**Error handling in javascript**

**<script>**

**try{**

**alet("hi");**

**}**

**catch(take){console.log(take);**

**}**

**</script> //referenceError:alet is not defined at tin.html**

* **<script>**

**try{**

**alet("hi");**

**}**

**catch(take){**

**alert(take);**

**}**

**</script> //referenceError:alet is not defined at tin.html # but as pop-up window**

* **<script>**

**try{**

**alet("hi");**

**}**

**catch(take){**

**alert(take.name);**

**}**

**</script> //referenceError:as pop-up window**

**//catch has two properties like .name & .message**

* **<script>**

**try{**

**alet("hi");**

**}**

**catch(take){**

**alert(take.message);//pop-up window..alet is not defined..**

**}**

**</script>**

* **<script>**

**try{**

**alet("hi");**

**}**

**catch(take){**

**alert(take.name);**

**alert(take.message);**

**}**

**</script>/\*here take.name will show like referenceError in pop-window & take.message will show alet is not defined\*/**

**/\*two times pop-up window will display with seperating messges that shown when we console the the take\*/**

* **<script>**

**try{**

**alet("hi");**

**}**

**catch(take){**

**alert(take.name);**

**alert(take.message);**

**alert(take);**

**}**

**</script>/\*if use only alert(take) then both messages ar4e get concate and then dispaly in pop-windows\*/**

* **<script>**

**function NAME(){**

**alert("karthik yadav");**

**}**

**try{**

**NAME();**

**}**

**catch(take){**

**}**

**</script>/\*if a function has called in in the try block it should have to catch have catch block otherwise it won't display in the pop-window\*/**

* **<script>**

**function NAME(){**

**document.write("karthik yadav");**

**}**

**try{**

**NAME();**

**}**

**catch(take){**

**}**

**</script>/\*if a function has called in in the try block it should have to catch have catch block otherwise it won't display in the in the body**

**it does'nt matter you kept alert for catch or not\*/**

**/\*the function we are calling is just sending the information to catch block by the name in the perenthesis\*/**

* **<script>**

**function NAME(){**

**alert("karthik yadav");**

**}**

**NAME();**

**</script>/\*it is like normal function\*/**

* **<script>**

**try{**

**milk();**

**}**

**catch(tonned){**

**alert(tonned);**

**}**

**console.log("nivea body milk");**

**</script>/\*undefined milk function is catched and stored in the argument of catch and dispalyed as an alert\*/**

**/\*console is also displayed in the libve server\*/**

* **<script>**

**throw "bad thoughts";**

**try{**

**milk();**

**}**

**catch(tonned){**

**alert(tonned);**

**}**

**console.log("nivea body milk");**

**</script>/\*if anything thrown in the script...the next functions wont'excute\*/**

* **<script>**

**try{**

**milk();**

**}**

**catch(tonned){**

**alert(tonned);**

**throw "bad thoughts";**

**}**

**console.log("nivea body milk");**

**</script>/\*thrown is displyed after to the catch alert in the catch block so no problem to the previous function it can caught up and displyed in alert box but message in console won't appears because before that some message has thrown in the catch box\*/**

* **<script>**

**try{**

**milk();**

**}**

**catch(tonned){**

**alert(tonned);**

**throw "bad thoughts";**

**}**

**finally{**

**console.log("nivea body milk");**

**}**

**</script>/\*if we keep console in finally block it will show content in the live server\*/**

* **<script>**

**try{**

**throw "deep moisturising serum";**

**}**

**catch(tonned){**

**alert(tonned);**

**throw "bad thoughts";**

**}**

**finally{**

**console.log("nivea body milk");**

**}**

**</script>/\*if we thrown something in the try and it will caught up in the catch block perenthesis \*/**

* **<script>**

**</script>/\*return can't catch in the catch block**

**because it should return some function\*/**

* **<script>**

**function hype(){**

**try{**

**return "link";**

**}**

**catch(yes){**

**}**

**}**

**console.log(hype());**

**</script>/\*function hype()return some value like "link" it has to catch in some argument in the catch block and at the end of the function we have to call the function by consoling itg\*/**

* **<script>**

**throw "don't dispaly";**

**function hype(){**

**try{**

**return "link";**

**}**

**catch(yes){**

**}**

**finally{**

**console.log("karthik yadav");**

**}**

**}**

**console.log(hype());**

**</script>/\*if we throw something before the function excute entire function unexcte\*/**

* **<script>**

**function hype(){**

**try{**

**return "link";**

**}**

**catch(yes){**

**}**

**console.log("karthik yadav");**

**}**

**console.log(hype());**

**</script>/\* in side the function console msgs cant excutes so karthik yadav don't dispaly \*/**

* **<script>**

**function hype(){**

**try{**

**return "link";**

**}**

**catch(yes){**

**}**

**finally{**

**console.log("karthik yadav");**

**return "autherity";**

**}**

**}**

**console.log(hype());**

**</script>/\*return value in finally block will display in the calling functions\*/**

* **<script>**

**function hype(){**

**try{**

**return "link";**

**}**

**catch(yes){**

**}**

**finally{**

**return "autherity";**

**console.log("karthik yadav");**

**}**

**}**

**console.log(hype());**

**</script>/\*if we return a value in the finally block 1st..then console msgs don't dispaly finally block come out with return value\*/**

