ScopeTreasure Courses

HTML COURSES · LESSON 1.3 · CODE SCOPE SERIES

Links & Images

Learn the a tag and href, target _blank, absolute vs relative paths, and the img tag with alt text — plus how to fix a broken image.

Introduction

A web page alone is an island. Links are the bridges: they join page to page, site to site — that is why it is called a *web*. Images do the second big job: they show what words cannot — a product, a place, a face. In this lesson you learn the `<a>` tag for links, the two kinds of addresses (absolute and relative), and the `<img>` tag with its `alt` text, so your pages can travel and can show. *(Roman Urdu: Link page ko page se jorta hai, aur tasveer wo dikhati hai jo lafz nahi bata sakte.)*

Key points

- A **link** is made with the `<a>` tag. The address goes in the **href** attribute: `<a href="https://www.google.com">Open Google</a>` — the text between the tags is what the visitor clicks. `href` is short for *hypertext reference* — in simple words: the address where the click takes you.
- `target="_blank"` opens the link in a **new tab**. Without it, the link opens in the same tab.
- **Absolute path** = the full address, starting with `https://`. Use it for **other websites**.
- **Relative path** = the address inside your own site folder, like `about.html` or `images/photo.jpg`. Use it for **your own pages and files**.

  (Roman Urdu: Apni site ke andar relative path, bahir ki site ke liye pura address.)
- An **image** is shown with the `<img>` tag. It has **no closing tag**: `<img src="images/photo.jpg" alt="My college building">`
  - `src` = where the image file is.
  - `alt` = a short description of the image.
- **Alt text matters:** it appears if the image cannot load, screen readers read it to blind users, and Google uses it to understand the image. Never leave it empty for a real photo. Good alt text describes the photo in one short line — as if you are telling a friend on the phone what the photo shows.
- **Broken image?** The file path is almost always wrong: check the file name spelling, the folder name, and whether the file is really in that folder.
- **Bonus from the lesson:** an image can be a link — put the `<img>` tag *inside* an `<a>` tag.

Code examples

Example 1 — Links: one outside, one inside

```
<a href="https://www.youtube.com" target="_blank">YouTube (new tab)</a>
<a href="about.html">About Me (my own page)</a>
```

What it does: The first link opens YouTube in a new tab (absolute path). The second opens your own `about.html` page from the same folder (relative path).

What you will see: two clickable lines, shown in the browser's link colour and underlined. Clicking the first opens YouTube in a fresh tab while your page stays open behind it; clicking the second replaces your page with the About page in the same tab.

Example 2 — An image that is also a link

```
<a href="https://toolscope.netlify.app">
  <img src="images/toolscope.png" alt="ToolScope free online tools">
</a>
```

What it does: Shows the image `toolscope.png` from the `images` folder. Clicking the image takes the visitor to ToolScope. If the image fails to load, the alt text "ToolScope free online tools" is shown instead.

What you will see: the picture itself on the page, and when you move the mouse over it, the pointer turns into a hand — the sign of a link. Click, and you land on ScopeTreasure. If the file is missing, you see a small broken-image icon with the alt text standing in its place.

Example 3 — The same photo, first broken, then fixed

```
<p>Wrong way – the photo is not in this folder:</p>
<img src="photo.jpg" alt="My college building">

<p>Right way – folder name is part of the path:</p>
<img src="images/photo.jpg" alt="My college building">
```

What it does: Both lines ask for the same photo, but the photo lives inside the `images` folder. The first `src` looks for it next to the page and fails; the second `src` walks into the folder and finds it.

What you will see: after the first line, a broken-image icon with the words "My college building" — the alt text doing its rescue job. After the second line, the real photo. One word (`images/`) is the whole difference.

Quick recap

- `<a href="...">` makes a link; the text between the tags is what gets clicked.
- `target="_blank"` opens the link in a new tab.

- **Absolute** path = full `https://` address, for other websites. **Relative** path = address inside your own folder, for your own pages and files.
- `<img>` shows a picture; it needs `src` (the file) and `alt` (its description), and has no closing tag.
- Broken image = the path is wrong: check the folder name and the file name spelling, letter by letter.

Common mistakes

1. **Wrong relative path** — the image is in the `images` folder but the code says `src="photo.jpg"`. The browser looks in the wrong place and shows a broken image.
2. **Missing or useless alt text** — `alt=""` or `alt="image"` tells Google and screen readers nothing. Write what the image shows.
3. **Forgetting the quotes** — write `href="about.html"`, not `href=about.html`. Values of attributes go inside quotes.

Practice

1. Make two pages, `index.html` and `about.html`, and link them to each other: index links to about, about links back to index.
2. Add a photo to your page with a proper `alt` text, then make the photo clickable so it opens your favourite website in a new tab.

Practice quiz

1. What does <a> do?

- Shows an image
- Creates a link to another page
- Makes a list
- Makes a heading

Answer: Creates a link to another page

2. What does <img> do?

- Shows an image on the page
- Plays a video
- Makes a table
- Collects input

Answer: Shows an image on the page

Watch the video lesson: https://youtu.be/qo7uDVTEeQc

Practice tool: Image Resizer — https://scopetreasure.com/tool/image-resize

Read online: https://scopetreasure.com/notes/html-links-and-images · More free tools and notes at https://scopetreasure.com