

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res-auto"
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   xmlns:tools="http://schemas.android.com/tools"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   tools:context=".MainActivity">
8
9   <TextView...>
10
22
23   <TextView...>
24
36
37   <TextView...>
38
51
52   <TextView...>
53
63
64   <TextView...>
65
77
78   <RadioGroup...>
79
114
115   <CheckBox...>
116
126
127   <EditText...>
128
142
143   <CheckBox...>
144
153
154   <EditText...>
155
170
171   <CheckBox...>
172
181
182   <EditText...>
183
197
198   <TextView...>
199
213
214   <Button...>
215
226
227   <Button...>
228
240
241   <Button...>
242
253
254 </androidx.constraintlayout.widget.ConstraintLayout>
```

activity_main.xml 🔍 🔄 🌙

Date Time

Laboratorio N°3

Cédula: 8-843-1065
Nombre: Pablo Testa

☐ Equilatero ☐ Isoseles ☐ Escaleno

☐ Lado 1 Ingrese primer valor

☐ Lado 2 Ingrese segundo valor

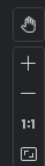
☐ Lado 3 Ingrese tercer valor

Resultado



androidx.constraintlayout.widget.ConstraintLayout

Component Tree Palette



Attributes



<> activity_main.xml

© MainActivity.java ×



```
1 package com.example.calculoareatriangulo_testap;
2
3 > import ...
18
19 </> public class MainActivity extends AppCompatActivity {
20     2 usages
21     TextView date, time, resultado;
22     3 usages
23     Calendar calendar;
24     3 usages
25     String currentDate, currentTime, val1, val2, val3, result;
26     13 usages
27     CheckBox l1, l2, l3;
28     2 usages
29     RadioGroup radiogroup;
30     10 usages
31     RadioButton rb1, rb2, rb3;
32     2 usages
33     Button b1, b2, b3;
34     24 usages
35     EditText ed1, ed2, ed3;
36     3 usages
37     Double a, b, c, s, area;
38
39     @Override
40     31 < > protected void onCreate(Bundle savedInstanceState) {...}
41
42     @Override
43     113 < > protected void onStart() {...}
44
45     @Override
46     123 < > protected void onPause() {...}
47
48     129 > private void cerrar() {...}
49
50     135 > private void closeKeyboard() {...}
51
52     142 > private boolean operacion() {...}
53
54     176 > private void radio1(){...}
55
56     193 > private void radio2(){...}
57
58     210 > private void radio3(){...}
59
60     227 > private void checkbox1() {...}
61
62     241 > private void checkbox2() {...}
63
64     255 > private void checkbox3() {...}
65
66     269 > private boolean validacion() {...}
67
68     338 > private void reset() {...}
69
70     360 }
```

13 88 ^ v



