DISKUSIKAN MATERI

LAPORKAN MATERI

Daftar Modul

Masukkan kata kunci

Module

Potongan kode dalam materi ini:

- Export dan Import dalam Node.js:
  https://repl.it/@dicodingacademy/163-02-export-import-nodejs?lite=true
- Export dan Import banyak Nilai dalam Node.js:
  https://repl.it/@dicodingacademy/163-02-multiple-export-nodejs?lite=true
- Export dan Import dalam ES6:
  https://repl.it/@dicodingacademy/163-02-exporting-importing-single-value-es6?lite=true
- Export dan import banyak Nilai dalam ES6:
  https://repl.it/@dicodingacademy/163-02-exporting-importing-multiple-value-es6?lite=true

Jika aplikasi kita akan terus berkembang, tentu kita tidak bisa menuliskan seluruh kode hanya pada satu berkas JavaScript. Ketika kita membaginya menjadi beberapa berkas JavaScript, di situlah kita perlu membuat sebuah modul JavaScript. Apa tujuannya? Tak lain untuk menghubungkan berkas JavaScript yang terpisah agar dapat saling digunakan satu sama lain.

Awalnya, tidak ada cara untuk melakukan modular menggunakan sintaks pada JavaScript.

Awalnya JavaScript terlahir tanpa adanya fitur standar import dan export (Fitur tersebut dapat diterapkan namun perlu menggunakan library tambahan). Karena tidak terdapat fitur standar untuk import dan export, untuk menggunakan banyak berkas JavaScript di browser kita lakukan dengan menggunakan tag <script> pada berkas HTML.

```
<script src="src/script/data/clubs.js"></script>
<script src="src/script/data/data-source.js"></script>
<script src="src/script/view/main.js"></script>
<script src="app.js"></script>
```

Menggunakan tag `<script>` memang simpel, namun urutan dari penulisan tag tersebut merupakan hal yang vital. Karena `app.js` tidak akan bekerja jika dituliskan di atas `main.js`, karena `app.js` membutuhkan kode `main.js` tersedia terlebih dahulu.

Belum lagi berkas yang kita buat akan semakin banyak dan kita perlu menuliskan tag `<script>` nya kembali dengan urutan yang benar. Hal klasik seperti ini sudah seharusnya diubah dengan pendekatan yang lebih modern.

Dengan menggunakan module, kita dapat melakukan `import` maupun `export` variabel, class, object, array, atau apapun itu. JavaScript module bersifat *reusable* sehingga dapat digunakan pada banyak aplikasi yang kita buat.

Pada materi kali ini, kita akan belajar bagaimana cara implementasi module pada Node.js dan ES6.

KEMBALI KE MATERI SEBELUMNYA LANJUTKAN KE MATERI BERIKUTNYA