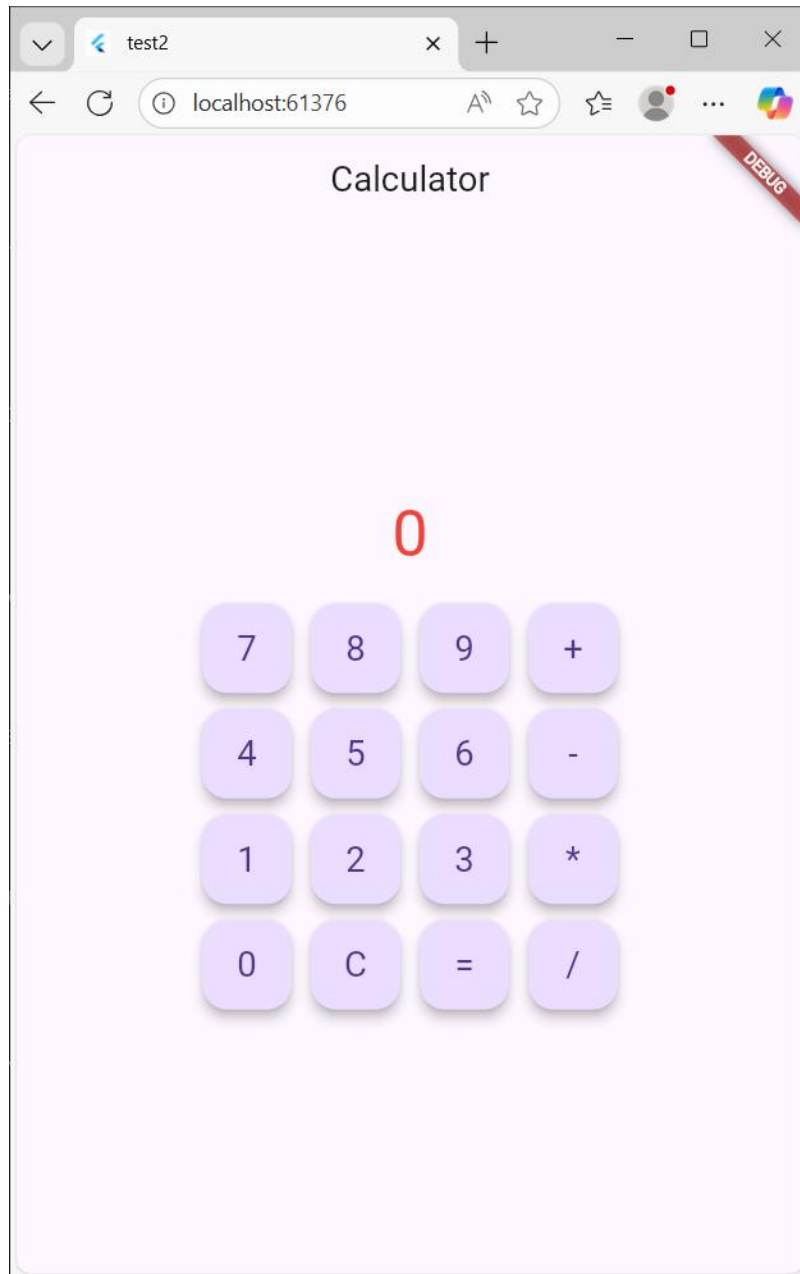
