

PRAKTIKUM 2

DART BASIC

A. TUJUAN PEMBELAJARAN

1. Mahasiswa dapat membuat program sederhana dengan bahasa pemrograman Dart!

B. DASAR TEORI

Dasar teori.

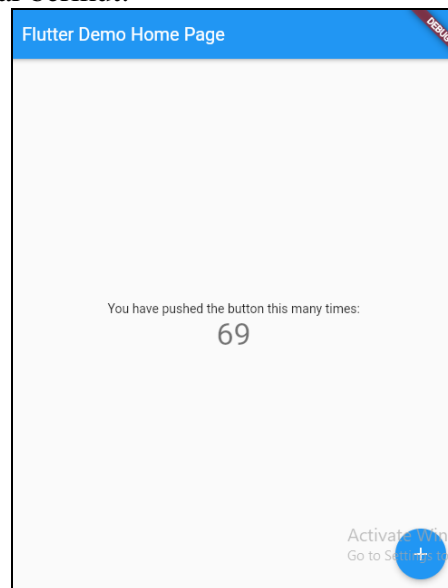
C. TUGAS PENDAHULUAN

1. Pelajari perintah-perintah dasar dalam pemrograman dengan menggunakan bahasa Dart!
2. Tuliskan contoh kode program untuk percabangan dan perulangan dalam bahasa Dart!
3. Tuliskan contoh kode program untuk menampilkan bilangan genap dalam bahasa Dart!

D. PERCOBAAN

Percobaan 1 : Menambahkan Kondisi

Saat pertama kali membuat project Flutter maka kita akan disugahi sebuah aplikasi untuk counter bilangan seperti gambar berikut:



Jika tombol + di pojok kiri bawah ditekan, maka angka yang ditunjukkan akan berubah dan selalu bertambah. Kita akan memodifikasi aplikasi Hello World Flutter tersebut.

Langkah Percobaan:

1. Buka aplikasi Hello World Flutter (default example project Flutter)
2. Buka file `main.dart` dalam folder `/lib`

3. Di dalam file tersebut ada 1 main method dan 3 class sebagai berikut:

```
+void main() {...}

+class MyApp extends StatelessWidget {...}

+class MyHomePage extends StatefulWidget {...}

+class _MyHomePageState extends State<MyHomePage> {...}
```

4. Bukalah class `_MyHomePageState`

5. Berikut adalah isi dari class `_MyHomePageState`

```
class _MyHomePageState extends State<MyHomePage> {
  int _counter = 0;

  void _incrementCounter() {
    setState(() {
      _counter++;
    });
  }

  @override
  Widget build(BuildContext context) {...}
}
```

6. Tambahkan kode program berikut pada method `_incrementCounter()`

```
class _MyHomePageState extends State<MyHomePage> {
  int _counter = 1;

  void _incrementCounter() {
    setState(() {
      _counter++;
      if(_counter>10){
        _counter = 1;
      }
    });
  }

  @override
  Widget build(BuildContext context) {...}
}
```

7. Jangan lupa mengubah nilai default variable `_counter` menjadi 1
8. Run Program
9. Aplikasi hanya akan mengkonter bilangan sampai nilai maksimal 10. Setelah 10, angka akan kembali menjadi 1

Percobaan 2 : Menampilkan Bilangan Ganjil atau Genap

Langkah Percobaan:

1. Buat sebuah variable String dengan nama `_text`

```
class _MyHomePageState extends State<MyHomePage> {  
  int _counter = 1;  
  String _text = "Ganjil";  
}
```

2. Tambahkan kode program berikut sebagai logika pengecekan bilangan ganjil dan genap:

```
void _incrementCounter() {  
  setState(() {  
    _counter++;  
    if(_counter>10){  
      _counter = 1;  
    }  
  
    if(_counter%2 == 0){  
      _text = "Genap";  
    }else{  
      _text = "Ganjil";  
    }  
  });  
}
```

