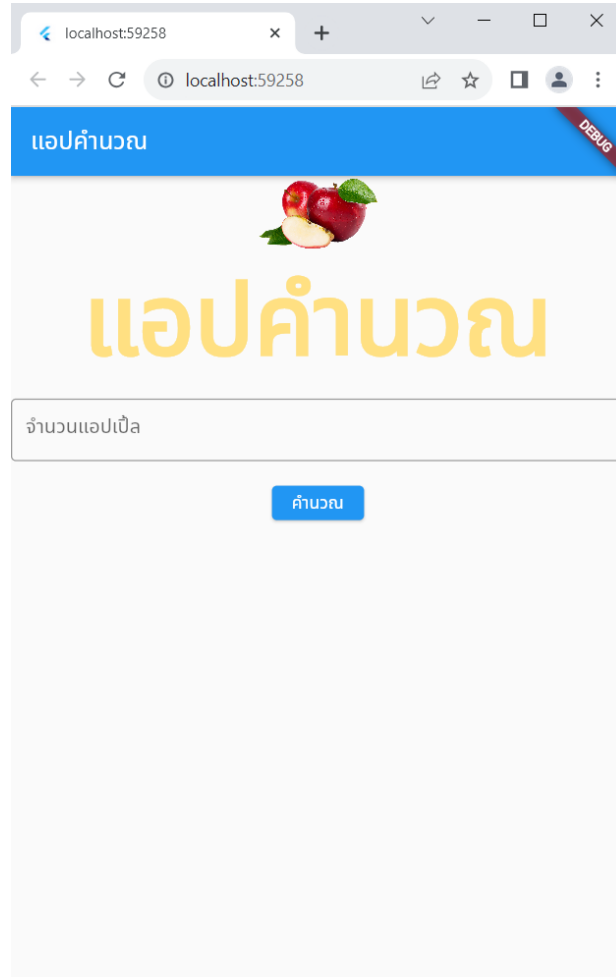


Chapter 4 การสร้างแอปพลิเคชันคำนวณ

part 1: กำหนด layout แบบคอลัมน์ การใส่รูปภาพ การสร้าง
TextField ใส่ปุ่ม และ SizedBox

ผลลัพธ์วันนี้



1. cd เข้า folder หน้า desktop โดยวิธีที่ถนัด

CA Command Prompt

```
Microsoft Windows [Version 10.0.19045.3693]  
(c) Microsoft Corporation. All rights reserved.
```

```
C:\Users\U21H1>
```

2. สร้าง project ใหม่ ตั้งชื่อว่า calculate

```
C:\Windows\System32\cmd.exe - flutter create calculate
Microsoft Windows [Version 10.0.19045.3693]
(c) Microsoft Corporation. All rights reserved.

C:\napaphatFlutter>flutter create calculate
Creating project calculate...
Resolving dependencies in calculate... (2.0s)
Got dependencies in calculate.
Wrote 129 files.

All done!
You can find general documentation for Flutter at: https://docs.flutter.dev/
Detailed API documentation is available at: https://api.flutter.dev/
If you prefer video documentation, consider: https://www.youtube.com/c/flutterdev

In order to run your application, type:

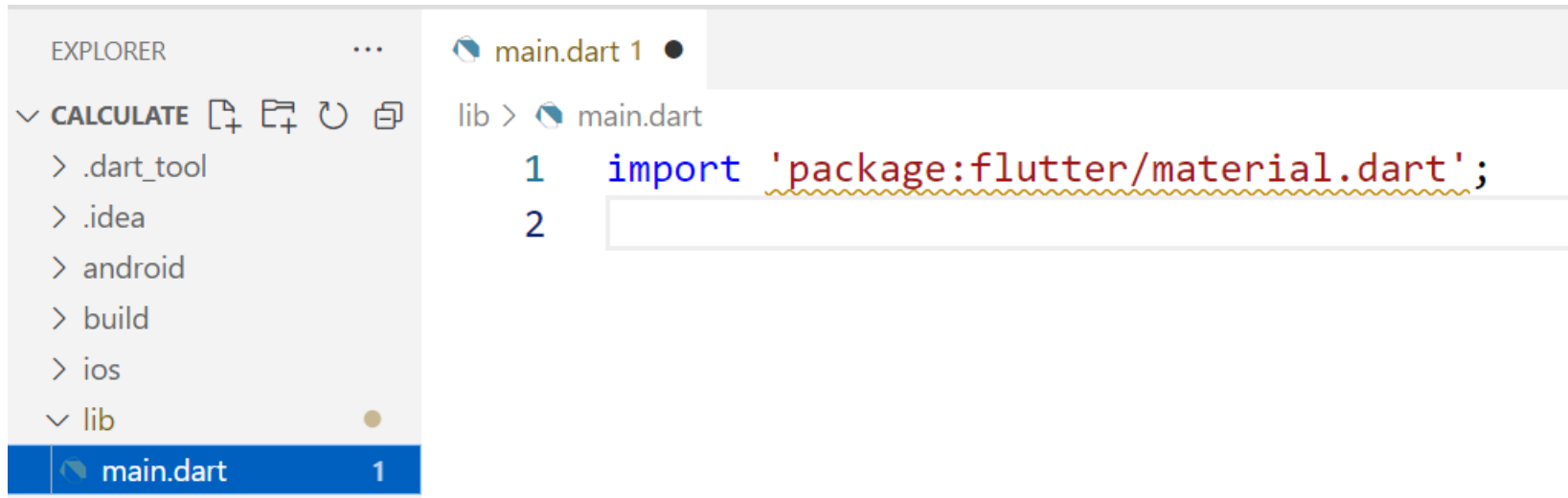
$ cd calculate
$ flutter run

Your application code is in calculate\lib\main.dart.

C:\napaphatFlutter>
```

