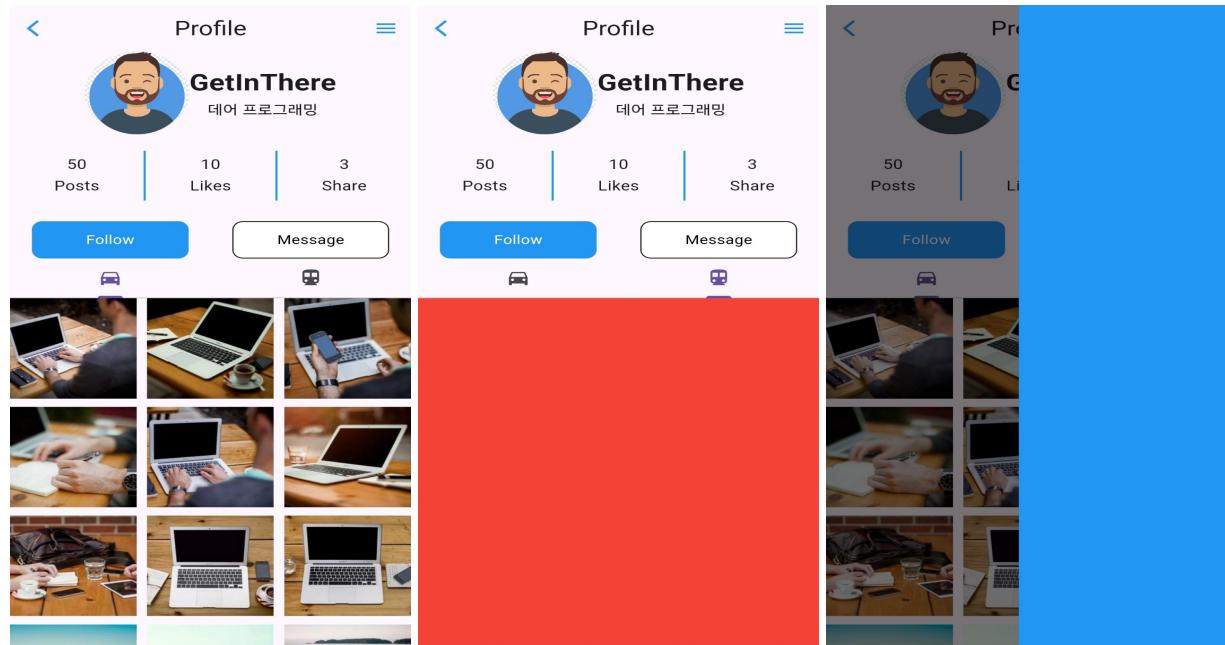


## Chapter 06 프로필 앱 만들기

## 완성 화면



## 1. 기본 세팅

 android [profile\_app\_and]  
▼ assets  
     avatar.png  
► ios

assets 폴더를 만들고 사진 파일을 넣는다.

```
0
1      # To add assets to your application
2  assets:
3      - assets/
4      #      - images/a_dot_ham.jpeg
5
```

pubspec.yaml 을 설정하고 pub get 을 누른다.

## 2. 앱 테마 만들기

## theme.dart

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

```

const MaterialColor primaryWhite = MaterialColor(
  0xFFFFFFFF,
  <int, Color>{
    50: Color(0xFFFFFFFF),
    100: Color(0xFFFFFFFF),
    200: Color(0xFFFFFFFF),
    300: Color(0xFFFFFFFF),
    400: Color(0xFFFFFFFF),
    500: Color(0xFFFFFFFF),
    600: Color(0xFFFFFFFF),
    700: Color(0xFFFFFFFF),
    800: Color(0xFFFFFFFF),
    900: Color(0xFFFFFFFF),
  },
);
}

ThemeData theme() {
  return ThemeData(
    primarySwatch: primaryWhite,
    appBarTheme: AppBarTheme(
      iconTheme: IconThemeData(color: Colors.blue), //앱바 아이콘 색깔 정하기
    )
  );
}
}

```

☞ 앱의 전체적인 색상을 흰색으로 만든다. 기본 색이 흰색이기 때문에 흰색을 원한다면 굳이 테마를 만들 필요는 없다.

## main.dart

```

import 'package:flutter/material.dart';
import 'package:profile_app/theme.dart';

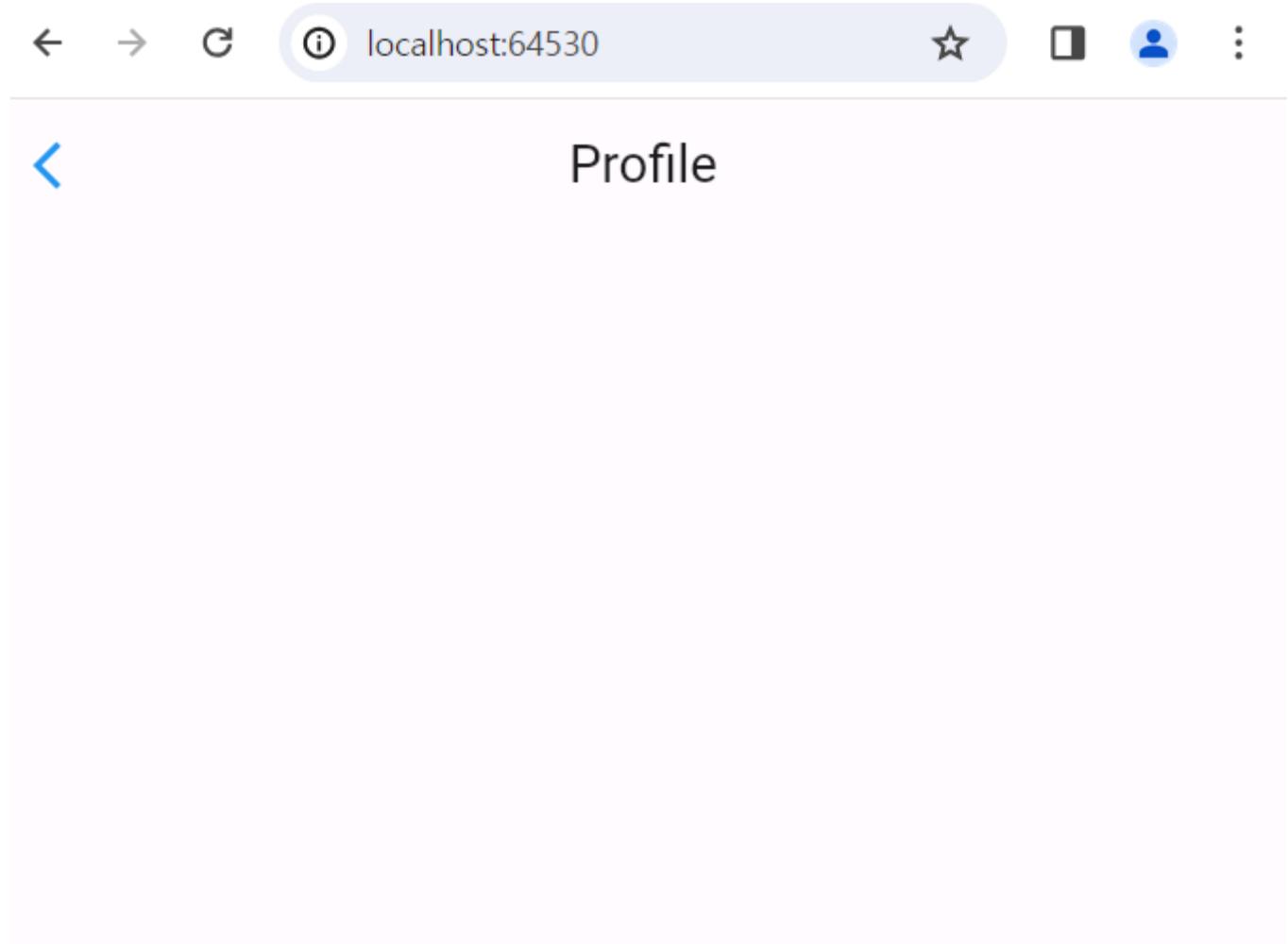
void main() {
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: theme(),
      home: Scaffold(
        appBar: _buildAppBar(),
      ),
    );
  }
}