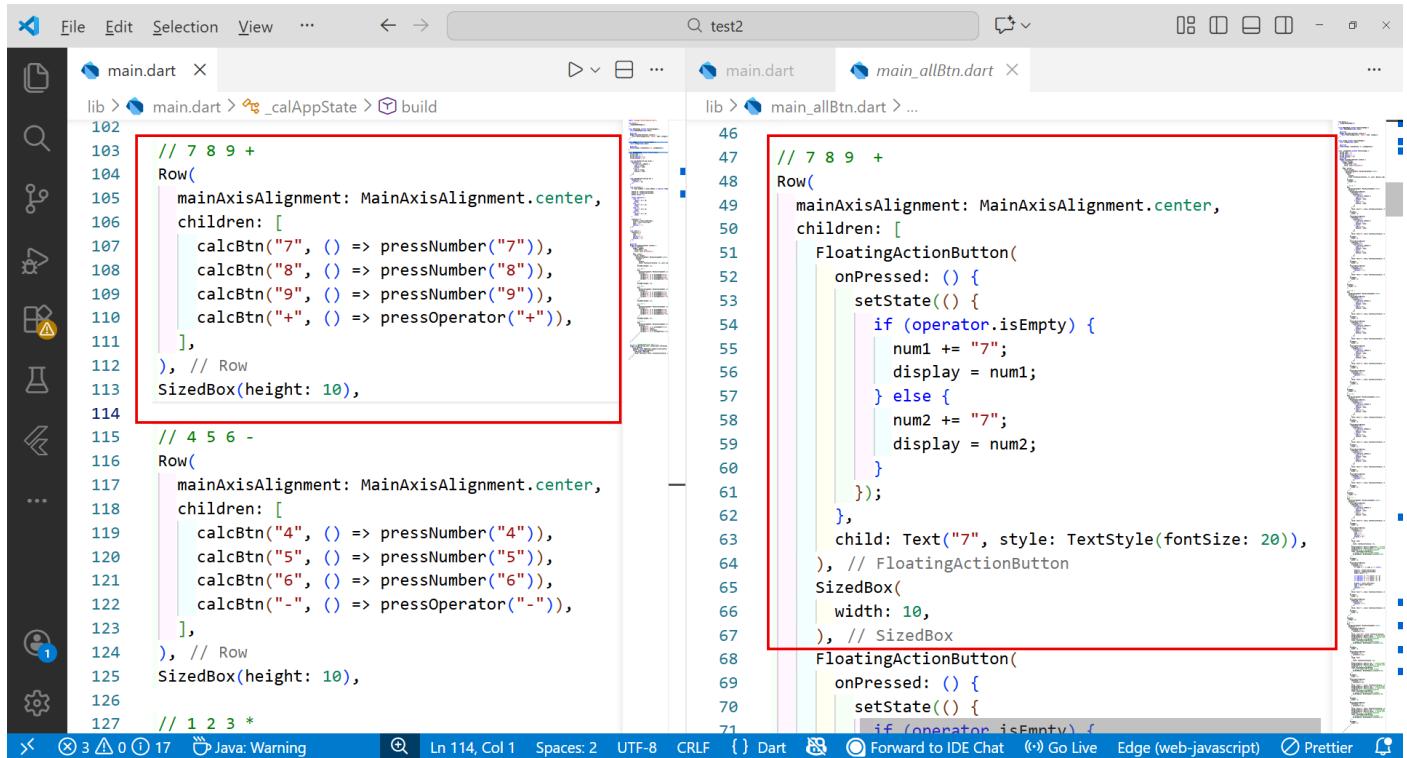


