**Flutter**

**Catatan code belajar Flutter.**

## Pengenalan Flutter

Flutter adalah SDK (Software Development Kit) yang dibuat oleh Google untuk membuat aplikasi di android dan ios.

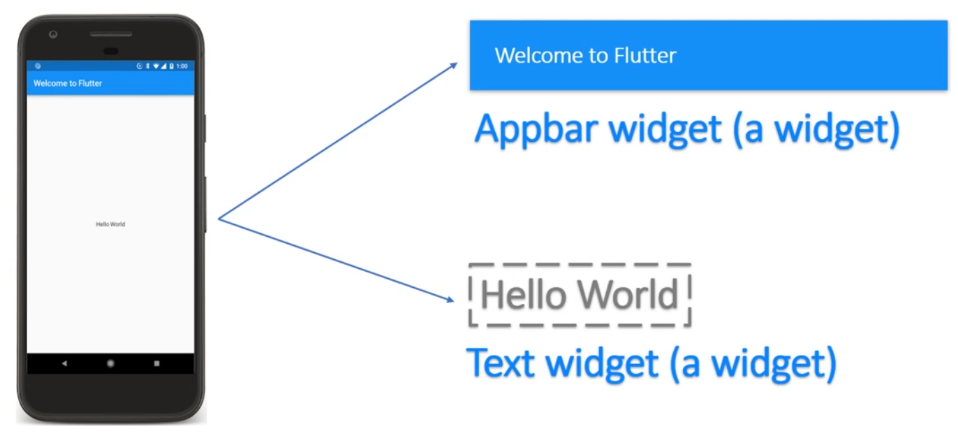
SDK, adalah sekumpulan fungsi fungsi, atau class, atau alat lainnya yang diperlukan oleh seorang programmer untuk membuat sebuah aplikasi.

Kelebihan Flutter :

1. Mengembangkan aplikasi ios atau android dengan lebih cepat, karena sudah disediakan widget / class yang diperlukan untuk membuat aplikasi.
2. Fitur hot reload, melihat perubahan yang dilakukan tanpa mengcompile ulang program.
3. Membangun sebuah aplikasi dengan ui yang flexsibel, ekspresif dan baik. Memiliki widget widget yang mendukung material design dan cuppertino design (design yang dipakai oleh ios).
4. Native performance, performance yang sesuai dengan platformnya (ios / android) masing masing.

Material Design, adalah sebuah Bahasa visual yang menggabungkan antara prinsip prinsip design yang baik dengan Teknologi dan science yang sudah ada saat ini. Material design terispirasi dari metafora / dunia nyata jadi terdapat teksture, cahaya, shadow, dll.

Widget, dapat diumpamakan seperti benda di dunia nyata.

Setiap widget terdiri dari widget dengan widget widget lainnya.

## Flutter Pertamaku.

***Flutter create nama\_project*** = membuat project baru menggunakan command.

Masuk ke dalam main.dart, kemudian hapus semua code sisakan 2 baris code berikut.

A screen shot of a computer

Description automatically generated

Void main() itu adalah fungsi yang akan dijalankan pertama kali.

Method runApp() adalah fungsi yang akan menjalankan class dari MyApp.

Untuk membuat class MyApp, dapat dengan menggunakan shortcut ***stl***.

Widget terdapat 2 macam, yaitu statelessWidget dan statefullWidget.

A screen shot of a computer program

Description automatically generated

@override Widget build(BuildContext context)

Metode ini bertugas untuk mengembalikan widget yang akan dirender oleh Flutter.

MaterialApp itu adalah sebuah widget yang berisi data data yang diperlukan oleh aplikasi yang menggunakan material design.

Selanjutnya kita panggil property dari MaterialApp yaitu *home*, ini adalah property tampilan awal yang akan ditampilkan oleh widget material app. Material App sebenarnya masih kosong / belum berisi apa apa.

home : Scaffold()

Scaffold() ini adalah tampilan dasar dari sebuah aplikasi android, dan memiliki banyak property.

appBar : untuk judul, biasanya ada widget khusus untuk yaitu AppBar().

Dan sama di dalamnya memiliki propertynya masing masing.

body : untuk body.

A screen shot of a computer code

Description automatically generatedA white background with black text

Description automatically generated

## Text Widget

Kita akan bungkus widget Text dengan widget Container.

Fungsi widget Container untuk membungkus widget lain, sehingga kita bisa memberikan padding, margin, gradient, dll.

Parameter Text Widget

maxLines: 2, = menamplikan text maksimal 2 baris.

overflow: TextOverflow.ellipsis = menampilkan text jika terlalu Panjang akan diganti dengan …

softWrap: false, = menampilkan text jika terlalu Panjang maka akan hidden tidak diturunkan ke bawah.

textAlign: TextAlign.center = mengatur posisi text.

style: TextStyle() = mengatur style pada text

contohnya

A screen shot of a computer program

Description automatically generated

## Row & Column Widget

Widget Column ini digunakan untuk mengatur tata letak widget yang ada di dalam column tersebut. Widget ini dapat terdiri beberapa widget lagi yang tata letaknya ini akan disusun ke bawah secara vertical.

A white background with black text

Description automatically generatedA screen shot of a computer program

Description automatically generated

Jika Widget Row ini digunakan untuk mengatur tata letak widget yang ada di dalam row tersebut secara horizontal.

A white background with black text

Description automatically generated

A diagram of a traffic light

Description automatically generatedProperty pada widget row dan column

Column

Property mainAxisAlignment dan crossAxisAlignment pada column.

A diagram of a diagram

Description automatically generated with medium confidenceRow

Property mainAxisAlignment dan crossAxisAlignment pada row.

