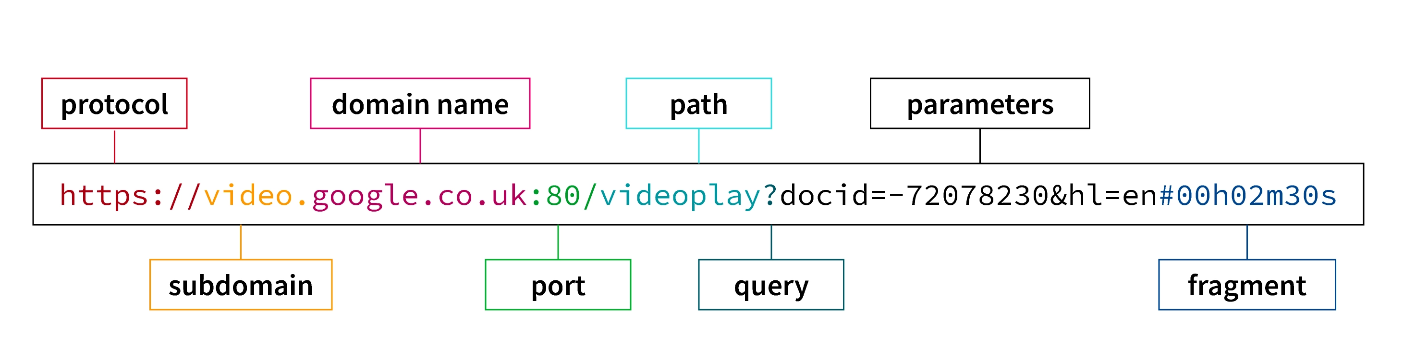
**Node JS-2**

**URL module**

The URL module contains functions that help in parsing a URL. In other words, we can split the different parts of a URL easily with the help of utilities provided by the URL module.



The syntax for including the url module in your application:

var url=require("url");

Parse an address with the **url.parse() method**, and it will return a URL object with each part of the address as properties.

**url.parse()** – This method takes the url string as a parameter and parses it. The url module returns an object with each part of the url as property of the object.

**Syntax of url.parse() :**

url.parse(url\_string, parse\_query\_string, slashes\_host)

Description of the parameters :

* **url\_string :** <string> It is the URL string.
* **parse\_query\_string :** <boolean> It is a boolean value. By default, its value is false. If it is set to true, then the query string is also parsed into an object. Otherwise, the query string is returned as an unparsed string.
* **slashes\_host :** <boolean> It is a boolean value. By default, its value is false. If it is set to true, then the token in between // and first / is considered host.

**Example**

* **LocalHost as URL**

var u=require("url");

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

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

console.log(q1);

console.log(q1.host);

console.log(q1.pathname);

console.log(q1.query);

**Output:**

Url {

protocol: 'http:',

slashes: true,

auth: null,

host: 'localhost:8080',

port: '8080',

hash: null,

search: '?year=2023&month=may',

query: [Object: null prototype] { year: '2023', month: 'may' },

pathname: '/default.htm',

path: '/default.htm?year=2023&month=may',

href: 'http://localhost:8080/default.htm?year=2023&month=may'

}

localhost:8080

/default.htm

[Object: null prototype] { year: '2023', month: 'may' }

* **Google Search Link as URL**

var u=require("url");

var adr1="https://www.google.com/search?q=good+morning";

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

console.log(q);

**Output:**

Url {

protocol: 'https:',

slashes: true,

auth: null,

host: 'www.google.com',

port: null,

hostname: 'www.google.com',

hash: null,

search: '?q=good+morning',

query: [Object: null prototype] { q: 'good morning' },

pathname: '/search',

path: '/search?q=good+morning',

href: 'https://www.google.com/search?q=good+morning'

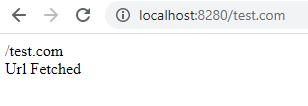
}

## **Read the URL**

The function passed into the http.createServer() has a req argument that represents the request from the client, as an object (http.IncomingMessage object).

This object has a property called "url" which holds the part of the url that comes after the domain name:

var http = require('http');  
http.createServer(function (**req**, res) {  
  res.writeHead(200, {'Content-Type': 'text/html'});  
  res.write(**req.url**);  
  res.end();  
}).listen(8280);



**Query string**

## **Get the details from query string**

We can fetch the values from url query string as mentioned below using URL module.

1. Add static url in code and request server to display data of query string

var http = require('http');

var url = require('url');

var addr="http://localhost:8080/default.html?year=2024&month=feb";

http.createServer(function (req, res) {

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

/\*Use the url module to get the querystring\*/

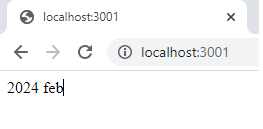
var q = url.parse(addr, true).query;

/\*Return the year and month from the query object:\*/

var txt = q.year + " " + q.month;

res.end(txt);

}).listen(3001);

****

1. Make changes in url at browser and request url to display data.

var http = require('http');

var url = require('url');

http.createServer(function (req, res) {

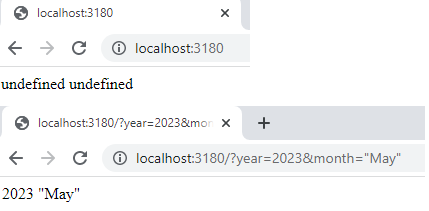
  res.writeHead(200, {'Content-Type': 'text/html'});

  var q = url.parse(req.url, true).query;

  var txt = q.year + " " + q.month;

  res.end(txt);

}).listen(3180);



**Task: Find a leap year from static url**

var u=require("url");

var addr="http://localhost:8080/default.html?year=2025&month=feb";

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

console.log(q);

// console.log(q.host);

// console.log(q.pathname);

// console.log(q.search);

var qdata=q.query;

console.log(qdata.year);

if(qdata.year%4==0)

{

   console.log(“Its a leap year")

}

else{

   console.log("Its not a leap year")

}

=============================================================================

**Output:**

**Url {**

**protocol: 'http:',**

**slashes: true,**

**auth: null,**

**host: 'localhost:8080',**

**port: '8080',**

**hostname: 'localhost',**

**hash: null,**

**search: '?year=2025&month=feb',**

**query: [Object: null prototype] { year: '2025', month: 'feb' },**

**pathname: '/default.html',**

**path: '/default.html?year=2025&month=feb',**

**href: 'http://localhost:8080/default.html?year=2025&month=feb'**

**}**

**2025**

**Its not a leap year**

**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=" http://localhost:8080/default.html?year=2025&month=feb";

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

var qdata=q1.query;

console.log(qdata);

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

{

console.log("completed");

});

**Task: Write a nodejs program which fetch filename from requested url and print that file’s data on http web server.**

var h=require("http");

var ps=require("fs");

var u=require("url");

var server=h.createServer(

function(req,res)

{

var q=u.parse(req.url,true);

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

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

res.write(data);

res.end();

});

server.listen(6052);

**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**"; (html file name)

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);

**How to create, export and use our own modules**

* **Own Module**

The node.js modules are a kind of package that contains certain functions or methods to be used by those who imports them. Some modules are present on the web to be used by developers such as fs, path,http,url etc. You can also make a package of your own and use it in your code.

**Example -**

Create two file with name – calc.js and index.js and copy the below code snippet. The calc.js is the custom node module which will hold the node functions. The index.js will import calc.js and use it in the node process.

***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)

* **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