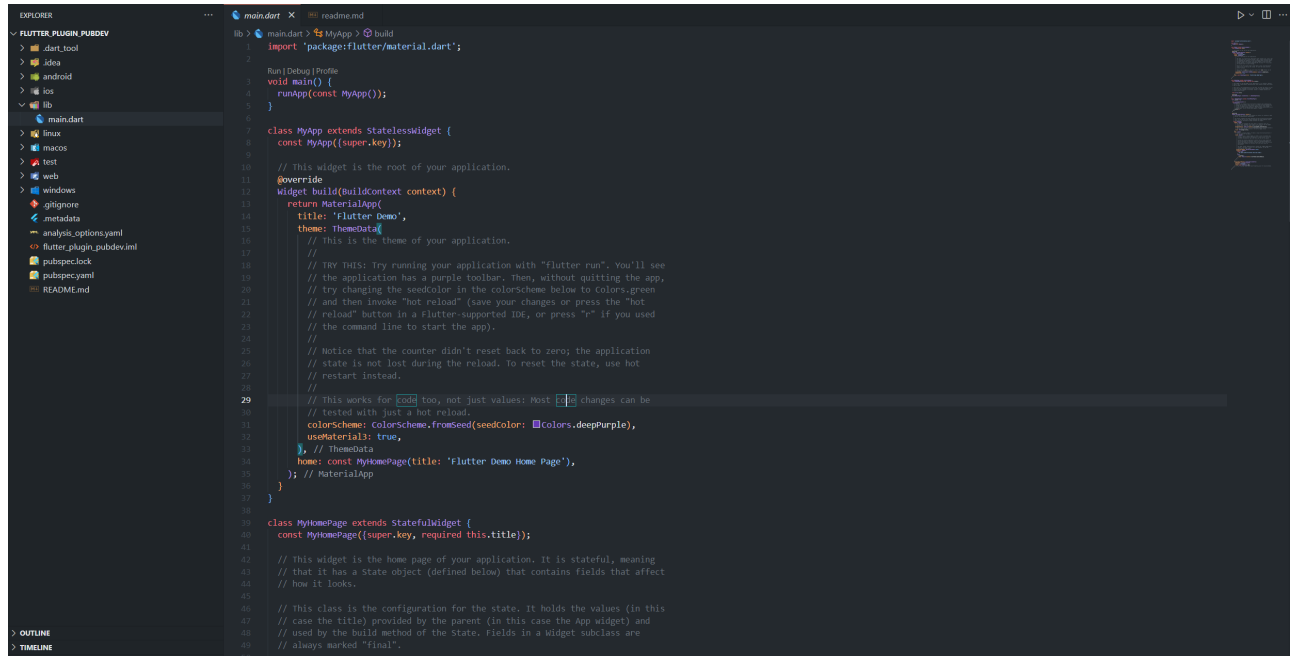


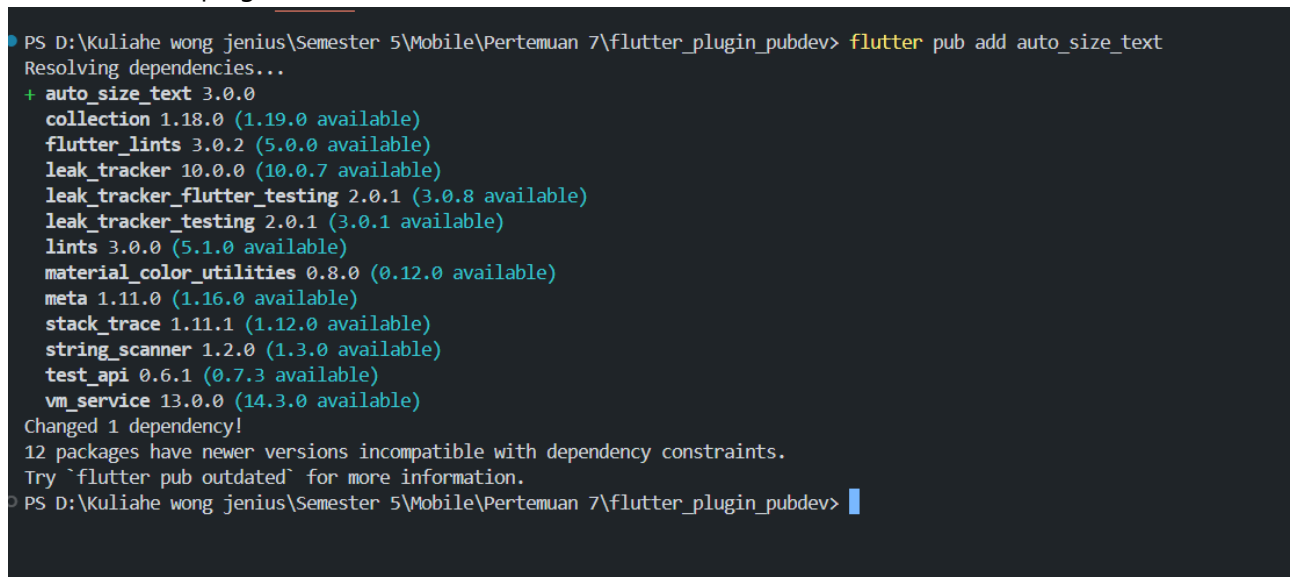
# Pertemuan 7

## Praktikum 7

### 1. Buat project baru



### 2. Menambahkan plugin



### 3. Langkah 3: Buat file red\_text\_widget.dart

```
main.dart  readme.md  red_text_widget.dart X
lib > red_text_widget.dart > RedTextWidget > build
1  import 'package:flutter/material.dart';
2
3  class RedTextWidget extends StatelessWidget{
4      const RedTextWidget({super.key});
5
6      @override
7      Widget build(BuildContext context){
8          return Container();
9      }
10 }
```

### 4. Tambah widget autosize text

```
main.dart  readme.md  red_text_widget.dart 1 X
lib > red_text_widget.dart > RedTextWidget > build
1  import 'package:auto_size_text/auto_size_text.dart';
2  import 'package:flutter/material.dart';
3
4  class RedTextWidget extends StatelessWidget{
5      const RedTextWidget({super.key});
6
7      @override
8      Widget build(BuildContext context){