## Container Widget

Kita dapat mengatur color, margin dan padding pada container.

A screen shot of a computer screen

Description automatically generated

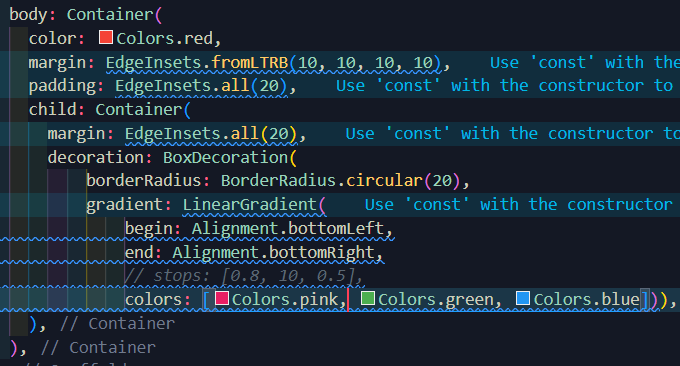
Untuk mengatur padding dan padding dapat kita gunakan Widget EdgeInsets.

Untuk mengatur ukuran secara spesifik kita dapat kita atur :

.fromLTRB() = mengatur secara spesifik (Left, Top, Right, Bottom)

.only(bottom : 20) = mengatur hanya bagian tertentunya saja

.all() = mengatur semua ukurannya sama



Selain bisa mengatur margin, padding dan color, kitab isa memberikan decoration untuk styling container dengan widget tertentu misalkan dengan memanggil Widget BoxDecoration kita bisa mengatur dekorasi pada box containernya seperti mengatur gradient, borderRadius dan lain sebagainya.

## Stateless vs Statefull Widget

Widget di flutter dibagi 2, yaitu :

* Stateless Widget

Widget yang tidak memiliki state / keadaan.

Misalkan kita punya appBar aplikasi dan di dalamnya ada title aplikasi, logo aplikasi dan itu tidak pernah berubah ini disebut stateless karena tidak pernah berubah ubah.

* Statefull Widget

Widget yang memiliki state.

Sebagai contoh adalah Air :

Air , memiliki state / keadaan yaitu (Cair, Padat, Uap)

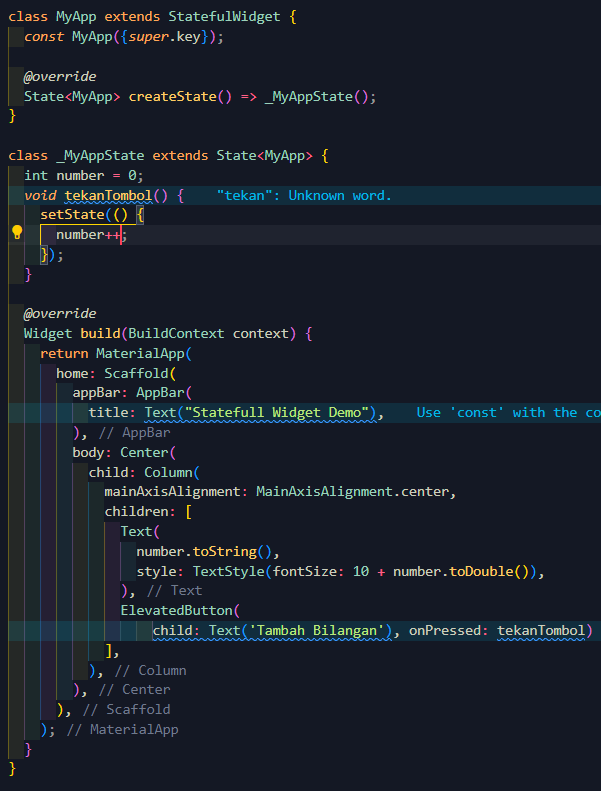
Atau misalkan kita punya bilangan Ketika di klik ia bertambah menjadi 2 atau 3 & seterusnya.

Atau button asalnya biru Ketika di klik jadi merah.

Untuk Membuat Statefull widget cukup ketikan stfl (Shortcut Statefull Widget).

Statefull widget memiliki 2 buah class,

yang 1 class statefull widgetnya bertujuan untuk mendefinisikan dan membuat state / keadaan, dan yang kedua berisi class statenya yang terjadi pada tamplan aplikasinya.



Disini kita mendefinisikan bahwa number = 0

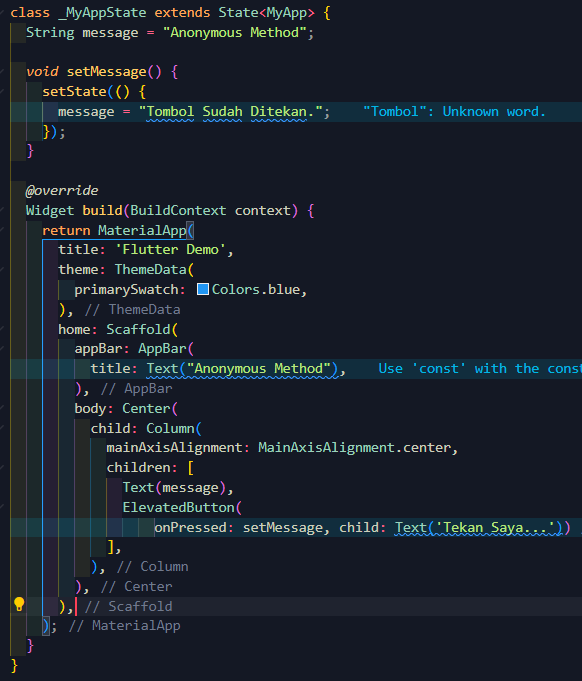
Lalu panggil ***void*** *(tipe data yang menentukan suatu fungsi mengembalikan data atau tidak).*

setState() berfungsi untuk mengubah tampilan yang ada saat ini. Jika tidak menggunakan setState maka tidak akan ada perubahan yang terjadi pada tampilan kita.

