1. จาก ตัวอย่าง โปรแกรม แสดงความแตกต่างระหว่าง row และ column

|  |
| --- |
| import 'package:flutter/material.dart';  final Color darkBlue = Color.fromARGB(255, 18, 32, 47);  void main() {  runApp(MyApp());  }  class MyApp extends StatelessWidget {  @override  Widget build(BuildContext context) {  return MaterialApp(  theme: ThemeData.dark().copyWith(scaffoldBackgroundColor: darkBlue),  debugShowCheckedModeBanner: false,  home: Scaffold(  body: Center(  child: MyWidget(),  ),  ),  );  }  }  class MyWidget extends StatelessWidget {  @override  Widget build(BuildContext context) {  return Column(  mainAxisSize: MainAxisSize.max,  children: [  BlueBox(),  BlueBox(),  BlueBox(),  ],  );  }  }  class BlueBox extends StatelessWidget {  @override  Widget build(BuildContext context) {  return Container(  width: 50,  height: 50,  decoration: BoxDecoration(  color: Colors.blue,  border: Border.all(),  ),  );  }  } |

1. กดปุ่ม run

2. เปลี่ยน Column เป็น Row และ run อีกครั้ง

ผลการทำงานและอธิบายการทำงานของ row และ column

|  |
| --- |
| Rowคือการเรียงแถวแนวนอน  Columnคือการเรียงแถวแนวตั้ง |

1. เปลี่ยนค่า property เป็น min

mainAxisSize: MainAxisSize.min,

ผลการทำงานและอธิบายสิ่งที่เปลี่ยนแปลง

|  |
| --- |
| Blueblockไปอยูตรงกลาง |

1. แก้ไข class MyWdiget เพิ่ม property mainAxisAlignment จากนั้น run และเปลี่ยนจาก start เป็น end

mainAxisAlignment: MainAxisAlignment.start,

|  |
| --- |
| class MyWidget extends StatelessWidget {  @override  Widget build(BuildContext context) {  return Row(  //mainAxisSize: MainAxisSize.min,  mainAxisAlignment: MainAxisAlignment.start,  children: [  BlueBox(),  BlueBox(),  BlueBox(),  ],  );  }  } |

ผลการทำงานและอธิบายสิ่งที่เปลี่ยนแปลง

|  |
| --- |
| Blueblockไปอยูด้านซ้าย |

1. แก้ไข class MyWidget เพิ่ม Flexible widget โดย run และเปลี่ยนจาก loose เป็น tight

|  |
| --- |
| class MyWidget extends StatelessWidget {  @override  Widget build(BuildContext context) {  return Row(  children: [  BlueBox(),  Flexible(  fit: FlexFit.loose,  flex: 1,  child: BlueBox(),  ),  Flexible(  fit: FlexFit.loose,  flex: 1,  child: BlueBox(),  ),  ],  );  }  } |

ผลการทำงานเมื่อ ลองเปลี่ยน flex เป็นเลขอื่น เช่น 1,2,3 และอธิบายผลลัพธ์

|  |
| --- |
| Blueboxมีขนาดต่างกันเพราะค่าflex |

1. แก้ไข class MyWidget เปลี่ยน Flexible widget เป็น expand widget ยกเว้น box แรก

Expanded(child: BlueBox(),),

ผลการทำงาน

|  |
| --- |
|  |

1. แก้ไข class MyWidget เป็นดังนี้

|  |
| --- |
| class MyWidget extends StatelessWidget {  @override  Widget build(BuildContext context) {  return Row(  crossAxisAlignment: CrossAxisAlignment.center,  textBaseline: TextBaseline.alphabetic,  children: [  Text(  'Hey!',  style: TextStyle(  fontSize: 30,  fontFamily: 'Futura',  color: Colors.blue,  ),  ),  Text(  'Hey!',  style: TextStyle(  fontSize: 50,  fontFamily: 'Futura',  color: Colors.green,  ),  ),  Text(  'Hey!',  style: TextStyle(  fontSize: 40,  fontFamily: 'Futura',  color: Colors.red,  ),  ),  ],  );  }  } |

ผลการทำงาน

|  |
| --- |
|  |

1. แก้ไข class MyWidget เป็นดังนี้

|  |
| --- |
| class MyWidget extends StatelessWidget {  @override  Widget build(BuildContext context) {  return Row(  crossAxisAlignment: CrossAxisAlignment.center,  textBaseline: TextBaseline.alphabetic,  children: [  Icon(  Icons.widgets,  size: 50,  color: Colors.blue,  ),  Icon(  Icons.widgets,  size: 50,  color: Colors.red,  ),  ],  );  }  } |

