

# 화면전환

## 화면전환

- Navigator
  - 스택 방식으로 자식 위젯을 관리
  - push()와 pop()으로 위젯을 쌓고 제거하는 방식으로 화면을 제어
  - 단순 화면전환

```
import 'package:flutter/material.dart';
main() => runApp(WidgetDemo());

class WidgetDemo extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo App',
      home : FirstPage()
    );
  }
}

class FirstPage extends StatelessWidget{
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('First Page'),
      ),
      body: Center(
        child: ElevatedButton(
          onPressed: (){
            // 버튼을 클릭할 경우 새로운 위젯을 호출
            Navigator.push(
              context,
              MaterialPageRoute(builder: (context) => SecondPage()),
            );
          },
          child: Text('Go to next Page'),
        ),
      ),
    );
  }
}

class SecondPage extends StatelessWidget{
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Second Page'),
      ),
      body: Center(
        child: ElevatedButton(
```

```

        onPressed: ()=> Navigator.pop(context), // 버튼을 클릭할 경우 현재 위젯을 제거
        child: Text('Go to previous Page'),
      ),
    ),
  );
}
}

```

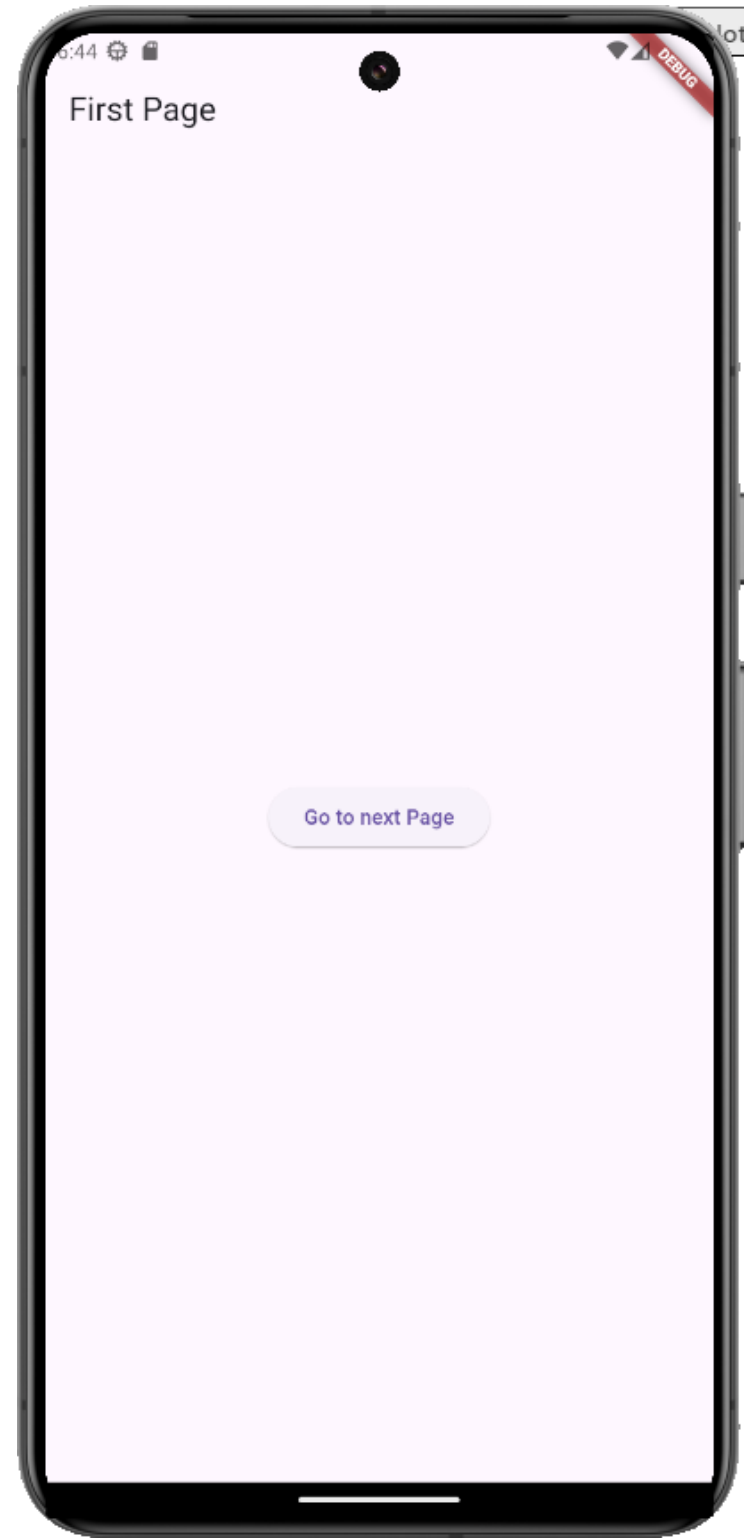
#### 1. Go to next Page 버튼 클릭

- Second Page 위젯으로 전환
  - Second Page 위젯 추가



#### 2. Go to previous Page 버튼 클릭

- First Page 위젯으로 전환
  - Second Page 위젯 제거



◦ 데이터 전달

```
import 'package:flutter/material.dart';
main() => runApp(WidgetDemo());

class WidgetDemo extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo App',
      home : FirstPage()
    );
  }
}

