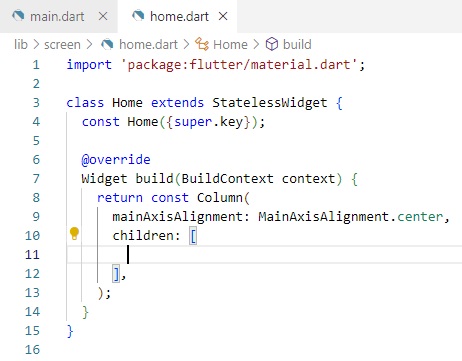
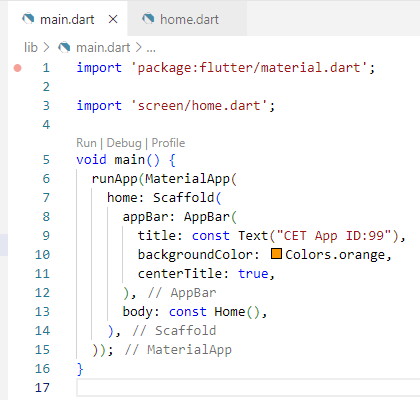
**ใบงานที่ 4**

**ชื่อ - นามสกุล...........นายศรสิวะพงษ์..โพธิวงศ์....รหัสนักศึกษา........................2661031741113.............................**

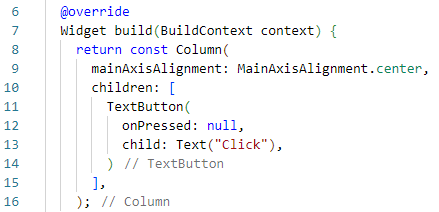
**Button Widget**

คือปุ่ม(Button)ต่างๆ ที่เราเคยเห็นจากการใช้งานแอพพลิเคชั่นอื่นๆ โดยแบ่งเป็น

1. Text Button เป็นการสร้างปุ่มแบบข้อความ
2. Filled Button เป็นการสร้างปุ่มที่มีการใส่พื้นผิวลงไป
3. Elevated Button เป็นการสร้างปุ่มที่มีลักษณะนูน
4. Outlined Button เป็นการสร้างปุ่มและมีเส้นขอบ
5. สร้างโปรเจคใหม่ขึ้นมาชื่อ mobilelab2cetรหัสนักศึกษา 3 หลักหลัง จากนั้นทำการสร้าง folder screen และสร้างไฟล์ใหม่ชื่อ home.dart ใน folder screen และจัดการเพิ่มคลาสให้กับไฟล์ main.dart รวมถึงสร้าง AppBar ให้เหมือน mobile1 หรือจะคัดลอกมาก็ได้



1. ที่ไฟล์ home.dart เพิ่ม Textbutton Widget เข้าไปโดย onPressed คือการกำหนดว่าเมื่อมีการคลิ๊กที่ปุ่มแล้วจะให้ทำอะไร ในที่นี้ใส่ null คือยังไม่ต้องการให้ทำอะไร ส่วน child จะใส่ข้อความ Click เข้าไปโดยใช้ Text widget เมื่อแก้ไขโค๊ดเสร็จอย่าลืมบันทึก



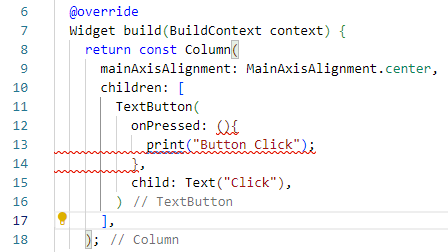
บันทึกผลการรัน.......................รูป......................................................................

A screen shot of a cell phone

AI-generated content may be incorrect.

ทดลองคลิ๊กที่ข้อความ Click สังเกตุผลที่เกิดที่ช่อง Debug Console ว่ามีการเปลี่ยนแปลงเกิดขึ้นหรือไม่ .........ไม่มี.......(มี/ไม่มี)

เพิ่มคำสั่งเพื่อให้เกิดเหตุการณ์เมื่อมีการกดปุ่ม Click โดยจะเพิ่มในส่วนของ onPressed ดังนี้



จะเกิด Error เส้นสีแดงให้ทำการลบ const ที่อยู่หน้าคำสั่ง Column ในบรรทัดที่ 8 ออก และให้ใส่ const หน้า Text widget แทนจากนั้นสั่งรันโปรแกรม และทดลองคลิ๊กที่ปุ่ม Click จะพบว่ามีผลการเปลี่ยนแปลงเกิดขึ้นที่ช่อง Debug Console

