

Online Class

Dart Language Programming

By Saiful Bahri

<https://www.linkedin.com/in/bahrie>



CODE with BAHRI

Pengantar

Flutter framework di kembangkan menggunakan bahasa pemrograman dart. Untuk itu mari kita mengenal dulu apa itu bahasa pemrograman dart.

Dart Programming

Kali ini kita akan mengenal bahasa pemrograman dart terlebih dahulu.

Apa itu bahasa pemrograman dart?

Bahasa pemrograman dart dirancang dan dikembangkan oleh Lars Bak dan Kasper Lund di Google. tujuan Google membuat bahasa pemrograman dart adalah untuk membangun aplikasi seluler dan desktop. Dart secara resmi diperkenalkan pada Oktober 2011 di konferensi GOTO, Denmark. Versi pertama Dart yaitu versi 1.0 resmi dirilis pada Nov 2013.

Dart Programming

Dart bisa digunakan untuk membuat aplikasi android, iOS, front-end web, IoT, backend (CLI), dan Game.

Contoh beberapa aplikasi yang dibangun dengan Dart:

- Google AdSense (front-end)
- Google AdWords (front-end)
- Google Fiber (front-end)
- Hamilton (Android dan iOS)

Dart multiplatform

Dart dapat digunakan pada tiga platform ini:

1. Flutter

Apabila kita ingin membuat aplikasi Android dan iOS dengan bahasa pemrograman Dart, maka kita harus menggunakan framework Flutter.

Flutter adalah sebuah framework yang dirancang khusus untuk membangun antarmuka (UI) aplikasi mobile.

Dart Programming

2. Web

Penggunaan Dart pada web biasanya dalam bentuk front-end. Kode program Dart dapat dikompilasi menjadi kode Javascript, sehingga web browser dapat mengeksekusinya.

Biasanya, pengembangan web dengan Dart menggunakan framework:

- AngularDart
- Material Design Lite (MDL)/Dart
- OverReact

Dart Programming

3. Server

Dart dapat juga digunakan pada server untuk membuat program berbasis teks (CLI), program IoT, backend, dll.

Beberapa framework yang bisa digunakan untuk mengembangkan backend:

- Dart Frog
- Aqueduct
- Conduit
- Alfred

Keunggulan Dart

Meskipun bahasa pemrograman dart dinilai masih baru. Namun berikut adalah beberapa keunggulan dart dari bahasa pemrograman lain.

Keunggulan Dart

1. Mudah Dipelajari

Setiap developer pasti menginginkan sebuah bahasa pemrograman yang mudah dipelajari, Anda bisa menemukannya saat Anda mempelajari bahasa pemrograman dart. Terlebih lagi, jika Anda sudah pandai dalam bahasa pemrograman JavaScript, Java, atau C# tentu akan sangat mudah Anda untuk bisa menyesuaikan penulisan kode di bahasa pemrograman dart. Anda hanya perlu mempelajari prinsip-prinsip dasar dari bahasa pemrograman tersebut.

2. Dokumentasi yang Lengkap

Dokumentasi yang disediakan oleh Google terhadap bahasa pemrograman dart sangat lengkap, Anda bisa mempelajarinya sendiri. Anda bisa dengan mudah mendapatkan jawaban atas pertanyaan Anda ketika Anda merasa kebingungan untuk menggunakan bahasa pemrograman dart. Ini akan memudahkan Anda ketika menulis kode program bahasa dart.

Berikut dokumentasi yang disediakan oleh dart.dev

<https://dart.dev/codelabs>

3. Lebih Cepat

Kinerja bahasa pemrograman dart sangat tinggi dan cepat. Program yang ditulis dalam bahasa dart lebih cepat berjalan dibandingkan dengan bahasa pemrograman seperti JavaScript. Ini akan membuat waktu Anda lebih efisien untuk membuat banyak program-program atau ketika Anda dikejar oleh deadline dari klien untuk menyelesaikan pembuatan website atau aplikasi.

