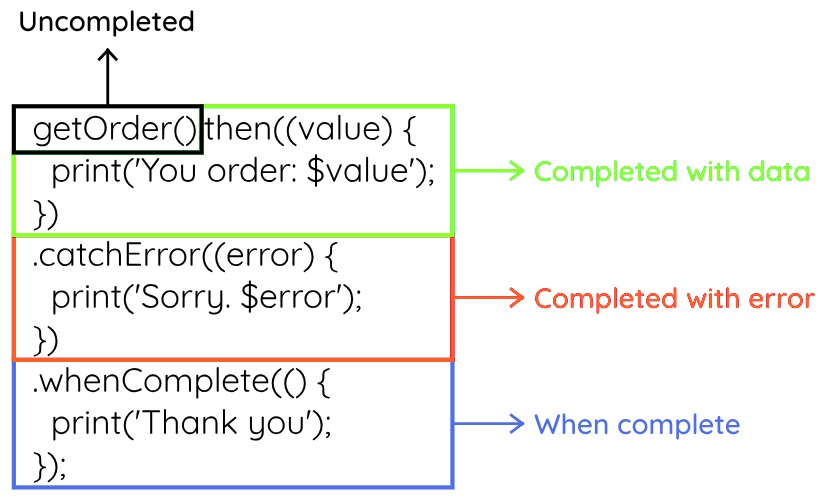
**Future dengan async-await**

Pada materi sebelumnya kita telah mempelajari *Future* untuk berurusan dengan proses *asynchronous*. Seperti yang kita tahu, penulisan kode *asynchronous* akan sedikit berbeda dengan proses *synchronous*. Contohnya program pesan kopi kita sebelumnya jika dituliskan secara *asynchronous* akan seperti berikut:

[](https://www.dicoding.com/academies/191/tutorials/7682?from=7678)

Dart memiliki *keyword* async dan await yang merupakan sebuah alternatif supaya kita bisa menuliskan proses *asynchronous* layaknya proses *synchronous*. Bagaimana caranya?

Dengan gaya penulisan *synchronous*, kira-kira kode program pesan kopi kita akan seperti berikut:

1. void main() {
2. print('Getting your order...');
3. var order = getOrder();
4. print('You order: $order');
5. }

Namun ketika kode di atas dijalankan hasilnya tidak akan sesuai yang kita harapkan karena fungsi getOrder() merupakan objek Future.

1. Getting your order...
2. You order: Instance of 'Future<String>'

Output ini disebabkan karena fungsi main() masih merupakan fungsi *synchronous*. Untuk mengubahnya menjadi fungsi *asynchronous* tambahkan *keyword* async sebelum *function body*.

1. void main() **async** { … }

Kemudian tambahkan *keyword* await sebelum fungsi yang mengembalikan nilai Future.

1. var order = **await** getOrder();

Dari perubahan di atas yang memanfaatkan *async-await* kita telah berhasil menuliskan proses *asynchronous* dengan gaya *synchronous*.

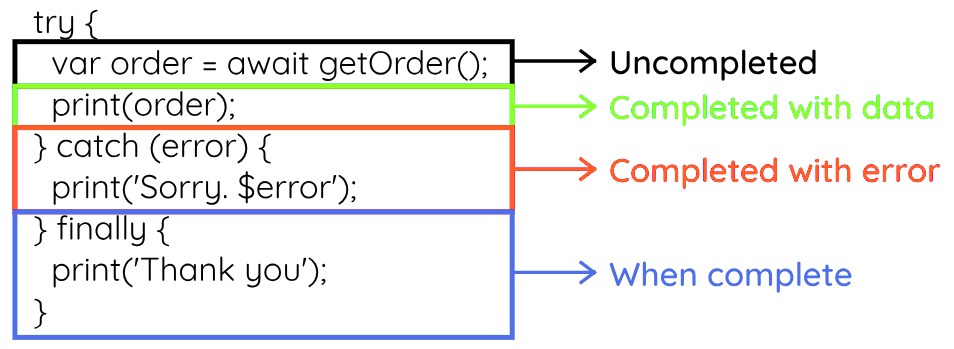
1. void main() **async** {
2. print('Getting your order...');
3. var order = **await** getOrder();
4. print('You order: $order');
5. }

8. /\*
9. Output :
10. Getting your order...
11. You order: Coffee Boba
12. \*/

Lalu bagaimana menangani ketika terjadi eror? Caranya cukup sederhana yaitu dengan memanfaatkan *try-catch*:

1. void main() async {
2. print('Getting your order...');
3. try {
4. var order = await getOrder();
5. print('You order: $order');
6. } catch (error) {
7. print('Sorry. $error');
8. }
9. }

Begitu juga seperti yang telah disebutkan, method whenComplete() bisa digantikan dengan blok finally. Sehingga keseluruhan kode akan menjadi seperti berikut:

[](https://www.dicoding.com/academies/191/tutorials/7682?from=7678)