**Apa itu Dart?**

Pernahkah kalian mendengar atau bahkan mengembangkan aplikasi Flutter? Jika iya, berarti kalian sudah tahu bahwa Flutter ditulis menggunakan bahasa Dart. Jika belum, maka selamat berkenalan dengan Dart.

[Dart](https://dart.dev/) adalah bahasa pemrograman yang *open source* dan *general purpose*. Dart dikembangkan oleh Google dan ditujukan untuk membuat aplikasi *multiplatform* seperti *mobile*, *desktop*, dan *web*.

Dart awalnya dikenalkan pada *GOTO conference* pada tahun 2012. *Project* ini didirikan oleh Lars Bak dan Kasper Lund dari Google, sampai akhirnya versi Dart 1.0 dirilis pada 14 November 2013. Bulan Agustus 2018, Dart 2.0 dirilis dengan perubahan bahasa seperti perubahan *type system*.

Jauh sebelum ada Flutter, Dart awalnya digunakan untuk membuat web yang ada di Google. Tujuan awal pembuatan Dart adalah untuk menggantikan JavaScript yang dinilai memiliki banyak kelemahan. Sejak saat itu, rilisnya Flutter SDK untuk pengembangan iOS, Android, dan web menjadi sorotan baru pada bahasa Dart.

**Kenapa perlu belajar Dart?**

Dalam buku “***The Pragmatic Programmer***” disebutkan bahwa untuk menjadi *professional software developer* kita perlu belajar setidaknya satu bahasa baru setiap tahunnya. Lalu adakah alasan yang bagus untuk memulai belajar pemrograman dengan Dart?

* Pertama, ***Dart adalah bahasa pemrograman yang fleksibel***. Dart bisa berjalan di mana pun baik itu Android, iOS, maupun web. Sebagai *developer*, tentunya sebuah keuntungan jika bisa menuliskan kode dan bisa berjalan di mana saja.
* ***Dart adalah project open-source***. Dart dibuat oleh Google, lalu bersama dengan komunitas developer Dart mengembangkan teknologi dan fitur-fitur menarik yang bisa ditambahkan pada Dart. Jika Anda menemukan *bug* atau masalah pada Dart, Anda dapat melaporkannya atau bahkan memperbaikinya sendiri. Selain itu Anda tidak perlu khawatir dalam masalah lisensi ketika menggunakan bahasa Dart. Anda dapat ikut berkontribusi pada bahasa Dart pada repositori berikut: <https://github.com/dart-lang>.
* ***Dart digunakan oleh Flutter***. Sejak kemunculan Flutter, Dart kembali menjadi perhatian. Saat ini ada banyak perusahaan yang menggunakan Flutter pada aplikasinya. Flutter bisa dibilang merupakan proyek yang revolusioner dari Google untuk mengembangkan aplikasi multiplatform dengan tampilan UI yang menarik. Untuk itu, jika Anda tertarik mengembangkan aplikasi dengan Flutter, maka menguasai Dart adalah hal yang fundamental.
* ***Dart memiliki dukungan tools yang lengkap***. Hampir setiap teks editor atau IDE memiliki dukungan besar untuk Dart. Anda dapat menggunakan IDE seperti IntelliJ IDEA, Webstorm, Android Studio maupun *editor* sederhana seperti VS Code, Sublime text, Atom, atau yang lainnya sesuai kenyamanan Anda.
* ***Dart mudah dipelajari dan bagus sebagai first language***. Anda akan bisa memahami Dart dengan cepat khususnya jika sudah familiar dengan bahasa pemrograman populer lain seperti Java, Python, JavaScript, dll. Bahkan jika Anda baru memulai pemrograman, Dart adalah bahasa yang bagus. Anda tidak perlu menginstal apapun, cukup memanfaatkan *online compiler* dari Dart, Anda sudah bisa menulis dan menjalankan aplikasi Dart. Selain itu, dokumentasi dan tutorial Dart yang disediakan Google cukup mudah untuk diikuti, ditambah dengan sintaks yang sederhana, dan komunitas yang bersahabat dalam membantu kita mempelajari Dart.