```
CalculoAreaTriangulo_TestaP ▾ Version control ▾ Pixel 7 Pro API 34 ▾ app ▾ ▶ 🔍 ⋮  
activity_main.xml MainActivity.java x  
21 Calendar calendar;  
3 usages  
22 String currentDate, currentTime, val1, val2, val3, result;  
13 usages  
23 CheckBox l1, l2, l3;  
2 usages  
24 RadioGroup radiogroup;  
10 usages  
25 RadioButton rb1, rb2, rb3;  
2 usages  
26 Button b1, b2, b3;  
24 usages  
27 EditText ed1, ed2, ed3;  
3 usages  
28 Double a, b, c, s, area;  
29  
30 @Override  
31 protected void onCreate(Bundle savedInstanceState) {  
32     super.onCreate(savedInstanceState);  
33     setContentView(R.layout.activity_main);  
34  
35     date = findViewById(R.id.date);  
36     time = findViewById(R.id.time);  
37     calendar = Calendar.getInstance();  
38     currentDate = DateFormat.getDateInstance(DateFormat.DEFAULT).format(calendar.getTime());  
39     currentTime = DateFormat.getTimeInstance(DateFormat.SHORT).format(calendar.getTime());  
40     date.setText(currentDate);  
41     time.setText(currentTime);  
42     ed1 = findViewById(R.id.ed1);  
43     ed2 = findViewById(R.id.ed2);  
44     ed3 = findViewById(R.id.ed3);  
45     rb1 = findViewById(R.id.rb1);  
46     rb2 = findViewById(R.id.rb2);  
47     rb3 = findViewById(R.id.rb3);  
48     l1 = findViewById(R.id.l1);  
49     l2 = findViewById(R.id.l2);  
50     l3 = findViewById(R.id.l3);  
51     b1 = findViewById(R.id.b1);  
52     b2 = findViewById(R.id.b2);  
53     b3 = findViewById(R.id.b3);  
54     radiogroup = findViewById(R.id.radiogroup);  
55     resultado = findViewById(R.id.resultado);  
56     b1.setOnClickListener(new View.OnClickListener() {...});  
64     rb1.setOnClickListener(new View.OnClickListener() {...});  
70     rb2.setOnClickListener(new View.OnClickListener() {...});  
76     rb3.setOnClickListener(new View.OnClickListener() {...});  
82     l1.setOnClickListener(new View.OnClickListener() {...});  
88     l2.setOnClickListener(new View.OnClickListener() {...});  
94     l3.setOnClickListener(new View.OnClickListener() {...});  
100     b2.setOnClickListener(new View.OnClickListener() {...});  
106     b3.setOnClickListener(new View.OnClickListener() {...});  
112 }  
113 @Override  
114 protected void onStart() {...}  
123 @Override  
124 protected void onPause() {...}  
1 usage  
129 private void cerrar() {...}  
1 usage  
135 private void closeKeyboard() {...}  
1 usage  
142 private boolean operacion() {...}  
1 usage  
176 private void radio1() {...}  
1 usage  
193 private void radio2() {...}  
1 usage  
210 private void radio3() {...}  
1 usage  
227 private void checkbox1() {...}  
1 usage  
241 private void checkbox2() {...}  
1 usage  
255 private void checkbox3() {...}  
1 usage  
269 private boolean validacion() {...}  
1 usage
```

```
1 package com.example.calculoareatriangulo_testap;
2
3 > import ...
18
19 </> public class MainActivity extends AppCompatActivity {
20     TextView date, time, resultado;
21     Calendar calendar;
22     String currentDate, currentTime, val1, val2, val3, result;
23     CheckBox l1, l2, l3;
24     RadioGroup radiogroup;
25     RadioButton rb1, rb2, rb3;
26     Button b1, b2, b3;
27     EditText ed1, ed2, ed3;
28     Double a, b, c, s, area;
29
30     @Override
31 > protected void onCreate(Bundle savedInstanceState) {...}
112
113 > protected void onStart() {
114     super.onStart();
115     Toast.makeText(context: MainActivity.this, text: "Iniciando", Toast.LENGTH_SHORT).show(); // La actividad es
116     Log.d(tag: "Date and Time", msg: "La fecha y hora actual es: " + currentDate + " " + currentTime);
117     Log.d(tag: "Start", msg: "Iniciando // La actividad está a punto de recurse visible");
118     Log.d(tag: "Materia", msg: "Materia: Programacion VI - Asignacion N2");
119     Log.d(tag: "Nombre", msg: "Estudiante: Pablo Testa");
120     Log.d(tag: "Cedula", msg: "Cedula: 8-843-1065");
121 }
122
123 > protected void onPause() {
124     super.onPause();
125     Toast.makeText(context: MainActivity.this, text: "en Pausa", Toast.LENGTH_SHORT).show(); // Enfocarse en otra
126     Log.d(tag: "Pause", msg: "es Pausa // Enfocarse en otra actividad (esta actividad está a punto de ser\detenida");
127 }
128
129 private void cerrar() {
130     Toast.makeText(context: MainActivity.this, text: "Cerrando", Toast.LENGTH_SHORT).show(); // La actividad ya
131     Log.d(tag: "Cerrar", msg: "Cerrando // La actividad ya no es visible (ahora esta \detenida\").");
132     MainActivity.this.finish();
133     System.exit(status: 0);
134 }
135
136 private void closeKeyboard() {
137     View view = this.getCurrentFocus();
138     if (view != null) {
139         InputMethodManager imm = (InputMethodManager) getSystemService(Context.INPUT_METHOD_SERVICE);
140         imm.hideSoftInputFromWindow(view.getWindowToken(), flags: 0);
141     }
142 }
143
144 private boolean operacion() {...}
145
146 private void radio1() {...}
147
148 private void radio2() {...}
149
150 private void radio3() {...}
151
152 private void checkbox1() {...}
153
154 private void checkbox2() {...}
155
156 private void checkbox3() {...}
157
158 private boolean validacion() {...}
159
160 private void reset() {...}
161 }
```

