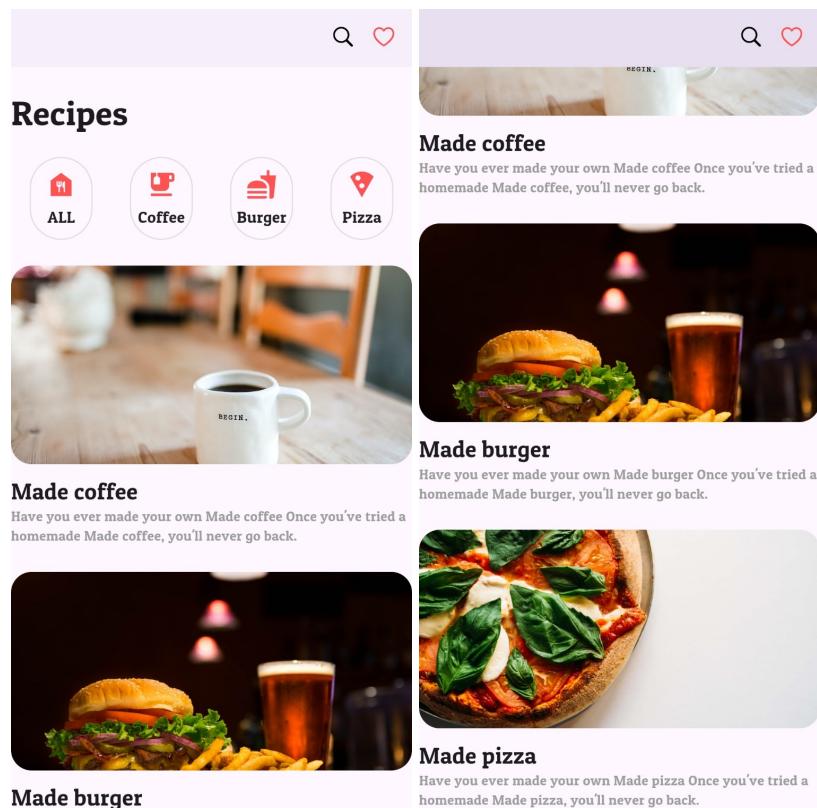
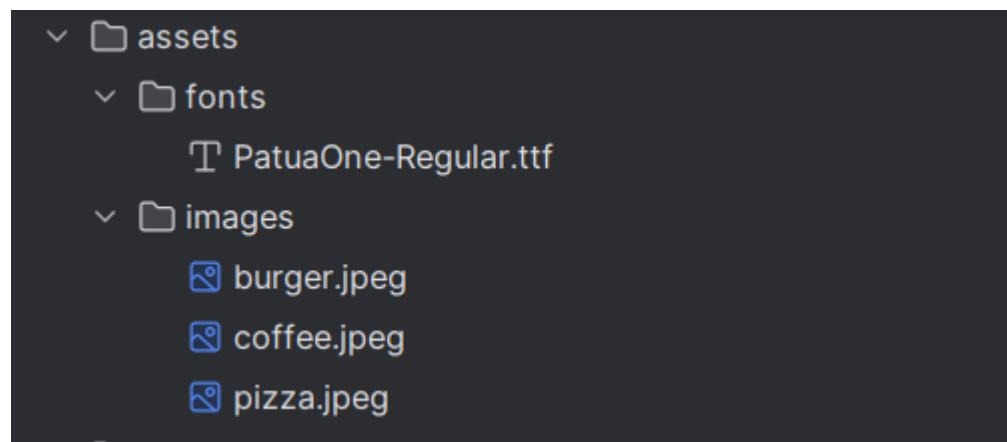


Chapter 05 레시피 앱 만들기

완성 화면



1. 기본 세팅



assets에 폰트와 사진을 넣는다.

pubspec.yaml 설정

```

# To add assets to your application's assets folder, run:
#   $ cd android
#   $ ./gradlew assembleRelease
#   $ cd ..
#   $ pub get

# example:
fonts:
  - family: PatuaOne
    fonts:
      - asset: assets/fonts/PatuaOne-Regular.ttf
#      - asset: fonts/Schyler-Italic.ttf
#        style: italic

```

pubspec.yaml 설정 후 pub get 을 누른다.