class FirstPage extends StatefulWidget {
  @override
  State<StatefulWidget> createState()=> _FirstPageState();
}

class _FirstPageState extends State<FirstPage>{
  var result;
```

```

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text('First Page'),
    ),
    body: Column(
      mainAxisAlignment: MainAxisAlignment.center,
      children: <Widget>[
        Center(
          child: ElevatedButton(
            onPressed: () async {
              // 버튼을 클릭할 경우 새로운 위젯을 호출
              var data = await Navigator
                .push(
                  context ,
                  MaterialPageRoute(
                    builder: (context) => SecondPage( data: '(request)')
                  )
                );
              // 상태 변경 알림
              setState(()=> result = data);
            },
            child: Text('Go to next Page'),
          ),
        ),
        Text('$result')
      ],
    )
  );
}

class SecondPage extends StatelessWidget{
  final String data;

  SecondPage({required this.data});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Second Page'),
      ),
      body: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: <Widget>[
          Center(
            child: ElevatedButton(
              // 버튼을 클릭할 경우 현재 위젯을 제거
              onPressed: ()=> Navigator.pop(context, '(result)'),
              child: Text('Go to previous Page'),
            ),
          ),
          Text('$data')
        ],
      )
    );
  }
}

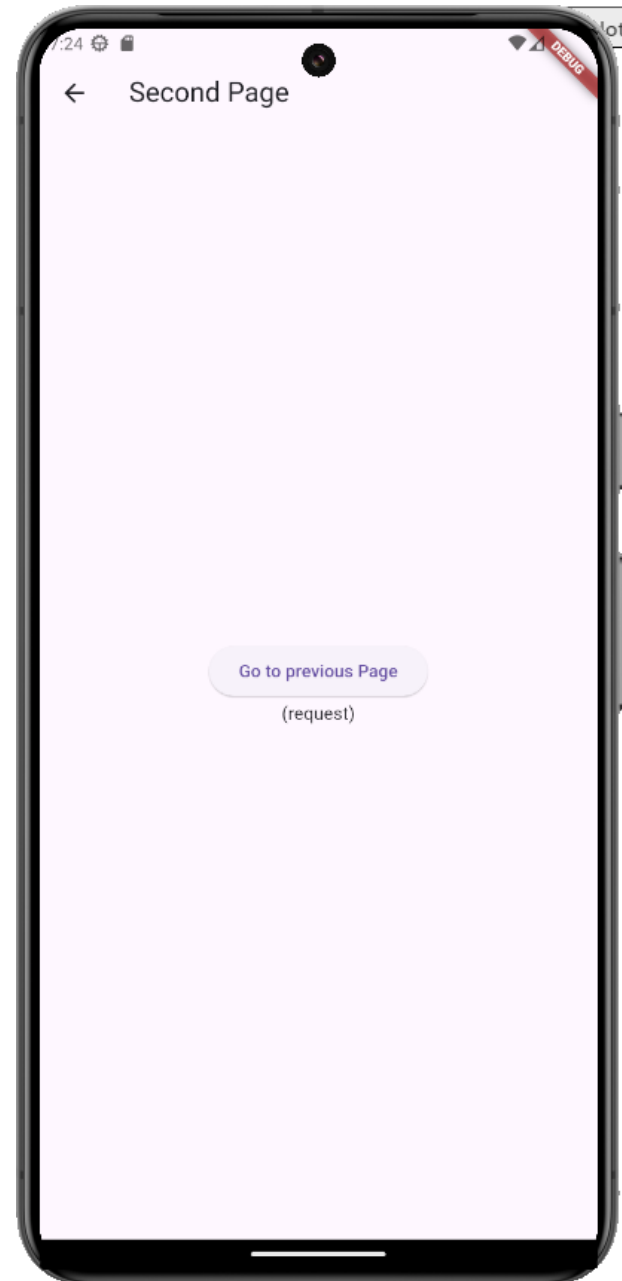
```

```
}  
}
```