3. เปิด project ใน vscode

ลบโค้ดทั้งหมดใน main.dart ให้เหลือเฉพาะส่วน import



```
EXPLORER  
CALCULATE  
> .dart_tool  
> .idea  
> android  
> build  
> ios  
lib  
main.dart 1  
main.dart 1  
lib > main.dart  
1 import 'package:flutter/material.dart';  
2
```

4. สร้าง void main

tip* พิมพ์ main แล้วเลือกที่ vscode เลือกให้ จะมาทั้งหมด

```
main.dart 1 ●
lib > main.dart > main
1 import 'package:flutter/material.dart';
   Run | Debug | Profile
2 void main(List<String> args) {
3
4 }
```

5. สร้าง StatelessWidget ด้วยการพิมพ์ stl แล้วเลือกที่ vscode เลือกให้ จะได้ผลโค้ดดังรูป

```
main.dart •  
lib > main.dart > MyWidget  
1 import 'package:flutter/material.dart';  
  Run | Debug | Profile  
2 void main(List<String> args) {  
3  
4 }  
5  
6 class MyWidget extends StatelessWidget {  
7   const MyWidget({super.key});  
8  
9   @override  
10  Widget build(BuildContext context) {  
11    return const Placeholder();  
12  }  
13 }
```

6. ตั้งชื่อว่า MyApp

หมายเหตุ* ตรง const จะขึ้นชื่อ MyApp ให้อัตโนมัติ

main.dart

lib > main.dart > MyApp

```
1 import 'package:flutter/material.dart';
  Run | Debug | Profile
2 void main(List<String> args) {
3
4 }
5
6 class MyApp extends StatelessWidget {
7   const MyApp({super.key});
8
9   @override
10  Widget build(BuildContext context) {
11    | return const Placeholder();
12  }
13 }
```


7. ลบตรง const Placeholder พิมพ์ MaterialApp แทน

```
main.dart •
lib > main.dart > MyApp > build
1  import 'package:flutter/material.dart';
   Run | Debug | Profile
2  void main(List<String> args) {
3
4  }
5
6  class MyApp extends StatelessWidget {
7    const MyApp({super.key});
8
9    @override
10   Widget build(BuildContext context) {
11     return MaterialApp();
12   }
13 }
```

8. เพิ่ม home: Scaffold () ด้านใน MaterialApp

```
6 class MyApp extends StatelessWidget {  
7   const MyApp({super.key});  
8  
9   @override  
10  Widget build(BuildContext context) {  
11    return MaterialApp(home: Scaffold(),);  
12  }  
13 }
```

9. ใน Scaffold >>ใส่ appBar และ ใส่ title และ text

```
6 class MyApp extends StatelessWidget {
7   const MyApp({super.key});
8
9   @override
10  Widget build(BuildContext context) {
11    return MaterialApp(home: Scaffold(
12      appBar: AppBar(title: Text("แอปคำนวณ")),
13    )); // Scaffold // MaterialApp
14  }
15 }
```

10. ใส่ body

```
6 class MyApp extends StatelessWidget {
7   const MyApp({super.key});
8
9   @override
10  Widget build(BuildContext context) {
11    return MaterialApp(home: Scaffold(
12      appBar: AppBar(title: Text("แอปคำนวณ"),),
13      body: |,
14    ),); // Scaffold // MaterialApp
15  }
16 }
```