2. main.dart

```

import 'package:flutter/material.dart';

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: ThemeData(fontFamily: "PatuaOne"), // 페이지의 폰트를 설정한다.
      home: Scaffold(),
    );
  }
}

```

💡 `ThemeData`는 Flutter 앱의 전반적인 테마를 설정하는 데 사용된다. 여기에는 글꼴, 색상, 버튼 스타일 등 앱의 시각적 요소를 정의하는 다양한 속성이 포함된다.

3. AppBar 만들기

3.1. AppBar 란?

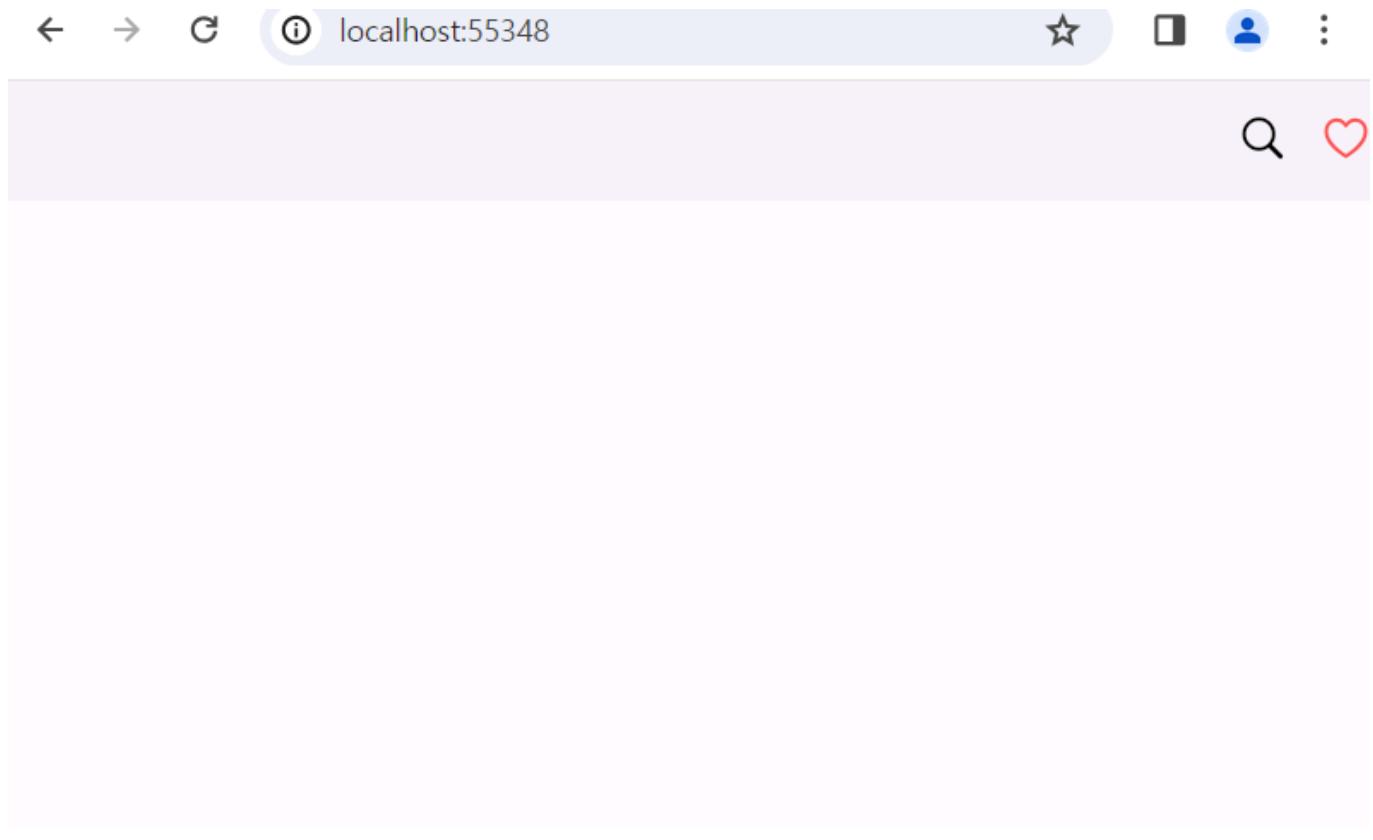
💡 `AppBar`는 앱의 상단에 위치하는 툴바로, 주로 앱의 제목을 표시하거나 메뉴 버튼, 액션 버튼 등을 포함하는 데 사용한다.