#### 1. Go to next Page 버튼 클릭

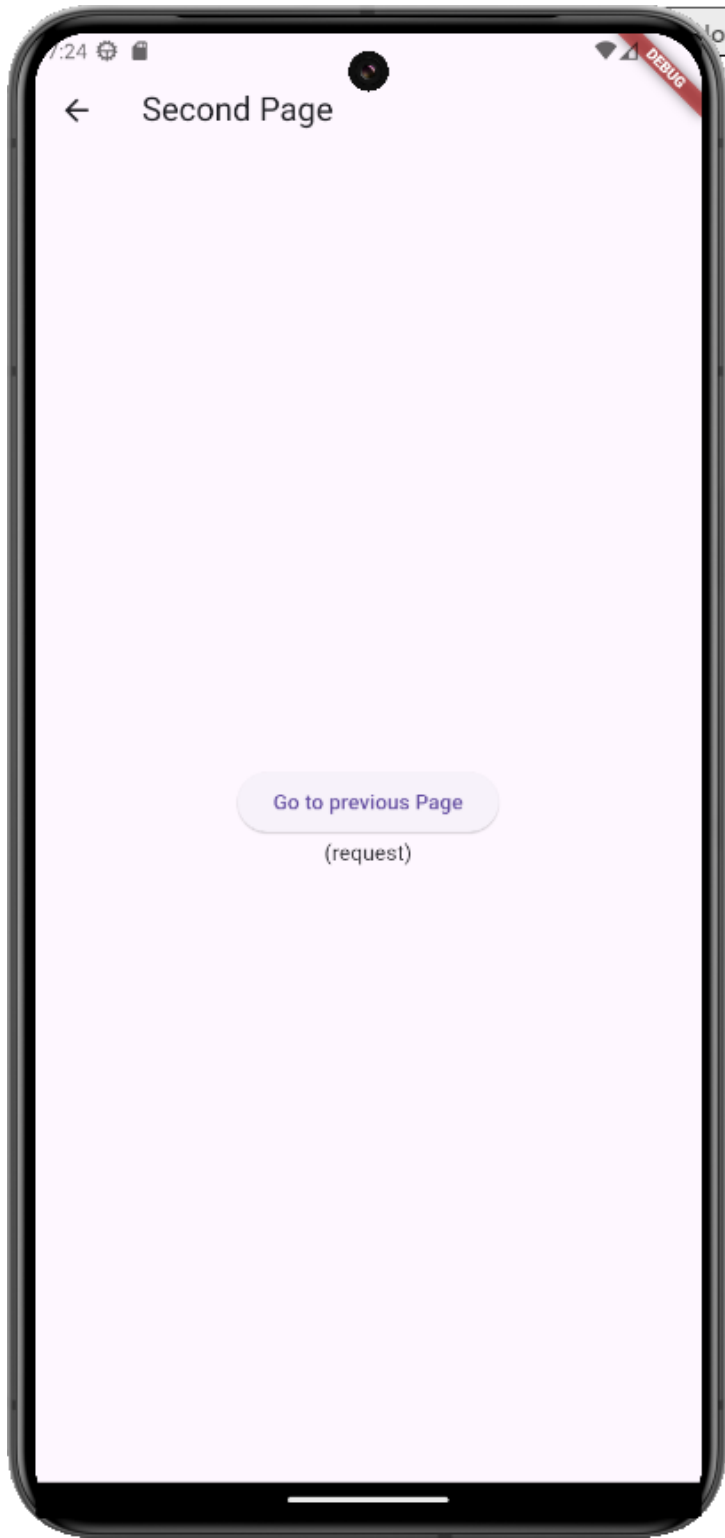


- SecondPage 위젯 호출 및 데이터 전달