```

```
AppBar _buildAppBar() {
    return AppBar(
        leading: Icon(Icons.arrow_back_ios), // 앱바 아이콘 만들기
        title: Text("Profile"),
        centerTitle: true,
    );
}
```



### 3. Drawer 만들기

#### 3.1. Drawer 란?

💡 `Drawer`는 앱의 화면 왼쪽에서 나오는 메뉴를 의미한다. 주로 앱의 내비게이션 링크나 설정 등을 제공한다. `endDrawer`를 사용하면 화면 오른쪽에 표시된다.

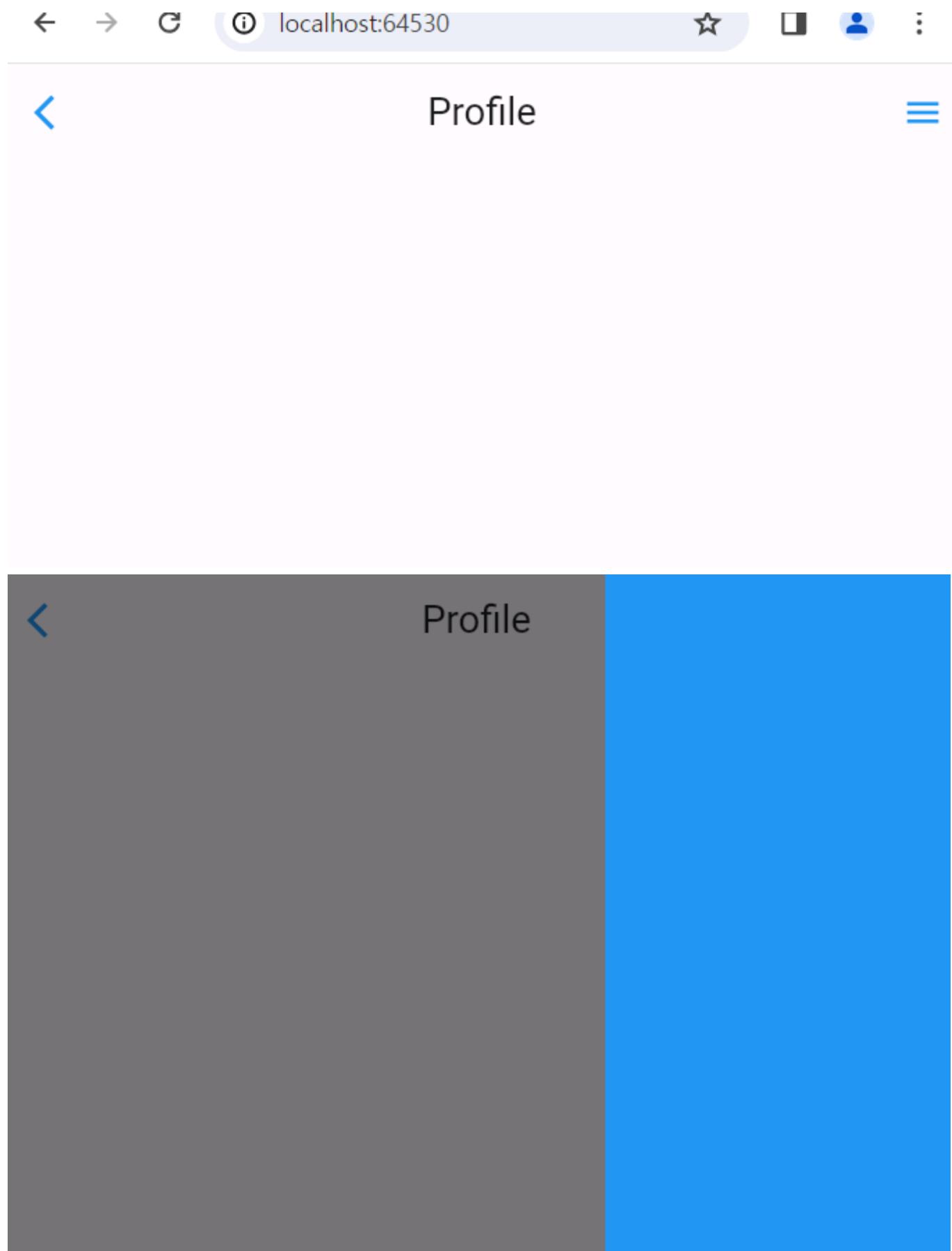
```
import 'package:flutter/material.dart';
import 'package:profile_app/theme.dart';

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

```
class MyApp extends StatelessWidget {
  const MyApp({super.key});

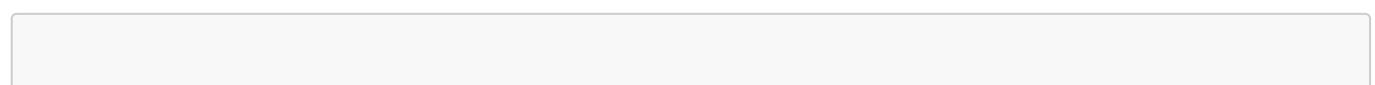
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: theme(),
      home: Scaffold(
        appBar: _buildAppBar(),
        endDrawer: Container(
          width: 200,
          height: double.infinity,
          color: Colors.blue,
        ),
      ),
    );
  }

