



DISKUSIKAN MATERI

LAPORKAN MATERI

Daftar Modul

Masukkan kata kunci

Persetujuan Hak Cipta ✓

Modul 1: Introduction Course (Gratis) ✓

Prasyarat Kelas (Gratis) ✓

Apa yang Akan Kita Pelajari? (Gratis) ✓

Tools Requirement ✓

Modul 2: ECMAScript 6 (Gratis) ✓

Running Snippet Code (Gratis) ✓

Persiapan Project Latihan (Gratis) ✓

onRejected with Catch Method

Salah satu cara menulis kode yang baik adalah mengikuti prinsip yang disebut *separation of concerns* yang artinya pemisahan masalah. Pemisahan masalah berarti mengorganisasikan kode ke dalam bagian-bagian yang berbeda berdasarkan tugas tertentu. Hal ini akan memudahkan kita kelak mencari kode yang salah jika aplikasi tidak bekerja dengan baik.

Perlu diketahui bahwa method `.then()` akan mengembalikan nilai promise yang sama dengan ketika objek promise itu dipanggil. Melalui sifatnya ini, daripada kita menetapkan logika `resolve` dan `reject` pada satu method `.then()`, kita dapat memisahkan kedua logika tersebut menggunakan masing-masing method `.then()` seperti ini:

```
1. const makeCoffee = new Promise(executorFunction);
2. makeCoffee
3.   .then(handlerSuccess)
4.   .then(null, handlerRejected);
5.
6. /* output:
7.  Mesin Kopi tidak bisa digunakan!
8.  */
```

Namun untuk menetapkan *rejected handler* kita perlu memberikan nilai `null` pada parameter method `.then()`. Well.. hal ini sedikit merepotkan bukan? Solusinya kita dapat menggunakan method lain, yakni `.catch()`.

Method `.catch()` mirip seperti `.then()`. Namun method ini hanya menerima satu parameter function yang digunakan untuk *rejected handler*. Method `.catch()` ini akan terpanggil bilamana objek promise memiliki kondisi *onRejected*. Berikut contoh penggunaan dari method `.catch()`:

```
1. const makeCoffee = new Promise(executorFunction);
2. makeCoffee
3.   .then(handlerSuccess)
4.   .catch(handlerRejected);
5.
6. /* output:
7.  Mesin Kopi tidak bisa digunakan!
8.  */
```

Dengan menggunakan method `.catch()`, kita dapat menerapkan prinsip *separation of concerns* sekaligus membuat kodenya lebih rapi.

← KEMBALI KE MATERI SEBELUMNYA

LANJUTKAN KE MATERI BERIKUTNYA →



image
click bisa
belum muncul

image
click bisa
belum muncul

PERUSAHAAN

Tentang Kami

Blog

Berita Terbaru



PROGRAM

Academy

Challenge

Event

Job

Rewards

SUPPORT

Bantuan

FAQ

Hubungi Kami



DIBANTU