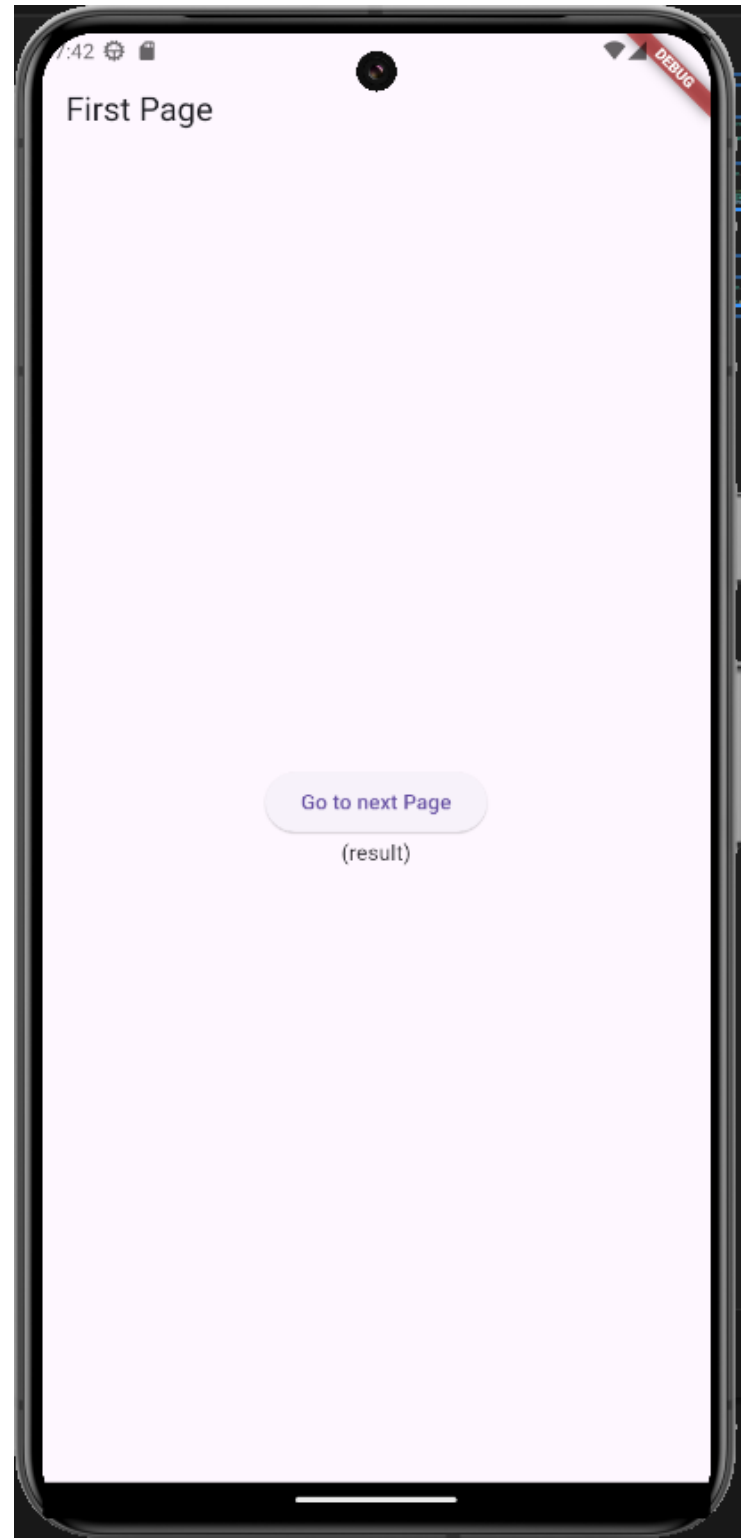


#### 2. Go to previous Page 버튼 클릭