  AppBar _buildAppBar() {
    return AppBar(
      leading: Icon(Icons.arrow_back_ios),
      title: Text("Profile"),
      centerTitle: true,
    );
  }
}
```



### 3.2. 컨포넌트 분리하기

**main.dart**



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

import 'components/profile_drawer.dart';

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

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: theme(),
      home: Scaffold(
        appBar: _buildAppBar(),
        endDrawer: ProfileDrawer(),
      ),
    );
  }

  AppBar _buildAppBar() {
    return AppBar(
      leading: Icon(Icons.arrow_back_ios),
      title: Text("Profile"),
      centerTitle: true,
    );
  }
}
```

## profile\_drawer.dart

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

class ProfileDrawer extends StatelessWidget {
  const ProfileDrawer({
    super.key,
  });

  @override
  Widget build(BuildContext context) {
    return Container(
      width: 200,
      height: double.infinity,
      color: Colors.blue,
    );
  }
}
```

}

## 4. CircleAvatar 사용하기

### 4.1. CircleAvatar 란?

❷ `CircleAvatar`는 주로 사용자의 프로필 이미지나 이니셜을 원형으로 표시하기 위해 사용되는 위젯이다. 이 위젯을 사용하면 간단하게 원형의 아바타를 만들 수 있으며, 이미지나 텍스트 등을 쉽게 적용할 수 있다.

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

import 'components/profile_drawer.dart';

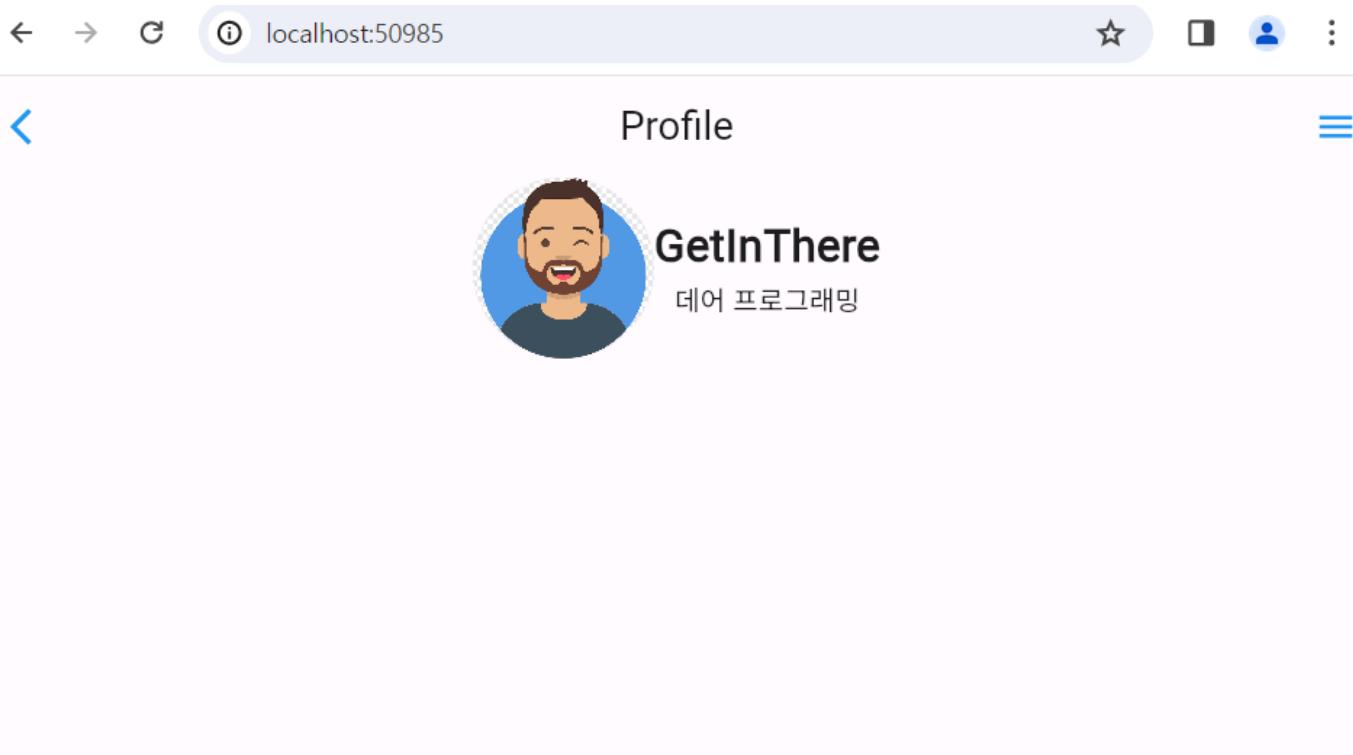
void main() {
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: theme(),
      home: Scaffold(
        appBar: _buildAppBar(),
        endDrawer: ProfileDrawer(),
        body: Column(
          children: [
            Row(
              mainAxisAlignment: MainAxisAlignment.center,
              children: [
                SizedBox(
                  width: 100,
                  height: 100,
                  child: CircleAvatar(
                    backgroundImage: AssetImage("assets/avatar.png"),
                  ),
                ),
                Column(
                  children: [
                    Text(
                      "GetInThere",
                      style:
                        TextStyle(fontSize: 25, fontWeight: FontWeight.w700),
                    ),
                    Text(
                      "데어 프로그래밍",
                    ),
                  ],
                ),
              ],
            ),
          ],
        ),
      ),
    );
  }
}
```

```
        style: TextStyle(
            fontSize: 15,
        ),
    ],
),
],
),
),
),
),
),
),
);
}

AppBar _buildAppBar() {
    return AppBar(
        leading: Icon(Icons.arrow_back_ios),
        title: Text("Profile"),
        centerTitle: true,
    );
}
}
```



## 4.2. 컴포넌트 분리하기

### main.dart

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

```
import 'components/profile_drawer.dart';
import 'components/profile_header.dart';

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

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: theme(),
      home: Scaffold(
        appBar: _buildAppBar(),
        endDrawer: ProfileDrawer(),
        body: Column(
          children: [
            ProfileHeader(),
          ],
        ),
      ),
    );
  }

  AppBar _buildAppBar() {
    return AppBar(
      leading: Icon(Icons.arrow_back_ios),
      title: Text("Profile"),
      centerTitle: true,
    );
  }
}
```

## components/profile\_header.dart

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

import 'header_avatar.dart';
import 'header_profile.dart';

class ProfileHeader extends StatelessWidget {
  const ProfileHeader({
    super.key,
  });

