Курсов Проект

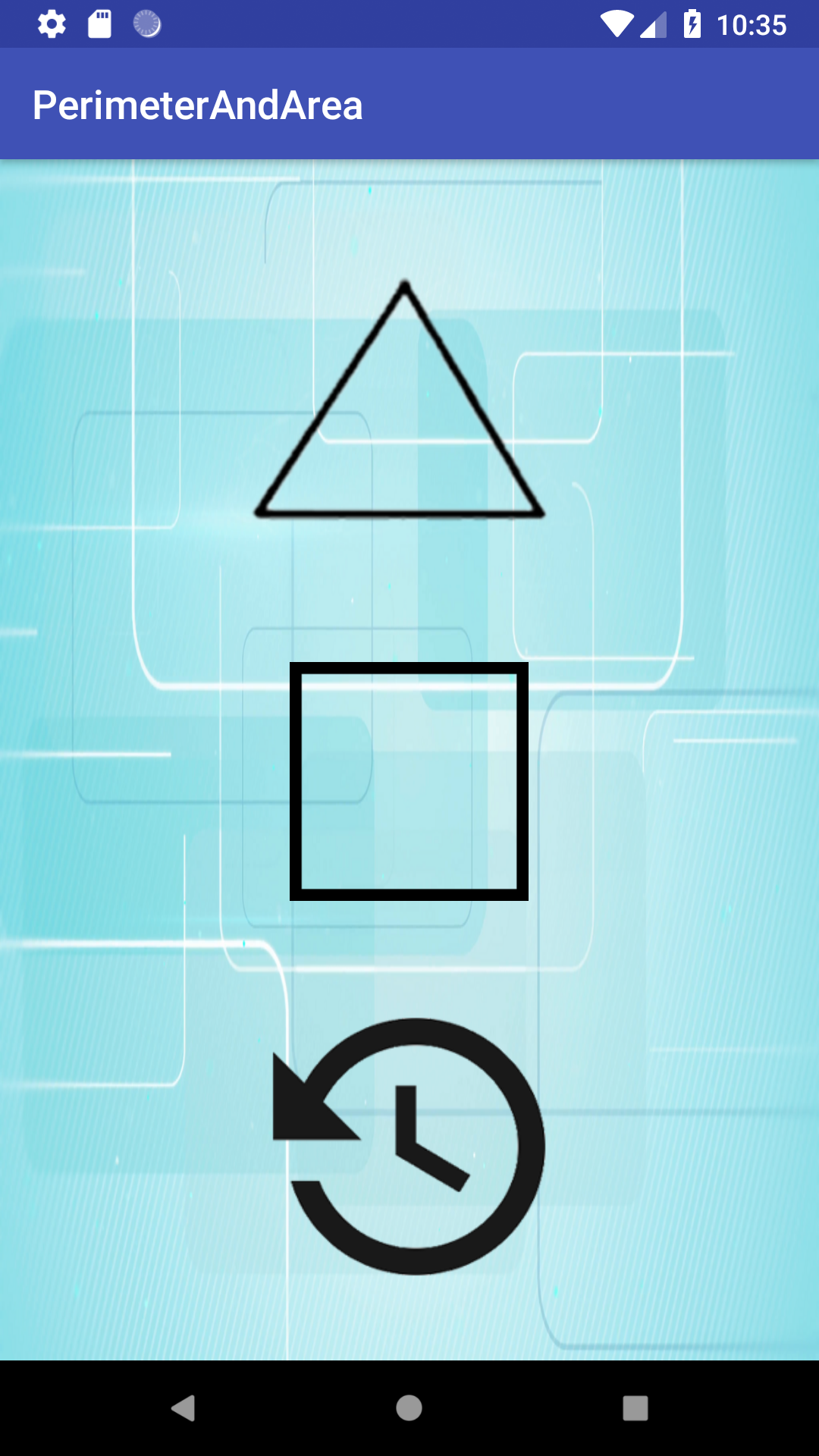
Тема: Приложение, изчисляващо лице и периметър на триъгълник и правоъгълник по различни зададени параметри

