

<https://www.halvorsen.blog>

django

# Django Static Files

Hans-Petter Halvorsen



# Contents

- Configuration of Static Files in Django
- Static Files Examples
  - Images
  - CSS
  - JavaScript

# Introduction

- “Static files” are files like CSS files, JavaScript files and images that are not dynamically generated by the application.
- Django provides a built-in app called “staticfiles” to manage these files.

# Resources

- Django - Add Static File:  
[https://www.w3schools.com/django/django\\_add\\_static\\_files.php](https://www.w3schools.com/django/django_add_static_files.php)
- How to manage static files (e.g. images, JavaScript, CSS):  
<https://docs.djangoproject.com/en/5.2/howto/static-files/>
- Django Static File:  
<https://www.geeksforgeeks.org/python/django-static-file/>

<https://www.halvorsen.blog>

django

# Configuration



Hans-Petter Halvorsen

[Table of Contents](#)

# How to Use Static Files

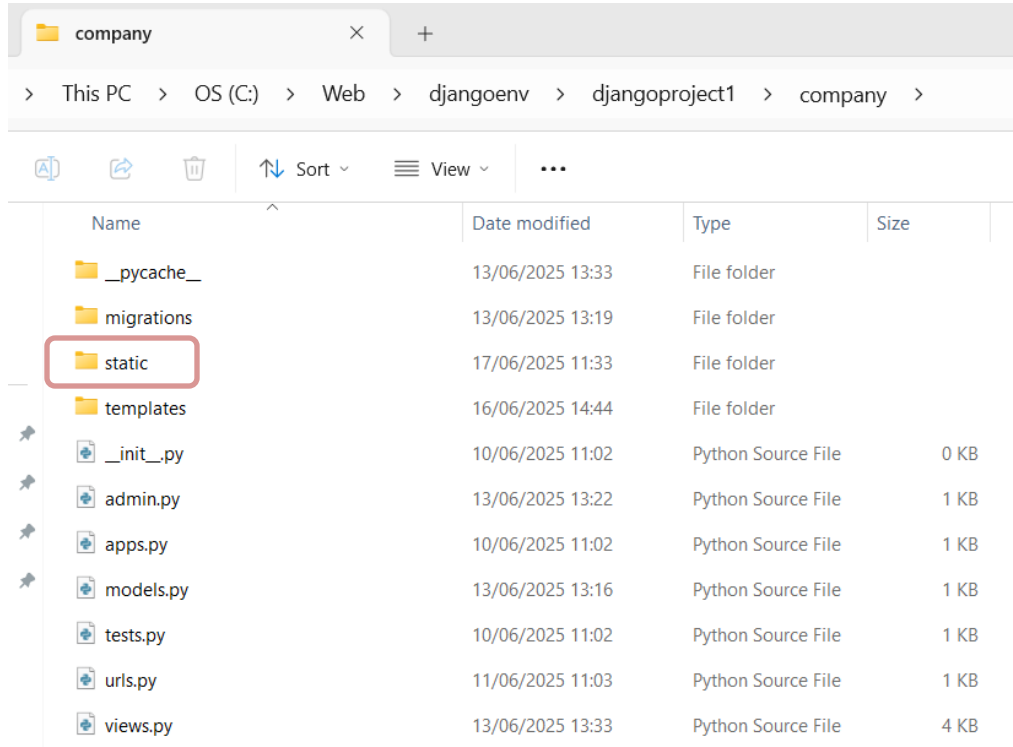
- Include “django.contrib.staticfiles” in “INSTALLED\_APPS” in “settings.py”.
- Define “STATIC\_URL” in “settings.py”.
- Optionally define “STATIC\_ROOT” and “STATICFILES\_DIRS” in “settings.py”.
- Place static files in a “static” folder within your app or in directories specified in “STATICFILES\_DIRS”.
- In templates, load the static tag using `{% load static %}` and use `{% static 'path/to/file' %}` to reference static files.
- During development, Django automatically serves static files when `DEBUG = True`.
- For production, run “python manage.py collectstatic” to gather all static files into `STATIC_ROOT` and configure your web server to serve them.