## Anonymous Method

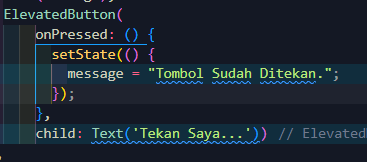
Anonymous Method adalah method yang tidak memiliki nama. Kita menggunakan anonymous method Ketika kita menekan widget tersebut 1x saja, maksudnya tidak akan pernah memanggil methodnya di tempat lain.

Sebagai contoh, bahwa kita akan mnampilkan ‘Tombol Sudah Ditekan.’ Ditempat lain hanya di tombol itu saja, oleh karena itu kita bisa menyingkat code nya dengan anonymous method.



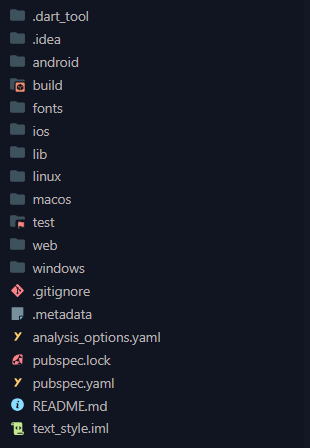
Perlu diingat anonymous method hanya digunakan disitu tanpa digunakan ditempat lain. Dengan anonymous method kita dapat mempersingkat penulisan code, tanpa harus mendeklarasikannya diatas.

Dapat kita sisipkan kedalam onPressed seperti ini dengan menuliskan () { setState(){…} }



## Text Style Widget

Menambahkan jenis font lain dapat kita lakukan seperti ini.



Buat folder baru dengan nama Fonts dan simpan semua font ke dalamnya, karena ini hanya fonts jadi tidak apa apa dibuat folder fonts.

Tapi jika sudah banyak adanya image, font, dll masukkann ke dalam folder assets

Jika sudah ubah pubspec.yaml

A screen shot of a computer

Description automatically generated

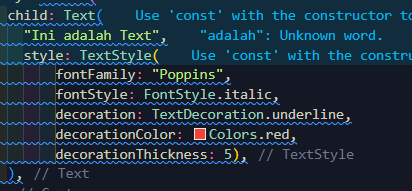
Jika sudah maka kita bisa mengubah font family nya dengan yang sudah kita tambahkan.

A screenshot of a computer program

Description automatically generated

Tidak hanya itu kita bisa merubah styling pada text seperti contoh berikut.

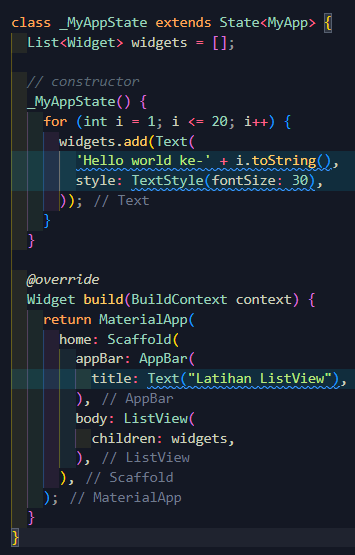
A white background with red text

Description automatically generated

## List & List View Widget

Jika kita menggunakan Column lalu menampilkan datanya tidak cukup maka akan tampil error. Sedangkan jika list view tidak akan menghasilkan error tapi akan menghasilkan scroll down atau scroll up.

Untuk membuat list kita dapat melakukannya seperti berikut

List dapat kita anggap sebagai tipe data

Misalkan kita buat List<int> nama\_variabel = [];

Maka list ini akan diisikan oleh nilai yang tipe datanya integer.

Dengan membuat constructor maka kita nantinya akan mengirikan ListView dengan data yang sudah dibuat di dalam Constru

Membuat Counter Menambah Data, dan Menghapus Data

A screen shot of a computer program

Description automatically generatedYang pertama definisikan bahwa List akan diisikan oleh widget.

Lalu set counter sebagai nilai awal

Ketika button di klik ubah keadaan dengan menambahkan text dan menambahkan nilai dari counternya.

## Animated Container Widget

Animated Container hamper sama dengan Container hanya saja bedanya di dalam animated Container ini diubah, misalkan tingginya, lebarnya, warnanya. Semuanya akan dianimasikan secara otomatis.

A computer screen shot of a program code

Description automatically generated

Random random = new Random();

Random digunakan untuk memanggil class Random dari dart, kemudian buat nama variable lalu inisialisasikan untuk membuat object random baru.

Animated Container Widget dapat kita gunakan untuk melakukan animasi / perubahan pada container kita disini kita akan set menggunakan Random <int> agar Animated Container ini dapat berjalan kita bungkus di dalam Widget GestureDetector.

GestureDetector Ketika ada gestur dengan aksinya tap. Maka jalankan setState() untuk melihat perubahan animasi containernya secara otomatis Ketika container itu di klik.

## Flexible Widget (Responsive Widget)

Membuat tampilan pada container widget kita menjadi responsive terhadap ukuran device.

