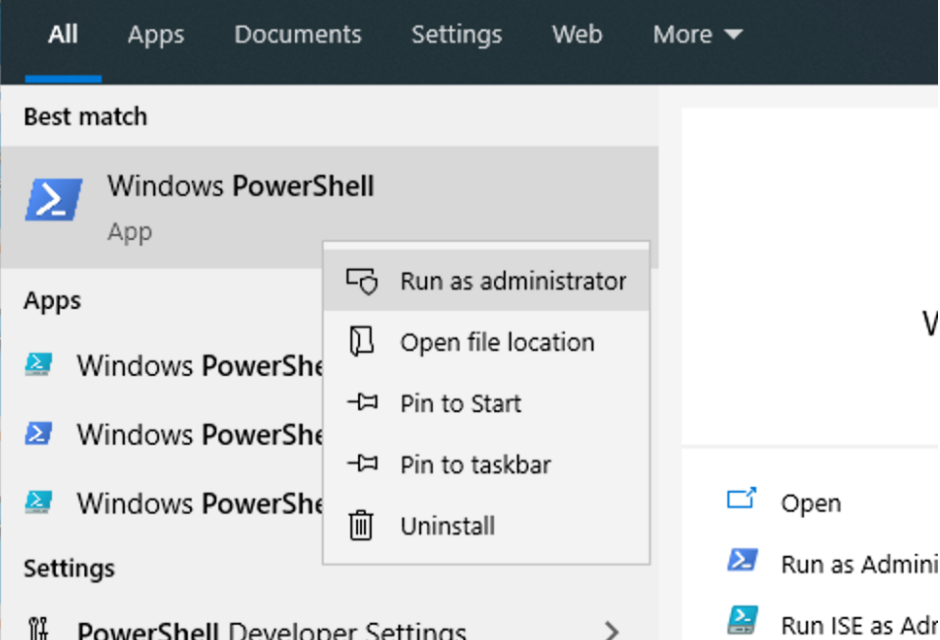
**Resume Video Tutorial yang ada di playlist**

1. **Chocolatey Package Manager Buat Windows**

[Chocolatey](https://chocolatey.org/) adalah sebuah package manager yang bisa melakukan instalasi software cukup dengan memberikan perintah via command prompt / powershell.

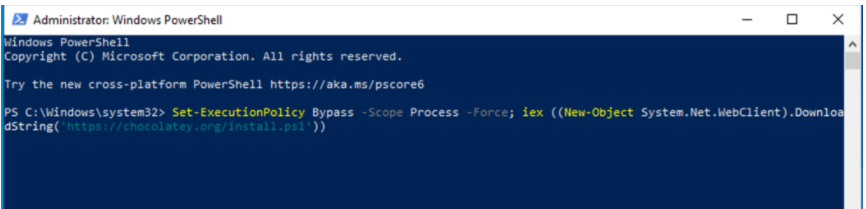
cara installnya:

1. Klik menu start dan cari Powershell kemudian klik kanan dan Run as administrator lalu klik Yes.



1. Kemudian setelah terbuka silahkan copas script berikut ke Powershell kemudian tekan Enter

-ExecutionPolicy Bypass -Scope Process -Force; iex ((New-Object System.Net.WebClient).DownloadString(‘https://chocolatey.org/install.ps1'))

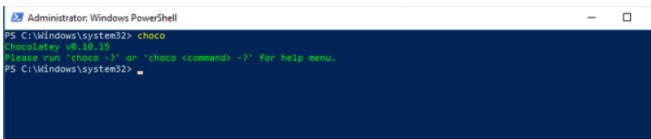


pastikan koneksi internet kamu lancar yah karena pada proses ini memerlukan koneksi internet untuk mengunduh si chocolatey.

1. Next step kita cek chocolatey-nya udah keinstall atau belum dengan cara mengetik perintah berikut pada command prompt atau powershell:

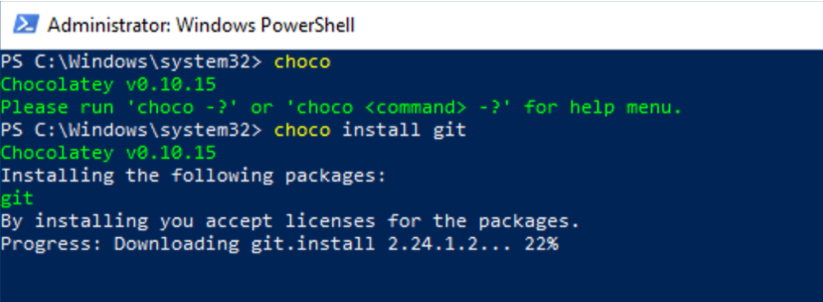
“choco”

