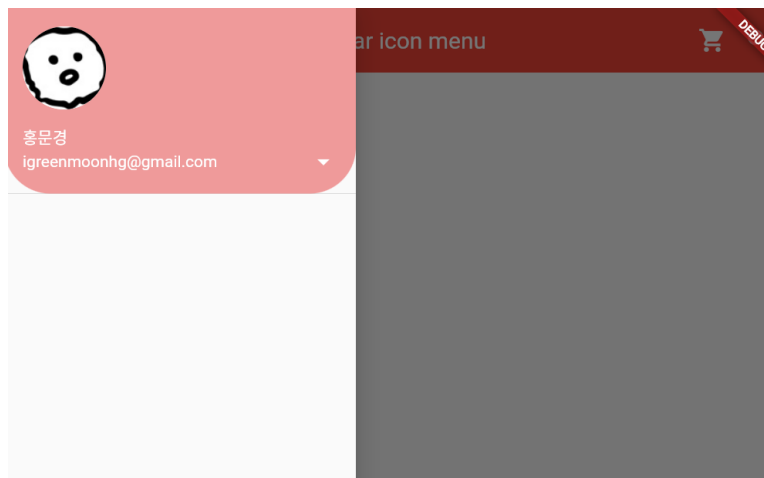
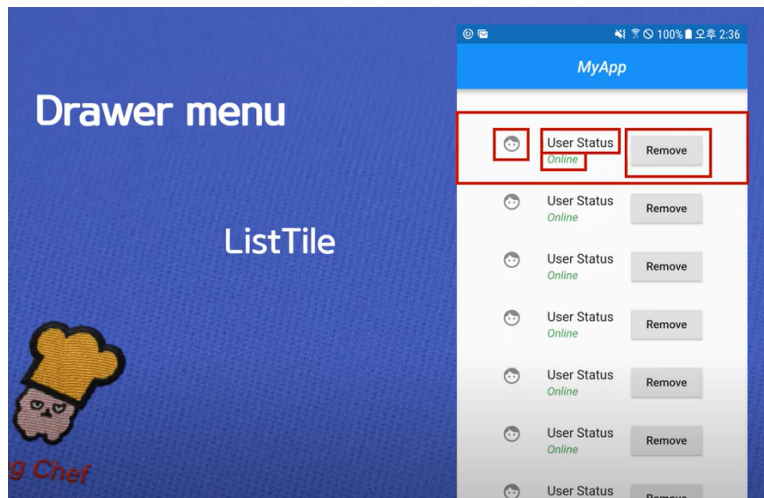




Basic 15-19 순한맛

15강. Drawer 메뉴 만들기 1

- Drawer를 추가하면 자동으로 햄버거 메뉴가 생성됨
- Drawer는 Scaffold 내 속성



```

drawer: Drawer(
  child: ListView(
    padding: EdgeInsets.zero,
    children: <Widget>[

      UserAccountsDrawerHeader(
        currentAccountPicture: CircleAvatar(
          backgroundImage: AssetImage('image1.jpg'),
          backgroundColor: Colors.white,
        ), //CircleAvatar
        accountName: Text('홍문경'),
        accountEmail: Text('igreenmoonhg@gmail.com'),
        //화살표 만들기
        onDetailsPressed: () {
          print('arrow is clicked'); //화살표 클릭하면 디버그 콘솔에서 확인할 문자
        }, //UserAccountDrawerHeader

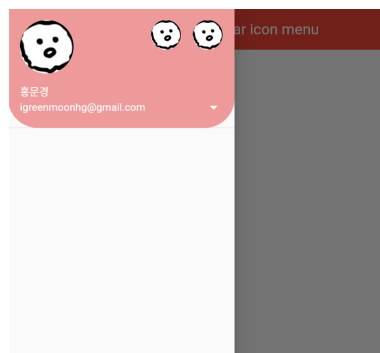
        decoration: BoxDecoration( //박스 속성 꾸미기
          color: Colors.red[200],
          borderRadius: BorderRadius.only( //하단 부분의곡선 효과 지정
            bottomLeft: Radius.circular(40.0),
            bottomRight: Radius.circular(40.0))), //Box 꾸미기

      ),
    ],
  ),
)

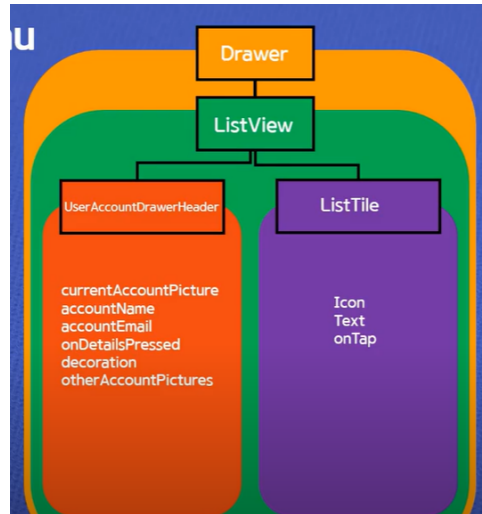
```

