

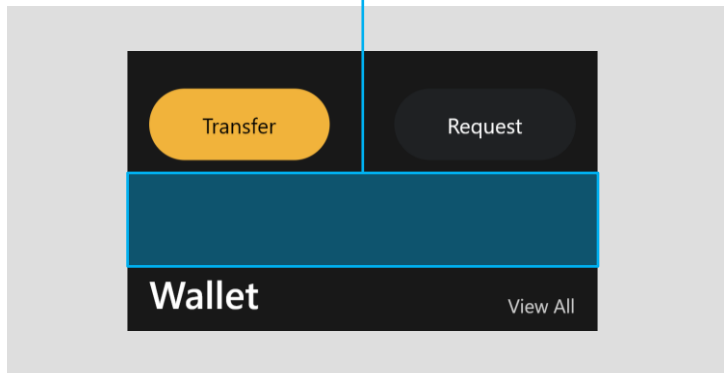
# App Front 4주차 강의 정리

```
lookup.KeyValue  
f.constant(['em  
=tf.constant([G  
lookup.StaticV  
_buckets=5)
```

## 3.6 Cards

SizedBox

```
const SizedBox(  
  height: 100,  
), // SizedBox
```

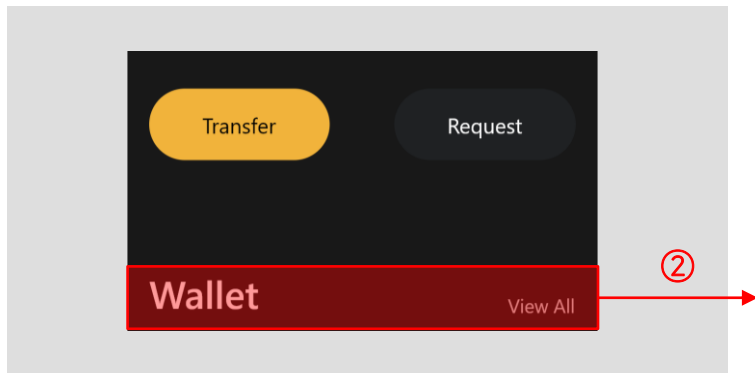


```
Row(  
  children: [  
    const Text(  
      'Wallet',  
      style: TextStyle(  
        color: Colors.white,  
        fontSize: 36,  
        fontWeight: FontWeight.w600,  
      ), // TextStyle  
    ), // Text  
    Text(  
      'View All',  
      style: TextStyle(  
        color: Colors.white.withOpacity(0.8),  
        fontSize: 18,  
      ), // TextStyle  
    ), // Text  
  ],  
), // Row
```



## 3.6 Cards

```
const SizedBox(  
  height: 100,  
), // SizedBox
```



```
Row(  
  children: [  
    const Text(  
      'Wallet',  
      style: TextStyle(  
        color: Colors.white,  
        fontSize: 36,  
        fontWeight: FontWeight.w600,  
      ), // TextStyle  
    ), // Text  
    Text(  
      'View All',  
      style: TextStyle(  
        color: Colors.white.withOpacity(0.8),  
        fontSize: 18,  
      ), // TextStyle  
    ), // Text  
  ],  
), // Row
```

Row

## 3.6 Cards

```
), // SizedBox
Row(
  crossAxisAlignment:
    CrossAxisAlignment.end,
  mainAxisAlignment:
    MainAxisAlignment.spaceBetween,
  children: [
    const Text(
      'Wallet',
      style: TextStyle(
        color: Colors.white,
        fontSize: 36,
        fontWeight: FontWeight.w600,
      ), // TextStyle
    ), // Text
    Text(
      'View All',
      style: TextStyle(
        color: Colors.white.withOpacity(0.8),
        fontSize: 18,
      ), // TextStyle
    ), // Text
  ],
),
```