# Note!

- When “DEBUG = False”, Django does not automatically serve static files. You must use “python manage.py collectstatic” and configure your web server.
- “STATIC\_ROOT” is primarily used for deployment in production, while “STATICFILES\_DIRS” specifies additional locations for static files in development.
- The command “python manage.py collectstatic” gathers static files from your apps and other locations into the “STATIC\_ROOT” directory for deployment.
- When using “DEBUG = False” you must specify allowed hosts using “ALLOWED\_HOSTS”.

# Create “static” folder

Create a new folder, either in your project folder or in your app folder.



# “settings.py”

Make sure that “django.contrib.staticfiles” is included in “INSTALLED\_APPS” section in the “settings.py” file.

```
settings.py X
django project1 > settings.py > ...
33 INSTALLED_APPS = [
34     'django.contrib.admin',
35     'django.contrib.auth',
36     'django.contrib.contenttypes',
37     'django.contrib.sessions',
38     'django.contrib.messages',
39     'django.contrib.staticfiles',
40     'company'
41 ]
```

Define “STATIC\_URL” the “settings.py” file as well:

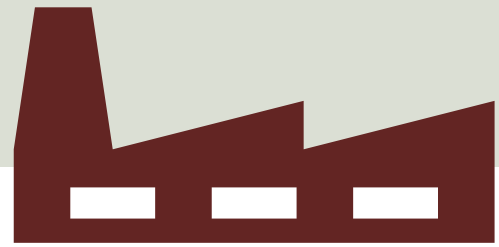
```
settings.py X
django project1 > settings.py > ...
114
115 # Static files (CSS, JavaScript, Images)
116 # https://docs.djangoproject.com/en/5.2/howto/static-files/
117
118 STATIC_URL = 'static/'
119
```

Both these settings was enabled by default in my case, so I did not need to add them.

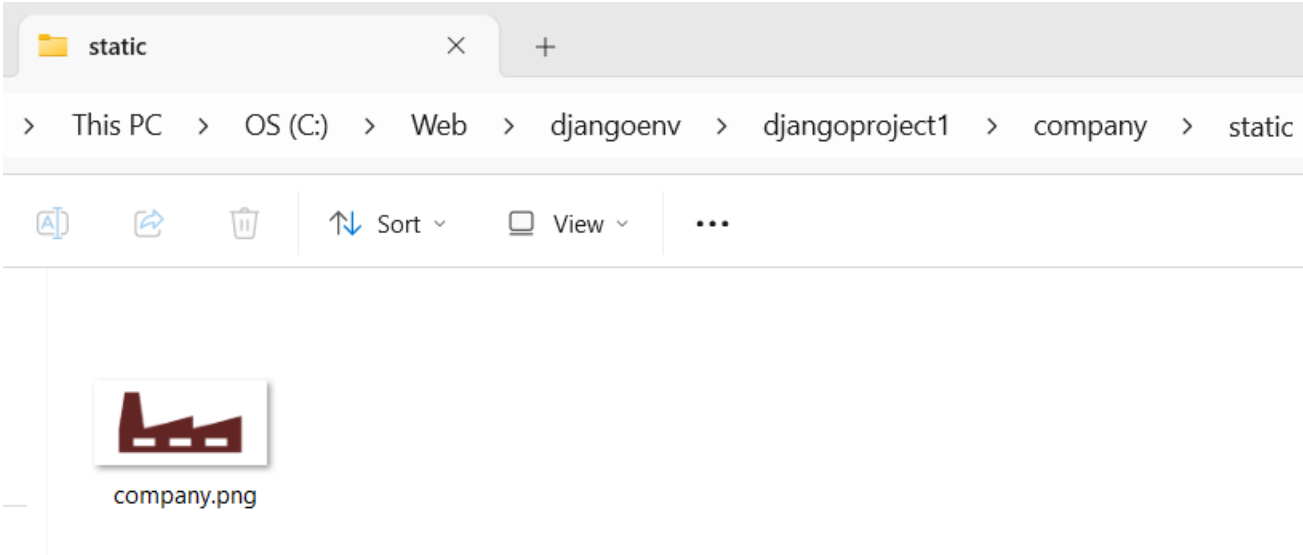
# Images



# Use Images



We add an image into the “static” folder:



# “company.html” Template

```
{% load static %}
```

```
<!DOCTYPE html>  
<html>  
<head>  
<title>Company App</title>  
<meta name="viewport" content="width=device-width, initial-scale=1">  
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet">  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>  
</head>  
<body>
```

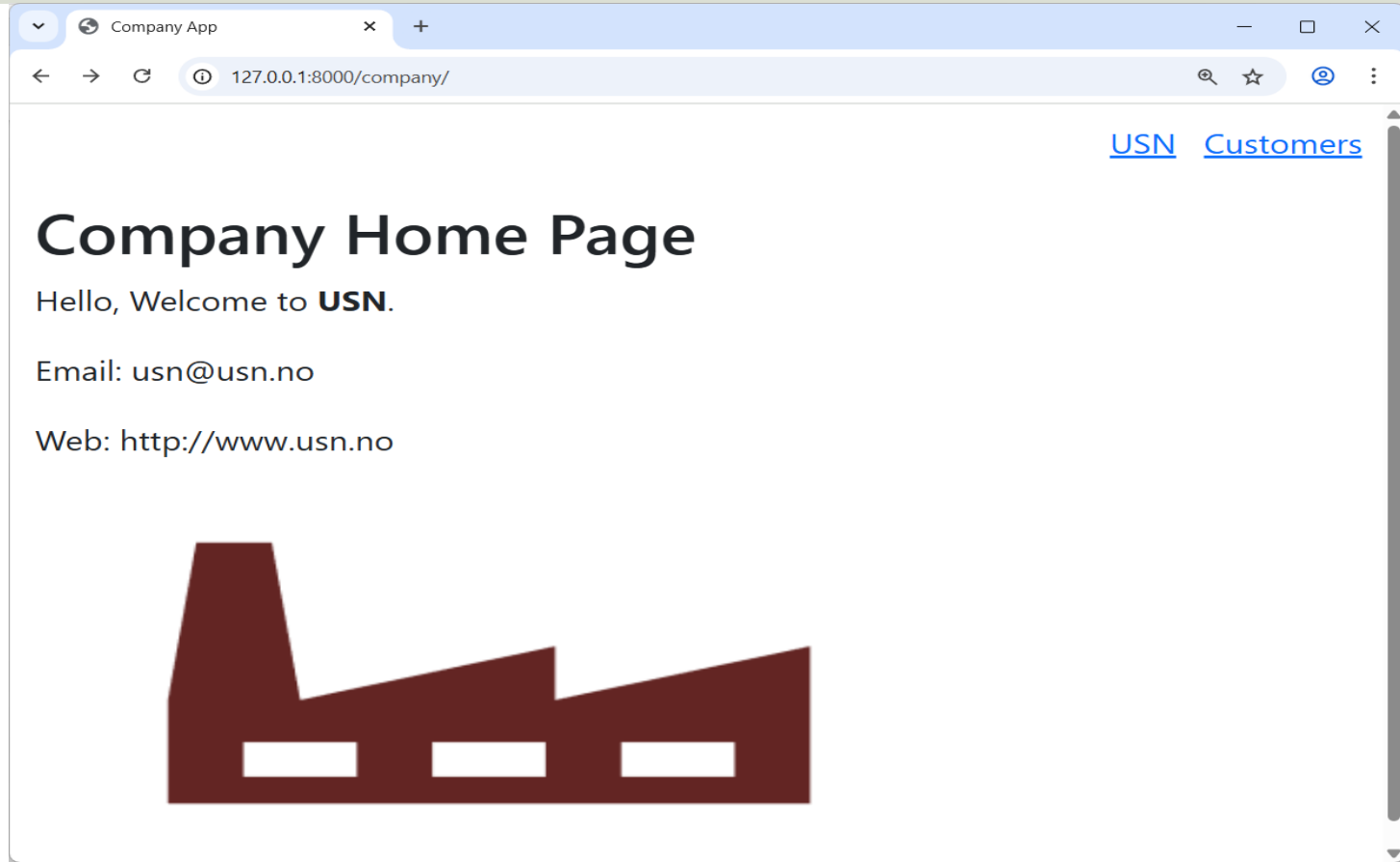
```
    {% include "header.html" with company=company.companyname %}
```

```
    <div class="container-fluid pt-1">  
        <h1>Company Home Page</h1>  
        <p>Hello, Welcome to <b>{{ company.companyname }}</b>.</p>  
        <p>Email: {{ company.email }}</p>  
        <p>Web: {{ company.website }}</p>
```

```
        <p></p>
```

```
    </div>  
</body>  
</html>
```