* **<script>**

**function hype(){**

**try{**

**return "link";**

**}**

**finally{**

**console.log("karthik yadav");**

**return "authority";**

**}**

**}**

**console.log(hype());**

**</script>/\*we can excute try without catch block\*/**

**NODE PACKAGE MANAGER**

**C:\Users\user>npm version //to the version**

**{**

**npm: '6.14.11',**

**ares: '1.16.1',**

**brotli: '1.0.9',**

**cldr: '37.0',**

**icu: '67.1',**

**llhttp: '2.1.3',**

**modules: '83',**

**napi: '7',**

**nghttp2: '1.41.0',**

**node: '14.15.5',**

**openssl: '1.1.1i',**

**tz: '2020a',**

**unicode: '13.0',**

**uv: '1.40.0',**

**v8: '8.4.371.19-node.18',**

**zlib: '1.2.11'**

**}**

**NPM INSTALL EXPRESS**

**C:\Users\user>npm install express**

**npm WARN saveError ENOENT: no such file or directory, open 'C:\Users\user\package.json'**

**npm WARN enoent ENOENT: no such file or directory, open 'C:\Users\user\package.json'**

**npm WARN user No description**

**npm WARN user No repository field.**

**npm WARN user No README data**

**npm WARN user No license field.**

**+ express@4.17.1**

**added 50 packages from 37 contributors and audited 51 packages in 6.833s**

**found 0 vulnerabilities**

**LOCALLY INSTALLED MODULES**

**C:\Users\user>npm ls**

**C:\Users\user**

**+-- express@4.17.1**

**| +-- accepts@1.3.7**

**| | +-- mime-types@2.1.29**

**| | | `-- mime-db@1.46.0**

**| | `-- negotiator@0.6.2**

**| +-- array-flatten@1.1.1**

**| +-- body-parser@1.19.0**

**| | +-- bytes@3.1.0**

**| | +-- content-type@1.0.4 deduped**

**| | +-- debug@2.6.9 deduped**

**| | +-- depd@1.1.2 deduped**

**| | +-- http-errors@1.7.2**

**| | | +-- depd@1.1.2 deduped**

**| | | +-- inherits@2.0.3**

**| | | +-- setprototypeof@1.1.1 deduped**

**| | | +-- statuses@1.5.0 deduped**

**| | | `-- toidentifier@1.0.0**

**| | +-- iconv-lite@0.4.24**

**| | | `-- safer-buffer@2.1.2**

**| | +-- on-finished@2.3.0 deduped**

**| | +-- qs@6.7.0 deduped**

**| | +-- raw-body@2.4.0**

**| | | +-- bytes@3.1.0 deduped**

**| | | +-- http-errors@1.7.2 deduped**

**| | | +-- iconv-lite@0.4.24 deduped**

**| | | `-- unpipe@1.0.0 deduped**

**| | `-- type-is@1.6.18 deduped**

**| +-- content-disposition@0.5.3**

**| | `-- safe-buffer@5.1.2 deduped**

**| +-- content-type@1.0.4**

**| +-- cookie@0.4.0**

**| +-- cookie-signature@1.0.6**

**| +-- debug@2.6.9**

**| | `-- ms@2.0.0**

**| +-- depd@1.1.2**

**| +-- encodeurl@1.0.2**

**| +-- escape-html@1.0.3**

**| +-- etag@1.8.1**

**| +-- finalhandler@1.1.2**

**| | +-- debug@2.6.9 deduped**

**| | +-- encodeurl@1.0.2 deduped**

**| | +-- escape-html@1.0.3 deduped**

**| | +-- on-finished@2.3.0 deduped**

**| | +-- parseurl@1.3.3 deduped**

**| | +-- statuses@1.5.0 deduped**

**| | `-- unpipe@1.0.0**

**| +-- fresh@0.5.2**

**| +-- merge-descriptors@1.0.1**

**| +-- methods@1.1.2**

**| +-- on-finished@2.3.0**

**| | `-- ee-first@1.1.1**

**| +-- parseurl@1.3.3**

**| +-- path-to-regexp@0.1.7**

**| +-- proxy-addr@2.0.6**

**| | +-- forwarded@0.1.2**

**| | `-- ipaddr.js@1.9.1**

**| +-- qs@6.7.0**

**| +-- range-parser@1.2.1**

**| +-- safe-buffer@5.1.2**

**| +-- send@0.17.1**

**| | +-- debug@2.6.9 deduped**

**| | +-- depd@1.1.2 deduped**

**| | +-- destroy@1.0.4**

**| | +-- encodeurl@1.0.2 deduped**

**| | +-- escape-html@1.0.3 deduped**

**| | +-- etag@1.8.1 deduped**

**| | +-- fresh@0.5.2 deduped**

**| | +-- http-errors@1.7.2 deduped**

**| | +-- mime@1.6.0**

**| | +-- ms@2.1.1 extraneous**

**| | +-- on-finished@2.3.0 deduped**

**| | +-- range-parser@1.2.1 deduped**

**| | `-- statuses@1.5.0 deduped**

**| +-- serve-static@1.14.1**

**| | +-- encodeurl@1.0.2 deduped**

**| | +-- escape-html@1.0.3 deduped**

**| | +-- parseurl@1.3.3 deduped**

**| | `-- send@0.17.1 deduped**

**| +-- setprototypeof@1.1.1**

**| +-- statuses@1.5.0**

**| +-- type-is@1.6.18**

**| | +-- media-typer@0.3.0**

**| | `-- mime-types@2.1.29 deduped**

**| +-- utils-merge@1.0.1**

**| `-- vary@1.1.2**

**`-- http@0.0.1-security**

**npm ERR! extraneous: ms@2.1.1 C:\Users\user\node\_modules\send\node\_modules\ms**

**GLOBALLY INSTALLED MODULES IN NODE JS PACKAGE MANAGER**

**C:\Users\user>npm install express\_g**

**npm ERR! code E404**

**npm ERR! 404 Not Found - GET https://registry.npmjs.org/express\_g - Not found**

**npm ERR! 404**

**npm ERR! 404 'express\_g@latest' is not in the npm registry.