Даниел Динков Динев

Фак № 1501681001

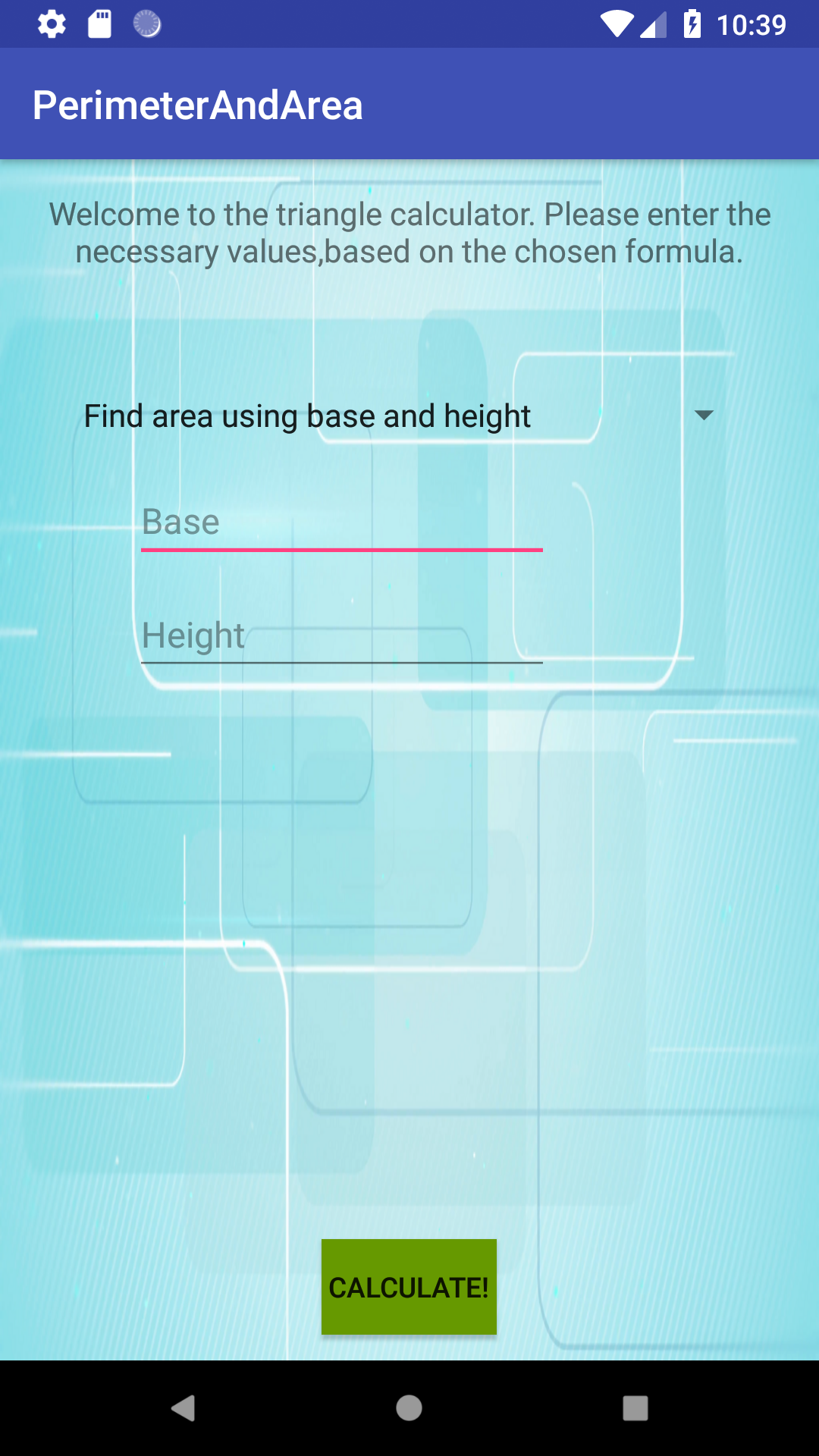
Целта на проекта е да се изгради приложение, което да може лесно да изчисли лице или периметър на правоъгълник или кръг. Също така да може да запазва информацията за предишни изчисления, ако се наложи да се направи справка.

Приложението е изградено в Android Studio и за целта е използвана версия на framework – 27 и минимално поддържана – 22, за да може да предоставим по-голям достъп за хора с по-остарели телефони. Приложението е лесно и интуитивно за употреба. При влизане виждаме следното:

