**Django REST Framework Artist API Documentation**

**Table of Contents**

1. [Overview](https://chat.openai.com/c/881691c6-7473-4f75-8370-5756e6e99b89#overview)
2. [Setup and Installation](https://chat.openai.com/c/881691c6-7473-4f75-8370-5756e6e99b89#setup-and-installation)
3. [API Endpoints](https://chat.openai.com/c/881691c6-7473-4f75-8370-5756e6e99b89#api-endpoints)
   * [Authentication](https://chat.openai.com/c/881691c6-7473-4f75-8370-5756e6e99b89#authentication)
   * [Artist Operations](https://chat.openai.com/c/881691c6-7473-4f75-8370-5756e6e99b89#artist-operations)
   * [Work Operations](https://chat.openai.com/c/881691c6-7473-4f75-8370-5756e6e99b89#work-operations)
4. [Sample Usage](https://chat.openai.com/c/881691c6-7473-4f75-8370-5756e6e99b89#sample-usage)
5. [Testing](https://chat.openai.com/c/881691c6-7473-4f75-8370-5756e6e99b89#testing)
6. [Troubleshooting](https://chat.openai.com/c/881691c6-7473-4f75-8370-5756e6e99b89#troubleshooting)

**Overview**

This documentation provides an overview of the Django REST Framework Artist API, outlining its features, endpoints, and usage guidelines.

### **Features**

• Token-based authentication for CRUD operations.

• User registration, login, and token retrieval.

• Artist and Work models with associated endpoints.

• Filtering and searching capabilities.

## Getting Started

### **Prerequisites**

Ensure you have the following installed:

* Python (version X.X)
* Django (version X.X)
* Django REST Framework (version X.X)

## Setup and Installation

1. Clone the repository:

git clone <https://github.com/your-username/artist-api.git>

1. Navigate to the project directory:

cd artist-api

1. Install dependencies:

Which are mentioned in prerequisites.

1. Apply migrations:

python manage.py migrate.

1. Create a superuser for accessing the Django admin:

python manage.py createsuperuser

1. Run the development server:

python manage.py runserver

The API is now accessible at <http://127.0.0.1:8000/>.

### **Django Admin**

Access the Django admin interface at <http://127.0.0.1:8000/admin/> to manage users, artists, and works. Use the superuser credentials created during the setup.

## API Endpoints

### Authentication

* **POST /api/register/:** Register a new user and retrieve an authentication token.
  + Request Body Example:

**{**

**"username": "your\_username",**

**"password": "your\_password",**

**"email": "your\_email@example.com"**

**}**

Response:

**{**

**"token": "your\_auth\_token"**

**}**

**POST /api/login/:** Log in and retrieve an authentication token.

### **Artist Operations**

* **GET /api/artists/:** Retrieve a list of all artists.
* **POST /api/artists/:** Create a new artist.

**Request body:**

**{**

**"name": "artist\_name"**

**}**

* **GET /api/artists/{id}/:** Retrieve details of a specific artist.
* **PUT /api/artists/{id}/:** Update details of a specific artist.

**Request body:**

**{**

**"name": "updated\_artist\_name"**

**}**

**DELETE /api/artists/{id}/:** Delete a specific artist.

### **Work Operations**

* **GET /api/works/:** Retrieve a list of all works.
* **POST /api/works/:** Create a new work.
  + Request Body Example

**{**

**"link": "https://example.com",**

**"work\_type": "YT"**

**}**

* **GET /api/works/{id}/: Retrieve details of a specific work.**
* **PUT /api/works/{id}/: Update details of a specific work.**
* **DELETE /api/works/{id}/:** Delete a specific work.

### Filtering and Searching

* **GET /api/works/?work\_type={type}/:** Retrieve a list of works filtered by work type (e.g., "YT" or "Instagram").
* **GET /api/works/?artist={artist\_name}/:** Retrieve a list of works associated with a specific artist.

## Sample Usage

1. **User Registration:**
   * Make a **POST** request to **/api/register/** with user details.
   * Retrieve the authentication token from the response.
2. **User Login:**
   * Make a **POST** request to **/api/login/** with user credentials.
   * Retrieve the authentication token from the response.
3. **Create Artist:**
   * Make a **POST** request to **/api/artists/** with artist details and the authentication token in the header.
4. **Retrieve Works:**
   * Make a **GET** request to **/api/works/** to get a list of all works.
5. **Filter Works by Type:**
   * Make a **GET** request to **/api/works/?work\_type=YT/** to get works filtered by type.
6. **Search Works by Artist:**
   * Make a **GET** request to **/api/works/?artist=artist\_name/** to get works associated with a specific artist.
7. **Update Artist:**
   * Make a **PUT** request to **/api/artists/{id}/** with updated artist details.
8. **Delete Artist:**
   * Make a **DELETE** request to **/api/artists/{id}/** to delete a specific artist.

**Troubleshooting**

* If you encounter issues, refer to the error messages in the API responses.
* Ensure that the request headers, especially the **Authorization** header, are correctly set.