

[Skip to main content](#)
[Accessibility help](#)
[AI Mode](#)
[All](#)
[Forums](#)
[Shopping](#)
[Images](#)
[More](#)
[Sign in](#)

(1) can python list all file names in a directory

Yes, Python can easily list all file names in a directory using several built-in modules.

The most common ways to do this depend on whether you want to list files in just the main folder or include subfolders as well.

1. In a single folder (Modern method)

The modern, recommended approach uses the `pathlib` module. It treats paths as objects, making your code easier to read.

python

```
from pathlib import Path
```

```
# Specify the directory path
```

```
dir_path = Path('/path/to/directory')
```

```
# List only files (excluding subfolders)
```

```
files = [f.name for f in dir_path.iterdir() if f.is_file()]
```

```
print(files)
```

Use code with caution.

2. In a single folder (Legacy method)

You can also use the traditional `os` module. By default, `os.listdir()` returns both files and folders, so you need to filter for files specifically.

python

```
import os
```

```
dir_path = '/path/to/directory'
```

```
# Filter out directories to get file names only
```

```
files = [f for f in os.listdir(dir_path) if os.path.isfile(os.path.join(dir_path, f))]
```

```
print(files)
```

Use code with caution.

3. Including all subfolders (Recursive)

If you need to look inside a folder and every subfolder beneath it, use `rglob` from `pathlib` or `os.walk`.

Using Pathlib:

python

```
from pathlib import Path
```

```
dir_path = Path('/path/to/directory')
```

```
# The "*" pattern grabs all files recursively
files = [f.name for f in dir_path.rglob('*') if f.is_file()]
print(files)
```

Use code with caution.

Using OS:

python

```
import os
```

```
dir_path = '/path/to/directory'
files = []
```

```
for root, directories, filenames in os.walk(dir_path):
    for filename in filenames:
        files.append(filename)
```

```
print(files)
```

Use code with caution.

Would you like to filter these files by a specific file extension (like .txt or .png), or do you need to get the full file path instead of just the names?

Stack Overflow

How do I list all files of a directory? - python - Stack Overflow

os.listdir() returns everything inside a directory -- including both files and directories. os.path 's isfile() can be used to onl...

GeeksforGeeks

Python - List Files in a Directory - GeeksforGeeks

Python - List Files in a Directory * Sometimes, while working with files in Python, a problem arises with how to get all files in ...

PYnative

Python List Files in a Directory [5 Ways] - PYnative

To list all files in a directory using Python, you can use the built-in os module. Also, there are multiple ways to list files in ...

Show all

(2) can it wrap the results in a html link tag

Yes, Python can wrap the file names in HTML link tags by using string formatting (like f-strings) as it loops through the directory.

Here is how you can do it using pathlib for a single folder:

python

```

from pathlib import Path

dir_path = Path('/path/to/directory')
base_url = "https://example.com" # The web folder where files are hosted

# Generate HTML links
html_links = [
    f'<a href="{base_url}{f.name}">{f.name}</a>'
    for f in dir_path.iterdir() if f.is_file()
]

# Print them out line by line
for link in html_links:
    print(link)

```

Use code with caution.

Output Example

If your folder contains report.pdf and photo.png, the script will output:
html

```

<a href="https://example.com">report.pdf</a>
<a href="https://example.com">photo.png</a>

```

Use code with caution.

Writing the output directly to an HTML file

If you want to save these links directly into an .html file you can open in a browser, use this script:

python

```

from pathlib import Path

dir_path = Path('/path/to/directory')
output_file = Path('links_list.html')
base_url = "https://example.com"

# Create and write to the HTML file
with open(output_file, 'w', encoding='utf-8') as html_file:
    html_file.write("<html>\n<body>\n  <ul>\n")

    for f in dir_path.iterdir():
        if f.is_file():
            link = f'    <li><a href="{base_url}{f.name}">{f.name}</a></li>\n'
            html_file.write(link)

    html_file.write("  </ul>\n</body>\n</html>")

print(f"HTML file created successfully at: {output_file.absolute()}")

```

Use code with caution.

Do you need the link's href attribute to point to a website URL (like the example above) or to the local file path on your computer?