```
172     }
173     return false;
174 }
175 1 usage
176 private void radio1(){
177     if (rb1.isChecked()) {
178         rb1.setError(null);
179         rb2.setError(null);
180         rb3.setError(null);
181         l1.setError(null);
182         l2.setError(null);
183         l3.setError(null);
184         ed1.setError(null);
185         ed2.setError(null);
186         ed3.setError(null);
187         ed1.setText(null);
188         ed2.setText(null);
189         ed3.setText(null);
190         resultado.setText(null);
191     }
192 1 usage
193 private void radio2(){
194     if (rb2.isChecked()) {
195         rb1.setError(null);
196         rb2.setError(null);
197         rb3.setError(null);
198         l1.setError(null);
199         l2.setError(null);
200         l3.setError(null);
201         ed1.setError(null);
202         ed2.setError(null);
203         ed3.setError(null);
204         ed1.setText(null);
205         ed2.setText(null);
206         ed3.setText(null);
207         resultado.setText(null);
208     }
209 1 usage
210 private void radio3(){
211     if (rb3.isChecked()) {
212         rb1.setError(null);
213         rb2.setError(null);
214         rb3.setError(null);
215         l1.setError(null);
216         l2.setError(null);
217         l3.setError(null);
218         ed1.setError(null);
219         ed2.setError(null);
220         ed3.setError(null);
221         ed1.setText(null);
222         ed2.setText(null);
223         ed3.setText(null);
224         resultado.setText(null);
225     }
226 1 usage
227 private void checkbox1() {
228     if (l1.isChecked()) {
229         ed1.setEnabled(true);
230         ed1.setError(null);
231         l1.setError(null);
232     } else if (l11.isChecked()) {
233         ed1.setText(null);
234         ed1.setError(null);
235         ed1.setEnabled(false);
236         l1.setError(null);
237         Toast.makeText(context, MainActivity.this, text: "Seleccione el lado 1", Toast.LENGTH_LONG).show();
238         Log.e( tag: "Validate", msg: "Required || Seleccione el lado 1");
239     }
240 1 usage
241 private void checkbox2() {...}
242 1 usage
243 private void checkbox3() {...}
```

```
CalculoAreaTriangulo_TestaP v Version control v Pixel 7 Pro API 34 v app v     
```

activity_main.xml MainActivity.java x

3 usages 13 88 ^ v

```
28     Double a, b, c, s, area;
29
30     @Override
31 > protected void onCreate(Bundle savedInstanceState) {...}
112     @Override
113 > protected void onStart() {...}
122     @Override
123 > protected void onPause() {...}
1 usage
128 > private void cerrar() {...}
1 usage
134 > private void closeKeyboard() {...}
1 usage
141 > private boolean operacion() {...}
1 usage
175 > private void radio1(){...}
1 usage
192 > private void radio2(){...}
1 usage
209 > private void radio3(){...}
1 usage
226 private void checkbox1() {
227     if (l1.isChecked()) {
228         ed1.setEnabled(true);
229         ed1.setError(null);
230         l1.setError(null);
231     } else if (!l1.isChecked()) {
232         ed1.setText(null);
233         ed1.setError(null);
234         ed1.setEnabled(false);
235         l1.setError(null);
236         Toast.makeText(context: MainActivity.this, text: "Seleccione el lado 1", Toast.LENGTH_LONG).show();
237         Log.e(tag: "Validate", msg: "Required || Seleccione el lado 1");
238     }
239 }
1 usage
240 private void checkbox2() {
241     if (l2.isChecked()) {
242         ed2.setEnabled(true);
243         ed2.setError(null);
244         l2.setError(null);
245     } else if (!l2.isChecked()) {
246         ed2.setText(null);
247         ed2.setEnabled(false);
248         ed2.setError(null);
249         l2.setError(null);
250         Toast.makeText(context: MainActivity.this, text: "Seleccione el lado 2", Toast.LENGTH_LONG).show();
251         Log.e(tag: "Validate", msg: "Required || Seleccione el lado 2");
252     }
253 }
1 usage
254 private void checkbox3() {
255     if (l3.isChecked()) {
256         ed3.setEnabled(true);
257         ed3.setError(null);
258         l3.setError(null);
259     } else if (!l3.isChecked()) {
260         ed3.setText(null);
261         ed3.setEnabled(false);
262         ed3.setError(null);
263         l3.setError(null);
264         Toast.makeText(context: MainActivity.this, text: "Seleccione el lado 3", Toast.LENGTH_LONG).show();
265         Log.e(tag: "Validate", msg: "Required || Seleccione el lado 3");
266     }
267 }
1 usage
268 > private boolean validacion() {...}
1 usage
337 > private void reset() {...}
359 }
```



activity_main.xml

MainActivity.java