A colorful squares in a row

Description automatically generatedUkuran Mobile Phone

Ukuran Tablet

A group of colorful rectangles

Description automatically generated

Dapat kita bungkus menggunakan Flexible Widget, lalu kita beri property flex

Property flex beguna untuk membagi bagian, misalkan ada 3 buah container yang dibungkus dengan property flexible dengan flex: 1

Maka setiap ukuran container akan berukuran 1/3 dari total flex dan membaginya dengan ukuran layar.

## Stack & Align Widget

Column Menyusun widget ke bawah, Row Menyusun widget kesamping.

Stack menumpukkan widget / layering widget.

Dengan menumpukkan stack dapat kita lakukan hasilnya seperti ini

A screenshot of a cell phone

Description automatically generatedDi paling belakang ada background seperti catur.

Lalu diatas nya ada listview yang berisi text text

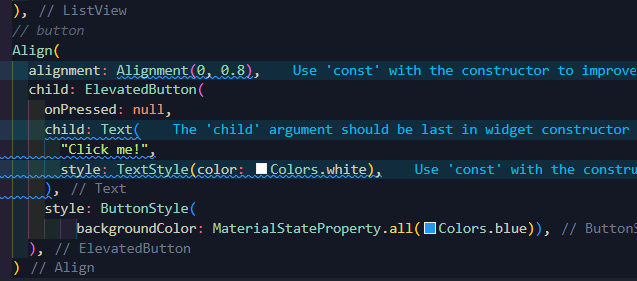
Kemdian diatas list view terdapat button yang diatur menggunakan align.

Maka didalam body kita awali dengan Stack kemudian childrennya dapat kita isi dengan Column

A screen shot of a computer

Description automatically generated

Kemudian button kita simpan setelah List view yang akan ditumpuk diatas list view



Untuk menggunakan alignment widget terdapat property widget tambahan yang dapat kita pilih seperti Alignment.bottomCenter()

Namun disini kita dapat atur juga dengan mengcustomized dari alignment nya

Alignment(0, 0.8)

Nilai x = -1 ke kiri / atas, tengah 0 dan y = 1 bawah / kanan.

## Images Widget

Image widget ada 2 dapat kita gunakan

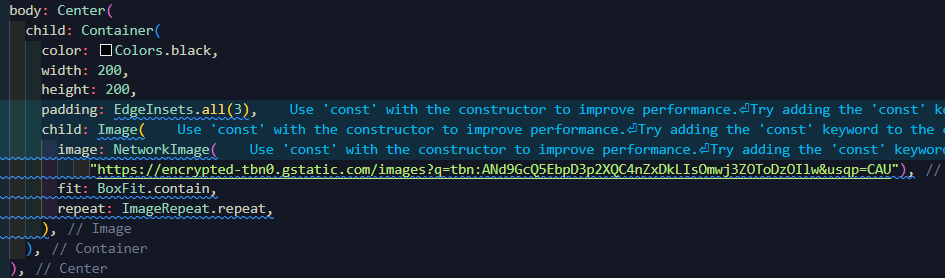
1. Kita bisa gunakan Widget NetworkImage

Image(image: NetworkImage(

url,

fit: BoxFit.cover,

repeat: ImageRepeat.repeat))



1. Cara ke 2 kita bisa menambahkan image dari local atau menggunakan assets yang kita miliki.

Sebelumnya kita simpan file ke dalam folder di dalam aplikasi, lalu daftarkan di pubspec.yml

Kemudian kita gunakan widget AssetImage

A screen shot of a computer program

Description automatically generated

## Spacer Widget

A group of squares in different colors

Description automatically generatedAda space kosong di dalamnya karena menggunakan space evenly, ada juga yang lain seperti space arround dan space between.

Bagaimana menambahkan ukuran space kosong yang berbeda ?

Kita menggunakan SpacerWidget()

A screen shot of a computer program

Description automatically generatedJika kita membuat code seperti ini maka hasilnya akan mirip seperti dengan kita menggunakan spaceBetween.

Karena spacernya disimpan diantara 1 & 2, 2 & 3.

A screenshot of a computer program

Description automatically generatedJika ditambahkan flex maka ukurannya akan dihitung bedasarkan total jumlahnya.

Maka jarak pertama akan 1/6

Jarak ke 2 akan 2/6

Dan yang ke 3 3/6

dari total keseluruhan.

A green square with white border

Description automatically generated

## [PR] - Draggable, DragTarget, SizedBox, Material Widget.

Drag, membuat fitur seperti layaknya kita melakukan drag and drop.

SizedBox digunakan untuk membungkus widget yang tidak memiliki property width dan height supaya dia bisa memiliki width dan height.

Sebelumnya kita bungkus ke dalam Dragable<Widget yang di Drag>()

data: data apa yang di drag. Misalkan : color 1

child : bentuk awal yang akan di drag.

childWhenDragging : bentuk sebelumnya yang terjadi ketika sedang di drag.

Feedback : bentuk yang sedang di drag

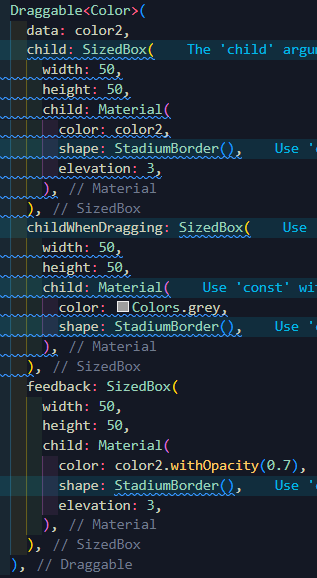
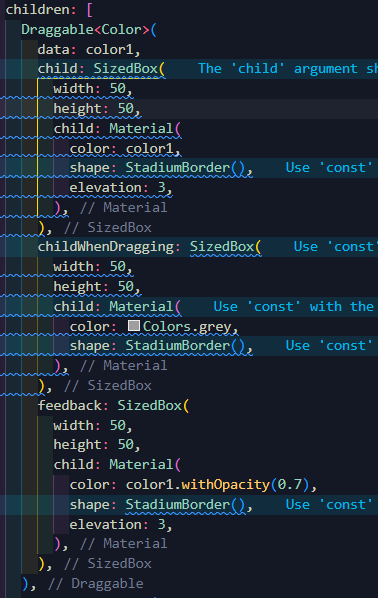
Widget Material bisa memiliki

Shape: Widget StadiumBorder bukan untuk membuat lingkaran, tapi membuat ujungnya kotak menjadi melengkung.