9          return AutoSizeText(
10             text,      Undefined name 'text'. Try correcting the name to one that is def
11             style: const TextStyle(color: Colors.red,fontSize: 14),
12             maxLines: 2,
13             overflow: TextOverflow.ellipsis,
14             ); // AutoSizeText
15     }
16 }
```

## 5. Buat variabel text dan parameter constructor

```

lib > red_text_widget.dart > RedTextWidget > build
1  import 'package:auto_size_text/auto_size_text.dart';
2  import 'package:flutter/material.dart';
3
4  class RedTextWidget extends StatelessWidget{
5    const RedTextWidget({super.key,required this.text});
6    final String text;
7    @override
8    Widget build(BuildContext context){
9      return AutoSizeText(
10        text,
11        style: const TextStyle(color: Colors.red,fontSize: 14),
12        maxLines: 2,
13        overflow: TextOverflow.ellipsis,
14      ); // AutoSizeText
15    }
16  }

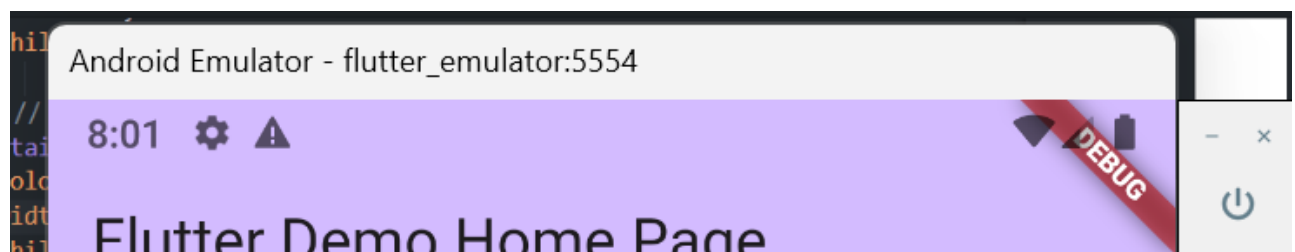
```

