Date: 19-02-2025

* Class
* var json='{"name":"Arwin","age":30}'
* var json1={
* name:"Arwin",
* age:30
* }
* const jsontoobject=JSON.parse(json)
* console.log(jsontoobject)
* const objecttojson=JSON.stringify(json1,["name"],5)
* console.log(objecttojson)
* Read data
* file.json
* {
* "name":"Kashish",
* "age":25,
* "City":"Jalandhar",
* "Phone":[
* 12345,
* 234565
* ],
* "State":"Punjab",
* "Country":"India"
* }
* fs.js
* //Synchronus
* const fs=require('fs')
* console.log("Started file reading sync")
* data=fs.readFileSync('file.json')
* parsed=JSON.parse(data)
* console.log(parsed)
* console.log("Sync reading done")
* //Asynchronus
* console.log("Started file reading async")
* fs.readFile('file.json',(err,data)=>{
* if(err)
* throw err
* else
* parsed=JSON.parse(data)
* console.log(parsed)
* })
* console.log("Async reading done")
* Write data in files
* var datatowrite={
* name:"Kashish",
* age:30
* }
* var parsed1=JSON.stringify(datatowrite,null,2)
* fs.writeFileSync('file1.json',parsed1)
* fs.writeFile('file2.json',parsed1,(err)=>{
* if(err)
* throw err
* })
* Rename file
* //Rename file
* fs.renameSync('file1.json','filesync.json')
* fs.rename('file2.json','fileasync.json',(err)=>{
* if(err)
* throw err
* else
* console.log("File renamed")
* })
* Delete File
* //Delete file
* fs.unlinkSync('filesync.json')
* fs.unlink('fileasync.json',(err)=>{
* if(err)
* throw err
* else
* console.log('File deleted')
* })
* Streams to read data from file
* var fs=require('fs')
* var data=''
* var reader=fs.createReadStream('test.txt')
* reader.setEncoding('utf8')
* reader.on('data',(chunk)=>{
* data=chunk
* })
* reader.on('end',()=>{
* console.log(data)
* })
* reader.on('error',(err)=>{
* console.log(err)
* })
* Streams to write data into file
* //Writable Stream
* var fs=require("fs");
* var data='This is node.js class';
* var writer=fs.createWriteStream('test.txt');
* writer.write(data,'UTF8');
* writer.end();
* writer.on('finish',function(){
* console.log("Write completed");
* });
* writer.on('error',function(err){
* console.log(err);
* });
* Piping to write
* //Piping data from one file to other file
* var fs=require("fs");
* var reader=fs.createReadStream('test.txt');
* var writer=fs.createWriteStream('test1.txt');
* writer.on('pipe',()=>{
* console.log('Something is piping into the writer');
* });
* reader.pipe(writer);
* Unpiping
* //Unpiping data from one file to other file
* var fs=require("fs");
* var reader=fs.createReadStream('test.txt');
* var writer=fs.createWriteStream('test1.txt');
* writer.on('unpipe',()=>{
* console.log('Unpiping occurred');
* });
* reader.unpipe(writer);
* Zip file using zlib
* //Zip file using zlib
* var zlib=require('zlib');
* var fs=require('fs');
* var gzip=zlib.createGzip();
* var r=fs.createReadStream('test.txt');
* var w =fs.createWriteStream('test.txt.gz');
* r.pipe(gzip).pipe(w);
* var zlib=require('zlib');
* var fs=require('fs');
* var defl=zlib.createDeflate();
* var r=fs.createReadStream('test.txt');
* var w =fs.createWriteStream('test.txt.gz');
* r.pipe(defl).pipe(w);
* Server
* var http=require('http');
* var server=http.createServer(function(req,res)
* {
* if(req.url=='/')
* {
* res.write('Welcome to index page');
* res.end();
* }
* else if(req.url=="/add")
* {
* function add(a,b){
* return a+b
* }
* res.write(`The result is ${add(4,5)}`);
* res.end();
* }
* else if(req.url=="/fact"){
* function fact(num){
* if(num==0 || num==1){
* return 1
* }
* else{
* return num\*fact(num-1)
* }
* }
* }
* })