**Panduan Effective Dart**

*Guidelines* dari Effective Dart dibagi menjadi empat bagian sesuai dengan fungsinya.

* *Style guide*. Mendefinisikan aturan untuk meletakkan dan mengatur kode. Panduan ini juga menentukan bagaimana format penamaan sebuah identifier, apakah menggunakan camelCase, \_underscore, dll.
* *Documentation guide*. Panduan ini terkait tentang apa yang boleh dan tidak ada di dalam komentar. Baik itu komentar dokumentasi atau komentar biasa.
* *Usage guide*. Panduan ini mengajarkan bagaimana memanfaatkan fitur bahasa secara terbaik untuk menerapkan perilaku. Penggunaan statement atau expression akan dibahas di sini.
* *Design guide*. Ini adalah panduan dengan cakupan terluas namun paling tidak mengikat. Panduan ini mencakup bagaimana mendesain API library yang konsisten dan bisa digunakan.

Effective Dart memiliki banyak *rules*. Masing-masing aturan akan diawali dengan kata kunci untuk menentukan bagaimana sifat dari aturan tersebut. Lima kata kunci tersebut, antara lain:

* ***DO***. Ketika aturan diawali dengan DO maka praktik tersebut harus selalu diikuti.
* ***DON’T***. Sebaliknya, aturan yang diawali dengan DON’T bukan merupakan hal yang baik untuk diterapkan.
* ***PREFER***. Ini adalah praktik yang harus diikuti. Namun, mungkin ada keadaan di mana lebih masuk akal untuk melakukan sebaliknya. Pastikan Anda memahami konsekuensi ketika Anda mengabaikan aturan ini.
* ***AVOID***. Ini adalah kebalikan dari PREFER. Panduan ini menjelaskan hal-hal yang tidak boleh dilakukan, namun kemungkinan ada alasan bagus untuk melakukannya pada beberapa kejadian.
* ***CONSIDER***. Panduan ini adalah praktik yang bisa Anda ikuti atau tidak Anda ikuti, tergantung pada keadaan dan preferensi Anda sendiri.

Meskipun ada banyak sekali aturan dan pedoman yang perlu diikuti dalam Effective Dart, kita tidak perlu khawatir dengan peraturan yang sangat ketat. Karena sebagian besar panduan yang ada bisa dibilang merupakan *common sense* dalam membuat program, bahkan jika tidak tertulis sekalipun. Selain itu konvensi bertujuan supaya kode kita menjadi lebih bagus, mudah dibaca, dan tentunya dipelihara.