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

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: ThemeData(fontFamily: "PatuaOne"), // 페이지의 폰트를 설정한다.
      home: Scaffold(
        appBar: AppBar(
          elevation: 1.0,
          actions: [
            Icon(
              CupertinoIcons.search, // 돋보기 아이콘
              color: Colors.black,
            ),
            SizedBox(width: 15),
            Icon(
              CupertinoIcons.heart, // 하트 아이콘
              color: Colors.redAccent,
            ),
            SizedBox(width: 15),
          ],
        ),
      );
  }
}
```



3.2. 컴포넌트 분리하기

💡 가독성을 위해 컴포넌트를 분리한다.

main.dart

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

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: ThemeData(fontFamily: "PatuaOne"), // 페이지의 폰트를 설정한다.
      home: Scaffold(
        appBar: _buildAppBar(),
      ),
    );
  }

  AppBar _buildAppBar() {
    return AppBar(
      elevation: 1.0,
```

```
actions: [
    Icon(
        CupertinoIcons.search, // 둘보기 아이콘
        color: Colors.black,
    ),
    SizedBox(width: 15),
    Icon(
        CupertinoIcons.heart, // 하트 아이콘
        color: Colors.redAccent,
    ),
    SizedBox(width: 15),
],
);
}
}
```

💡 _ 는 클래스 내부에서만 사용할 수 있다.

4. ListView 사용하기

4.1. ListView 란?

💡 ListView는 스크롤 가능한 항목들의 리스트를 구현할 때 사용되는 기본적이고 중요한 위젯이다. 사용자 정의 데이터를 수직으로 배열하여 화면에 표시하는 데 사용되며, 쉬운 구현과 유연한 기능성을 제공한다.

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

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

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

    @override
    Widget build(BuildContext context) {
        return MaterialApp(
            debugShowCheckedModeBanner: false,
            theme: ThemeData(fontFamily: "PatuaOne"), // 페이지의 폰트를 설정한다.
            home: Scaffold(
                appBar: _buildAppBar(),
                body: ListView(
                    children: [
                        Padding(
                            padding: const EdgeInsets.only(top: 20),
                            child: Text(
                                "Recipes",
                                style: TextStyle(fontSize: 30),
                            ),
                        ),
                    ],
                ),
            ),
        );
    }
}
```

```
        ),
      ],
    ),
  );
}

AppBar _buildAppBar() {
  return AppBar(
    elevation: 1.0,
    actions: [
      Icon(
        CupertinoIcons.search, // 둘보기 아이콘
        color: Colors.black,
      ),
      SizedBox(width: 15),
      Icon(
        CupertinoIcons.heart, // 하트 아이콘
        color: Colors.redAccent,
      ),
      SizedBox(width: 15),
    ],
  );
}
}
```



Recipes

4.2. 컴포넌트 분리하기

main.dart

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

import 'components/recipe_title.dart';

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: ThemeData(fontFamily: "PatuaOne"), // 페이지의 폰트를 설정한다.
      home: Scaffold(
        appBar: _buildAppBar(),
        body: ListView(
          children: [
            RecipeTitle(),
          ],
        ),
      ),
    );
  }

  AppBar _buildAppBar() {
    return AppBar(
      elevation: 1.0,
      actions: [
        Icon(
          CupertinoIcons.search, // 둘보기 아이콘
          color: Colors.black,
        ),
        SizedBox(width: 15),
        Icon(
          CupertinoIcons.heart, // 하트 아이콘
          color: Colors.redAccent,
        ),
        SizedBox(width: 15),
      ],
    );
  }
}
```

