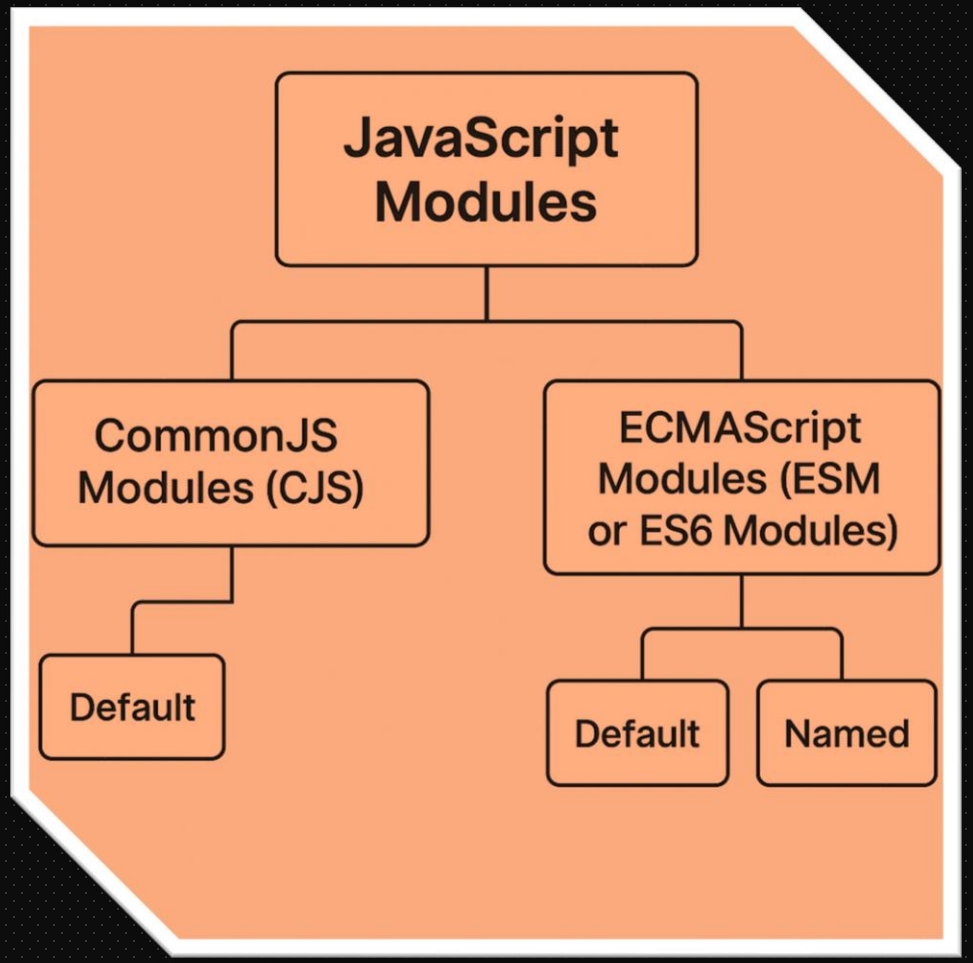
# Modules




- Modules are a way to organize code into reusable files.
- They help you split your code into smaller, manageable parts, making it easier to maintain, reuse, and avoid name collisions.

# CommonJS Modules (CJS)

## Export single item

```
// utils.js
function greet(name) {
    console.log(`Hello, ${name}`);
}
module.exports = greet;
```

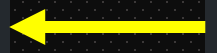
```
// script.js
const greetFunc = require('./utils');
greetFunc('Manas');
```

Note2:

- using file extension while importing is standard practice.
- Example: const alpha = require('./utils.**js**')

## Export multiple item

```
// script.js
const math = require('./math');
let result1 = math.add(2, 3); // 5
let result2 = math.subtract(10, 3); // 7
console.log(result1, result2)
```



```
// math.js
function add(a, b) {
    return a + b
}

const subtract = (a, b) => {
    return a - b;
}

module.exports = {
    add,
    subtract
};
```

❑ In CommonJS, “exports” is a shorthand alias for module.exports.

```
// In CommonJS, "exports" is a shortcut for module.exports.

exports.add = (a, b) => a + b;
exports.sub = (a, b) => a - b;

// The above code is similar to below code 👇👇

const add = (a, b) => a + b;
const sub = (a, b) => a + b;

module.exports = {
    add,
    sub,
}

// ❌ If you reassign module.exports, it breaks the link with exports.
```

# ES6 Modules (modern JavaScript)

## Default Export

```
// math.js
export default function add(a, b) {
    console.log(a * b);
}
```

```
// script.js
import defaultFunc from './math.js';

defaultFunc(5,2); // 10
```

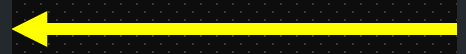
Note2:

- ❑ Make Sure You're Using type: "module" in Node.js

# Named Export

```
// script.js
import {
    subtract,
    multiply,
    divide,
    modulo
} from './math.js';

subtract(5, 2);
multiply(5, 2);
divide(5, 2);
modulo(5, 2);
```



```
// math.js
export function subtract(a, b) {
    console.log(a - b);
}

export const multiply = (a, b) => {
    console.log(a * b);
}

function divide(a, b) {
    console.log(a / b);
}

function modulo(a, b) {
    console.log(a % b);
}

export {
    divide, modulo
}
```