4. Stabil untuk Membangun Aplikasi Real-time

Jika Anda ingin membangun aplikasi real-time, bahasa pemrograman dart adalah pilihan yang tepat untuk Anda gunakan. Pasalnya, bahasa pemrograman dart lebih stabil untuk membangun aplikasi real-time, ini karena bahasa pemrograman dart merupakan bahasa yang berbasis pada objek, yang didalamnya terdapat class abstract dan class stream yang dapat menangani integrasi websocket dalam kode dengan rapi.

5. Adanya dukungan Kompilasi AOT dan JIT

Kemampuan unik dari bahasa pemrograman dart adalah kompilasi AOT (Ahead of Time) dan JIT (Just in Time). Saat di AOT, ini membuat kode yang terdapat di dart bisa diubah menjadi kode mesin asli. Kemudian, begitu masuk ke dalam mode JIT, semua itu bisa dikompilasi untuk membuat siklus development yang sangat cepat.

6. Tanpa Konfigurasi

DartPad tersedia untuk Anda yang tidak ingin menginstal atau mengkonfigurasi apa pun, Anda bisa menuliskan kode tanpa melakukan hal tersebut. DartPad juga didesain dengan sederhana, Anda bisa langsung menuliskan kode dan klik 'Jalankan' untuk menjalankan program aplikasi yang sedang Anda kerjakan. Hanya saja, dukungan untuk perpustakaan di DartPad terbatas pada tingkat dasar.

<https://dartpad.dev/>

7. Fitur Hot Reload

Dengan fitur hot reload, developer dapat melihat hasil code mereka di layar secara real-time.

Fitur hot reload membantu Anda bereksperimen dengan cepat dan mudah, membangun UI, menambahkan fitur, dan memperbaiki bug. Hot reload bekerja dengan menyuntikkan file source code yang diperbarui ke dalam Dart Virtual Machine (VM) yang sedang berjalan. Setelah VM memperbarui class dengan bidang dan fungsi versi baru, framework Flutter secara otomatis membuat/render ulang widget tree (tampilan), memungkinkan Anda melihat efek perubahan tampilan dengan cepat.

Dart pada Flutter

**Seberapa penting bahasa pemrograman dart untuk flutter?
Yuk kita bahas bersama sama.**

Dart pada Flutter

Flutter sebagai framework

Sebagai framework, semua library flutter ditulis menggunakan bahasa pemrograman dart, oleh karena itu semua keunggulan dart bisa dinikmati juga di framework flutter. Dalam proses development aplikasi mobile dengan flutter, kita harus menggunakan bahasa pemrograman dart. Jadi dart ini sangat penting bagi flutter untuk terus digunakan oleh developer developer nya.

Dart pada Flutter

Dart sdk ada di dalam flutter

Ketika kita install flutter, maka dart sdk otomatis ikut terinstall, jadi tidak perlu install secara terpisah.

Update versi pada flutter sering kali diikuti oleh update versi juga pada dart.

Dart Basic

By Saiful Bahri

<https://github.com/bahrie127>

Pengantar

Setelah sebelumnya kita mengenal apa itu bahasa pemrograman dart, kali ini kita akan masuk belajar algoritma bahasa pemrograman dart dasar

Starting points

Bahasa Pemrograman dart mempunyai sintak atau cara penulisannya sendiri. Namun sintak yang dimiliki dart mirip dengan java/javascript/c#. Yuk mulai dari starting point nya dulu.

Starting points

Starting point

Program yang ditulis menggunakan bahasa pemrograman dart dimulai dengan sintak seperti dibawah:

```
void main() {  
  
//code  
  
}
```

Pertama dimulai dari kata void lalu ikuti dengan kata main(){}. Kode kita berada didalam kurung kurawal.

Data Types

Data types atau tipe data adalah klasifikasi variable untuk menentukan data yang akan disimpan ke dalam memori. Tentunya, berbagai macam tipe data tersebut sangat bermanfaat bagi kinerja komputer karena dengan tipe data, programmer mengenalkan compiler, data yang akan digunakan.