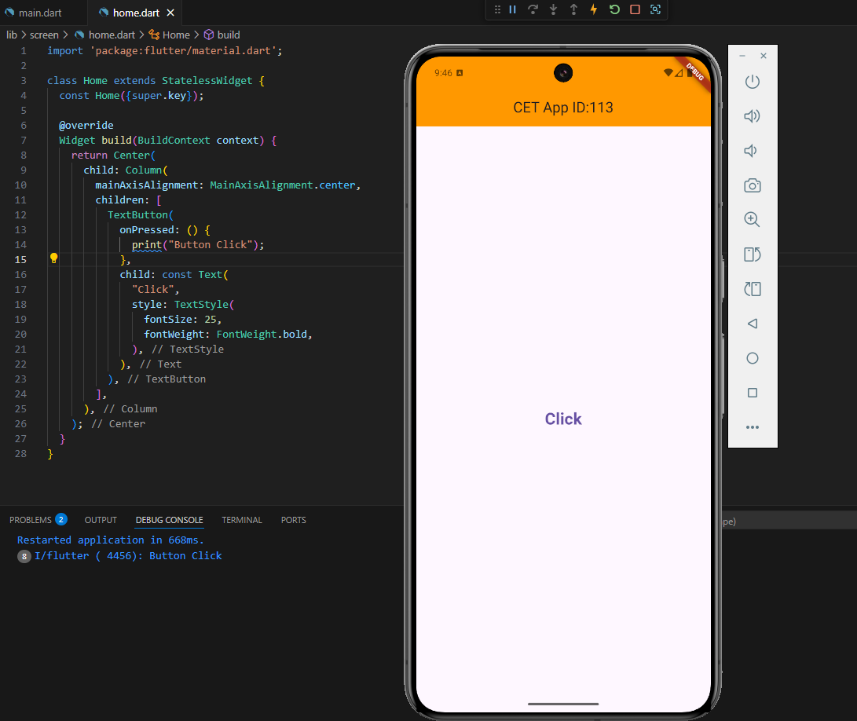
**บันทึกผล Debug Console ..........................................รูป......................................................................................................**

**A screen shot of a computer

AI-generated content may be incorrect.**

หากต้องการย้ายปุ่มไว้ที่กลางจอ ให้ทำการคลิ๊กที่ Column และคลิ๊กที่หลอดไฟเลือก Warp with center จากนั้นให้กำหนดค่าเกี่ยวกับข้อความใน Text widget โดยลองกำหนดขนาดเป็น 25 และรูปแบบของข้อความเป็นตัวหนา

**ระบุคำสั่งที่ใช้....................................................................รูป..............................................................................................**

****

1. เพิ่ม Widget Button แบบอื่นๆ เข้าไปต่อจาก Button ก่อนหน้าได้แก่

**Filled Button**

const SizedBox(height: 10,),

FilledButton(onPressed: (){}, child: const Text("Click",

style: TextStyle(fontSize: 25, fontWeight: FontWeight.bold),

),),

**Outline Button**

const SizedBox(height: 10,),

OutlinedButton(onPressed: (){}, child: const Text("Click",

style: TextStyle(fontSize: 25, fontWeight: FontWeight.bold),

),),

**Elevated Button**

const SizedBox(height: 10,),

ElevatedButton(onPressed: (){}, child: const Text("Click",

style: TextStyle(fontSize: 25, fontWeight: FontWeight.bold),

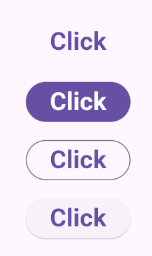
),),

**คำถาม** จากรูปต่อไปนี้ให้เขียนชื่อของ Widget Button ให้ถูกต้อง

................**TextButton**..................

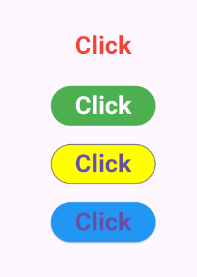
..................**FilledButton**.........................

.................**OutlinedButton**...........................

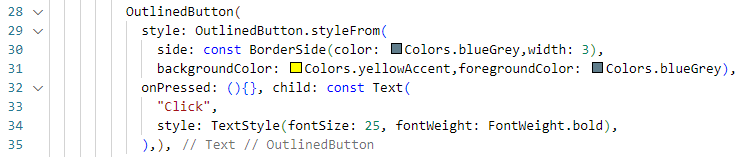


....................**ElevatedButton**..........................

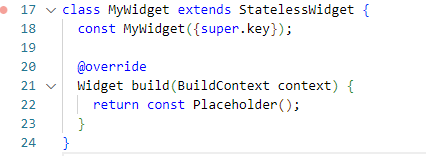
การกำหนดสีพื้นหรือ Style ให้กับ Button Widget โดยใช้คำสั่งดังตัวอย่าง



หรือสามารถกำหนดเพิ่มเติมเช่นในส่วนของเส้นขอบของ Outline Button เช่น กำหนดสีของเส้นและความหนาของขอบของปุ่มให้เป็นไปตามที่เราต้องการ โดยกำหนดในส่วนของ side :



1. ที่ไฟล์ main.dart เพื่อให้ง่ายต่อการจัดการให้ทำการย้าย MaterialApp ไปเป็น Widget ประเภท Stateless Widget โดยไปคลิ๊กที่บรรทัดล้างสุดของโค๊ดในไฟล์ main.dart และพิมพ์ st เลือก StatelessWidget โปรแกรมจะสร้างส่วนที่จำเป็นให้เรา



เปลี่ยนชื่อ Widget จาก MyWidget เป็น MyApp จากนั้นย้ายส่วนของ MaterialApp ไปใส่แทนในส่วนของ build และในส่วน runApp() ให้ระบุเรียก Widget ที่เราเพิ่งสร้างไปคือ MyApp() แทนทั้งนี้จะเห็นว่าต้องใส่ const ด้วย