ผลการทำงาน

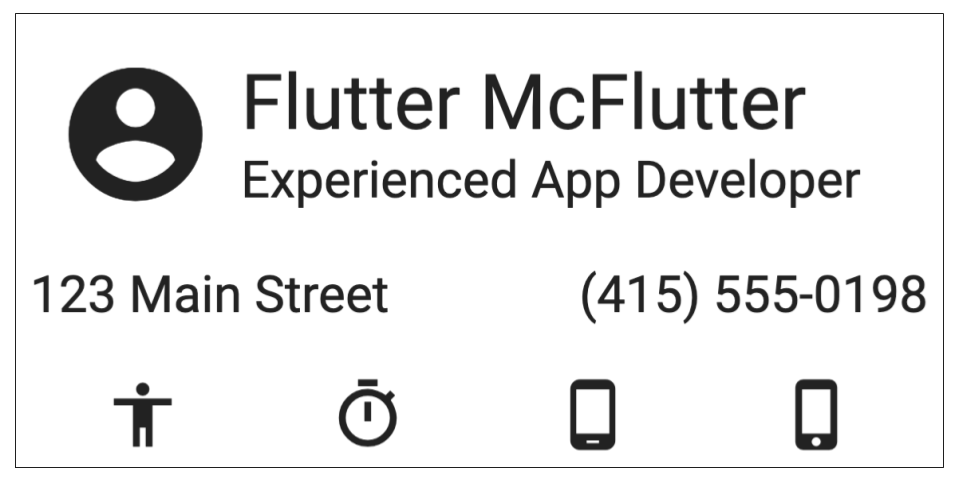
|  |
| --- |
|  |

1. แก้ไข class MyWidget โดยเพิ่ม icon อีก 1 icon และ ให้สีเป็น amber พร้อมทั้งเปลี่ยนรูป icon ด้วย

|  |
| --- |
| CODE  Icon(            Icons.wifi,            size: 50,            color: Colors.amber,          ) |

1. แก้ไข class MyWidget โดยแสดงผลลัพธ์ดังภาพ

เลือก รูปไอคอน ปุ่ม material design ที่เวบ material.io



|  |
| --- |
| CODE ต้นฉบับ  class MyWidget extends StatelessWidget {  @override  Widget build(BuildContext context) {  return Column(  mainAxisSize: MainAxisSize.min,  crossAxisAlignment: CrossAxisAlignment.stretch,  children: [  Row(  children: [  Padding(  padding: const EdgeInsets.all(8.0),  child: Icon(Icons.account\_circle, size: 50)),  Column(  crossAxisAlignment: CrossAxisAlignment.start,  mainAxisSize: MainAxisSize.min,  children: [  Text('Flutter McFlutter',  style: Theme.of(context).textTheme.headline5),  Text('Experienced App Developer'),  ],  ),  ],  ),  SizedBox(height: 8),  Row(  mainAxisSize: MainAxisSize.min,  mainAxisAlignment: MainAxisAlignment.start,  children: [  Text('123 Main Street'),  SizedBox(width: 50),  Text('415-555-0198'),  ],  ),  SizedBox(height: 16),  Row(children:  [Icon(  Icons.widgets,  size: 30,  color: Colors.red,  ),  ],  ),  ],  );  }  } |

