**Cascade Notation**

Dart juga dilengkapi dengan cascade notation atau cascade operator. Operator ini memungkinkan kita untuk melakukan beberapa urutan operasi pada objek yang sama. Kita bisa mengakses property dari object dan menjalankan method di dalamnya bersamaan ketika kita menginstansiasi object. Cascade operator dituliskan dengan dua tanda titik (.. atau ?..).

Perhatikan contoh kode yang menggunakan cascade operator berikut berikut:

1. void main() {
2. var dicodingCat = Animal('', 2, 4.2)
3. ..name = 'Gray'
4. ..eat();
5. }

Apakah Anda mengerti maksud kode di atas? Kita menginstansiasi object Animal dengan constructor seperti biasa. Kemudian cascade operator yang mengikutinya akan melakukan operasi berdasarkan object yang dikembalikan oleh constructor. Contoh kode tersebut melakukan hal yang sama apabila kita menuliskan kode seperti ini:

1. var dicodingCat = Animal('', 2, 4.2);
2. dicodingCat.name = 'Gray';
3. dicodingCat.eat();

Cascade operator ini sering kali menghemat langkah dalam membuat variabel sementara, sehingga kode yang kita tulis menjadi lebih ringkas.

1. Animal('', 2, 4.2)
2. ..name = 'Gray'
3. ..eat();

6. /\* output
7. Gray is eating.
8. \*/

Cascade notation juga akan sering kita temui pada [*builder pattern*](https://dev.to/inakiarroyo/the-builder-pattern-in-dart-efg) seperti ini:

1. final addressBook = (AddressBookBuilder()
2. ..name = 'jenny'
3. ..email = 'jenny@example.com'
4. ..phone = '415-555-0100')
5. .build();