The freeCodeCamp Forum

Step 14 - Turn the image into a link - Programming

Jun 29, 2022 – You've already turned the words “cat photos” into a link in the p element above the image. You did this by wrapping the words in o...

YouTube

HTML Links and Anchor Tags - YouTube

Apr 27, 2020 – HTML Links and Anchor Tags made of if you have a look at each and every one of these links. if you right click on it and click ins...

YouTube

HTML Anchor Tag #htmlacademy #htmlbasics #code #html #beginners #htmlbasics #htmltutorial #learnhtml - YouTube

Nov 21, 2024 – HTML Anchor Tag #htmlacademy #htmlbasics #code #html #beginners #htmlbasics #htmltutorial #learnhtml website all you need is the A...

Show all

(3) wrap results in an image tag instead of a link tag

Yes, Python can wrap the file names in HTML image tags by using an tag and assigning the file name to the src attribute.

Here is how you can do it using pathlib for a single folder:

python

```
from pathlib import Path
```

```
dir_path = Path('/path/to/directory')
```

```
base_url = "https://example.com" # The web folder where images are hosted
```

```
# Generate HTML image tags
```

```
html_images = [  
    f''   
    for f in dir_path.iterdir() if f.is_file()  
]
```

```
# Print them out line by line
```

```
for img in html_images:  
    print(img)
```

Use code with caution.

Output Example

If your folder contains logo.png and banner.jpg, the script will output:

html

```
  

```

Use code with caution.

Writing the output directly to an HTML file

If you want to save these image tags directly into an .html file to preview a gallery in your browser, use this script:

python

```
from pathlib import Path
```

```
dir_path = Path('/path/to/directory')
```

```
output_file = Path('image_gallery.html')
```

```
base_url = "https://example.com"
```

```
# Create and write to the HTML gallery file
```

```
with open(output_file, 'w', encoding='utf-8') as html_file:
```

```
    html_file.write("<html>\n<body>\n  <div class=\"gallery\">\n")
```

```
    for f in dir_path.iterdir():
```

```
        if f.is_file():
```

```
            # Using f.stem extracts just the file name without the extension for the  
alt text
```

```
            img_tag = f'    \n'
```

```
            html_file.write(img_tag)
```

```
    html_file.write("  </div>\n</body>\n</html>")
```

```
print(f"HTML gallery created successfully at: {output_file.absolute()}")
```

Use code with caution.

Would you like to filter this script so it only processes image file types (like .png, .jpg, or .webp) and ignores other files?

YouTube

HTML Image Tag Explained | How To Add Image In HTML | HTML Tutorial for Beginners | Simplicode - YouTube

Mar 21, 2022 – HTML Image Tag Explained | How To Add Image In HTML | HTML Tutorial for Beginners | Simplicode so this is the tag. and we already ...

YouTube

How to Use CSS Javascript and Images With Flask Static Files - Flask Fridays #7 - YouTube

Feb 26, 2021 – How to Use CSS Javascript and Images With Flask Static Files - Flask Fridays #7 now in each of these folders we're going to put ea...

YouTube

HTML Tutorial: Page Formatting

Sep 12, 2011 – Anytime you want to preview your webpage just save it as an html file and open it up in a web browser. Likewise to add a picture, ...

Show all

(4) can the base_url be a windows directory

Yes, the src attribute can point to a Windows directory, but it requires a specific format known as a File URL (file:///) for web browsers to display the images correctly.

Browsers cannot read standard Windows paths like C:\images\logo.png directly inside HTML tags due to security restrictions. Instead, they must be formatted like this: file:///C:/images/logo.png (using forward slashes).

Python's pathlib module handles this transformation automatically using the .as_uri() method.