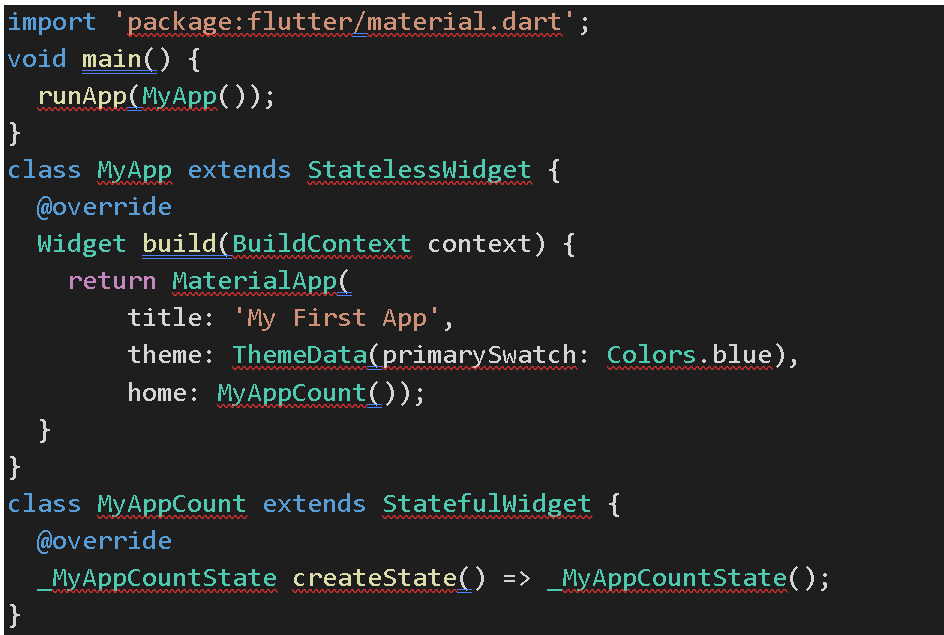
|  |
| --- |
| CODE แก้ไข  class MyWidget extends StatefulWidget {    const MyWidget({Key? key}) : super(key: key);    @override    State<MyWidget> createState() => \_MyWidgetState();  }  class \_MyWidgetState extends State<MyWidget> {    @override    Widget build(BuildContext context) {      return Column(        mainAxisSize: MainAxisSize.min,        crossAxisAlignment: CrossAxisAlignment.stretch,        children: [          Row(            mainAxisAlignment: MainAxisAlignment.center,            children: [              const Padding(                  padding: EdgeInsets.all(8.0),                  child: Icon(Icons.account\_circle, size: 50)),              Column(                crossAxisAlignment: CrossAxisAlignment.start,                mainAxisSize: MainAxisSize.min,                children: [                  Text('Flutter McFlutter',                      style: Theme.of(context).textTheme.headline5),                  const Text('Experienced App Developer'),                ],              ),            ],          ),          const SizedBox(height: 8),          Row(            mainAxisSize: MainAxisSize.min,            mainAxisAlignment: MainAxisAlignment.center,            children: [              const Text('123 Main Street'),              const SizedBox(width: 50),              const Text('415-555-0198'),            ],          ),          SizedBox(height: 16),          Row(            mainAxisAlignment: MainAxisAlignment.spaceEvenly,            children: [              Icon(                Icons.accessibility,                size: 30,                color: Colors.purple,              ),              Icon(                Icons.timer,                size: 30,                color: Colors.pinkAccent,              ),              Icon(                Icons.phone\_android,                size: 30,                color: Colors.white,              ),              Icon(                Icons.phone\_iphone,                size: 30,                color: Colors.tealAccent,              ),            ],          ),        ],      );    }  } |

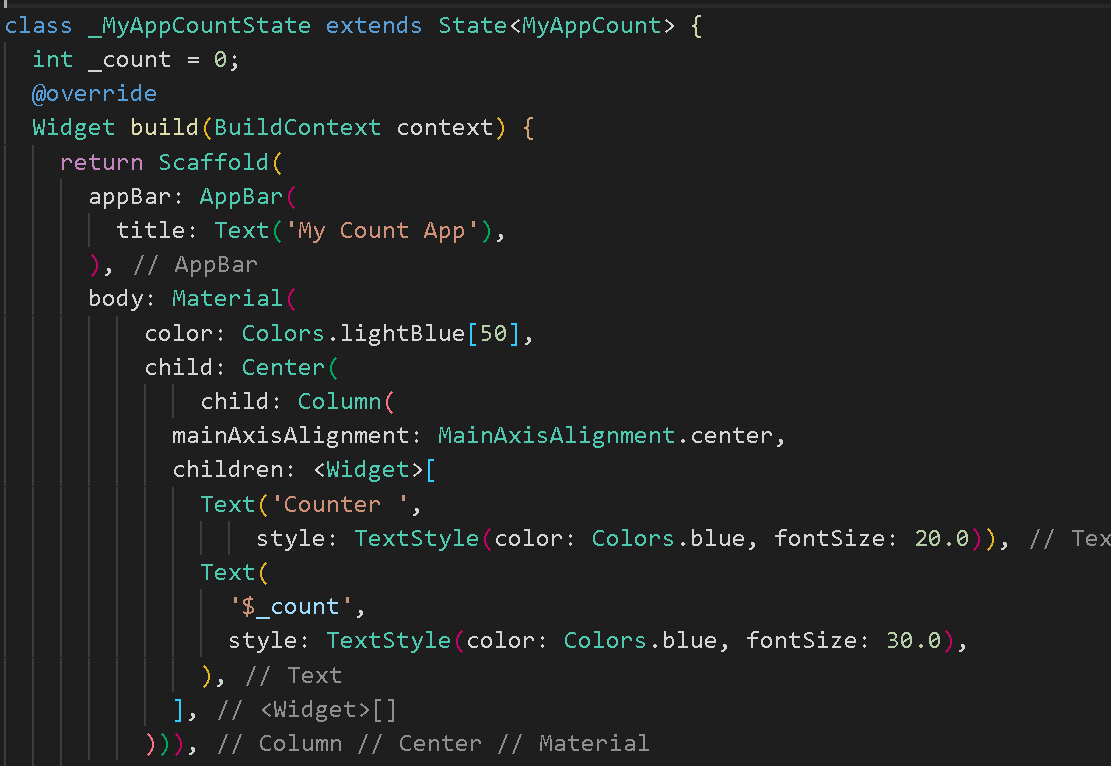
1. จงเขียนโปรแกรม ตาม slide ที่เรียนช่วงทฤษฎี ดังนี้
   1. Stateless widget

