

Expanded & Flexible

Sejauh ini kita telah mempelajari beberapa *widget* dasar dan bagaimana menyusunnya secara horizontal maupun vertikal. Dalam pengembangan aplikasi *mobile* kita tahu bahwa terdapat banyak sekali perangkat dengan ukuran layar yang berbeda pula. Untuk itu penting bagi kita untuk bisa menyusun tampilan yang responsif terhadap ukuran layar.

Kira-kira bagaimana Anda akan menyusun layout dengan tampilan seperti berikut?



Tentunya akan sangat merepotkan apabila kita mengatur tinggi dari masing-masing kotak, bukan? Belum lagi jika harus mengembangkan aplikasi di ukuran yang lebih besar seperti perangkat tablet.

Expanded

Flutter memiliki widget Expanded yang dapat mengembangkan child dari Row atau Column sesuai dengan ruang yang tersedia. Cara menggunakannya Anda cukup membungkus masing-masing child ke dalam Expanded.

```
1. class Rainbow extends StatelessWidget {  
2.   @override  
3.   Widget build(BuildContext context) {  
4.     return Column(  
5.       children: <Widget>[  
6.         Expanded(  
7.           child: Container(  
8.             color: Colors.red,  
9.           ),  
10.        ),  
11.        Expanded(  
12.          child: Container(  
13.            color: Colors.orange,  
14.          ),  
15.        ),  
16.        Expanded(  
17.          child: Container(  
18.            color: Colors.yellow,  
19.          ),  
20.        ),  
21.        Expanded(  
22.          child: Container(  
23.            color: Colors.green,  
24.          ),  
25.        ),  
26.        Expanded(  
27.          child: Container(  
28.            color: Colors.blue,  
29.          ),  
30.        ),  
31.        Expanded(  
32.          child: Container(  
33.            color: Colors.indigo,  
34.          ),  
35.        ),  
36.        Expanded(  

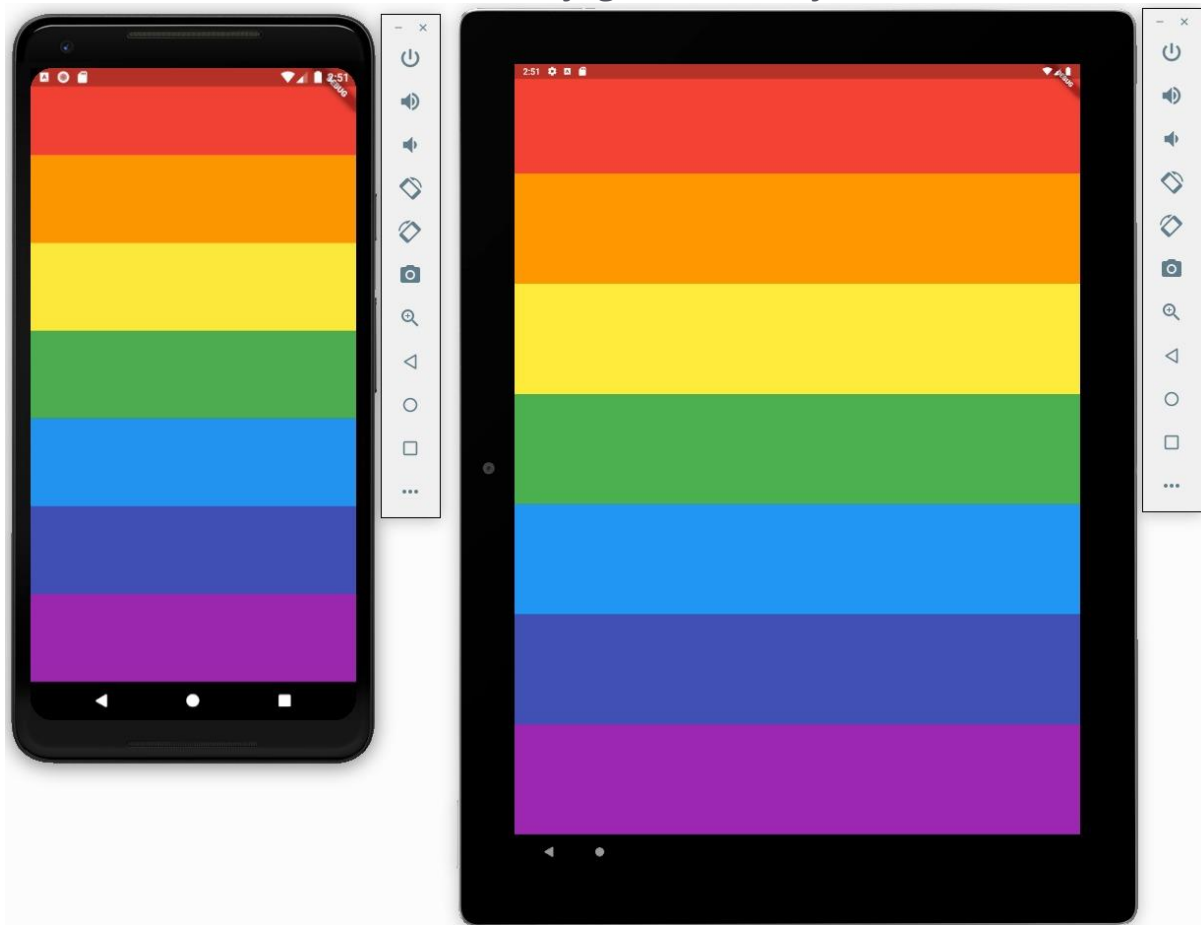
```

```

37.         child: Container(
38.             color: Colors.purple,
39.         ),
40.     ),
41. ],
42. );
43. }
44. }