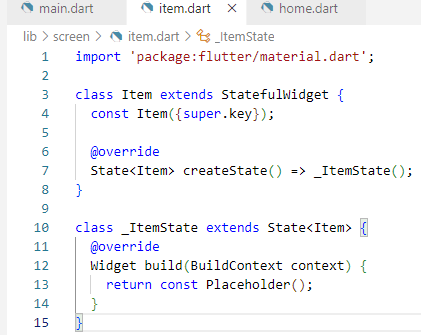
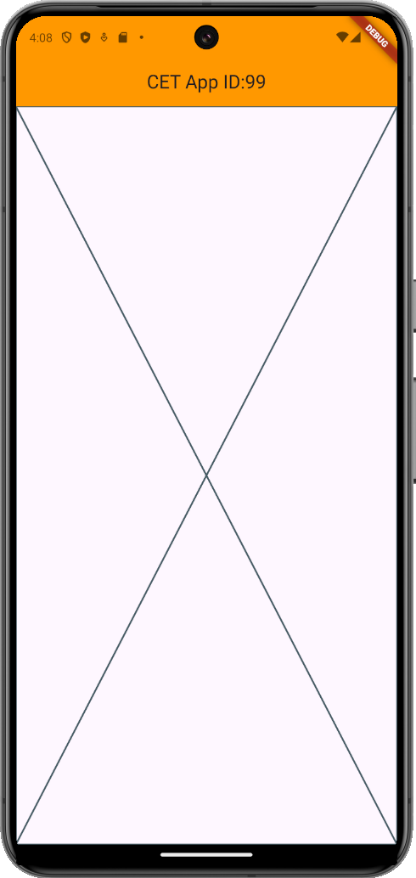


1. Stateful Widget และ Stateless Widget
   1. Stateful Widget สามารถเปลี่ยนแปลงค่าได้ และทำงานได้หลาย State เช่น Checkbox, Slider, Text Field
   2. Stateless Widget ไม่สามารถเปลี่ยนแปลงค่าได้เหมาะกับการสร้าง Widget แบบคงที่ เช่น Text , Icon
   3. ทั้งสองแบบสามารถเปลี่ยนจาก Stateful เป็น Stateless หรือ Stateless เป็น Stateful

State หมายถึง ตัวแปรหรือข้อมูลที่ใช้ควบคุมการทำงานของแอพ ให้สามารถโต้ตอบกับผู้ใช้งานได้ โดยสามารถใช้คำสั่ง setState() เพื่อแจ้งว่าในแอพมีบางอย่างเปลี่ยนแปลงเกิดขึ้น โดยที่เราไม่ต้อง build app ใหม่

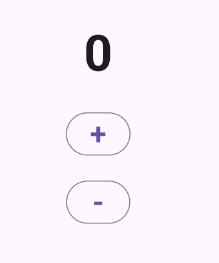
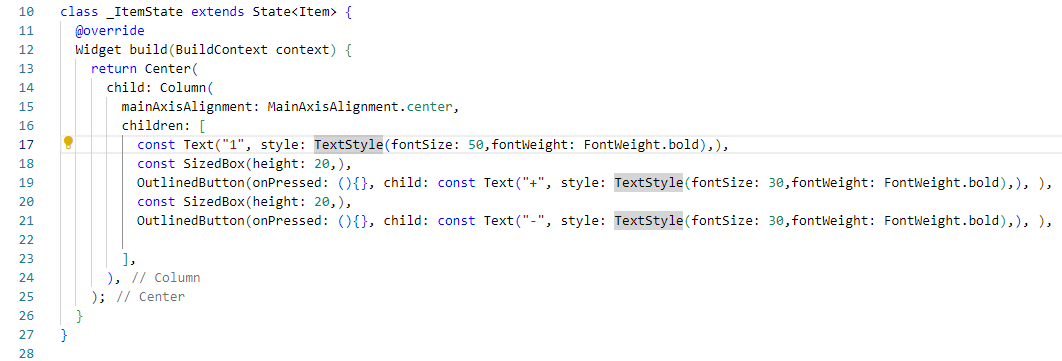
ให้นักศึกษาสร้างไฟล์ขึ้นมาใหม่ชื่อ item.dart ภายใน Folder screen สำหรับแสดงข้อมูลและค่าที่เปลี่ยนแปลงค่าได้ โดยใช้ Stateful และกำหนดชื่อ Widget เป็น Item ก็จะได้ StatefulWidget ที่ชื่อ Item ขึ้นมา

จากนั้นให้แก้ไขที่ไฟล์ main.dart ในส่วน body ให้โหลดหน้า item.dart มาแทน home.dart ในที่นี้ให้คอมเม้นหน้า home.dart ไปและคอมเม้น import ด้วย ตามนี้



ผลจาก Placeholder()

1. จากข้อ 5 ให้จะออกแบบแอพดังรูปด้านล่าง โดยใช้ Text widget และให้อยู่กลางจอโดยแทนที่โค้ดดังนี้



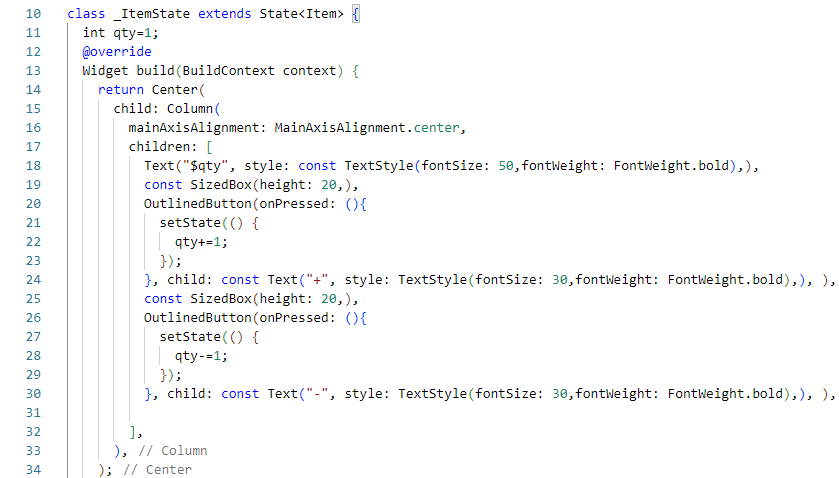
**บันทึกผลการรัน...................................รูป....................................................................**