11. สร้าง StatefulWidget ด้วยการพิมพ์ stf แล้วเลือกที่ vscode เลือกให้ จะให้โค้ดมาทั้งหมด >> แล้วตั้งชื่อ Home

```
18 class Home extends StatefulWidget {  
19   const Home({super.key});  
20  
21   @override  
22   State<Home> createState() => _HomeState();  
23 }  
24  
25 class _HomeState extends State<Home> {  
26   @override  
27   Widget build(BuildContext context) {  
28     return const Placeholder();  
29   }  
30 }
```

12. ที่ body ของ class แรก ให้ใส่ Home()

หมายเหตุ* คือการเรียกใช้ class home ด้านล่างนั่นเอง

```
6  ∨ class MyApp extends StatelessWidget {
7    const MyApp({super.key});
8
9    @override
10  ∨ Widget build(BuildContext context) {
11    |   return MaterialApp(home: Scaffold(
12  ∨    |     appBar: AppBar(title: Text("แอปคำนวณ")),),
13  ∨    |     body: Home()),
14    |   ),); // Scaffold // MaterialApp
15  |   }
16  }
17
```

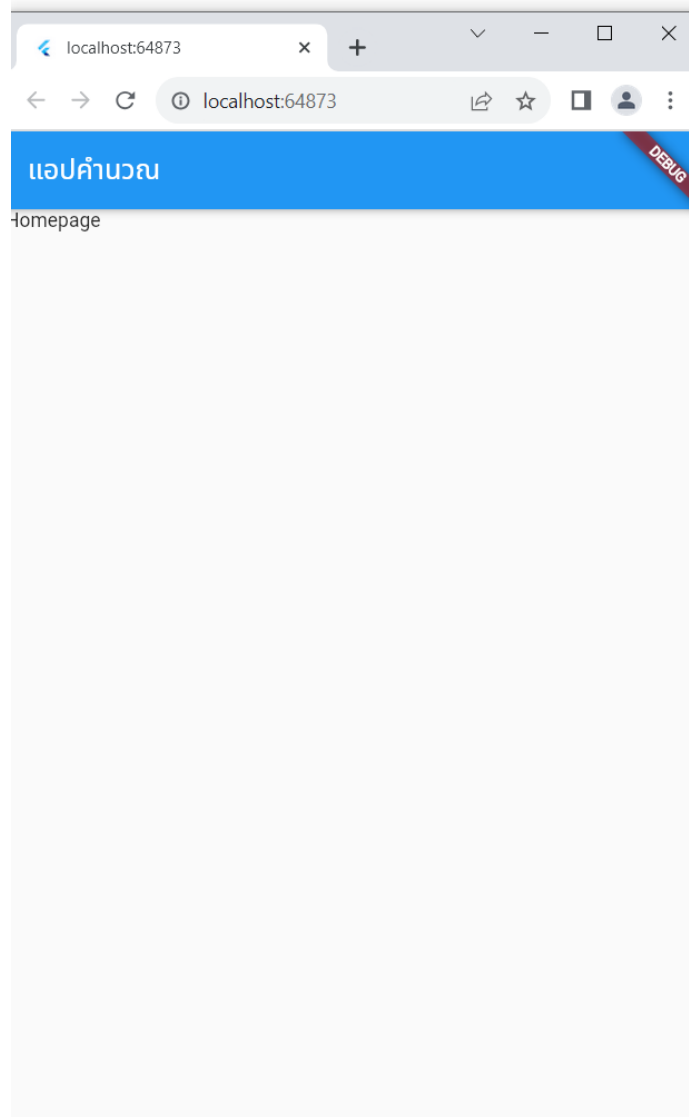
13. void main คำนบน ใส่ runApp (MyApp());
หมายเหตุ* คือ การสั่ง run class MyApp()

```
main.dart •  
lib > main.dart > main  
1 import 'package:flutter/material.dart';  
   Run | Debug | Profile  
2 void main(List<String> args) {  
3   | runApp(MyApp());|  
4 }  
5
```

14. แก้ return ที่ class สุดท้าย ให้ Text แทน const Placeholder() >> และใส่ข้อความว่า Homepage

```
25  ∨ class _HomeState extends State<Home> {  
• 26  |   @override  
27  ∨  |   Widget build(BuildContext context) {  
28  |   |   return Text("Homepage");  
29  |   |   }  
30  |   }  
    }
```


