

# Django Authentication Cheat Sheet: Registrierung, Login und Logout

---

[User Authentication in den Django Docs](#)

## Vorbereitung

App erstellen

```
python manage.py startapp accounts
```

---

## URLs einrichten

Hauptprojekt `urls.py`:

```
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('accounts/', include('accounts.urls')), # Accounts-App einbinden
]
```

Accounts-App `urls.py`:

Im Ordner `accounts` eine neue Datei `urls.py` erstellen:

```
from django.urls import path
from . import views

urlpatterns = [
    path('register/', views.register, name='register'),
    path('login/', views.login_view, name='login'),
    path('logout/', views.logout_view, name='logout'),
]
```

---

## Views erstellen

### Option A: Standard-Formular für Registrierung

`views.py` in der `accounts`-App:

```
from django.shortcuts import render, redirect
from django.contrib.auth import authenticate, login, logout
from django.contrib.auth.forms import UserCreationForm, AuthenticationForm

# Registrierung
def register(request):
    if request.method == 'POST':
        form = UserCreationForm(request.POST)
        if form.is_valid():
            form.save()
            return redirect('login') # Weiterleitung nach erfolgreicher Registrierung
    else:
        form = UserCreationForm()
    return render(request, 'accounts/register.html', {'form': form})
```

### Option B: Eigenes Formular für Registrierung

Ein eigenes Formular bietet mehr Flexibilität. Beispiel:

`forms.py` in der `accounts`-App:

```
from django import forms
from django.contrib.auth.models import User

class CustomUserCreationForm(forms.ModelForm):
    password1 = forms.CharField(label='Password', widget=forms.PasswordInput)
    password2 = forms.CharField(label='Confirm Password', widget=forms.PasswordInput)

    class Meta:
        model = User
        fields = ['username', 'email']

    def clean_password2(self):
        password1 = self.cleaned_data.get("password1")
        password2 = self.cleaned_data.get("password2")
        if password1 and password2 and password1 != password2:
            raise forms.ValidationError("Passwords don't match")
        return password2

    def save(self, commit=True):
        user = super().save(commit=False)
        user.set_password(self.cleaned_data["password1"])
        if commit:
            user.save()
        return user
```

### Registrierungs-View anpassen:

```
from .forms import CustomUserCreationForm

def register(request):
    if request.method == 'POST':
        form = CustomUserCreationForm(request.POST)
        if form.is_valid():
```

```

        form.save()
        return redirect('login')
    else:
        form = CustomUserCreationForm()
        return render(request, 'accounts/register.html', {'form': form})

```

## Views für Login und Logout erstellen

`views.py` in der `accounts` App

```

from django.shortcuts import render, redirect
from django.contrib.auth import authenticate, login, logout
from django.contrib.auth.forms import UserCreationForm, AuthenticationForm

# Registrierung
def register(request):
    ...

# Login
def login_view(request):
    if request.method == 'POST':
        form = AuthenticationForm(data=request.POST)
        if form.is_valid():
            user = form.get_user()
            login(request, user)
            return redirect('home') # Weiterleitung nach erfolgreichem Login
    else:
        form = AuthenticationForm()
    return render(request, 'accounts/login.html', {'form': form})

# Logout
def logout_view(request):
    if request.method == 'POST':
        logout(request)
        return redirect('login') # Weiterleitung nach Logout

```

## Templates erstellen

Ordnerstruktur:

Im Ordner `accounts` einen neuen Ordner `templates/accounts/` erstellen.

`register.html`:

```

<!DOCTYPE html>
<html>
<head>
    <title>Register</title>
</head>
<body>
    <h2>Register</h2>
    <form method="post">
        {% csrf_token %}
        {{ form.as_p }}
        <button type="submit">Register</button>
    </form>
    <a href="{% url 'accounts:login' %}">Already have an account? Login here</a>
</body>
</html>

```

login.html:

```
<!DOCTYPE html>
<html>
<head>
  <title>Login</title>
</head>
<body>
  <h2>Login</h2>
  <form method="post">
    {% csrf_token %}
    {{ form.as_p }}
    <button type="submit">Login</button>
  </form>
  <a href="{% url 'accounts:register' %}">Don't have an account? Register here</a>
</body>
</html>
```

---