components/recipe_title.dart

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

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

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: const EdgeInsets.only(top: 20),
      child: Text(
        "Recipes",
        style: TextStyle(fontSize: 30),
      ),
    );
  }
}
```

5. Container 사용하기

5.1. Container 란?

💡 Container는 가장 기본적이면서도 유용한 레이아웃 위젯 중 하나이다. 단 하나의 자식 위젯을 포함할 수 있으며, 크기 지정, 색상, 패딩, 마진 등 다양한 스타일링 옵션을 가진다. Container는 위젯의 배치, 크기 조정, 데코레이션(배경색, 테두리 등)을 쉽게 할 수 있게 해주는 역할을 한다.

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

import 'components/recipe_title.dart';

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

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

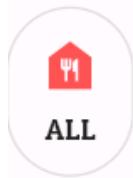
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: ThemeData(fontFamily: "PatuaOne"), // 페이지의 폰트를 설정한다.
      home: Scaffold(
        appBar: _buildAppBar(),
        body: ListView( // 스크롤을 만들 때 사용
          children: [
            RecipeTitle(), // 컨퍼넌트 분리
          ],
        ),
      ),
    );
  }
}
```

```
Row( // 가로 정렬할 때 사용
    children: [
        Container(
            width: 60,
            height: 80,
            decoration: BoxDecoration( // 컨테이너의 배경을 바꿀 때 사용
                borderRadius: BorderRadius.circular(30),
                border: Border.all(color: Colors.black12),
            ),
            child: Column(
                mainAxisAlignment: MainAxisAlignment.center,
                children: [
                    Icon(Icons.food_bank, color: Colors.redAccent, size: 30),
                    SizedBox(height: 5),
                    Text(
                        "ALL",
                        style: TextStyle(color: Colors.black87),
                    )
                ],
            ),
        ),
    ],
),
),
);
}

AppBar _buildAppBar() {
return AppBar(
elevation: 1.0,
actions: [
Icon(
    CupertinoIcons.search, // 둘보기 아이콘
    color: Colors.black,
),
SizedBox(width: 15),
Icon(
    CupertinoIcons.heart, // 하트 아이콘
    color: Colors.redAccent,
),
SizedBox(width: 15),
],
);
}
}
```



Recipes



5.2. 컴포넌트 분리 후 값 변수로 만들기

main.dart

```
@override
Widget build(BuildContext context) {
    return MaterialApp(
        debugShowCheckedModeBanner: false,
        theme: ThemeData(fontFamily: "PatuaOne"), // 페이지의 폰트를 설정한다.
        home: Scaffold(
            appBar: _buildAppBar(),
            body: ListView(
                children: [
                    RecipeTitle(),
                    recipe_menu() // 컴포넌트 분리
                ],
            ),
        ),
    );
}
```

components/recipe_menu.dart

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

class recipe_menu extends StatelessWidget {
    const recipe_menu({
        super.key,
```

```
});  
  
@override  
Widget build(BuildContext context) {  
    return Row(  
        children: [  
            Container(  
                width: 60,  
                height: 80,  
                decoration: BoxDecoration(  
                    borderRadius: BorderRadius.circular(30),  
                    border: Border.all(color: Colors.black12),  
                ),  
            child: Column(  
                mainAxisAlignment: MainAxisAlignment.center,  
                children: [  
                    Icon(Icons.food_bank, color: Colors.redAccent, size: 30),  
                    SizedBox(height: 5),  
                    Text(  
                        "ALL",  
                        style: TextStyle(color: Colors.black87),  
                    ),  
                ],  
            ),  
        ],  
    );  
}  
}  
}
```

컴포넌트를 분리한다.

현재 변경될 데이터는 Icons.food_bank, 와 "ALL", 이다. 이 두 데이터를 변수로 만들자.

recipe_menu_item.dart

```
import 'package:flutter/material.dart';  
  
class RecipeMenuItem extends StatelessWidget {  
    IconData mIcon ; //아이콘과 text 를 변수로 분리한다.  
    String text ;  
  
    recipe_menu_item({required this.mIcon,required this.text});  
  
    @override  
    Widget build(BuildContext context) {  
        return Row(  
            children: [  
                Container(  
                    width: 60,
```