가로축을 기준으로 아래로 정렬  
시작과 끝에 배치

**Wallets** View All



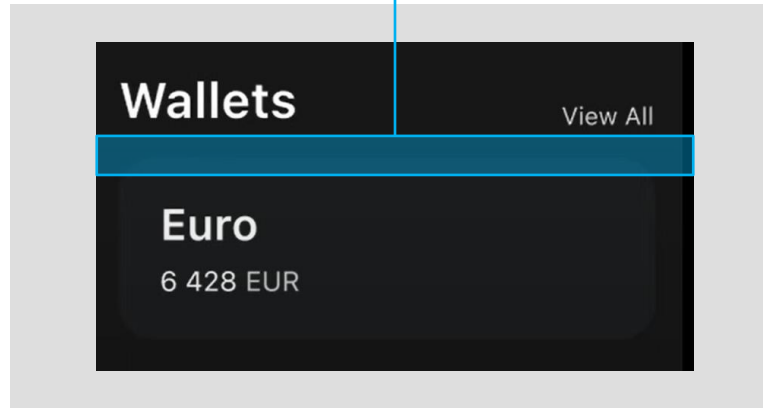
**Wallets**

View All

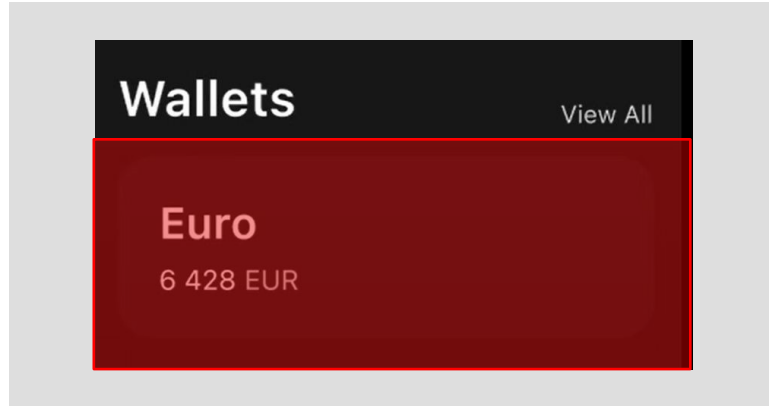
## 3.6 Cards

SizedBox

```
const SizedBox(  
  height: 20,  
) , // SizedBox
```



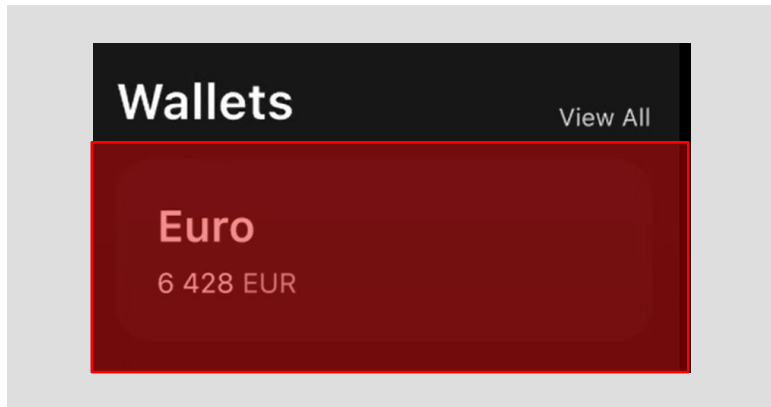
## 3.6 Cards



```
Container(  
  decoration: const BoxDecoration(  
    color: Color(0xFF1F2123),  
  ), // BoxDecoration  
  child: const Row(  
    children: [  
      Column(  
        children: [  
          Text('Euro', style: TextStyle(color: Colors.white)),  
          SizedBox(  
            height: 10,  
          ), // SizedBox  
          Row(  
            children: [Text('6428'), Text('EUR')],  
          ) // Row  
        ],  
      ), // Column  
    ],  
  ), // Row  
), // Container
```

Container-decoration

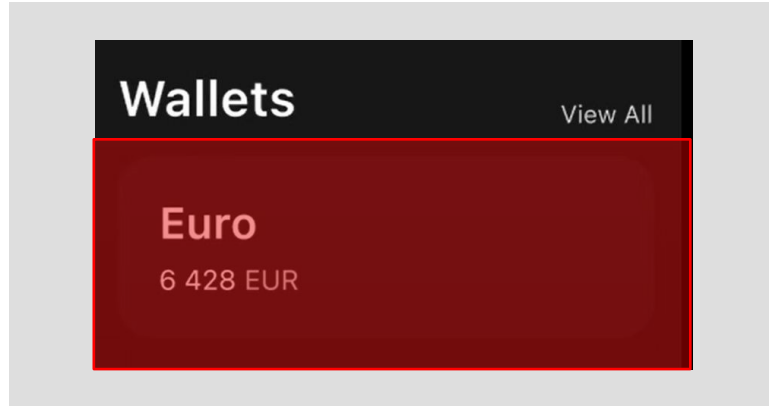
## 3.6 Cards



```
Container(  
  decoration: BoxDecoration(  
    color: const Color(0xFF1F2123),  
    borderRadius: BorderRadius.circular(25),  
  ), // BoxDecoration  
  child: const Padding(  
    padding: EdgeInsets.all(20),  
    child: Row(  
      children: [  
        Column(  
          children: [  
            Text(  
              'Euro',  
              style: TextStyle(  
                color: Colors.white,  
                fontSize: 32,  
                fontWeight: FontWeight.w600,  
              ), // TextStyle  
            ), // Text  
            SizedBox(  
              height: 10,  
            ), // SizedBox  
            Row(  
              children: [Text('6428'), Text('EUR')],  
            ), // Row  
          ],  
        ), // Column  
      ],  
    ), // Row  
  ), // Padding  
), // Container
```