```

Saat aplikasi dijalankan, masing-masing container akan menempati ruang kosong yang ada. Jika Anda menjalankan di ukuran layar yang berbeda, maka ukuran container juga akan menyesuaikan.



Bisa kita lihat seluruh container menempati ruang dengan ukuran yang sama. Ini disebabkan Expanded memiliki parameter flex yang memiliki nilai default 1. Anda dapat mengubah nilai flex ini sesuai perbandingan yang diinginkan. Misalnya Anda memberikan nilai flex 2 pada salah satu container.

```

1. Expanded(
2.   flex: 2,
3.   child: Container(
4.     color: Colors.blue,

```

5.),

6.),

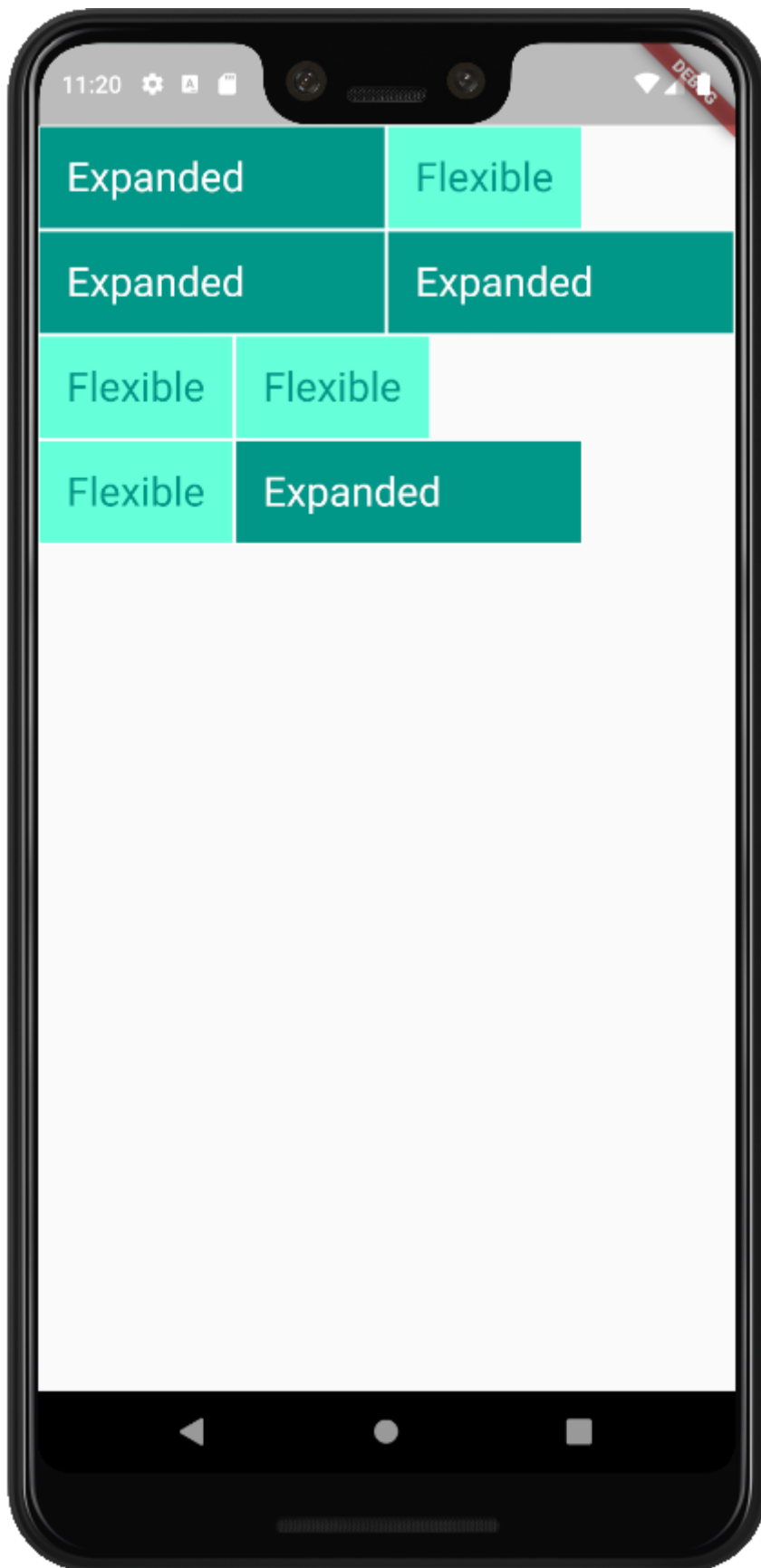
Maka container berwarna biru ini akan menjadi lebih besar dengan perbandingan $2/(1 + 1 + 1 + 1 + 2 + 1 + 1)$ atau $2/8$ dari halaman.