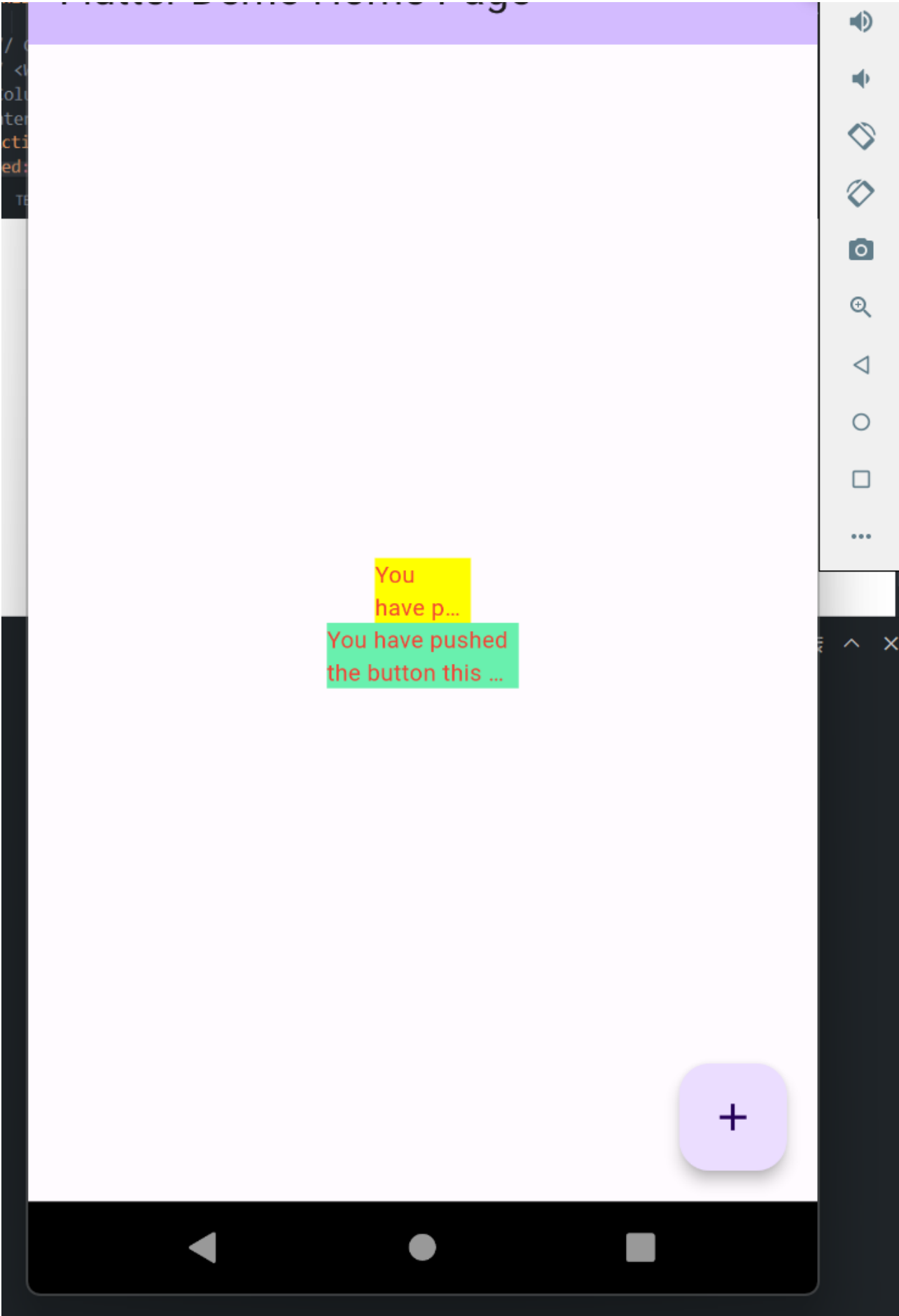
## 6. Tambahkan widget di main.dart

```

lib > main.dart > _MyHomePageState > build
58  class _MyHomePageState extends State<MyHomePage> {
73    Widget build(BuildContext context) {
102      // horizontal).
103      //
104      // TRY THIS: Invoke "debug painting" (choose the "Toggle Debug Paint"
105      // action in the IDE, or press "p" in the console), to see the
106      // wireframe for each widget.
107      mainAxisAlignment: MainAxisAlignment.center,
108      children: <Widget>[
109        Container(
110          color: Colors.yellowAccent,
111          width: 50,
112          child: const RedTextWidget(
113            text: 'You have pushed the button this many times:'), // RedTextWidget
114          ), // Container
115        Container(
116          color: Colors.greenAccent,
117          width: 100,
118          child: const RedTextWidget(
119            text: 'You have pushed the button this many times:'), // RedTextWidget
120          ), // Container
121      ], // <Widget>[]
122    ), // Column
123  ), // Center
124  floatingActionButton: FloatingActionButton(
125    onPressed: _incrementCounter,