  @override
  Widget build(BuildContext context) {
    return Row(
      mainAxisAlignment: MainAxisAlignment.center,
```

```
        children: [
            HeaderAvatar(),
            HeaderProfile(),
        ],
    );
}
}
```

## components/header\_avatar.dart

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

class HeaderAvatar extends StatelessWidget {
    const HeaderAvatar({
        super.key,
    });

    @override
    Widget build(BuildContext context) {
        return SizedBox(
            width: 100,
            height: 100,
            child: CircleAvatar(
                backgroundImage: AssetImage("assets/avatar.png"),
            ),
        );
    }
}
```

## components/header\_profile.dart

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

class HeaderProfile extends StatelessWidget {
    const HeaderProfile({
        super.key,
    });

    @override
    Widget build(BuildContext context) {
        return Column(
            children: [
                Text(
                    "GetInThere",
                    style:
                        TextStyle(fontSize: 25, fontWeight: FontWeight.w700),
                ),
            ],
        );
    }
}
```

```
        Text(
            "데어 프로그래밍",
            style: TextStyle(
                fontSize: 15,
            ),
        ),
    ],
);
}
}
```

## 5. count\_info 만들기

```
import 'package:flutter/material.dart';
import 'package:profile_app/theme.dart';
import 'components/profile_drawer.dart';
import 'components/profile_header.dart';

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

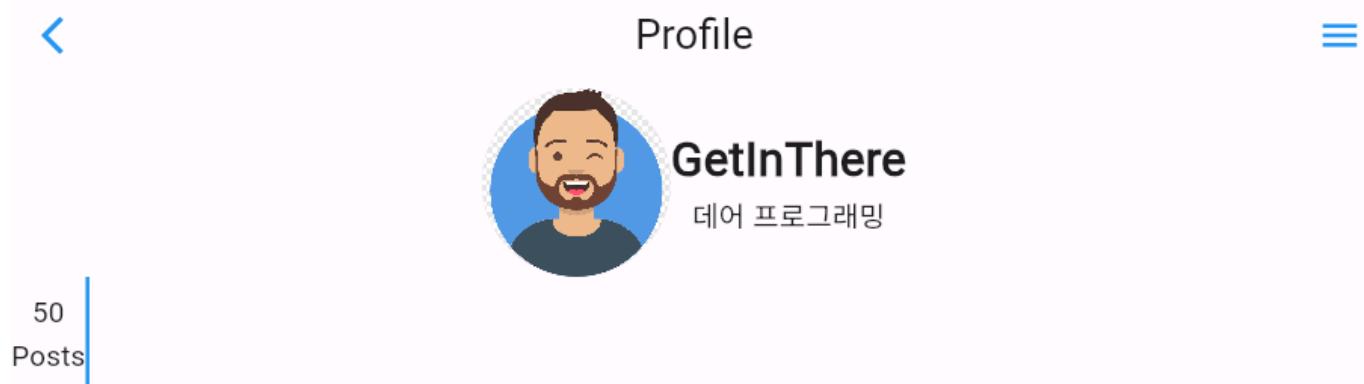
class MyApp extends StatelessWidget {
    const MyApp({super.key});

    @override
    Widget build(BuildContext context) {
        return MaterialApp(
            debugShowCheckedModeBanner: false,
            theme: theme(),
            home: Scaffold(
                appBar: _buildAppBar(),
                endDrawer: ProfileDrawer(),
                body: Column(
                    children: [
                        ProfileHeader(),
                        Row(
                            children: [
                                Column(
                                    children: [
                                        Text(
                                            "50",
                                            style: TextStyle(fontSize: 15),
                                        ),
                                        Text(
                                            "Posts",
                                            style: TextStyle(fontSize: 15),
                                        ),
                                    ],
                                ),
                            ],
                        ),
                        Container(

```

```
        width: 2, height: 60, color: Colors.blue,
    ),
],
),
],
),
),
);
}

AppBar _buildAppBar() {
    return AppBar(
        leading: Icon(Icons.arrow_back_ios),
        title: Text("Profile"),
        centerTitle: true,
    );
}
}
```



## 5.2. 컴포넌트 분리, 변수화

### components/count\_info.dart

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

class CountInfo extends StatelessWidget {
    final countNumber ;
    final countMenu ;

    CountInfo({required this.countNumber, required this.countMenu});

    void incrementCount() {
        countNumber++;
    }

    void showCountMenu() {
        countMenu();
    }
}
```

```
@override
Widget build(BuildContext context) {
  return Column(
    children: [
      Text(
        countNumber,
        style: TextStyle(fontSize: 15),
      ),
      SizedBox(height: 2),
      Text(
        countMenu,
        style: TextStyle(fontSize: 15),
      ),
    ],
  );
}
```

## components/count\_line.dart

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

class CountLine extends StatelessWidget {
  const CountLine({
    super.key,
  });

  @override
  Widget build(BuildContext context) {
    return Container(
      width: 2, height: 60, color: Colors.blue,
    );
  }
}
```

## profile\_count\_info.dart

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

import 'count_info.dart';
import 'count_line.dart';

class ProfileCountInfo extends StatelessWidget {
  const ProfileCountInfo({
    super.key,
  });
}
```

```
@override
Widget build(BuildContext context) {
  return Padding(
    padding: const EdgeInsets.only(top: 20),
    child: Row(
      mainAxisAlignment: MainAxisAlignment.spaceEvenly,
      children: [
        CountInfo(countNumber: "50", countMenu: "Posts",),
        CountLine(),
        CountInfo(countNumber: "10", countMenu: "Likes",),
        CountLine(),
        CountInfo(countNumber: "3", countMenu: "Share",),
      ],
    ),
  );
}
```