และทดลอง run แอปจากการแก้ไขทั้งหมด ได้ผลดังรูป



Recap : Child VS Children

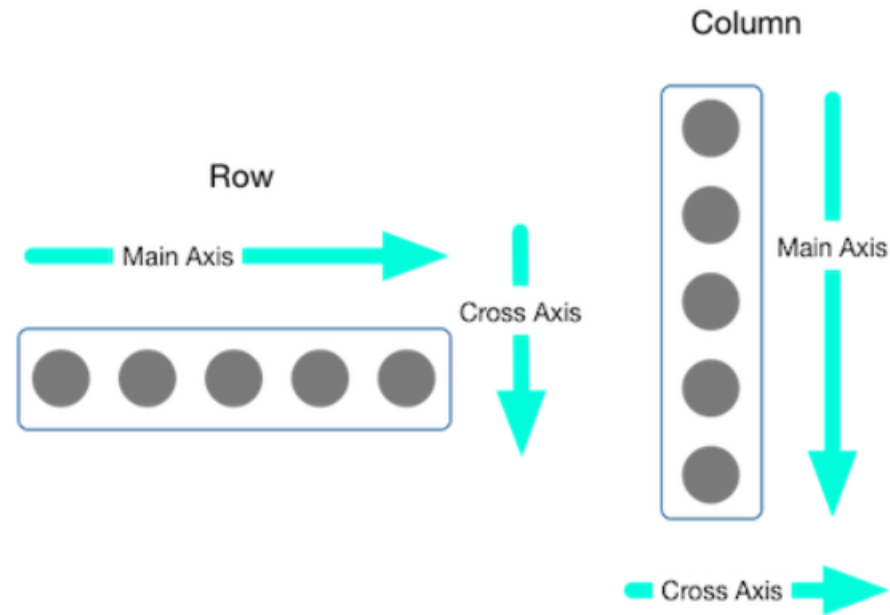
`child` takes a single widget

```
child: Text('foo')
```

`children` takes a list of widgets

```
children: <Widget>[Text('foo'), Text('bar')]
```

column



ศึกษาเพิ่มเติมได้ที่ <https://docs.flutter.dev/development/ui/layout>

column (ต่อ)

```
30 class _HomeState extends State<Home> {
31   @override
32   Widget build(BuildContext context) {
33     return Center(
34       child: Column(
35         children: [
36           Image.asset("assets/apple.jpg", width: 100),
37           Text("โปรแกรมคำนวณ", style: TextStyle(fontSize: 50, fontWeight: FontWeight.bold, color: Colors.blue)),
38           TextField(
39             decoration: InputDecoration(
40               labelText: "จำนวนแอปเปิ้ล", border: OutlineInputBorder(), ), // InputDecoration // TextField
41           SizedBox(height: 20, ),
42           ElevatedButton(onPressed: () {}, child: Text("คำนวณ"))
43         ],
44       ) // Column
45     ); // Center
46   }
47 }
48
```

ศึกษาเพิ่มเติมได้ที่ <https://docs.flutter.dev/development/ui/layout>

15. ที่ class สุดท้าย ตรง return แก้เป็น Center(child: Column (children:

- 25 `class _HomeState extends State<Home> {`
26 `@override`
27 `Widget build(BuildContext context) {`
28 `return Center(child: Column(children: [Text("picture"), Text("แอปคำนวณ")],),);`
29 `}`
30 `}`

16. ใส่ text field

```
25 class _HomeState extends State<Home> {
26   @override
27   Widget build(BuildContext context) {
28     return Center(child: Column(children: [Text("picture"), Text("แอปคำนวณ"),
29     TextField(decoration: InputDecoration(labelText: "จำนวนแอปเปิ้ล",
30     border: OutlineInputBorder()))],),),); // InputDecoration // TextField // Column //
31   }
32 }
```


17. กด alt + shift + f ในการจัดโค้ด

```
30 class _HomeState extends State<Home> {
31   @override
32   Widget build(BuildContext context) {
33     return Center(
34       child: Column(
35         children: [
36           Text("picture"),
37           Text("แอปคำนวณ"),
38           TextField(
39             decoration: InputDecoration(
40               labelText: "จำนวนแอปเปิ้ล", border: OutlineInputBorder()) // Input
41         ],
42       ), // Column
43     ); // Center
44   }
45 }
```

18. เริ่มใส่รูป โดยเริ่มค้นรูป png ที่ไม่มีลิขสิทธิ์ ที่เว็บ <https://www.cleanpng.com/>

Apps Facebook YouTube พัฒนาตัวเอง งานสาขา ขอทำห้องปฏิบัติการ สอน 2/2564 Reading list