Workshop : Basic Calculator with Widget Button Template



จากเดิมที่ได้สร้างปุ่มบนเครื่องคิดเลข โดยสร้างของปุ่มแต่ละปุ่ม และแยกการทำงานของแต่ละปุ่ม ทำให้เกิดการเขียนโค้ดจำนวนหลายบรรทัด ซึ่งในกรณีที่ต้องการสร้างโดยทุกปุ่มมีลักษณะเหมือนกัน ก็สามารถสร้างปุ่มต่าง ๆ โดยสืบทอดคุณสมบัติจากต้นแบบได้ โดยการสร้างปุ่มบนเครื่องคิดเลข จาก Widget ต้นแบบ คือ (Model หรือ Template) ของปุ่ม ซึ่งสามารถใช้สร้างปุ่มต่าง ๆ ได้ ใน Flutter เรียกแนวคิดนี้ว่า Reusable Widget (วิดเจ็ตที่ใช้ซ้ำได้)

เปรียบเทียบการเขียนโปรแกรม



สร้างตัวแปร สำหรับเก็บค่าตัวเลขสำหรับการคำนวณ และ Template calcBtn ให้เป็น Template ต้นแบบ สำหรับสร้างปุ่ม

```
23 class _calAppState extends State<calApp> {
24   String num1 = "";
25   String num2 = "";
26   String operator = "";
27   String display = "0";
28
29   // ----- ปุ่มลัดสำหรับสร้าง FAB -----
30   Widget calcBtn(String text, VoidCallback onPressed) {
31     return Padding(
32       padding: const EdgeInsets.symmetric(horizontal: 6),
33       child: FloatingActionButton(
34         onPressed: onPressed,
35         child: Text(text, style: TextStyle(fontSize: 22)),
36       ), // FloatingActionButton
37     ); // Padding
38   }
39 }
```

สร้างฟังก์ชัน เพื่อรองรับการ -กดปุ่มตัวเลข และ ปุ่มเครื่องหมาย สำหรับการคำนวณ และ ปุ่ม C เพื่อ clear ค่าทั้งหมด

```
40     void pressNumber(String value) {
41         setState(() {
42             if (operator.isEmpty) {
43                 num1 += value;
44                 display = num1;
45             } else {
46                 num2 += value;
47                 display = num2;
48             }
49         });
50     }
51
52     void pressOperator(String op) {
53         setState(() {
54             operator = op;
55         });
56     }
57
58     void clear() {
59         setState(() {
60             num1 = "";
61             num2 = "";
62             operator = "";
63             display = "0";
64         });
65     }
```

```
67     void calculate() {
68         if (num1.isEmpty() || num2.isEmpty() || operator.isEmpty()) return;
69
70         double n1 = double.parse(num1);
71         double n2 = double.parse(num2);
72         double result = 0;
73
74         switch (operator) {
75             case "+":
76                 result = n1 + n2;
77                 break;
78             case "-":
79                 result = n1 - n2;
80                 break;
81             case "*":
82                 result = n1 * n2;
83                 break;
84             case "/":
85                 result = n1 / n2;
86                 break;
87         }
88
89         setState(() {
90             display = result.toString();
91             num1 = result.toString();
92             num2 = "";
93             operator = "";
94         });
95     }
```

สร้างปุ่มในส่วนของการแสดงผลใน Scaffold

```
97   @override
98   Widget build(BuildContext context) {
99     return Scaffold(
100       appBar: AppBar(
101         centerTitle: true,
102         title: Text("Calculator"),
103       ), // AppBar
104       body: Center(
105         child: Column(
106           mainAxisAlignment: MainAxisAlignment.center,
107           children: [
108             Text(
109               display,
110               style: TextStyle(fontSize: 40, color: Colors.red),
111             ), // Text
112             SizedBox(height: 15),
113
114             // 7 8 9 +
115             Row(
116               mainAxisAlignment: MainAxisAlignment.center,
117               children: [
118                 calcBtn("7", () => pressNumber("7")),
119                 calcBtn("8", () => pressNumber("8")),
120                 calcBtn("9", () => pressNumber("9")),
121                 calcBtn("+", () => pressOperator("+")),
122               ],
123             ), // Row
124             SizedBox(height: 10),
125
126             // 4 5 6 -
127             Row(
128               mainAxisAlignment: MainAxisAlignment.center,
129               children: [
130                 calcBtn("4", () => pressNumber("4")),
131                 calcBtn("5", () => pressNumber("5")),
132                 calcBtn("6", () => pressNumber("6")),
133                 calcBtn("-", () => pressOperator("-")),
134               ],
135             ), // Row
136             SizedBox(height: 10),
137
```

```

138 // 1 2 3 *
139 Row(
140   mainAxisAlignment: MainAxisAlignment.center,
141   children: [
142     calcBtn("1", () => pressNumber("1")),
143     calcBtn("2", () => pressNumber("2")),
144     calcBtn("3", () => pressNumber("3")),
145     calcBtn("*", () => pressOperator("*")),
146   ],
147 ), // Row
148 SizedBox(height: 10),
149
150 // 0 C = /
151 Row(
152   mainAxisAlignment: MainAxisAlignment.center,
153   children: [
154     calcBtn("0", () => pressNumber("0")),
155     calcBtn("C", clear),
156     calcBtn("=", calculate),
157     calcBtn("/", () => pressOperator("/")),
158   ],
159 ), // Row
160 ],
161 ), // Column
162 ), // Center
163 ); // Scaffold
164 }
165 }
166

```

ทดสอบการทำงานของ Application

