



Universidad Nacional Autónoma de México

Alumno: Juan Manuel Martínez Chávez

Carrera: Licenciatura en informática

Materia: Programación de dispositivos móviles

Asesor: Juan Manuel Martínez Fernández

Con base en la actividad anterior

a) Convierte los botones en componentes, en la clase CalcButton.

b) La clase CalcButton deberá recibir como parámetros, el texto del botón, el callback de la función al presionar el botón, el color del texto, y el color del fondo del botón.

c) el callback deberá hacerse a la función onPressed con el texto del botón como parámetro.

c) Ejecuta la aplicación.

d) Crea un Gist público en Github con el contenido del archivo main.dart que modificaste.

e) Anota tus respuestas, la liga del Gist público y las capturas de pantalla en un archivo PDF.

Sube el archivo PDF a la plataforma

```

1 import 'package:flutter/material.dart';
2
3 void main() {
4   runApp(MaterialApp(
5     home: CalculatorApp(),
6   ));
7 }
8
9 class CalculatorApp extends StatefulWidget {
10   @override
11   CalculatorAppState createState() => CalculatorAppState();
12 }
13
14 class CalculatorAppState extends State<CalculatorApp> {
15   double operand1 = 0.0;
16   double operand2 = 0.0;
17   String operation;
18   String calculation = "0";
19
20   void onPressedButton(String text) {
21     //Por hacer
22     print(text);
23   }
24
25   @override
26   Widget build(BuildContext context) {
27     return Scaffold(
28       body: Card(
29         child: Column(
30           mainAxisAlignment: MainAxisAlignment.center,
31           children: <Widget>[
32             Padding(
33               padding: EdgeInsets.all(16),
34             ),
35           ],
36         ),
37       ),
38     );
39   }
40 }

```

▶ RUN

Console 10 Documentation

```
32      padding(  
33        padding: EdgeInsets.all(16),  
34        child: Align(  
35          alignment: Alignment.centerRight,  
36          child: Text(calculation, style: TextStyle(fontSize: 30)),  
37        )),  
38      Row(  
39        mainAxisAlignment: MainAxisAlignment.spaceEvenly,  
40        children: <Widget>[  
41          //En lugar de tener el componente definido en esta clase,  
creamos una clase aparte. Esto facilita la reutilización de código y facilita la  
lectura.  
42          //Los parámetros que recibe ComponenteBoton son el texto que  
mostrará y el callback a la función que se ejecutará al oprimir el botón.  
43          //La función onPressed se ha definido en esta clase, no en  
la clase del componente.  
44          ComponenteBoton(text: "7", callback: (){  
45            onPressed("7");  
46          })),  
47          //En lugar de tener el componente definido en esta clase,  
creamos una clase aparte. Esto facilita la reutilización de código y facilita la  
lectura.  
49          //Los parámetros que recibe ComponenteBoton son el texto que  
mostrará y el callback a la función que se ejecutará al oprimir el botón.  
50          //La función onPressed se ha definido en esta clase, no en  
la clase del componente.  
51          ComponenteBoton(text: "8", callback: (){  
52            onPressed("8");  
53          })),  
54          //En lugar de tener el componente definido en esta clase,  
creamos una clase aparte. Esto facilita la reutilización de código y facilita la  
lectura.  
55          //En lugar de tener el componente definido en esta clase,  
creamos una clase aparte. Esto facilita la reutilización de código y facilita la  
lectura.
```

▶ RUN

Console 10 Documentation

**DartPad**

<> New Pad

↺ Reset

☰ Format

⬇ Install SDK

```
54      ...
55      //En lugar de tener el componente definido en esta clase,
    creamos una clase aparte. Esto facilita la reutilización de código y facilita la
    lectura.
56      //Los parámetros que recibe ComponenteBoton son el texto que
    mostrará y el callback a la función que se ejecutará al oprimir el botón.
57      //La función onPressedButton se ha definido en esta clase, no en
    la clase del componente.
58      ComponenteBoton(text: "9", callback: (){
59          onPressedButton("9");
60      }),
61
62      //En lugar de tener el componente definido en esta clase,
    creamos una clase aparte. Esto facilita la reutilización de código y facilita la
    lectura.
63      //Los parámetros que recibe ComponenteBoton son el texto que
    mostrará y el callback a la función que se ejecutará al oprimir el botón.
64      //La función onPressedButton se ha definido en esta clase, no en
    la clase del componente.
65      ComponenteBoton(text: "%", callback: (){
66          onPressedButton("%");
67      }),
68      ],
69
70 Row(
71     mainAxisAlignment: MainAxisAlignment.spaceEvenly,
72     children: <Widget>[
73         //En lugar de tener el componente definido en esta clase,
    creamos una clase aparte. Esto facilita la reutilización de código y facilita la
    lectura.
74         //Los parámetros que recibe ComponenteBoton son el texto que
    mostrará y el callback a la función que se ejecutará al oprimir el botón.
75         //La función onPressedButton se ha definido en esta clase, no en
    la clase del componente
```