# “company.html” Template



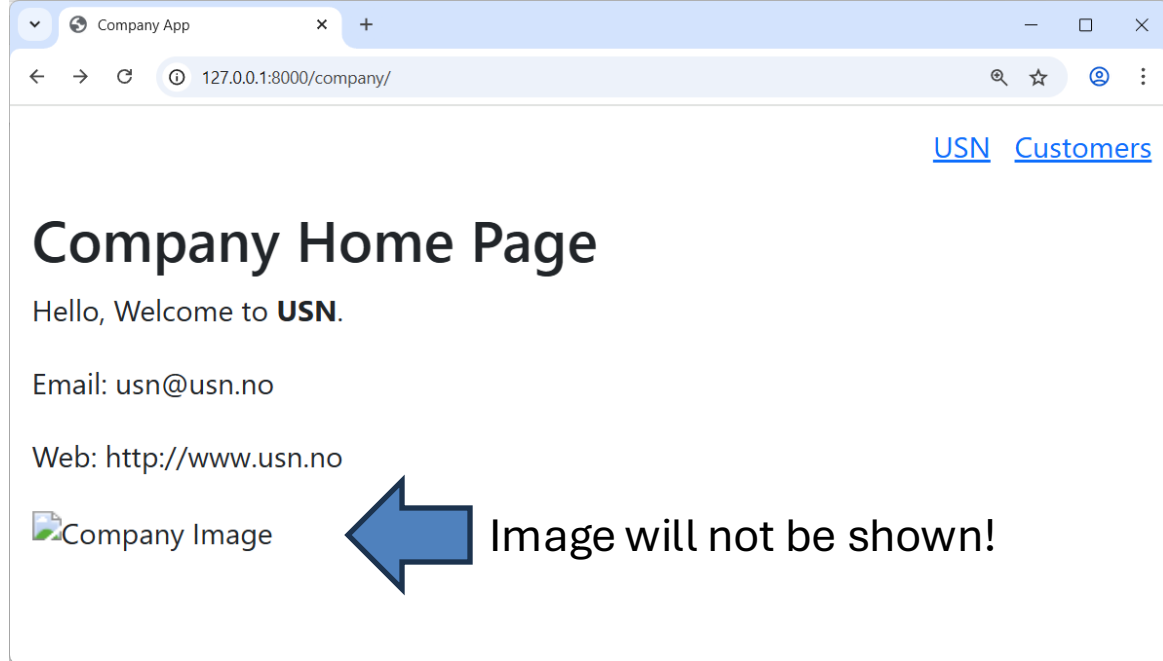
# DEBUG = False

settings.py X

djangoproject1 > settings.py > ...

```
24
25 # SECURITY WARNING: don't run with debug turned on in production!
26 DEBUG = False
27
28 ALLOWED_HOSTS = ['*']
29
```

Note! When DEBUG=False, you also need to specify “ALLOWED\_HOSTS”. Using ‘\*’ means “anything” and should not be used in production.



