1\_ primeiro Programa

console.log('Hello World Node!')

2\_Utilizando\_Modulos

const fs = require('fs')

fs.readFile('arquivo.txt', 'utf8', (err, data) => {

  console.log(data)

})

E criar o arquivo arquivo.txt

3\_Criando um programa e rodando ele (Pode até puxar uma questão mais elaborada)

const a = 10;

const b = 5;

console.log(a + b);

1\_Modulo interno

Meu\_modulo.js

module.exports = {

  soma(a, b) {

    console.log(a + b)

  },

}

Index.js

const meuModulo = require('./meu\_modulo')

const soma = meuModulo.soma

soma(2, 3)

2\_export\_import

Index.mjs

import soma from "./meu\_modulo.mjs";

soma(2, 3);

meu\_modulo.mjs

function soma(a, b) {

  console.log(a + b);

}

export default soma;

3\_core\_module

const path = require("path");

const extension = path.extname("arquivo.php");

console.log(extension);

4\_Ler\_argumentos

// nome=Matheus

console.log(process.argv);

const args = process.argv.slice(2);

console.log(args);

const nome = args[0].split("=")[1];

console.log(nome);

5\_Modulos\_externos

Index.js

Npm install minimist

// --nome=Matheus --idade=30

const minimist = require("minimist");

const args = minimist(process.argv.slice(2));

console.log(args);

const nome = args["nome"];

const idade = args["idade"];

console.log(nome);

console.log(idade);

6\_pratica\_com\_args

Index.js

// --a=10 --b=20

// externo

const minimist = require("minimist");

// interno

const meuModulo = require("./meu\_modulo");

const soma = meuModulo.soma;

const args = minimist(process.argv.slice(2));

const a = args["a"];

const b = args["b"];

soma(a, b);

meu\_modulo.js

module.exports = {

  soma(a, b) {

    console.log(a + b)

  },

}

7\_explorando\_console

// mais de uma valor

const x = 10;

const y = "Matheus";

console.log(x, y);

// contagem de impressões

console.count("O valor de x é: " + x + " -> contagem:");

console.count("O valor de x é: " + x + " -> contagem:");

console.count("O valor de x é: " + x + " -> contagem:");

console.count("O valor de y é: " + y + " -> contagem:");

// variável entre string

console.log("O nome dele é %s", y);

// limpando console

setTimeout(() => {

  console.clear();

}, 2000);

8\_melhorando\_visual

Npm install chalk

const chalk = require("chalk");

const nota = 8;

if (nota >= 7) {

  console.log(chalk.green.bold("Parabéns, você passou!"));

} else {

  console.log(chalk.bgRed.black("Você precisa fazer a prova final!"));

}

9\_lendo\_input

const readline = require("readline").createInterface({

  input: process.stdin,

  output: process.stdout,

});

readline.question(`Qual a sua linguagem preferida? `, (language) => {

  console.log(`A minha linguagem preferida é: ${language}`);

  readline.close();

});

10\_abstração\_input

npm install inquirer@8.1.2

const inquirer = require('inquirer')

inquirer

  .prompt([

    { name: 'p1', message: 'Qual a primeira nota?' },

    { name: 'p2', message: 'Qual a segunda nota?' },

  ])

  .then((answers) => {

    console.log(answers)

    const media = (parseInt(answers.p1) + parseInt(answers.p2)) / 2

    console.log(`A média do aluno é ${media}`)

  })

  .catch((err) => {

    console.log(err)

  })

11\_sync\_async

assincrono

const fs = require("fs");

console.log("Início");

fs.writeFile("arquivo.txt", "Oi", function (err) {

  setTimeout(function () {

    console.log("Arquivo criado!");

  }, 1000);

});

console.log("Fim");

síncrono

const fs = require("fs");

console.log("Início");

fs.writeFileSync("arquivo.txt", "Oi");

console.log("Fim");

12\_erros

Throws.js

const x = "10";

if (!Number.isInteger(x)) {

  throw new Error("O valor de x não é um número inteiro");

}

Try\_catch.js

const x = 10;

try {

  x = 2;

} catch (err) {

  console.log(`Erro: ${err}`);

}

13\_Resposta\_tarefa\_2

const inquirer = require('inquirer')

const chalk = require('chalk')

inquirer

  .prompt([

    { name: 'nome', message: 'Qual o seu nome?' },

    { name: 'idade', message: 'Qual a sua idade?' },

  ])

  .then((answers) => {

    const response = `O nome do usuário é ${answers.nome} e ele tem ${answers.idade} anos`

    console.log(chalk.bgYellow.black(response))

  })

  .catch((err) => {

    console.log(err)

  })