Understanding urls

<http://www.webreference.com/html/tutorial2/2.html>

<https://developer.mozilla.org/en-US/Learn/Understanding_URLs>

**URL** stands for *Uniform Resource Locator*.

**http://www.example.com:80/path/to/myfile.html?key1=value1&key2=value2#SomewhereInTheDocument**

1. **http:// ( HyperText Transfer Protocol (HTTP) or HyperText Transfer Protocol Secure ([HTTPS](https://www.instantssl.com/ssl-certificate-products/https.html?key5sk1=eaf2dac9e6c6065da886c52906de7306e4719c6f" \t "_blank)))**

This is the protocol.

With **HTTPS** if anyone in between the sender and the recipient could open the message, they still could not understand it. Only the sender and the recipient, who know the "code," can decipher the message.

1. [**www.example.com**](http://www.example.com)

This is the domain name. It indicates which web server is being requested.

1. **:80**

This is the port. It indicates the technical “gate” used to access the resources on the web server. 80 for HTTP and 443 for HTTPS).  These ports allow software applications to share hardware resources without interfering with each other. Port numbers are most commonly used with [TCP/IP](http://compnetworking.about.com/cs/basictcpip/g/bldef_tcpip.htm) connections. Port numbers are associated with [network addresses](http://compnetworking.about.com/od/basicnetworkingconcepts/g/bldef_address.htm). In both TCP and UDP, port numbers start at 0 and go up to 65535. Numbers in the lower ranges are dedicated to common Internet protocols (like 21 for [FTP](http://compnetworking.about.com/od/networkprotocols/g/bldef_ftp.htm) and 80 for [HTTP](http://compnetworking.about.com/od/networkprotocols/g/bldef_http.htm)).

1. **path/to/myfile.html**

This is the path to the resource on the Web server. In the early days of the Web, a path like this represented a physical file location on the Web server. Nowadays, it is mostly an abstraction handled by Web servers without any physical reality.

1. **?key1=value1&key2=value2**

These 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. Each Web server has its own rules regarding parameters, and the only reliable way to know if a specific Web server is handling parameters is by asking the Web server owner.

1. **#SomewhereInTheDocument**

This 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 located at that "bookmarked" spot. On an HTML document, for example, the browser will scroll to the point where the anchor is defined; on a video or audio document, the browser will try to go to the time the anchor represents. It is worth noting that the part after the #, also known as fragment identifier, is never sent to the server with the request.