```
        height: 80,
        decoration: BoxDecoration(
            borderRadius: BorderRadius.circular(30),
            border: Border.all(color: Colors.black12),
        ),
        child: Column(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
                Icon(mIcon, color: Colors.redAccent, size: 30),
                SizedBox(height: 5),
                Text(
                    text,
                    style: TextStyle(color: Colors.black87),
                )
            ],
        ),
    ],
),
],
);
}
}
```

💡 생성자에 클래스명({})의 형태를 선택적 생성자라고 한다. 선택적 생성자를 만들면 데이터를 받을 때 키 : 값의 형태로 받을 수 있다. 반드시 받아야 할 데이터는 required 를 사용한다.

main.dart

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

import 'components/recipe_menu_item.dart';
import 'components/recipe_title.dart';

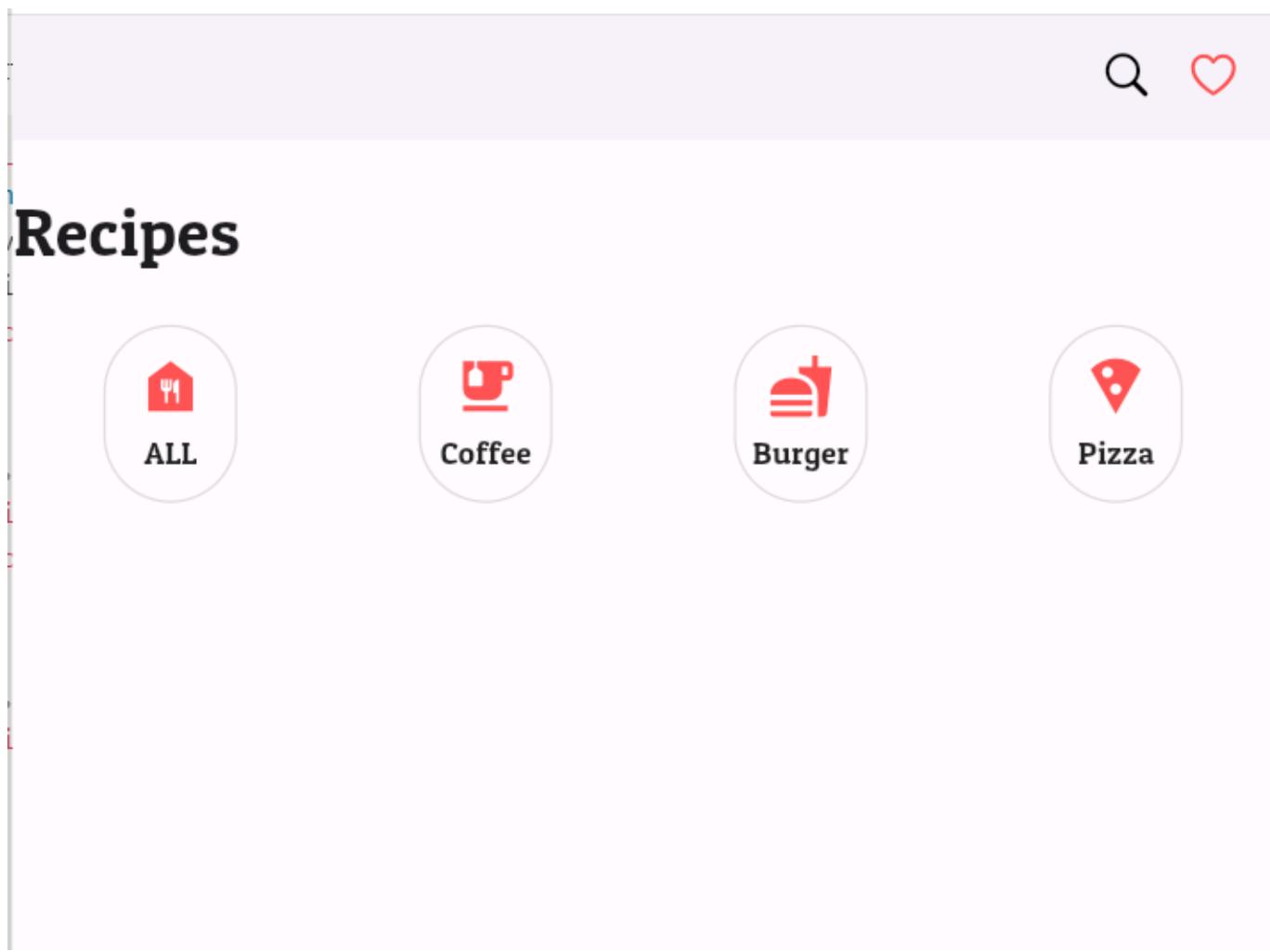
void main() {
    runApp(const MyApp());
}

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

    @override
    Widget build(BuildContext context) {
        return MaterialApp(
            debugShowCheckedModeBanner: false,
            theme: ThemeData(fontFamily: "PatuaOne"), // 페이지의 폰트를 설정한다.
            home: Scaffold(
                appBar: _buildAppBar(),
                body: ListView(
                    children: [
                        RecipeTitle(),