Jika sudah terinstall maka akan muncul seperti pada gambar berikut



1. **Tutorial instal software menggunakan chocolatey**

Pada contoh kali ini instal GIT, untuk penginstalannya cukup mudah, tinggal ketik perintah ini di powershell



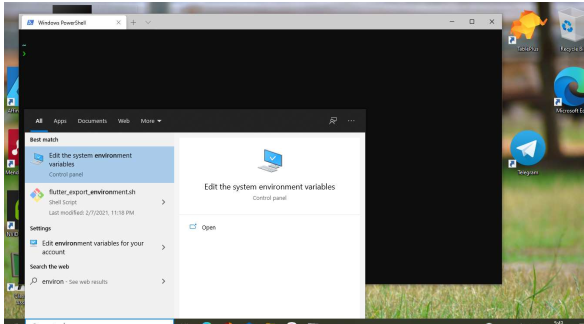
Tidak cuma buat install git, bisa juga install aplikasi lainnya seperti *virtualbox flutter* dll. untuk menggunakan perintah *choco*ini dibutuhkan hak akses administartor jadi pastikan selalu *run as administartor*pada saat membuka *powershell*atau pun *command prompt.*

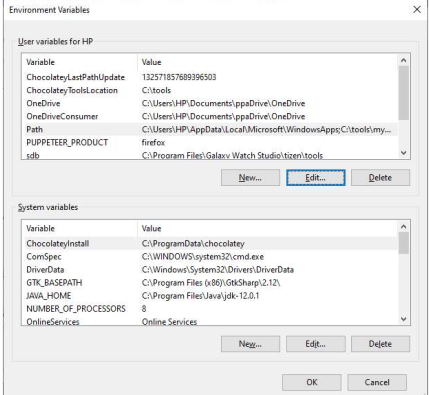
1. **Tutorial instal flutter**

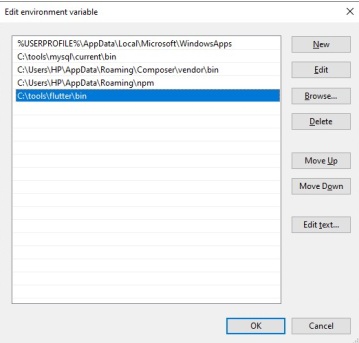
Alat dan bahan yang dibutuhkan untuk menginstall flutter :  
1. Sistem Operasi Windows minimal Windows 7 SP1 atau setelahnya 64 bit dan x86-64.  
2. Harddisk 1,32 GB untuk Flutter saja selain IDE atau alat lain  
3. Windows PowerShell   
4. Git For Windows

Langkah-langkah instal

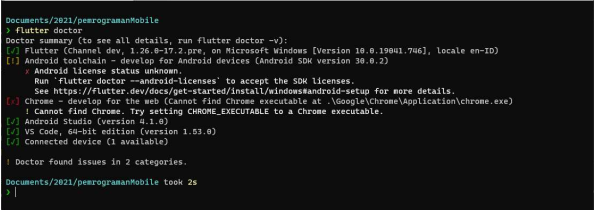
1. Download Flutter SDK
2. Extract file yang di download ke harddisk anda contoh lokasi ke C:\src\flutter (  
   JANGAN di install ke folder C:\Program Files\ karena membutuhkan akses admin )
3. Atau jika anda sudah menginstall git buatlah folder src di dalam drive C kemudian buka terminal di folder tersebut dan ketik  
   “git clone https://github.com/flutter/flutter.git -b stable”
4. Update Windows PATH tambahkan path menuju folder C:\src\flutter\bin



