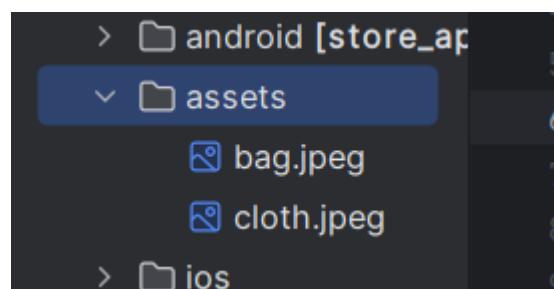


Chapter 04 스토어 앱 만들기

완성 화면



1. 기본 세팅



assets 폴더를 만든 후 사진을 넣는다.

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

pubspec.yaml에서 assets를 설정 후 pub get을 누른다.

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();
    }
}
```

💡 플러터에서 `MaterialApp()`은 Material Design 애플리케이션을 위한 편리한 위젯으로, 특정 기능들을 제공한다. 플러터 프로젝트를 새로 만들 때 기본적으로 `MaterialApp()`을 사용하여 UI를 생성한다. Container, Scaffold, Drawer, AppBar 등의 위젯을 함께 사용한다.

2. Scaffold

💡 `Scaffold`는 기본적인 Material Design의 시각적 레이아웃 구조를 구현하는 위젯이다. 앱의 기본적인 디자인 구조를 제공하며, 드로어(drawers), 바텀 시트(bottom sheets), 플로팅 액션 버튼(FloatingActionButton) 등 다양한 API를 제공한다. 이를 통해 기능적이고 반응형 앱을 쉽게 만들 수 있다.

- **앱 바(AppBar):** 상단에 앱 바를 추가하여 앱의 제목이나 액션 버튼을 표시할 수 있다.
- **바디(Body):** 앱의 주요 콘텐츠를 표시하는 영역. 다양한 위젯을 바디에 배치하여 앱의 주요 기능을 구현할 수 있다.
- **플로팅 액션 버튼(FloatingActionButton):** 화면에 고정된 액션 버튼을 추가하여 사용자가 중요한 작업을 쉽게 수행할 수 있도록 도와준다.

```
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(
      home : Scaffold(
        );
    );
  }
}
```

3. Column , Row

💡 `Column`과 `Row`는 레이아웃을 구성하는 핵심 위젯이다. `Column`은 자식 위젯들을 세로로 배열하고, `Row`는 자식 위젯들을 가로로 배열한다.

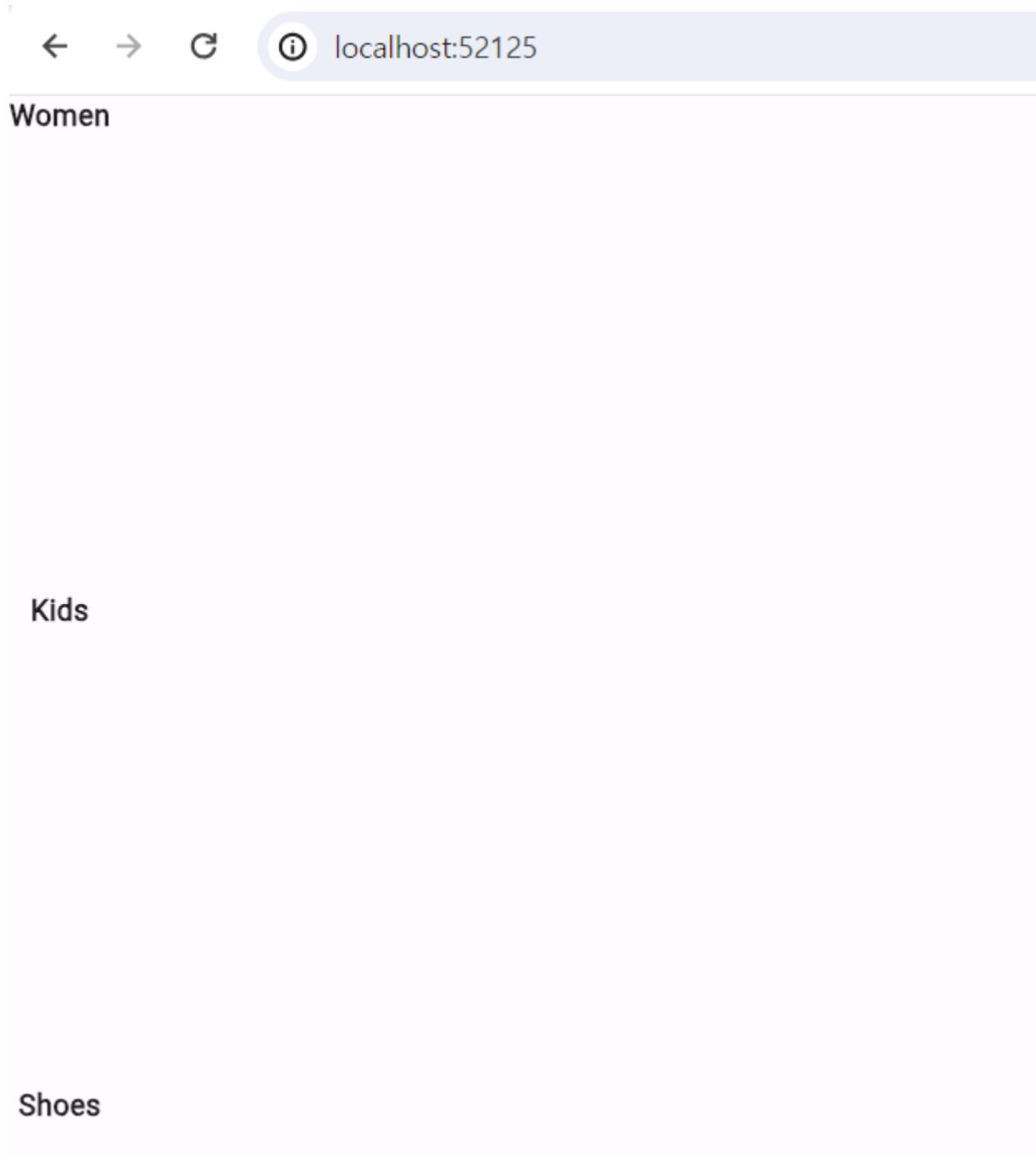
```
import 'package:flutter/material.dart';

