

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
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   android:textAlignment="inherit"
8   tools:context=".MainActivity">
9
10  <TextView...>
11
12
13
14
15
16
17
18
19
20
21
22
23  <TextView...>
24
25
26
27
28
29
30
31
32
33
34
35
36
37  <TextView...>
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52  <TextView...>
53
54
55
56
57
58
59
60
61
62
63
64  <TextView...>
65
66
67
68
69
70
71
72
73
74
75
76
77  <TextView...>
78
79
80
81
82
83
84
85
86
87
88
89  <EditText...>
90
91
92
93  <EditText...>
94
95
96
97
98
99
100
101
102
103
104
105
106  <EditText...>
107
108
109
110
111
112
113
114
115
116
117
118  <Button...>
119
120
121
122
123  <Button...>
124
125
126
127
128
129
130
131
132
133
134
135  <Button...>
136
137
138
139
140
141
142
143
144
145
146
147  <TextView...>
148
149
150
151  <TextView...>
152
153
154
155
156
157
158
159  <TextView...>
160
161
162
163  <TextView...>
164
165
166
167
168
169
170
171  <TextView...>
172
173
174
175
176
177
178
179  <TextView...>
180
181
182
183
184
185
186
187  <TextView...>
188
189
190
191
192
193
194
195  <TextView...>
196
197
198
199
200
201
202
203  <TextView...>
204
205
206
207
208
209
210
211  <TextView...>
212
213
214
215
216
217
218
219  <TextView...>
220
221
222
223
224
225
226
227  <TextView...>
228
229
230
231
232 </androidx.constraintlayout.widget.ConstraintLayout>
```



activity_main.xml

MainActivity.java



1 package com.example.quiz3_testap;

2



3 > import ...

16

17 <> public class MainActivity extends AppCompatActivity {

2 usages

18 private TextView date, time, resultado;

12 usages

19 private EditText ed1, ed2, ed3;

3 usages

20 private Calendar calendar;

3 usages

21 private String currentDate, currentTime, val1, val2, val3, result;

2 usages

22 private Double a, b, e, x;

2 usages

23 private Button operation, reset, close;

24

25 @Override

26 ⌕ > protected void onCreate(Bundle savedInstanceState) {...}

1 usage

58 > private boolean validation() {...}

1 usage

113 > private boolean operation() {...}

1 usage

127 > private void cerrar() {...}

1 usage

133 > private void closeKeyboard() {...}

140 @Override

141 ⌕ > protected void onStart() {...}

1 usage

156 > private void reset() {...}

165 @Override

166 ⌕ > protected void onPause() {...}

171 }





activity_main.xml

MainActivity.java



```
1 package com.example.quiz3_testap;
2
3 > import ...
4
5
6
7
8
9
10
11
12
13
14
15
16
17 <> public class MainActivity extends AppCompatActivity {
18     2 usages
19     private TextView date, time, resultado;
20     12 usages
21     private EditText ed1, ed2, ed3;
22     3 usages
23     private Calendar calendar;
24     3 usages
25     private String currentDate, currentTime, val1, val2, val3, result;
26     2 usages
27     private Double a, b, e, x;
28     2 usages
29     private Button operation, reset, close;
30
31     @Override
32     protected void onCreate(Bundle savedInstanceState) {
33         super.onCreate(savedInstanceState);
34         setContentView(R.layout.activity_main);
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         operation = findViewById(R.id.operation);
46         reset = findViewById(R.id.reset);
47         close = findViewById(R.id.close);
48         resultado = findViewById(R.id.resultado);
49         operation.setOnClickListener(new View.OnClickListener() {
50             @Override
51             public void onClick(View View) {
52                 validation();
53                 closeKeyboard();
54             }
55         });
56         close.setOnClickListener(new View.OnClickListener() {...});
57     }
58     1 usage
59     private boolean validation() {...}
60     1 usage
61     private boolean operation() {...}
62     1 usage
63     private void cerrar() {...}
64     1 usage
65     private void closeKeyboard() {...}
66     @Override
67     protected void onStart() {...}
68     1 usage
69     private void reset() {...}
70     @Override
71     protected void onPause() {...}
72 }
```





activity_main.xml

MainActivity.java



