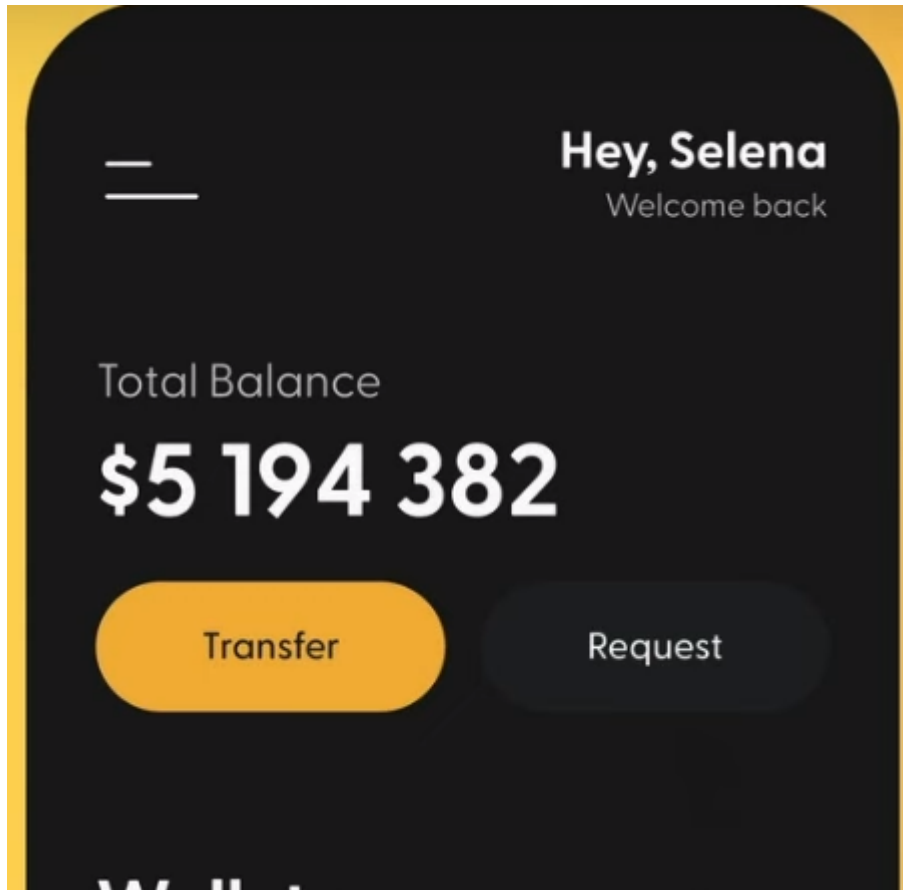


# UI Challenge

## Header



- 해당 부분을 만들기 위해
- 전체 배경 색 #181818
- 패딩 좌우로 40
- 상단의 공간을 위해 height 50을 가진 박스

```

import 'package:flutter/material.dart';

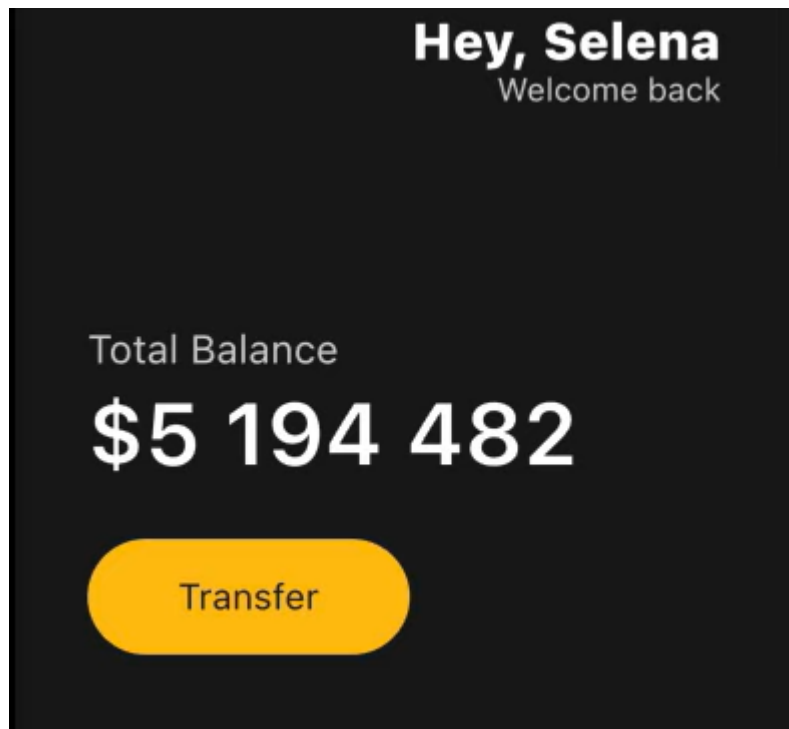
void main() {
  runApp(App());
}

class App extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        backgroundColor: Color(0xFF181818),
        body: Padding(
          padding: EdgeInsets.symmetric(horizontal: 40),
          child: Column(
            children: [
              SizedBox(
                height: 50,
              ),
              Row(
                mainAxisAlignment: MainAxisAlignment.end,
                children: [
                  Column(
                    crossAxisAlignment: CrossAxisAlignment.end,
                    children: [
                      Text(
                        'Hey, Selena',
                        style: TextStyle(
                          color: Colors.white,
                          fontSize: 32,
                          fontWeight: FontWeight.w700,
                        ),
                      ),
                      Text(
                        'Welcome back',
                        style: TextStyle(
                          color: Colors.white.withOpacity(0.8),
                          fontSize: 20,
                        ),
                      ),
                    ],
                  ),
                ],
              ),
            ],
          ),
        ),
      ),
    );
  }
}