Company App

127.0.0.1:8000/company/


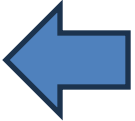
[USN](#) [Customers](#)

## Company Home Page

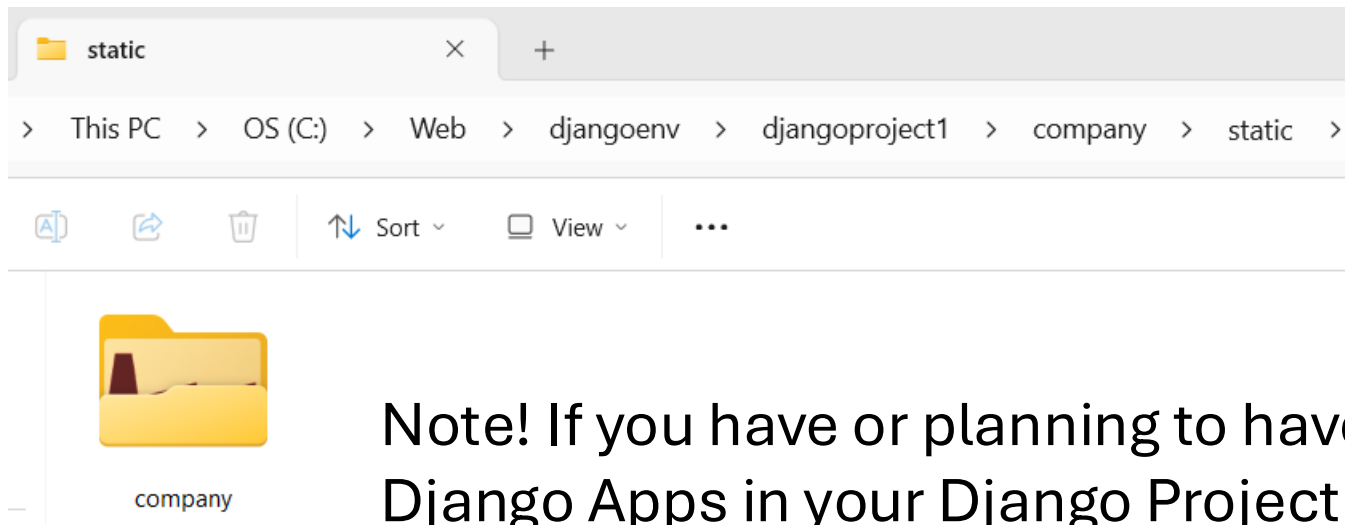
Hello, Welcome to **USN**.

Email: usn@usn.no

Web: http://www.usn.no

 Company Image  Image will not be shown!

# Multiple Apps?



Note! If you have or planning to have multiple Django Apps in your Django Project, you should create a subfolder in the static folder that has the same name as your App, in my case “company”.

# “STATIC\_ROOT”

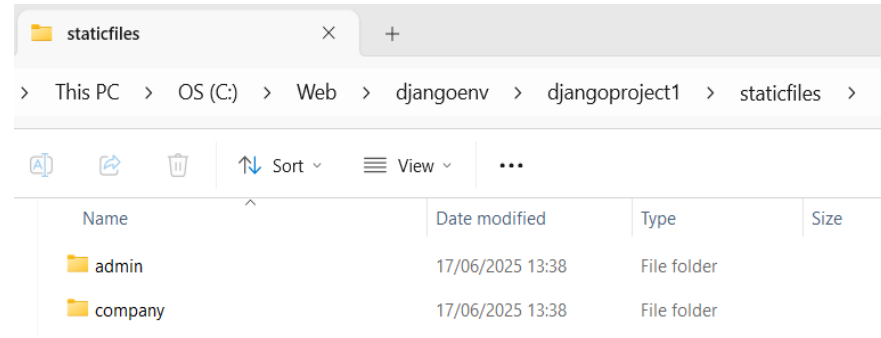
“STATIC\_ROOT” is primarily used for deployment in production

```
settings.py x
djangoenv > settings.py > ...
114
115 # Static files (CSS, JavaScript, Images)
116 # https://docs.djangoproject.com/en/5.2/howto/static-files/
117
118 STATIC_URL = 'static/'
119
120 STATIC_ROOT = BASE_DIR / 'staticfiles'
...
```

