**Closures**

Suatu fungsi dapat dibuat dalam lingkup global atau di dalam fungsi lain. Suatu fungsi yang dapat mengakses variabel di dalam *lexical scope*-nya disebut dengan ***closure***. *Lexical scope* berarti bahwa pada sebuah fungsi bersarang (*nested functions*), fungsi yang berada di dalam memiliki akses ke variabel di lingkup induknya.

Berikut ini adalah contoh kode implementasi *closure*:

1. void main() {
2. var closureExample = calculate(2);
3. closureExample();
4. closureExample();
5. }
7. Function calculate(base) {
8. var count = 1;
10. return () => print("Value is ${base + count++}");
11. }

Ketika kode di atas dijalankan, konsol akan tampil seperti berikut:

1. Value is 3
2. Value is 4

Di dalam fungsi calculate() terdapat variabel count dan mengembalikan nilai berupa fungsi. Fungsi *lambda* di dalamnya memiliki akses ke variabel count karena berada pada lingkup yang sama. Karena variabel count berada pada *scope* calculate, maka umumnya variabel tersebut akan hilang atau dihapus ketika fungsinya selesai dijalankan. Namun pada kasus di atas fungsi *lambda* atau closureExample masih memiliki referensi atau akses ke variabel count sehingga bisa diubah. Variabel pada mekanisme di atas telah tertutup (*close covered*), yang berarti variabel tersebut berada di dalam *closure*.