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        body: SafeArea(
          child: Column(
            children: [
              Text("Women", style: TextStyle(fontWeight: FontWeight.bold)),
              Spacer(),
              Text("Kids", style: TextStyle(fontWeight: FontWeight.bold)),
              Spacer(),
              Text("Shoes", style: TextStyle(fontWeight: FontWeight.bold)),
              Spacer(),
              Text("Women", style: TextStyle(fontWeight: FontWeight.bold)),
            ],
          ),
        ),
      );
}
```

```
}
```



Column 으로 배치하면 세로로 배치가 된다.

가로 배치를 위해 Row 로 감싼다.

```
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(
        home: Scaffold(
            body: SafeArea(
                child: Column(
                    children: [
                        Padding( // 좌우위아래 간격을 25씩 준다.
                            padding: const EdgeInsets.all(25.0),
                            child: Row(
                                children: [
                                    Text("Women", style: TextStyle(fontWeight: FontWeight.bold)),
                                    Spacer(),
                                    Text("Kids", style: TextStyle(fontWeight: FontWeight.bold)),
                                    Spacer(),
                                    Text("Shoes", style: TextStyle(fontWeight: FontWeight.bold)),
                                    Spacer(),
                                    Text("Women", style: TextStyle(fontWeight: FontWeight.bold)),
                                ],
                            ),
                        ),
                    ],
                ),
            );
}
}

```



💡 앱의 전체 위젯은 세로 배치가 되지만 메뉴 부분은 Row 를 사용해 가로 배치를 한다.

4. 0|마지 넣기

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(
      home: Scaffold(
        body: SafeArea(
          child: Column(
            children: [
              Padding( // 좌우위아래 간격을 25씩 준다.
                padding: const EdgeInsets.all(25.0),
                child: Row(
                  children: [
                    Text("Women", style: TextStyle(fontWeight: FontWeight.bold)),
                    Spacer(),
                    Text("Kids", style: TextStyle(fontWeight: FontWeight.bold)),
                    Spacer(),
                    Text("Shoes", style: TextStyle(fontWeight: FontWeight.bold)),
                    Spacer(),
                    Text("Women", style: TextStyle(fontWeight: FontWeight.bold)),
                  ],
                ),
              ),
              Expanded(child: Image.asset("assets/bag.jpeg", fit: BoxFit.cover,)),
              SizedBox(height: 2), // 사이 간격을 위해 사용
              Expanded(child: Image.asset("assets/cloth.jpeg", fit:
                BoxFit.cover,)),
            ],
          ),
        ),
      );
    }
}
```