Data types

1. String

String adalah kumpulan huruf, angka, dan tanda baca lainnya. Jadi isinya bebas asalkan tulisan.

```
void main() {  
    String data = 'Hello Code with Bahri';  
    print(data);  
}
```

Data types

2. Num (num)

Num, merupakan bilangan bulat dan bilangan pecahan. Tipe data ini bisa menjadi integer ataupun double, sesuai nanti data yang diinputkan.

```
void main() {  
    num data = 20;  
    print(data);  
    num data2 = 22.2;  
    print(data);  
}
```

Data types

3. Integer (int)

Integer merupakan tipe data yang merepresentasikan angka tanpa desimal.

```
void main() {  
    int myInteger = 10;  
    print(myInteger);  
}
```

Data types

4. Double (double)

double merupakan angka desimal.

```
void main() {  
    double myDouble = 0.1;  
    print(myDouble);  
}
```

Data types

5. Boolean (bool)

Merupakan tipe data yang berisikan kebenaran (true/false)

```
void main() {  
    bool data = true;  
    print(data);  
}
```

Data types

6. List

List merupakan tumpukan data. Jika kalian familiar dengan java atau javascript, List merupakan sebuah Array.

```
void main() {  
    List data = [1,2,3,'a','b', 'c'];  
    print(data);  
}
```

Data types

7. Map

Jika kalian mengerti Javascript Object Notation (JSON), maka kalian akan lebih mudah memahami apa itu Map. Map merupakan pasangan kunci dan isi. Kunci dan isi tersebut biasa dipanggil dengan key dan value. Setiap key dan value dapat dideklarasikan tipe datanya.

```
void main() {  
  
  Map<String, dynamic> data = {  
  
    'url': 'https://jagoflutter.com/',  
  
    'topic': 'Flutter Engineering',  
  
  };  
  
  print(data);  
  
}
```

8. Dynamic (dynamic)

Dynamic merupakan tipe data yang isinya bisa berubah, dynamic bisa berupa integer, string, double, num, object, dan lain sebagainya. Contoh sebagai berikut:

```
void main() {  
    dynamic a = 12.2;  
    a = 'Hello Bahri!';  
    print(a);  
}
```


9. Object (object)

Merupakan kelas dasar dari semua object yang ada di dart.

```
void main() {  
    Object object = "Flutter Engineering";  
    print(object);  
}
```

10. var

var merupakan tipe data yang digunakan untuk mendeklarasikan Object dari variable yang dipanggil.

```
void main() {  
    var penulis = 'Sayuti';  
    print(penulis);  
}
```

Variables

Variabel merupakan lokasi penyimpanan yang ada di memori. Setiap variabel memiliki kemampuan menyimpan suatu informasi sesuai dengan tipe data yang dideklarasikan untuk variabel tersebut saja

Variables

Contoh Variable

```
var umur = 18;
```

```
dynamic nama = 'bahri';
```

```
String kota = 'Sleman';
```

Kode di atas mendeklarasikan suatu variabel yang bernama kota dengan tipe data String

Operators

Operator yaitu sebuah simbol yang digunakan melakukan operasi, biasanya digunakan untuk memanipulasi atau memodifikasi data.

Jenis Operator

1. Operator Aritmatika
2. Operator Perbandingan dan Relasional
3. Operator Logika
4. Operator Cek Tipe data

Operators

Operator aritmatika

Operator Aritmatika adalah operator yang melakukan operasi seperti penjumlahan, pengurangan, perkalian, dan pembagian. Di bawah ini adalah daftar operator dalam pemrograman dart.

Operator	Deskripsi
+	Penjumlahan
-	Pengurangan
*	Perkalian
/	Pembagian
%	Hasil bagi

```
int c = 6;  
c = c+ 6; // c = 12
```

Operators

Operator Relasional

Operator relasional adalah operator yang melakukan operasi membandingkan operand tipe data misalnya saja kurang dari, lebih besar dari dan sebagainya.

Operator	Deskripsi
>	Lebih besar dari
<	Lebih kecil dari
>=	Lebih besar atau sama dengan
<=	Lebih kecil atau sama dengan
!=	Tidak sama dengan
==	Tepat sama dengan

```
String kota = 'Sleman';  
if(kota == 'Sleman') true else false;
```


Operator Cek Tipe data

Digunakan untuk memeriksa jenis tipe data objek saat runtime.

