



IBTISSAM OULAHCHOUM

JavaScript for Beginners Series

(مع شرح بالدارجة)

Part13





IBTISSAM OULAHCHOUM

ES6 Modules (Import & Export)





Technical Explanation

- As your applications grow, you can't keep all your code in one giant file. Modules allow you to split your code into separate, reusable files. You can then share functions, variables, or classes between these files. This is the foundation of how modern frameworks like React organize code into components.
 - export: The keyword used to make variables or functions available to other files.
 - import: The keyword used to bring those exported variables or functions into the current file.



Technical Explanation

- فاش كيكبر البرنامج دياك، مستحيل تخلي الكود كامل فملف (file) واحد. ال Modules كيخليوك تقسم الكود دياك على بزاف ديال الملفات لي تقدر تعاود تستعملهم. هكذا كتقدر تشارك الدوال (functions) والمتغيرات (variables) بين هاد الملفات. هاد المبدأ هو الأساس لي خدامين بيه ال frameworks بحال React باش كينظمو الكود على شكل components.
- export (تصدير): هي الكلمة لي كتستعمل باش تخلي شي متغير أو دالة متاحة للاستعمال بالملفات الأخرى.
- import (إستيراد): هي الكلمة لي كتستعمل باش تجيب دوك المتغيرات أو الدوال لداخل الملف لي نتا خدام فيه.



Types of Exports

There are two main types of exports:

1. Named Exports: You can export multiple variables/functions from a single file. When you import them, you must use the exact same name inside curly braces {}

- كتقدر تصدر بزاف ديال الحوايج من ملف واحد. فاش كتبغي تجيبهم، خاصك تستعمل نفس التسمية ديالهم وسط {}.

2. Default Export: You can only have one default export per file. It's used for the "main" thing you want to export. When you import it, you can give it any name you want.

- كتقدر دير غير وحدة فكل ملف. كتستعملها للحاجة الرئيسية لي باغي تصدرها. فاش كتبغي تجيبها، كتقدر تعطيه أي تسمية بغيتي.



```
// 📁 helpers.js (This is one file)

// Named Export
export const PI = 3.14;

// Named Export
export function add(a, b) {
  return a + b;
}

// Default Export
export default function sayHello(name) {
  return `Hello, ${name}`;
}

// -----

// 📁 main.js (This is another file)

// Import the default export and give it a name
import greet from './helpers.js';

// Import the named exports using their exact names
import { PI, add } from './helpers.js';

console.log(greet("Amina")); // Output: Hello, Amina
console.log(`The value of PI is ${PI}`); // Output: The value of PI is 3.14
console.log(`2 + 3 = ${add(2, 3)}`); // Output: 2 + 3 = 5
```



QUESTION FOR YOU:

- Imagine you have a file config.js. You want to export a DATABASE_URL string and an API_KEY string. Which type of export (Named or Default) would be more appropriate here, and why?



IBTISSAM OULAHCHOUM

See you in other parts :)