Container-Text, borderRadius, Padding

## 3.6 Cards



```
Row(  
  children: [  
    const Text(  
      '6 428',  
      style: TextStyle(  
        color: Colors.white,  
        fontSize: 20,  
      ), // TextStyle  
    ), // Text  
    Text('EUR',  
      style: TextStyle(  
        color: Colors.white.withOpacity(0.8))  
      ),  
  ],  
) // Row
```

Container-TextStyle



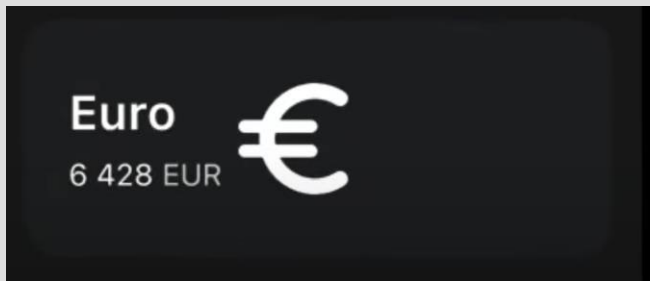
## 3.6 Cards

```
const SizedBox(
  height: 20,
), // SizedBox
Container(
  decoration: BoxDecoration(
    color: const Color(0xFF1F2123),
    borderRadius: BorderRadius.circular(25),
  ), // BoxDecoration
  child: Padding(
    padding: const EdgeInsets.all(30),
    child: Row(
      children: [
        Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            const Text(
              'Euro',
              style: TextStyle(
                color: Colors.white,
                fontSize: 32,
                fontWeight: FontWeight.w600,
              ), // TextStyle
            ), // Text
            const SizedBox(
              height: 10,
            ), // SizedBox
            Row(
              children: [
                const Text(
                  '6 428',
                  style: TextStyle(
                    color: Colors.white,
                    fontSize: 20,
                  ), // TextStyle
                ), // Text
                const SizedBox(width: 5),
                Text(
                  'EUR',
                  style: TextStyle(
                    color: Colors.white.withOpacity(0.8),
                  ), // TextStyle
                ), // Text
              ],
            ), // Row
          ],
        ), // Column
      ],
    ), // Row
  ), // Container
```



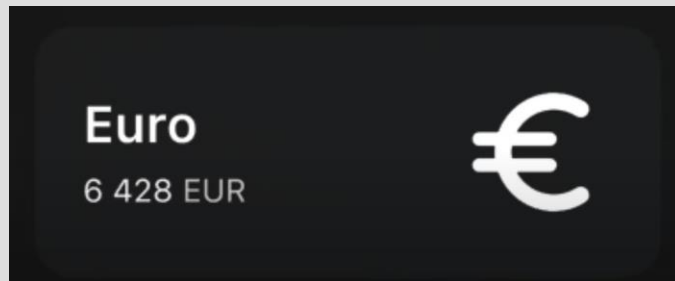
## 3.7 Icons and Transforms

```
const Icon(  
  Icons.euro_rounded,  
  color: Colors.white,  
  size: 98,  
) // Icon
```



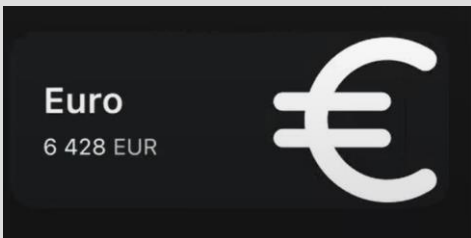
시작과 끝에 배치

```
mainAxisAlignment: MainAxisAlignment.spaceBetween,  
children: [  
  Column( // Column ...  
    const Icon(  
      Icons.euro_rounded,  
      color: Colors.white,  
      size: 98,  
    ) // Icon
```