```

268     private boolean validacion() {
269         if (!rb1.isChecked() && !rb2.isChecked() && !rb3.isChecked()) {
270             rb1.setError("Required");
271             rb2.setError("Required");
272             rb3.setError("Required");
273             Toast.makeText( context: MainActivity.this, text: "Seleccione un tipo de triangulo", Toast.LENGTH_LONG).show();
274             Log.e( tag: "Validate", msg: "Required || Seleccione un tipo de triangulo");
275             return false;
276         }
277         if (!l1.isChecked() && !l2.isChecked() && !l3.isChecked()) {
278             l1.setError("");
279             l2.setError("");
280             l3.setError("");
281             Toast.makeText( context: MainActivity.this, text: "Seleccione una opcion", Toast.LENGTH_LONG).show();
282             Log.e( tag: "Validate", msg: "Required // Seleccione una opcion");
283         }
284         if (ed1.length() == 0 && ed2.length() == 0 && ed3.length() == 0) {
285             ed1.setError("Required");
286             ed2.setError("Required");
287             ed3.setError("Required");
288             Toast.makeText( context: MainActivity.this, text: "Ingrese informacion en los campos requeridos", Toast.LENGTH_LONG).show();
289             Log.e( tag: "Validate", msg: "Required // Ingrese informacion para todos los campos requeridos");
290             return false;
291         }
292         if (ed1.length() == 0 && ed2.length() == 0) {
293             ed1.setError("Required");
294             ed2.setError("Required");
295             Toast.makeText( context: MainActivity.this, text: "Ingrese informacion en los campos requeridos", Toast.LENGTH_LONG).show();
296             Log.e( tag: "Validate", msg: "Required // Ingrese informacion para todos los campos requeridos");
297             return false;
298         }
299         if (ed1.length() == 0 && ed3.length() == 0) {
300             ed1.setError("Required");
301             ed3.setError("Required");
302             Toast.makeText( context: MainActivity.this, text: "Ingrese informacion en los campos requeridos", Toast.LENGTH_LONG).show();
303             Log.e( tag: "Validate", msg: "Required // Ingrese informacion para todos los campos requeridos");
304             return false;
305         }
306         if (ed2.length() == 0 && ed3.length() == 0) {
307             ed2.setError("Required");
308             ed3.setError("Required");
309             Toast.makeText( context: MainActivity.this, text: "Ingrese informacion en los campos requeridos", Toast.LENGTH_LONG).show();
310             Log.e( tag: "Validate", msg: "Required // Ingrese informacion para todos los campos requeridos");
311             return false;
312         }
313         if (ed1.length() == 0) {
314             ed1.setError("Required");
315             Toast.makeText( context: MainActivity.this, text: "Ingrese informacion en el segundo campo", Toast.LENGTH_LONG).show();
316             Log.e( tag: "Validate", msg: "Required // Ingrese informacion en el segundo campo");
317             return false;
318         }
319         if (ed2.length() == 0) {
320             ed2.setError("Required");
321             Toast.makeText( context: MainActivity.this, text: "Ingrese informacion en el segundo campo", Toast.LENGTH_LONG).show();
322             Log.e( tag: "Validate", msg: "Required // Ingrese informacion en el segundo campo");
323             return false;
324         }
325         if (ed3.length() == 0) {
326             ed3.setError("Required");
327             Toast.makeText( context: MainActivity.this, text: "Ingrese informacion en el tercer campo", Toast.LENGTH_LONG).show();
328             Log.e( tag: "Validate", msg: "Required // Ingrese informacion en el tercer campo");
329             return false;
330         } else {
331             Toast.makeText( context: MainActivity.this, text: "Validando la informacion", Toast.LENGTH_LONG).show();
332             Log.e( tag: "Validate", msg: "Validando // Validando los campos con la informacion suministrada ");
333         }
334         operacion();
335         return false;
336     }
337     1 usage
338     private void reset() {...}
339 }

```