**

**npm ERR! 404 You should bug the author to publish it (or use the name yourself!)**

**npm ERR! 404**

**npm ERR! 404 Note that you can also install from a**

**npm ERR! 404 tarball, folder, http url, or git url.**

**npm ERR! A complete log of this run can be found in:**

**npm ERR! C:\Users\user\AppData\Roaming\npm-cache\\_logs\2021-03-24T06\_32\_42\_283Z-debug.log**

**UNINSTALL NPM EXPESS**

**C:\Users\user>npm unistall express**

**Usage: npm <command>**

**where <command> is one of:**

**access, adduser, audit, bin, bugs, c, cache, ci, cit,**

**clean-install, clean-install-test, completion, config,**

**create, ddp, dedupe, deprecate, dist-tag, docs, doctor,**

**edit, explore, fund, get, help, help-search, hook, i, init,**

**install, install-ci-test, install-test, it, link, list, ln,**

**login, logout, ls, org, outdated, owner, pack, ping, prefix,**

**profile, prune, publish, rb, rebuild, repo, restart, root,**

**run, run-script, s, se, search, set, shrinkwrap, star,**

**stars, start, stop, t, team, test, token, tst, un,**

**uninstall, unpublish, unstar, up, update, v, version, view,**

**whoami**

**npm <command> -h quick help on <command>**

**npm -l display full usage info**

**npm help <term> search for help on <term>**

**npm help npm involved overview**

**Specify configs in the ini-formatted file:**

**C:\Users\user\.npmrc**

**or on the command line via: npm <command> --key value**

**Config info can be viewed via: npm help config**

**npm@6.14.11 C:\Program Files\nodejs\node\_modules\npm**

**Did you mean one of these?**

**uninstall**

**install**

**unstar**

**C:\Users\user>npm ls**

**C:\Users\user**

**+-- express@4.17.1**

**| +-- accepts@1.3.7**

**| | +-- mime-types@2.1.29**

**| | | `-- mime-db@1.46.0**

**| | `-- negotiator@0.6.2**

**| +-- array-flatten@1.1.1**

**| +-- body-parser@1.19.0**

**| | +-- bytes@3.1.0**

**| | +-- content-type@1.0.4 deduped**

**| | +-- debug@2.6.9 deduped**

**| | +-- depd@1.1.2 deduped**

**| | +-- http-errors@1.7.2**

**| | | +-- depd@1.1.2 deduped**

**| | | +-- inherits@2.0.3**

**| | | +-- setprototypeof@1.1.1 deduped**

**| | | +-- statuses@1.5.0 deduped**

**| | | `-- toidentifier@1.0.0**

**| | +-- iconv-lite@0.4.24**

**| | | `-- safer-buffer@2.1.2**

**| | +-- on-finished@2.3.0 deduped**

**| | +-- qs@6.7.0 deduped**

**| | +-- raw-body@2.4.0**

**| | | +-- bytes@3.1.0 deduped**

**| | | +-- http-errors@1.7.2 deduped**

**| | | +-- iconv-lite@0.4.24 deduped**

**| | | `-- unpipe@1.0.0 deduped**

**| | `-- type-is@1.6.18 deduped**

**| +-- content-disposition@0.5.3**

**| | `-- safe-buffer@5.1.2 deduped**

**| +-- content-type@1.0.4**

**| +-- cookie@0.4.0**

**| +-- cookie-signature@1.0.6**

**| +-- debug@2.6.9**

**| | `-- ms@2.0.0**

**| +-- depd@1.1.2**

**| +-- encodeurl@1.0.2**

**| +-- escape-html@1.0.3**

**| +-- etag@1.8.1**

**| +-- finalhandler@1.1.2**

**| | +-- debug@2.6.9 deduped**

**| | +-- encodeurl@1.0.2 deduped**

**| | +-- escape-html@1.0.3 deduped**

**| | +-- on-finished@2.3.0 deduped**

**| | +-- parseurl@1.3.3 deduped**

**| | +-- statuses@1.5.0 deduped**

**| | `-- unpipe@1.0.0**

**| +-- fresh@0.5.2**

**| +-- merge-descriptors@1.0.1**

**| +-- methods@1.1.2**

**| +-- on-finished@2.3.0**

**| | `-- ee-first@1.1.1**

**| +-- parseurl@1.3.3**

**| +-- path-to-regexp@0.1.7**

**| +-- proxy-addr@2.0.6**

**| | +-- forwarded@0.1.2**

**| | `-- ipaddr.js@1.9.1**

**| +-- qs@6.7.0**

**| +-- range-parser@1.2.1**

**| +-- safe-buffer@5.1.2**

**| +-- send@0.17.1**

**| | +-- debug@2.6.9 deduped**

**| | +-- depd@1.1.2 deduped**

**| | +-- destroy@1.0.4**

**| | +-- encodeurl@1.0.2 deduped**

**| | +-- escape-html@1.0.3 deduped**

**| | +-- etag@1.8.1 deduped**

**| | +-- fresh@0.5.2 deduped**

**| | +-- http-errors@1.7.2 deduped**

**| | +-- mime@1.6.0**

**| | +-- ms@2.1.1 extraneous**

**| | +-- on-finished@2.3.0 deduped**

**| | +-- range-parser@1.2.1 deduped**

**| | `-- statuses@1.5.0 deduped**

**| +-- serve-static@1.14.1**

**| | +-- encodeurl@1.0.2 deduped**

**| | +-- escape-html@1.0.3 deduped**

**| | +-- parseurl@1.3.3 deduped**

**| | `-- send@0.17.1 deduped**

**| +-- setprototypeof@1.1.1**

**| +-- statuses@1.5.0**

**| +-- type-is@1.6.18**

**| | +-- media-typer@0.3.0**

**| | `-- mime-types@2.1.29 deduped**

**| +-- utils-merge@1.0.1**

**| `-- vary@1.1.2**

**`-- http@0.0.1-security**

**npm ERR! extraneous: ms@2.1.1 C:\Users\user\node\_modules\send\node\_modules\ms**

**JAVASCRIPT MODULES WITH ES6**

// turn.js is the file name.