Elevation: untuk mengatur ketinggian sehingga menghasilkan shadow

Dragable

Color 1 Color 2



Selanjutnya Tempat dragnya kemana

DragTarget<Color>

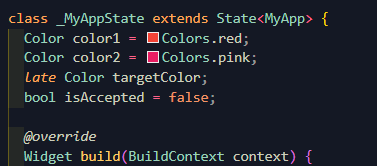
onWillAccept: (value) => true, = apakah ini dapat menerima drag yang terjadi.

onAccept: (value) {isAccepted = true;} = fungsi yang terjadi, apakah bisa menerima warna

builder: (3 parameter yaitu, context, candidates, rejected) {

(isAccepted) ?   
}

= sebuah widget yang digunakan untuk membuat drag target ini.





Hasil aplikasi

A group of red dots

Description automatically generated

## Navigation MultiPage Screen

Kita akan membuat Login Page > Main Page > Second Page, Ketika misalnya sudah login maka tidak akan bisa kembali ke login page, tapi langsung ke main page.

Konsep Page / Screen = Route (diletakkan di sebuah stack).

Jadi ketika kita punya halaman Login akan disimpan di stack paling utama. Berikutnya akan tampilkan main page maka stack login page akan dibuang dan diganti dengan main page. Kita akan menggunakan Push Replacement.

Lalu nanti second page akan dipush menimpa main page. Nanti kita akan gunakan Push untuk menaruh tumpukannya, misalnya ketika kita klik back, maka seccond page akan menghilang. Akan menggunakan POP

Push Replacement = Menimpa stack dan tidak bisa kembali

Push = Menimpa stack

Pop = Menghilangkan stack.

Yang memanagement push, pop disebutnya Navigator.

Kita tidak akan membuat langsung Material App, lalu Scaffold.

Pastikan untuk membuat setiap page itu terpisah, dan diawali dengan scaffold

A screen shot of a computer program

Description automatically generatedDisini kita hanya membangun dengan Widget material App saja, karena agar di page lain kita bisa langsung menggunakan Scaffold.

A computer screen shot of a program code

Description automatically generatedKetika button di klik jalankan Navigator dengan push replacement, sehingga tidak bisa balik ke login page.

Dan buat Route dengan MaterialPageRoute dengan mengarah ke Main Page()

A screen shot of a computer program

Description automatically generated

Main page kita lakukan hal yang sama namun denga menggunakan Navigator.push, agar route ini masih bisa diakses.

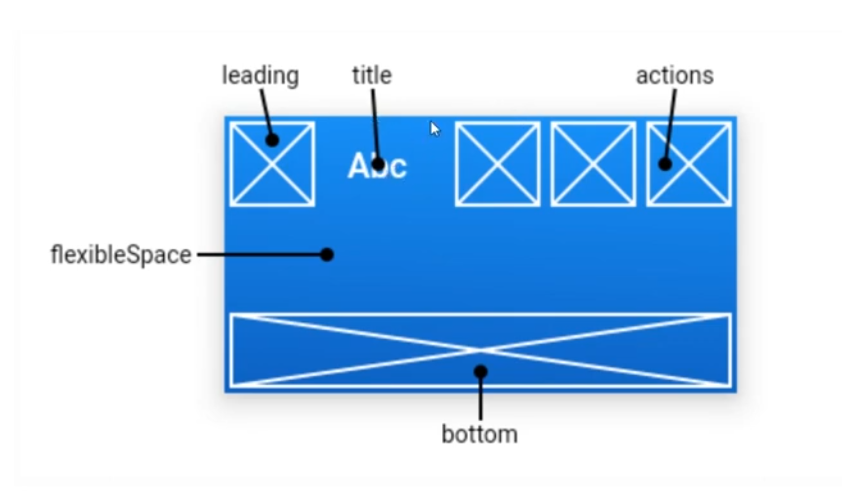
A computer screen shot of a program code

Description automatically generatedDan untuk second page, cukup dengan menggunakan Navigator.pop

Context dapat diibaratkan seperti, di halaman ini button ketika di klik jalankan widget navigator.pop untuk menghilangkan tumpukan stack, jadi stack dapat diibaratkan seperti stack nya.

## AppBar Gradasi dengan Corak

Membuat Gradasi warna pada appbar, dan memberikan Corak agar appbar tidak terlalu polos.



A screen shot of a computer

Description automatically generated A screen shot of a computer program

Description automatically generated

Berikut contoh pembuatan gradasi & menambahkan icon pada AppBar.

Untuk icon kita gunakan leading dan widget Icon.

Untuk action kita gunakan action dan gunakan widget Icon button agar icon dapat di klik, dan berikan icon nya.