**Karakteristik Dart**

Dart merupakan bahasa modern dan berfitur lengkap. Dart juga memiliki banyak kemiripan dengan bahasa lain yang sudah banyak dikenal seperti Java, C#, Javascript, Swift, dan Kotlin. Salah satu rancangan utama dari Dart adalah supaya bahasa ini familiar bagi *developer* Javascript dan Java/C#. Artinya,  yang telah familiar dengan kedua bahasa tersebut dapat memulai belajar bahasa Dart dengan lebih mudah. Namun, jika Anda adalah calon *developer* yang baru memulai belajar pemrograman dan memutuskan Dart sebagai *first language*, tenang saja. Dart adalah bahasa yang nyaman dan mudah dipelajari untuk memulai pemrograman.

Kita ambil contoh potongan kode Dart berikut:

main() {

  var name = 'Dicoding';

  String language = 'Dart';

  print('Hello $name. Welcome to $language!');

}

Jika Anda telah familiar dengan bahasa pemrograman lain seperti Java, Kotlin, atau Swift, tentu Anda telah paham bagaimana kode di atas bekerja. Jadi, kode di atas akan menampilkan “***Hello Dicoding. Welcome to Dart!***” pada konsol.

Dart sebagai bahasa memiliki beberapa karakteristik berikut:

* *Statically typed*,
* *Type inference*,
* *String Interpolation*,
* *Multi-paradigm: OOP & Functional*.

Dart adalah bahasa yang ***statically typed***, artinya kita perlu mendefinisikan variabel sebelum bisa menggunakannya. Potongan kode berikut adalah contoh deklarasi variabel pada Dart.

var name = 'Dicoding';

String language = 'Dart';

Bisa dilihat bahwa pada Dart kita tidak perlu mendefinisikan tipe data variabel secara eksplisit. Ini disebabkan karena Dart juga mendukung ***type inference***, di mana tipe data akan secara otomatis terdeteksi ketika suatu variabel diinisialisasi. Sebagai contoh variabel name di atas akan terdeteksi sebagai String. Selain itu, Dart juga memiliki *dynamic variable*. Apa itu? Bahasan ini akan kita dalami pada modul *variable*.

print('Hello $name. Welcome to $language!');

Kode di atas juga menunjukkan kalau Dart memiliki fitur ***String interpolation***. Ini adalah fitur di mana kita bisa menyisipkan variabel ke dalam sebuah objek String tanpa *concatenation* (penggabungan objek String menggunakan +). Dengan fitur ini, tentunya kita menjadi lebih mudah dalam membuat objek String yang dinamis.

## Dart Platform

Pada modul sebelumnya kita telah mempelajari bahwa Dart digunakan untuk menulis aplikasi multi-platform. Teknologi compiler yang fleksibel memungkinkan kode Dart dapat dijalankan dengan cara yang berbeda, tergantung target platform yang dituju.

* ***Dart Native*** : Ditujukan untuk program yang menargetkan perangkat seluler, desktop, server, dan lainnya. Dart Native mencakup ***Dart VM*** dengan kompilasi ***JIT*** (just-in-time) dan kompiler ***AOT*** (ahead-of-time) untuk menghasilkan kode mesin.
* ***Dart Web*** : Ditujukan untuk program yang menargetkan web. Dart Web menyertakan kompiler development (***dartdevc***) dan kompiler production (***dart2js***).

### Dart Native (VM JIT dan AOT)

Dart Native memungkinkan kode Dart dijalankan dan dikompilasi dengan kode mesin ARM atau X64 native untuk aplikasi mobile, desktop, dan server.