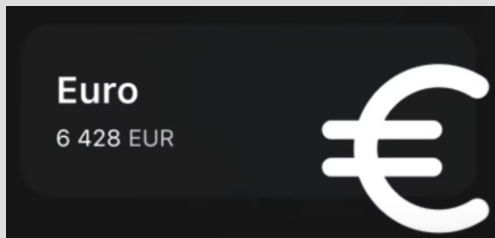


## 3.7 Icons and Transforms

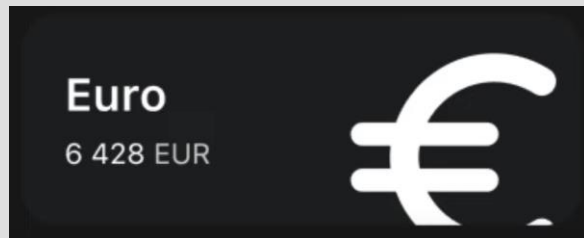
```
mainAxisAlignment: MainAxisAlignment.spaceBetween,  
children: [  
  Column( // Column ...  
    Transform.scale(  
      scale: 5,  
      child: const Icon(  
        Icons.euro_rounded,  
        color: Colors.white,  
        size: 98,  
      ), // Icon  
    ), // Transform.scale  
  ],
```



```
Transform.scale(  
  scale: 2.2,  
  child: Transform.translate(  
    offset: const Offset(8, 15),  
    child: const Icon(  
      Icons.euro_rounded,  
      color: Colors.white,  
      size: 88,  
    ), // Icon  
  ), // Transform.translate  
), // Transform.scale
```



```
Container(  
  clipBehavior: Clip.hardEdge,
```



## 3.8 Reusable Cards

```
lib > widgets > currency_card.dart > CurrencyCard
1
2
3 import 'package:flutter/material.dart';
4
5 class CurrencyCard extends StatelessWidget {
6
7   @override
8   Widget build(BuildContext context) {
9     // TODO: implement build
10    throw UnimplementedError();
11  }
12
13 }
```

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

class CurrencyCard extends StatelessWidget {
  // Constructors for public widgets shown here

  @override
  Widget build(BuildContext context) {
    // The body might complete normally, causing
    // container(
    clipBehavior: Clip.hardEdge,
    decoration: BoxDecoration(
      color: const Color(0xFF1F2123),
      borderRadius: BorderRadius.circular(25),
    ), // BoxDecoration
    child: Padding(
      padding: const EdgeInsets.all(30),
      child: Row(
        mainAxisAlignment: MainAxisAlignment.spaceBetween,
        children: [
          Column(
            crossAxisAlignment: CrossAxisAlignment.start,
            children: [
              const Text(
                'Euro',
                style: TextStyle(
                  color: Colors.white,
                  fontSize: 32,
                  fontWeight: FontWeight.w600,
                ), // TextStyle
              ), // Text
              const SizedBox(
                height: 10,
              ), // SizedBox
              Row(
                children: [
                  const Text(
                    '6.428',
                    style: TextStyle(
                      color: Colors.white,
                      fontSize: 20,
                    ), // TextStyle
                  ), // Text
                  const SizedBox(width: 5),
                  Text(
                    'EUR',
                    style: TextStyle(
                      color: Colors.white.withOpacity(0.8),
                    ),
                  ),
                ],
              ),
            ],
          ),
        ],
      ),
    ),
  ),
}
```

## 3.8 Reusable Cards

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

class CurrencyCard extends StatelessWidget {
  final String name, code, amount;
  final IconData icon;

  const CurrencyCard(
    {super.key,
    required this.name,
    required this.code,
    required this.amount,
    required this.icon});
```

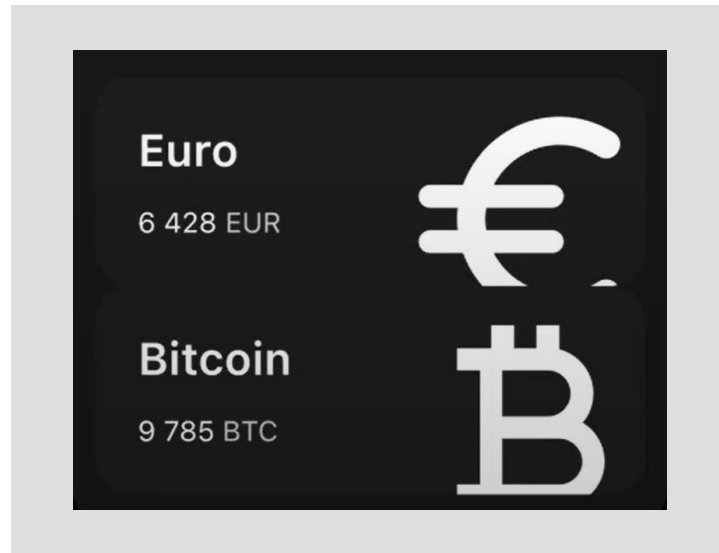
```
Text(
  name,
  style: TextStyle(
    color: Colors.white,
    fontSize: 32,
    fontWeight: FontWeight.w600,
  ), // TextStyle
), // Text
```

```
Text(
  amount,
  style: TextStyle(
    color: Colors.white,
    fontSize: 20,
  ), // TextStyle
), // Text
```

```
child: Icon(
  icon,
  color: Colors.white,
  size: 88,
), // Icon
```