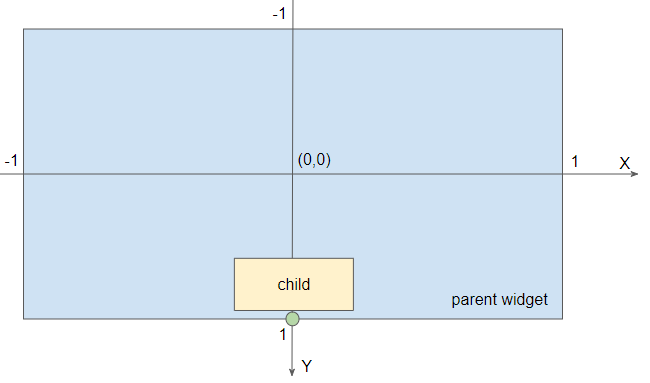
Lalu untuk gradasi kita gunakan flexibleSpace dan bungkus ke dalam widget Container

Lalu berikan Box Decoration linear gradient, dan untuk begin: dan end: gunakan FractionalOffset widget

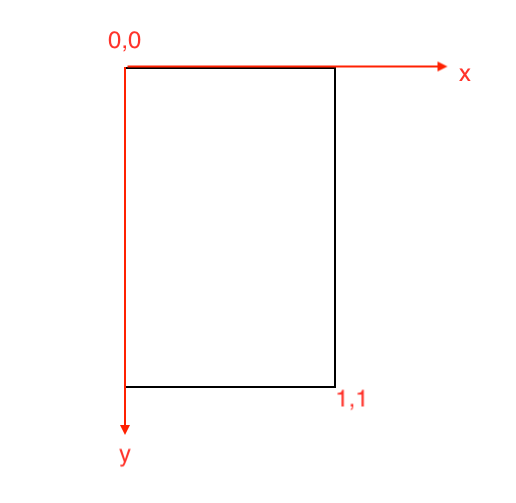
### Alignment vs FractionalOffset

Jika Alignment digunakan untuk menentukan posisi relatif sebuah widget dalam ruang yang lebih besar (misalnya, dalam sebuah container).

Koordinat Alignment dinyatakan dalam rentang dari -1 hingga 1 untuk sumbu x (horizontal) dan sumbu y (vertikal).



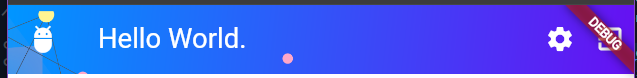
FractionalOffset digunakan untuk menentukan posisi relatif sebuah widget dengan menggunakan koordinat fraksional relatif terhadap ukuran widget itu sendiri.

Koordinat FractionalOffset juga dinyatakan dalam rentang dari 0 hingga 1, di mana (0,0) adalah sudut kiri atas widget dan (1,1) adalah sudut kanan bawah widget.

A computer screen shot of a program code

Description automatically generated

Untuk imagenya dapat kita simpan di dalam BoxDecorationnya.

Maka hasilnya seperti ini 

## Card Widget

A screenshot of a computer program

Description automatically generatedBiasanya digunakan untuk mengelopokkan, data yang saling berkaitan.

Asalnya kita buat Card langsung di dalam children yang ada di dalam ListView lalu agar tidak melakukan copas terhadap card, kita lakukan refactor extract method

Maka akan dibuatkan method baru khusus untuk membuat card, dengan menerima parameter.

A screenshot of a phone

Description automatically generated

## TextField Widget

Meminta inputan dengan menggunakan TextField. Untuk Text Field kita membutuhkan Controller, untuk mengambil nilai dari text fieldnya menggunakan controller juga.

Obscure text digunakan untuk input password (boolean).

maxLength: digunakan untuk membatasi inputan.

maxLines jika sudah max tidak bisa menambah baris baru.

A screen shot of a computer program

Description automatically generatedKita buat terlebih dahulu controller. Karena data inputan akan ditangkap menggunakan controller dan dikembalikan oleh controller.

Untuk membuat realtime perubahan dapat kita gunakan onChanged (value) adalah setiap inputan yang ditangkap ke controller, lalu jalankan setState untuk perubahan secara langsung.

## TextField Decoration

Property decoration untuk menghias Text Field.

Border:

* InputBorder.none = menghilangkan border garis bawah
* OutlineInputBorder() = memberikan border outline

Icon, memberikan Icon diluar border, menggunakan property icon: dan berikan widget Icon()

Untuk memberikan icon di dalam field menggunakan prefixIcon.

Ada juga prefixText untuk memberikan text di dalam input.

Untuk PrefixStyle sama sepetri TextStyle widget

LabelText untuk memberikan title pada inputan, dapat kita ubah menggunakan labelStyle. Sama seperti kita menggunakan TextStyle

Ada juga hintText memiliki fungsi seperti placeholder pada inputan, dan dapat kita style menggunakan hintStyle menggunakan widget TextStyle.

Ada juga property prefix: dan ini akan berupa widget jika kita menggunakan prefix tidak boleh menggunakan property prefixx yang lain.

Suffix: mirip seperti prefix: ada juga suffix style, suffix text dan suffix icon. Semuanya sama hanya saja letaknya berada diakhir / kanan.

fillColor: untuk memberikan warna, filled: true, agar warna yang diberikan diterapkan di field input.

A computer screen shot of a program

Description automatically generated

A rectangular object with a white background

Description automatically generated

## MediaQuery Widget (Responsive App)

Mengubah ukuran container dengan ukuran layar.

MediaQuery.of(context).size.width = ini adalah lebar dari layarnya.

Kita bisa mengatur responsive layarnya dengan mediaQuery dan mebagi nya.

Namun bagaimana dengan Landscape?

A screen shot of a computer program

Description automatically generated Sebelumnya kita bungkus container ke dalam method terpisah, lalu di body kita lakukan pengecekkan.