## main.dart

```
import 'package:flutter/material.dart';
import 'package:profile_app/theme.dart';
import 'components/count_info.dart';
import 'components/count_line.dart';
import 'components/profile_count_info.dart';
import 'components/profile_drawer.dart';
import 'components/profile_header.dart';

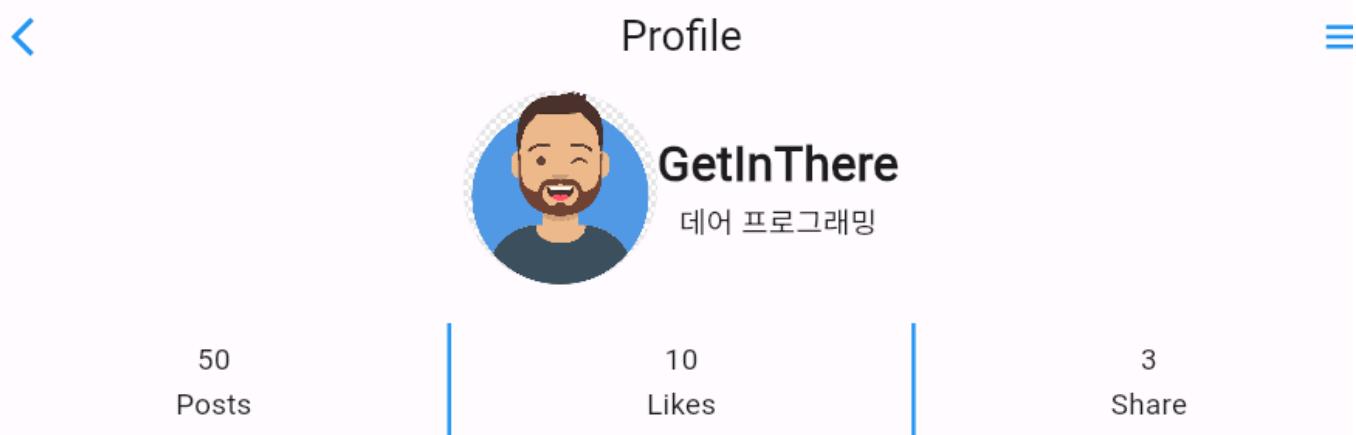
void main() {
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: theme(),
      home: Scaffold(
        appBar: _buildAppBar(),
        endDrawer: ProfileDrawer(),
        body: Column(
          children: [
            ProfileHeader(),
            ProfileCountInfo(),
          ],
        ),
      ),
    );
}
```

```
    );
}

AppBar _buildAppBar() {
    return AppBar(
        leading: Icon(Icons.arrow_back_ios),
        title: Text("Profile"),
        centerTitle: true,
    );
}
}
```



## 6. InkWell 사용하기

### 6.1. InkWell 란?

💡 `InkWell` 위젯은 사용자의 터치에 반응하여 물결 효과를 생성하고, 탭 이벤트를 처리할 수 있어서 버튼처럼 사용할 수 있다.

```
import 'package:flutter/material.dart';
import 'package:profile_app/theme.dart';
import 'components/profile_count_info.dart';
import 'components/profile_drawer.dart';
```

```
import 'components/profile_header.dart';

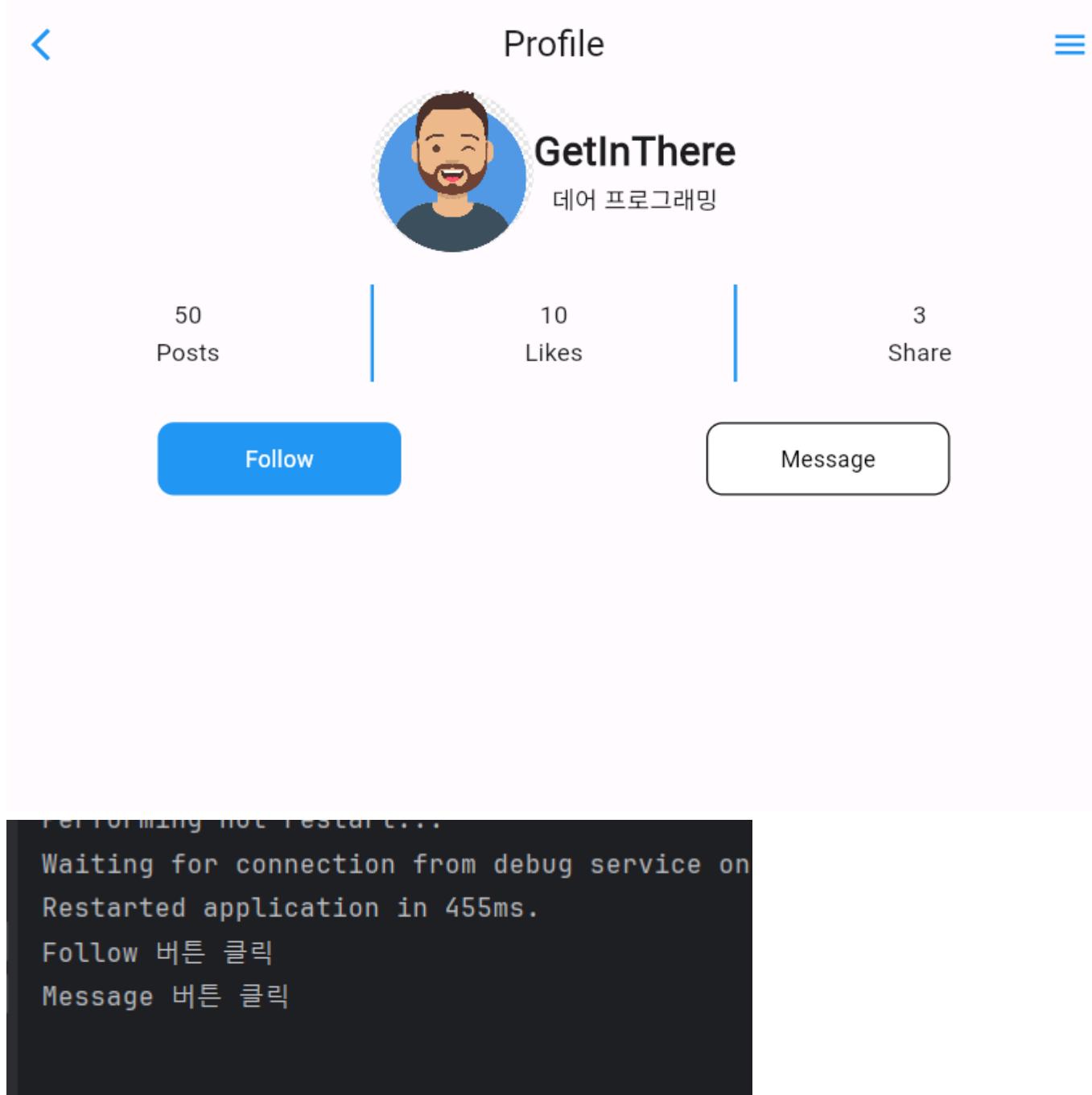
void main() {
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: theme(),
      home: Scaffold(
        appBar: _buildAppBar(),
        endDrawer: ProfileDrawer(),
        body: Column(
          children: [
            ProfileHeader(),
            ProfileCountInfo(),
            Padding(
              padding: const EdgeInsets.only(top: 25),
              child: Row(
                mainAxisAlignment: MainAxisAlignment.spaceAround,
                children: [
                  InkWell(
                    onTap: () { // 버튼을 눌렀을 때 이벤트 실행
                      print("Follow 버튼 클릭");
                    },
                    child: Container(
                      alignment: Alignment.center,
                      width: 150,
                      height: 45,
                      child: Text(
                        "Follow",
                        style: TextStyle(color: Colors.white),
                      ),
                      decoration: BoxDecoration(
                        color: Colors.blue,
                        borderRadius: BorderRadius.circular(10),
                      ),
                    ),
                  ),
                  InkWell(
                    onTap: () {
                      print("Message 버튼 클릭");
                    },
                    child: Container(
                      alignment: Alignment.center,
                      width: 150,
                      height: 45,
                      child: Text(
                        "Message",
                      ),
                    ),
                  ),
                ],
              ),
            ),
          ],
        ),
      ),
    );
  }
}
```

```
        style: TextStyle(color: Colors.black, ),
    ),
    decoration: BoxDecoration(
        color: Colors.white,
        borderRadius: BorderRadius.circular(10),
        border: Border.all(),
    ),
),
),
),
),
],
),
),
],
),
),
),
);
}
}

AppBar _buildAppBar() {
    return AppBar(
        leading: Icon(Icons.arrow_back_ios),
        title: Text("Profile"),
        centerTitle: true,
    );
}
}
```