```
17 </> public class MainActivity extends AppCompatActivity {  
    2 usages  
18     private TextView date, time, resultado;  
    12 usages  
19     private EditText ed1, ed2, ed3;  
    3 usages  
20     private Calendar calendar;  
    3 usages  
21     private String currentDate, currentTime, val1, val2, val3, result;  
    2 usages  
22     private Double a, b, c, x;  
    2 usages  
23     private Button operation, reset, close;  
24  
25     @Override  
26     protected void onCreate(Bundle savedInstanceState) {...}  
    1 usage  
58     private boolean validation() {  
59         if (ed1.length() == 0 && ed2.length() == 0 && ed3.length() == 0) {  
60             ed1.setError("Required");  
61             ed2.setError("Required");  
62             ed3.setError("Required");  
63             Toast.makeText( context: MainActivity.this, text: "Ingrese informacion en los campos requeridos", Toast.LENGTH_SHORT).show();  
64             Log.e( tag: "Validate", msg: "Required // Ingrese informacion para todos los campos requeridos");  
65             return false;  
66         }  
67         if (ed1.length() == 0 && ed2.length() == 0) {  
68             ed1.setError("Required");  
69             ed2.setError("Required");  
70             Toast.makeText( context: MainActivity.this, text: "Ingrese informacion en los campos requeridos", Toast.LENGTH_SHORT).show();  
71             Log.d( tag: "Validate", msg: "Required // Ingrese informacion para todos los campos requeridos");  
72             return false;  
73         }  
74         if (ed1.length() == 0 && ed3.length() == 0) {  
75             ed1.setError("Required");  
76             ed3.setError("Required");  
77             Toast.makeText( context: MainActivity.this, text: "Ingrese informacion en los campos requeridos", Toast.LENGTH_SHORT).show();  
78             Log.d( tag: "Validate", msg: "Required // Ingrese informacion para todos los campos requeridos");  
79             return false;  
80         }  
81         if (ed2.length() == 0 && ed3.length() == 0) {  
82             ed2.setError("Required");  
83             ed3.setError("Required");  
84             Toast.makeText( context: MainActivity.this, text: "Ingrese informacion en los campos requeridos", Toast.LENGTH_SHORT).show();  
85             Log.d( tag: "Validate", msg: "Required // Ingrese informacion para todos los campos requeridos");  
86             return false;  
87         }  
88         if (ed1.length() == 0) {  
89             ed1.setError("Required");  
90             Toast.makeText( context: MainActivity.this, text: "Ingrese informacion en el segundo campo", Toast.LENGTH_SHORT).show();  
91             Log.d( tag: "Validate", msg: "Required // Ingrese informacion en el segundo campo");  
92             return false;  
93         }  
94         if (ed2.length() == 0) {  
95             ed2.setError("Required");  
96             Toast.makeText( context: MainActivity.this, text: "Ingrese informacion en el segundo campo", Toast.LENGTH_SHORT).show();  
97             Log.d( tag: "Validate", msg: "Required // Ingrese informacion en el segundo campo");  
98             return false;  
99         }  
100        if (ed3.length() == 0) {  
101            ed3.setError("Required");  
102            Toast.makeText( context: MainActivity.this, text: "Ingrese informacion en el tercer campo", Toast.LENGTH_SHORT).show();  
103            Log.d( tag: "Validate", msg: "Required // Ingrese informacion en el tercer campo");  
104            return false;  
105        }  
106        else {  
107            Toast.makeText( context: MainActivity.this, text: "Validando la informacion", Toast.LENGTH_LONG).show();  
108            Log.e( tag: "Validate", msg: "Validando // Validando los campos con la informacion suministrada ");  
109        }  
110        operation();  
111        return false;  
112    }  
    1 usage  
113    private boolean operation() {...}  
    1 usage  
127    private void cerrar() {...}
```

```
1 <?xml version="1.0" encoding="utf-8" >
2 <androidx.constraintlayout.widget.ConstraintLayout
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     android:textAlignment="inherit"
8     tools:context=".MainActivity">
9
10 >
11 </androidx.constraintlayout.widget.ConstraintLayout>
```

Quiz N°3

Cédula: 8-843-1065
Nombre: Pablo Testa

Un vendedor tiene # cajas de frutas, cada caja contiene # frutas. Si cada fruta se vende a \$ precio, ¿cuánto dinero gana al vender todas las frutas?

¿Cuántas cajas de frutas tiene?

Ingrese primer valor

¿Cuántas frutas contiene cada caja?

Ingrese segundo valor

¿Qué precio tienen las frutas por unidad?

Ingrese tercer valor

Resultado

Close Operation Reset

