

Pemrograman Mobile Jobsheet-7: Manajemen Plugin



NIM : 2341760017

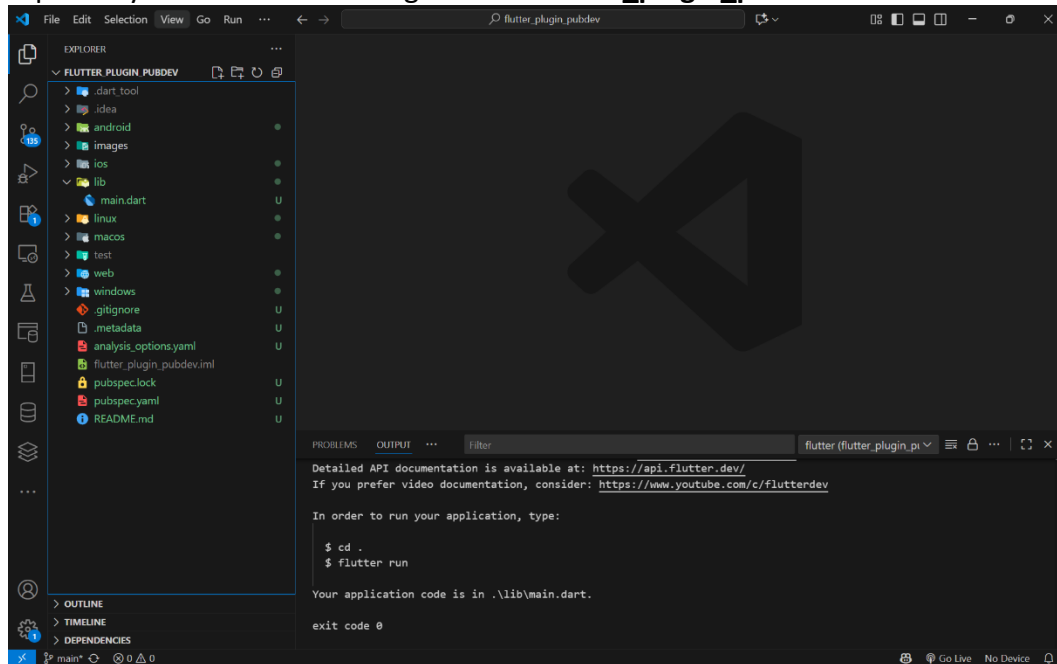
Nama : Marga Reta Novia Putri

Kelas : SIB-3D / 10

Praktikum Menerapkan Plugin di Project Flutter

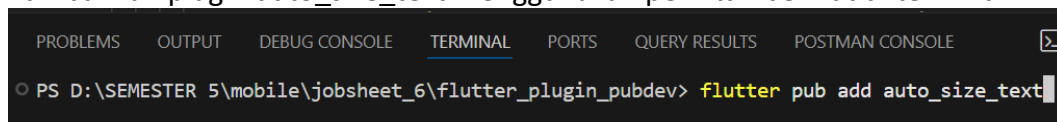
1. Langkah 1: Buat Project Baru

Buatlah sebuah project flutter baru dengan nama **flutter_plugin_pubdev**. Lalu jadikan repository di GitHub Anda dengan nama **flutter_plugin_pubdev**.

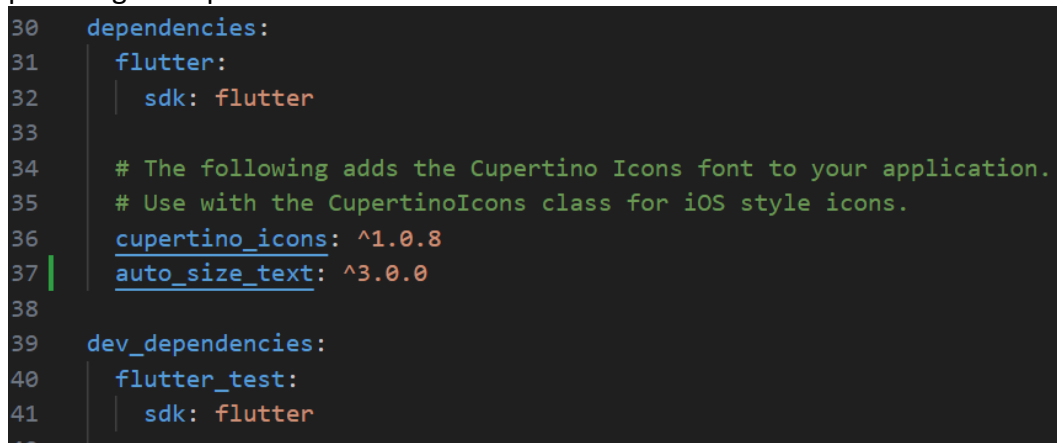


2. Langkah 2: Menambahkan Plugin

Tambahkan plugin **auto_size_text** menggunakan perintah berikut di terminal



Jika berhasil, maka akan tampil nama plugin beserta versinya di file **pubspec.yaml** pada bagian **dependencies**.