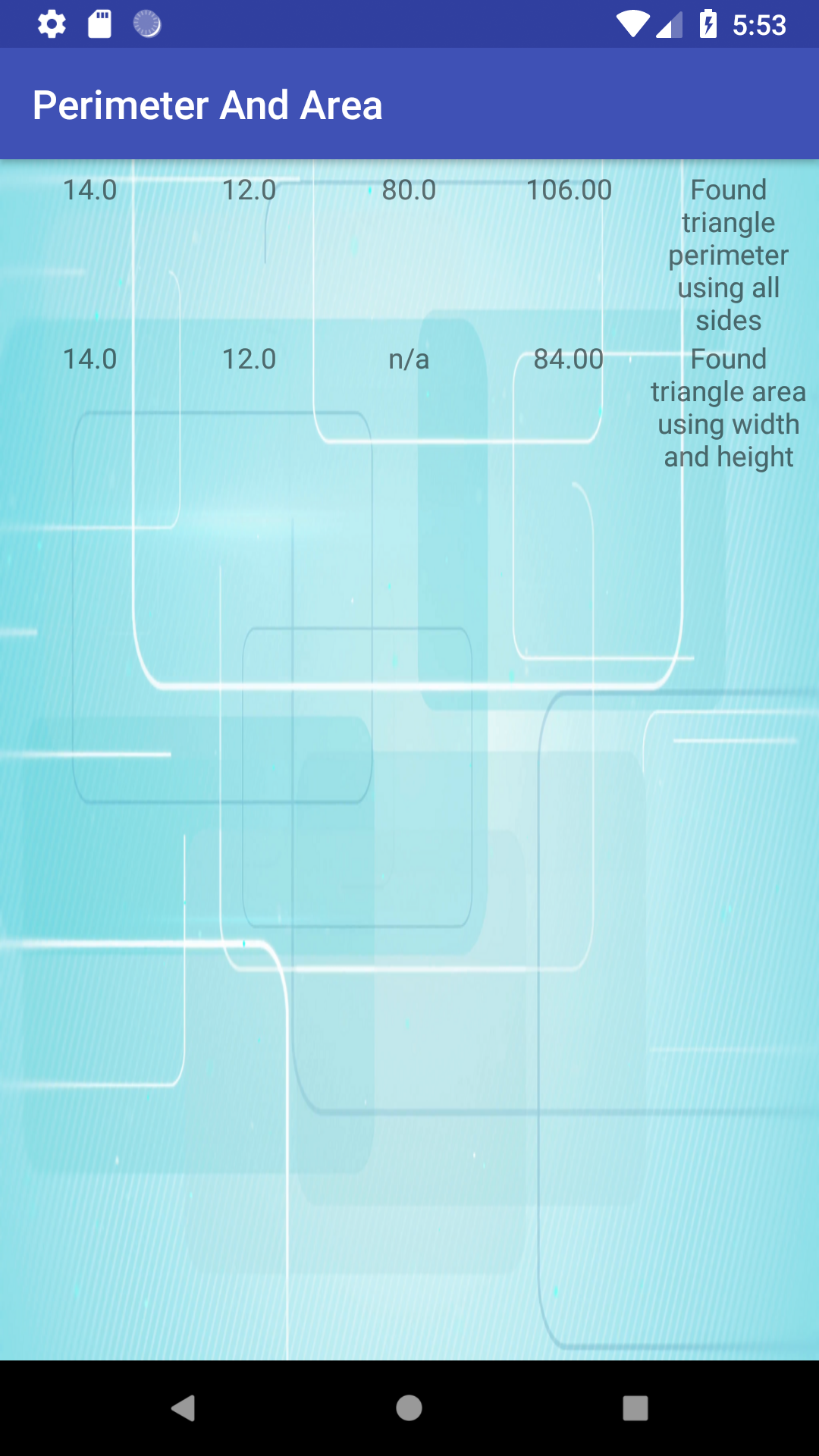


Всеки потребител може лесно да достъпи различните функционалности през големите икони на началния екран.

За по-голямо удобство, екраните за изчисляване на формули, и на правоъгълник и на квадрат, не се различават и изглеждат по следния начин:



От падащото меню, може да бъде избрана формулата, по която да се изчисли лицето или периметъра на фигурата. След избиране на формула, трябва да се въведат нужните стойности, и да се натисне бутона „Изчисли“ , който връща резултата.  
Резултата, заедно със зададените числа и използваната формула биват прихванати и използвани в 3-ия бутон, който е история на изчисляванията.



Тук излизат стойностите, които са зададени от потребителя , резултата и използваната формула. Бутона е неактивен докато не са вкарани цифрови стойности и в двете полета.

Първо да започнем с главния екран:

package com.example.ddine.perimeterandarea;  
  
import android.content.Intent;  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.ImageButton;  
  
  
public class MainActivity extends AppCompatActivity {  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 ImageButton triangle = findViewById(R.id.*triangleButton*);  
 ImageButton square = findViewById(R.id.*squareButton*);  
 ImageButton history = findViewById(R.id.*historyButton*);  
  
 triangle.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Intent triangleCalculator = new Intent(MainActivity.this,TriangleCalculator.class);  
 startActivity(triangleCalculator);  
 }  
 });  
 square.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Intent squareCalculator = new Intent(MainActivity.this,SquareCalculator.class);  
 startActivity(squareCalculator);  
 }  
 });  
 history.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Intent calculationHistory = new Intent(MainActivity.this,CalculationHistory.class);  
 startActivity(calculationHistory);  
 }  
 });  
 }  
}

<?xml version="1.0" encoding="utf-8"?>  
<android.support.constraint.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"  
 android:background="@mipmap/background"  
 tools:context=".MainActivity">  
  
 <ImageButton  
 android:background="@android:color/transparent"  
 android:id="@+id/triangleButton"  
 android:layout\_width="190dp"  
 android:layout\_height="137dp"  
 android:layout\_marginBottom="19dp"  
 android:layout\_marginTop="16dp"  
 android:contentDescription="@string/triangle"  
 app:layout\_constraintBottom\_toTopOf="@+id/squareButton"  
 app:layout\_constraintStart\_toStartOf="@+id/squareButton"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:srcCompat="@mipmap/triangle" />  
  