Inkwell 버튼을 누르면 이벤트가 발생한다.

## 6.2. 컴포넌트 분리

**components/profile\_buttons.dart**

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

class ProfileButtons extends StatelessWidget {
  const ProfileButtons({
    super.key,
  });

  @override
  Widget build(BuildContext context) {
```

```
return Padding(
  padding: const EdgeInsets.only(top: 25),
  child: Row(
    mainAxisAlignment: MainAxisAlignment.spaceAround,
    children: [
      InkWell(
        onTap: () {
          print("Follow 버튼 클릭");
        },
        child: Container(
          alignment: Alignment.center,
          width: 150,
          height: 45,
          child: Text(
            "Follow",
            style: TextStyle(color: Colors.white),
          ),
          decoration: BoxDecoration(
            color: Colors.blue,
            borderRadius: BorderRadius.circular(10),
          ),
        ),
      ),
      InkWell(
        onTap: () {
          print("Message 버튼 클릭");
        },
        child: Container(
          alignment: Alignment.center,
          width: 150,
          height: 45,
          child: Text(
            "Message",
            style: TextStyle(
              color: Colors.black,
            ),
          ),
          decoration: BoxDecoration(
            color: Colors.white,
            borderRadius: BorderRadius.circular(10),
            border: Border.all(),
          ),
        ),
      ),
    ],
  );
}
```

## main.dart

```
import 'package:flutter/material.dart';
import 'package:profile_app/theme.dart';
import 'components/profile_buttons.dart';
import 'components/profile_count_info.dart';
import 'components/profile_drawer.dart';
import 'components/profile_header.dart';

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

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: theme(),
      home: Scaffold(
        appBar: _buildAppBar(),
        endDrawer: ProfileDrawer(),
        body: Column(
          children: [
            ProfileHeader(),
            ProfileCountInfo(),
            ProfileButtons(),
          ],
        ),
      ),
    );
  }

  AppBar _buildAppBar() {
    return AppBar(
      leading: Icon(Icons.arrow_back_ios),
      title: Text("Profile"),
      centerTitle: true,
    );
  }
}
```

## 7. DefaultTabController 사용하기

💡 `DefaultTabController`는 탭을 사용하여 여러 페이지를 구성할 때 사용되는 위젯이다.  
`DefaultTabController`를 사용하면 Flutter에서 탭 기반의 인터페이스를 쉽게 구현할 수 있다.

### main.dart

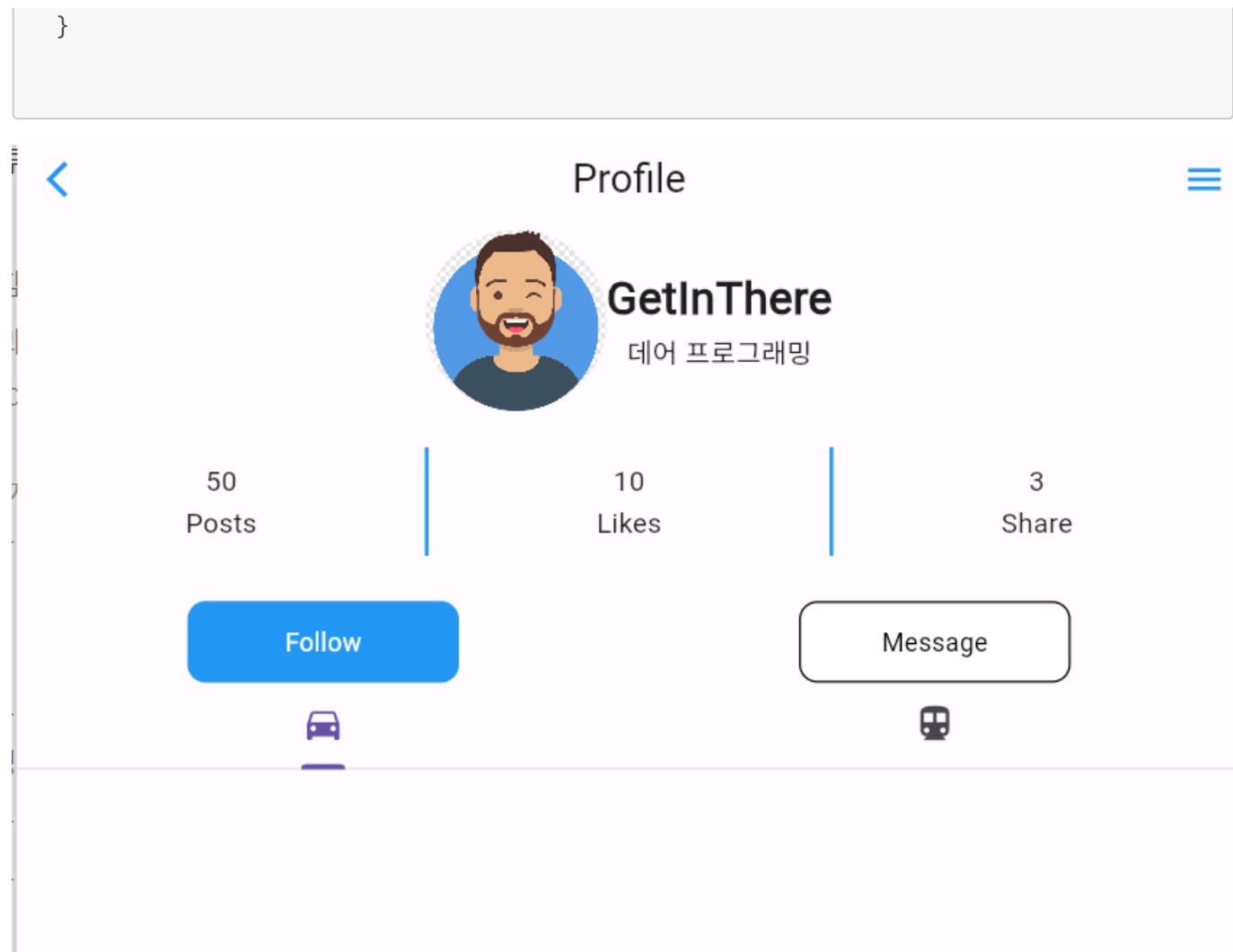
```
import 'package:flutter/material.dart';
import 'package:profile_app/theme.dart';
import 'components/profile_buttons.dart';
import 'components/profile_count_info.dart';
import 'components/profile_drawer.dart';
import 'components/profile_header.dart';