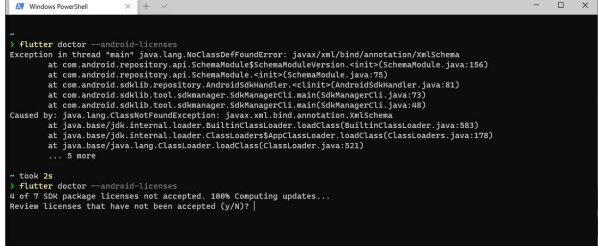




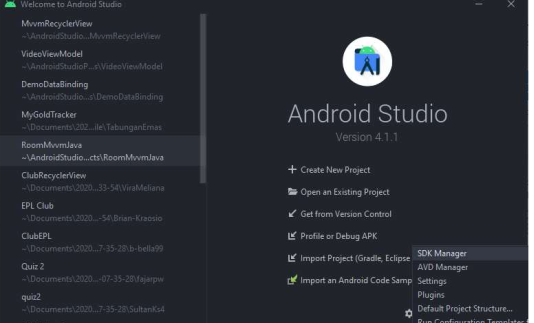
* 1. Install Android Studio
  2. Setup Android Device / Emulator
  3. Install Visual Studio Code <https://code.visualstudio.com/>
  4. Install Flutter Plugin
  5. Validasi Install dengan mengetikkan perintah berikut di terminal  
     “flutter doctor”
  6. Jika perintah flutter doctor mengeluarkan hasil seperti gambar di atas maka anda perlu menerima lisensi android SDK dengan mengetikkan perintah berikut ini di terminal.  
     “flutter doctor –android-licenses”

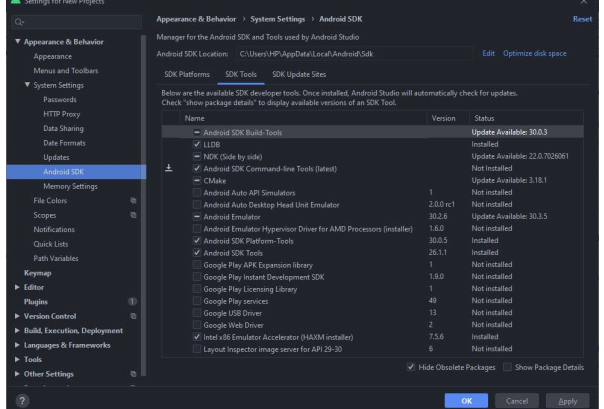


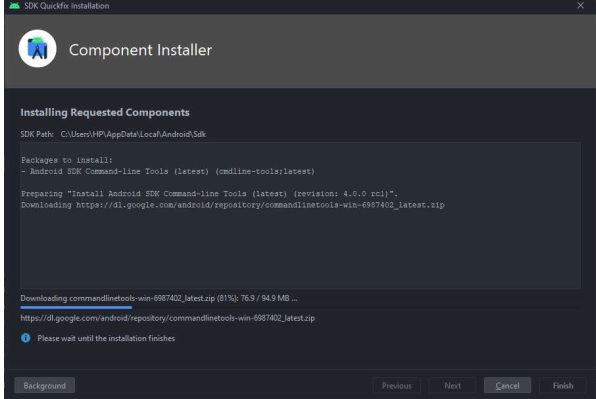
1. Jika tidak ada error lanjutkan ke langkah praktikum selanjutnya, jika muncul error seperti gambar dibawah ini.



1. Lakukan update di Android SDK pada Android Studio dengan menginstall Android Command Line Tools pada android SDK langkah melakukannya lihat gambar selanjutnya.







# Cara Mudah Publish Repository Git Lokal ke Github Langsung dari Visual Studio Code

# Mendaftar akun github ke <https://github.com/>

# Update konfigurasi git di komputer anda sesuai dengan username dan email yang anda buat di github.com.

# git config --global user.name usernameAnda git config --global user.email emailAnda

# Bukalah project yang anda buat sebelumnya menggunakan Visual Studio Code

# Jika anda belum menginisialisasi repository git pada project tersebut akan keluar tampilan seperti ini ketika anda mengklik icon source control di sebelah kiri.

# 

# Perhatikan ada dua tombol yaitu Initialize Repository dan Publish to Github, Klik lah tombol Initialize Repository. Sidebar akan berubah menjadi seperti berikut ini.

# 

# Klik dropdown changes ini sehingga daftar file yang berubah diminimize, kemudian klik kanan dan klik menu “Stage All Changes”

# 

# Setelah di klik git mencatat semua perubahan yang kita buat pada kode program dan sekarang semua file yang ada di Changes berpindah ke Dropdown Staged Changes

# 

# Isilah pesan tentang perubahan kode yang kita buat, pada kolom isian Message, kemudian klik tombol centang di atas.

# 

# Jika di klik menu “Staged Change” akan hilang dan semua perubahan sudah di catat oleh git.

# Selanjutnya untuk melakukan publikasi repository lokal ke github lakukan perintah berikut ini tekan Ctrl + Shift + P kemudian ketik publish to github

# 

# Visual studio code akan membuka browser dan meminta otorisasi untuk akun github

# 

# Setelah otorisasi berhasil anda dapat melanjutkan kembali command Ctrl + Shift + P dan pilih publish to github kemudian berilah nama yang sesuai