Jika MediaQuery.of(context) orientation == Orientation .potrait

Maka jika potrait tampilkan dalam bentuk Column sedangkan jika Landscape tampilkan dalam bentuk Landscape.

## InkWell Widget (Customize Button)

Membuat button sendiri dengan menggunakan widget InkWell.

Kita bisa menggunakan Elevated Button untuk membuat button, tapi hanya bisa memberikan warna pada button tersebut & tidak bisa memberikan warna gradasi / background.

Untuk mengatasinya kita gunakan Inkwell

Inkwell itu sebenarnya seperti gelombang air ketika button di tekan.

Inkwell akan berjalan diatas widget material.

*Untuk memberikan bayangan pada button diberikannya elevation tapi Container kita bungkus lagi menggunakan material widget.*

*A white background with black and white clouds

Description automatically generated* *A white background with black and white clouds

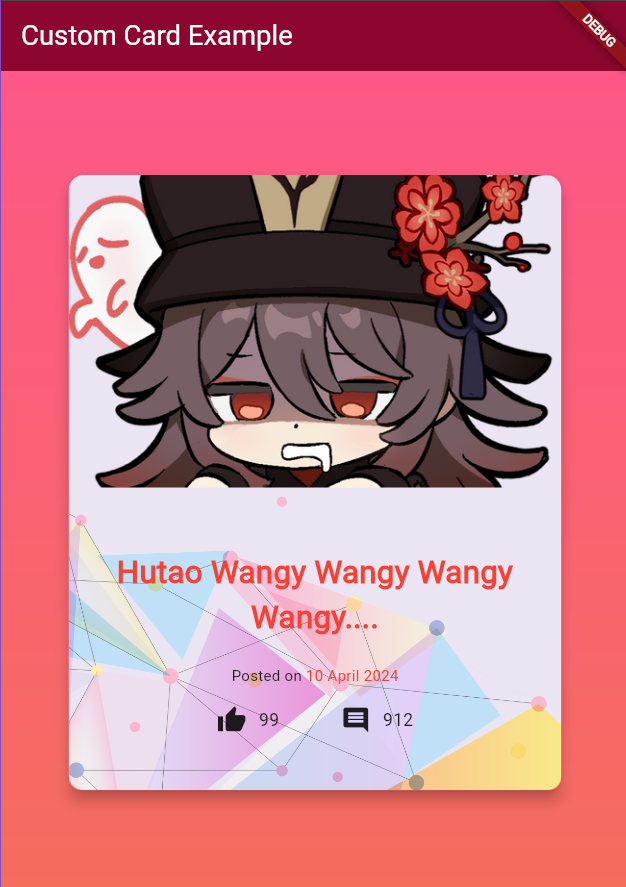
Description automatically generatedA computer screen shot of a program

Description automatically generated*

## [Latihan] - Opacity (Custom Card & Corak)

SizedBox, digunakan untuk menentukan ukuran tetap (lebar dan tinggi) dari ruang yang tersedia.

Container, digunakan untuk mengatur tata letak, penempatan, dan dekorasi widget.



Stack

* Container (untuk background awal / bagian belakang)
* Center (Agar Konten semuaditengah)
* SizedBox (Mengatur Ukuran tetap Untuk Card)
  + - * + Card (Membuat Card)

Stack (Membuat Stack)

Opacity (untuk pattern)

Container (Image Atas)

Container (Detail Konten)

A computer screen shot of a program

Description automatically generated

A screen shot of a computer program

Description automatically generated

A screen shot of a computer program

Description automatically generated

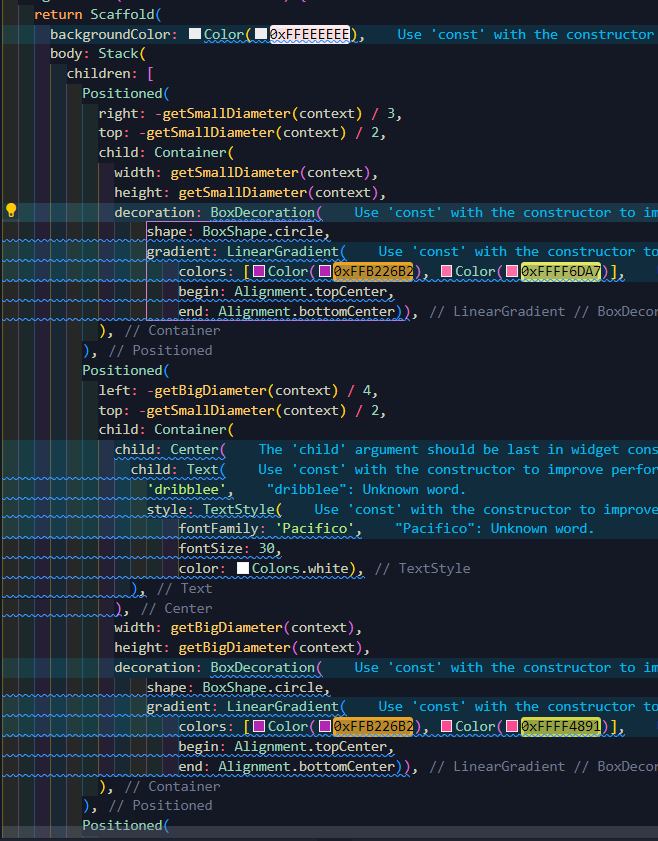
## [PR] - Positioned, Floating ActionButton, Login Page

Positioned untuk mengatur posisi, bahkan bisa disimpan keluar dari ukuran device.