```





Tugas Praktikum

2. Jelaskan maksud dari langkah 2 pada praktikum tersebut Menambahkan plugin atau package eksternal dengan menggunakan terminal :

```
pub get
```

3. Jelaskan maksud dari langkah 5 pada praktikum tersebut Menggunakan plugin autosized text, serta mengatur style teks yang akan digunakan nanti sebagai widget.

4. pada langkah 6 terdapat dua widget yang ditambahkan, jelaskan fungsi dan perbedaannya

```
Container(  
  color: Colors.yellowAccent,  
  width: 50,  
  child: const RedTextWidget(  
    text: 'You have pushed the button this many times:'),  
),
```

jawaban : menyiapkan ukuran kotak dengan lebar 50 pixel, dengan child RedTextWidget dengan isi text berupa string.

```
Container(  
  color: Colors.greenAccent,  
  width: 100,  
  child: const RedTextWidget(  
    text: 'You have pushed the button this many times:'),  
)
```

jawaban : menyiapkan ukuran kotak dengan lebar 100 pixel, dengan child RedTextWidget dengan isi text berupa string.

Perbedaannya terdapat pada lebar containernya yang menjadikan text yang ditampilkan mengikuti lebar dari containernya sehingga lebar pada container dengan width 100 menampilkan teks yang lebih banyak dari lebar 50

5. Jelaskan maksud dari tiap parameter yang ada di dalam plugin auto\_size\_text berdasarkan tautan pada dokumentasi ini !

Parameter	Deskripsi
key*	Mengontrol bagaimana satu widget menggantikan widget lain di pohon.
TextKey	Mengatur kunci untuk Text widget yang dihasilkan
Style	Jika bukan null, gaya yang akan digunakan untuk teks ini
minFontSize	Batasan ukuran teks minimum yang akan digunakan saat mengubah ukuran teks secara otomatis. Diabaikan jika presetFontSizesditetapkan.

Parameter	Deskripsi
<code>maxFontSize</code>	Batasan ukuran teks <b>maksimum</b> yang akan digunakan saat mengubah ukuran teks secara otomatis. Diabaikan jika <code>presetFontSizes</code> ditetapkan.
<code>stepGranularity</code>	Ukuran langkah di mana ukuran font disesuaikan dengan batasan.
<code>presentFontSizes</code>	Menentukan terlebih dahulu semua ukuran font yang memungkinkan <b>Penting:</b> <code>presetFontSizes</code> harus dalam urutan menurun.
<code>group</code>	Sinkronisasikan ukuran beberapa <code>AutoSizeText</code>
<code>textAlign*</code>	Bagaimana teks harus disejajarkan secara horizontal.
<code>textDirection*</code>	Arah teks. Ini menentukan bagaimana <code>textAlign</code> nilai seperti <code>TextAlign.start</code> dan <code>TextAlign.end</code> diinterpretasikan.
<code>locale*</code>	Digunakan untuk memilih font ketika karakter Unicode yang sama dapat ditampilkan secara berbeda, tergantung pada lokal.
<code>softwrap*</code>	Apakah teks harus diputus pada jeda baris lunak.
<code>wrapWords*</code>	Apakah kata-kata yang tidak muat dalam satu baris harus dibungkus. Defaultnya adalah <code>true</code> perilaku seperti <code>Text</code> .
<code>overflow*</code>	Bagaimana luapan visual harus ditangani.
<code>overflowReplacement*</code>	Jika teksnya meluap dan tidak sesuai batasnya, widget ini akan ditampilkan sebagai gantinya.
<code>textScaleFactor*</code>	Jumlah piksel font untuk setiap piksel logis. Juga memengaruhi <code>minFontSize</code> , <code>maxFontSize</code> dan <code>presetFontSizes</code> .
<code>maxLines*</code>	Jumlah baris maksimum opsional untuk teks yang akan direntangkan.
<code>semanticsLabel*</code>	Label semantik alternatif untuk teks ini.