```

```
Padding(  
    padding: const EdgeInsets.only(top: 20),  
    child: Row(  
        mainAxisAlignment: MainAxisAlignment.spaceAround, // Row 는 가로가  
메인축, 세로가 크로스축  
        children: [  
            recipe_menu_item(mIcon: Icons.food_bank, text: "ALL"),  
            recipe_menu_item(  
                mIcon: Icons.emoji_food_beverage, text: "Coffee"),  
            recipe_menu_item(mIcon: Icons.fastfood, text: "Burger"),  
            recipe_menu_item(mIcon: Icons.local_pizza, text: "Pizza"),  
        ],  
    ),  
),  
],  
),  
),  
);  
};  
  
AppBar _buildAppBar() {  
    return AppBar(  
        elevation: 1.0,  
        actions: [  
            Icon(  
                CupertinoIcons.search, // 둘보기 아이콘  
                color: Colors.black,  
            ),  
            SizedBox(width: 15),  
            Icon(  
                CupertinoIcons.heart, // 하트 아이콘  
                color: Colors.redAccent,  
            ),  
            SizedBox(width: 15),  
        ],  
    );  
}  
}
```



💡 컴포넌트를 분리하고, 변수를 만들어 사용하면 함수만 호출하면 재사용할 수 있다.



6. AspectRatio 사용하기

6.1. AspectRatio 란?

💡 `AspectRatio` 위젯은 자식 위젯이 특정 종횡비(aspect ratio)를 유지하도록 설정하는 데 사용된다. 종횡비는 너비 대 높이의 비율로 표현되며, 이를 통해 위젯의 크기를 조절할 수 있다.

aspectRatio : 2/1 : 가로 :2 = 세로 :1 를 의미한다.

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

import 'components/recipe_menu_item.dart';
import 'components/recipe_title.dart';

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

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

```
@override
Widget build(BuildContext context) {
    return MaterialApp(
        debugShowCheckedModeBanner: false,
        theme: ThemeData(fontFamily: "PatuaOne"), // 페이지의 폰트를 설정한다.
        home: Scaffold(
            appBar: _buildAppBar(),
            body: ListView(
                children: [
                    RecipeTitle(),
                    Padding(
                        padding: const EdgeInsets.only(top: 20),
                        child: Row(
                            mainAxisAlignment: MainAxisAlignment.spaceAround,
                            // Row 는 가로가 메인축, 세로가 크로스축
                            children: [
                                recipe_menu_item(mIcon: Icons.food_bank, text: "ALL"),
                                recipe_menu_item(
                                    mIcon: Icons.emoji_food_beverage, text: "Coffee"),
                                recipe_menu_item(mIcon: Icons.fastfood, text: "Burger"),
                                recipe_menu_item(mIcon: Icons.local_pizza, text: "Pizza"),
                            ],
                        ),
                    ),
                    Column(
                        children: [
                            AspectRatio(
                                aspectRatio: 2 / 1, // 사진의 비율을 가로 2 세로 1로 한다.
                                child: ClipRRect(
                                    borderRadius: BorderRadius.circular(20),
                                    child: Image.asset(
                                        "assets/images/coffee.jpeg",
                                        fit: BoxFit.cover,
                                    ),
                                ),
                            ),
                            SizedBox(height: 10),
                            Text(
                                "Coffee",
                                style: TextStyle(fontSize: 20),
                            ),
                            Text(
                                "Have you ever made your own Coffee Once you've tried a homemade Coffee, you'll never go back.",
                                style: TextStyle(color: Colors.grey, fontSize: 12),
                            )
                        ],
                    ),
                ],
            );
        );
    }
}
```

