📝 Summary - NodeJS Modules

📚 What is module in NodeJS?

- Module is a piece of code that performs a specific task in our node application.
- For example, for building a Car. We have many parts like engine, wheels, seats etc. This each part is built separately and also it can be replaced or repaired without affecting the entire car. So engine, wheels, seats are modules of car.
- Each module represents a different part of our application, and these modules can work together to form a complete application.

🔢 There are 3 types of modules

1. Local Modules: which we will create for our own application. (i.g. db, routes...)
2. Core(Built-in) Modules: which we get with node js. (i.g. http, fs, os...)
3. Third-Party Modules: which are created & published by other developers. (i.g. express, mongoose...)

✍️ Create own Module

- To create our own module, we have to create a simple JS file in our node application. Define some variables in it. That’s it.
- Now when we define a variable in the module, that variable is only accessible within that module.
- We have to export that variable from that module, so we can access those variable in other modules.
- For exporting the module:

```
const greeting = "Happy Coding by Code Bless You";

// export within object
module.exports.greeting = greeting

// export as default
module.exports = greeting
```

🔑 Accessing Module in another module

- For accessing variables of module in another modules we can use import statement.

```
// for accessing within object
import { greeting } from './greeting.js'

// for deafult export
import greeting from './greeting.js'
```

🔧 Core Modules

- There are many core modules of node js which we can use in our application.
- Path module is used for working with file and directory paths. In path module we have many methods like `path.parse(__filename)` for getting some information about that file.
- Also we have `path.join("path1", "path2")` for join two or more paths.
- Os module is used for getting the information about operating system like which operating system they are using and how much memory they have etc. Methods - `os.platform()` , `os.totalMem()` , `os.freeMem()`
- Next we have fs module or we can say file system module which is used to work with files like write the file, read the file, delete the file etc.

- Http module allow us to create a http server and also for handling different http requests.

🌐 Create server using http module

```
const http = require("http");

const server = http.createServer((req, res) => {
  res.write("Hello World!");
  res.end();
});

server.listen(3000, () => {
  console.log("Server start listening on post 3000");
});
```

🚪 Multiple routes of Request

```
const server = http.createServer((req, res) => {
  if (req.url === "/") {
    res.write("Hello World!");
  } else if(req.url === "/about") {
    res.write("This is about route.");
  } else {
    res.write("Route not found!")
  }

  res.end();
});
```

- This is not practical because if we have 10 or 20 routes then we have to write else if condition for 18 more routes.
- This will make our code messy and also hard to manage.

The ultimate Node JS Course ~ Code Bless You❤️