1.what is view and why it is use in Django?

In the model view controller (MVC) architecture, the view component deals with how data is presented to users for consumption and viewing. In the Django framework, views are Python functions or classes that receive a web request and return a web response. The response can be a simple HTTP response, an HTML template response, or an HTTP redirect response that redirects a user to another page. Views hold the logic that is required to return information as a response in whatever form to the user. As a matter of best practice, the logic that deals with views is held in the `views.py` file in a Django app.

This guide will explore how to use views in Django as a data display and viewing tool. It will also explore the two major types of views: class-based and function-based views. It assumes that you have at least beginner level knowledge of Django and a general understanding of the Django MVC. An introductory guide to Django can be found here.

2.what is rest api

The term REST stands for **RE**presentational **S**tate **T**ransfer. It is an architectural style that defines a set of rules in order to create Web Services. In a client-server communication, REST suggests to create an object of the data requested by the client and send the values of the object in response to the user. For example, if the user is requesting for a movie in Bangalore at a certain place and time, then you can create an object on the server-side.

So, over here, you have an object and you are sending the state of an object. This is why REST is known as Representational State Transfer.

The architectural style of REST helps in leveraging the lesser use of bandwidth to make an application more suitable for the internet. It is often regarded as the "*language of the internet*" and is completely based on the resources.

Now that you know what it is, let us move on and understand the need for REST API.