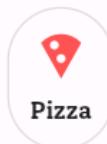
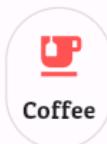
```
AppBar _buildAppBar() {
    return AppBar(
        elevation: 1.0,
        actions: [
            Icon(
                CupertinoIcons.search, // 둘보기 아이콘
                color: Colors.black,
            ),
            SizedBox(width: 15),
            Icon(
                CupertinoIcons.heart, // 하트 아이콘
                color: Colors.redAccent,
            ),
            SizedBox(width: 15),
        ],
    );
}
```

```
children: [
    recipe_menu_item(mIcon: Icons.food_bank, text: "ALL"),
    recipe_menu_item(
        mIcon: Icons.emoji_food_beverage, text: "Coffee"), // recipe_menu_item
    recipe_menu_item(mIcon: Icons.fastfood, text: "Burger"),
    recipe_menu_item(mIcon: Icons.local_pizza, text: "Pizza"),
],
), // Row
```

💡 Column 의 세로의 디폴트 값은 위쪽이지만, 가로는 센터가 디폴트 값이다. 그래서 크로스축을 start 로 정렬해야 한다.



Recipes



Coffee

Have you ever made your own Coffee Once you've tried a homemade Coffee, you'll never go back.

6.2. 컴포넌트 분리하고 변수 만들기

components/recipe_list_item.dart

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

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

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: const EdgeInsets.only(top: 25),
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
```

```

    children: [
      AspectRatio(
        aspectRatio: 2 / 1,
        child: ClipRRect(
          borderRadius: BorderRadius.circular(20),
          child: Image.asset(
            "assets/images/coffee.jpeg",
            fit: BoxFit.cover,
          ),
        ),
      ),
      SizedBox(height: 10),
      Text(
        "Coffee",
        style: TextStyle(fontSize: 20),
      ),
      Text(
        "Have you ever made your own Coffee Once you've tried a homemade Coffee, you'll never go back.",
        style: TextStyle(color: Colors.grey, fontSize: 12),
      )
    ],
  ),
);
}
}
}

```

```

class RecipeListItem extends StatelessWidget {
  final String imageName ;
  final String title ;

  RecipeListItem({required this.imageName,required this.title});


```

클래스에 변수와 생성자를 만든다.

```

    child: Image.asset(
      "assets/images/${imageName}.jpeg",
      fit: BoxFit.cover,
    ), // Image.asset
  ), // ClipRRect
), // AspectRatio
SizedBox(height: 10),
Text(
  title,
  style: TextStyle(fontSize: 20),
), // Text
Text(
  "Have you ever made your own ${title} Once you've tried a homemade ${title}, you'll never go back.",
  style: TextStyle(color: Colors.grey, fontSize: 12),
)

```

변수를 변경되어야 하는 곳에 대체한다.

main.dart

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

import 'components/recipe_list_item.dart';
import 'components/recipe_menu_item.dart';
import 'components/recipe_title.dart';

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

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

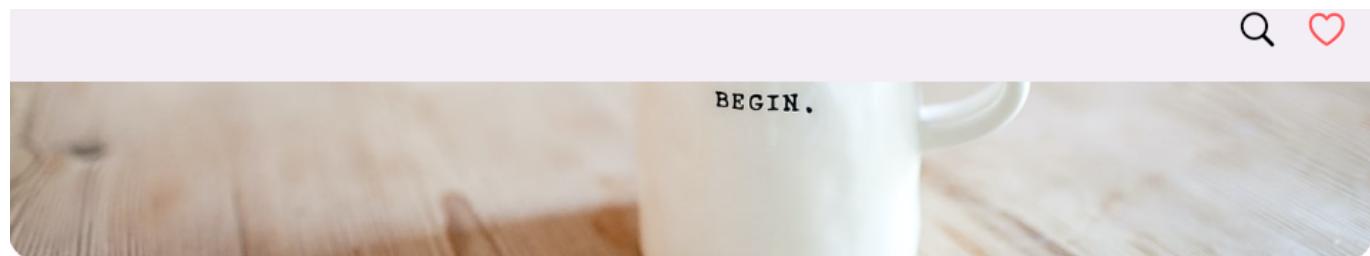
    @override
    Widget build(BuildContext context) {
        return MaterialApp(
            debugShowCheckedModeBanner: false,
            theme: ThemeData(fontFamily: "PatuaOne"), // 페이지의 폰트를 설정한다.
            home: Scaffold(
                appBar: _buildAppBar(),
                body: ListView(
                    children: [
                        RecipeTitle(),
                        Padding(
                            padding: const EdgeInsets.only(top: 20),
                            child: Row(
                                mainAxisAlignment: MainAxisAlignment.spaceAround,
                                // Row 는 가로가 메인축, 세로가 크로스축
                                children: [
                                    recipe_menu_item(mIcon: Icons.food_bank, text: "ALL"),
                                    recipe_menu_item(
                                        mIcon: Icons.emoji_food_beverage, text: "Coffee"),
                                    recipe_menu_item(mIcon: Icons.fastfood, text: "Burger"),
                                    recipe_menu_item(mIcon: Icons.local_pizza, text: "Pizza"),
                                ],
                            ),
                        ),
                        RecipeListItem(imageName: "coffee", title: "Made coffee"),
                        RecipeListItem(imageName: "burger", title: "Made burger"),
                        RecipeListItem(imageName: "pizza", title: "Made pizza"),
                    ],
                ),
            );
    }

    AppBar _buildAppBar() {
        return AppBar(
            elevation: 1.0,
            actions: [
                Icon(