3. Langkah 3: Buat file red_text_widget.dart

Buat file baru bernama red_text_widget.dart di dalam folder lib lalu isi kode seperti berikut.

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

4. Langkah 4: Tambah Widget AutoSizeText

Masih di file red_text_widget.dart, untuk menggunakan plugin auto_size_text, ubahlah kode return Container() menjadi seperti berikut.

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

class RedTextWidget extends StatelessWidget {
  const RedTextWidget({Key? key}): super(key: key);

  @override
  Widget build(BuildContext context) {
    return AutoSizeText (
      text,
      style: const TextStyle( color: Colors.red, fontSize: 14)
      maxLines: 2,
      overflow: TextOverflow.ellipsis,
    ); // AutoSizeText
  }
}
```

Setelah Anda menambahkan kode di atas, Anda akan mendapatkan info error. Mengapa demikian? Jelaskan dalam laporan praktikum Anda!

Error terjadi karena variabel text belum didefinisikan di dalam class RedTextWidget

5. Langkah 5: Buat Variabel text dan parameter di constructor

Tambahkan variabel text dan parameter di constructor seperti berikut.

```
4  class RedTextWidget extends StatelessWidget {
5    final String text;
6
7    const RedTextWidget({Key? key, required this.text}) : super(key: key);
8  }
```

6. Langkah 6: Tambahkan widget di main.dart

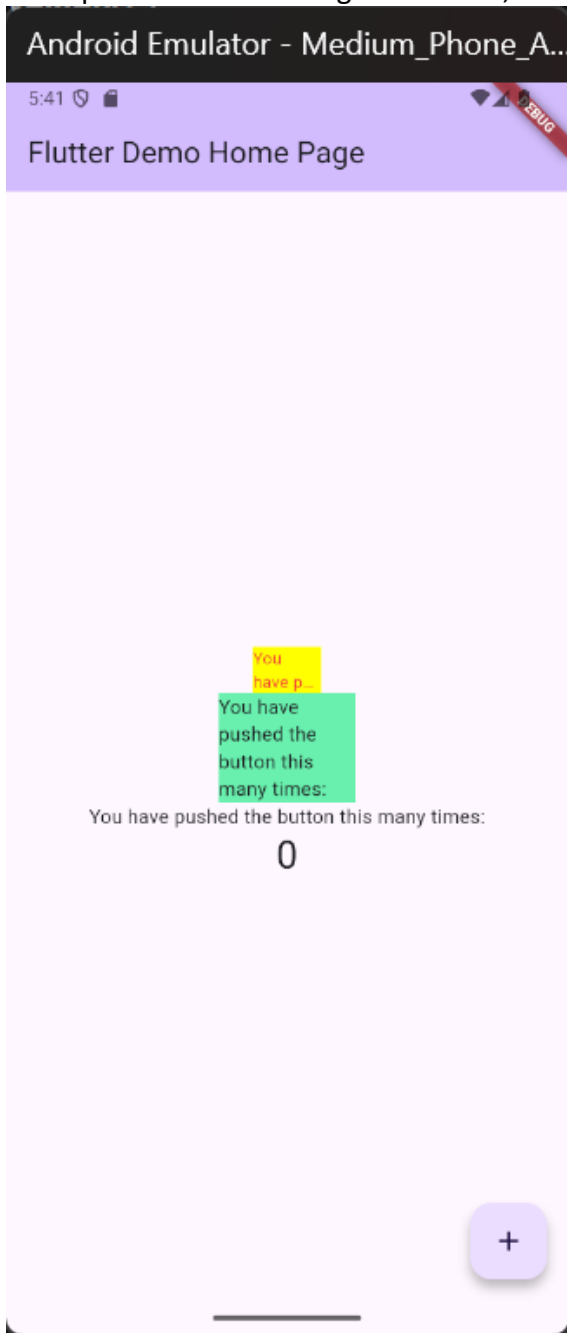
Buka file main.dart lalu tambahkan di dalam children: pada class _MyHomePageState