export default class Uniorn{

    constructor(name,fam,klas){

        this.name = name

        this.fam = fam

        this.klas=klas

    }

}

export function printName(phyto){

    console.log(`plant is ${phyto.name}`)

}

export function printFam(phyto){

    console.log(`plant is ${phyto.fam} `)

}

export function printKlas(phyto){

    console.log(`this plant belongs to klas ${phyto.klas}`)

}

//red.js is the file name

import Unicorn,{printName,printFam,printKlas} from './turn.js'

const phyto = new Unicorn('zea mays','poaceae','monocotyledonae')

console.log(phyto)

printName(phyto)

printFam(phyto)

printKlas(phyto)

//green.html

<html>

    <title>js modules</title>

    <head>

        <script type="module" src="red.js"></script>

    </head>

</html>

**Making bridge between 3 javascript and importing functions**

//this is crust.js

export default{

        name:'karthik yadav'

    }

//this turn.js

export{

    default

}from"./crust.js"

//this is mantle.js

import DefaultEport from"./turn.js"

console.log(DefaultEport);

//core.html

<html>

    <title>modules in javascript</title>

    <head>

        <script type="module" src="mantle.js"></script>

    </head>

</html>

**We can also import named functions from by using default**

//this is crust.js

const tinge="lucious spalattering of colors at everywhere in holyday";

   let tycoon={

        name:"wealthy powerful person in business industry"

    }

    export default{

        name:'karthik yadav'

    }

    export{tinge,tycoon};

//this turn.js

export{

    default,

    tinge,

    tycoon

}from"./crust.js"

//this is mantle.js

import DefaultExport,{tinge,tycoon} from"./turn.js"

console.log(DefaultExport);

console.log(tinge);

console.log(tycoon);

//core.html

<html>

    <title>modules in javascript</title>

    <head>

        <script type="module" src="mantle.js"></script>

    </head>

</html>

**Default export changed as named export**

export{

    default as mobile,

    tinge,

    tycoon

}from"./crust.js"

//turn.js

const tinge="lucious spalattering of colors at everywhere in holyday";

   let tycoon={

        name:"wealthy powerful person in business industry"

    }

    export default{

        name:'karthik yadav'

    }

    export{tinge,tycoon};

//crust.js

//mantle.js import {mobile,tinge,tycoon} from"./turn.js"

console.log(mobile);

console.log(tinge);

console.log(tycoon);

<html>

    <title>modules in javascript</title>

    <head>

        <script type="module" src="mantle.js"></script>

    </head>

</html>

//core.html

**Navigating to html file with http server…**

const {createServer} =require('http');

const server=createServer((request,Response)=>{

    Response.writeHead(200,{'content-Type':'kinley.html'});

    Response.write("<h1>hai world<h1>");

    return Response.end();

});

server.listen(3000);

/\*it was taken from karthik yadav/call.js

So node call.js we have to type in new terminal by opening the npm server http\*/

//open chrome server and type localhost:3000/kinley.html

//it will display message in response.write “hai world”

//inspect on the page by opening live server..open network and then click alt+r

//it will disply kinley.html favicon details..

const {createServer} =require("http");

const server=createServer((request,response)=>{

    response.writeHead(200,{'content-Type':'green.html'});

    response.write("restricted world");

    return response.end();

});

server.listen(8080);

/\*it was taken from karthik yadav/call.js

So node call.js we have to type in new terminal by opening the npm server http\*/

//open chrome server and type localhost:8080/green.html

//it will display message in response.write “restricted world”

//inspect on the page by opening live server..open network and then click alt+r

//it will disply kinley.html favicon details..