 <ImageButton  
 android:background="@android:color/transparent"  
 android:id="@+id/squareButton"  
 android:layout\_width="190dp"  
 android:layout\_height="137dp"  
 android:layout\_marginBottom="11dp"  
 android:contentDescription="@string/square"  
 app:layout\_constraintBottom\_toTopOf="@+id/historyButton"  
 app:layout\_constraintEnd\_toEndOf="@+id/historyButton"  
 app:layout\_constraintStart\_toStartOf="@+id/historyButton"  
 app:layout\_constraintTop\_toBottomOf="@+id/triangleButton"  
 app:srcCompat="@mipmap/png\_square" />  
  
 <ImageButton  
 android:background="@android:color/transparent"  
 android:id="@+id/historyButton"  
 android:layout\_width="190dp"  
 android:layout\_height="137dp"  
 android:layout\_marginBottom="3dp"  
 android:contentDescription="@string/history"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/squareButton"  
 app:srcCompat="@mipmap/history" />  
</android.support.constraint.ConstraintLayout>

Тук единственото,което имаме са инициализиране на бутоните, за препратка към другите екрани. Самата база данни и нейните функции са създадени в отделен клас.

package com.example.ddine.perimeterandarea;  
  
import android.content.ContentValues;  
import android.content.Context;  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteOpenHelper;  
import android.util.Log;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class DBOpenHelper extends SQLiteOpenHelper {  
  
 public static final String *DATABASE\_NAME* = "CalculatingShapeDB";  
 public static final int *DATABASE\_VERSION* = 4;  
  
 private static final String *TRIANGLE\_TABLE\_NAME* = "triangle";  
 private static final String *TRIANGLE\_COL\_ID* = "id";  
 private static final String *TRIANGLE\_VALUE\_A* = "valueA";  
 private static final String *TRIANGLE\_VALUE\_B* = "valueB";  
 private static final String *TRIANGLE\_VALUE\_C* = "valueC";  
 private static final String *TRIANGLE\_RESULT* = "result";  
 private static final String *TRIANGLE\_HISTORY\_INFORMATION* = "history";  
  
  
  
 private static final String *QUERY\_TRIANGLE\_CREATE\_TABLE* =  
 "CREATE TABLE " + *TRIANGLE\_TABLE\_NAME* + " (" + *TRIANGLE\_COL\_ID* + " INTEGER PRIMARY KEY, " +  
 *TRIANGLE\_VALUE\_A* + " TEXT, " + *TRIANGLE\_VALUE\_C* + " TEXT, " + *TRIANGLE\_VALUE\_B* + " TEXT, " + *TRIANGLE\_RESULT* + " TEXT," + *TRIANGLE\_HISTORY\_INFORMATION* + " TEXT" + " );";  
  
  
 public DBOpenHelper(Context context) {  
 super(context, *DATABASE\_NAME*, null, *DATABASE\_VERSION*);  
 SQLiteDatabase mReadableDB = getReadableDatabase();  
 SQLiteDatabase mWritableDB = getWritableDatabase();  
 }  
 @Override  
 public void onCreate(SQLiteDatabase db) {  
 Log.*e*("query", *QUERY\_TRIANGLE\_CREATE\_TABLE*);  
 db.execSQL(*QUERY\_TRIANGLE\_CREATE\_TABLE*);  
 }  
  
 @Override  
 public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {  
 db.execSQL("DROP TABLE IF EXISTS " + *TRIANGLE\_TABLE\_NAME* );  
 onCreate(db);  
 }  
  
 public void insertTriangleCalc(String a,String b, String c,String d,String e){  
 SQLiteDatabase database = this.getWritableDatabase();  
 ContentValues values = new ContentValues();  
 values.put(*TRIANGLE\_VALUE\_A*,a);  
 values.put(*TRIANGLE\_VALUE\_B*,b);  
 values.put(*TRIANGLE\_VALUE\_C*,c);  
 values.put(*TRIANGLE\_RESULT*,d);  
 values.put(*TRIANGLE\_HISTORY\_INFORMATION*,e);  
  
 database.insert("triangle", null, values);  
 database.close();  
  
 }  
  
 public List<Triangle> ObjectRead() {  
  
 List<Triangle> valuesList = new ArrayList<>();  
  
 String selectQuery = "SELECT \* FROM " + *TRIANGLE\_TABLE\_NAME* + " ORDER BY " + *TRIANGLE\_COL\_ID* + " DESC";  
 SQLiteDatabase db = this.getReadableDatabase();  
 Cursor cursor = db.rawQuery(selectQuery,null);  
  
 if ( cursor != null && cursor.moveToFirst()) {  
 do {  
  
 int id = Integer.*parseInt*(cursor.getString(cursor.getColumnIndex("id")));  
 String valueA = cursor.getString(cursor.getColumnIndex("valueA"));  
 String valueB = cursor.getString(cursor.getColumnIndex("valueB"));  
 String valueC = cursor.getString(cursor.getColumnIndex("valueC"));  
 String result = cursor.getString(cursor.getColumnIndex("result"));  
 String history = cursor.getString(cursor.getColumnIndex("history"));  
  
