Perfect! That's actually the **recommended way in Django**—keeping static files per app—because it scales better for multiple apps. Let's go **step by step**.

Step 1: Create `static` Folder Inside Your App

Suppose your app is called `myapp`. Your folder structure should look like this:

```
myproject/
├── myapp/
│   ├── static/
│   │   └── myapp/
│   │       ├── images/
│   │       │   ├── image1.jpg
│   │       │   └── image2.png
│   │       └── css/
│   │           └── style.css
│   ├── templates/
│   │   └── home.html
│   ├── views.py
├── manage.py
├── settings.py
```

Important:

- Inside `static/` folder, create a folder with your app's name (`myapp`) to avoid name clashes when you have multiple apps.
- Images go in `static/myapp/images/`.
- CSS can go in `static/myapp/css/`.

Step 2: Configure `settings.py` (Optional for app static)

If you are using per-app static files, Django automatically detects them. You just need:

```python
# settings.py

import os
STATIC_URL = '/static/'

# Add this if you have a separate folder for static files
STATICFILES_DIRS = [os.path.join(BASE_DIR, 'static')]
```

No need to modify `STATICFILES_DIRS` unless you have global static folders outside apps.

Step 3: Load Static in Template

In your template `home.html`:

```
{% load static %}

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>App Static Images</title>
    <link rel="stylesheet" href="{% static 'myapp/css/style.css' %}">
</head>
<body>
    <h1>Images from App Static Folder</h1>

    <!-- JPG Image -->
    <img src="{% static 'myapp/images/image1.jpg' %}" alt="Image 1"
width="300">

    <!-- PNG Image -->
    <img src="{% static 'myapp/images/image2.png' %}" alt="Image 2"
width="300">

    <!-- Styled Image -->
    <img src="{% static 'myapp/images/image1.jpg' %}" alt="Styled Image"
class="styled-image">
</body>
</html>
```

Step 4: CSS for Styling (`static/myapp/css/style.css`)

```
.styled-image {
    border: 2px solid #2196F3;
    border-radius: 8px;
    box-shadow: 3px 3px 10px rgba(0,0,0,0.2);
}
```

Step 5: Update `views.py`

`# myapp/views.py`

```
from django.shortcuts import render

def home(request):
    return render(request, 'home.html')
```

Step 6: Update `urls.py`

```python
# myproject/urls.py
from django.urls import path
from myapp import views

urlpatterns = [
    path('', views.home, name='home'),
]
```

No need to add `static()` in `urls.py` for development; Django automatically serves app-level static files in `DEBUG = True`.

Step 7: Run Server and Test

```
python manage.py runserver
```

Go to `http://127.0.0.1:8000/` and you should see:

- JPG image
- PNG image
- Styled image

All **served from `myapp/static/myapp/`.**

This setup is **cleaner** because:

1. Each app manages its own static files.
2. No conflicts between apps with same file names.
3. Easy to scale for bigger projects.