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

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: theme(),
      home: Scaffold(
        appBar: _buildAppBar(),
        endDrawer: ProfileDrawer(),
        body: Column(
          children: [
            ProfileHeader(),
            ProfileCountInfo(),
            ProfileButtons(),
            DefaultTabController(
              length: 2,
              initialIndex: 0,
              child: Column(
                children: [
                  TabBar(tabs: [
                    Tab(icon: Icon(Icons.directions_car)),
                    Tab(icon: Icon(Icons.directions_transit)),
                  ],),
                ],
              ),
            ),
            ],
          );
    );
  }

  AppBar _buildAppBar() {
    return AppBar(
      leading: Icon(Icons.arrow_back_ios),
      title: Text("Profile"),
      centerTitle: true,
    );
  }
}
```



## 8. TabBarView 사용하기, 높이 오류 해결

### 8.1. TabBarView 란?

💡 `TabBarView`는 탭을 통해 여러 페이지를 스와이프하여 볼 수 있게 해주는 위젯이다. 주로 `TabBar`와 함께 사용되며, 사용자가 탭을 선택할 때 해당하는 페이지를 보여준다.

```
import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import 'package:flutter/widgets.dart';
import 'package:profile_app/theme.dart';
import 'components/profile_buttons.dart';
import 'components/profile_count_info.dart';
import 'components/profile_drawer.dart';
import 'components/profile_header.dart';

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

class MyApp extends StatelessWidget {
  const MyApp({super.key});
```

```
@override
Widget build(BuildContext context) {
    return MaterialApp(
        debugShowCheckedModeBanner: false,
        theme: theme(),
        home: Scaffold(
            appBar: _buildAppBar(),
            endDrawer: ProfileDrawer(),
            body: Column(
                children: [
                    ProfileHeader(),
                    ProfileCountInfo(),
                    ProfileButtons(),
                    DefaultTabController(
                        length: 2,
                        initialIndex: 0,
                        child: Column(
                            children: [
                                TabBar(
                                    tabs: [
                                        Tab(icon: Icon(Icons.directions_car)),
                                        Tab(icon: Icon(Icons.directions_transit)),
                                    ],
                                ),
                                TabBarView(
                                    children: [
                                        GridView.builder(
                                            gridDelegate:
                                                SliverGridDelegateWithFixedCrossAxisCount(
                                                    crossAxisSpacing: 10,
                                                    crossAxisCount: 3,
                                                    mainAxisSpacing: 10,
                                                ),
                                            itemCount: 42,
                                            itemBuilder: (context, index) {
                                                return Image.network(
                                                    "https://picsum.photos/id/${index + 1}/200/200");
                                            },
                                        ),
                                        Container(
                                            color: Colors.red,
                                        )
                                    ],
                                ),
                            ],
                        ),
                    );
    });
}

AppBar _buildAppBar() {
```

```

        return AppBar(
            leading: Icon(Icons.arrow_back_ios),
            title: Text("Profile"),
            centerTitle: true,
        );
    }
}

```

## 8.2. Horizontal viewport was given unbounded height 오류

💡 해당 코드를 실행하면 위와 같은 오류가 발생한다. 이는 부모 Column 내부에 무제한의 높이를 가질 수 있는 GridView나 ListView, TabBarView 등이 있을 때 발생한다. 이때 'Expanded' 위젯으로 자식 위젯을 감싼다. 'Expanded' 위젯은 가능한 모든 공간을 차지하도록 자식 위젯을 확장시킨다.

```

        ), // TabBar
        Expanded(
            child: TabBarView(
                children: [
                    GridView.builder(
                        gridDelegate:

```

```

The following assertion was thrown during performLayout():
RenderFlex children have non-zero flex but incoming height constraints are unbounded.

When a column is in a parent that does not provide a finite height constraint, for example
the vertical axis. Setting a flex on a child (e.g. using Expanded) indicates that the child

```

## 8.3. RenderFlex children have non-zero flex but incoming height constraints are unbounded 오류

💡 이렇게 되면 또 다른 오류가 발생한다. 이 오류는 'Column'과 같은 위젯이 자식에게 무한한 높이를 허용할 때, 그 안에 있는 'Expanded' 또는 'Flexible' 위젯이 얼마나 많은 공간을 차지해야 하는지 결정할 수 없을 때 발생한다.

Column 내부에서 Expanded 또는 Flexible 위젯을 사용할 때, 이들이 명확한 높이 제약을 가질 수 있도록 해야 한다. 예를 들어, Column을 Scaffold의 body로 직접 사용하는 경우, Column의 높이가 화면의 높이로 제한되므로 Expanded 위젯을 사용해야 한다.

```

        ProfileButtons(),
        Expanded(
            child: DefaultTabController(
                length: 2,
                initialIndex: 0,
                child: Column(

```

## main.dart

```
import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import 'package:flutter/widgets.dart';
import 'package:profile_app/theme.dart';
import 'components/profile_buttons.dart';
import 'components/profile_count_info.dart';
import 'components/profile_drawer.dart';
import 'components/profile_header.dart';