 Triangle triangle = new Triangle();  
 triangle.id = id;  
 triangle.valueA = valueA;  
 triangle.valueB = valueB;  
 triangle.valueC = valueC;  
 triangle.Result = result;  
 triangle.history = history;  
 valuesList.add(triangle);  
  
 } while (cursor.moveToNext());  
 }  
 cursor.close();  
 db.close();  
  
 return valuesList;  
 }  
  
  
}

Тук забелязваме, че за извикване на данните е използван отделен клас „Triangle”, който служи за обект, да държи нужната информация и да я подава на екрана отговорен за историята.

package com.example.ddine.perimeterandarea;  
  
public class Triangle {  
 int id;  
 String valueA;  
 String valueB;  
 String valueC;  
 String Result;  
 String history;  
 public Triangle(){  
  
 }  
}

Класът отговорен да инициализира екрана с историята е :

package com.example.ddine.perimeterandarea;  
  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.widget.LinearLayout;  
View tableRow = LayoutInflater.*from*(this).inflate(R.layout.*table\_item*,null,false);  
TextView valueA = tableRow.findViewById(R.id.*valueA*);  
TextView valueB = tableRow.findViewById(R.id.*valueB*);  
TextView valueC = tableRow.findViewById(R.id.*valueC*);  
TextView result = tableRow.findViewById(R.id.*result*);  
TextView explanation = tableRow.findViewById(R.id.*explanation*);

<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_height="match\_parent"  
 android:layout\_width="match\_parent"  
 android:background="@mipmap/background"  
 android:layout\_weight="1"  
 android:scrollbars="none"  
 >  
  
<TableLayout  
 android:id="@+id/tableLayoutRecords"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:padding="5dp"  
 android:orientation="vertical"  
 android:layout\_below="@+id/textViewRecordCount"  
 >  
  
</TableLayout>  
</ScrollView>

<?xml version="1.0" encoding="utf-8"?>  
<TableRow xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal">  
  
 <TextView  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/valueA"  
 android:layout\_weight="0.25"  
 android:gravity="center"/>  
  
 <TextView  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/valueB"  
 android:layout\_weight="0.25"  
 android:gravity="center"/>  
  
 <TextView  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/valueC"  
 android:layout\_weight="0.25"  
 android:gravity="center"/>  
 <TextView  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/result"  
 android:layout\_weight="0.25"  
 android:gravity="center"/>  
 <TextView  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/explanation"  
 android:layout\_weight="0.25"  
 android:gravity="center"/>  
  
</TableRow>

Следват класовете отговорни за създаването на екраните за правоъгълник и триъгълник, както и тяхната логика.

package com.example.ddine.perimeterandarea;  
  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.text.Editable;  
import android.text.TextWatcher;  
import android.view.View;  
import android.widget.AdapterView;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.Spinner;  
import android.widget.TextView;  
  
public class SquareCalculator extends AppCompatActivity implements TextWatcher {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_square\_calculator*);  
 final Spinner menuSelect = findViewById(R.id.*rectangle\_formula\_spinner*);  
 final EditText input1 = findViewById(R.id.*SquareInput1*);  
 final EditText input2 = findViewById(R.id.*SquareInput2*);  
 Button calculate = findViewById(R.id.*calculate*);  
 final TextView Result = findViewById(R.id.*SquareResult*);  
 Result.setVisibility(View.*GONE*);  
 final DBOpenHelper dbOpenHelper = new DBOpenHelper(this);  
 input1.addTextChangedListener(this);  
 input2.addTextChangedListener(this);  
 calculate.setEnabled(false);  
 menuSelect.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {  
 @Override  
 public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {  
 switch (position){  
 case 0:  
 input1.setHint("Width");  
 input2.setHint("Height");  
  
  
 break;  
 case 1:  
 input1.setHint("Length");  
 input2.setHint("Diagonal");  
  
 break;  
  
 case 2:  
 input1.setHint("Width");  
 input2.setHint("Height");  
  
 break;  
 }  
 }  
  
 @Override  
 public void onNothingSelected(AdapterView<?> parent) {  
  
 }  
 });  
  
