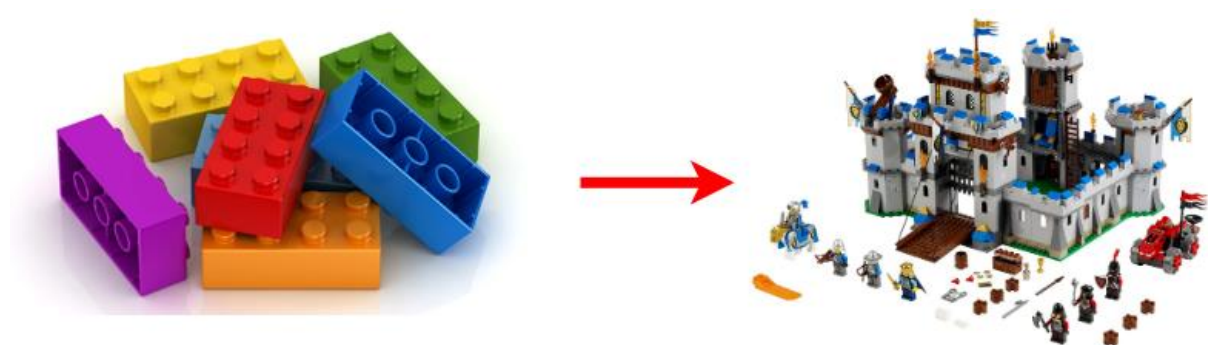


## Apa itu Widget

Kita telah berkenalan, menginstal, dan belajar fundamental Flutter. Itu *entrée* alias hidangan pembukanya. Nah, sekarang kita akan menyibak ke *plat du jour* alias menu utamanya. Apa inti dari Flutter? Yap, **widget** jawabnya! Sebagian besar yang ada pada Flutter adalah widget. Jadi, relevan jika kita bilang bahwa “Flutter is all about widget.” Text sendiri adalah widget. Button juga widget. Selain itu widget juga dapat dibangun dari kumpulan beberapa widget. Lantas mengapa widget begitu penting dalam flutter? Penasaran? Mari kita bahas!

### World of components

Perlu kita ketahui bahwa konsep Widget pada Flutter itu terinspirasi oleh salah framework JavaScript yang digunakan untuk membangun sebuah website yaitu ReactJS. ReactJS memiliki konsep Component. Mari kita analogikan dengan mainan lego! Di lego terdapat *block-block* kecil yang nantinya kita susun untuk membangun sebuah istana lego. Berarti *component* dalam *programming* adalah sekumpulan *block-block code* yang digunakan untuk membangun sebuah aplikasi.



Widget sama halnya dengan component yang merupakan kumpulan block code untuk membangun aplikasi Flutter. Ketika membangun aplikasi Flutter kita harus berpikir layaknya bermain lego. Kita harus bisa membuat dan menyusun widget-widget dengan tepat. Tujuannya, agar aplikasi yang kita buat lebih mudah untuk dikembangkan.

## Bagaimana cara menulis widget

Sebetulnya pada pembahasan sebelumnya secara tidak sadar kita telah membuat sebuah widget dan menggunakan widget yang telah disediakan.

```
1. class MyApp extends StatelessWidget {  
2.   @override  
3.   Widget build(BuildContext context) {  
4.     return MaterialApp(  
5.       title: 'Flutter Demo',  
6.       theme: ThemeData(  
7.         primarySwatch: Colors.blue,  
8.       ),  
9.       home: Scaffold(  
10.        appBar: AppBar(  
11.          title: Text('Hello, world!'),  
12.        ),  
13.        body: Center(  
14.          child: Text('Hello, world!'),
```

```

15.         ),
16.     ),
17. );
18. }
19. }

```

Pada kode di atas kita telah membuat sebuah Widget MyApp dan telah menggunakan widget-widjet bawaan Flutter di antaranya *MaterialApp*, *Scaffold*, *Center*, dan *Text*. Ketika menggunakan widget, kita tinggal panggil nama widget dan bila ada properti atau parameter pada widget tersebut tinggal kita isikan properti atau parameteranya.

```

1. Center(
2.   child: Text('Hello world!'),
3. )

```

Kode di atas merupakan contoh pemanggilan widget **Center**. Widget Center ini digunakan untuk membuat widget yang ada di dalamnya berada di posisi tengah (mirip seperti alignment center). Tinggal ketikkan **Center** lalu tambahkan properti child di dalamnya.

Perlu diketahui bahwa sebagian besar widget bawaan memiliki pola *parent-child*, seperti halnya Center yang memiliki child yang artinya di dalam child bisa terdapat widget lagi. Maka penulisan parent child akan seperti di bawah ini.

```

1. Center( // parent dari Button

```

```
2.   child: Button( // child dari Center dan parent dari Text
3.     child: Text(), // child dari Button
4.   ),
5. )
```

Pada contoh di atas widget Center dan Button hanya dapat memiliki satu anak atau bisa disebut child. Ada pula widget yang dapat memiliki banyak anak atau bisa disebut children, seperti Row, Column, ListView, GridView, dan sebagainya.

```
1. Row(
2.   children: <Widget>[
3.     //di dalam children akan berisi banyak widget
4.   ]
5. )
```

Contoh di atas adalah widget Row yang memiliki children. Di dalam children nantinya kita bisa menambahkan banyak widget. Berbeda dengan child yang diisi langsung dengan sebuah Widget, children akan berisi sebuah list yang di dalamnya diisi dengan banyak widget.