Watercolor Christmas Coronavirus Frame Arrow Flower Snow Heart



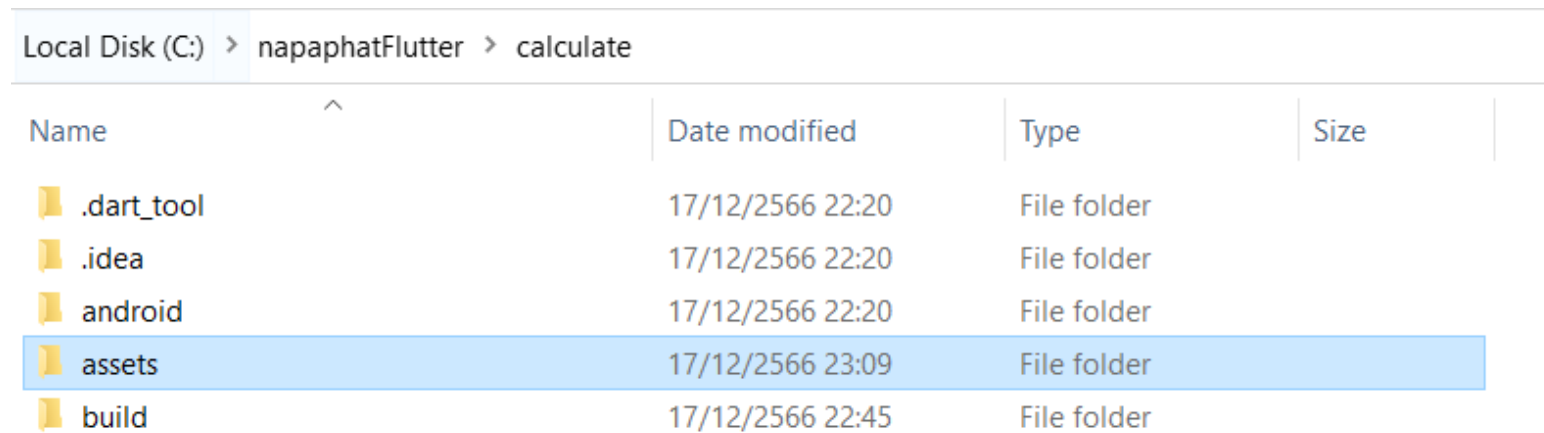
Upload & discover png files without a hitch!

apple

christmas snow flower frame background christmas tree merry christmas 2020 santa new year light tree

19. สร้าง folder assets ใน folder project

หมายเหตุ* จะสร้างใน vscode เหมือนการทำ font ครั้งที่แล้วก็ได้



Name	Date modified	Type	Size
^			
.dart_tool	17/12/2566 22:20	File folder	
.idea	17/12/2566 22:20	File folder	
android	17/12/2566 22:20	File folder	
assets	17/12/2566 23:09	File folder	
build	17/12/2566 22:45	File folder	

20. นำรูปจากการค้นหา แก้ไขชื่อรูป และนำรูปไปใส่ folder assets ที่สร้างขึ้น หมายเหตุ* จะนำไปใส่ เหมือนการทำ font ครั้งที่แล้วก็ได้

This PC > Local Disk (C:) > napaphatFlutter > calculate > assets



apple.png

21. set ค่าในไฟล์ pubspec.yaml แล้วบันทึกไฟล์

```
61 # To add assets to your application, add an assets section, like this:  
62 assets:  
63   - assets/apple.png  
64   # - images/a_dot_ham.jpeg  
65
```

22. แก้โค้ดที่ไฟล์ main.dart ใส่ image

```
29
30 class _HomeState extends State<Home> {
31   @override
32   Widget build(BuildContext context) {
33     return Center(
34       child: Column(
35         children: [
36           Image.asset("assets/apple.png", width: 100,),
37           Text("แอปคำนวณ"),
38           TextField(
39             decoration: InputDecoration(
40               labelText: "จำนวนแอปเปิ้ล", border: OutlineInputBorder()),
41         ],
42       ), // Column
43     ); // Center
44   }
45 }
46
```