Application/json……………………………………………………………………..

const {createServer} =require("http");

const server=createServer((request,response)=>{

    response.writeHead(200,{'content-Type':'application/json'});

    response.write(JSON.stringify({a:1, b:2, c:3}));

    return response.end();

});

server.listen(8080);

READING A FILE WITH NODE JS

//reading file with node.js.................................................

/\*var fs = require('fs')

fs.readFile("good.txt",function(error, data){

    if(error){

        console.log(error)

    }

    else{

        console.log("async from of data:"+data)

    }

})

console.log("welcome to file system world");\*/

/\*var fs = require('fs')

fs.readFile("exotic.txt",function(error,data){

    if(error){

        console.log(error)

    }

    else{

        console.log("this is story told by gaur gopal das:"+data)

    }

})

console.log("inspiration is the vital ingredient to achieve the unsourrmountable goals");\*/

/\*const fs = require('fs')  //with arrow function

fs.readFile("peace.txt",(error,data)=>{

    if(error){

        console.log(error)

    }

    else{

        console.log("most valueble five lettered word-PEACE:"+data)

    }

})

console.log("karthik yadav MSc botany andhra university");\*/

//I have done this in call.js for junk.html

var fs = require('fs')

fs.readFile("junk.html",(error,data)=>{

    if(error){

        console.log("this is an error message")

    }else{

        console.log("karthik yadav:"+data)

    }

})

console.log("gajalaxmi paramasiavam");

SYNCHRONOUS MODE OF DATA RETRIVING

//retriving the data in synchronous mode........................................................................

var fs = require('fs')

var data = fs.readFileSync("life.txt")

console.log("sync mode of data:"+data)

console.log("a litte change in life shows big differrence");

//in synchronous way waiting of statements for data…..

var fs = require('fs')

var genda = fs.readFileSync("flag.txt")

console.log("synchronous mode of data retriving:"+genda)

console.log("feeling proud to be in a country with large diversified in sense of languages");

OPENING A FILE AND CLOSING

//opening the file.................................................................................................

/\*const lifefile = "life.txt";

const flagfile = "flag.txt";

const fs = require("fs");

fs.open("flag.txt",(error,fd)=>{

    if(error){

        console.log("this is an error message");

    }else{

        console.log("opened succesfully");

        fs.close(fd,(error)=>{

            console.log("file has to close after opening");

        })

    }

})\*/

ERROR THROWING IN OPENING FILE..

//an error throwing.........................................................................................

/\*const lifefile = "life.txt";

const flagfile = "flag.txt";

const fs = require("fs");

fs.open("fla.txt",(error,fd)=>{

    if(error){

        console.log("this is an error message");

    }else{

        console.log("opened succesfully");

        fs.close(fd,(error)=>{

            console.log("file has to close after opening");

        })

    }

})\*/

NODE.JS SYNCHRONOUS MODE FUNCTION EXCUTION

//node.js synchronous mode function excution......................................................................

function myName(){

    console.log("karthik yadav");

}

function qualification(){

    console.log("MSc botany in andhra university");

}

myName();

qualification();

NODE.JS ASYNCHRONOUS MODE OF FUNCTION EXCUTION BY DEFAULT

//By default node.js asynchronous type of function excution.................................................................

function myName(){

    setTimeout(()=>{

        console.log("karthik yadav");

    },10000)

}

function qualification(){

    console.log("MSc botany in andhra university");

}

myName();

qualification();

SYNCHRONOUS CAN BE USED THROUGH TRY CATCH BLOCK OR THROUGH PROMISES

//synchronous can be used through try catch block method.....................................................................

function myName(){

    setTimeout(()=>{

        console.log("karthik yadav");

    },10000)

}

function qualification(){

    console.log("MSc botany in andhra university");

}

    try{

        console.log("karthik yadav")

    }

    catch(error){

        console.log("if an error while getting myName."+error)

    }

    finally{

        qualification();

    }

SYNCHRONOUS MODE WITH PROMISE

//synchronous can be used through promises.............................................................

    function myName(){

        setTimeout(()=>{

            console.log("karthik yadav");

        },10000)

    }

    function qualification(){

        console.log("MSc botany in andhra university");

    }

    var me=new Promise((res,rej)=>{

        console.log("karthik yadav")

        res("myName has recived")

    }).then((data)=>{

        console.log(data)

        qualification()

    }).catch((error)=>{

        console.log(error)

    })

--------------------------------------------------------------------------

function myName(){

        setTimeout(()=>{

            console.log("karthik yadav");

        },10000)

    }

    function qualification(){

        console.log("MSc botany in andhra university");

    }

    var me=new Promise((res,rej)=>{

        console.log("karthik yadav")

        res("myName has recived")

    }).then((data)=>{

        //console.log(data)

        qualification()

    }).catch((error)=>{

        console.log(error)

    })

//here we can display only myName,qualification function only..resolved stri8ng can’t be displayed..here we no need to write string in resolve

-----------------------------------------------------------------------------

function myName(){

        setTimeout(()=>{

            console.log("karthik yadav");

        },10000)

    }

    function qualification(){

        console.log("MSc botany in andhra university");

    }

    var me=new Promise((res,rej)=>{

        console.log("karthik yadav")

        res() //no need of writing any string information 8in the res()

    }).then((data)=>{

        //console.log(data)

        qualification()