```

- Row와 Column을 이해해야 한다.
  - MainAxisAlignment와 CrossAxisAlignment 속성을 가진다.

## Button Section



- 텍스트가 중앙 정렬돼 있을텐데
- body 뒤 바로 오는 column에
- `crossAxisAlignment: CrossAxisAlignment.start`를 작성해주어야 한다.

```
// 위 헤더와 공간을 주기위해 sizebox 활용
    SizedBox(
      height: 80,
    ),
    Text(
      'Total Balance',
      style: TextStyle(
        fontSize: 22,
        color: Colors.white.withOpacity(0.8),
      ),
    ),
    SizedBox(
      height: 5,
    ),
    Text(
      '\$5 194 482',
      // dart에서 $를 변수를 넣을 때 사용하기 때문에
      // \를 넣어 변수가 아니라는 표시를 해야함
      style: TextStyle(
        fontSize: 48,
        fontWeight: FontWeight.w600,
        color: Colors.white,
      ),
    ),
```

```

    ),
  ),
  SizedBox(
    height: 20,
  ),
  Row(
    children: [
      Container(
        decoration: BoxDecoration(
          color: Colors.amber,
          borderRadius: BorderRadius.circular(30),
        ),
        child: Padding(
          padding: EdgeInsets.symmetric(
            vertical: 20,
            horizontal: 45,
          ),
          child: Text(
            'Transfer',
            style: TextStyle(
              fontSize: 18,
              fontWeight: FontWeight.w500
            ),
          ),
        ),
      ),
    ],
  ),
)

```

## VSCode Settings

- 파란 줄 오류 뜨는 것 해결 방법
- dart의 const 때문인데,
  - const는 컴파일 전에 value를 알고 있어서
  - 해당 변수를 위한 메모리를 마련하는게 아니라
  - value를 검토하고 사용되는 곳에 value를 대체시킨다.
    - 최적화 굳
- 위젯 몇몇은 const로 되어있음
- 그래서 const를 써서 최적화하기 좋게 코드를 작성하라는 의미
  - 근데 const가 적용되는 위젯을 전부 기억 할 수 없음

## setting

설정 > 명령 팔레트 > open user settings 입력 > setting.json 열기 > 아래 코드 입력

```
"editor.codeActionsOnSave": {
  "source.fixAll": true
},
```

++ 자식 컴포넌트 표시 설정

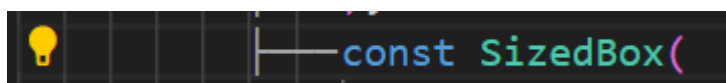
"dart.previewFlutterUiGuides": true, 입력하면

```
Widget build(BuildContext context) {
  return MaterialApp(
    home: Scaffold(
      backgroundColor: Colors.white,
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            const SizedBox(
              height: 50,
            ), // SizedBox
            Row(
              mainAxisAlignment: MainAxisAlignment.start,
              children: [
                Column(
```

자식 컴포넌트를 나타내는 선이 생긴다 개꿀

## Code Actions

- 코드를 매우 간단한 방법으로 리팩토링 해주는 기능



```
const SizedBox(
```

- 전구를 클릭하거나 'ctrl + .' 를 입력하면 된다.

- 해당 기능을 통해 위젯 안 코드를 다 빼거나 다른 위젯으로 감쌀 수 있다.