The Python Script
python

```
from pathlib import Path
```

```
# Use 'r' before the string to handle Windows backslashes properly
```

```
image_folder = Path(r"C:\Users\YourName\Pictures")
```

```
output_file = Path("local_gallery.html")
```

```
with open(output_file, "w", encoding="utf-8") as html_file:
```

```
    html_file.write("<html>\n<body>\n  <div class='gallery'>\n")
```

```
    for f in image_folder.iterdir():
```

```
        # Check if it is a file and ensure it is an image
```

```
        if f.is_file() and f.suffix.lower() in [".jpg", ".jpeg", ".png", ".webp",  
".gif"]:
```

```
            # f.as_uri() automatically converts 'C:\path\to\file.png' into  
'file:///C:/path/to/file.png'
```

```
            file_url = f.as_uri()
```

```
            img_tag = f'    \n'
```

```
            html_file.write(img_tag)
```

```
    html_file.write("  </div>\n</body>\n</html>")
```

```
print(f"HTML gallery created! Open this file in your browser:  
{output_file.absolute()}")
```

Use code with caution.

Output Example

Inside the generated HTML file, the image tags will look like this:

html

```

```

```

```

Use code with caution.

Important Security Note

Modern web browsers (like Chrome, Edge, and Firefox) will only display these local

file:/// images if you open the generated HTML file locally from your own computer (e.g., double-clicking the local_gallery.html file). If you upload this HTML file to a live website, the images will look broken to other users because the website cannot look inside their local C: drives.

Do you plan to look at this HTML gallery locally on your own machine, or will this script be running on a local web server?

script will be used to create a photo album of images inside the same directory as the html

If the HTML file is in the same folder as the images, you do not need a base_url or Windows path at all. Instead, you should use relative paths, which simply mean using the file name itself (e.g., src="photo.jpg").

This is the best approach because it makes your photo album completely portable. You can move the entire folder to another drive, a different computer, or upload it to the web, and the images will never break.

The Python Script

Place this script inside your image folder and run it. It will find all images in its own directory and generate the HTML file right next to them.

python

```
from pathlib import Path
```

```
# Get the directory where this script is currently running
```

```
current_dir = Path(__file__).parent if "__file__" in locals() else Path.cwd()
```

```
output_file = current_dir / "photo_album.html"
```

```
# Supported image extensions
```

```
valid_extensions = {".jpg", ".jpeg", ".png", ".webp", ".gif", ".bmp"}
```

```
with open(output_file, "w", encoding="utf-8") as html_file:
```

```
    # Basic HTML setup with responsive CSS grid styling
```

```
    html_file.write("""<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
    <meta charset="UTF-8">
```

```
    <title>Photo Album</title>
```

```
    <style>
```

```
        body { font-family: sans-serif; background-color: #f4f4f4; margin: 20px; }
```

```
        .album-grid { display: grid; grid-template-columns: repeat(auto-fill, minmax(250px, 1fr)); gap: 20px; }
```

```
        .photo-card { background: white; padding: 10px; border-radius: 8px; box-shadow: 0 2px 5px rgba(0,0,0,0.1); text-align: center; }
```

```
        .photo-card img { max-width: 100%; height: 200px; object-fit: cover; border-radius: 4px; }
```

```
        .photo-title { margin-top: 10px; font-size: 14px; color: #333; word-break: break-all; }
```

```
    </style>
```

```
</head>
```

```
<body>
```

```
    <h1>My Photo Album</h1>
```

```
    <div class="album-grid">
```

```

"""
# Loop through the files in the current folder
for f in current_dir.iterdir():
    # Ensure we only include valid images and skip the output HTML file itself
    if f.is_file() and f.suffix.lower() in valid_extensions:
        # Using just f.name creates a clean, portable relative path
        img_tag = f"""
            <div class="photo-card">
            
            <div class="photo-title">{f.name}</div>
            </div>\n"""
        html_file.write(img_tag)

    html_file.write("""
</div>
</body>
</html>""")

print(f"Success! Created '{output_file.name}' in the image directory.")

```

Use code with caution.

Output Example

Because the HTML file is in the same directory, the generated code stays minimal and clean:

html

```

<div class="photo-card">
    
    <div class="photo-title">vacation_01.jpg</div>
</div>

```

Use code with caution.

Would you like to add a feature to sort the photos by a specific order (such as alphabetically or by the date they were taken)?

YouTube

Level 2 HTML Lesson 5: Inserting an image and using attributes - YouTube

May 21, 2022 – Level 2 HTML Lesson 5: Inserting an image and using attributes

and then we go src in other words. source okay image source equals.

