URL module:

The URL module provides utilities for URL resolution and parsing. It can be accessed using:

```
var url = require('url');
```

Url module is one of the core modules that comes with node.js, which is used to parse the URL and its other properties.

By using URL module, it provides us with so many properties to work with.

These all are listed below:

Property	Description
.href	Provides us the complete url string
.host	Gives us host name and port number
.hostname	Hostname in the url
.path	Gives us path name of the url
.pathname	Provides host name , port and pathname
.port	Gives us port number specified in url
.auth	Authorization part of url
.protocol	Protocol used for the request
.search	Returns query string attached with url

As you can see in the above screen, there are various properties used for URL module.

Below is the snippet to check the URL properties with URL : localhost:4200.

MyApp.js

```
1. var http = require('http');
2. var url = require('url');
3.
```

```
4. http.createServer(function (req, res) {
5. 
6.     var queryString = url.parse(req.url, true);
7.     console.log(queryString);
8. 
9. }).listen(4200);
```

Now run the above snippet using node MyApp.js, and you can see the console like this:

```
Url {
  protocol: null,
  slashes: null,
  auth: null,
  host: null,
  port: null,
  hostname: null,
  hash: null,
  search: '',
  query: {},
  pathname: '/',
  path: '/',
  href: '/' }
```

Because we do not have a path attached to URL, now I'm writing text within my URL like below:

1. http://localhost:4200/bhavin

And again I compiled the snippet and got output like this:

```
H:\BACKUP\5thceit>node url.js
Url {
  protocol: null,
  slashes: null,
  auth: null,
  host: null,
  port: null,
  hostname: null,
  hash: null,
  search: null,
  query: [Object: null prototype] {},
  pathname: '/bhavin',
  path: '/bhavin',
  href: '/bhavin'
}
```

So now, I got my complete pathname along with href as well as the path name and other properties with null values.

href

Href property returns the complete URL along with all search terms and other information as well.

href.js

```
var http = require('http');
var url = require('url');

http.createServer(function (req, res) {
    // Parsing url
    var queryString = url.parse(req.url,true);
    // Accessing href property of an URL
    console.log("Complete href is :-"+queryString.href);

}).listen(4200);
```

Execute the above snippet by writing node url.js and you can see the console like this:

```
H:\BACKUP\5thceit>node url.js
Complete href is :-/
Complete href is :-/bhavin
```

If we change our URL to www.customway.com/abc.html:

```
var http = require('http');
var url = require('url');

http.createServer(function (req, res) {

    var queryString = url.parse(req.url,true);
    queryString.href = "https://www.google.com//uvpce.html";
    console.log("Complete href is :-"+queryString.href);

}).listen(4200);
```

And after that, you may get href like :

```
H:\BACKUP\5thceit>node url.js
Complete href is :-https://www.google.com//uvpce.html
Complete href is :-https://www.google.com//uvpce.html
```

So in this way we can get href from the requested URL

host and hostname

Host property of a URL module provides the host associated with URL.

host.js

```
1.  var http = require('http');
2.  const { URL } = require('url');
3.
4.  http.createServer(function (req, res) {
5.
6.      var queryString = url.parse(req.url,true);
7.      // Prints the host
8.      console.log("Host is :-"+queryString.host);
9.
10.     // Prints the host name
11.     console.log("Host name is :-"+queryString.hostname);
12.
13. }).listen(4200);
```

After executing the above snippet you may get output like this:

```
PS C:\[redacted]> node host.js
Host is :-null
Host name is :-null
```

What is the difference between host and hostname?

There is one major difference between both of them, that is that host includes the port name along with hostname, where hostname property does not include the port number.

Now I'm going to create a new URL using the below snippet:

```
1.  var http = require('http');
2.  const { URL } = require('url');
3.
4.  http.createServer(function (req, res) {
5.
6.      const queryString1 = new URL('https://www.google.com:11/uvpce');
7.      console.log("Host is :-"+queryString1.host);
8.      console.log("Host name is :-"+queryString1.hostname);
9.  }).listen(4200);
```

So now, I have specified custom URL with port number 11, let's see the difference :

```
H:\BACKUP\5thceit>node url.js
Host is :-www.google.com:11
Host name is :-www.google.com
```

The host includes the port number, whereas hostname does not contain port number along with URL.