RUN

Console 10 Documentation

```
75      //La función onPressed se ha definido en esta clase
    la clase del componente.
76      ComponenteBoton(text: "4", callback: (){
77          onPressed("4");
78      }),
79
80      //En lugar de tener el componente definido en esta clase,
    creamos una clase aparte. Esto facilita la reutilización de código y facilita la
    lectura.
81      //Los parámetros que recibe ComponenteBoton son el texto que
    mostrará y el callback a la función que se ejecutará al oprimir el botón.
82      //La función onPressed se ha definido en esta clase, no en
    la clase del componente.
83      ComponenteBoton(text: "5", callback: (){
84          onPressed("5");
85      }),
86
87      //En lugar de tener el componente definido en esta clase,
    creamos una clase aparte. Esto facilita la reutilización de código y facilita la
    lectura.
88      //Los parámetros que recibe ComponenteBoton son el texto que
    mostrará y el callback a la función que se ejecutará al oprimir el botón.
89      //La función onPressed se ha definido en esta clase, no en
    la clase del componente.
90      ComponenteBoton(text: "6", callback: (){
91          onPressed("6");
92      }),
93
94      //En lugar de tener el componente definido en esta clase,
    creamos una clase aparte. Esto facilita la reutilización de código y facilita la
    lectura.
95      //Los parámetros que recibe ComponenteBoton son el texto que
    mostrará y el callback a la función que se ejecutará al oprimir el botón.
96      //La función onPressed se ha definido en esta clase. no en
```

▶ RUN

Console 10 Documentation

```

96          //La función onPressed se ha definido en esta clase
    la clase del componente.
97          ComponenteBoton(text: "/", callback: (){
98              onPressed("/");
99          }),
100      ]),
101
102      Row(
103          mainAxisAlignment: MainAxisAlignment.spaceEvenly,
104          children: <Widget>[
105              //En lugar de tener el componente definido en esta clase,
    creamos una clase aparte. Esto facilita la reutilización de código y facilita la
    lectura.
106              //Los parámetros que recibe ComponenteBoton son el texto que
    mostrará y el callback a la función que se ejecutará al oprimir el botón.
107              //La función onPressed se ha definido en esta clase, no en
    la clase del componente.
108              ComponenteBoton(text: "1", callback: (){
109                  onPressed("1");
110              }),
111
112              //En lugar de tener el componente definido en esta clase,
    creamos una clase aparte. Esto facilita la reutilización de código y facilita la
    lectura.
113              //Los parámetros que recibe ComponenteBoton son el texto que
    mostrará y el callback a la función que se ejecutará al oprimir el botón.
114              //La función onPressed se ha definido en esta clase, no en
    la clase del componente.
115              ComponenteBoton(text: "2", callback: (){
116                  onPressed("2");
117              }),
118
119              //En lugar de tener el componente definido en esta clase,
    creamos una clase aparte. Esto facilita la reutilización de código y facilita la

```

▶ RUN

Console 10 Documentation

```

119      //En lugar de tener el componente definido en esta clase,
    creamos una clase aparte. Esto facilita la reutilización de código y facilita la
    lectura.
120      //Los parámetros que recibe ComponenteBoton son el texto que
    mostrará y el callback a la función que se ejecutará al oprimir el botón.
121      //La función onPressed se ha definido en esta clase, no en
    la clase del componente.
122      ComponenteBoton(text: "3", callback: (){
123        onPressed("3");
124      }),
125
126      //En lugar de tener el componente definido en esta clase,
    creamos una clase aparte. Esto facilita la reutilización de código y facilita la
    lectura.
127      //Los parámetros que recibe ComponenteBoton son el texto que
    mostrará y el callback a la función que se ejecutará al oprimir el botón.
128      //La función onPressed se ha definido en esta clase, no en
    la clase del componente.
129      ComponenteBoton(text: "-", callback: (){
130        onPressed("-");
131      }),
132    ]),
133
134 Row(
135   mainAxisAlignment: MainAxisAlignment.spaceEvenly,
136   children: <Widget>[
137     //En lugar de tener el componente definido en esta clase,
    creamos una clase aparte. Esto facilita la reutilización de código y facilita la
    lectura.
138     //Los parámetros que recibe ComponenteBoton son el texto que
    mostrará y el callback a la función que se ejecutará al oprimir el botón.
139     //La función onPressed se ha definido en esta clase, no en
    la clase del componente.
140     ComponenteBoton(text: "C", callback: (){