Align, untuk membuat posisi dari widget / mengatur posisi dari widget.



Sebelumnya kita buat fungsi dimana akan menghitung mediaQuery / responsivable dari ukuran device nya.

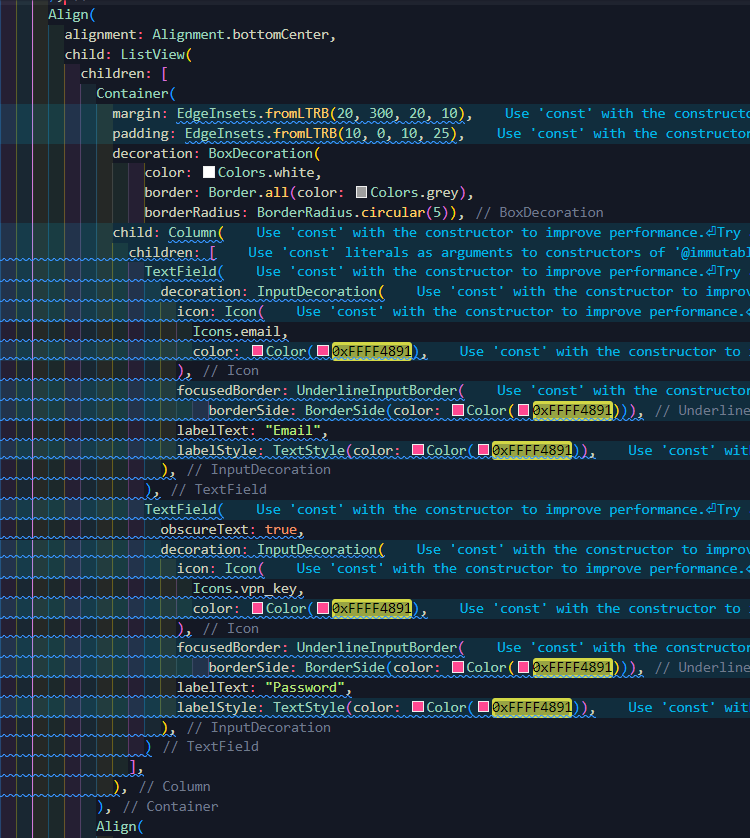
Selanjutnya Kita tambahkan Positioned untuk mengatur posisi bahkan membuat posisinya nanti seolah olah keluar dari ukuran devicenya sehingga terpotong.

Buat Container untuk membuat Container dan set Ukurannya dengan fungsi yang sudah dibuat.

A screen shot of a computer program

Description automatically generatedSelanjutnya set Alignment, untuk apa?

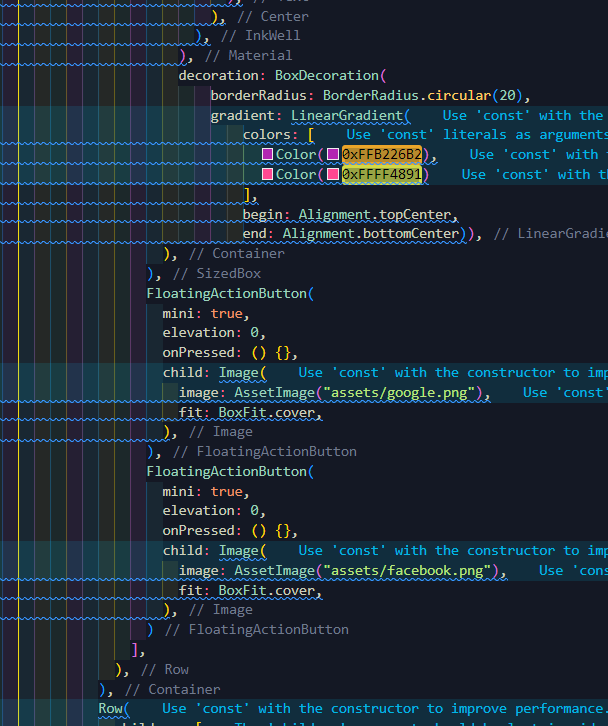
Untuk membuat Posisi nya nanti berada di bagian bottomCenter / bawah tengah

Kita simpan ListView ke dalam align dimana Align akan menyimpan widget selanjutnya bedasarkan ukuran parent nya.

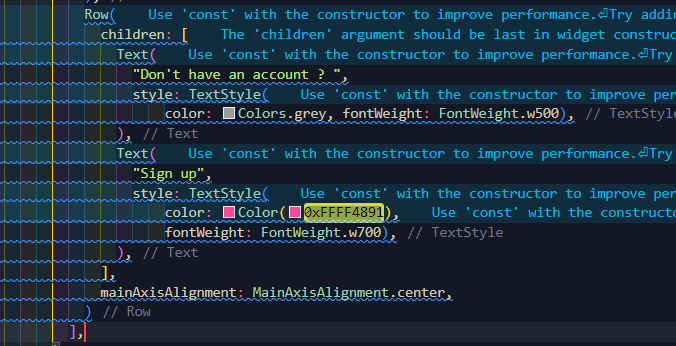
Buat Container Untuk Pembungkus, lalu Buat Column untuk membuat 2 Text Field

A screen shot of a computer program

Description automatically generatedSelanjutnya kita buat Align lagi untuk menyimpan text forgot password, buat juga untuk button sign in.

Berikan dekorasi design pada button, dan buat 2 button untuk login dengan facebook / google menggunakan FloatingActionButton.

Set mini untuk merubahnya menjadi kecil

Lalu untuk Row terakhir berikan text jika user belum punya akun.

Hasilnya

A screenshot of a login form

Description automatically generated