3. Bukalah method `Widget build(BuildContext context)`

```

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(...), // AppBar
    body: Center(
      child: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: <Widget>[
          const Text(
            'You have pushed the button this many times:',
          ), // Text
          Text(
            '$_counter',
            style: Theme.of(context).textTheme.headline4,
          ), // Text
        ], // <Widget>[]
      ), // Column
    ), // Center
    floatingActionButton: FloatingActionButton(...), // This trailing
  ); // Scaffold
}

```

4. Buatlah sebuah Widget Text baru dibawah Widget Text untuk menampilkan bilangan counter

```

      child: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: <Widget>[
          const Text(
            'You have pushed the button this many times:',
          ), // Text
          Text(
            '$_counter',
            style: Theme.of(context).textTheme.headline4,
          ), // Text
          //tambahkan kode disini.....
        ], // <Widget>[]
      ), // Column
    ), // Center
    floatingActionButton: FloatingActionButton(...), // This trailing
  ); // Scaffold
}

```

5. Berikut kode program yang harus ditambahkan

```

Text(
  _text,
  style: Theme.of(context).textTheme.headline4,
), // Text

```

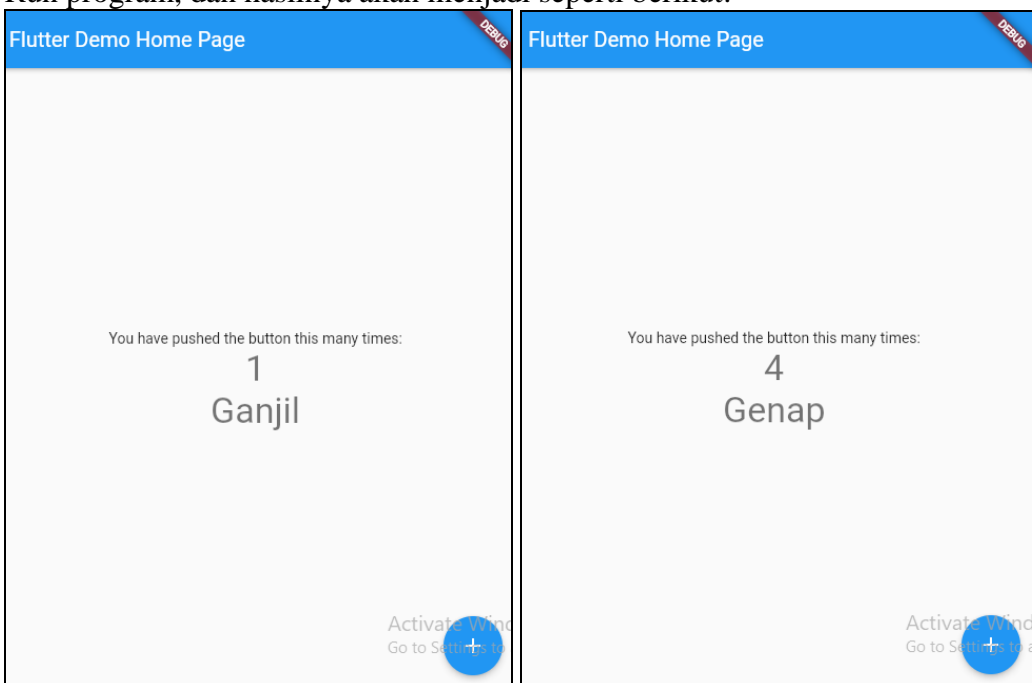
6. Sehingga menjadi kumpulan widget seperti berikut:

```

child: Column(
  mainAxisAlignment: MainAxisAlignment.center,
  children: <Widget>[
    const Text(
      'You have pushed the button this many times:',
    ), // Text
    Text(
      '$_counter',
      style: Theme.of(context).textTheme.headline4,
    ), // Text
    Text(
      _text,
      style: Theme.of(context).textTheme.headline4,
    ), // Text
  ], // <Widget>[]
), // Column