Canto

Easy Guide to Insert an HTML Photo Step-by-Step

Mar 31, 2026 – How to insert an image with HTML Path type When to use Example

Relative path Image is in the same directory or subfolders src="ima...

Iqra Technology

Comprehensive HTML File Paths Guide | Free HTML Training

• Relative Paths: • Same Directory: Use the filename directly (e.g., src="image.jpg"). • Subdirectory: Include the subdirectory i...

Show all

(5) add link to display image in a new window when the image is clicked

You can do this by wrapping the tag inside an <a> (link) tag and setting the target="_blank" attribute. This tells the browser to open the image file in a brand-new tab or window when a user clicks on it.

Here is the updated script. It uses the same clean, portable relative paths as before.

python

```
from pathlib import Path
```

```
# Get the directory where this script is currently running
```

```
current_dir = Path(__file__).parent if "__file__" in locals() else Path.cwd()
```

```
output_file = current_dir / "photo_album.html"
```

```
# Supported image extensions
```

```
valid_extensions = {".jpg", ".jpeg", ".png", ".webp", ".gif", ".bmp"}
```

```
with open(output_file, "w", encoding="utf-8") as html_file:
```

```
    # HTML setup with grid styling and a zoom effect on hover
```

```
    html_file.write("""<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
    <meta charset="UTF-8">
```

```
    <title>Photo Album</title>
```

```
    <style>
```

```
        body { font-family: sans-serif; background-color: #f4f4f4; margin: 20px; }
```

```
        .album-grid { display: grid; grid-template-columns: repeat(auto-fill, minmax(250px, 1fr)); gap: 20px; }
```

```
        .photo-card { background: white; padding: 10px; border-radius: 8px;
```

```
box-shadow: 0 2px 5px rgba(0,0,0,0.1); text-align: center; }
```

```
        .photo-card img { max-width: 100%; height: 200px; object-fit: cover; border-radius: 4px; transition: transform 0.2s; }
```

```
        /* Slight zoom effect to let users know the image is clickable */
```

```
        .photo-card img:hover { transform: scale(1.03); }
```

```
        .photo-title { margin-top: 10px; font-size: 14px; color: #333; word-break: break-all; }
```

```
    </style>
```

```
</head>
```

```
<body>
```

```
    <h1>My Photo Album</h1>
```

```
    <p style="color: #666; margin-bottom: 20px;">Click any image to view it full size in a new window.</p>
```

```
    <div class="album-grid">
```

```
""")
```

```
# Loop through the files in the current folder
```

```
for f in current_dir.iterdir():
```

```
    if f.is_file() and f.suffix.lower() in valid_extensions:
```

```
        # The <a> tag wraps the <img> tag. target="_blank" opens it in a new window.
```

```

        img_tag = f"""            <div class="photo-card">
        <a href="{f.name}" target="_blank">
            
        </a>
        <div class="photo-title">{f.name}</div>
    </div>\n"""
        html_file.write(img_tag)

    html_file.write("""    </div>
</body>
</html>""")

print(f"Success! Created '{output_file.name}' with clickable images.")

```

Use code with caution.

Generated HTML Structure

The resulting code inside your HTML file will now look like this for each image:
html

```

<div class="photo-card">
    <a href="vacation_01.jpg" target="_blank">
        
    </a>
    <div class="photo-title">vacation_01.jpg</div>
</div>

```

Use code with caution.

Would you like to add sorting options next, or perhaps a way to display captions instead of just the raw file names?

Cloudinary

How to Hyperlink an Image in HTML (and Emails) The Right Way

Jul 14, 2025 – Common Mistakes and Fixes Only using Wrap it in an tag Missing alt text Always add for SEO & accessibility Using relative paths in...

YouTube

Videoguide - Links in Web Page Using Images and Maps in Dreamweaver, Insert Map, Rollover, Target - YouTube

Nov 22, 2024 – Videoguide - Links in Web Page Using Images and Maps in Dreamweaver, Insert Map, Rollover, Target the the widths so it goes also h...
edureka.co

What are HTML Images and How to Modify you Web Page?

Oct 22, 2024 – You can also use image as a link where the user can click on the image and land on a new webpage. To do so, you just have to put t...

Show all

AI Mode response is ready

Ask about