

Nama : Eddo Dava Alfarisi

Nim : 2241720232

No/cls : 07/3C

Pengantar Bahasa Pemrograman Dart - Bagian 1

1. Kelebihan bahasa pemrograman dart
 - Productive tooling: merupakan fitur kakas (tool) untuk menganalisis kode, plugin IDE, dan ekosistem paket yang besar.
 - Garbage collection: untuk mengelola atau menangani dealokasi memori (terutama memori yang ditempati oleh objek yang tidak lagi digunakan).
 - Type annotations (opsional): untuk keamanan dan konsistensi dalam mengontrol semua data dalam aplikasi.
 - Statically typed: Meskipun type annotations bersifat opsional, Dart tetap aman karena menggunakan fitur type-safe dan type inference untuk menganalisis types saat runtime. Fitur ini penting untuk menemukan bug selama kompilasi kode.
 - Portability: bahasa Dart tidak hanya untuk web (yang dapat diterjemahkan ke JavaScript) tetapi juga dapat dikompilasi secara native ke kode Advanced RISC Machines (ARM) dan x86.
2. Evolusi dari *Dart*

Pertama diluncurkan pada 2011, dan merilis versi stabil pada 2013 dilanjutkan perubahan besar yaitu *Dart 2.0* pada akhir 2018. awalnya fokus pada pengembangan web untuk menggantikan *Javascript* (menjadi penerus javascript). Kemudian *Dart* juga menawarkan performa dan alat yang lebih baik untuk proyek yang lebih besar karena telah disediakan plugin IDE agar mendapat performa terbaik dan dinamis. *Dart* juga dibentuk a dibentuk kuat dan fleksibel dengan menambahkan fitur OOP yang dapat menyeimbangkan fleksibilitas dan ketangguhan
3. Ada dua cara utama untuk menjalankan kode *Dart*:
 - a. Dart Virtual Machines (VMs)
 - b. Kompilasi JavaScript

Kode Dart bisa dijalankan di lingkungan yang mendukung bahasa Dart dengan mempertimbangkan beberapa fitur penting seperti sistem runtime, pustaka inti Dart, dan pengumpul sampah (garbage collectors).

Ada dua mode eksekusi kode Dart:

- **Kompilasi Just-In-Time (JIT):** Kode dikompilasi saat dibutuhkan, memungkinkan fitur seperti debugging dan hot reload. Digunakan selama pengembangan aplikasi mobile.
- **Kompilasi Ahead-Of-Time (AOT):** Kode dikompilasi sebelumnya, memberikan performa yang lebih baik tetapi tanpa fitur seperti debugging dan hot reload.

Fitur hot reload, yang sangat terkenal di Flutter, memungkinkan pengembang melihat perubahan kode secara cepat, meningkatkan kecepatan dan kualitas pengembangan perangkat lunak.

4. Struktur dari *Dart*

dart dirancang untuk *OOP(Object Oriented Programming)* . yaitu memiliki fitur enkapsulasi, komposisi, abstraksi dan polimorf.*dart* juga memiliki beberapa operator, yaitu

- a. operator aritmatika
- b. operator *increment* dan *decrement*
- c. operator perbandingan
- d. operator logika

5. *Dart Style*

Main Function : seperti bahasa yang lain untuk mengeksekusi pemrograman *dart* menggunakan fungsi *Main*