```

7. Run program, dan hasilnya akan menjadi seperti berikut:

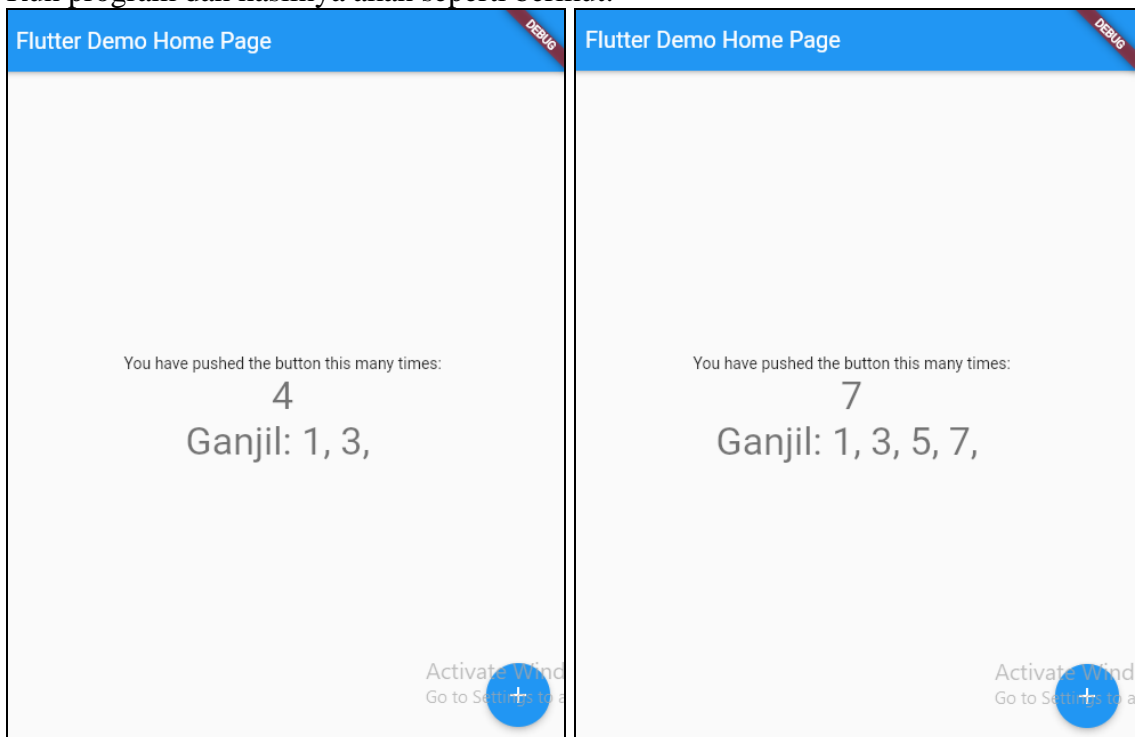


Percobaan 3 : Menampilkan bilangan ganjil mulai dari 1 sampai n
Langkah Percobaan:

1. Ubahlah method `incrementCounter()` sebagai berikut:

```
void _incrementCounter() {  
  setState(() {  
    _counter++;  
    if(_counter>10){  
      _counter = 0;  
    }  
  
    _text = "Ganjil: ";  
    for(int i=0; i<=_counter; i++){  
      if(i%2 != 0){  
        _text += '${i}, ';  
      }  
    }  
  });  
}
```

2. Run program dan hasilnya akan seperti berikut:



E. LATIHAN

Buat program untuk menampilkan bilangan genap kelipatan 3 mulai dari 1 sampai n!

F. TUGAS

1. Buatlah program untuk menampilkan bilangan prima mulai dari 1 sampai n!

G. LAPORAN RESMI

Kumpulkan hasil Percobaan, Latihan dan Tugas di atas dan tambahkan analisa untuk tiap percobaan, latihan, dan tugas yang telah dibuat.