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

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: theme(),
      home: Scaffold(
        appBar: _buildAppBar(),
        endDrawer: ProfileDrawer(),
        body: Column(
          children: [
            ProfileHeader(),
            ProfileCountInfo(),
            ProfileButtons(),
            Expanded(
              child: DefaultTabController(
                length: 2,
                initialIndex: 0,
                child: Column(
                  children: [
                    TabBar(
                      tabs: [
                        Tab(icon: Icon(Icons.directions_car)),
                        Tab(icon: Icon(Icons.directions_transit)),
                      ],
                    ),
                    Expanded(
                      child: TabBarView(
                        children: [
                          GridView.builder(
                            gridDelegate:
                              SliverGridDelegateWithFixedCrossAxisCount(
                                crossAxisSpacing: 10,
                                crossAxisCount: 3,
                                mainAxisSpacing: 10,
                          ),
                        ],
                      ),
                    ),
                  ],
                ),
              ),
            ),
          ],
        ),
      ),
    );
  }
}
```



Profile

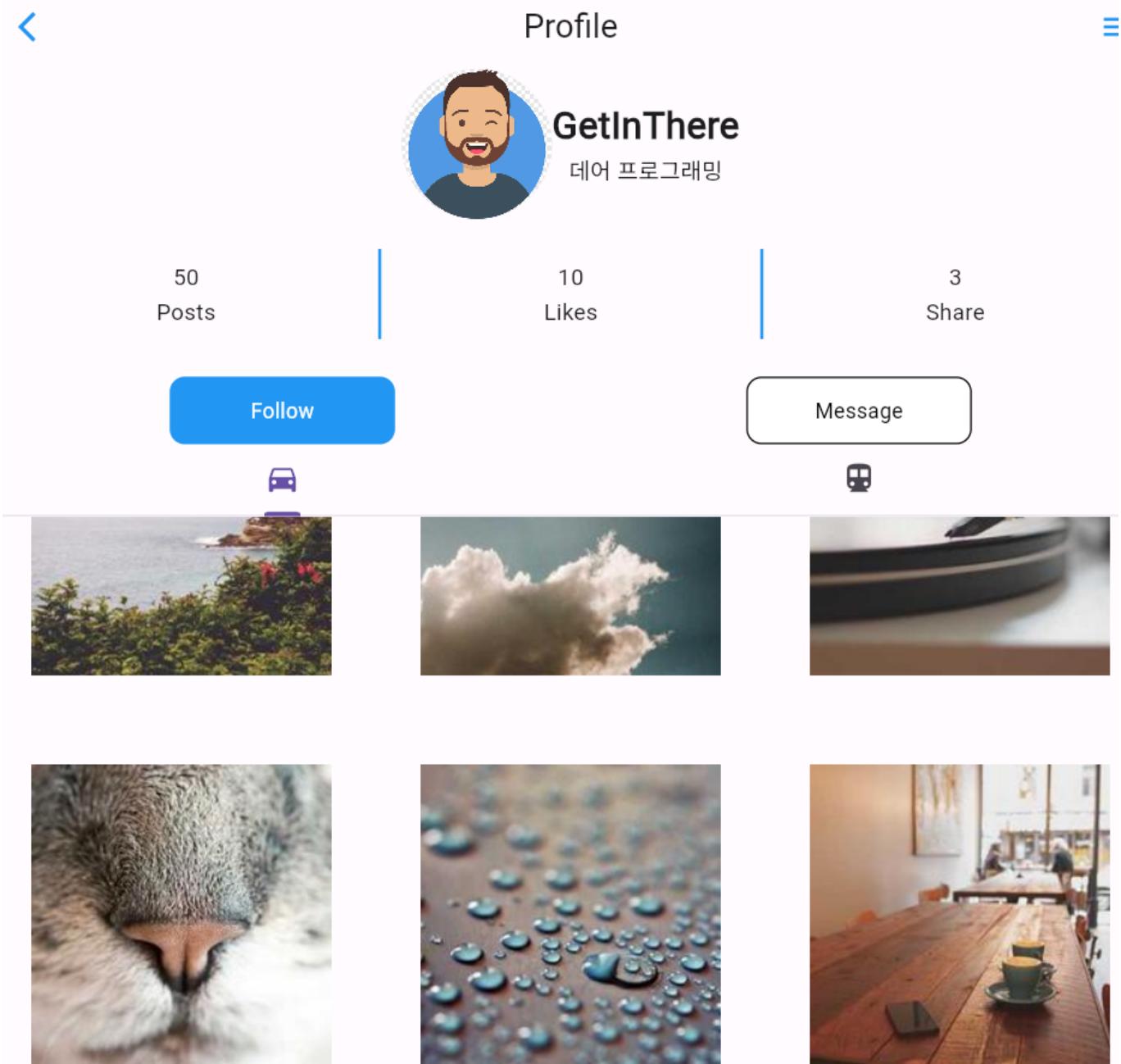


GetInThere  
데어 프로그래밍

50 Posts      10 Likes      3 Share

Follow      Message





스크롤을 내리면 사진을 볼 수 있다.

## 9. 컴포넌트 분리로 코드 정리하기

**components/profile\_tab.dart**

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

class ProfileTab extends StatelessWidget {
  const ProfileTab({
    super.key,
  });

  @override
  Widget build(BuildContext context) {
    return DefaultTabController(
```

```
length: 2,
initialIndex: 0,
child: Column(
  children: [
    _TabBar(),
    Expanded(child: _TabBarView()),
  ],
),
);
}
}

class _TabBarView extends StatelessWidget {
const _TabBarView({
  super.key,
});

@Override
Widget build(BuildContext context) {
  return TabBarView(
    children: [
      GridView.builder(
        gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(
          crossAxisSpacing: 10,
          crossAxisCount: 3,
          mainAxisSpacing: 10,
        ),
        itemCount: 42,
        itemBuilder: (context, index) {
          return Image.network(
            "https://picsum.photos/id/${index + 1}/200/200");
        },
      ),
      Container(
        color: Colors.red,
      )
    ],
);
}

class _TabBar extends StatelessWidget {
const _TabBar({
  super.key,
});

@Override
Widget build(BuildContext context) {
  return TabBar(
    tabs: [
      Tab(icon: Icon(Icons.directions_car)),
      Tab(icon: Icon(Icons.directions_transit)),
    ],
);
}
```

```
    }
}
```

## pages/profile\_page.dart

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

import '../components/profile_buttons.dart';
import '../components/profile_count_info.dart';
import '../components/profile_drawer.dart';
import '../components/profile_header.dart';
import '../components/profile_tab.dart';

class ProfilePage extends StatelessWidget {
  const ProfilePage({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: _buildAppBar(),
      endDrawer: ProfileDrawer(),
      body: Column(
        children: [
          ProfileHeader(),
          ProfileCountInfo(),
          ProfileButtons(),
          Expanded(child: ProfileTab())
        ],
      ),
    );
  }

  AppBar _buildAppBar() {
    return AppBar(
      leading: Icon(Icons.arrow_back_ios),
      title: Text("Profile"),
      centerTitle: true,
    );
  }
}
```

## main.dart

```
import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import 'package:flutter/widgets.dart';
import 'package:profile_app/pages/profile_page.dart';
```

```
import 'package:profile_app/theme.dart';

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

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: theme(),
      home: ProfilePage(),
    );
  }
}
```