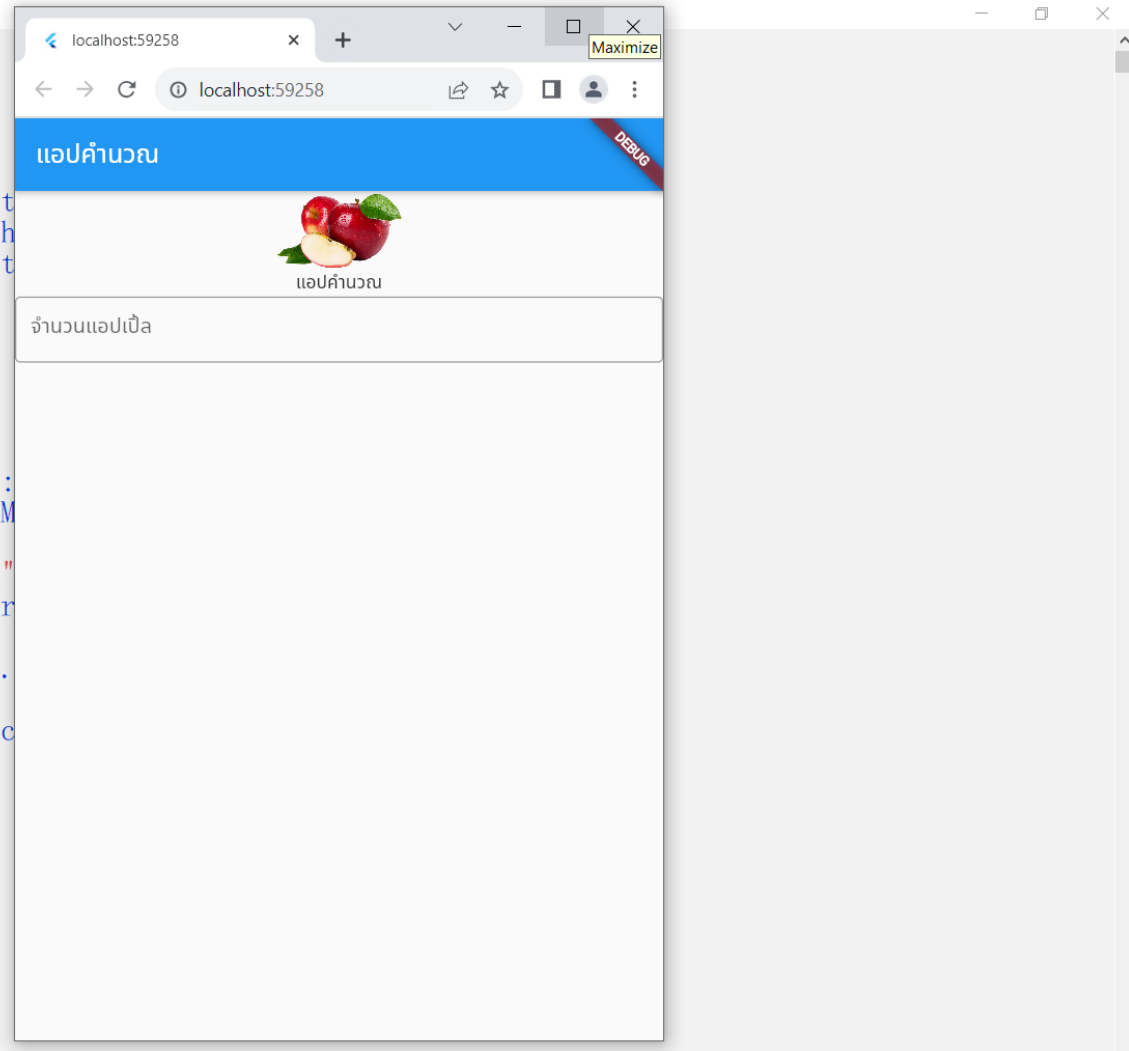
และ run จะได้ผลลัพธ์ดังภาพ

```
C:\Windows\System32\cmd.exe - flutter run
Microsoft Windows [Version 10.0.19045.3693]
(c) Microsoft Corporation. All rights reserved.

C:\napaphatFlutter\calculate>flutter run
Connected devices:
Windows (desktop) • windows • windows-x64 • Microsoft
Chrome (web) • chrome • web-javascript • Google Ch
Edge (web) • edge • web-javascript • Microsoft
[1]: Windows (windows)
[2]: Chrome (chrome)
[3]: Edge (edge)
Please choose one (or "q" to quit): 2
Launching lib\main.dart on Chrome in debug mode...
Waiting for connection from debug service on Chrome...
This app is linked to the debug service: ws://127.0.0.1:
Debug service listening on ws://127.0.0.1:59314/_9U6HYcM

To hot restart changes while running, press "r" or "R"
For a more detailed help message, press "h". To quit, pr

A Dart VM Service on Chrome is available at: http://127.
The Flutter DevTools debugger and profiler on Chrome is
http://127.0.0.1:9102?uri=http://127.0.0.1:59314/_9U6HYc
```