 calculate.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
  
 if (menuSelect.getSelectedItemPosition() == 0 ){  
 double width = Double.*parseDouble*(input1.getText().toString());  
 double height = Double.*parseDouble*(input2.getText().toString());  
 double result = width\*height;  
 dbOpenHelper.insertTriangleCalc(String.*valueOf*(width),String.*valueOf*(height),"n/a",String.*valueOf*(result),"Found rectangle area using width and length");  
 Result.setText(Double.*toString*(result));  
 Result.setVisibility(View.*VISIBLE*);  
 } else if(menuSelect.getSelectedItemPosition() == 1 ){  
 double a = Double.*parseDouble*(input1.getText().toString());  
 double b = Double.*parseDouble*(input2.getText().toString());  
 double d = Math.*sqrt*((Math.*pow*(a,2)+ Math.*pow*(b,2)));  
 double result = a\*d;  
 dbOpenHelper.insertTriangleCalc(String.*valueOf*(a),String.*valueOf*(b),"n/a",String.*valueOf*(result),"Found rectangle area using length and diagonal");  
 Result.setText(Double.*toString*(result));  
 Result.setVisibility(View.*VISIBLE*);  
 }  
 else if(menuSelect.getSelectedItemPosition() == 2){  
 double a = Double.*parseDouble*(input1.getText().toString());  
 double b = Double.*parseDouble*(input2.getText().toString());  
 double result = 2\*a+2\*b;  
 dbOpenHelper.insertTriangleCalc(String.*valueOf*(a),String.*valueOf*(b),"n/a",String.*valueOf*(result),"Found rectangle perimeter using width and length");  
 Result.setText(Double.*toString*(result));  
 Result.setVisibility(View.*VISIBLE*);  
  
 }  
  
 }  
  
 });  
 }  
 @Override  
 public void beforeTextChanged(CharSequence s, int start, int count, int after) {  
  
 }  
  
 @Override  
 public void onTextChanged(CharSequence s, int start, int before, int count) {  
  
 }  
  
 @Override  
 public void afterTextChanged(Editable s) {  
 final EditText input1 = findViewById(R.id.*SquareInput1*);  
 final EditText input2 = findViewById(R.id.*SquareInput2*);  
 Button calculate = findViewById(R.id.*calculate*);  
  
 if (input1.getText().length() > 0  
 && input2.getText().length() > 0  
 ) {  
  
 calculate.setEnabled(true);  
 } else {  
  
 calculate.setEnabled(false);  
 }  
 }  
  
}

<?xml version="1.0" encoding="utf-8"?>  
<android.support.constraint.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"  
 android:background="@mipmap/background"  
 tools:context=".TriangleCalculator">  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="16dp"  
 android:text="@string/squareGreeting"  
 android:textAlignment="center"  
 android:textSize="16sp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 tools:text="This is the calculator for Rectangles. Choose a formula and enter the necessary values" />  
  
 <Spinner  
 android:id="@+id/rectangle\_formula\_spinner"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginEnd="34dp"  
 android:layout\_marginStart="34dp"  
 android:layout\_marginTop="116dp"  
 android:contentDescription="@string/chooseFormula"  
 android:entries="@array/rectangle\_formula\_arrays"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <EditText  
 android:hint="@string/length"  
 android:id="@+id/SquareInput1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="33dp"  
 android:layout\_marginTop="19dp"  
 android:ems="10"  
 android:inputType="number"  
 app:layout\_constraintStart\_toStartOf="@+id/rectangle\_formula\_spinner"  
 app:layout\_constraintTop\_toBottomOf="@+id/rectangle\_formula\_spinner" />  
  
 <EditText  
 android:hint="@string/width"  
 android:id="@+id/SquareInput2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="12dp"  
 android:ems="10"  
 android:inputType="number"  
 app:layout\_constraintStart\_toStartOf="@+id/SquareInput1"  
 app:layout\_constraintTop\_toBottomOf="@+id/SquareInput1" />  
  
 <TextView  
 android:id="@+id/SquareResult"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="30dp"  
 android:layout\_marginBottom="80dp"  
 android:layout\_marginEnd="8dp"  
 android:layout\_marginStart="8dp"  
 android:ems="14"  
 android:text="@string/result"  
 android:textAlignment="center"  
 android:textSize="16sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="@+id/rectangle\_formula\_spinner"  
 app:layout\_constraintHorizontal\_bias="0.327"  
 app:layout\_constraintStart\_toStartOf="@+id/rectangle\_formula\_spinner" />  
  
 <Button  
 android:id="@+id/calculate"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="13dp"  
 android:background="@android:color/holo\_green\_dark"  
 android:text="@string/calculate"  
 app:layout\_constraintBottom\_toTopOf="@+id/SquareResult"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent" />  
  