## 3.8 Reusable Cards

```
const CurrencyCard(  
  name: 'Euro',  
  code: 'EUR',  
  amount: '6 428',  
  icon: Icons.euro_rounded,  
), // CurrencyCard  
const CurrencyCard(  
  name: 'Bitcoin',  
  code: 'BTC',  
  amount: '9 785',  
  icon: Icons.currency_bitcoin,  
), // CurrencyCard
```



## 3.8 Reusable Cards

```
class CurrencyCard extends StatelessWidget {  
  final String name, code, amount;  
  final IconData icon;  
  final bool isInverted;  
  
  const CurrencyCard({  
    super.key,  
    required this.name,  
    required this.code,  
    required this.amount,  
    required this.icon,  
    required this.isInverted,  
  });
```

```
(  
  final _blackcolor = const Color(0xFF1F2123);  
)
```

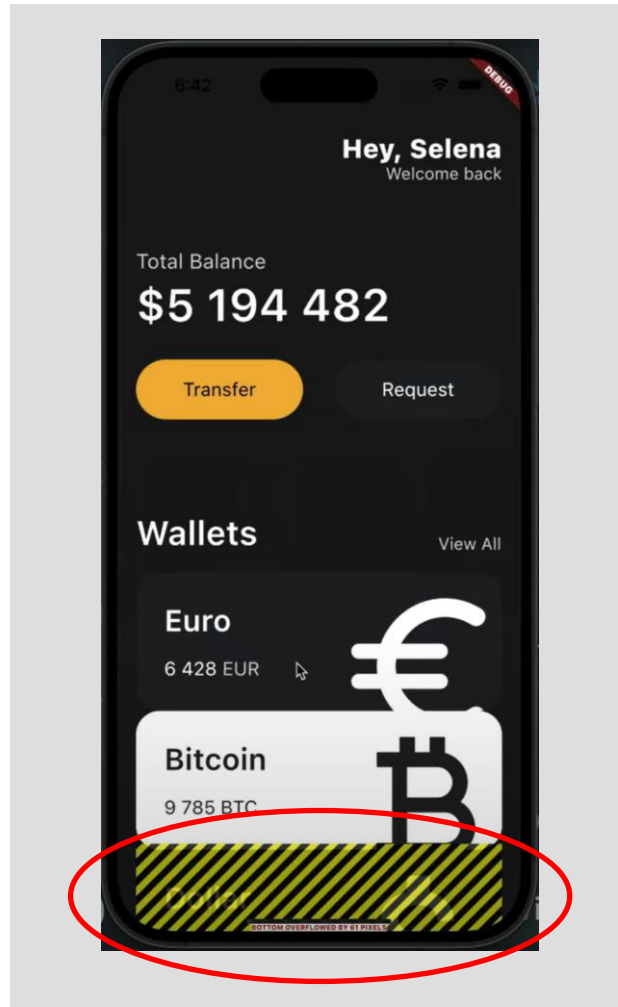
\_ : private한 변수 (현재 widget 내에서만 사용 가능한 변수)

```
isInverted ? Colors.white : _blackcolor
```

```
const SizedBox(  
  height: 20,  
), // SizedBox  
const CurrencyCard(  
  name: 'Euro',  
  code: 'EUR',  
  amount: '6 428',  
  icon: Icons.euro_rounded,  
  isInverted: false,  
), // CurrencyCard  
const CurrencyCard(  
  name: 'Bitcoin',  
  code: 'BTC',  
  amount: '9 785',  
  icon: Icons.currency_bitcoin,  
  isInverted: true,  
), // CurrencyCard  
const CurrencyCard(  
  name: 'Dollar',  
  code: 'USD',  
  amount: '428',  
  icon: Icons.attach_money_outlined,  
  isInverted: false,  
), // CurrencyCard
```



## 3.8 Reusable Cards





## 3.8 Reusable Cards

```
@override
Widget build(BuildContext context) {
  return MaterialApp(
    home: Scaffold(
      backgroundColor: const Color(0xFF181818),
      body: SingleChildScrollView(
        child: Padding(
          padding: const EdgeInsets.symmetric(
```