    }).catch((error)=>{

        console.log(error)

    })

EXPORTING&REQUIRE

function hello(){

    console.log("once up on a time mountains has wings,they fly around the world and making problems to people by landing on them.then lord indhra who the god of fornament cut off the wings and wings turned in to clouds but still they are hanging around the mountains")

}

exports.karthikYadav = hello;

//this from pen.js

let x = ('./pen')

    console.log(x.karthikYadav)

//this from call.js..but result is undefined because we have to use require

function hello(){

    console.log("once up on a time mountains has wings,they fly around the world and making problems to people by landing on them.then lord indhra who the god of fornament cut off the wings and wings turned in to clouds but still they are hanging around the mountains")

}

exports.karthikYadav = hello;/\*function hello is storing in to the key word like karthikyadav..but while consoling also we still call like

keyword hence it only display like  hello function assigned to karthikyadav keyword..

let x = require('./pen.js')

    console.log(x.karthikYadav)

    //result is function:hello..

……………………………………………………………………………………………………………….

function hello(){

    console.log("once up on a time mountains has wings,they fly around the world and making problems to people by landing on them.then lord indhra who the god of fornament cut off the wings and wings turned in to clouds but still they are hanging around the mountains")

}

exports.karthikYadav = hello;

//this from pen.js

let x = require('./pen.js')

    console.log(x.karthikYadav())

    //we have to function assigned keyword like a function() karthikyadav()

then it will display information in the hello()

……………………………………………………………………………………………………………………………………………………….

//here both functions and information in function both will print

let x = require('./pen.js')

    console.log(x.karthikyadav)

    x.karthikyadav();

//this is from call.js

function hello(){

    console.log("be patient & do your work and keep on effort,evrything will come in right way and on right time")

}

exports.karthikyadav = hello;

//this is from pen.js

………………………………………………………………………………………………………………………………………………………….

function hello(){

    console.log("be patient & do your work and keep on effort,evrything will come in right way and on right time")

}

exports.karthikyadav = hello;

exports.name = "karthik yadav"

this.user = welcome =>{

    console.log("user of lenovo core i3 processor is."+this.name)//if we write string in arrow function not consoling that argument what we passed in the arrow function we will get the string along with concatination..

}

this.user = welcome=>{

    console.log(welcome+this.name)    //in arrow function it'self we are passing one argument like welcome,when we console that argument along concatination we will get undefined in place of consoling argument..

}

this.user = welcome =>{

    console.log(`user of lenovo core i3 processor is: ${welcome}`)//if we write string in arrow function not consoling that argument what we passed in the arrow function we will get the string along with concatination..

}

//this is from pen.js

let x = require('./pen.js')

    console.log(x.karthikyadav)

    x.karthikyadav();

    console.log(x.name) //exporting property

    x.user("it's me karthik yadav")

//this is from call.js

……………………………………………………………………………………………………………………………………………….

EXPORTING ENTIRE MODULE NOT INDIVIDUALLY

function hello(){

    console.log("singles will create wonders,couples will see wonders,married will wonder what will happen")

}

module.exports={

    name:"karthik yadav",

    barla:hello,

    user:welcome=>{

        console.log(`user of lenovo core i3 processor is: ${welcome}`)

    }

}

--------------------------------------------------------------------------

STEAMSINGS IN NODE.JS

readStream In node.js

var fs = require('fs')

var readStream = fs.createReadStream('life.txt')

var start ='';

readStream.setEncoding('utf8')

readStream.on('data',function(chunk){

    start +=chunk

})

readStream.on('end',function(){

    console.log(start)

})

-----------------------------------------------------------------

var fs = require('fs')

var readStream = fs.createReadStream('good.txt')

var gaja ='';

readStream.setEncoding('utf8')

readStream.on('data',function(laxmi){

    gaja +=laxmi

})

readStream.on('end',function(){

    console.log(gaja)

})

//readStream.on event must has provided with data and end..

-----------------------------------------------------------------------

var fs = require('fs');

    var readStream = fs.createReadStream('kinley.html')

    var orange ='';

    readStream.setEncoding('utf8')

    readStream.on('data',function(ramcharan){

        orange +=ramcharan

    })

    readStream.on('end',function(){

        console.log(orange)

    })

writeStream

    var fs = require('fs')

    var writeData = "karthik yadav MSc botany andhra university in vishakapttanam,during the year 2016-2018"

    var writeStream = fs.createWriteStream('myProfile.txt')

    writeStream.write(writeData,'utf8')

    writeStream.end()

    writeStream.on('finish',function(){

        console.log("this data has been printed")

    })

---------------------------------------------------------------------------

var fs = require('fs')

var write =`a man who is much reserved by his nature..very much hesitate ask his to his boss to rise up his salary.

one fine day the man got up hope and courage,tell his wife darling today ill definately do this..the man went to his office

and the whole day the man was anxious and feeling scary..

the rightt moment come on the day he went yto the boss asked his to rise his salry to his much delight the boss agreed

and man in the top of the world and no border to hos zeal..he decided to share this happy moment with his wife..

he came to his home with much enthusiasum..

he saw fine delightful dinner table right at his hall.

on the table fine crockery set,dellight,delicious festive meal,light candles lit up on the table..

he thought her wife must have no about this before saying.her wife said congratulation his husband.