Pathname and Searchparam

Path and pathname is the combination of pathname with URL and also contains a search term along with the URL

Let's say we have URL with multiple search parameters like :

- www.demo.com/test1/test2/test3?qstring=value

In above url:

- path = /test1/test2/test3
- Search param = qstring=value

In the same way we can get this via URL module property; find the snippet below :

path.js

```
1. var http = require('http');
2. const { URL } = require('url');
3.
4. http.createServer(function (req, res) {
5.
6.     const queryString2 = new
   URL('https://www.google.com/test/test1/test2/test3?username=bhavin');
7.     console.log("Path name is :-"+queryString2.pathname);
8.     console.log("Search Parameter is :-"+queryString2.searchParams);
9. }).listen(4200);
```

Now execute it by writing node path.js, and you will get output like this:

```
H:\BACKUP\5thceit>node url.js
Path name is :-/test/test1/test2/test3
Search Parameter is :-username=bhavin
```

This way you can access the complete path along with search parameters.

Search

Search property is the same as searchParams that we have seen before

But the main difference is search property includes [?] along with all search parameters

search.js

```
1. var http = require('http');
2. const { URL } = require('url');
3. 
4. http.createServer(function (req, res) {
5. 
6.     const queryString = new URL('https://www.google.com/test/test1/test2/test3?username
   =bhavin&password=123');
7.     console.log("Search terms are :-"+queryString.search);
8. 
9. }).listen(4200);
```

Now you will be able to get the whole search term along with [?] symbol attached :

```
H:\BACKUP\5thceit>node url.js
Path name is :-/test/test1/test2/test3
Search Parameter is :-?username=bhavin&pass=joshi
Path name is :-/test/test1/test2/test3
Search Parameter is :-?username=bhavin&pass=joshi
```

As you can see both of my search terms were included along with [?] symbol

Port

Port property of a URL module returns the port associated with a URL.

port.js

```
1. var http = require('http');
2. const { URL } = require('url');
3. 
4. http.createServer(function (req, res) {
5. 
6.     const queryString = new URL('https://www.customway.com:4200');
7.     console.log("Port is :-"+queryString.port);
8. 
9. }).listen(4200);
```

I've used port number 4200 along with URL, and may get output like this:

```
PS C:\[redacted] node port.js
Port is :-4200
```

Now if I want to change port number, than we can do like this:

```
1. const queryString = new URL('https://www.customway.com:4200');
2.   queryString.port = '4500'; // changed port number to 4500
3.   console.log("Port is :-"+queryString.port);
```

And probably you will get output in the console:

- Port is :- 4500

Protocol

Protocol property is used to get specific protocols used for any request.

protocol.js

```
1. var http = require('http');
2. const { URL } = require('url');
3. 
4. http.createServer(function (req, res) {
5. 
6.     const queryString = new URL('https://www.customway.com');
7.     console.log("Protocol used :-"+ queryString.protocol);
8. 
9. }).listen(4200);
```

Now, you will get the protocol name for what you have requested :

```
PS C:\[redacted] node protocol.js
Protocol used :-https:
```

Hash

Hash property of a URL returns decorated with [#] sybmol.

Sometimes, we have pages with multiple div, and we have provided ids along with #name, so in a node, we can also access hash fragment portion attached to URL.

hash.js

```
1. var http = require('http');
2. const { URL } = require('url');
```

```
3.
4.  http.createServer(function (req, res) {
5.
6.      const queryString = new URL('https://www.customway.com#bhavin');
7.      console.log("Hash Fragment Is :-"+ queryString.hash);
8.
9.  }).listen(4200);
```

After executing node hash.js you will get output like :

```
H:\BACKUP\5thceit>node url.js
Path name is :-/test/test1/test2/test3
Search Parameter is :-username=bhavin&pass=joshi
Port is :-4500
Protocol used :-https:
Protocol used :-#bhavin
Path name is :-/test/test1/test2/test3
Search Parameter is :-username=bhavin&pass=joshi
Port is :-4500
Protocol used :-https:
Protocol used :-#bhavin
```

Example:

```
var url = require('url');
var adr = 'http://localhost:8080/default.htm?year=2020&month=AUGUST';
var q = url.parse(adr, true);

console.log(q.host); //returns 'localhost:8080'
console.log(q.pathname); //returns '/default.htm'
console.log(q.search); //returns '?year=2020&month=AUGUST’

var qdata = q.query; //returns an object: { year: 2020, month: 'AUGUST' }
console.log(qdata.month); //returns 'AUGUST'
```