Then we run the following command in the Terminal:

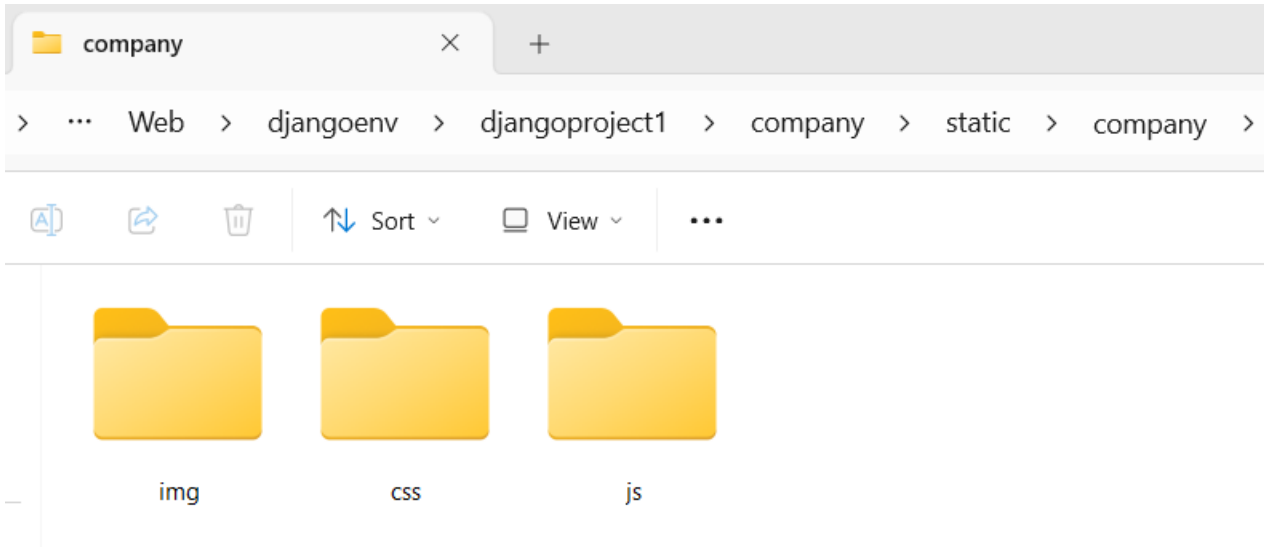
```
..>python manage.py collectstatic
```

The command “python manage.py collectstatic” gathers static files from your apps and other locations into the “STATIC\_ROOT” directory for deployment.



# Different Static Files

We typically will include different static files like images, CSS files and JavaScript files, so it is a good idea to also create subfolders for those different categories



```
<p></p>
```

<https://www.halvorsen.blog>

django

# CSS

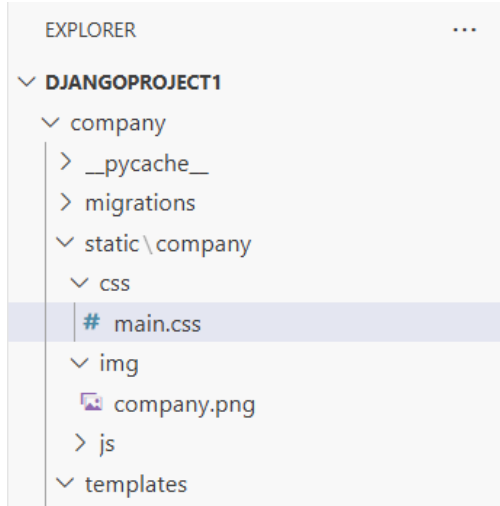


[Table of Contents](#)

Hans-Petter Halvorsen

# Create CSS File

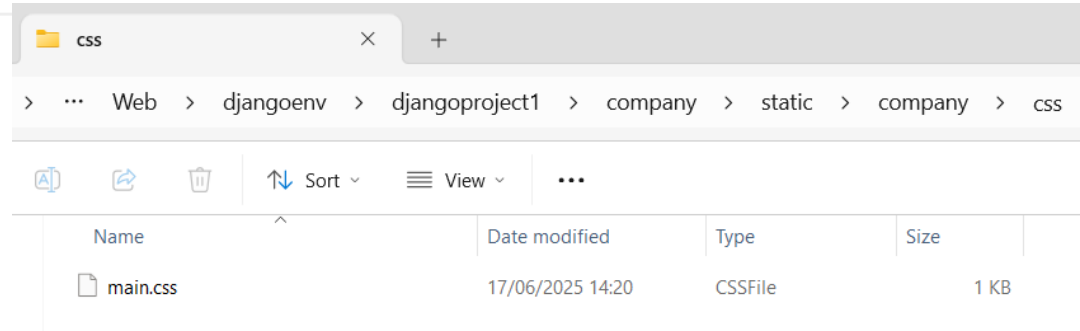
We create a CSS file that we want to use in all our templates:



```
# main.css

company > static > company > css > # main.css > h1

1 body {
2     background-color: beige;
3 }
4
5 h1 {
6     color: burlywood;
7 }
```



# “company.html”

```
{% load static %}
```

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title>Company App</title>
```