SingleChildScrollView

or

```
const CurrencyCard(
  name: 'Euro',
  code: 'EUR',
  amount: '6 428',
  icon: Icons.euro_rounded,
  isInverted: false,
), // CurrencyCard
Transform.translate(
  offset: const Offset(0, -20),
  child: const CurrencyCard(
    name: 'Bitcoin',
    code: 'BTC',
    amount: '9 785',
    icon: Icons.currency_bitcoin,
    isInverted: true,
  ), // CurrencyCard
), // Transform.translate
Transform.translate(
  offset: const Offset(0, -40),
  child: const CurrencyCard(
    name: 'Dollar',
    code: 'USD',
    amount: '428',
    icon: Icons.attach_money_outlined,
    isInverted: false,
  ), // CurrencyCard
), // Transform.translate
```

Offset

# #4 Stateful Widgets

## 4.0 State

```
6 class App extends StatefulWidget {  
7   @override  
8   State<App> createState() => _AppState();  
9 }
```

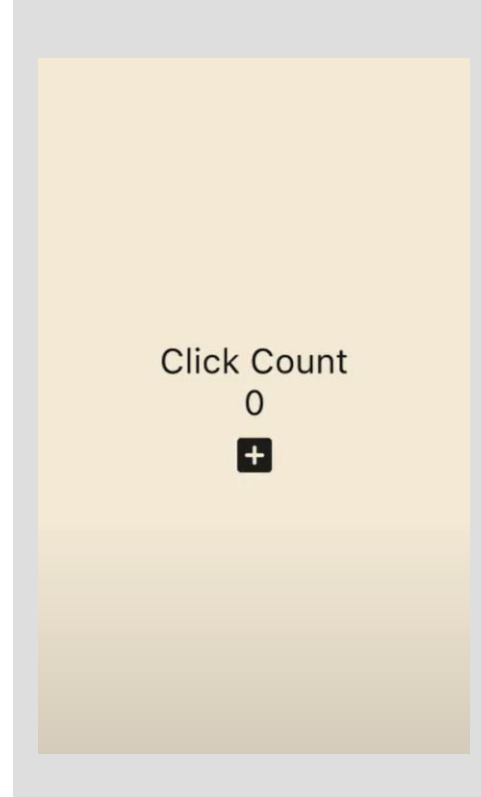
Widget 그 자체

```
2 class _AppState extends State<App> {  
3   @override  
4   Widget build(BuildContext context) {  
5     return MaterialApp(  
6       home: Scaffold(  
7         backgroundColor: const Color(0xFF181818),  
8         body: Contatiner(),  
9       ),  
10    );  
11  }  
12 }
```

데이터 및 UI

## 4.0 State

```
void main() {  
  runApp(const App());  
}  
  
class App extends StatefulWidget {  
  @override  
  State<App> createState() => _AppState();  
}  
  
class _AppState extends State<App> {  
  int counter = 0;  
  
  void onClicked() {  
    counter = counter + 1;  
  }  
  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      home: Scaffold(  
        backgroundColor: const Color(0xFF181818),  
        body: Center(  
          child: Column(  
            mainAxisAlignment: MainAxisAlignment.center,  
            children: [  
              Text(  
                'Click Count',  
                style: TextStyle(fontSize: 30),  
              ),  
              Text(  
                '$counter',  
                style: TextStyle(fontSize: 30),  
              ),  
              IconButton(  
                iconSize: 40,  
                onPressed: onClicked,  
                icon: Icon(Icons.add_box_rounded))  
            ],  
          ),  
        ),  
      );  
    }  
  }  
}
```



## 4.1 setState

```
class _Appstate extends State<App> {  
  int counter = 0;  
  
  void onCliked() {  
    setState(() {  
      counter = counter + 1;  
    });  
  }  
}
```

or

```
void onCliked() {  
  counter = counter + 1;  
  setState(() {});  
}
```



