Text Instructions: Module 7  
  
  
GitHub Link for Node.js:  
  
<https://github.com/Apollo-Level2-Web-Dev/batch4-learning-node>

GitHub Link for Express.js:  
  
<https://github.com/Apollo-Level2-Web-Dev/batch4-project-1>

7-1 What is Nodejs , a high level overview of node.js

7-2 What is module, commonjs vs es

Create: index.js , local-1.js

index.js:

node .\local-1.js

{

id: '.',

path: 'E:\\web\\Programming Hero\\Level 2\\Mission 01-Be A Typescript Technocrat\\Module 7-Mastering The Foundation of Express\\learning-node',

exports: {},

filename: 'E:\\web\\Programming Hero\\Level 2\\Mission 01-Be A Typescript Technocrat\\Module 7-Mastering The Foundation of Express\\learning-node\\local-1.js',

loaded: false,

children: [],

paths: [

'E:\\web\\Programming Hero\\Level 2\\Mission 01-Be A Typescript Technocrat\\Module 7-Mastering The Foundation of Express\\learning-node\\node\_modules',

'E:\\web\\Programming Hero\\Level 2\\Mission 01-Be A Typescript Technocrat\\Module 7-Mastering The Foundation of Express\\node\_modules',

'E:\\web\\Programming Hero\\Level 2\\Mission 01-Be A Typescript Technocrat\\node\_modules',

'E:\\web\\Programming Hero\\Level 2\\node\_modules',

'E:\\web\\Programming Hero\\node\_modules',

'E:\\web\\node\_modules',

'E:\\node\_modules'

],

[Symbol(kIsMainSymbol)]: true,

[Symbol(kIsCachedByESMLoader)]: false,

[Symbol(kIsExecuting)]: true

}

Index.js:

const add = require("./local-1")

console.log(add(2, 3));

Now,

Local-1.js:

const add = (param1, param2) => param1+ param2

const a = 10;

module.exports = {

    a,

    add

}

// console.log(module);

Index.js:

const add = require("./local-1")

console.log(add); // { a: 10, add: [Function: add] }

const add = require("./local-1")

console.log(add.add(2,3));

const { a, add } = require("./local-1")

console.log(a);

console.log(add(2, 3));

Now,

Local-1.js, Local-2.js, index.js

Local-1.js:

const add = (param1, param2) => param1+ param2

const a = 10;

module.exports = {

    a,

    add

}

Local-2.js:

const add = (param1, param2, param3) => param1 + param2 + param3

const a = 20

module.exports = {

    a,

    add

}

Index.js:

const { a, add } = require("./local-1")

const { a, add } = require("./local-2")

console.log(a);

console.log(add(2, 3));

console.log(a2);

console.log(add2(2, 3, 7));

// error

But, Index.js:

const { a, add } = require("./local-1")

const { a:a2, add:add2 } = require("./local-2")

console.log(a);

console.log(add(2, 3));

console.log(a2);

console.log(add2(2, 3, 7));

// OK , no error

Docs : <https://nodejs.org/docs/latest/api/path.html>

**7-3 File System Module , synchronous vs asynchronous**

Google > node js . > file system > search ( Ctrl+F ): fs.readfilesync

<https://nodejs.org/docs/latest/api/fs.html>

google > lorem ipsum generator >

const fs = require('fs') // fs-module

// reading a file text

const readText = fs.readFileSync('./texts/read.txt')

console.log(readText);

output : error

Again,

const fs = require('fs') // fs-module

// reading a file text

const readText = fs.readFileSync('./texts/read.txt', 'utf-8')

console.log(readText);

output -> ok

// writing a text

const writtenText = fs.writeFileSync('./texts/write.txt', readText+ 'This is my written text')

console.log(writtenText);

search : fs.readfile

const fs = require('fs')

// reading text asynchronously

fs.readFile('./texts/read.txt', 'utf-8', (err, data) => {

    if (err) {

        throw Error('Error reading text')

    }

    // console.log(data);

    // writing text asynchronously

    fs.writeFile('./texts/read2.text', data, 'utf-8', (err) => {

        if(err){

            throw Error("Error writing data")

        }

    })

})

//console.log('testing asynchronous'); // after uncommeting this the output only this text