</android.support.constraint.ConstraintLayout>

package com.example.ddine.perimeterandarea;  
  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.text.Editable;  
import android.text.TextWatcher;  
import android.view.View;  
import android.widget.AdapterView;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.Spinner;  
import android.widget.TextView;  
  
public class TriangleCalculator extends AppCompatActivity implements TextWatcher {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_triangle\_calculator*);  
 final Spinner menuSelect = findViewById(R.id.*triangle\_formula\_spinner*);  
 final EditText input1 = findViewById(R.id.*SquareInput1*);  
 final EditText input2 = findViewById(R.id.*SquareInput2*);  
 Button calculate = findViewById(R.id.*calculate*);  
 final EditText input3 = findViewById(R.id.*Input3*);  
  
 final TextView Result = findViewById(R.id.*SquareResult*);  
 Result.setVisibility(View.*GONE*);  
 input3.setVisibility(View.*INVISIBLE*);  
 final DBOpenHelper dbOpenHelper = new DBOpenHelper(this);  
 calculate.setEnabled(false);  
  
 input1.addTextChangedListener(this);  
 input2.addTextChangedListener(this);  
  
 menuSelect.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {  
 @Override  
 public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {  
 switch (position){  
 case 0:  
 input1.setHint("Base");  
 input2.setHint("Height");  
 input3.setVisibility(View.*INVISIBLE*);  
  
 break;  
 case 1:  
 input1.setHint("Side A");  
 input2.setHint("Side B");  
 input3.setHint("Side C");  
 input3.setVisibility(View.*VISIBLE*);  
 break;  
 case 2:  
 input1.setHint("Side A");  
 input2.setHint("Side B");  
 input3.setHint("Angle C");  
 input3.setVisibility(View.*VISIBLE*);  
 break;  
 case 3:  
 input1.setHint("Side A");  
 input2.setHint("Side B");  
 input3.setHint("Side C");  
 input3.setVisibility(View.*VISIBLE*);  
 break;  
 case 4:  
 input1.setHint("Side A");  
 input2.setHint("Side B");  
 input3.setHint("Angle C");  
 input3.setVisibility(View.*VISIBLE*);  
 }  
 }  
  
 @Override  
 public void onNothingSelected(AdapterView<?> parent) {  
  
 }  
 });  
  
 calculate.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
  
 if (menuSelect.getSelectedItemPosition() == 0 ){  
 double base = Double.*parseDouble*(input1.getText().toString());  
 double height = Double.*parseDouble*(input2.getText().toString());  
 double result = (base\*height)/2;  
 dbOpenHelper.insertTriangleCalc(String.*valueOf*(base),String.*valueOf*(height),"n/a",String.*valueOf*(result),"Found triangle area using width and height");  
 Result.setText(Double.*toString*(result));  
 Result.setVisibility(View.*VISIBLE*);  
  
 }  
 else if(menuSelect.getSelectedItemPosition() == 1 ){  
 double a = Double.*parseDouble*(input1.getText().toString());  
 double b = Double.*parseDouble*(input2.getText().toString());  
 double c = Double.*parseDouble*(input3.getText().toString());  
 double p = (a+b+c)/2;  
 double area = Math.*sqrt*((p\*(p-a)\*(p-b)\*(p-c)));  
 dbOpenHelper.insertTriangleCalc(String.*valueOf*(a),String.*valueOf*(b),String.*valueOf*(c),String.*valueOf*(area),"Found triangle area using all sides");  
 Result.setText(Double.*toString*(area));  
 Result.setVisibility(View.*VISIBLE*);  
  
 }  
 else if(menuSelect.getSelectedItemPosition() == 2){  
 double a = Double.*parseDouble*(input1.getText().toString());  
 double b = Double.*parseDouble*(input2.getText().toString());  
 double c = Double.*parseDouble*(input3.getText().toString());  
 double radians = Math.*toRadians*(c);  
 double area = Math.*sin*(radians) \* ((a\*b)/2);  
 dbOpenHelper.insertTriangleCalc(String.*valueOf*(a),String.*valueOf*(b),String.*valueOf*(c),String.*valueOf*(area),"Found triangle area using trig functions");  
 Result.setText(Double.*toString*(area));  
 Result.setVisibility(View.*VISIBLE*);  
 }  
  
 else if(menuSelect.getSelectedItemPosition() == 3){  
 double a = Double.*parseDouble*(input1.getText().toString());  
 double b = Double.*parseDouble*(input2.getText().toString());  
 double c = Double.*parseDouble*(input3.getText().toString());  
 double p = a+b+c;  
 dbOpenHelper.insertTriangleCalc(String.*valueOf*(a),String.*valueOf*(b),String.*valueOf*(c),String.*valueOf*(p),"Found triangle perimeter using all sides");  
 Result.setText(Double.*toString*(p));  
 Result.setVisibility(View.*VISIBLE*);  
  
 }  
 else if(menuSelect.getSelectedItemPosition() == 4) {  
 double a = Double.*parseDouble*(input1.getText().toString());  
 double b = Double.*parseDouble*(input2.getText().toString());  
 double c = Double.*parseDouble*(input3.getText().toString());  
 double radians = Math.*toRadians*(c);  
 double d = Math.*sqrt*((Math.*pow*(a,2)+Math.*pow*(b,2)-(2\*a\*b\*Math.*cos*(radians))));  
  
 double result = d + a + b;  
 dbOpenHelper.insertTriangleCalc(String.*valueOf*(a),String.*valueOf*(b),String.*valueOf*(c),String.*valueOf*(result),"Found triangle perimeter using Law of Cosines");  
 Result.setText(Double.*toString*(result));  
 Result.setVisibility(View.*VISIBLE*);  
 }  
  