```

107     children: <Widget>[
108       Container(
109         color: Colors.yellowAccent,
110         width: 50,
111         child: const RedTextWidget(
112           text: 'You have pushed the button this many times:',
113         ), // RedTextWidget
114       ), // Container
115       Container(
116         color: Colors.greenAccent,
117         width: 100,
118         child: const Text('You have pushed the button this many times:'),
119       ), // Container
120       const Text('You have pushed the button this many times:'),

```

Run aplikasi tersebut dengan tekan **F5**, maka hasilnya akan seperti berikut.



Tugas Praktikum

1. Selesaikan Praktikum tersebut, lalu dokumentasikan dan push ke repository Anda berupa screenshot hasil pekerjaan beserta penjelasannya di file README.md!
2. Jelaskan maksud dari langkah 2 pada praktikum tersebut!

Jawab: Menambahkan plugin ke proyek dengan cara menjalankan perintah flutter pada terminal

3. Jelaskan maksud dari langkah 5 pada praktikum tersebut!

Jawab: Mendefinisikan variabel text agar widget tahu teks apa yang akan ditampilkan, menambahkan parameter di constructor supaya saat RedTextWidget digunakan bisa memberikan nilai text dari luar, dan memastikan nilai text wajib diisi (required), dan tidak bisa berubah (final)sesuai konsep StatelessWidget.

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

Jawab: Widget pertama (kuning) menunjukkan kelebihan AutoSizeText dan Widget kedua (hijau) menunjukkan perilaku Text biasa.

5. Jelaskan maksud dari tiap parameter yang ada di dalam plugin auto_size_text berdasarkan tautan pada dokumentasi ini !

Jawab:

Parameter	Penjelasan
key	Standard parameter untuk semua widget di Flutter: mengidentifikasi widget dalam tree, mengontrol bagaimana widget diganti/re-build.
textKey	Kunci (Key) khusus yang akan diteruskan ke widget Text yang akhirnya dirender di dalam AutoSizeText.
style	TextStyle yang akan digunakan sebagai basis ukuran font, warna, gaya teks, dll.
minFontSize	Menentukan ukuran font terkecil yang diperbolehkan ketika teks di-resize agar muat dalam batasan layout.
maxFontSize	Menentukan ukuran font terbesar yang diperbolehkan (bila style.fontSize terlalu besar atau ingin dibatasi).
stepGranularity	Menentukan seberapa banyak ukuran font akan dikurangi tiap langkah supaya teks muat. Misalnya: dari 40 → 39 → 38 (step=1) vs 40 → 30 → 20 (step=10).
presetFontSizes	Sebuah daftar ukuran font yang sudah ditetapkan dan dicoba satu per satu alih-alih menurunkan ukuran secara bertahap.
group	Mengelompokkan beberapa instance AutoSizeText agar semua mengikuti ukuran font yang sama (yakni size terkecil dari anggota grup) agar tampilannya konsisten.
textAlign	Menentukan bagaimana teks disejajarkan secara horizontal (misalnya TextAlign.left, center, right).
textDirection	Arah teks: kiri-ke-kanan atau kanan-ke-kiri.
locale	Locale yang digunakan untuk memilih font sesuai karakter lokal atau kebiasaan locale tersebut.
softWrap	Apakah teks boleh dibungkus ("wrap") pada batasan baris lembut (soft line break).

wrapWords	Menentukan apakah kata-kata yang tidak muat dalam satu baris boleh dipindah ke baris berikutnya (true), atau harus dipotong.
overflow	Menentukan bagaimana perilaku jika teks masih tidak muat setelah penyesuaian ukuran. Misalnya: <code>TextOverflow.ellipsis</code> , <code>clip</code> , <code>fade</code> .
overflowReplacement	Jika teks tidak muat dan ukuran sudah turun sampai batas (<code>minFontSize</code>) tetapi masih overflow, maka widget ini akan ditampilkan sebagai pengganti teks.
textScaleFactor	Faktor skala teks (misalnya untuk accessibility). Juga mempengaruhi perhitungan <code>minFontSize</code> , <code>maxFontSize</code> , <code>presetFontSizes</code> .
maxLines	Maksimum jumlah baris yang diizinkan untuk teks. Mirip dengan parameter <code>maxLines</code> di <code>Text</code> .

6. Kumpulkan laporan praktikum Anda berupa link repository GitHub kepada dosen!

Jawab: <https://github.com/margaretanp/Pemrograman-Mobile>