DRF Task 1

1) What is Django REST Framework?

Django REST Framework (DRF) is a widely used, full-featured API framework designed for building RESTful APIs with Django. At its core, DRF integrates with Django's core features -- models, views, and URLs -- making it simple and seamless to create a RESTful API.

DRF is composed of the following components:

Serializers are used to convert Django Query Sets and model instances to (serialization) and from (deserialization) JSON (and a number of other data rendering formats like XML and YAML).

Views (along with View Sets), which are similar to traditional Django views, handle RESTful HTTP requests and responses. The view itself uses serializers to validate incoming payloads and contains the necessary logic to return the response. Viewsets are coupled with **routers**, which map the views back to the exposed URLs.

Authentication: DRF includes built-in support for various authentication methods, including session-based authentication, token-based authentication, OAuth, and more. This makes it easy to secure your APIs and control access to resources.

Permissions: DRF provides a flexible permissions system that allows you to restrict access to views based on user permissions or other criteria. You can easily define custom permission classes to enforce fine-grained access control.

Routers: DRF includes a simple router system that automatically generates URL patterns for your API views and viewsets, reducing the amount of boilerplate code you need to write.

Browsable API: DRF includes a browsable API feature that generates a human-readable web interface for your API, allowing developers to interact with the API directly from their browser. This can be incredibly useful for testing and debugging APIs.

2)what is restful api ?

RESTful API is an interface that two computer systems use to exchange information securely over the internet. Most business applications have to communicate with other internal and third-party applications to perform various tasks. For example, to generate monthly payslips, your internal accounts system has to share data with your customer's banking system to automate invoicing and communicate with an internal timesheet application. RESTful APIs support this information exchange because they follow secure, reliable, and efficient software communication standards.

REST APIs are the most commonly used APIs due to the several benefits they offer: Here is why developers prefer working with REST APIs:

1. **Simplicity and Ease of Use:**
2. **Scalability:**
3. **Flexibility:**
4. **Statelessness:**
5. **Interoperability:**
6. **Cacheability:**
7. **Uniform Interface:**
8. **Reduced Latency:**
9. **Ease of Integration:**
10. **Security:**