

2022

PBW

Modul 4 – JavaScript (3)

Silakan buat halaman web sesuai petunjuk. Gunakan template yang telah disediakan.



PETUNJUK Pengerjaan

1. Rename folder TEMPLATE menjadi M04xxyyy.
2. Gunakan template yang sudah diberikan.
3. JANGAN MENGUBAH NAMA FILE!
4. Upload folder (di zip) dan file txt Anda ke IDE sesuai waktu pengumpulan yang tertera.

Asynchronous JS

Asynchronous JavaScript adalah cara memproses kode JS secara asynchronous (tidak selalu terurut). Untuk melatih pemanfaatan fungsi Promise, async, dan await, maka silahkan mengerjakan modul berikut. Pada *folder* TEMPLATE telah disediakan 2 *file template* yaitu halaman html kosong yang tidak perlu diubah dan 1 file javascript yang perlu dimodifikasi.

Pertama-tama, silahkan buka *file script.js* yang berisi 2 scope fungsi yaitu fungsi `dumbFunction` dan “tester” dalam bentuk IIFE sebagai berikut:

- Fungsi **`dumbFunction`** merupakan fungsi yang jika dipanggil, hanya punya 10% kemungkinan mengeluarkan fungsi yang benar. Artinya lebih banyak kemungkinan gagalnya. Ketika gagal akan mengembalikan promise reject.
- Fungsi “tester” ini tidak boleh diubah dan merupakan kode memanggil fungsi yang perlu dibuat.

Tugas anda adalah mengimplementasikan 2 buah fungsi sebagai berikut:

- Fungsi **`retryUntilSucceed`**
 - Fungsi ini akan memanggil fungsi `dumbFunction` dengan parameter “success 1” terus menerus sampai fungsi tersebut mengembalikan kembali parameter tersebut.
 - Fungsi ini pasti akan berhenti dikarenakan `Math.random` mengembalikan rentang nilai 0 – 1.
- Fungsi **`retryNTimes`**
 - Mempunyai 2 parameter tambahan yaitu `n` dan `retryInterval` (dalam ms).
 - Fungsi ini akan memanggil fungsi `dumbFunction` dengan parameter “success 2” maksimal sebanyak `n` kali.
 - Jika dalam percobaan tersebut berhasil mengeluarkan nilai. Maka nilai ini yang dikembalikan. Akan tetapi jika sampai `n` kali tidak juga mengeluarkan nilai, maka program akan berhenti dan tidak mengeluarkan apa-apa.
 - Setiap kali fungsi `dumbFunction` gagal, maka perlu jeda selama `retryInterval` sampai pemanggilan berikutnya.

Contoh (n = 3, retryInterval = 1000):

Pemanggilan pertama -> gagal

Tunggu 1 detik (1000ms)

Pemanggilan kedua -> gagal

Tunggu 1 detik

Pemanggilan ketiga -> berhasil

Mengembalikan "success 2"

Manfaatkan penggunaan promise, async, dan await yang sudah diajarkan.

Kumpulkan dalam bentuk zip dengan nama M04xxyyy.zip

= SELAMAT MENGERJAKAN =