

WEBTOKEN API

Projekt einrichten

Erstelle ein neues Django-Projekt namens tutorial und starte dann eine neue App namens quickstart.

Erstelle das Projektverzeichnis

```
mkdir api
```

```
cd api
```

Erstelle eine virtuelle Umgebung, um unsere Paketabhängigkeiten lokal zu isolieren

```
python3 -m venv .venv
```

```
source .venv/bin/activate
```

(# Unter Windows .venv/Scripts/activate verwenden)

Installiere Django und das Django REST Framework in der virtuellen Umgebung

```
py -m pip install django
```

```
pip install djangorestframework
```

```
pip install djangorestframework_simplejwt
```

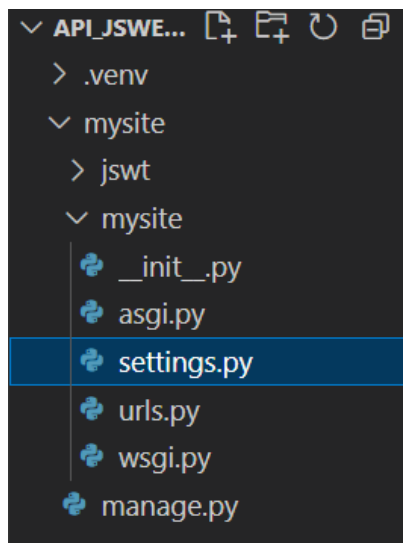
Richte ein neues Projekt mit einer einzelnen Anwendung ein

```
django-admin startproject mysite
```

```
cd mysite
```

```
django-admin startapp jwt
```

App mit dem Projekt verknüpfen



```
INSTALLED_APPS = [  
    'django.contrib.admin',  
    'django.contrib.auth',  
    'django.contrib.contenttypes',  
    'django.contrib.sessions',  
    'django.contrib.messages',  
    'django.contrib.staticfiles',  
    'rest_framework',  
    'rest_framework_simplejwt',  
    'jswt',  
]
```

JWS Settings konfigurieren

settings.py:

```
For the full list of settings and their values, see  
https://docs.djangoproject.com/en/5.0/ref/settings/
```

```
"""
```

```
from pathlib import Path
```

```
from datetime import timedelta
```

unter INSTALLED_APPS

```
REST_FRAMEWORK = {  
    'DEFAULT_PERMISSION_CLASSES': (  
        'rest_framework.permissions.IsAuthenticated',  
    ),  
    'DEFAULT_AUTHENTICATION_CLASSES': (  
        'rest_framework_simplejwt.authentication.JWTAuthentication'  
    ),  
}
```

```
SIMPLE_JWT = {  
    'ACCESS_TOKEN_LIFETIME': timedelta(minutes=60),  
    'SLIDING_TOKEN_REFRESH_LIFETIME': timedelta(days=1),  
    'SLIDING_TOKEN_LIFETIME': timedelta(days=30),  
    'SLIDING_TOKEN_REFRESH_LIFETIME_LATE_USER': timedelta(days=1),  
    'SLIDING_TOKEN_LIFETIME_LATE_USER': timedelta(days=30),  
}
```

Token Endpunkte

urls.py:

```
from rest_framework_simplejwt.views import TokenObtainPairView,
TokenRefreshView

from django.contrib import admin

from django.urls import path, include

urlpatterns = [

    path('admin/', admin.site.urls),

    path('api/token/', TokenObtainPairView.as_view(),
name='token_obtain_pair'),

    path('api/token/refresh/', TokenRefreshView.as_view(),
name='token_refresh'),

    path('', include('jswt.urls')), # include your app urls.py here

]
```

Api-Views

jswt/views.py:

```
from rest_framework.views import APIView

from rest_framework.response import Response

from rest_framework.permissions import IsAuthenticated

from rest_framework_simplejwt.authentication import JWTAuthentication
```

```
class Home(APIView):  
    authentication_classes = [JWTAuthentication]  
    permission_classes = [IsAuthenticated]  
  
    def get(self, request):  
        content = {'message': 'Hello, World!'}  
        return Response(content)
```

URL-Konfiguration

Erstellen Sie zunächst die Datei `jwt/urls.py` in Ihrer Anwendung.

jwt/urls.py:

```
from django.urls import path  
from .views import Home  
  
urlpatterns = [  
    path('', Home.as_view()),  
]
```

