Full Stack Assignment - 1 (Node JS)

1) Create a function to Print prime numbers in an interval of 10 to 50.

-> function printPrimeNumbers(min, max)

{

for (let i = min; i <= max; i++)

{

      if (isPrime(i))

{

        console.log(i);

      }

    }

}

function isPrime(num)

{

for (let i = 2; i < Math.sqrt(num); i++)

{

      if (num % i === 0)

{

        return false;

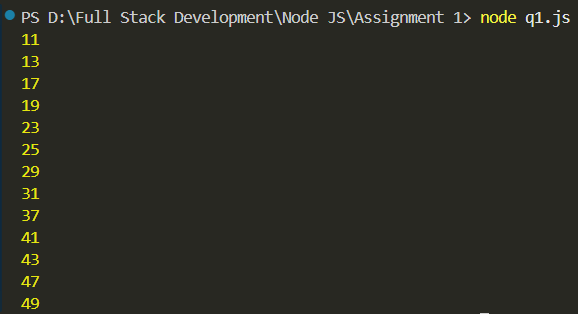
      }

    }

    return num > 1;

 }

 printPrimeNumbers(10, 50);

-> Output:

2) Create a function to Check if a number is Armstrong number.

-> prompt = require('prompt-sync') ()

function checkArmstrong(num)

{

let sum = 0;

  let rem = 0;

  let temp = num;

  while (num > 0)

  {

    rem = num % 10;

    sum = sum + rem \* rem \* rem;

    num = Math.floor(num / 10);

  }

  if (sum === temp)

{

    return `${sum} is an armstrong number`;

  }

else

{

    return `${sum} is not an armstrong number`;

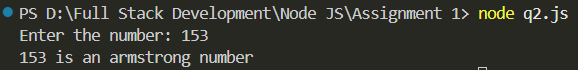
  }

}

num = parseInt(prompt("Enter the number: "));

armstrong = checkArmstrong(num)

console.log(armstrong);

-> Output:

3) Create a function to Check if a number is Perfect number.

-> prompt = require('prompt-sync') ()

function isPerfectNumber(num)

{

var temp = 0;

    for(var i=1;i<=num/2;i++)

{

        if(num%i === 0)

{

            temp += i;

        }

    }

    if (temp === num && temp != 0)

{

        return `${temp} is a perfect number`;

    }

    else

{

        return `${temp} is not a perfect number`;

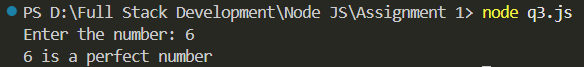
    }

}

num = parseInt(prompt("Enter the number: "))

perfect = isPerfectNumber(num);

console.log(perfect);

-> Output:

4) Create a function to Count occurrence of character in a string.

-> prompt = require('prompt-sync') ()

function countChar(str, char)

{

let count = 0;

    for(let i = 0; i < str.length; i++)

{

      if(str.charAt(i) === char)

{

        count++;

      }

    }

    return `The occurence of ${char} in ${str} is ${count} times`;

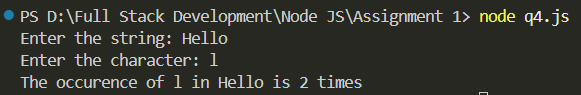
  }

string = prompt("Enter the string: ")

char = prompt("Enter the character: ")

count = countChar(string, char);

console.log(count);

-> Output:

5) Create a function to Create array, sort array, concatenate two arrays.

-> function createArray(length)

{

let arr = new Array(length);

    const prompt = require('prompt-sync')();

    for (let i = 0; i < length; i++)

{

        val = parseInt(prompt("Enter the value: "));

        arr[i] = val;

    }

    return arr;

}

function sortArray(arr)

{

return arr.sort((a, b) => a - b);

}

function concatArrays(arr1, arr2)

{

return (arr1.concat(arr2));

}

const prompt = require('prompt-sync')();

length = parseInt(prompt("Enter the length: "));

arr = createArray(length);

console.log(arr);

sortedArr = sortArray(arr);

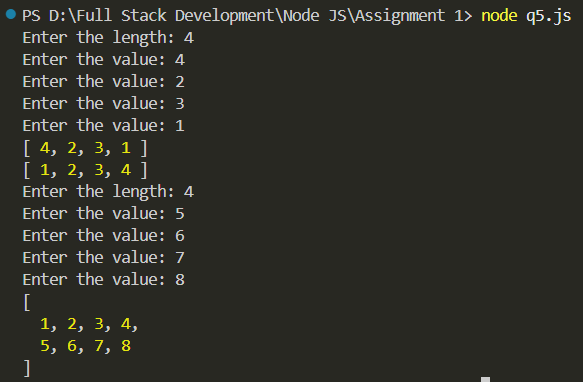
console.log(sortedArr);

length = parseInt(prompt("Enter the length: "));

arr2 = createArray(length);

concat = concatArrays(arr, arr2);

console.log(concat);

-> Output:

6) Create a function to Create array, get the sum of all elements in an array, remove specific element from array.

