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)

Exporting and importing single value (Default Export)

Pada Node.js sebelumnya tidak ada perbedaan antara exporting multiple value dan single value. Semua nilai yang akan diekspor, dijadikan nilai dari properti `module.exports`. Pada ES6 module, jika kita hanya mengekspor satu nilai pada sebuah berkas JavaScript baik itu primitive value, function, array, object ataupun class, kita gunakan keyword export default. Contohnya seperti ini:

```
const coffeeStock = {
 arabica: 100,
 robusta: 150,
 liberica: 200
}

export default coffeeStock;
```

Lalu bagaimana cara untuk mengimpor nilainya? Kita dapat melakukannya dengan menggunakan keyword `import` ... `from` seperti berikut ini:

```
import coffeeStock from "./state.js";
```

Berbeda dengan gaya Node.js, kita gunakan keyword `import` untuk menggantikan `const`, `let`, ataupun `var` dalam mendeklarasi variabel yang diimpor. Lalu di sana juga kita menggunakan from dalam menspesifikasikan lokasi berkas JavaScript-nya.

Ketika menggunakan `export default`, kita dapat menggunakan penamaan apa saja ketika mendeklarasikan variabel dalam mengimpor nilainya.

```
// Kita dapat mengubah penamaan coffeeStock sesuai kebutuhan kita.
import stock from "./state.js";
```

Hal tersebut aman untuk dilakukan karena dengan menggunakan ***export default***, dapat dipastikan hanya satu nilai yang diekspor pada satu berkas JavaScript.

Setelah kita berhasil mendapatkan nilai yang diekspor, kita dapat menggunakan nilainya layaknya variabel lokal biasa.

index.html **app.js** state.js

```
import coffeeStock from "./state.js";


const displayStock = stock => {
 const coffeeStockListElement = document.querySelector("#coffee-stock-list");
 for(const type in stock) {
   const coffeeStockItemElement = document.createElement("li");
   coffeeStockItemElement.innerText = `${type}: ${stock[type]}`;
   coffeeStockListElement.appendChild(coffeeStockItemElement);
 }
}


displayStock(coffeeStock);
```

Exporting and Importing Multiple Value (Named Export/Import)

Jika sebelumnya kita hanya melakukan ekspor satu nilai pada berkas JavaScript menggunakan `default export`, pada materi kali ini kita akan membahas bagaimana cara melakukan ekspor banyak nilai dalam satu berkas JavaScript dengan menggunakan ES6.

Named export digunakan untuk mengekspor banyak nilai dalam berkas JavaScript. Cara kerjanya mirip seperti pada Node.js. Nilai yang akan diekspor dituliskan di dalam objek literals, seperti ini:

```
const coffeeStock = {
 arabica: 100,
 robusta: 150,
 liberica: 200
}

const isCoffeeMakerReady = true;

export { coffeeStock, isCoffeeMakerReady };
```

Lalu untuk mendapatkan nilai yang diekspor menggunakan named export, kita gunakan teknik destructuring object.

```
import { coffeeStock, isCoffeeMakerReady } from "./state.js";

console.log(coffeeStock);
```

```
console.log(isCoffeeMakerReady);

/* output:
{ arabica: 100, robusta: 150, liberica: 200 }
true
*/
```

Karena named import menggunakan teknik destructuring object untuk mendapatkan nilainya, maka pastikan penamaan variabel sesuai dengan nama variabel yang diekspor. Jika terjadi kesalahan penulisan, maka akan terjadi eror seperti berikut:

```
import { stock, isCoffeeMakerReady } from "./state.js";

/* output:
SyntaxError: The requested module './state.js' does not provide an export named 'stock'
*/
```

Namun jika kita tetap ingin mengubah penamaan variabel dari named import, kita bisa melakukannya dengan menambahkan keyword as setelah penamaan variabelnya.

```
import { coffeeStock as stock, isCoffeeMakerReady } from "./state.js";

console.log(stock);
console.log(isCoffeeMakerReady);

/* output:
{ arabica: 100, robusta: 150, liberica: 200 }
true
*/
```

Setelah kita berhasil mendapatkan nilai yang diekspor, kita dapat menggunakan nilainya layaknya variabel lokal biasa.

index.html **app.js** state.js

```
import { coffeeStock, isCoffeeMakerReady } from "./state.js";


const displayStock = stock => {
  const coffeeStockListElement = document.querySelector("#coffee-stock-list");
  for (const type in stock) {
    const coffeeStockItemElement = document.createElement("li");
    coffeeStockItemElement.innerText = `${type}: ${stock[type]}`;
    coffeeStockListElement.appendChild(coffeeStockItemElement);
  }
}


const coffeeOrder = (type, miligrams) => {
  return new Promise((resolve, reject) => {
    if (isCoffeeMakerReady) {
      if (coffeeStock[type] >= miligrams) {
        resolve("Kopi berhasil dipesan!");
      } else {
        reject("Maaf Stock kopi habis!")
      }
    } else {
      reject("Maaf mesin sedang rusak!")
```

← KEMBALI KE MATERI SEBELUMNYA

LANJUTKAN KE MATERI BERIKUTNYA →