**A screen shot of a cell phone

AI-generated content may be incorrect.**

**ตอบคำถาม** เมื่อกดปุ่ม + และ - แล้วเกิดการเปลี่ยนแปลงหรือไม่ .....เกิด......(เกิด / ไม่เกิด)

1. จากข้อ 6 Widget ที่แสดงข้อความ + และ – ความหมายคือ ออกแบบให้สามารถเพิ่มค่า และลดค่าของ Text Widget ที่แสดงเลข 0 อยู่ในตอนนี้ซึ่งยังจะไม่เกิดผลใดๆ เมื่อรันโปรแกรมและกดปุ่ม + หรือ - ดังนั้นจะต้องเพิ่มคำสั่งเพื่อให้เพิ่มและลดค่าได้ แต่การจะทำได้นั้นค่าจะตองถูกเก็บไว้ในตัวแปร เพื่อให้สามารถเปลี่ยนแปลงค่าได้ ดังนั้นจะต้องทำการสร้างตัวแปรขึ้นมาก่อน โดยตัวแปรที่จะสร้างนี้จะเป็น State ที่สามารถเปลี่ยนแปลงค่าได้(Stateful Widget)



สร้างตัวแปรชนิด Integer และกำหนดค่าเริ่มต้นเท่ากับ 1

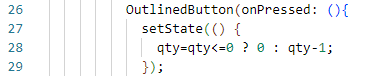
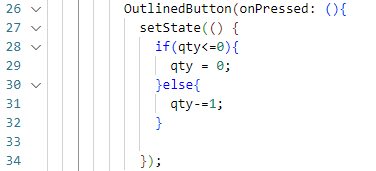
ดึงค่าจากตัวแปล qty มาใช้และต้องนำ const ออก

ใช้คำสั่ง setState เพื่อสั่งให้ลดค่าในตัวแปล ที่ละ 1 ทุกครั้งที่มีการกดปุ่ม

**บันทึกผลการรัน...................................รูป....................................................................**

**ตอบคำถาม** เมื่อกดปุ่ม + จำนวณ 5 ครั้ง และกดปุ่ม - ให้มีค่าน้อยกว่า 0 เกิดค่าติดลบหรือไม่ .....**เกิด**.......(เกิด / ไม่เกิด)

1. หากวัตถุประสงค์ของการออกแบบแอพของเราคือห้ามมีค่าน้อยกว่า 0 เราจะต้องออกแบบโค้ดให้มีการตรวจสอบเงื่อนไข โดยให้มีการตรวจสอบทุกครั้งเมื่อกดปุ่มว่าค่าในตัวแปรเท่ากับ 0 หรือไม่ หากเท่ากับ 0 ให้ตัวแปรนั้นมีค่าเท่ากับ 0 ไม่ลดค่าลงไปน้อยกว่านั้น แต่หากไม่ใช่ 0 ให้ลดค่าทีละ 1



แบบที่ 1

แบบที่ 2

ให้นักศึกษาทดลองเพิ่มโค้ดไปในโปรแกรมทั้งสองแบบ โดยทำทีละแบบและทดลองรันโดยการกด Stop ก่อนและกด F5 ใหม่เพื่อให้ได้ผลการรันที่แท้จริง

**บันทึกผลการรัน...................................รูปแบบที่1....................................................................**

A screen shot of a phone

AI-generated content may be incorrect.

**บันทึกผลการรัน...................................รูปแบบที่2....................................................................**

**A screen shot of a phone

AI-generated content may be incorrect.**

**ตอบคำถาม** เมื่อกดปุ่ม + จำนวณ 5 ครั้ง และกดปุ่ม - ให้มีค่าน้อยกว่า 0 เกิดค่าติดลบหรือไม่ .....ไม่เกิด......(เกิด / ไม่เกิด)

**ให้อธิบายโค้ดทั้งแบบที่ 1 และแบบที่ 2 ว่ามีความหมายอย่างไร................................................**

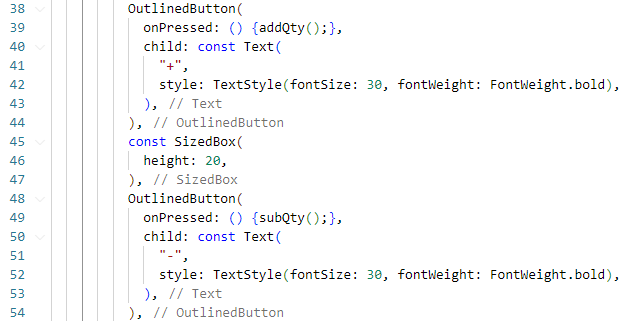
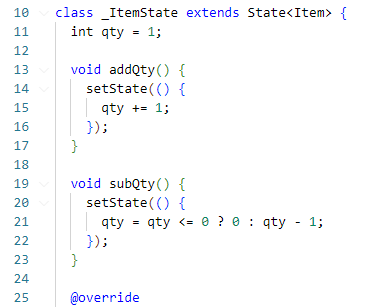
**แบบที่ 1: การใช้ if-else**