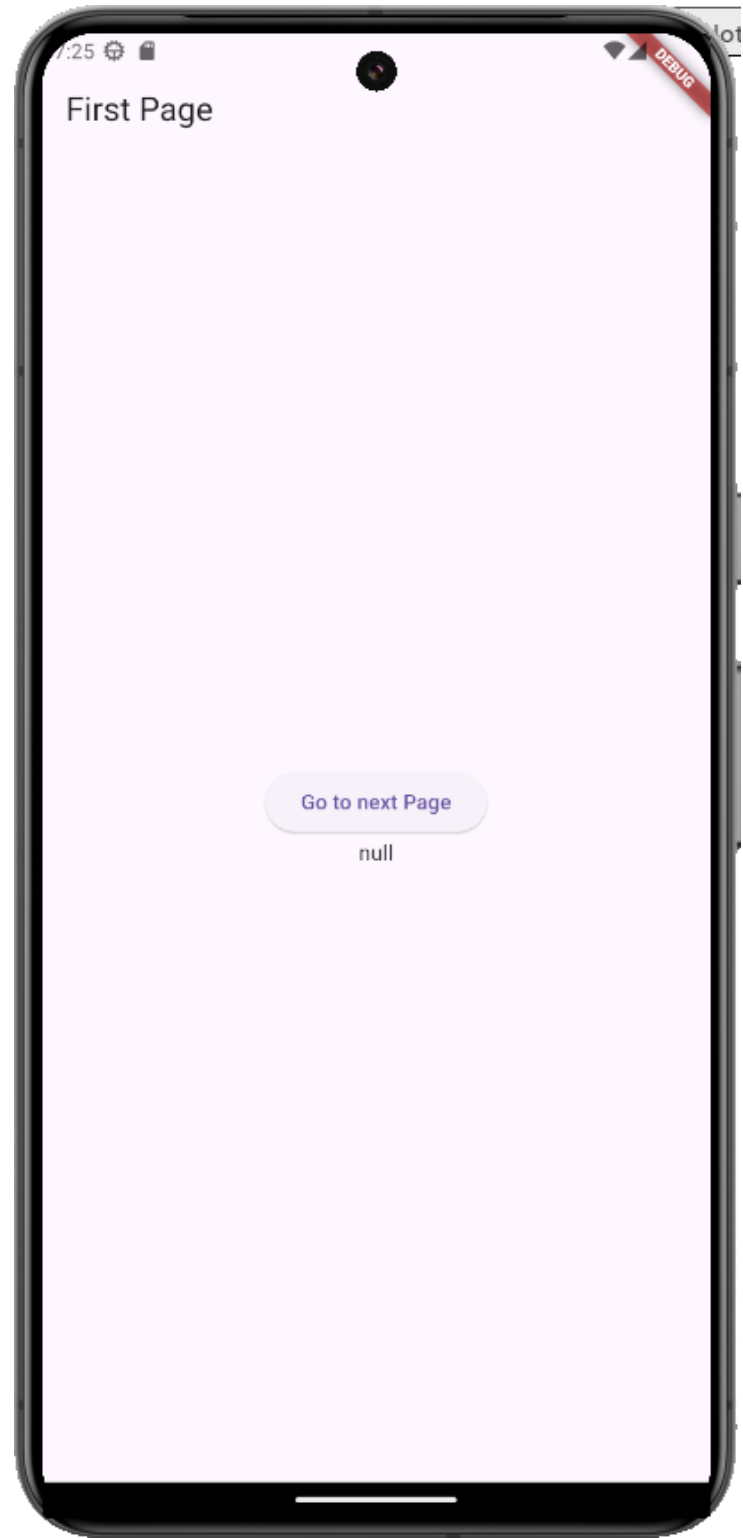
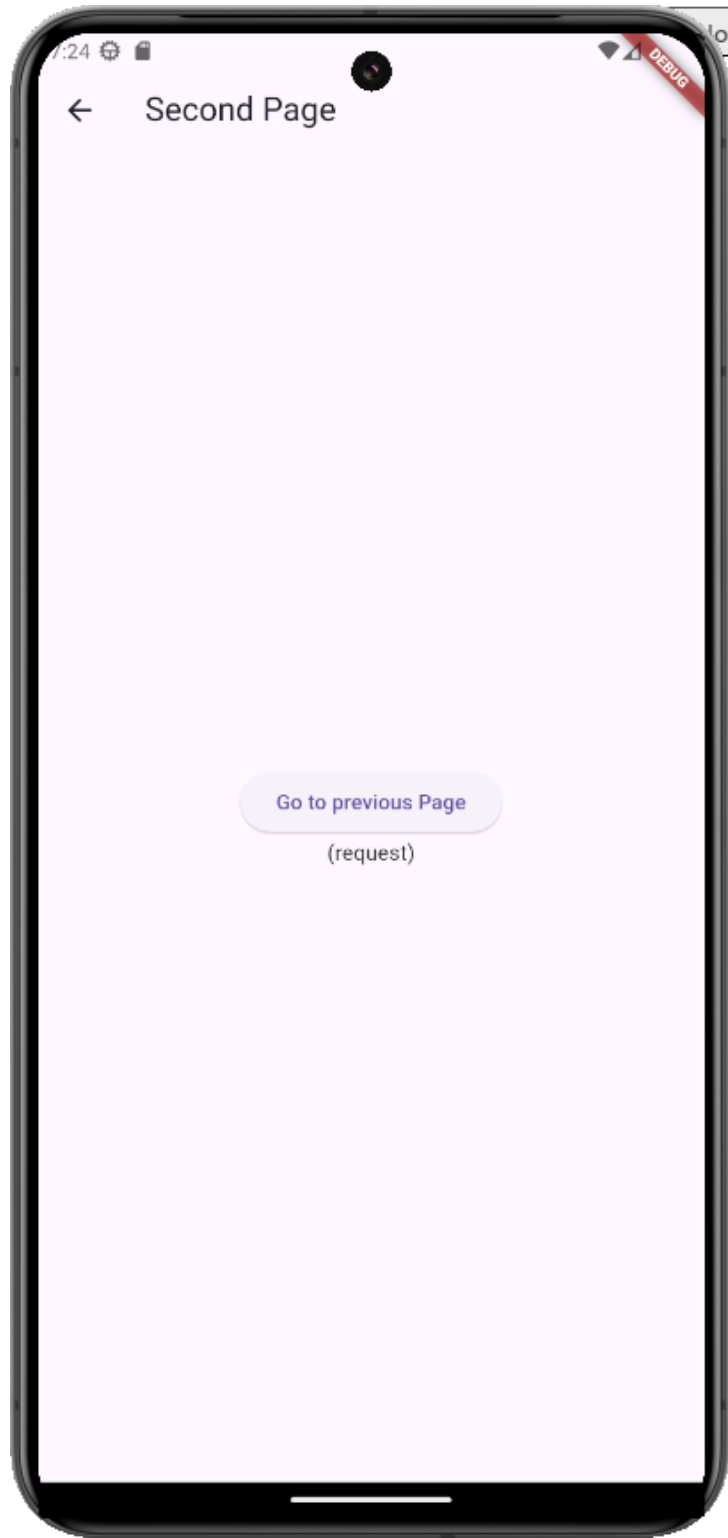
- SecondPage 위젯 제거 및 데이터 전달



