  
 @Override  
 public void onClick(View v)  
 {  
 if (nameEditText.getText().length() != 0)  
 {  
 AsyncTask<Object, Object, Object> saveContactTask =  
 new AsyncTask<Object, Object, Object>()  
 {  
 @Override  
 protected Object doInBackground(Object... params)  
 {  
 saveContact(); // save contact to the database  
 return null;  
 } // end method doInBackground  
  
 @Override  
 protected void onPostExecute(Object result)  
 {  
 finish(); // return to the previous Activity  
 } // end method onPostExecute  
 }; // end AsyncTask  
  
 // save the contact to the database using  
 // a separate thread  
 saveContactTask.execute((Object[]) null);  
 } // end if  
 else  
 {  
 // create a new AlertDialog Builder  
 AlertDialog.Builder builder =  
 new AlertDialog.Builder(AddEditContact.this);  
  
 // set dialog title & message, and provide Button  
 // to dismiss  
 builder.setTitle(R.string.*errorTitle*);  
 builder.setMessage(R.string.*errorMessage*);  
 builder.setPositiveButton(R.string.*errorButton*, null);  
 builder.show(); // display the Dialog  
 } // end else  
 } // end method onClick  
 }; // end OnClickListener saveContactButtonClicked  
  
 private void saveContact()  
 {  
 // get DatabaseConnector to interact with the SQLite database  
 DatabaseConnector databaseConnector = new DatabaseConnector(this);  
 if (getIntent().getExtras() == null)  
 {  
 // insert the contact information into the database  
 databaseConnector.insertContact(  
 nameEditText.getText().toString(),  
 emailEditText.getText().toString(),  
 phoneEditText.getText().toString(),  
 streetEditText.getText().toString(),  
 cityEditText.getText().toString());  
 } // end if  
 else  
 {  
 databaseConnector.updateContact(rowID,  
 nameEditText.getText().toString(),  
 emailEditText.getText().toString(),  
 phoneEditText.getText().toString(),  
 streetEditText.getText().toString(),  
 cityEditText.getText().toString());  
 } // end else  
 } // end class saveContact  
} // end class AddEditContact

**MainActivity.java**

package com.example.addressbook;  
import com.example.addressbook.R;  
import android.app.ListActivity;  
import android.content.Intent;  
import android.database.Cursor;  
import android.os.AsyncTask;  
import android.os.Bundle;  
import android.view.Menu;  
import android.view.MenuInflater;  
import android.view.MenuItem;  
import android.view.View;  
import android.widget.AdapterView;  
import android.widget.AdapterView.OnItemClickListener;  
import android.widget.CursorAdapter;  
import android.widget.ListView;  
import android.widget.SimpleCursorAdapter;  
  
 public class MainActivity extends ListActivity  
{  
 public static final String *ROW\_ID* = "row\_id"; // Intent extra key  
 private ListView contactListView; // the ListActivity's ListView  
 private CursorAdapter contactAdapter; // adapter for ListView  
 @Override  
 public void onCreate(Bundle savedInstanceState)  
 {  
 super.onCreate(savedInstanceState); // call super's onCreate  
 contactListView = getListView(); // get the built-in ListView  
 contactListView.setOnItemClickListener(viewContactListener);  
  
 // map each contact's name to a TextView in the ListView layout  
 String[] from = new String[] { "name" };  
 int[] to = new int[] { R.id.*contactTextView* };  
 contactAdapter = new SimpleCursorAdapter(  
 MainActivity.this, R.layout.*activity\_main*, null, from, to);  
 setListAdapter(contactAdapter); // set contactView's adapter  
 } // end method onCreate  
 @Override  
 protected void onResume()  
 {  
 super.onResume(); // call super's onResume method  
  
 // create new GetContactsTask and execute it  
 new GetContactsTask().execute((Object[]) null);  
 } // end method onResume  
 @Override  
 protected void onStop()  
 {  
 Cursor cursor = contactAdapter.getCursor(); // get current Cursor  
  
 if (cursor != null)  
 cursor.deactivate(); // deactivate it  
  
 contactAdapter.changeCursor(null); // adapted now has no Cursor  
 super.onStop();  
 } // end method onStop  
  
 private class GetContactsTask extends AsyncTask<Object, Object,  
 Cursor>  
 {  
 DatabaseConnector databaseConnector =  
 new DatabaseConnector(MainActivity.this);  
  
 // perform the database access  
 @Override  
 protected Cursor doInBackground(Object... params)  
 {  
 databaseConnector.open();  
  
 // get a cursor containing call contacts  
 return databaseConnector.getAllContacts();  
 } // end method doInBackground  
  
 // use the Cursor returned from the doInBackground method  
 @Override  
 protected void onPostExecute(Cursor result)  
 {  
 contactAdapter.changeCursor(result); // set the adapter's  
 // Cursor  
 databaseConnector.close();  
 } // end method onPostExecute  
 } // end class GetContactsTask  
 @Override  
 public boolean onCreateOptionsMenu(Menu menu)  
 {  
 MenuInflater inflater = getMenuInflater();  
 inflater.inflate(R.menu.*main\_activity\_menu*, menu);  
 return super.onCreateOptionsMenu(menu);  
 } // end method onCreateOptionsMenu  
  
 // handle choice from options menu  
 @Override  
 public boolean onOptionsItemSelected(MenuItem item)  
 {  
 // create a new Intent to launch the AddEditContact Activity  
 Intent addNewContact =  
 new Intent(MainActivity.this, AddEditContact.class);  
 startActivity(addNewContact); // start the AddEditContact  
 // Activity  
 return super.onOptionsItemSelected(item); // call super's  
 // method  
 }  
 OnItemClickListener viewContactListener = new  
 OnItemClickListener()  
 {  
 @Override  
 public void onItemClick(AdapterView<?> arg0, View arg1,  
 int arg2, long arg3)  
 {  
 // create an Intent to launch the ViewContact Activity  
 Intent viewContact =  
 new Intent(MainActivity.this, ViewContact.class);  
  
 // pass the selected contact's row ID as an extra  
 // with the Intent  
 viewContact.putExtra(*ROW\_ID*, arg3);  
 startActivity(viewContact); // start the ViewContact  
 // Activity  
 } // end method onItemClick  
 }; // end viewContactListener  
 } // end class MainActivity

**Висновок:** Навчився розробляти Android – додатки з декількома активностями, використовуючи списки, меню та СУБД SQLite.