Dart VM dilengkapi dengan ***just-in-time compiler (JIT)*** yang mendukung interpretasi murni dan optimasi runtime. Lalu apa keuntungan dari menggunakan JIT? Compiler bertugas untuk mengubah bahasa high-level yang kita tulis menjadi bahasa low-level yang dimengerti oleh mesin. JIT compiler akan mengubah bahasa pemrograman yang kita tulis menjadi intermediate language atau bytecode seperti pada Java, selanjutnya instruksi ke mesin akan dilakukan hanya ketika dibutuhkan, sehingga disebut just-in-time. Metode ini akan membuat proses iterasi program menjadi lebih efisien.

Saat aplikasi siap digunakan untuk production, Anda dapat memanfaatkan ***Dart AOT compiler***. Apa ini? Apa bedanya dengan JIT? AOT compilation akan mengubah bahasa high-level atau intermediate-level menjadi kode mesin pada mesin atau server sebelum aplikasi dijalankan. Kompilasi AOT akan menghasilkan rendering aplikasi yang lebih cepat dan ukuran yang lebih kecil karena kode telah dikompilasi sebelum aplikasi dijalankan.

### Dart Web (JavaScript)

Dart Web memungkinkan kode Dart dijalankan pada platform web yang didukung oleh JavaScript. Dengan Dart Web, kode Dart akan dikompilasi ke kode JavaScript sehingga nantinya bisa berjalan di browser.

Dart Web menggunakan ***Dart dev compiler (dartdevc),*** compiler yang mengonversi kode Dart menjadi JavaScript. Alih-alih menggunakan dartdevc secara langsung, Anda dapat menggunakan ***webdev***, yakni alat yang mendukung tugas inti developer seperti menjalankan, membangun, dan debugging.

Untuk mengoptimalkan kode pada lingkungan production, terdapat ***dart2js compiler*** yang dapat mengompilasi kode Dart dengan cepat dan ringkas. Dart2js menggunakan teknik yang bisa mengeliminasi kode yang tidak perlu.

**Rangkuman Materi**

Selamat! Anda telah menyelesaikan modul pertama pada kelas ini. Pada modul ini kita telah berkenalan Dart, mengetahui karakteristik, dan platform yang menjalankan Dart. Ini adalah awal yang baik! Mari kita uraikan apa yang sudah kita pelajari di modul ini:

* Dart adalah bahasa pemrograman open source dan general purpose buatan Google yang ditujukan untuk membuat aplikasi multiplatform seperti mobile, desktop, dan web.
* Tujuan awal pembuatan Dart adalah untuk menggantikan JavaScript yang dinilai memiliki banyak kelemahan.
* Beberapa alasan yang bisa Anda pertimbangkan untuk mulai belajar Dart, antara lain:
  + Dart adalah bahasa pemrograman yang fleksibel.
  + Dart adalah project open-source.
  + Dart digunakan oleh Flutter.
  + Dart memiliki dukungan tools yang lengkap.
  + Dart mudah dipelajari dan bagus sebagai first language.
* Dart didesain agar mirip dengan bahasa pemrograman populer yang sudah ada, seperti Java, C#, Javascript, Swift, dan Kotlin. Artinya, developer yang telah familiar dengan kedua bahasa tersebut dapat memulai belajar bahasa Dart dengan lebih mudah.
* Dart sebagai bahasa memiliki beberapa karakteristik berikut:
  + Statically typed, di mana kita perlu mendefinisikan variabel sebelum bisa menggunakannya.
  + Type inference, di mana tipe data tidak perlu didefinisikan secara eksplisit.
  + String expressions, bisa menyisipkan variabel ke dalam sebuah objek String tanpa concantenation (penggabungan objek String menggunakan tanda +).
  + Multi-paradigm: OOP & Functional, mendukung konsep object oriented programming dan menggunakan gaya functional programming.

Pada modul selanjutnya, kita akan menyiapkan aplikasi apa saja yang dibutuhkan untuk mulai membuat program dengan Dart pada laptop atau komputer kita. Sampai bertemu di modul selanjutnya!