Click Count  
42



## 4.2 Recap

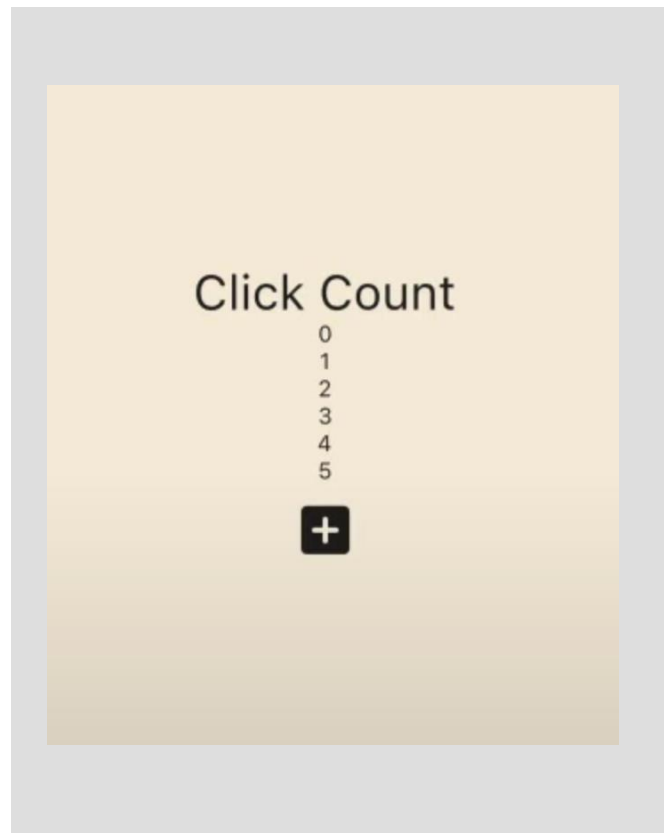
```
class App extends StatefulWidget {
  @override
  State<App> createState() => _AppState();
}

class _AppState extends State<App> {

  List<int> numbers = [];

  void onClicked() {
    setState(() {
      numbers.add(numbers.length);
    });
  }

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        backgroundColor: const Color(0xFF181818),
        body: Center(
          child: Column(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
              Text(
                'Click Count',
                style: TextStyle(fontSize: 30),
              ),
              for(var n in numbers) Text('$n'),
              IconButton(
                iconSize: 40,
                onPressed: onClicked,
                icon: Icon(Icons.add_box_rounded))
            ],
          ),
        ),
      ),
    );
  }
}
```



## 4.3 BuildContext

```
class App extends StatefulWidget {  
  @override  
  State<App> createState() => _AppState();  
}  
  
class _AppState extends State<App> {  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      theme: ThemeData(  
        textTheme: const TextTheme(  
          titleLarge: TextStyle(  
            color: Colors.red,  
          ),  
        ),  
      ),  
      home: Scaffold(  
        backgroundColor: const Color(0xFF4EDDB),  
        body: Center(  
          child: Column(  
            mainAxisAlignment: MainAxisAlignment.center,  
            children: const [  
              MyLargeTitle(),  
            ],  
          ),  
        ),  
      ),  
    );  
  }  
}
```

```
class MyLargeTitle extends StatelessWidget {  
  const MyLargeTitle({  
    Key? key,  
  }) : super(key: key);  
  
  @override  
  Widget build(BuildContext context) {  
    return Text(  
      'My Large Title',  
      style: TextStyle(  
        fontSize: 30,  
        color: Theme.of(context).textTheme.titleLarge?.color,  
      ),  
    );  
  }  
}
```

### Widget Tree

```
✓ [root]  
  ✓ A App  
    ✓ M MaterialApp  
      ✓ S Scaffold  
        ✓ C Center  
          ✓ C Column  
            ✓ M MyLargeTitle: "My Large Title"  
              T Text: "My Large Title"
```

## 4.3 Widget Lifecycle

```
7 class MyLargeTitle extends StatefulWidget {  
8   const MyLargeTitle({  
9     Key? key,  
10  }) : super(key: key);  
11  
12  @override  
13  State<MyLargeTitle> createState() => _MyLargeTitleState();  
14 }  
15  
16 class _MyLargeTitleState extends State<MyLargeTitle> {  
17   @override  
18   Widget build(BuildContext context) {  
19     return Text(  
20       'My Large Title',  
21       style: TextStyle(  
22         fontSize: 30,  
23         color: Theme.of(context).textTheme.titleLarge?.color,  
24       ),  
25     );  
26   }  
27 }  
28 }
```

```
@override  
void initState() {  
  super.initState();  
  print('initState!');  
}  
  
@override  
Widget build(BuildContext context) {  
  print('build!');  
  return Text(  
    'My Large Title',  
    style: TextStyle(  
      fontSize: 30,  
      color: Theme.of(context).textTheme.titleLarge?.color,  
    ),  
  );  
}
```

```
Restarted application in 225ms.  
flutter: initState!  
flutter: build!
```





