**Node**

**1)**

var ps=require("fs");

var data=ps.readFileSync("Hello.txt");

console.log(data)

console.log(data.toString());

console.log("Program ended");

**2)**

var ps=require("fs");

ps.readFile("Hello.txt", function(e,data)

{

if(e)

{

return console.error(e);

}

console.log(data.toString()); // if you want buffer data then remove to string

console.error("complet");

}

);

console.log("Program ended");

**3) Write a js code that display ”Hello” with increasing font size in interval of 50ms in blue colour and it should stop when font size reaches to 50px.**

<html>

<head>

<style>

p{

color:blue;

}

</style>

</head>

<body>

<p id="p1"> Hello</p>

<button onclick="fun2()">font-size</button>

<script>

font="2";

function fun(font)

{

document.getElementById("p1").style.fontSize=font;

}

function fun2()

{

setInterval(

function()

{

if(font<=50)

{

fun(font++);

}

},50

);

}

</script>

</body>

</html>

**4) File Module (CRUD)**

var ps=require("fs");

ps.mkdirSync("Hello");

ps.writeFileSync("Hello/user.txt","Hello");

ps.appendFileSync("Hello/user.txt","\nWorld");

var data=ps.readFileSync("Hello/user.txt","utf-8");

ps.renameSync("Hello/user.txt","Hello/user1.txt");

console.log(data.toString());

ps.unlinkSync("Hello/user1.txt");

ps.rmdirSync("Hello");

**5) File module with callback**

ps=require("fs");

ps.writeFile("a2.txt","Today is cold day123",

()=>

{

console.log("completed");

});

ps.readFile("a1.txt","utf-8",(err,data)=>

{

console.log(data); //use *data.toString()* if not using utf-8

});

**Output:**

done

completed

Today is cold day

**6) Write a Nodejs script to take 5 elements separated by white space in .txt file. Print sorted array of these 5 elements on Node Js server.**

var ps=require("fs");

ps.writeFileSync("task.txt","0 1 99 20 33 -44 50");

data=ps.readFileSync("task.txt","utf-8")

console.log(data);

data=data.split(" ");

console.log(data);

for(i=0;i<data.length;i++)

{

data[i]=parseInt(data[i]);

}

d1=data.sort();

console.log(d1);

**Output:**

0 1 99 20 33 -44 50

(7) ['0', '1', '99', '20', '33', '-44', '50']

(7) [-44, 0, 1, 20, 33, 50, 99]

**7) OS module:**

os=require("os");

console.log(os.arch());

console.log(os.hostname());

console.log(os.platform());

console.log(os.tmpdir());

console.log(os.freemem());

a1=os.freemem();

console.log(`${a1/1024/1024/1024}`);

**Output:**

x64

406-3-104

win32

C:\Users\LJENG\AppData\Local\Temp

3818008576

3.5555381774902344

**8) Path module**

var pm=require("path");

path1=pm.dirname("D:/ZKG/DEMO2/abc.txt");

console.log("Path: " + path1);

path2=pm.extname("D:/ZKG/DEMO2/abc.txt");

console.log("Extension: "+path2);

path2=pm.basename("D:/ZKG/DEMO2/abc.txt");

console.log("Basename: "+ path2);

path2=pm.parse("D:/ZKG/DEMO2/abc.txt");

console.log(path2);

console.log(path2.root);

console.log(path2.dir);

console.log(path2.base);

console.log(path2.ext);

console.log(path2.name);

**Output:**

Path: D:/ZKG/DEMO2

Extension: .txt

Basename: abc.txt

{root: 'D:/', dir: 'D:/ZKG/DEMO2', base: 'abc.txt', ext: '.txt', name: 'abc'}

D:/

D:/ZKG/DEMO2

abc.txt

.txt

abc

**9) HTTP module**

var h=require("http");

var server=h.createServer(

function(req,res)

{

res.write("FSD-2");

res.end(); //can be empty or include string

}

);

server.listen(5051);

console.log("Thanks for run");

**Task 1:**

var h=require("http");

var server=h.createServer(

function(req,res)

{

if(req.url=="/")

{

res.writeHead(200,{"content-type":"text/html"});

res.write("<b> Home page </b>");

res.end();

}

else if(req.url=="/student")

{

res.writeHead(200,{"content-type":"text/plain"});

res.write("<i> Home page1 </i>");

res.end();

}

else

{

res.writeHead(404,{"content-type":"text/html"});

res.write("<h1> Page Not found </h1>");

res.end("Thanks");

res.write("Bye");

}

}

);

server.listen(5001);

console.log("Thanks for run");

**Task2:**

var http=require("http");

var server=http.createServer(

function(req,res)

{

if(req.url=="/")

{

const a={"Name":"ABC", "Age":35};

res.writeHead(200,

{"content-type":"application/json"

});

res.write("Thank you..!");

res.write(JSON.stringify(a));

res.end();

