Task Management System API Documentation

## Overview

This API provides a complete task management solution with user authentication and full CRUD (Create, Read, Update, Delete) functionality for tasks.  
It is built using:  
- Flask for the web framework  
- SQLAlchemy for database interactions  
- JWT (JSON Web Tokens) for user authentication

# Base URL

http://localhost:5000

# Getting Started

## Requirements

- Python 3.6 or higher  
- Required Python packages:

## Installation

pip install -r requirements.txt

## Running the Application

1. Save the provided code into a file named app.py.  
2. Run the application:  
 python app.py  
3. The server will start at: http://localhost:5000

# Authentication

All task-related endpoints require JWT authentication.  
First, register a user, then log in to obtain an access token.

# API Endpoints

## 1. Authentication

### Register User

Endpoint: POST /auth/register  
Headers: Content-Type: application/json  
Request Body:  
{  
 "username": "testuser",  
 "email": "test@example.com",  
 "password": "testpass"  
}  
  
Success Response (201 Created):  
{"message": "User created successfully"}  
  
Error Responses:  
- 400 Bad Request: Missing required fields  
- 400 Bad Request: Username already exists  
- 400 Bad Request: Email already exists

### Login

Endpoint: POST /auth/login  
Headers: Content-Type: application/json  
Request Body:  
{  
 "username": "testuser",  
 "password": "testpass"  
}  
  
Success Response (200 OK):  
{"access\_token": "<jwt\_token>"}  
  
Error Responses:  
- 400 Bad Request: Missing username or password  
- 401 Unauthorized: Invalid username or password

## 2. Task Management

Note: All task endpoints require the following header:  
Authorization: Bearer <access\_token>

### Get All Tasks (Paginated)

Endpoint: GET /tasks?page=1&per\_page=10&completed=false  
  
Query Parameters:  
- page (default: 1)  
- per\_page (default: 10)  
- completed (optional)  
  
Success Response (200 OK):  
[...] (Sample JSON shown earlier)

### Get a Single Task

Endpoint: GET /tasks/<task\_id>  
  
Success Response (200 OK):  
[...] (Sample JSON shown earlier)  
  
Error Response (404 Not Found):  
{"message": "Task not found"}

### Create a New Task

Endpoint: POST /tasks  
Request Body:  
{  
 "title": "My first task",  
 "description": "This is important"  
}  
  
Success Response (201 Created):  
[...] (Sample JSON shown earlier)  
  
Error Response (400 Bad Request):  
{"message": "Title is required"}

### Update a Task

Endpoint: PUT /tasks/<task\_id>  
Request Body:  
{  
 "title": "Updated task title",  
 "description": "Updated description",  
 "completed": true  
}  
  
Success Response (200 OK):  
[...] (Sample JSON shown earlier)  
  