* **if (qty <= 0): "ถ้าหากว่า ค่าของ qty ในปัจจุบันน้อยกว่าหรือเท่ากับ 0"**
* **{ qty = 0; }: "ให้กำหนดค่า qty เป็น 0" (เป็นการบังคับไม่ให้ค่าลดลงไปอีก)**
* **else: "แต่ถ้าเงื่อนไขข้างบนไม่เป็นจริง (คือ qty มากกว่า 0)"**
* **{ qty -= 1; }: "ให้ลดค่า qty ลง 1"**

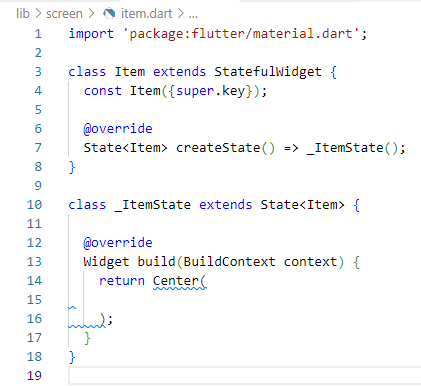
**แบบที่ 2: การใช้ Ternary Operator (? :)**

* **qty <= 0: คือเงื่อนไขที่ใช้ตรวจสอบ (เหมือนกับแบบที่ 1)**
* **?: ถ้าเงื่อนไขเป็น จริง ให้ใช้ค่าที่อยู่หลังเครื่องหมายนี้**
* **0: คือค่าที่จะถูกกำหนดให้ qty ถ้าเงื่อนไขเป็นจริง**
* **:: ถ้าเงื่อนไขเป็น เท็จ ให้ใช้ค่าที่อยู่หลังเครื่องหมายนี้**
* **qty - 1: คือค่าที่จะถูกกำหนดให้ qty ถ้าเงื่อนไขเป็นเท็จ**

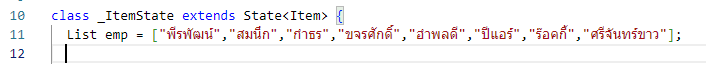
1. เราสามารถปรับปรุงโค้ดในข้อ 8 ให้อยู่ในรูปแบบ State Method ที่แยกออกมาจากตัวโปรแกรม และสามารถนำไปเรียกใช้ที่จุดไหนของโปรแกรมก็ได้ โดยไม่ต้องเขียนโค้ดชุดเดิมซ้ำ เพียงแค่เรียกผ่านชื่อของ State Method เท่านั้น โดยขั้นต่อไปนี้เราจะนำโค้ดส่วนของการเพิ่ม ลด ค่าไปไว้นอก Method Build โดยไปสร้าง Method ไว้ที่ใต้คำสั่ง Class และใช้งานร่วมกับคำสั่ง void เพื่อสร้าง Method ที่ชื่อว่า addQty และ subQty



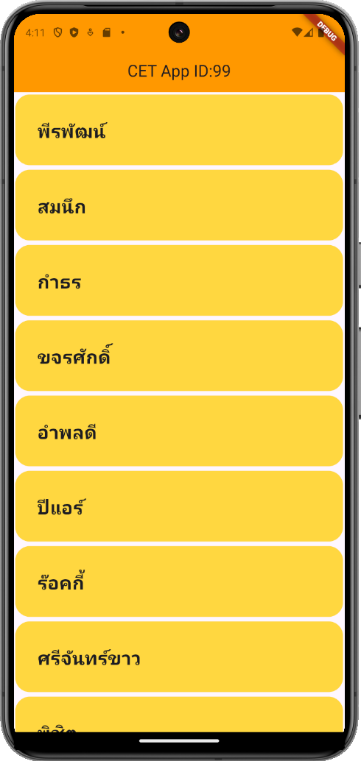
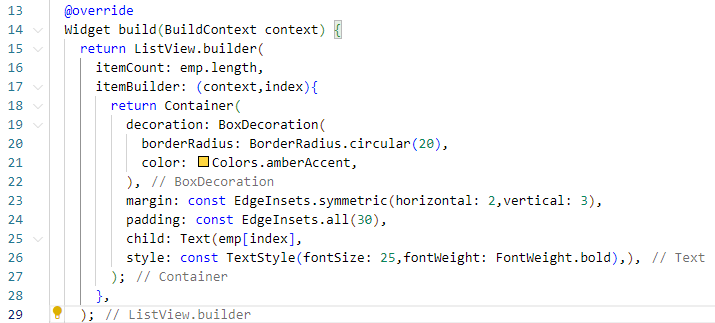
1. การใช้งาน Listview widget List View คือ Widget ที่สามารถแสดงข้อมูลในรูปแบบรายการ และหากมีจำนวนมากเกินพื้นที่ จะสามารถเลื่อนเพื่อดูรายการโดยการ Scroll Up/ Scroll Down ได้ โดย List คือ โครงสร้างข้อมูลส่วน Listview.builder สำหรับที่แสดงรายการที่มีจำนวนมากได้โดยมีการระบุจำนวนรายการผ่าน Properties ที่ชื่อว่า itemcount

 สำหรับงานในข้อนี้ให้นักศึกษาลบโค้ดในส่วนของ Method Build ของ item.dart ให้เหลือแค่ส่วนของ Center() และลบคำสั่งจากข้อ 9 ที่เพิ่มไปในส่วนของ State ออกให้หมดสุดท้ายจะเหลือโค้ดที่จำเป็นตามรูป