 }  
  
 });  
}  
  
 @Override  
 public void beforeTextChanged(CharSequence s, int start, int count, int after) {  
  
 }  
  
 @Override  
 public void onTextChanged(CharSequence s, int start, int before, int count) {  
  
 }  
  
 @Override  
 public void afterTextChanged(Editable s) {  
 final EditText input1 = findViewById(R.id.*SquareInput1*);  
 final EditText input2 = findViewById(R.id.*SquareInput2*);  
 Button calculate = findViewById(R.id.*calculate*);  
  
 if (input1.getText().length() > 0  
 && input2.getText().length() > 0  
 ) {  
  
 calculate.setEnabled(true);  
 } else {  
  
 calculate.setEnabled(false);  
 }  
 }  
}

<?xml version="1.0" encoding="utf-8"?>  
<android.support.constraint.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"  
 android:background="@mipmap/background"  
 tools:context=".TriangleCalculator">  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="16dp"  
 android:text="@string/triangleGreeting"  
 android:textAlignment="center"  
 android:textSize="16sp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
  
 tools:text="This is the calculator for Triangles. Choose a formula and enter the necessary values" />  
  
 <Spinner  
 android:id="@+id/triangle\_formula\_spinner"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginEnd="34dp"  
 android:layout\_marginStart="34dp"  
 android:layout\_marginTop="116dp"  
 android:contentDescription="@string/chooseFormula"  
 android:entries="@array/triangle\_formula\_arrays"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <EditText  
 android:id="@+id/SquareInput1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="33dp"  
 android:layout\_marginTop="19dp"  
 android:ems="10"  
 android:inputType="number"  
 android:hint="@string/base"  
 app:layout\_constraintStart\_toStartOf="@+id/triangle\_formula\_spinner"  
 app:layout\_constraintTop\_toBottomOf="@+id/triangle\_formula\_spinner" />  
  
 <EditText  
 android:id="@+id/SquareInput2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="12dp"  
 android:ems="10"  
 android:inputType="number"  
 android:hint="@string/height"  
 app:layout\_constraintStart\_toStartOf="@+id/SquareInput1"  
 app:layout\_constraintTop\_toBottomOf="@+id/SquareInput1" />  
  
 <EditText  
 android:id="@+id/Input3"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="24dp"  
 android:ems="10"  
 android:inputType="number"  
 android:hint="@string/name"  
 app:layout\_constraintStart\_toStartOf="@+id/SquareInput2"  
 app:layout\_constraintTop\_toBottomOf="@+id/SquareInput2" />  
  
 <TextView  
 android:id="@+id/SquareResult"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="30dp"  
 android:layout\_marginBottom="80dp"  
 android:layout\_marginStart="8dp"  
 android:ems="14"  
 android:text="@string/result"  
 android:textAlignment="center"  
 android:textSize="16sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="@+id/triangle\_formula\_spinner"  
 app:layout\_constraintHorizontal\_bias="0.327"  
 app:layout\_constraintStart\_toStartOf="@+id/triangle\_formula\_spinner" />  
  
 <Button  
 android:id="@+id/calculate"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="13dp"  
 android:background="@android:color/holo\_green\_dark"  
 android:text="@string/calculate"  
 app:layout\_constraintBottom\_toTopOf="@+id/SquareResult"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent" />  
  
</android.support.constraint.ConstraintLayout>

За стойностите в XML файловете, които използват текст, е използван отделен файл, от където те да бъдат съдържани :

<resources>  
 <string name="app\_name">PerimeterAndArea</string>  
 <string name="square">square</string>  
 <string name="triangle">triangle</string>  
 <string name="history">history</string>  
 <string name="chooseFormula">Choose Formula</string>  
 <string name="calculate">Calculate!</string>  
 <string name="result">Result</string>  
 <string name="name">Name</string>  
 <string name="height">Height</string>  
 <string name="base">Base</string>  
 <string name="triangleGreeting">Welcome to the triangle calculator. Please enter the necessary values,based on the chosen formula.</string>  
 <string name="squareGreeting">Welcome to the triangle calculator. Please enter the necessary values,based on the chosen formula.</string>  
 <string name="width">Width</string>  
 <string name="length">Length</string>  
  
 <string-array name="triangle\_formula\_arrays">  
 <item>Find area using base and height</item>  
 <item>Find area using all sides</item>  
 <item>Find area using trigonometry</item>  
 <item>Find perimeter using all sides</item>  
 <item>Find perimeter using Law of Cosines</item>  
 </string-array>  
 <string-array name="rectangle\_formula\_arrays">  
 <item>Find area using length and width</item>  
 <item>Find area using length and diagonal</item>  
 <item>Find perimeter using width and height</item>  
 </string-array>  
</resources>