Symbol	Meaning	Example
<code>as</code>	Cast a type to another	<code>obj as String</code>
<code>is</code>	True if the object has a certain type	<code>obj is double</code>
<code>is!</code>	False if the object has a certain type	<code>obj is! int</code>

If else condition

If else condition atau percabangan.

Pernyataan if digunakan untuk menguji suatu kondisi kemudian mengerjakan pernyataan yang lain sesuai hasil pengujian.

Studi kasus IF

Di dalam studi kasus menggunakan if kali ini kita akan mencoba mengosongkan sebuah daftar dengan kondisi if hanya saja jika kondisi tersebut tidak kosong. kita dapat memeriksa apakah sebuah koleksi kosong atau tidak, menggunakan properti isEmpty dan isEmpty. properti isEmpty memiliki pengertian bernilai true jika koleksi yang kita masukkan kosong, sedangkan properti isEmpty akan bernilai true jika koleksi yang kita masukkan tidak kosong.

If else condition

```
void main() {  
    var testList = [2,4,8,16,32];  
    print(testList);  
    if(testList.isNotEmpty){  
        print('Tulisan ini akan tampil, jika kondisi bernilai true');  
        testList.clear();  
    }  
    print(testList);  
}
```

If else condition

Switch case

Switch dan case adalah controll flow yang sejenis dengan if dan else dimana switch case ini digunakan untuk percabangan.

If else condition

Studi kasus

Seorang pendaki melakukan pendakiannya pada pagi hari, dia sudah melakukan instruksi kepada semua anggota pendakinya untuk melakukan hal hal ini :

1. Saat tiba di pos 1 team akan makan snack
2. Saat tiba di pos 2 team akan makan daging
3. Saat tiba di pos 3 team akan makan sayur
4. saat tiba dipuncak team akan bergembira

If else condition

```
void main(){
int posSaatIni = 10;
switch(posSaatIni){
    case 1:
        print("Makan Snack");
        break;
    case 2:
        print("Makan daging");
        break;
    case 3:
        print("Makan Sayur");
        break;
    case 4:
        print("Berbahagia");
        break;
    default:
        print("Pos yang anda masukan tidak terdaftar");
}
}
```

Looping

Looping atau perulangan

Looping

Perulangan for

Pernyataan for digunakan untuk melakukan perulangan dengan menentukan kondisi perulangan dan pernyataan increment / decrement.

```
for(inisialisasi ; kondisi; increment){  
    pernyataan1;  
    pernyataan2;  
    ....  
}
```

```
for(var i = 0; i <= 10; ++i)  
    print("Number $i");
```

Looping

Perulangan while

Pernyataan while digunakan untuk melakukan perulangan dengan menentukan kondisi yang menyebabkan perulangan dihentikan.

```
while (ekspresi) {  
    pernyataan1;  
    pernyataan2;  
    ....  
}
```

```
int i = 0;
```

```
while (i < 10) {
```

```
    print('nilai while : $i');
```

```
    i++;
```

```
}
```

<https://instagram.com/codewithbahri>

Looping

Perulangan do while

Statement do while hampir sama dengan while. Perbedaannya, jika do while hanya melakukan satu kali perulangan dulu, kemudian mengecek kondisinya. Sedangkan while kondisi di cek dulu baru kemudian statement perulangannya dijalankan.

```
var i = 0;

do {
    print("Number $i");
    ++i;
} while (i <= 10)
```

Looping

Perulangan for in

Struktur perulangan for in digunakan untuk menampilkan isi yang ada dalam array atau list, perulangan for in sebenarnya sama dengan perulangan for yang bedakan cuma kata kunci in dan tidak mengeluarkan nilai index nya.

```
List<String> listNama = ['bahri', 'rozak', 'amri'];  
  
for(String nama in listNama){  
    print(nama);  
}
```

Support with Follow/Subscribe

- Youtube : <https://youtube.com/@codewithbahri>
- Instagram: <https://instagram.com/codewithbahri>
- Linkedin: <https://linkedin.com/in/bahrie>
- Github: <https://github.com/bahrie127>
- Tiktok: <https://tiktok.com/@codewithbahri>