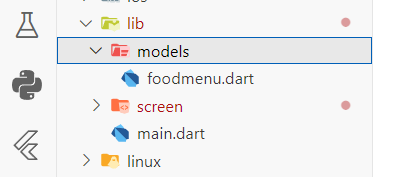
จากนั้นจะทำการสร้าง ListView Widgert ขึ้นมาเพื่อแสดงข้อมูลของพนักงานโดยจะนำข้อมูลทั้งหมดใส่ไปใน List ก่อนเพื่อเตรียมข้อมูลไว้โดยใส่ในส่วนของ State

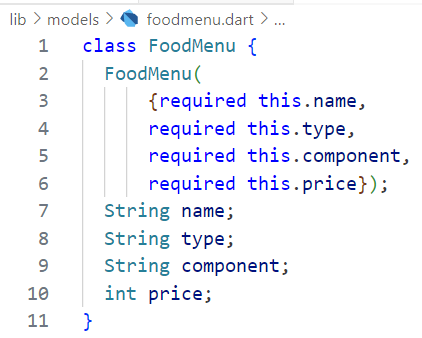


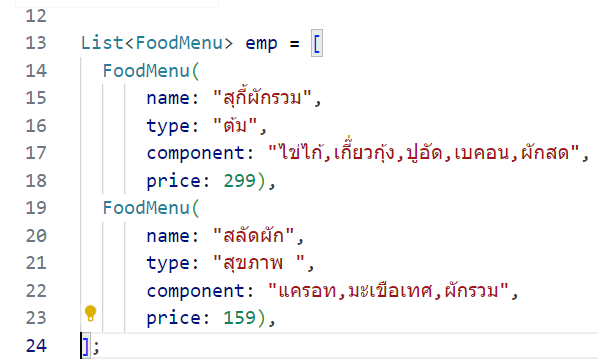
ในการดึงข้อมูลจาก List มาใช้งานจะดึงผ่าน ListView.builder และ จำนวนของข้อมูลจะถูกระบุโดย itemcount จากนั้นใช้ itembuilder สร้างรายการข้อมูลตามลำดับของ index ที่ได้มา ในการแสดงค่าข้อมูลที่ดึงมาจาก List จะใช้ Text Widget ดึงค่าโดยระบุชื่อ List และอ้างอิงด้วย index ดังนี้



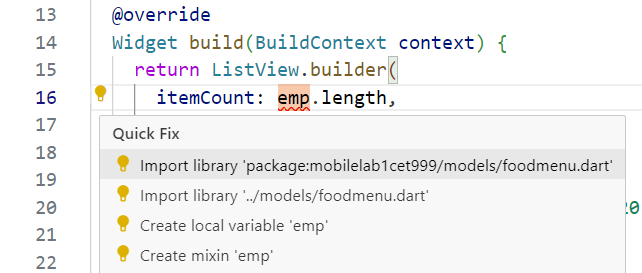
1. การใช้งาน Data Models เพื่อใช้จัดกลุ่มข้อมูล หรืออาจเรียกว่าเป็นโครงสร้างข้อมูล โดยเราจะสร้างข้อมูลที่เกี่ยวข้องกันไว้ในกลุ่มข้อมูลเดียวกันอย่างมีรูปแบบ(Models) โดยในการสร้าง Data Models เราต้องสร้าง Class ขึ้นมาเพื่อบรรจุตัวข้อมูลโดยในตัวอย่างที่ผ่านมาเราได้มีการเตรียมรูปของเมนูอาหารไว้ ขั้นต่อไปเราจะสร้าง Data models ที่จัดเก็บรายละเอียดของเมนูอาหารแต่ละชนิด
   1. ขั้นแรกเราจะสร้างโฟลเดอร์ใหม่ภายใต้ lib ชื่อ models เพื่อเอาไว้เป็น Class ของ Data Models และสร้างไฟล์ใหม่ชื่อ foodmenu.dart ในโฟลเดอร์ models ในไฟล์นี้จะใส่ข้อมูลที่เกี่ยวข้องกับอาจารย์แต่ละเมนู



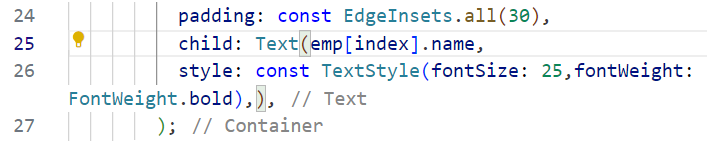
* 1. สร้าง Class ชื่อ FoodMenu ภายในบรรจุตัวแปรเพื่อใช้เก็บข้อมูลรายละเอียดขอเมนูอาหารประกอบด้วย ชื่ออาหาร(name), ประภาทอาหาร(type), ส่วนประกอบอาหาร(component) และ ราคาอาหาร(price)
  2. ในการใช้งานคลาสนั้นต้องทำให้เป็น Object โดยสามารถสร้างได้โดยใช้คำสั่ง required(ระบุว่าจำเป็นต้องส่งเข้ามาทำงาน) และ this.ชื่อตัวแปร
  3. ทำการลบ List data ใน item.dart ออกไป แล้วมาสร้าง List FoodMenu ใหม่ใน foodmenu.dart และใส่ข้อมูลตามObject ที่เตรียมไว้



* 1. ที่ไฟล์ item.dart จะต้อง import ไฟล์ foodmenu.dart เข้าไปเพื่อให้เจอข้อมูลที่เตรียมไว้ โดยคลิ๊กที่ข้อความที่มีเส้นสีแดง แล้วคลิ๊กที่หลอดไฟเลือก import foodmenu.dart เข้ามา



* 1. จะพบว่ายังมีส่วน Error ที่มีเส้นสีแดงอยู่ในส่วนนี้แก้ไขโดยต้องระบุว่าให้รันค่าตาม index ของ name

