**Paradigma Functional Programming**

Seperti yang sudah disampaikan di awal, Dart adalah bahasa yang mendukung *multiparadigm*. Artinya selain merupakan bahasa pemrograman berorientasi objek, penulisan *syntax* Dart juga menggunakan gaya *functional programming*.

*Functional programming* adalah paradigma pemrograman di mana proses komputasi didasarkan pada fungsi matematika murni. *Functional programming* (FP) ditulis dengan gaya deklaratif yang berfokus pada “*what to solve*” dibandingkan “*how to solve*” pada gaya imperatif.

Berikut ini beberapa konsep dan karakteristik *functional programming*:

**Pure functions**

*Pure functions* berarti sebuah fungsi bergantung pada argumen atau parameter yang dimasukkan ke dalamnya. Sehingga pemanggilan fungsi dengan nilai argumen yang sama akan selalu memberikan hasil yang sama pula. Contohnya pada fungsi sum() berikut nilai yang dikembalikan akan bergantung pada argumen yang diberikan.

1. int sum(int num1, int num2) {
2. return num1 + num2;
3. }

**Recursion**

Pada *functional programming* tidak ada konsep perulangan seperti for atau while. Iterasi pada *functional programming* dilakukan melalui rekursi atau pemanggilan fungsi dari fungsi itu sendiri, hingga mencapai kasus dasar.

1. int fibonacci(n) {
2. if (n <= 0) {
3. return 0;
4. } else if(n == 1) {
5. return 1;
6. } else {
7. return fibonacci(n - 1) + fibonacci(n - 2);
8. }
9. }

**Immutable variables**

Variabel pada *functional programming* bersifat *immutable*, artinya kita tidak bisa mengubah sebuah variabel ketika sudah diinisialisasi. Alih-alih mengubah nilai variabel, kita bisa membuat variabel baru untuk menyimpan data. Mekanisme ini bertujuan agar kode kita menjadi lebih aman karena *state* dari aplikasi tidak akan berubah sepanjang aplikasi berjalan.

1. var x = 5;
2. x = x + 1; // Contoh variable yang tidak immutable

**Functions are first-class citizen and can be higher-order**

Maksud dari *function* merupakan *first-class citizen* adalah bahwa *function* berlaku sama seperti komponen pemrograman yang lain. Sebuah fungsi bisa dimasukkan ke variabel menjadi parameter dalam suatu fungsi dan juga menjadi nilai kembalian pada fungsi. *Higher order functions* adalah fungsi yang mengambil fungsi lain sebagai argumen dan juga dapat mengembalikan fungsi.

Pada modul ini kita akan mempelajari bagaimana penulisan gaya *functional* dengan bahasa Dart.