23. เปิด new window ดูโค้ดเก่าเพื่อดูโค้ดตกแต่ง font

```
Go Run ... < > secondapp
main.dart 1 x
lib > main.dart > _MyHomePageState > build
71 @override
72 Widget build(BuildContext context) {
73   // This method is rerun every time setState is called, for instance as done
74   // by the _incrementCounter method above.
75   //
76   // The Flutter framework has been optimized to make rerunning build methods
77   // fast, so that you can just rebuild anything that needs updating rather
78   // than having to individually change instances of widgets.
79   return Scaffold(
80
81     body: Center(child: Text("test", style: TextStyle(fontSize: 80,
82     fontWeight: FontWeight.bold, color: Colors.amber[200], fontFamily: "donburi"),) ,),
83     floatingActionButton: FloatingActionButton(
84       onPressed: _incrementCounter,
85       tooltip: 'Increment',
+ 86       child: const Icon(Icons.add),
87     ), floatingActionButtonLocation: FloatingActionButtonLocation.startFloat, // This trail:
88   ); // Scaffold
89 }
90 }
```

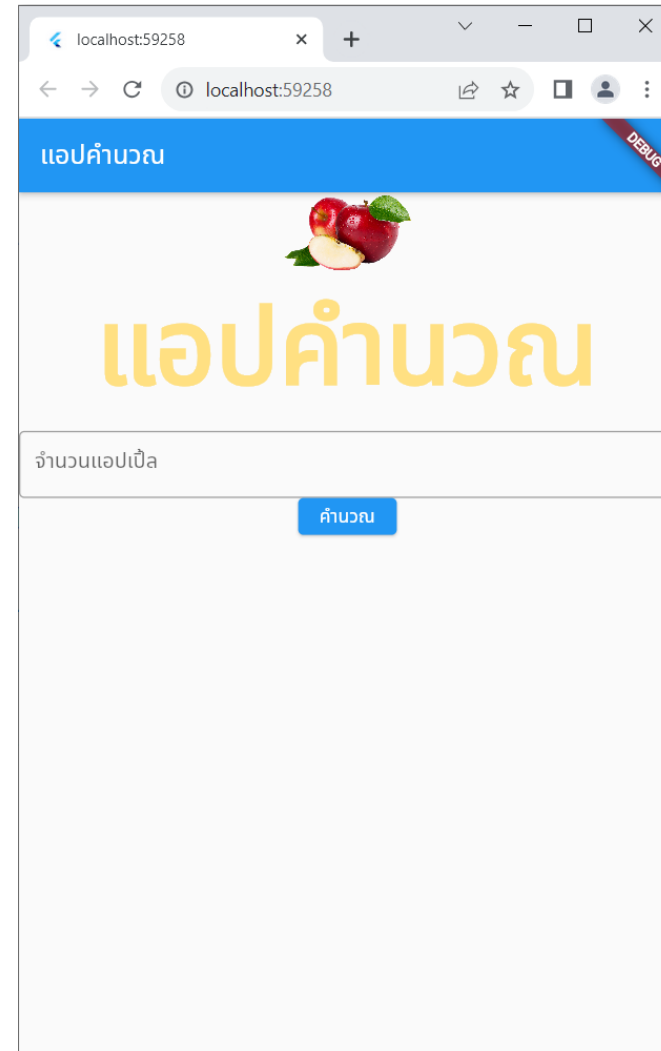
24. นำโค้ด มาปรับตัวอักษร

```
30 class _HomeState extends State<Home> {
31   @override
32   Widget build(BuildContext context) {
33     return Center(
34       child: Column(
35         children: [
36           Image.asset("assets/apple.png", width: 100,),
37           Text("แอปคำนวณ" , style: TextStyle(fontSize: 80,
38             fontWeight: FontWeight.bold, color: Colors.amber[200]), // TextStyle
39           TextField(
40             decoration: InputDecoration(
41               labelText: "จำนวนแอปเปิ้ล", border: OutlineInputBorder()) // :
42         ],
43       ), // Column
44     ); // Center
45   }
46 }
47
```

25. ใส่ปุ่มบันทึก

```
30 class _HomeState extends State<Home> {
31   @override
32   Widget build(BuildContext context) {
33     return Center(
34       child: Column(
35         children: [
36           Image.asset(
37             "assets/apple.png",
38             width: 100,
39           ), // Image.asset
40           Text(
41             "แอปคำนวณ",
42             style: TextStyle(
43               fontSize: 80,
44               fontWeight: FontWeight.bold,
45               color: Colors.amber[200]), // TextStyle
46           ), // Text
47           TextField(
48             decoration: InputDecoration(
49               labelText: "จำนวนแอปเปิ้ล", border: OutlineInputBorder()),
50           ), // TextField
51           ElevatedButton(onPressed: () {}, child: Text("คำนวณ")),
52         ],
53       ), // Column
54     ); // Center
55   }
56 }
```