```
1 package com.example.calculoareatriangulo_testap;
2
3 > import ...
18
19 </> public class MainActivity extends AppCompatActivity {
20     TextView date, time, resultado;
21     Calendar calendar;
22     String currentDate, currentTime, val1, val2, val3, result;
23     CheckBox l1, l2, l3;
24     RadioGroup radiogroup;
25     RadioButton rb1, rb2, rb3;
26     Button b1, b2, b3;
27     EditText ed1, ed2, ed3;
28     Double a, b, c, s, area;
29
30     @Override
31 > protected void onCreate(Bundle savedInstanceState) {...}
112
113 > @Override protected void onStart() {...}
122
123 > @Override protected void onPause() {...}
128 > private void cerrar() {...}
134 > private void closeKeyboard() {...}
141 > private boolean operacion() {...}
175 > private void radio1(){...}
192 > private void radio2(){...}
209 > private void radio3(){...}
226 > private void checkbox1() {...}
240 > private void checkbox2() {...}
254 > private void checkbox3() {...}
268 > private boolean validacion() {...}
337
338     private void reset() {
339         l1.setChecked(false);
340         l2.setChecked(false);
341         l3.setChecked(false);
342         rb1.setError(null);
343         rb2.setError(null);
344         rb3.setError(null);
345         l1.setError(null);
346         l2.setError(null);
347         l3.setError(null);
348         ed1.setError(null);
349         ed2.setError(null);
350         ed3.setError(null);
351         ed1.setText(null);
352         ed2.setText(null);
353         ed3.setText(null);
354         ed1.setEnabled(false);
355         ed2.setEnabled(false);
356         ed3.setEnabled(false);
357         resultado.setText(null);
358         radiogroup.clearCheck();
359     }
}
```


3:59



Feb 10, 2024

3:59 AM

Laboratorio N°3

Cédula: 8-843-1065

Nombre: Pablo Testa

☐ Equilatero ☐ Isoseles ☐ Escaleno☐ Lado 1 Ingrese primer valor☐ Lado 2 Ingrese segundo valor☐ Lado 3 Ingrese tercer valor

Resultado

Reset

Operation

Close

3:59



Feb 10, 2024

3:59 AM

Laboratorio N°3

Cédula: 8-843-1065

Nombre: Pablo Testa

☐ Equilatero ☐ Isoseles ☐ Escalen

☐ Lado 1 Ingrese primer valor

☐ Lado 2 Ingrese segundo valor

☐ Lado 3 Ingrese tercer valor

Resultado

Reset

Operation

Close

4:00



Feb 10, 2024

3:59 AM

Laboratorio N°3

Cédula: 8-843-1065

Nombre: Pablo Testa

☒ Equilatero ☐ Isoseles ☐ Escaleno☒ Lado 1 ☒ Lado 2 ☒ Lado 3

El area del triangulo equilatero es:
11.0 cm²

Reset

Operation

Close



Validando la informacion

4:00



Feb 10, 2024

4:00 AM

Laboratorio N°3

Cédula: 8-843-1065

Nombre: Pablo Testa

☐ Equilatero ☐ Isoseles ☐ Escaleno☐ Lado 1 Ingrese primer valor☐ Lado 2 Ingrese segundo valor☐ Lado 3 Ingrese tercer valor

Resultado

Reset

Operation

Close



Cerrando