## 4.3 Widget Lifecycle

```
class _Appstate extends State<App> {
  bool showTitle = true;

  void toggleTitle() {
    setState(() {
      showTitle = !showTitle;
    });
  }

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      theme: ThemeData(
        textTheme: const TextTheme(
          titleLarge: TextStyle(
            color: Colors.red,
          ),
        ),
      ),
      home: Scaffold(
        backgroundColor: const Color(0xFF4EDDB),
        body: Center(
          child: Column(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
              showTitle ? MyLargeTitle() : const Text('nothing'),
              IconButton(
                onPressed: toggleTitle,
                icon: const Icon(Icons.remove_red_eye),
              ),
            ],
          ),
        ),
      ),
    );
  }
}
```

```
class MyLargeTitle extends StatefulWidget {
  const MyLargeTitle({
    Key? key,
  }) : super(key: key);

  @override
  State<MyLargeTitle> createState() => _MyLargeTitleState();
}

class _MyLargeTitleState extends State<MyLargeTitle> {
  int count = 0;

  @override
  void initState() {
    super.initState();
    print('initState!');
  }

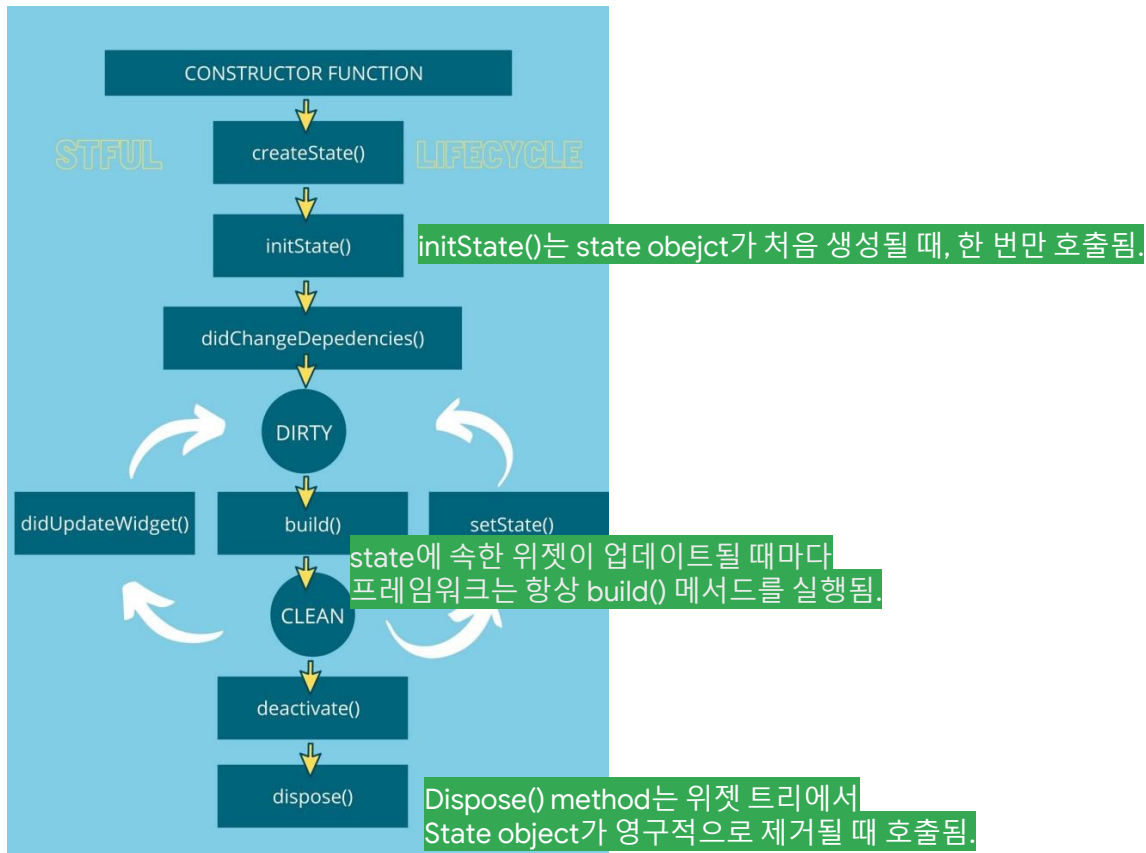
  @override
  void dispose() {
    super.dispose();
    print("dispose!");
  }

  @override
  Widget build(BuildContext context) {
    print('build!');
    return Text(
      'My Large Title',
      style: TextStyle(
        fontSize: 30,
        color: Theme.of(context).textTheme.titleLarge?.color,
      ),
    );
  }
}
```

```
Restarted application in 225ms.
flutter: initState!
flutter: build!
flutter: dispose!
flutter: initState!
flutter: build!
```



## 4.3 Widget Lifecycle



# Thank You!