แล้วทดลอง run ได้ผลดังภาพ



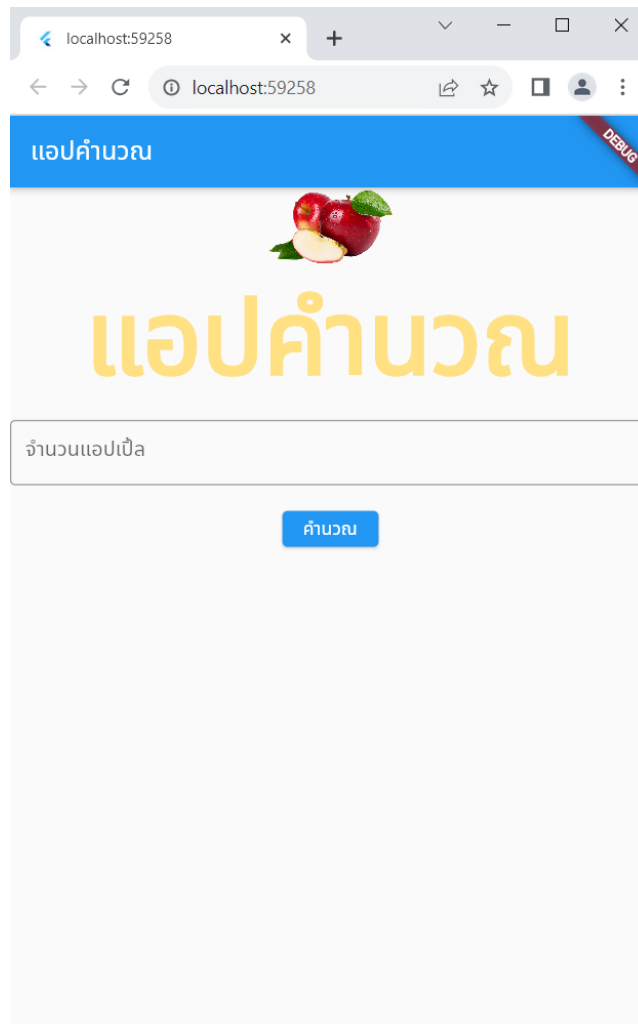
26. ใส่ `SizeBox` เพื่อเว้นระยะ

หมายเหตุ* หากกำหนด `height` มากเกินไป อาจ `error` ได้ เนื่องจากสเกลใหญ่เกินไป

```
47   TextField(  
48     decoration: InputDecoration(  
49       labelText: "จำนวนแอปเปิ้ล", border: OutlineInputBorder()),  
50     SizeBox(height: 20,),  
51     ElevatedButton(onPressed: () {}, child: Text("คำนวณ"))  
52   ],  
53   ), // Column  
54 ); // Center  
55 } }  
56
```

อ้างอิงได้จาก <https://api.flutter.dev/flutter/widgets/SizeBox-class.html>

และ run จะได้ผลลัพธ์ดังภาพ

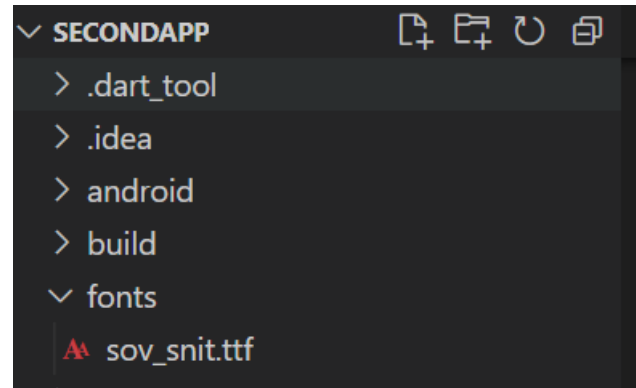


27. เพิ่มเติม* สามารถกำหนดรูปแบบ font ให้กับ text ได้ (ทบทวน เนื้อหาจากครั้งที่แล้ว)

download font



สร้าง folder



File ชื่อ pubspec.yaml

```
fonts:  
- family: sovsnit  
  fonts:  
    - asset: fonts/sov_snit.ttf
```

ใส่ fontFamily ที่สร้างใน main.dart

```
body: Center(child: Text("สวัสดี",  
style: TextStyle(fontSize: 100, fontWeight: FontWeight.bold,  
color: Colors.blue, fontFamily: 'sovsnit'),),), // TextStyle // Text // Center
```

Reference

ลุงวิศวกร สอนคำนวณ. Flutter 2 Mobile App Bootcamp 2021 - Uncle Engineer

คอลัมภ์ > <https://docs.flutter.dev/development/ui/layout>

SizedBox > <https://api.flutter.dev/flutter/widgets/SizedBox-class.html>