16강. Drawer 메뉴 만들기 2

- **otherAccountsPictures** 로 다른 계정 이미지 추가
 - 한 개 이상의 다른 계정 이미지 추가 가능 >위젯 리스트 이용



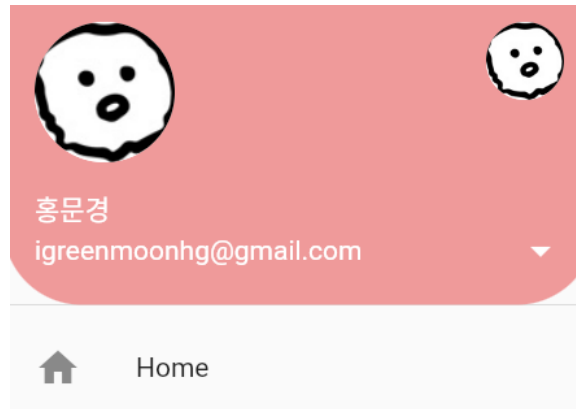
otherAccountsPictures



```

drawer: Drawer(
  child: ListView(padding: EdgeInsets.zero, children: <Widget>[
    UserAccountsDrawerHeader(
      currentAccountPicture: CircleAvatar(
        backgroundImage: AssetImage('image1.jpg'),
        backgroundColor: Colors.white,
      ),
      otherAccountsPictures: <Widget>[
        CircleAvatar(
          backgroundImage: AssetImage('image1.jpg'),
          backgroundColor: Colors.white,
        ),
        CircleAvatar(
          backgroundImage: AssetImage('image1.jpg'),
          backgroundColor: Colors.white,
        ),
      ],
      accountName: Text('홍문경'),
      accountEmail: Text('igreenmoonhg@gmail.com'),
      onDetailsPressed: () {
        print('arrow is clicked');
      },
      decoration: BoxDecoration(
        color: Colors.red[200],
        borderRadius: BorderRadius.only(
          bottomLeft: Radius.circular(40.0),
          bottomRight: Radius.circular(40.0))),
    ),
  ]),
),

```



- `leading` 은 ListTile 시작점에 배치 위한 속성

```
ListTile(
  leading: Icon(
    Icons.home,
    color: Colors.grey[850],
  ),
  title: Text('Home'),
)
```