-> function createArray(length)

{

let arr = new Array(length);

    const prompt = require('prompt-sync')();

    for (let i = 0; i < length; i++)

{

        val = parseInt(prompt("Enter the value: "));

        arr[i] = val;

    }

    return arr;

}

function sumArray(arr)

{

return arr.reduce((a, b) => a + b, 0);

}

function removeElement(arr, value)

{

return arr.filter(item => item !== value);

}

const prompt = require('prompt-sync')();

length = parseInt(prompt("Enter the length: "));

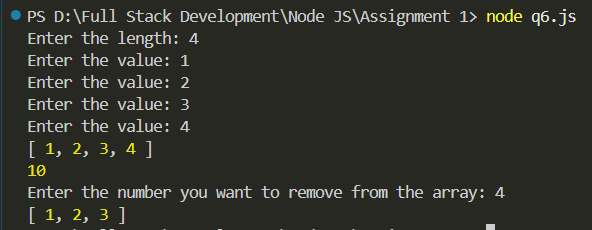
arr = createArray(length);

console.log(arr);

console.log(sumArray(arr));

num = parseInt(prompt("Enter the number you want to remove from the array: "));

console.log(removeElement(arr, num));

-> Output:

7) Create a function to Implement stack operations.

-> class Stack

{

constructor()

{

      this.items = [];

    }

    push(element)

{

      this.items.push(element);

    }

    pop()

{

      if (this.isEmpty())

{

        return "Stack is empty.";

      }

      return this.items.pop();

    }

    peek()

{

      if (this.isEmpty())

{

        return "Stack is empty.";

      }

      return this.items[this.items.length - 1];

    }

    isEmpty()

{

      return this.items.length === 0;

    }

    size()

{

      return this.items.length;

    }

    clear()

{

      this.items = [];

    }

  }

const myStack = new Stack();

myStack.push(1);

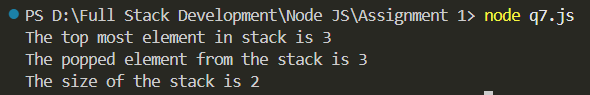
myStack.push(2);

myStack.push(3);

console.log(myStack.peek());

console.log(myStack.pop());

console.log(myStack.size());

-> Output:

8) Create a function to Print Multiplication table.

-> prompt = require('prompt-sync') ()

function printMultiplicationTable(num)

{

for (let i=1;i<=10;i++)

    {

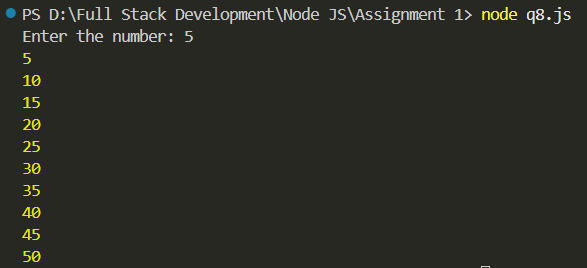
        console.log(i\*num);

    }

}

num = parseInt(prompt("Enter the number: "))

printMultiplicationTable(num)

-> Output:

9) Create a function to Convert decimal number to binary.

-> prompt = require('prompt-sync') ()

function decimalToBinary(decimal)

{

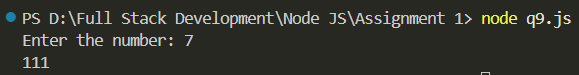
return (decimal >>> 0).toString(2);

}

num = parseInt(prompt("Enter the number: "))

binary = decimalToBinary(num);

console.log(binary);

-> Output:

10) Create a function which will display a message “welcome to node js” 10 times with a delay of 2 seconds between iterations (use setTimeout).

-> function welcomeMessage()

{

for (let i = 0; i < 10; i++)

{

      setTimeout(() =>

{

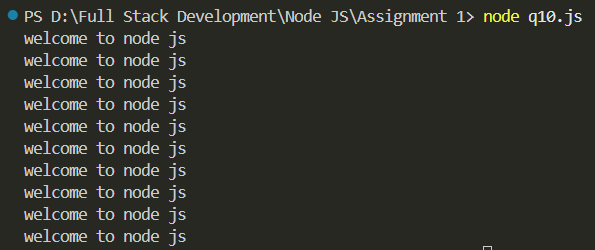
        console.log("welcome to node js");

      }, 2000 \* i);

    }

  }

welcomeMessage();

-> Output:

11) Create a Node JS Script file that displays hostname, platform, home directory, total memory, free memory details of current system on the console.