she is serving all items on the table and atlast she went to kitchen to dates.from the apron of her a piece of paper has fallen..

on the paper it was written..don't worry darling if you don't get rise in the salary you will deserve much more than that.

total supoort,total love,her love ion him unconditional,be as one who really deserve it`

var writeStream = fs.createWriteStream('love.txt')

writeStream.write(write,'utf8')

writeStream.end()

writeStream.on('finish',function(){

    console.log("this story has told by the gour gopal das")

})

-----------------------------------------------------------------------------

var fs = require('fs')

var money = "money makes many things and most prominent 5 letter used"

var writeStream = fs.createWriteStream('quote.txt')

writeStream.write(money,'utf8')

writeStream.end()

writeStream.on('finish',function(){

    console.log("karthik yadav")

})

// money writing into quote.txt

//this is quote.txt

makes many things and most prominent 5 letter used

------------------------------------------------------------------------------

var fs = require ('fs')

var writing ="never loose your enthusiasum with zeal you can achieve unsourrmountable goals"

var writeStream = fs.createWriteStream('hope.txt')

writeStream.write(writing,'utf8')

writeStream.end()

writeStream.on('finish',function(){

    console.log("this was given by swami mukundhanandha")

})

//in formation writing in hope.txt

//this hope.txt

never loose your enthusiasum with zeal you can achieve unsourrmountable goals

----------------------------------------------------------------------------

EVENTS IN NODE.JS

var events = require('events');

    var eventEmitter = new events.EventEmitter();

    var button = function(){

        console.log("karthik yadav clicking the door bell")

    }

    eventEmitter.on('DoorBell',button)

    eventEmitter.emit('DoorBell')

 var events = require('events')

    var eventEmitter = new events.EventEmitter();

    var hungry = function(){

        console.log("when you feel like hungry and you are sleeping on the right moment you have some ice in fridge then what will you open 1st")

    }

    eventEmitter.on('eyesLashes',hungry)

    eventEmitter.emit('eyesLashes')

var events = require('events')

    var eventEmitter = new events.EventEmitter();

    var Switch = function(){

        console.log("fan has revolve")

    }

    eventEmitter.on('power',Switch)

    eventEmitter.emit('power')

var events = require('events')

    var eventEmitter = new events.EventEmitter()

    var power = function(){

        console.log("if we click the power button on remote it'll turn on the televison")

    }

    var selfButton = function(){

        console.log("after unlock your ingnition if press the button motor engeine will start")

    }

    eventEmitter.on('remote',power)

    eventEmitter.emit('remote')

    eventEmitter.on('engine',selfButton)

    eventEmitter.emit('engine')

we can pass arguments in function parenthesis

var events = require('events')

    var eventEmitter = new events.EventEmitter()

    var power = function(){

        console.log("if we click the power button on remote it'll turn on the televison")

    }

    var selfButton = function(byk){

        console.log("after unlock your ingnition if press the button motor engeine will start"+byk)

    } //we can pass arguments also in anonymous function perenthesis

    eventEmitter.on('remote',power)

    eventEmitter.emit('remote')

    eventEmitter.on('engine',selfButton)

    eventEmitter.emit('engine',"vikranth 15 a ghazi ship metal made vehicle")

----------------------------------------------------------------------------

GLOBAL OBJECTS FACTORY

//global.js

var studentNumber = "sunandhini";

module.exports.stdNo = studentNumber

//accounts.js

var acc = require('./global.js')

var accCount = function(){

    acc.stdNo = "sai chaitanya"; //if we comment this stdNo get sunandhini

    console.log("Accounts:"+acc.stdNo)

}

module.exports.stdAcc = accCount

//admin.js

var adm = require('./global.js')

var admCount = function(){

    adm.stdNo ="gajalaxmi paramasivam";

    console.log("Administration:"+adm.stdNo)

}

module.exports.stdAdm = admCount

//index.js

var acc = require('./accounts.js')

var adm = require('./admin.js')

acc.stdAcc()

adm.stdAdm()

-----------------------------------------------------------------------------

//global.js

var myEducation = "life sciences";

module.exports.studies = myEducation

//Accounts.js

var inter = require('./global.js')

var BiPC = function(){

    inter.studies = "biology,physics & chemistry";

    console.log("intermediate education:"+inter.studies)

}

module.exports.bio = BiPC

//admin.js

var degree = require('./global.js')

 var BSc = function(){

    degree.studies = "botany,biotechnolgy,chemistry";

    console.log("bachelore of sciences:"+degree.studies)

 }

 module.exports.bioTech = BSc

//index.js

var inter = require('./accounts.js')

var degree = require('./admin')

inter.bio()

degree.bioTech()

FIXING THE PROBLEM GLOBAL OBJECT IN SHARED MODULE

//if any modules if we are changing the values for any purpose it could be local to that particular to that module it’ll never display as the global value for shared module for all as the final value..