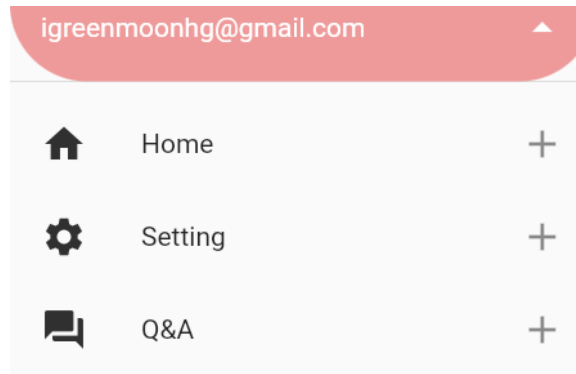
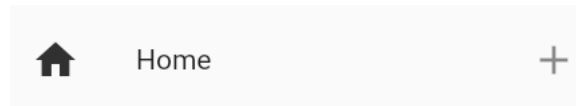
- 메뉴를 누르면 디버그 콘솔에 "home is clicked"라는 문자 출력을 위해 `onTap` 함수

```
ListTile(
  leading: Icon(
    Icons.home,
    color: Colors.grey[850],
  ),
  title: Text('Home'),
  onTap: () {
    print('Home is clicked');
  }
)
```

```
... Filter (e.g. text, !exclude)
Launching lib\main.dart on Chrome in
This app is linked to the debug serv
dDffc%3D/ws
Debug service listening on ws://127.
Running with sound null safety
Connecting to VM Service at ws://127
Restarted application in 87,462ms.
Restarted application in 14,377ms.
Restarted application in 15,091ms.
Home is clicked
```

- `trailing` 우측 끝나는 점에 배치 위한 속성

```
ListTile(
  leading: Icon(
    Icons.home,
    color: Colors.grey[850],
  ),
  title: Text('Home'),
  onTap: () {
    print('Home is clicked');
  },
  trailing: Icon(Icons.add),
)
```



```
ListTile(
  leading: Icon(
    Icons.home,
    color: Colors.grey[850],
  ),
  title: Text('Home'),
  onTap: () {
    print('Home is clicked');
  },
  trailing: Icon(Icons.add),
),
ListTile(
```

```

        leading: Icon(
          Icons.settings,
          color: Colors.grey[850],
        ),
        title: Text('Setting'),
        onTap: () {
          print('Setting is clicked');
        },
        trailing: Icon(Icons.add),
      ),
    ListTile(
      leading: Icon(
        Icons.question_answer,
        color: Colors.grey[850],
      ),
      title: Text('Q&A'),
      onTap: () {
        print('Q&A is clicked');
      },
      trailing: Icon(Icons.add),
    ),
  ),

```

▼ 전체 코드

```

import 'package:flutter/material.dart';

void main() => runApp(MyApp());

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Appbar',
      theme: ThemeData(primarySwatch: Colors.red),
      home: MyPage(),
    );
  }
}

class MyPage extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Appbar icon menu'),
        centerTitle: true,
        elevation: 0.0,
        actions: <Widget>[
          IconButton(
            icon: Icon(Icons.shopping_cart),
            onPressed: () {

```

```

        print('Shopping cart button is clicked');
    },
),
IconButton(
    icon: Icon(Icons.search),
    onPressed: () {
        print('search button is clicked');
    },
),
],
),
drawer: Drawer(
    child: ListView(
        padding: EdgeInsets.zero,
        children: <Widget>[
            UserAccountsDrawerHeader(
                currentAccountPicture: CircleAvatar(
                    backgroundImage: AssetImage('image1.jpg'),
                    backgroundColor: Colors.white,
                ),
                otherAccountsPictures: <Widget>[
                    // CircleAvatar(
                    //     backgroundImage: AssetImage('image1.jpg'),
                    //     backgroundColor: Colors.white,
                    // ),
                    CircleAvatar(
                        backgroundImage: AssetImage('image1.jpg'),
                        backgroundColor: Colors.white,
                    ),
                ],
                accountName: Text('홍문경'),
                accountEmail: Text('igreenmoonhg@gmail.com'),
                onDetailsPressed: () {
                    print('arrow is clicked');
                },
                decoration: BoxDecoration(
                    color: Colors.red[200],
                    borderRadius: BorderRadius.only(
                        bottomLeft: Radius.circular(40.0),
                        bottomRight: Radius.circular(40.0)),
                ),
            ),
            ListTile(
                leading: Icon(
                    Icons.home,
                    color: Colors.grey[850],
                ),
                title: Text('Home'),
                onTap: () {
                    print('Home is clicked');
                },
                trailing: Icon(Icons.add),
            ),
            ListTile(
                leading: Icon(

```

```

        Icons.settings,
        color: Colors.grey[850],
      ),
      title: Text('Setting'),
      onTap: () {
        print('Setting is clicked');
      },
      trailing: Icon(Icons.add),
    ),
    ListTile(
      leading: Icon(
        Icons.question_answer,
        color: Colors.grey[850],
      ),
      title: Text('Q&A'),
      onTap: () {
        print('Q&A is clicked');
      },
      trailing: Icon(Icons.add),
    ),
  ],
),
);
}
}

```

17강. 아리송한 BuildContext 이해하기

BuildContext

- A handle to the location of a widget in the widget tree.

widget tree에서 현재 widget의 위치를 알 수 있는 정보

Widget > build method

- 빌드 함수는 Scaffold Widget을 리턴
 - 이 때, widget tree상에서 어디에 존재하는 지에 대한 정보인 context를 넣어 리턴
- Each widget has its own BuildContext, which becomes the parent of the widget returned by the StatelessWidget.build or State.buildfunction.

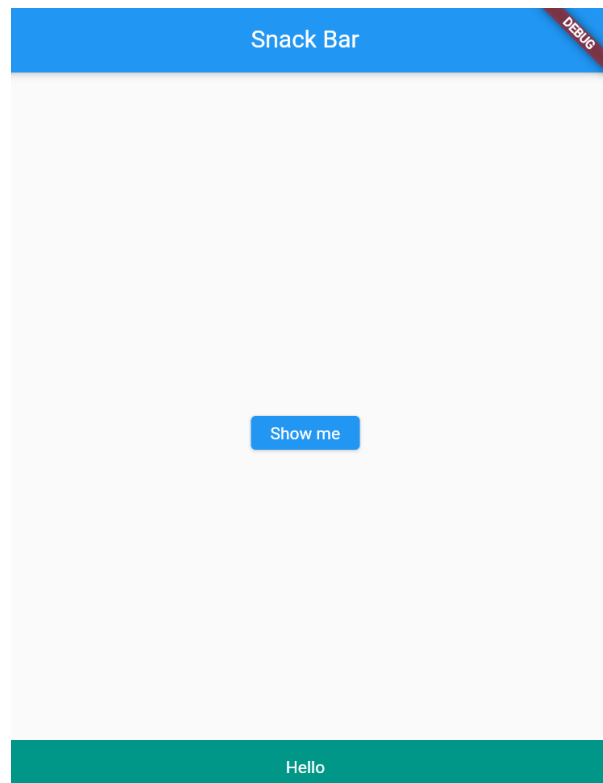
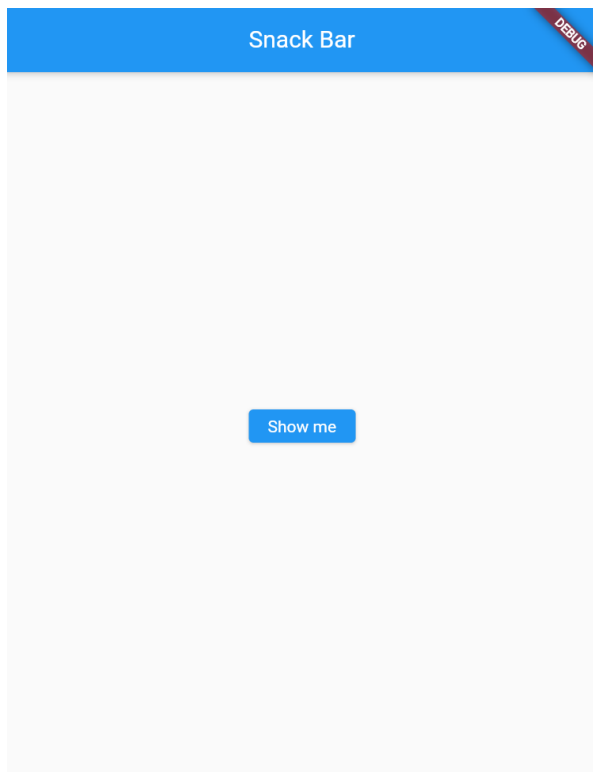
이 BuildContext는 stateless 위젯이나 state 빌드 메서드에 의해서 리턴된 위젯의 부모가 된다.

19강. 빌더 위젯 없이 스낵바 만들기과 토스트 메시지 구현하기

1. 빌더 위젯없이 Snack bar 만들기

2. Toast message 구현하기

사용자에게 짧은 정보를 팝업 형태로 제시하는 것



```
import 'package:flutter/material.dart';

void main() {
```

```

    runApp(MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({Key? key}) : super(key: key);

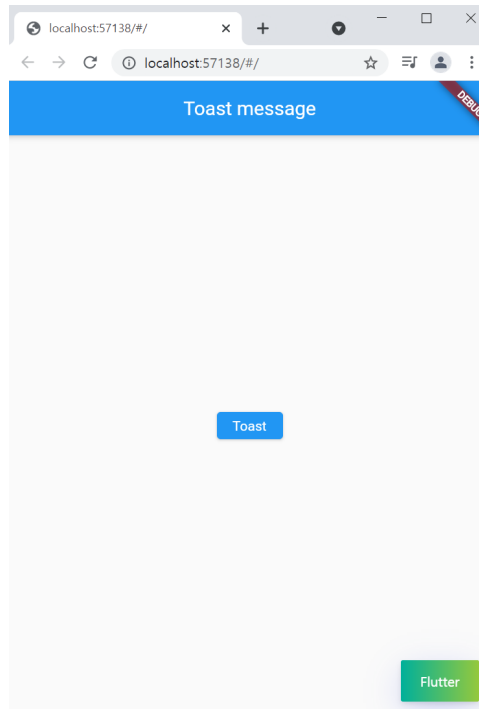
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      theme: ThemeData(primarySwatch: Colors.blue),
      home: MyPage(),
    );
  }
}

class MyPage extends StatelessWidget {
  const MyPage({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Snack Bar'),
        centerTitle: true,
      ),
      body: MySnackBar(),
    );
  }
}

class MySnackBar extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Center(
      child: ElevatedButton(
        child: Text('Show me'),
        onPressed: () {
          ScaffoldMessenger.of(context).showSnackBar(SnackBar(
            content: Text(
              'Hello',
              textAlign: TextAlign.center,
              style: TextStyle(color: Colors.white),
            ),
            backgroundColor: Colors.teal,
            duration: Duration(milliseconds: 1000),
          ));
        },
      ),
    );
  }
}

```



```
import 'package:flutter/material.dart';
import 'package:fluttertoast/fluttertoast.dart';

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      theme: ThemeData(primarySwatch: Colors.blue),
      home: MyPage(),
    );
  }
}

class MyPage extends StatelessWidget {
  const MyPage({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Toast message'),
        centerTitle: true,
      ),
```

```

        body: Center(
          child: ElevatedButton(
            onPressed: () {
              fluttertoast();
            },
            child: Text('Toast'),
            // color: Colors.blue,
          ),
        ),
      );
    }
  }

  void fluttertoast() {
    Fluttertoast.showToast(
      msg: 'Flutter',
      gravity: ToastGravity.BOTTOM,
      backgroundColor: Colors.redAccent,
      fontSize: 20.0,
      textColor: Colors.white,
      toastLength: Toast.LENGTH_SHORT);
  }
}

```