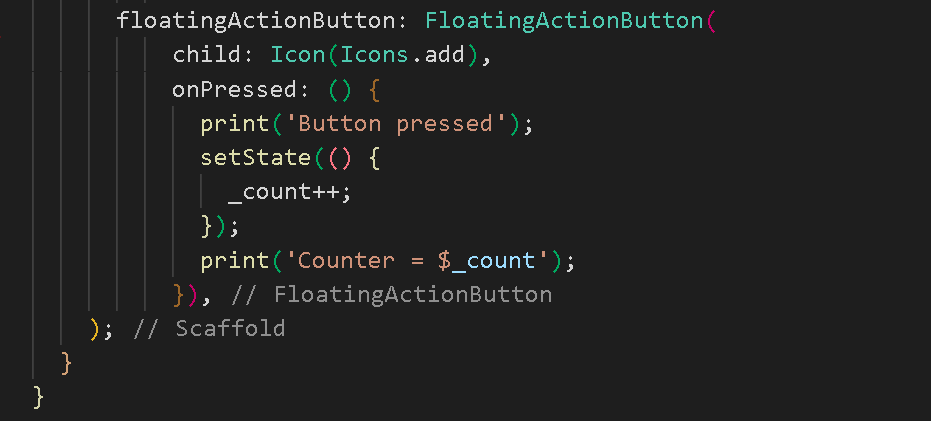


|  |
| --- |
| CODE และผลลัพธ์  import 'package:flutter/material.dart';  void main(List<String> args) {    runApp(MyApp());  }  class MyApp extends StatelessWidget {    const MyApp({Key? key}) : super(key: key);    @override    Widget build(BuildContext context) {      return MaterialApp(        title: 'My First App',        theme: ThemeData(primaryColor: Colors.blue),        home: Scaffold(          appBar: AppBar(            title: const Text('My App'),          ),          body: Material(            color: Colors.lightBlue[50],            child: const Center(              child: Text(                'Hello Flutter App',                style: TextStyle(color: Colors.blue, fontSize: 20.0),              ),            ),          ),        ),      );    }  } |

* 1. Stateful widget







|  |
| --- |
| CODE และผลลัพธ์  import 'package:flutter/cupertino.dart';  import 'package:flutter/material.dart';  void main(List<String> args) {    runApp(Myapp());  }  class Myapp extends StatelessWidget {    const Myapp({Key? key}) : super(key: key);    @override    Widget build(BuildContext context) {      return MaterialApp(        title: 'My First App',        theme: ThemeData(primarySwatch: Colors.blue),        home: const MyAppCount(),      );    }  }  class MyAppCount extends StatefulWidget {    const MyAppCount({Key? key}) : super(key: key);    @override    State<MyAppCount> createState() => \_MyAppCountState();  }  class \_MyAppCountState extends State<MyAppCount> {    int \_count = 0;    @override    Widget build(BuildContext context) {      return Scaffold(        appBar: AppBar(          title: const Text('My Count App'),        ),        body: Material(            color: Colors.lightBlue[50],            child: Center(                child: Column(              mainAxisAlignment: MainAxisAlignment.center,              children: <Widget>[                const Text('Counter ',                    style: TextStyle(color: Colors.blue, fontSize: 20.0)),                Text('$\_count',                    style: TextStyle(color: Colors.blue, fontSize: 20.0)),              ],            ))),        floatingActionButton: FloatingActionButton(          child: Icon(Icons.add),          onPressed: () {            print('Button preswd');            setState(() {              \_count++;            });            print('counter = $\_count');          },        ),      );    }  } |