

Http

Hypertext Transfer Protocol (HTTP) . HTTP is a [stateless protocol](#), meaning that the server does not keep any data (state) between two requests.

URL

Uniform Resource Locator.

http:// is the protocol.

Url 中出现的特殊字符

特殊字符	代表含义	替换内容
+	URL 中+号表示空格	%2B
空格	URL中的空格可以用+号或者编码	%20
/	分隔目录和子目录	%2F
?	分隔实际的URL和参数	%3F
%	指定特殊字符	%25
#	表示书签	%23
&	URL 中指定的参数间的分隔符	%26
=	URL 中指定参数的值	%3D

you see such protocols. Common schemes are:

Scheme	Description
data	Data URLs
file	Host-specific file names
ftp	File Transfer Protocol
http/https	Hyper text transfer protocol (Secure).
javascript	URL-embedded JavaScript code
mailto	Electronic mail address
ssh	Secure shell
tel	telephone
urn	Uniform Resource Names
view-source	Source code of the resource
ws/wss	WebSocket connections (Secure).

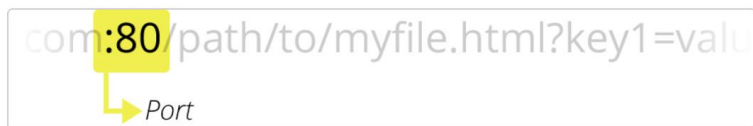
Authority



`www.example.com` is the domain name or authority that governs the namespace. It indicates which Web server is being requested.

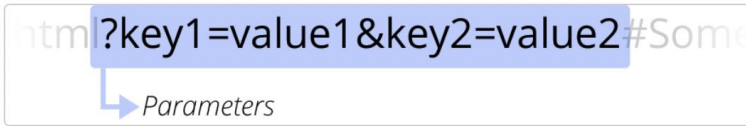
Alternatively, it is possible to directly use an [IP address](#), but because it is less convenient, it is not often used on the Web.

Port



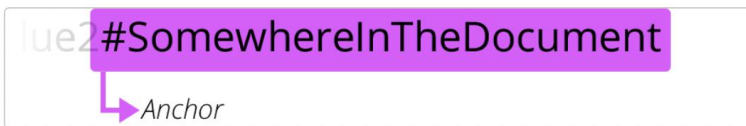
`:80` is the port in this instance. It indicates the technical "gate" used to access the resources on the web server. It is usually omitted if the web server uses the standard ports of the HTTP protocol (80 for HTTP and 443 for HTTPS) to grant access to its resources. Otherwise, it is mandatory.

Query



`?key1=value1&key2=value2` are extra parameters provided to the Web server. Those parameters are a list of key/value pairs separated with the `&` symbol. The Web server can use those parameters to do extra stuff before returning the resource to the user. Each Web server has its own rules regarding parameters, and the only reliable way to know how a specific Web server is handling parameters is by asking the Web server owner.

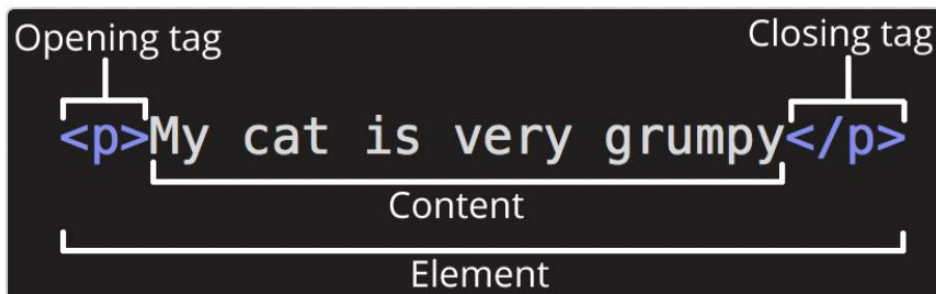
Fragment



`#SomewhereInTheDocument` is an anchor to another part of the resource itself. An anchor represents a sort of "bookmark" inside the resource, giving the browser the directions to show the content

Html

元素剖析



``斜体

``黑体

<a>

Inline elements above

Image:

Bool

<input type="text" disabled="disabled" />

<!-- using the disabled attribute prevents the end user from entering text into the input box -->

<input type="text" disabled />

<!-- text input is allowed, as it doesn't contain the disabled attribute -->

<input type="text" />

1. <!DOCTYPE html>: 文档类型。在 HTML 还年轻的时候（1991-1992），
doctypes 旨在充当一组规则的链接，HTML 页面必须遵循这些规则才能被认为是好的 HTML。Doctypes 曾经看起来像这样：
2. <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"<http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd>">
复制到剪贴板
3. 最近，doctype 是一个历史工件，需要包含它才能使其他一切正常工作。
<!DOCTYPE html>是算作有效文档类型的最短字符串。这就是您需要知道的一切！
4. <html></html>:html元素。该元素包装了页面上的所有内容。它有时被称为根元素。
5. <head></head>:head元素。此元素充当您要包含在 HTML 页面上的所有内容的容器，**这不是**页面将显示给查看者的内容。这包括将出现在搜索结果中的关键字和页面描述、样式内容的 CSS、字符集声明等。您将在本系列的下一篇文章中了解更多相关信息。

6. `<meta charset="utf-8">`:[<meta>](#)元素。此元素表示不能由其他 HTML 元相关元素（如[<base>](#)、[<link>](#)、[<script>](#)或[<style>](#)）表示的元数据[<title>](#)。这些 [charset](#) 属性将文档的字符集设置为 UTF-8，其中包括来自绝大多数人类书面语言的大多数字符。使用此设置，页面现在可以处理它可能包含的任何文本内容。没有理由不设置这个，它可以帮助避免以后出现一些问题。

7. `<title></title>`:[<title>](#)元素。这设置了页面的标题，它是显示在加载页面的浏览器选项卡中的标题。页面标题还用于在添加书签时描述页面。

8. `<body></body>`:[<body>](#)元素。这包含页面上显示的所有内容，包括文本、图像、视频、游戏、可播放的音轨或其他任何内容。

特殊字符

文字字符	字符引用等价物
<	<code>&lt;</code>
>	<code>&gt;</code>
"	<code>&quot;</code>
'	<code>&apos;</code>
&	<code>&amp;</code>

注释

```
<p>I'm not inside a comment</p>
```

```
<!-- <p>I am!</p> -->
```

Metadata（元数据

```
<head>
```

```
<meta charset="utf-8" />
```

```
<title>My test page</title>
```

</head>

Title 是 document

<meta name="author" content="Chris Mills" />

<meta

name="description"

content="The MDN Web Docs Learning Area aims to provide
complete beginners to the Web with all they need to know to get
started with developing web sites and applications." />

自定义图标:

<link rel="icon" href="favicon.ico" type="image/x-icon" />

Css link

<link rel="stylesheet" href="my-css-file.css" />

设置语言

<html lang="en-US">

...

</html>

<p>Japanese example: ご飯が熱い。.</p>