```

1 package com.example.quiz3_testap;
2
3 > import ...
16
17 <> public class MainActivity extends AppCompatActivity {
18     2 usages
19     private TextView date, time, resultado;
20     12 usages
21     private EditText ed1, ed2, ed3;
22     3 usages
23     private Calendar calendar;
24     3 usages
25     private String currentDate, currentTime, val1, val2, val3, result;
26     2 usages
27     private Double a, b, c, x;
28     2 usages
29     private Button operation, reset, close;
30
31     @Override
32     protected void onCreate(Bundle savedInstanceState) {...}
33     1 usage
34
35     private boolean validation() {...}
36     1 usage
37
38     private boolean operation() {
39         val1 = ed1.getText().toString();
40         val2 = ed2.getText().toString();
41         val3 = ed3.getText().toString();
42         a = Double.parseDouble(val1);
43         b = Double.parseDouble(val2);
44         c = Double.parseDouble(val3);
45
46         x = (a*b)*c;
47         result = "El precio final es: \n$ " + x;
48         resultado.setText(result);
49
50         return false;
51     }
52     1 usage
53
54     private void cerrar() {
55         Toast.makeText(context: MainActivity.this, text: "Cerrando", Toast.LENGTH_SHORT).show(); // La actividad ya
56         Log.d(tag: "Cerrar", msg: "Cerrando // La actividad ya no es visible (ahora esta \"detenida\").");
57         MainActivity.this.finish();
58         System.exit(status: 0);
59     }
60     1 usage
61
62     private void closeKeyboard() {
63         View view = this.getCurrentFocus();
64         if (view != null) {
65             InputMethodManager imm = (InputMethodManager) getSystemService(Context.INPUT_METHOD_SERVICE);
66             imm.hideSoftInputFromWindow(view.getWindowToken(), flags: 0);
67         }
68     }
69
70     @Override
71     protected void onStart() {
72         super.onStart();
73         Toast.makeText(context: MainActivity.this, text: "Iniciando", Toast.LENGTH_SHORT).show(); // La actividad es
74         Log.d(tag: "Date and Time", msg: "La fecha y hora actual es: " + currentDate + " " + currentTime);
75         Log.d(tag: "Start", msg: "Iniciando // La actividad está a punto de recurse visible");
76         Log.d(tag: "Materia", msg: "Materia: Programacion VI - Quiz N3");
77         Log.d(tag: "Nombre", msg: "Estudiante: Pablo Testa");
78         Log.d(tag: "Cedula", msg: "Cedula: 8-843-1065");
79         reset.setOnClickListener(new View.OnClickListener() {...});
80     }
81     1 usage
82
83     private void reset() {
84         ed1.setText(null);
85         ed2.setText(null);
86         ed3.setText(null);
87         ed1.setError(null);
88         ed2.setError(null);
89         ed3.setError(null);
90         resultado.setText(null);
91     }
92
93     @Override
94     protected void onPause() {...}
95
96 }

```


Feb 8, 2024

1:21 AM

Quiz N°3

Cédula: 8-843-1065

Nombre: Pablo Testa

Un vendedor tiene # cajas de frutas, cada caja contiene # frutas. Si cada fruta se vende a \$ precio, ¿cuánto dinero gana al vender todas las frutas?

¿Cuántas cajas de frutas tiene?

Ingrese primer valor

¿Cuántas frutas contiene cada caja?

Ingrese segundo valor

¿Qué precio tienen las frutas por unidad?

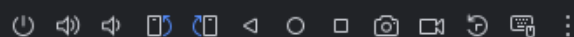
Ingrese tercer valor

Resultado

Close

Operation

Reset



Feb 8, 2024

1:21 AM

Quiz N°3

Cédula: 8-843-1065

Nombre: Pablo Testa

Un vendedor tiene # cajas de frutas, cada caja contiene # frutas. Si cada fruta se vende a \$ precio, ¿cuánto dinero gana al vender todas las frutas?

¿Cuántas cajas de frutas tiene?

5

¿Cuántas frutas contiene cada caja?

30

¿Qué precio tienen las frutas por unidad?

1.50

El precio final es:
\$ 225.0

Close

Operation

Reset

Validando la informacion

+

-

1:1



Feb 8, 2024

1:21 AM

Quiz N°3

Cédula: 8-843-1065

Nombre: Pablo Testa

Un vendedor tiene # cajas de frutas, cada caja contiene # frutas. Si cada fruta se vende a \$ precio, ¿cuánto dinero gana al vender todas las frutas?

¿Cuántas cajas de frutas tiene?

Ingrese primer valor



¿Cuántas frutas contiene cada caja?

Ingrese segundo valor



¿Qué precio tienen las frutas por unidad?

Ingrese tercer valor



Required

Resultado

Close

Operation

Reset



1:1

