

Animasi

Penerapan animasi bukanlah hal yang wajib untuk selalu kita buat pada setiap halaman aplikasi. Melainkan harus sesuai dengan fungsinya. Jadi, jikalau animasi yang kita buat itu ternyata mengurangi kenyamanan si pengguna dalam memakai app kita maka, jelas animasi tersebut tidak ada gunanya. Di Flutter, untuk proses pembuatan animasi dibagi menjadi 2 tipe dasar yaitu, implicit animation dan explicit animation.

Implicit & Explicit Animation

Apa perbedaan dari kedua tipe dasar animasi tersebut? Secara sederhana akan saya jelaskan saja bahwa implicit animation merupakan proses pembuatan animasi yang paling gampang karena kita cukup menggunakan widget-widget animasi yang sudah disediakan oleh Flutter seperti `AnimatedContainer`, `AnimatedOpacity` dan lain sebagainya. Sementara, explicit animation adalah proses pembuatan animasi yang cukup susah karena kita harus membuat animasinya secara awal.

Buatlah file main.dart pada proyek lalu masukkan kode berikut

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

void main() => runApp(App());

class App extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: HomePage(),
    );
  }
}

class HomePage extends StatefulWidget {
  @override
  _HomePageState createState() => _HomePageState();
}
```

```

class _HomePageState extends State<HomePage> {
  var isChanged = false;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Demo AnimatedContainer'),
      ),
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            AnimatedContainer(
              duration: Duration(milliseconds: 500),
              width: 100,
              height: 100,
              decoration: BoxDecoration(
                color: isChanged ? Colors.orange : Colors.blue,
                borderRadius: BorderRadius.all(
                  Radius.circular(isChanged ? 16 : 0),
                ),
                border: Border.all(
                  color: isChanged ? Colors.blue : Colors.orange,
                  width: 4,
                ),
              ),
              transform: isChanged
                ? (Matrix4.identity()
                  ..scale(0.5, 0.5)
                  ..translate(0.5 * 100, 0.5 * 100))
                : Matrix4.identity(),
            ),
            ElevatedButton(
              child: Text('Tap Me'),
              onPressed: () {
                setState(() => isChanged = !isChanged);
              },
            ),
          ],
        ),
      ),
    );
  }
}

```

```
}
```

Jalankan kodenya maka akan tampil hasil sebagai berikut