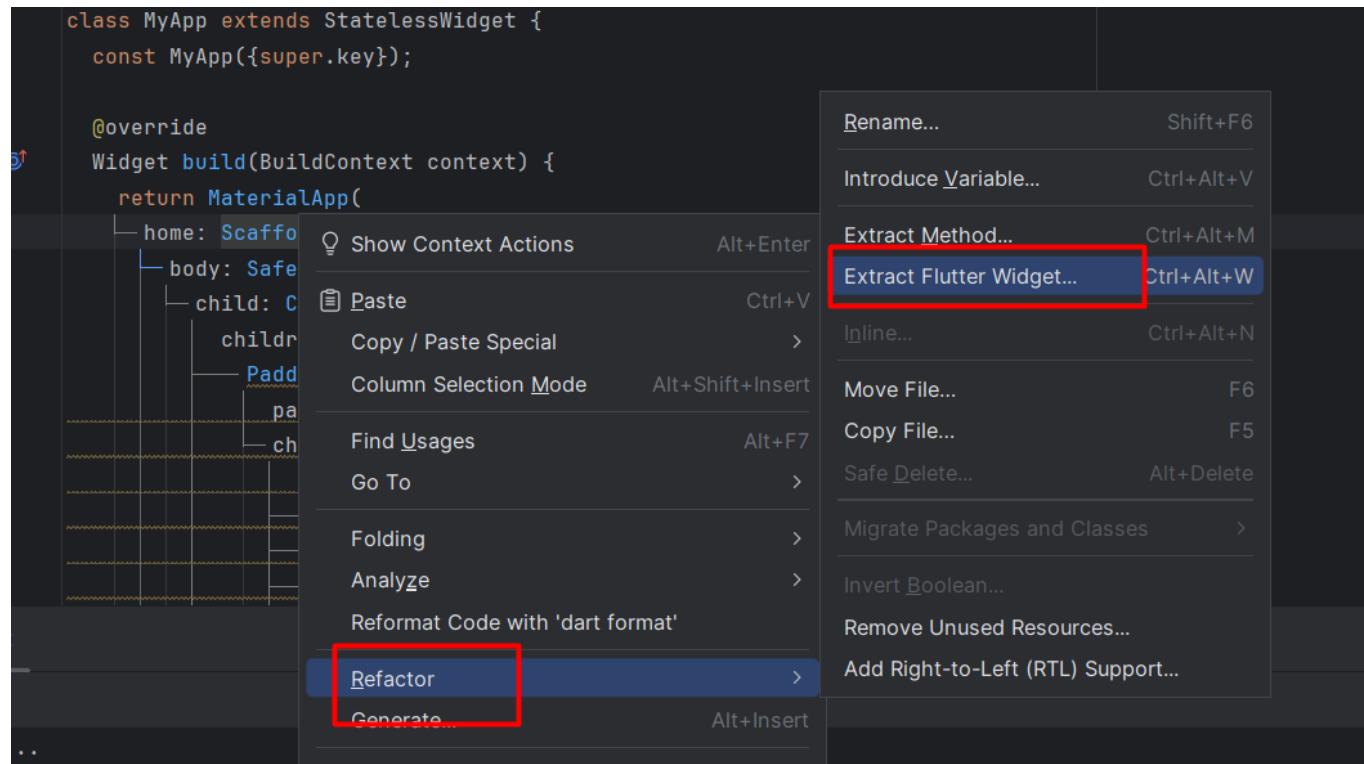
💡 **`Expanded`** : 자식 위젯이 최대 크기를 차지할 수 있도록 확장시킨다. `Image.asset` : 를 통해 이미지를 위젯으로 표현한다. `fit: BoxFit.cover` : 이미지가 부모 위젯의 경계 내에서 가능한 한 크게, 그러나 원본 비율을 유지하면서 전체를 덮도록 조정된다. 즉, 이미지가 잘리더라도 비율을 유지하며 전체 공간을 채운다.**

[Women](#)[Kids](#)[Shoes](#)[Women](#)

DEBUG



5. 컴포넌트 분리하기



main.dart

```

import 'package:flutter/material.dart';

import 'component/store_page.dart';

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false, // 상단의 debug 를 지운다.
      home: StorePage(),
    );
  }
}

```

💡 debugShowCheckedModeBanner 는 플러터 개발 모드에서 나타나는 개발 환경일 때 DEBUG로 실행되며 나타나는 배너이다. 릴리즈 모드에선 나타나지 않으며 속성을 false 로 설정하면 배너를 숨길 수 있다.

component/store_page.dart

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

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

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: SafeArea(
        child: Column(
          children: [
            Padding( // 좌우위아래 간격을 25씩 준다.
              padding: const EdgeInsets.all(25.0),
              child: Row(
                children: [
                  Text("Women", style: TextStyle(fontWeight: FontWeight.bold)),
                  Spacer(),
                  Text("Kids", style: TextStyle(fontWeight: FontWeight.bold)),
                  Spacer(),
                  Text("Shoes", style: TextStyle(fontWeight: FontWeight.bold)),
                  Spacer(),
                  Text("Women", style: TextStyle(fontWeight: FontWeight.bold)),
                ],
              ),
            ),
            Expanded(child: Image.asset("assets/bag.jpeg", fit: BoxFit.cover,)),
            SizedBox(height: 2), // 사이 간격을 위해 사용
            Expanded(child: Image.asset("assets/cloth.jpeg", fit: BoxFit.cover,)),
          ],
        ),
      );
    }
}
```

💡 컴포넌트를 기능별로 나누면 가독성이 높아지고 유지보수가 편해진다.

[Women](#)[Kids](#)[Shoes](#)[Women](#)