3. AppBar 의 뒤로가기(←) 클릭



- SecondPage 위젯만 제거



- Route
  - Navigator가 관리하는 위젯을 추상화하여 코드적으로 좀더 간결함을 제공

```
import 'package:flutter/material.dart';
main() => runApp(WidgetDemo());

class WidgetDemo extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo App',
      // home : FirstPage()
      initialRoute: '/page1',
      // 사용될 화면을 이름과 함께 routes에 등록
      routes: {
        '/page1' : (context) => FirstPage(),
        '/page2' : (context) => SecondPage(data: '(request)'),
      },
    );
  }
}

class FirstPage extends StatefulWidget {
```

```

@override
State<StatefulWidget> createState()=> _FirstPageState();

}

class _FirstPageState extends State<FirstPage>{
  var result;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('First Page'),
      ),
      body: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: <Widget>[
          Center(
            child: ElevatedButton(
              onPressed: () async {
                // 버튼을 클릭할 경우 새로운 위젯을 호출 : routes에 등록된 이름
                var data = await Navigator
                  .pushNamed(context , '/page2');
                // 상태 변경 알림
                setState((){
                  result = data;
                });
              },
              child: Text('Go to next Page'),
            ),
          ),
          Text('$result')
        ],
      ),
    );
  }
}

class SecondPage extends StatelessWidget{
  final String data;

  SecondPage({required this.data});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Second Page'),
      ),
      body: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: <Widget>[
          Center(
            child: ElevatedButton(
              // 버튼을 클릭할 경우 현재 위젯을 제거
              onPressed: ()=> Navigator.pop(context, '(result)'),
              child: Text('Go to previous Page'),
            ),
          ),
        ],
      ),
    );
  }
}

```



```
        ),  
        Text('$data')  
    ]  
)  
);  
}  
}
```