# Tutorial Hello World Flutter Bahasa Indonesia

# Flutter create nama projek hruf kecil/ underscore

# 

• .dart\_tools : Konfigurasi untuk dart language  
• .idea : Konfigurasi untuk android studio  
• gitignore : File git yang digunakan untuk mengelola source code. Hal ini akan  
berguna jika developer sudah bekerja dengan git.  
• metadata : File yang berisi metadata dari project  
• packages : File yang berisi alamat path  
• flutter\_basic.iml: File yang berisi detail dari project.  
• pubspec.lock : File yang berisi versi library atau package yang digunakan pada  
project yang degenerate sesuai dengan file pubspec.yaml.  
• pubspec.yaml : File yang berisi library atau package yang dibutuhkan untuk  
pengembangan aplikasi.  
• Readme.md : File markdown yang dapat digunakan untuk menjelaskan cara setup aplikasi atau informasi penting yang perlu untuk diketahui oleh  
developer lain

1. **Multi os ios dan android**pada folder project flutter terdapat dua folder yaitu folder ios dan folder  
   android, dengan menggunakan kedua folder tersebut flutter dapat membuat aplikasi berbasis ios dan berbasis android dalam satu project.
2. **Flutter Hot Reload**Pada flutter terdapat fungsi hot reload dan hot restart yang digunakan untuk  
   pengembangan aplikasi dengan flutter. Hote reload mencompile source code yang baru ditambahkan dan dikirimkan ke dart virtual machine diupdate. Setelah selesai update, dart virtual machine akan memperbarui UI sesuai dengan perubahan. Keunggulan hot reload adalah waktu prosenya cepat disbanding hot restart. Akan tetapi hot reload mempertahan state yang ada sehingga jika menggunakan state maka nilai dari widget tidak akan berubah.
3. **Flutter Hot Restart**Hot restart akan mencompile ulang aplikasi dan mereset (destroy) state yang ada. Jadi hot restart akan membuild ulang widget tree sesuai dengan code yang telah diperbarui.
4. **Import Statement**Seperti halnya kode program pada umumnya dart dapat menggunakan statement import untuk mengimport package, library, atau file lain yang digunakan pada file yang dieksekusi. import 'package:flutter/material.dart';
5. **Main function**Main function pada flutter dibuat dengan menggunakan kode program berikut ini dimana semua proses aplikasi dimulai dari mengeksekusi fungsi main.

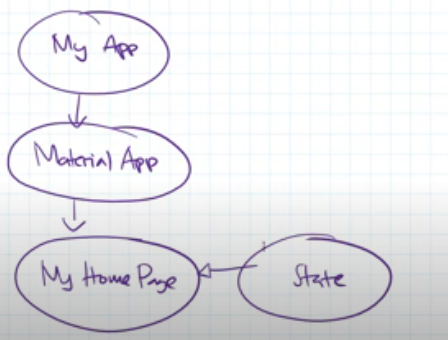
|  |
| --- |
| void main() { runApp(MyApp()); } |

1. **Stateless Widget**Stateless widget bersifat statis / final dimana nilai atau konfigurasi telah diinisiasi sejak awal, nilai variabel pada widget ini tidak dapat diubah oleh widget ini sendiri tetapi dapat diubah oleh parent widget nya jika parent nya adalah StatefullWidget.

|  |
| --- |
| class exampleStateless extends StatelessWidget{  @override  Widget build(BuildContext context) {  } } |

1. **Statefull Widget**Statefull widget bersifat dinamis, Perubahan data pada statefull widget di trigger oleh perubahan state oleh karena itu sebuah StatefullWidget selalu memiliki State. Struktur dasar statefull widget adalah sebagai berikut:

|  |
| --- |
| class exampleStateless extends StatefulWidget{  @override  State<StatefulWidget> createState() {  }  } |

1. 

# Konsep Widget Di Flutter

* Widget

1. Building blocks app
2. Stateless
3. Statefull
4. Widget tree
5. Base
6. Navigation
7. Button
8. Input
9. Dialog
10. Information
11. Layout

* Single
* Multiple
* **Scaffold**Scaffold widget digunakan untuk mengatur tata letak sesuai dengan material design.
* **Material app**

Digunakan untuk impelement material design

# Konsep Layout dan Proses Layout di Flutter

# 

# ****Container**** adalah widget itu sendiri termasuk child widget juga mempunyai kemampuan membuat style properties seperti padding, background warna dan lain sebagainya