```

▶ RUN

Console 10 Documentation


```
139 //La función onPressed se ha definido en esta clase
    la clase del componente.
140     ComponenteBoton(text: "C", callback: (){
141         onPressed("C");
142     }),
143
144     //En lugar de tener el componente definido en esta clase,
    creamos una clase aparte. Esto facilita la reutilización de código y facilita la
    lectura.
145     //Los parámetros que recibe ComponenteBoton son el texto que
    mostrará y el callback a la función que se ejecutará al oprimir el botón.
146     //La función onPressed se ha definido en esta clase, no en
    la clase del componente.
147     ComponenteBoton(text: "0", callback: (){
148         onPressed("0");
149     }),
150
151     //En lugar de tener el componente definido en esta clase,
    creamos una clase aparte. Esto facilita la reutilización de código y facilita la
    lectura.
152     //Los parámetros que recibe ComponenteBoton son el texto que
    mostrará y el callback a la función que se ejecutará al oprimir el botón.
153     //La función onPressed se ha definido en esta clase, no en
    la clase del componente.
154     ComponenteBoton(text: "=", callback: (){
155         onPressed("=");
156     }),
157
158     //En lugar de tener el componente definido en esta clase,
    creamos una clase aparte. Esto facilita la reutilización de código y facilita la
    lectura.
159     //Los parámetros que recibe ComponenteBoton son el texto que
    mostrará y el callback a la función que se ejecutará al oprimir el botón.
160     //La función onPressed se ha definido en esta clase, no en
    la clase del componente.
```

▶ RUN

Console 10 Documentation



Privacy notice

Send feedback



Null Safety

```
160 //La función onPressed se ha definido en esta clase
    la clase del componente. ▶ RUN
161     ComponenteBoton(text: "+", callback: (){
162         onPressed("+");
163     }),
164   ],
165 ],
166 ],
167 ),
168 ),
169 );
170 }
171 }
172
173 class ComponenteBoton extends StatelessWidget {
174   final Function callback;
175   final String text;
176
177
178   //Este es el constructor del componente.
179   //Recibe dos parámetros, text y callback
180   //Para saber más al respecto https://dart.dev/guides/language/language-
   tour#constructors
181   const ComponenteBoton({Key key, @required this.text, @required this.callback}):
   super(key: key);
182
183   @override
184   Widget build(BuildContext context) {
185     return
186       Container(
187         padding: EdgeInsets.all(2.0),
188         child: RaisedButton(
189           child: Text(
190             //Este es el parámetro text
```

```
168     },
169   );
170 }
171 }
172
173 class ComponenteBoton extends StatelessWidget {
174   final Function callback;
175   final String text;
176
177   //Este es el constructor del componente.
178   //Recibe dos parámetros, text y callback
179   //Para saber más al respecto https://dart.dev/guides/language/language-tour#constructors
180   const ComponenteBoton({Key key, @required this.text, @required this.callback}):
181     super(key: key);
182
183   @override
184   Widget build(BuildContext context) {
185     return
186       Container(
187         padding: EdgeInsets.all(2.0),
188         child: RaisedButton(
189           child: Text(
190             //Este es el parámetro text
191             this.text,
192             style: TextStyle(color: Colors.white),
193           ),
194           color: Colors.black,
195           //Este es el parámetro callback
196           onPressed: callback,
197         ));
198   }
199 }
```

▶ RUN

Console 10 Documentation



Privacy notice

Send feedback



Null Safety

DartPad

<> New Pad

Reset

Format

Install SDK

Actividad 2

Samples

100
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199

```
    },  
  );  
}  
  
class ComponenteBoton extends StatelessWidget {  
  final Function callback;  
  final String text;  
  
  //Este es el constructor del componente.  
  //Recibe dos parámetros, text y callback  
  //Para saber más al respecto https://dart.dev/guides/language/language-  
  tour#constructors  
  const ComponenteBoton({Key key, @required this.text, @required this.callback}):  
    super(key: key);  
  
  @override  
  Widget build(BuildContext context) {  
    return  
      Container(  
        padding: EdgeInsets.all(2.0),  
        child: RaisedButton(  
          child: Text(  
            //Este es el parámetro text  
            this.text,  
            style: TextStyle(color: Colors.white),  
          ),  
          color: Colors.black,  
          //Este es el parámetro callback  
          onPressed: callback,  
        ));  
  }  
}
```

RUN

0

7

8

9

%

4

5

6

/

1

2

3

-

C

0

=

+

Console

Documentation

Privacy notice

Send feedback

Null Safety

1 issue

show

Based on Flutter 2.0.4 Dart SDK 2.12.2

Link: https://github.com/89451/Unidad3_actividad1.git