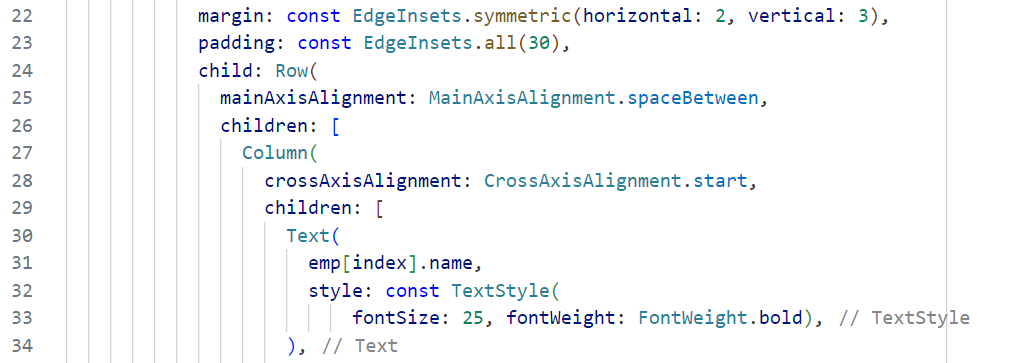


**บันทึกผลการรัน...................................รูป...................................................................**

**A screen shot of a phone

AI-generated content may be incorrect.**

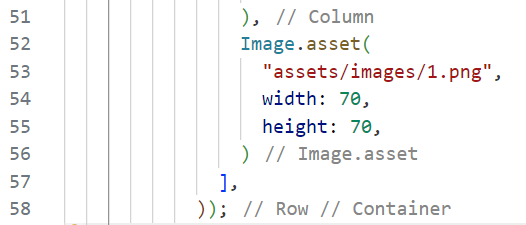
* 1. จากข้อ 11(f) พบว่าข้อมูลจะถูกเรียกไปแสดงเพียงแค่ชื่ออาหารเท่านั้น หากต้องการเรียกข้อมูลอื่นๆ ด้วย ก็ทำได้โดยสร้าง Text widget ขึ้นมาใหม่และเรียกผ่าน index.ชื่อข้อมูลที่ต้องการ เช่น data[index].name, data[index].price โดยปรับแต่ง style ให้ดูเหมาะสม ทั้งนี้ต้องทำงานรวมกับ Row widget และ Column widget



* 1. ตัวอย่างการเพิ่มข้อความเข้าไปเช่น ประเภทอาหาร และราคา จากนั้นใช้การระบุค่าจากตัวแปรโดยต้องใส่ $ ก่อนจากนั้นระบุชื่อ Data models ค่า index และ ชื่อข้อมูลในโครงสร้าง ในที่นี้คือ type และ price



* 1. ขั้นต่อมาทดลองเพิ่มรูปเข้าไป โดยในการเพิ่มรูปจะต้องอยู่คนละ Column กับดังนั้นต้องเพิ่มนอกบล๊อคของ Column เดิม ดังตัวอย่าง หากเกิดปัญหาล้นขอบให้ครอบ Container ด้วย SingleChildScrollView

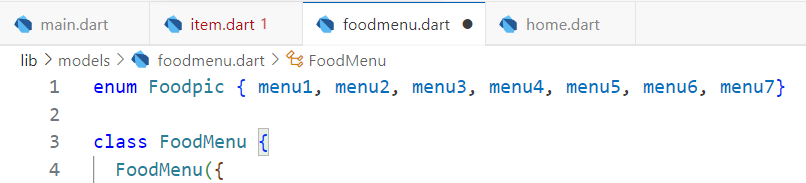


**บันทึกผลการรัน...................................รูป...................................................................**

**A screenshot of a phone

AI-generated content may be incorrect.**

* 1. จะเห็นว่าจากข้อ 11(i) รูปที่ได้ทุกเมนูจะเป็นรูปเดิมหมดเลย ซึ่งความต้องการของเราคือต้องการให้แสดงรูปให้ตรงกับเมนูแต่ละอย่าง เช่น สเต็กหมู ก็ต้องแสดงรูป สเต็กหมู โดยคำสั่งที่จะใช้ทำนั้นคือคำสั่ง Enum โดยจะต้องสร้าง enum โดยจะให้ชื่อว่า Foodpic เพื่อเก็บค่าของตัวแปร menu1-menu7 ตามรายการอาหารที่เรามี 7 อย่าง



จากนั้นต้องทำให้ค่าของ menu1-7 ถูกระบุอยู่ใน List FoodMenu เพื่อจับคู่ระหว่างข้อมูลรูปภาพที่เราจะนำมาใส่ใน menu1- menu7 กับชุดข้อมูลรายการอาหารเพิ่มให้ตรงกัน รูปต้องตรงกับชื่อ ซึ่งเราต้องกำหนด required this.foodpic ใน class FoodMenu และกำหนดชื่อ้อมูลของตัวแปรใหม่ foodpic ให้เป็นชนิด Foodpic Enum จากนั้นจึงสามารถจับคู่ ที1 – ที7 กับรายการอาหารใน List FoodMenu ได้ด้วยคำสั่ง foodpic: Foodpic.menu1



ย้ายมาจาก item.dart มาไว้ข้างใต้

**บันทึกผลการรัน...................................รูป...................................................................**

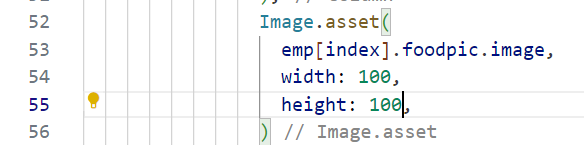
A screen shot of a phone

AI-generated content may be incorrect.