//Global.js

module.exports = function(){

    return{

        studies:""

    }

}

//Accounts.js

var inter = require('./global.js')

var bio = inter.studies;

var BiPC = function(){

    bio = "biology,physics & chemistry";

    console.log("intermediate education:"+bio)

}

module.exports.bio = BiPC

//admin.js

var degree = require('./global.js')

 var bioTech = degree.studies;

 var BSc = function(){

    bioTech = "botany,biotechnolgy,chemistry";

    console.log("bachelore of sciences:"+bioTech)

 }

 module.exports.bioTech = BSc

//index.js

var inter = require('./accounts.js')

var degree = require('./admin')

inter.bio()

degree.bioTech()

inter.bio()

//if we comment out any one of the module it will display only finally declared values only

//kannada.js

var sandleWood = "kolor gold fields";

module.exports.movie = sandleWood

//tamil.js

var kollyWood = require('./kannada.js')

var tamiz = function(){

    //kollyWood.movie = "robo";

    console.log("tamil nadu:"+kollyWood.movie)

}

module.exports.padam = tamiz

//telugu.js

var tollyWood = require('./kannada.js')

var telungu = function(){

    tollyWood.movie = "bahuabli";

    console.log("Andhra pradhesh:"+tollyWood.movie)

}

module.exports.chithram = telungu

//film.js

var kollyWood = require('./tamil.js')

var tollyWood = require('./telugu.js')

kollyWood.padam()

tollyWood.chithram()

kollyWood.padam()

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

FILE SYSTEM MODULE THAT ALLOWS TO CREATE DIRECTORIES

//mk\_rmdir.js

const fs = require("fs");

const dirnames = {

    sync:"createdDir",

    async:"asyncCreatedDir",

};

function printMy(){

console.log("karthik yadav loves gajalaxmi");

}

fs.mkdirSync(dirnames.sync, 0o777);

fs.writeFileSync(`${dirnames.sync}/tile.js`,`(${printMy.toString()}())`)

//through dirnames fuction we can create createdDir folder.

//tile.js is a file

(function printMy(){

console.log("karthik yadav loves gajalaxmi");

}()) //get information in the printMy() function by node createdDir/tile.js

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

//mk\_gkdir.js

const fs = require ("fs")

const dirnames = {

      sync:"discoveDir",

      async:"asyncCreatedDir",

};

function epicalyx(){

    console.log("in some malvaceae members calyx is persistant even the after ripennig of friut some examples are:capsicum frutiscence,solanum melangina,lycoperscon esculentum");

}

fs.mkdirSync(dirnames.sync, 0o777);

fs.writeFileSync(`${dirnames.sync}/malva.js`,`(${epicalyx.toString()}())`)

//malva.js

(function epicalyx(){

    console.log("in some malvaceae members calyx is persistant even the after ripennig of friut some examples are:capsicum frutiscence,solanum melangina,lycoperscon esculentum");

}())

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

//mk\_gkdir.js

const fs = require('fs')

const dirnames = {

    sync:"developedDir",

    async:"asyncDevelopedDir",

};

function goodGem(){

    console.log("dont't torture,torment,burden yourself because of past action");

}

fs.mkdirSync(dirnames.sync,0o777);

fs.writeFileSync(`${dirnames.sync}/file.js`,`(${goodGem.toString()}())`)

//file.js

(function goodGem(){

    console.log("dont't torture,torment,burden yourself because of past action");

}())

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

CREATING DIRECTORY WITH FILE SYSTEM MODULE IN ASYNCHRONOUS WAY

//mk\_gkdir.js

const fs = require('fs');

const dirnames = {

    sync:"dicoverinedDir",

    async:"asyncDiscoveredDir",

};

function exotic(){

    console.log("anything that not native");

}

fs.mkdir(dirnames.async,(error)=>{

    if(error){

        console.log(error.message);

    }else{

        console.log(`${dirnames.async} discoverd!!!`)

        fs.writeFileSync(`${dirnames.async}/novo.js`,`(${exotic.toString()}())`)

    }

})

// novo.js

(function exotic(){

    console.log("anything that not native");

}())

//we can access information present in the novo.js by node asyncDiscoveredDir/novo.js

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

//mk\_gkdir.js

const fs = require('fs')

const dirnames = {

    sync:"generatedDir",

    async:"asyncGeneratedDir",

};

function diet(){

    console.log("you are what you eat..");

}

fs.mkdir(dirnames.async,(error)=>{

    if(error){

        console.log(error.message);

    }else{

        console.log(`${dirnames.async} discoverd!!!`)

        fs.writeFileSync(`${dirnames.async}/intake.js`,`(${diet.toString()}())`)

    }

})

//intake.js

(function diet(){

    console.log("you are what you eat..");

}())

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

//mk\_gkdir.js

const fs = require('fs')

const dirnames = {

    syn:"fabricatedDir",

    async:"asyncFabricatedDir",

};

function ink(){

    console.log("don't believe the things that go against common sense unless you dont have 1st hand proof");

}

fs.mkdir(dirnames.async,(error)=>{

    if(error){

        console.log(error.message);

    }else{

        console.log(`${dirnames.async} discovered!!`);

        fs.writeFileSync(`${dirnames.async}/coin.js`,`(${ink.toString()}())`)

    }

})

//Coin.js

(function ink(){

    console.log("don't believe the things that go against common sense unless you dont have 1st hand proof");

}())

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

App.js from react…

<div className="App">

      <header className="App-header">

        <img src={logo} className="App-logo" alt="logo" />

        <p>

          Hai karthik yadav welcome to react world

        </p>

        <a

          className="App-link"

          href="https://reactjs.org"

          target="\_blank"

          rel="noopener noreferrer"

        >

          Learn React

        </a>

      </header>