Absolute and Relative Paths

You are at sometime going to want to use links within your web page. These may be links to other pages, images, included items and so on. There are two ways of specifying these links, ABOLUTE and RELATIVE. Get them wrong and you'll have some very strange results. This explanation might help you in learning the difference.

Absolute Path

These are absolute server paths, they are relative to your main folder (i.e. the folder where your index or default file is held – also known as the Root folder.

/ tells the server to look for the Root folder

/mainpage.html tells the server to find the mainpage file which is is in a folder called folder, which is in the Root folder

/folder/mainpage.html tells the server to find the mainpage file which is in a folder called folder, which is in the Root folder

The following absolute path is relative to the world wide web: in other words it tells the server to go onto the internet and find mainpage within your domain

http://yourdomain.com/mainpage.html

Relative Path

The following is relative to a file residing in the same folder:

anotherpage.html

This relative path points to a file which is one folder up:

`../anotherpage.html`

More Examples:

The absolute path to the main page of a typical web site would be:

`http://yourdomain.com/index.html`

A page residing in http://yourdomain.com/folder/mainpage.html that links to your home page would use one of these absolute paths:

`http://yourdomain.com/index.html`

`/index.html`

`/`

The last two examples are absolute server paths, you may think of the first forward slash in these paths as representing your domain.

Note: Your main page must be named "index" (a few hosts use "default" instead) but may bear any of the following file extensions: index.shtml, index.htm, index.html, index.shtm, index.php, index.cgi …., in each case using the absolute server path: / would send you to the home page.

If your html file is in /home/example.html and your image file is in /images/example.jpg you can embed the image into the page example.html using an absolute path to the file

`<img src="images/example.jpg">`

A relative path to the same file would be:

`<img scr="../images/example.jpg">`

With an absolute path, it doesn't matter where the html file calling the other file resides.

You may place example.html in: /pages/example.html and embed the image as:

<img src="homes/images/example.jpg:>

With a relative path, if you place example.htm in : /pages/example.html, and embed the image as:

<img src="images/example.jpg">   a relative path)

then your image file will not load. This is because example.html is looking for /pages/images/example.jpg, which does not exist.

LINKS

Paths used in links work in the same manner as those used in images. For example image example.html is in /home/example.html, and it contains a relative link to another page.html. This link would appear as:

<a href="homes/anotherpage.html">

If you move your example.html file to some other folder within your web site them the link will no longer work. This is because, by asking for a file via a relative link, you are telling the user agent, the browser, that the file you have created a link to is located within the same folder as that of the calling file.

Problems Using Absolute paths with SSL

If you are not using a SSL certificate when a webpage changes from http protocol to https (SSL), if you embed any images by absolute paths without domain names (/images/o.jpg), they will be broken. If you embed images with full URL absolute paths

(http://yourdomain.com/images/o.jpg), the images will show up, but the user will get warning messages that the page is mix between secure and non-secure items.

On web pages that make transitions between http and https:, one should use relative paths to avoid these problems.

Credit: I learned this back in 2007 from jbvisions.co.uk who was a mentor when I was first learning to hand code website. She is no longer in business but I owe credit to her.
This foundational concept is used in coding for web design and development, CCS styling and HTML code.