-> const os = require('os');

console.log(`Hostname: ${os.hostname()}`);

console.log(`Platform: ${os.platform()}`);

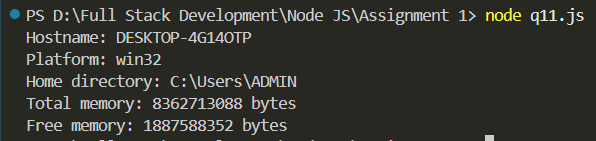
console.log(`Home directory: ${os.homedir()}`);

const totalMemory = os.totalmem();

const freeMemory = os.freemem();

console.log(`Total memory: ${totalMemory} bytes`);

console.log(`Free memory: ${freeMemory} bytes`);

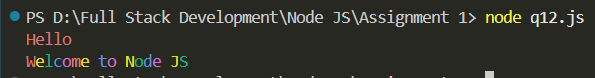
-> Output:

12) Create a Node JS script file that displays Hello text in red color and “Welcome to Node JS” text in rainbow colors on the console.

-> const colors = require('colors');

console.log('Hello'.red);

console.log('Welcome to Node JS'.rainbow);

-> Output:

13) Create a user defined module named Math with four functions Addition, Subtraction, Multiplication, Division and export them. Import Math module form other Node JS Script file and invoke all the four functions to perform operations on given input.

-> math.js

module.exports =

{

addition: (a, b) => a + b,

    subtraction: (a, b) => a - b,

    multiplication: (a, b) => a \* b,

    division: (a, b) => a / b

};

-> arithmetic\_operations.js

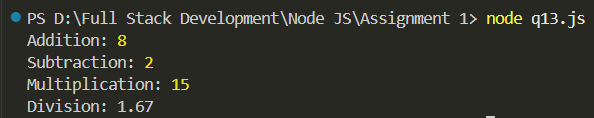
const math = require('./math');

console.log('Addition:', math.addition(5, 3));

console.log('Subtraction:', math.subtraction(5, 3));

console.log('Multiplication:', math.multiplication(5, 3));

console.log('Division:', (math.division(5, 3)).toFixed(2));

-> Output:

14) Write a node script file to display current date time by using user defined date module.

-> date.js

module.exports =

{

getCurrentDateTime: () =>

{

      const now = new Date();

      const options =

{

        year: 'numeric',

        month: 'long',

        day: 'numeric',

        hour: '2-digit',

        minute: '2-digit',

        second: '2-digit'

      };

      return now.toLocaleString('en-IN', options);

    }

};

-> get\_current\_date.js

const date = require('./date');

console.log('Current date and time:', date.getCurrentDateTime());

-> Output:

15) Write a Node script file to find out how many seconds are there in a year. How many seconds are there in a century and writes the result into a file.

-> const fs = require('fs');

const secondsInMinute = 60;

const secondsInHour = secondsInMinute \* 60;

const secondsInDay = secondsInHour \* 24;

const secondsInYear = secondsInDay \* 365;

const secondsInCentury = secondsInYear \* 100;

const result = `Number of seconds in a year: ${secondsInYear}\nNumber of seconds in a century: ${secondsInCentury}`;

fs.writeFile('result.txt', result, (err) =>

{

if (err)

{

      console.error('Error writing to file:', err);

    }

else

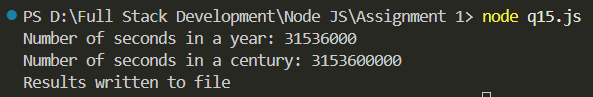
{

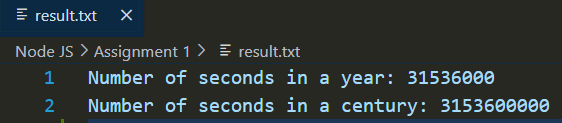
      console.log('Results written to file');

    }

  });

console.log(result);

-> Output:

-> result.txt

16) Write a program to create http module to transfer data over the HTTP protocol.

-> server.js

const http = require('http');

const server = http.createServer((req, res) =>

{

res.writeHead(200, {'Content-Type': 'text/plain'});

    res.end('Hello, world!\n');

});

server.listen(3000, () =>

{

console.log('Server running at http://localhost:3000/');

});

-> client.js

const http = require('http');

const options =

{

hostname: 'localhost',

    port: 3000,

    path: '/',

    method: 'GET'

};

const req = http.request(options, (res) =>

{

let data = '';

    res.on('data', (chunk) =>

{

        data += chunk;

    });

    res.on('end', () =>

{

        console.log(data);

    });

});

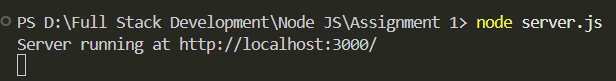
req.on('error', (error) =>

{

console.error(`Problem with request: ${error.message}`);

});

req.end();

-> Output:

