

# LOGGING IN LOGOUT

Now that we have created a sign up page we need to create a way for users to log in and log out of their account. The first part of this tutorial will show you how to create a log in page, the next will show logout.

The Django logo, featuring the word "django" in a bold, lowercase, sans-serif font. The background is split diagonally from the top-left to the bottom-right. The upper-left triangle is dark blue, and the lower-right triangle is black. The word "django" is positioned in the white triangular area in the center-right of the image.

**django**

# LOGGING IN

Django comes with a few build in applications that automatically generate specific user authentication forms like a log in form. To use these we need to add the following to our urls.py file from within the mysite directory.

*# urls.py file*

```
from django.contrib import admin
from django.urls import include, path
from register import views as v

urlpatterns = [
    path('admin/', admin.site.urls),
    path("register/", v.register, name="register"),
    path('', include('main.urls')),
    path('', include("django.contrib.auth.urls")),
]
```

Add  
code

Now we have a valid path to a /login and /logout page.

---

# LOGGING IN

The next step is to create the views for each of these pages. We will start by creating a folder inside of our register apps template folder called "registration". IT IS VERY IMPORTANT THAT YOU NAME IT EXACTLY THAT. It can be placed inside any applications templates folder but it must be in one.

Inside this folder we will create a file called: "login.html". Again this must be the name.

Now we can code the login.html file to display whatever we'd like:

*# login.html file*

```
# login.html
{% extends "main/base.html" %}

{% block title %}
Login Here
{% endblock %}

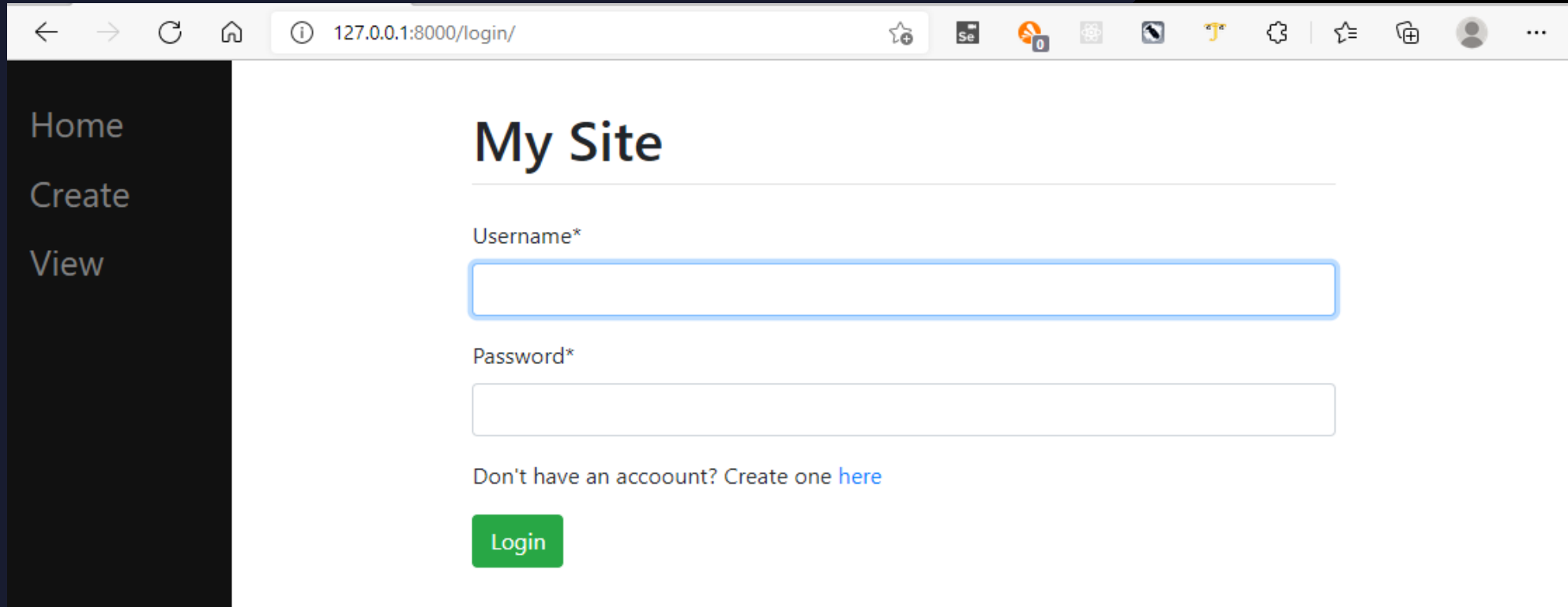
{% load crispy_forms_tags %}

{% block content %}
    <form method="post" class="from-group">
    {% csrf_token %}
    {{form|crispy}}
    <p>Don't have an accoount? Create one <a href="/register">here</a></p>
    <button type="submit" class="btn btn-success">Login</button>
    </form>
{% endblock %}
```

---

# LOGGING IN

And now we have a working login page!



The screenshot shows a web browser window with the address bar displaying `127.0.0.1:8000/login/`. The page has a dark sidebar on the left with the links "Home", "Create", and "View". The main content area is white and features the title "My Site" at the top. Below the title, there are two input fields: "Username\*" and "Password\*", each with a blue border. Below the password field, there is a link that says "Don't have an accooount? Create one [here](#)". At the bottom of the form, there is a green "Login" button.

Home  
Create  
View

## My Site

Username\*

Password\*

Don't have an accooount? Create one [here](#)

Login

# LOGIN/LOGOUT REDIRECT

As of now visiting /login and /logout work but after submitting the form we are not redirected to another page. To change where you go after logging in or logging out you can add the following to your settings.py file.

*# settings.py file*

```
LOGIN_REDIRECT_URL = "/"  
LOGOUT_REDIRECT_URL = "/"
```

Simply define the url that you'd like to be redirected to as a string.

---