Error Response (404 Not Found):  
{"message": "Task not found"}

### Delete a Task

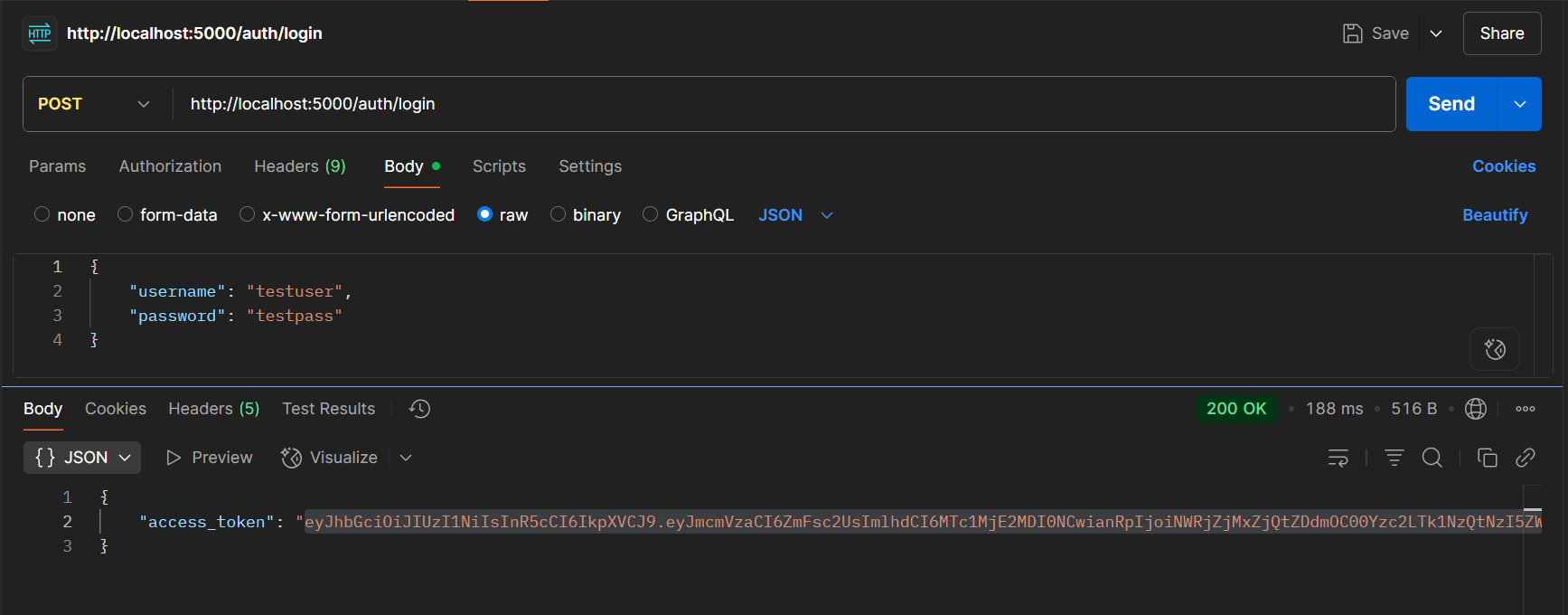
Endpoint: DELETE /tasks/<task\_id>  
  
Success Response (200 OK):  
{"message": "Task deleted successfully"}  
  
Error Response (404 Not Found):  
{"message": "Task not found"}

Screenshot:

Register a user  
A screenshot of a computer

AI-generated content may be incorrect.

Login a created user



Creating a task

A screenshot of a computer

AI-generated content may be incorrect.

Check the task with task

A screenshot of a computer

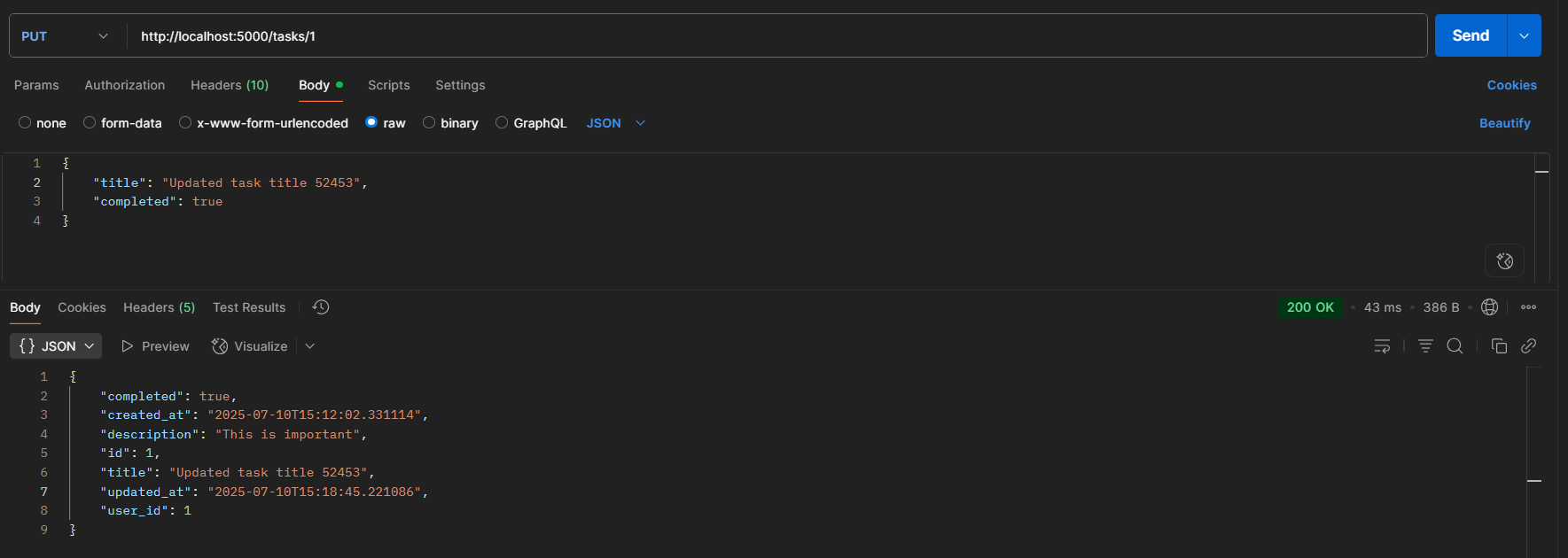
AI-generated content may be incorrect.

Get task individually

A screenshot of a computer

AI-generated content may be incorrect.

Update the task status and content



Delete the task with task id

A screenshot of a computer

AI-generated content may be incorrect.