Migrations ausführen

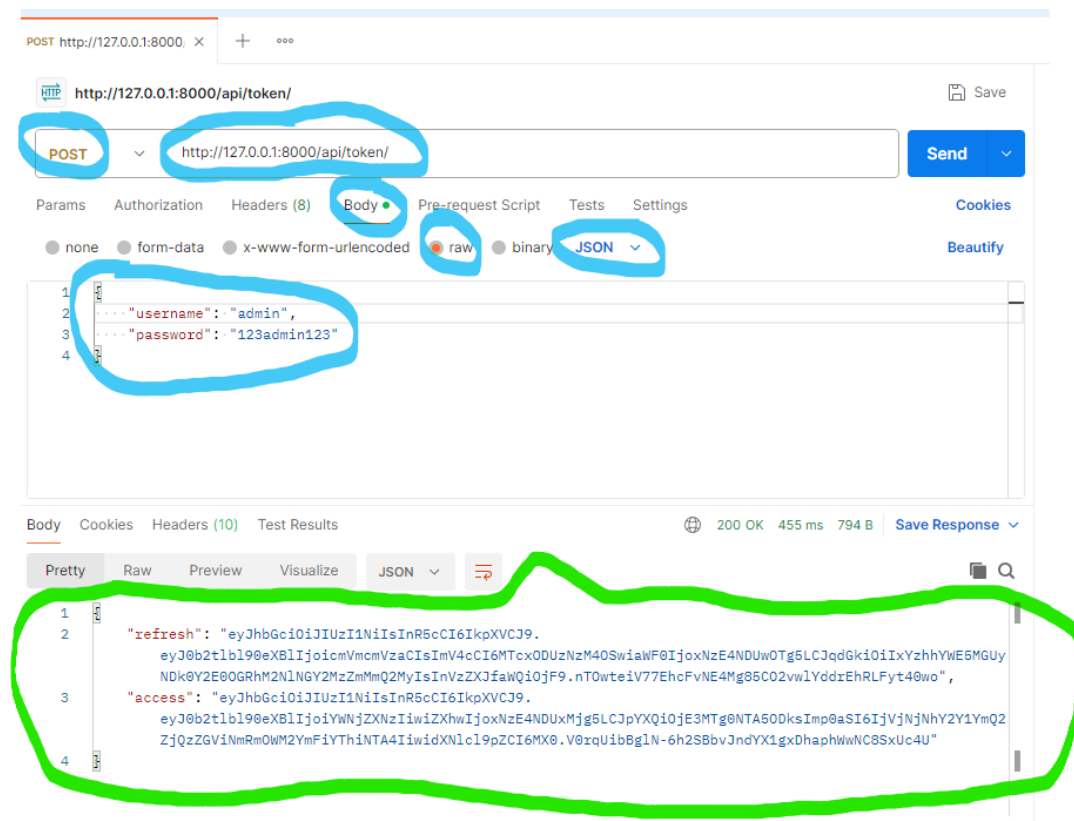
```
python3 manage.py migrate
```

Testen

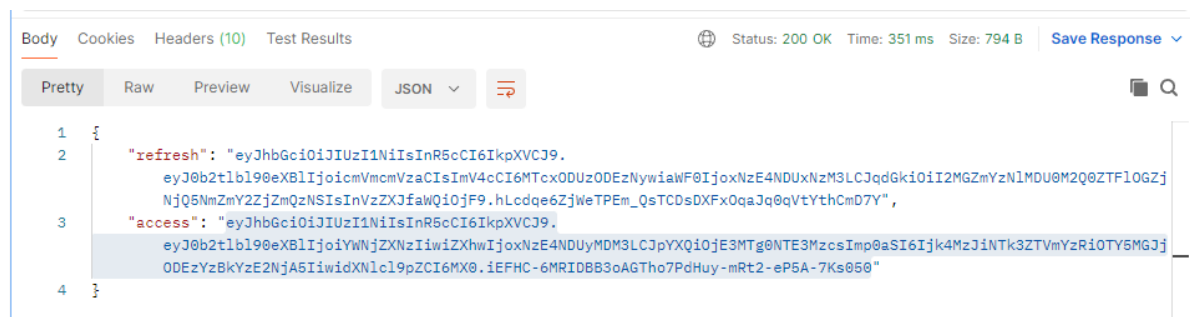
```
python3 manage.py createsuperuser # Username: admin | Email adress: admin@web.de | PW:  
123admin123
```

```
python3 manage.py runserver
```

Postman:



Den access-Token kopiëren



Verwenden Sie dann dieses Zugriffstoken, um Anfragen an authentifizierte Endpunkte zu stellen, wie z. B. (<http://127.0.0.1:8000/>)Typ muss als Bearer Token geändert werden.

HTTP

http://127.0.0.1:8000/

Save

GET

http://127.0.0.1:8000/

Send

Params

Authorization

Headers (7)

Body

Pre-request Script

Tests

Settings

Cookies

Type

Bearer To...

Token

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ...

The authorization header will be automatically generated when you send the request.

[Learn more about authorization](#)

Body

Cookies

Headers (10)

Test Results

Status: 200 OK

Time: 22 ms

Size: 364 B

Save Response

Pretty

Raw

Preview

Visualize

JSON

1

2

3

4

body

"message": "Hello, World!",