```
    <meta name="viewport" content="width=device-width, initial-scale=1">
```

```
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet">
```

```
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>
```

```
    <link rel="stylesheet" href="{% static 'company/css/main.css' %}">
```

```
  </head>
```

```
<body>
```

```
  {% include "header.html" with company=company.companyname %}
```

```
  <div class="container-fluid pt-1">
```

```
    <h1>Company Home Page</h1>
```

```
    <p>Hello, Welcome to <b>{{ company.companyname }}</b>.</p>
```

```
    <p>Email: {{ company.email }}</p>
```

```
    <p>Web: {{ company.website }}</p>
```

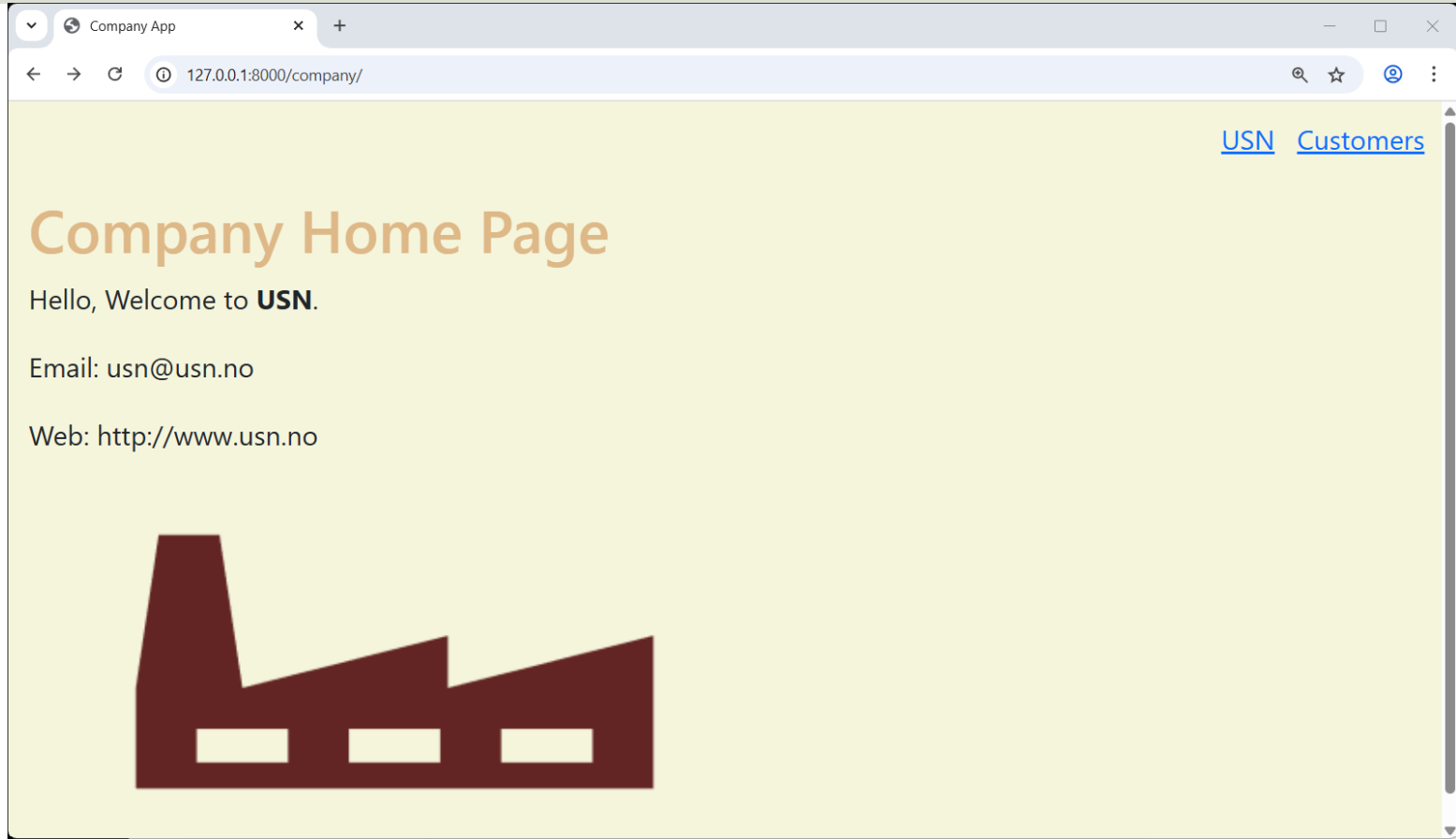
```
    <p></p>
```

```
  </div>
```

```
</body>
```

```
</html>
```

# Results



<https://www.halvorsen.blog>

django

# JavaScript



Hans-Petter Halvorsen

[Table of Contents](#)

# Create JavaScript File

The image shows a screenshot of the Visual Studio Code editor. On the left, the Explorer sidebar shows a project named 'DJANGOPROJECT1' with a file tree including 'company', 'static\company', and 'js'. The 'js' folder is selected, and 'scripts.js' is highlighted. The main editor window shows the content of 'scripts.js' with the following code:

```
company > static > company > js > JS scripts.js > welcome
1 function welcome() {
2     alert("Hello, how are today? Welcome!");
3 }
```

Below the editor, a file explorer window shows the 'js' folder. The breadcrumb path is 'Web > djangoenv > djangoenv > djangoenv > company > static > company > js'. The file list below shows:

Name	Date modified	Type	Size
scripts.js	17/06/2025 14:49	JSFile	1 KB

# “company.html”

```
{% load static %}
<!DOCTYPE html>
<html>
  <head>
    <title>Company App</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet">
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>
    <link rel="stylesheet" href="{% static 'company/css/main.css' %}">
    <script src="{% static 'company/js/scripts.js' %}"></script>
  </head>
  <body>
    <script>welcome()</script>
    {% include "header.html" with company=company.companyname %}
    <div class="container-fluid pt-1">
      <h1>Company Home Page</h1>
      <p>Hello, Welcome to <b>{{ company.companyname }}</b>.</p>
      <p>Email: {{ company.email }}</p>
      <p>Web: {{ company.website }}</p>

      <p></p>
    </div>
  </body>
</html>
```

# Hans-Petter Halvorsen

University of South-Eastern Norway

[www.usn.no](http://www.usn.no)

E-mail: [hans.p.halvorsen@usn.no](mailto:hans.p.halvorsen@usn.no)

Web: <https://www.halvorsen.blog>