```

```
CupertinoIcons.search, // 돋보기 아이콘  
color: Colors.black,  
,  
SizedBox(width: 15),  
Icon(  
CupertinoIcons.heart, // 하트 아이콘  
color: Colors.redAccent,  
,  
SizedBox(width: 15),  
],  
);  
}  
}
```

RecipeListItem 를 호출할 때 변수를 넣으면 재사용할 수 있다.



Made coffee

Have you ever made your own Made coffee Once you've tried a homemade Made coffee, you'll never go back.



Made burger

Have you ever made your own Made burger Once you've tried a homemade Made burger, you'll never go back.



7. 남은 컴포넌트 분리하기

main.dart

```
import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import 'package:recipe_app/pages/recipe_page.dart';

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: ThemeData(fontFamily: "PatuaOne"), // 페이지의 폰트를 설정한다.
      home: RecipePage(),
    );
  }
}
```

components/recipe_page.dart

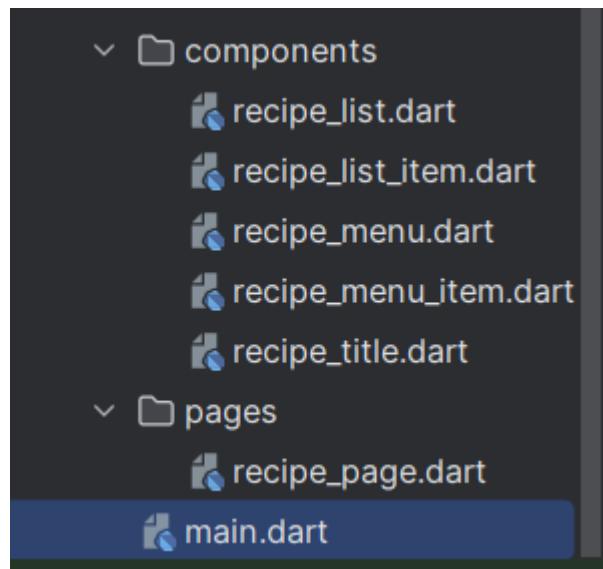
```
import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import '../components/recipe_list.dart';
import '../components/recipe_menu.dart';
import '../components/recipe_title.dart';

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

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: _buildAppBar(),
      body: ListView(
        children: [
          RecipeTitle(),
          RecipeMenu(),
          RecipeList(),
        ],
      ),
    );
  }

  AppBar _buildAppBar() {
```

```
return AppBar(  
    elevation: 1.0,  
    actions: [  
        Icon(  
            CupertinoIcons.search, // 둘보기 아이콘  
            color: Colors.black,  
        ),  
        SizedBox(width: 15),  
        Icon(  
            CupertinoIcons.heart, // 하트 아이콘  
            color: Colors.redAccent,  
        ),  
        SizedBox(width: 15),  
    ],  
);  
}  
}
```



💡 모든 기능을 분리함으로써 각 페이지는 원하는 기능만 호출하면 된다. 가독성과 유지보수가 매우 편리해졌다.