Flexible

Sama seperti Expanded, widget Flexible digunakan untuk mengatur ukuran widget di dalam Row atau Column secara fleksibel. Perbedaan Flexible dan Expanded adalah widget Flexible memungkinkan *child widget*-nya berukuran lebih kecil dibandingkan ukuran Flexible. Sementara, child widget dari Expanded harus menempati ruang yang tersisa dari Column atau Row.

Berikut ini adalah contoh perbedaan antara Expanded dan Flexible:



Kode untuk tampilan seperti di atas adalah seperti berikut:

```

1. class ExpandedFlexiblePage extends StatelessWidget {
2.   @override
3.   Widget build(BuildContext context) {
4.     return Scaffold(
5.       body: SafeArea(
6.         child: Column(
7.           children: [
8.             Row(
9.               children: [
10.                ExpandedWidget(),
11.                FlexibleWidget(),
12.              ],
13.            ),
14.            Row(
15.              children: [
16.                ExpandedWidget(),
17.                ExpandedWidget(),
18.              ],
19.            ),
20.            Row(
21.              children: [
22.                FlexibleWidget(),
23.                FlexibleWidget(),
24.              ],
25.            ),
26.            Row(
27.              children: [
28.                FlexibleWidget(),
29.                ExpandedWidget(),
30.              ],
31.            ),
32.          ],
33.        ),
34.      ),
35.    );
36.  }
37. }
38.
39. class ExpandedWidget extends StatelessWidget {
40.   @override
41.   Widget build(BuildContext context) {
42.     return Expanded(
43.       child: Container(
44.         decoration: BoxDecoration(
45.           color: Colors.teal,
46.           border: Border.all(color: Colors.white),
47.         ),
48.       child: Padding(
49.         padding: const EdgeInsets.all(16.0),
50.         child: Text(
51.           'Expanded',
52.           style: TextStyle(
53.             color: Colors.white,
54.             fontSize: 24,

```

```

55.         ),
56.     ),
57. ),
58. ),
59. );
60. }
61. }
62.
63. class FlexibleWidget extends StatelessWidget {
64.     @override
65.     Widget build(BuildContext context) {
66.         return Flexible(
67.             child: Container(
68.                 decoration: BoxDecoration(
69.                     color: Colors.tealAccent,
70.                     border: Border.all(color: Colors.white),
71.                 ),
72.             child: Padding(
73.                 padding: const EdgeInsets.all(16.0),
74.                 child: Text(
75.                     'Flexible',
76.                     style: TextStyle(
77.                         color: Colors.teal,
78.                         fontSize: 24,
79.                     ),
80.                 ),
81.             ),
82.         ),
83.     );
84. }
85. }

```

Dokumentasi berikut ini dapat Anda pelajari untuk memaksimalkan penggunaan widget Expanded dan Flexible:

- [Expanded Class](#)
- [Flexible Class](#)