* 1. จากข้อ 11(j) ยังไม่มีการเปลี่ยนแปลงใดๆ กับผลการรันเนื่องจากยังไม่ได้มีการระบุตำแหน่งที่อยู่ของรูปภาพและยังไม่มีการเรียกตัวแปร enum Foodpic ขั้นตอนต่อไปต้องระบุค่าที่จำเป็นในการเรียกใช้งานโดยจะใช้ตัวแปรชื่อ image ที่สร้างขึ้นมาใหม่เพื่อเก็บพาธที่อยู่ของไฟล์ภาพ โดยต้องระบุให้ตรงกับรายการอาหารที่จับคู่กับ menu1-7 จะสลับกันไม่ได้ดังนี้



ในการเรียกใช้งานจะต้องแก้ไขที่ไฟล์ item.dart ในส่วนที่เรียกใช้งานรูปภาพโดยแก้ไขจากการเรียกรูปภาพด้วยพาธที่อยู่เป็น enum ที่สร้างไว้ใน List FoodMenu emp ด้วย index ดังนี้



ปรับแต่งขนาดรูปภาพให้เหมาะสม

**บันทึกผลการรัน...................................รูป...................................................................**

**A screen shot of a phone

AI-generated content may be incorrect.**

**คำสั่ง** ให้นักศึกษากำหนดสีพื้นหลังของรายการอาหารแต่ละรายการ โดยให้แยกสีตามประเภทของอาหาร ตัวอย่าง



**บันทึกผลการรัน.................................................................................รูป......................................................................**

**A screen shot of a cell phone

AI-generated content may be incorrect.**

**บันทึกโค๊ดส่วนที่มีการปรับปรุงพร้อมคำอธิบาย...................................รูป...................................................................**

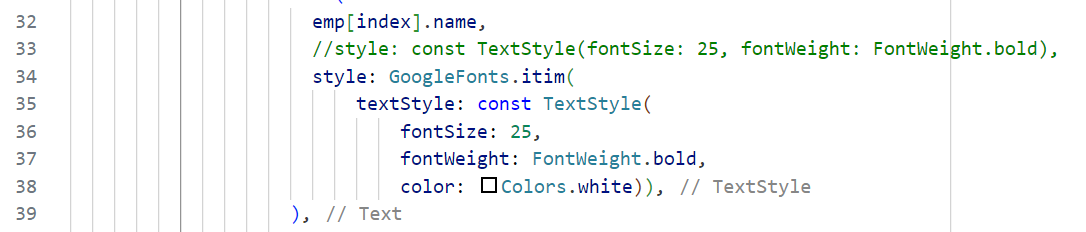
**A screen shot of a computer program

AI-generated content may be incorrect.**

**คำอธิบาย:**

1. **สร้างฟังก์ชัน getBackgroundColor(): เราได้สร้างฟังก์ชันใหม่ขึ้นมาเพื่อจัดการเรื่องสีโดยเฉพาะ ฟังก์ชันนี้รับ "ประเภทอาหาร" (String) เข้ามา แล้วใช้ switch statement เพื่อตรวจสอบและคืนค่า Color ที่แตกต่างกันออกไป ทำให้โค้ดส่วนแสดงผล (UI) กับส่วนตรรกะ (Logic) แยกจากกันชัดเจน**
2. **เรียกใช้ฟังก์ชันใน BoxDecoration: ใน Container ที่ใช้สร้างการ์ดแต่ละใบ เราได้แก้ไขคุณสมบัติ color ให้ไม่เป็นค่าสีคงที่อีกต่อไป แต่เปลี่ยนมาเป็นการเรียกใช้ฟังก์ชัน getBackgroundColor() โดยส่งประเภทอาหารของรายการนั้นๆ (emp[index].type) เข้าไปเป็นพารามิเตอร์**
3. **ผลลัพธ์: การทำเช่นนี้ทำให้สีพื้นหลังของการ์ดแต่ละใบสามารถเปลี่ยนแปลงได้แบบไดนามิกตามข้อมูล "ประเภทอาหาร" ที่กำหนดไว้ใน List ของเรา**

**\*\*\***การเรียกใช้งาน Google fonts สามารถเข้าไปเลือกได้ที่ google font เมื่อได้ชื่อแล้วให้จดไว้ จากนั้นไปที่ <https://pub.dev/> ค้นคำว่า Google font ใช้คำสั่ง **flutter pub add google\_fonts** ที่ terminal ของ VsCode ในการเรียกใช้ทำได้โดย



1. ให้นักศึกษาใช้ความรู้ที่ได้จากใบงานที่ 1 และ 2 พัฒนาแอพพลิเคชั่นนำเสนอข้อมูลเพื่อนโดยกำหนดให้รูปเพื่อนแสดงทางขวามือ และข้อมูลที่จะแสดงให้นักศึกษาเลือกว่าจะนำเสนออะไรเช่น ชื่อ-นามสกุล, ชื่อเล่น, อายุ, บ้านเกิด, อาหารที่ชอบ, กิจกรรมที่ชอบ(เล่นเกมส์อะไร, เล่นกีฬาอะไร,ดูหนังแนวไหน, ฟังเพลง ฯลฯ) ให้ใช้ความรู้การจัดการ Font, สีพื้นหลัง, สีข้อความ