

URL模块

- url模块用于处理与解析 URL。使用方法如下：

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

- URL 字符串是结构化的字符串，包含多个含义不同的组成部分。
- 解析字符串后返回的 URL 对象，每个属性对应字符串的各个组成部分。

| href | | | | | | | | | |
|----------|--------|----------|----------|----------|------|-------------------|--------|--------------|-------|
| protocol | | auth | | host | | path | | hash | |
| | | | | hostname | port | pathname | search | | |
| | | | | | | | query | | |
| " | https: | // | user | : | pass | @ sub.example.com | : | 8080 | |
| | | | | hostname | port | /p/a/t/h | ? | query=string | #hash |
| | | | | | | | | | " |
| protocol | | username | password | host | | | | | |
| origin | | | | origin | | pathname | search | | hash |
| href | | | | | | | | | |

- url 模块提供了两套 API 来处理 URL：
 - 一个是旧版本遗留的 API。

```
const url = require('url');
const myURL =
  url.parse('https://user:pass@sub.host.com:8080/p/a/t/h?
query=string#hash');
```

- 一个是实现了 WHATWG 标准的新 API。

```
const myURL =
  new URL('https://user:pass@sub.host.com:8080/p/a/t/h?
query=string#hash');
```

- 案例：

```
const myURL = new URL('https://www.abcd.com:80/p/a/t/h.html?
query=string&id=200&name=lisi#hash');

console.log(myURL.hostname); //主机名
console.log(myURL.pathname); //url路径
console.log(myURL.search); //url地址中的参数
//解析url地址中的参数
console.log(myURL.searchParams.get("id"));
console.log(myURL.searchParams.get("name"));
console.log(myURL.searchParams.get("query"));
```