**Documentation on Django and REST api**

**1. Introduction to Django**

Django is a high-level Python web framework designed to encourage rapid development and clean design. It focuses on automating as much as possible while following the "Don't Repeat Yourself" (DRY) principle.

**Key Features of Django:**

* **MVC Framework:** Django follows the Model-View-Controller (MVC) architecture, but it's referred to as Model-View-Template (MVT) in Django terminology.
* **ORM (Object-Relational Mapping):** Django comes with its own ORM, making it easy to interact with databases without writing raw SQL.
* **Admin Interface:** Automatically generated admin panel based on models.
* **Security:** Provides built-in protections against common web attacks (SQL injection, cross-site scripting, etc.).

**2. REST API Overview**

A REST API (Representational State Transfer Application Programming Interface) allows communication between clients and servers in a stateless manner using HTTP. REST is widely used in web services due to its simplicity and scalability.

**Key Principles of REST:**

* **Stateless:** Each request from client to server must contain all the information the server needs to fulfill the request.
* **Uniform Interface:** Resources are accessed via a consistent interface, typically using URLs.
* **CRUD Operations:** REST APIs generally use HTTP methods to perform operations:
  + **GET** – Retrieve data
  + **POST** – Create data
  + **PUT** – Update data
  + **DELETE** – Delete data

**3. Django REST Framework**

Django REST Framework is a powerful toolkit for building web APIs in Django. It is widely used for creating RESTful APIs quickly and easily.

**Key Features:**

* **Serializers:** Convert Django model instances to JSON and vice versa.
* **Viewsets:** Simplify the creation of views by combining logic for multiple HTTP methods into a single class.
* **Authentication:** Supports multiple authentication methods like token-based authentication, session authentication, etc.
* **Permissions:** Provides customizable permission classes for restricting API access.
* **Browsable API:** DRF automatically generates a web-browsable API for easy testing and debugging.