}

});

server.listen(6008);

**10) URL module:**

var u=require("url");

var addr="http://localhost:8080/default.htm?year=2017&month=february";

var adr1="https://www.google.com/search?q=url+module+in+node+js&rlz=1C1YTUH\_enIN 1039IN1039&oq=url+modu&aqs=chrome.0.0i512j69i57j0i512l4j0i390l4.2814j0j7&sourceid=chrome&ie=UTF-8";

var q1=u.parse(addr,true);

console.log(q1);

console.log(q1.host);

console.log(q1.pathname);

console.log(q1.query);

console.log(q1.search);

var q=u.parse(adr1,true); //query will be given as JSON Object

console.log(q);

console.log(q.host);

console.log(q.pathname);

console.log(q.query);

console.log(q.search);

**Output:**

Url {protocol: 'http:', slashes: true, auth: null, host: 'localhost:8080', port: '8080', …}

localhost:8080

/default.htm

{year: '2017', month: 'february'}

?year=2017&month=february

Url {protocol: 'https:', slashes: true, auth: null, host: 'www.google.com', port: null, …}

www.google.com

/search

{q: 'url module in node js', rlz: '1C1YTUH\_enIN1039IN1039', oq: 'url modu', aqs: 'chrome.0.0i512j69i57j0i512l4j0i390l4.2814j0j7', sourceid: 'chrome', …}

?q=url+module+in+node+js&rlz=1C1YTUH\_enIN1039IN1039&oq=url+modu&aqs=chrome.0.0i512j69i57j0i512l4j0i390l4.2814j0j7&sourceid=chrome&ie=UTF-8

**11) Write a nodejs program load a simple html file on nodejs web server and print its content as html content.**

var h=require("http");

var ps=require("fs");

var u=require("url");

var addr="http://localhost:8080/16.html";

var q=u.parse(addr,true);

data=ps.readFileSync("."+q.pathname);

var server=h.createServer(

function(req,res)

{

res.writeHead(200,{"content-type":"text/html"});

*//res.writeHead(200,{"content-type":"text/plain"}); gives content of file(Whole program will display in port)*

res.write(data);

res.end();

});

server.listen(6051);

**12) Write a nodejs script to print query string of url on console as well as on file using ES6 callback.**

var u=require("url");

var ps=require("fs");

var adr1="https://www.google.com/search?q=url+module+in+node+js&rlz=1C1YTUH\_enIN1039IN1039&oq=url+modu&aqs=chrome.0.0i512j69i57j0i512l4j0i390l4.2814j0j7&sourceid=chrome&ie=UTF-8";

var q1=u.parse(adr1,true);

var qdata=q1.query;

console.log(qdata);

ps.writeFile("fsd2.txt",qdata.q,(err)=>

{

console.log("completed");

});

**13) Create own module:**

***Method 1***

**In 29.js file:**

const add=(a,b)=>

{

return(a+b);

}

module.exports=add;

**In another file:**

var d=require("./29.js");

console.log(d(10,15));

***Method 2***

**In 29.js file:**

const sub=(a,b)=>

{

return(a-b);

}

const mul=(a,b)=>

{

return(a\*b);

}

module.exports.s=sub;

module.exports.m=mul;

**In another file:**

var d1=require("./29.js");

console.log(d1.s(10,5));

console.log(d1.m(10,15));

***Method 3***

**In 29.js file:**

const sub=(a,b)=>

{

return(a-b);

}

const mul=(a,b)=>

{

return(a\*b);

}

module.exports.d2=sub;

module.exports.e2=mul;

**In another file:**

var {d2,e2}=require("./29.js");

console.log(d2(10,7));

console.log(e2(10,12));

***Method 4***

**In 29.js file:**

const sub=(a,b)=>

{

return(a-b);

}

const mul=(a,b)=>

{

return(a\*b);

}

const name="Hello"

module.exports={sub,mul,name};

**In another file:**

var {sub,mul,name}=require("./29.js");

console.log(sub(100,20));

console.log(mul(10,2));

console.log(name)

**14) Write a nodejs script to create my own module to calculate reverse of a given . That module should be used to compute all numbers between 1 to 100 in which square of reverse and reverse of square is same. These has call of reverse twice so call it from module.**

**In 31.js file**

const rev=(n)=>

{

var r=0;

r2=n\*n;

while(n>0)

{

r=(r\*10)+(n%10);

n=parseInt(n/10);

}

r3=r\*r;

return [r2,r3];

}

module.exports=rev;

**In another file:**

var rev =require("./31.js");

arr=rev(49);

console.log("Square of reverse is: " + arr[0]);

console.log("Reverse of square is: " + arr[1]);

fno=arr[0];

sno=arr[1];

if(fno==sno)

{

console.log("Equal");

}

else

{

console.log("not equal");

}

**Output:**

Square of reverse is: 2401

Reverse of square is: 8836

not equal