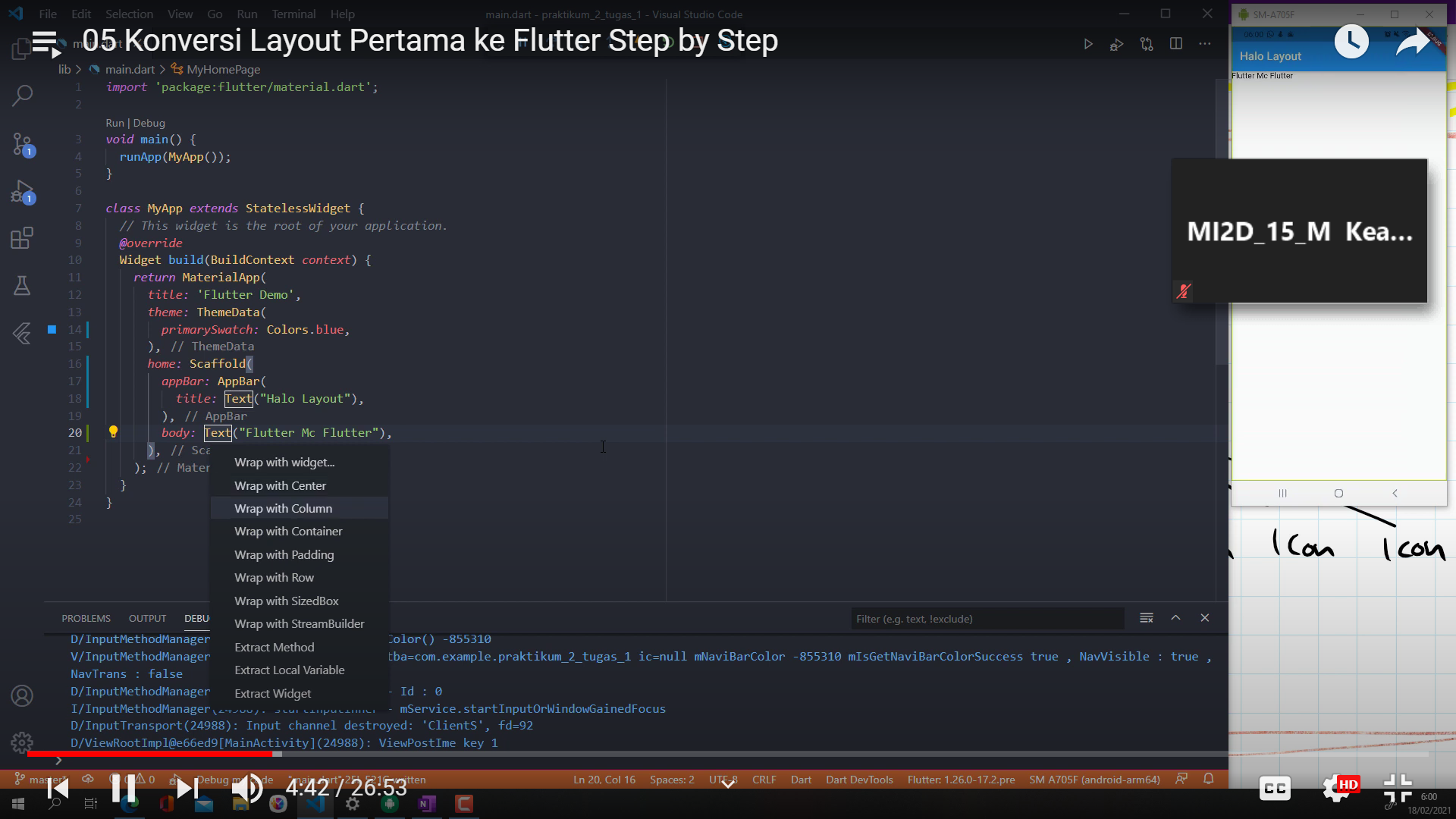
# Row dan Column Column widget digunakan untuk mangatur tata letak widget secara vertikal. Sedangkan row digunakan untuk mengatur tata letak widget secara horizontal.

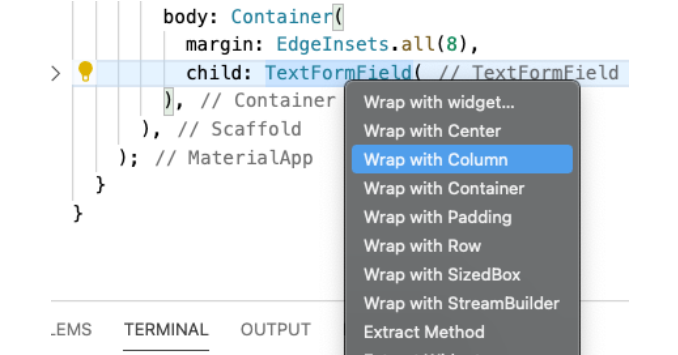
# Stack Stack Widget digunakan untuk menumpuk beberapa widget pada beberapa lapisan

**Image Widget**Image widget digunakan untuk menampilkan image.

**Text Widget**Text widget digunakan untuk menampilkan string yang dapat terdiri satu baris maupun beberapa baris

# Konversi Layout Pertama ke Flutter Step by Step



1. menambahkan margin pada container dan memberikan child berupa  
   sebuah InputForm pada container tersebut
2. walaupun TextFormField bukan merupakan parent widget pada widget tree yang kita butuhkan namun kita dapat melakukan wrapping dengan widget lain menggunakan shortcut Ctrl + .
3. 

# Cara Mudah Memahami State di Flutter

# State

# Apakah dapat berubah selama runtime ? state

# apa yang menyebabkan berubah ? event

# semua widgets yang terkait dengan variable state adalah render ulang

# Mini Project Flutter Konverter Suhu Part 1

# 

# menambahkan margin pada container dan memberikan child berupa sebuah InputForm pada container tersebut

# 

# walaupun TextFormField bukan merupakan parent widget pada widget tree yang kita butuhkan namun kita dapat melakukan wrapping dengan widget lain menggunakan shortcut Ctrl + .

# Lanjutkan dengan melakukan modifikasi pada TextFormField carilah properties agar a. Memiliki hint dengan text “Masukkan Suhu Dalam Celcius” b. Memiliki validasi hanya angka. c. Memiliki tampilan input keyboard khusus angka.

# 

# Wrap TextFormField kedalam sebuah kolom dengan menekan Ctrl + .

# 

# Langkah selanjutnya adalah mengisi Row untuk Text Info Suhu dalam Kelvin dan Reamur

# Langkah selanjutnya adalah mengisi sebuah button yang memiliki warna biru text putih dan even on click null.

# Konverter Suhu Part 2

# Menconnectkan event

# Set state

# Tambahkan logic

# Variable state

# 

# Set state

# 

# Menconnectkan event

# 

# Projects Konverter Suhu Part 3

# Extract widget

# Split jadi widget kecil

# Cara mengirim data dan event

# Named constructor

# Extract widget

# 

# Dipisahkan ke class yang baru

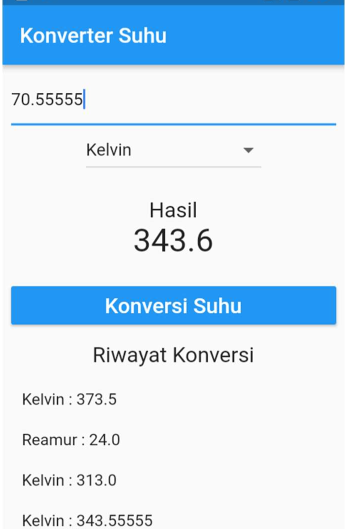
# 

# Import library di class yang baru dan di main.dart nya

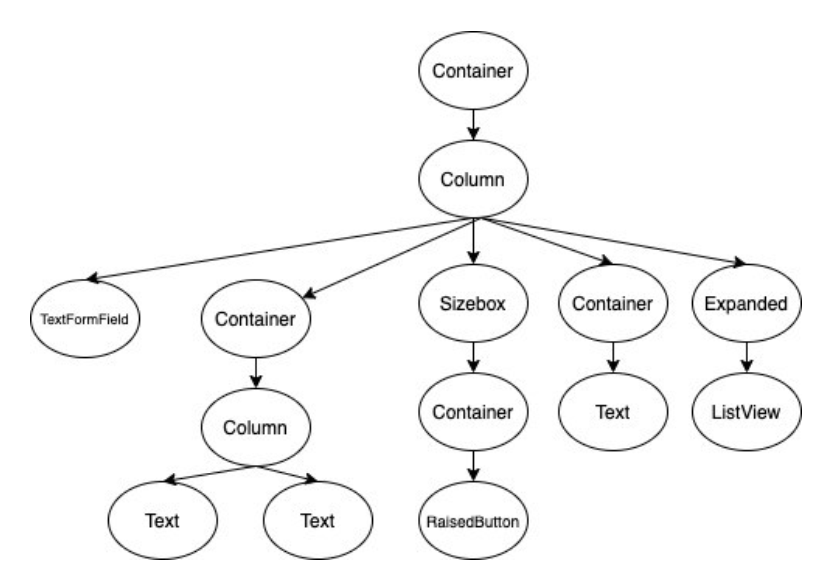
# 

# Flutter Intro Mini Project Konversi Suhu Versi 2

**Selanjutnya** belajar mengenai cara menggunakan array atau listuntukmembuat aplikasi dengan menggunakan flutter. Aplikasi yang dibuat akanmengembangkan aplikasi yang telah dibuat pada bab 3 yaitu untukmengkonversi suhu.



Widget tree yang dibuat tidak terbatas atau harus seperti pada gambar di bawah ini untuk membuat tampilan seperti mockup di atas.



# Mini Project Konversi Suhu Versi 2 Membuat Dropdown Button di Flutter

Dropdown Menu Button adalah sebuah menu yang menampilkan sebuah daftar data yang fungsinya untuk memilih suatu nilai data itu sendiri untuk di inputkan kedalam suatu data global.

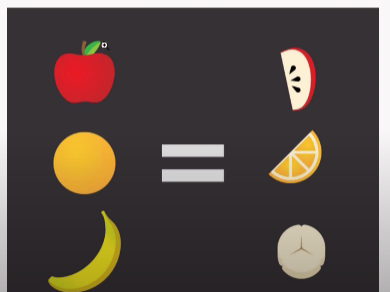
# Membuat dropdown

# 

DropdownButton<String> = String digunakan untuk memberi tipe data value dari Dropdown adalah bertipe String.  
listItem.map((String value) = Digunakan untuk melakukan iterasi untuk setiap item dari listItem sesuai dengan parameter bertipe String

# Flutter Mini Project Konversi Suhu Versi 2 Membuat dropdown dinamis dengan map

# Array Map adalah suatu proses mengubah bentuk data menjadi bentuk data yang baru



listItem.map((String value) = Digunakan untuk melakukan iterasi untuk setiap item dari listItem sesuai dengan parameter bertipe String

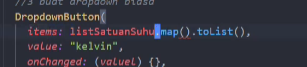
* membuat array datanya



Kemudian diubah menjadi widget



Kemudian disusun menjadi list



# Flutter Mini Project Konversi Suhu Versi 2 Menyambungkan State ke Dropdown Button

# Membuat variable baru kemudian menjadi value dari selecteddropdown

# 

# menSetState

# 

# Flutter Mini Project Konversi Suhu Versi 2 Mengkonversi Result Widget

# Mengirimkan resultnya

# 

# Merubah yang awalnya tiga menjadi Satu

# 

# Dan merubah layoutnya dengan cara menghapus mainaxis.aligment.space

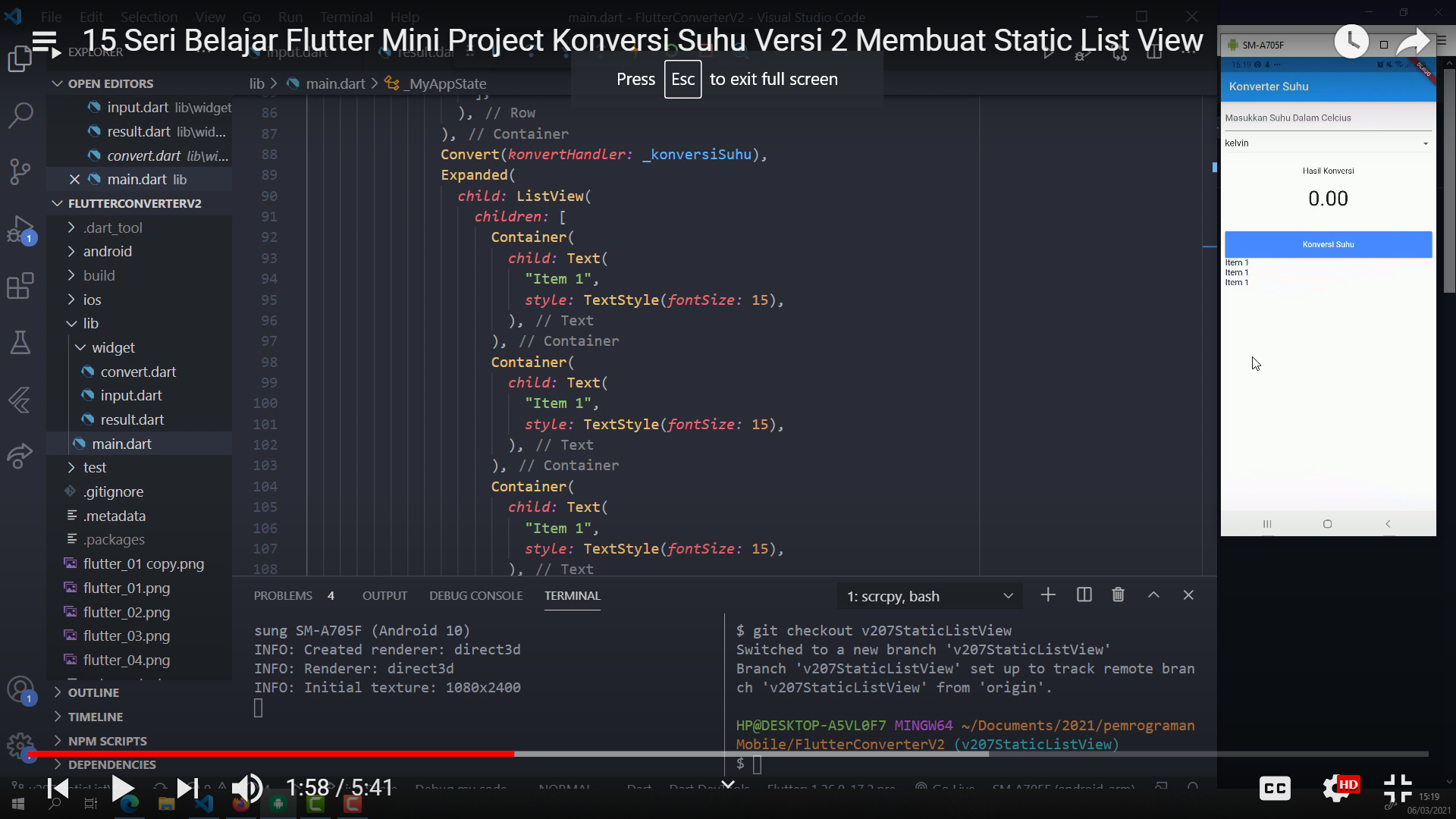
# 

# Layoutnya menjadi naik keatas

# 

# Flutter Mini Project Konversi Suhu Versi 2 Membuat Static List View

Membuat widget list view yang memiliki child yang akan menjadi item-item list view



# Flutter Mini Project Konversi Suhu Versi 2 Menambahkan Logika Aplikasi

# Membuat switch case

# 

# Flutter Mini Project Konversi Suhu Versi 2 Menambahkan ListView Dengan Map

# Membuat variable bertipe List<String> dengan menggunakan code berikut “List<String> listViewItem = List<String>();”

# Mengisi listViewItem setiap kali terjadi proses konversi

# Menampilkan hasil listViewItem menggunakan map, menggunakan code berikut.

|  |
| --- |
| listViewItem.map((String value) {  return Container(  margin: EdgeInsets.all(10),  child: Text(  value,  style: TextStyle(fontSize: 15),  ));  }).toList() |

# Menjadikan widget yang lebih kecil dengan menggunakan extract widget. Hasil akhir setelah diextract widget adalah seperti berikut.

|  |
| --- |
| body: Container(  margin: EdgeInsets.all(8),  child: Column(  children: [  Input(inputController: inputController),  DropdownKonversi(listItem: listItem, newValue: \_newValue,  dropdownOnChanged : dropdownOnChanged),  Result(result: \_result),  Convert(konvertHandler: perhitunganSuhu),  Container(  margin: EdgeInsets.only(top: 10, bottom: 10),  child: Text(  "Riwayat Konversi",  style: TextStyle(fontSize: 20),  ),  ),  Expanded(  child: RiwayatKonversi(listViewItem: listViewItem),  ),  ],  ),  ), |

# Flutter Mini Project Konversi Suhu Versi 2 ListView dengan ListView Builder

### **ListView.builder**

Untuk list yang bersifat dinamis ( jumlah list item mengikuti dari jumlah data ) maka gunakan ListView.builder.

ListView.builder memiliki dua properti utama yaitu **itemCount** (jumlah list item) dan **itemBuilder** (untuk membangun tampilan dari list item). Secara sederhana penggunaan kode listview.builder sebagai berikut

ListView.builder(

itemCount: itemCount,

itemBuilder: (context, position) {

return listItem();

},

),

Penggunaan ListView juga biasanya digabungnya dengan **ListTile Widget**. Widget ini umumnya digunakan untuk membuat kontak list atau layout yang membutuhkan thumbnail beserta subtitle.

Contoh penerapan listview builder

# 