**REST API:**

API or Application Programming Interfaces are mechanisms that enable two software components to communicate with each other using a set of definitions and protocols.

A REST API is used for systems to expose useful functions and data. REST stands for representational state transfer, which can be made up of one or more resources that can be accessed at a given URL and returned in various formats, like JSON, images, HTML, and more.

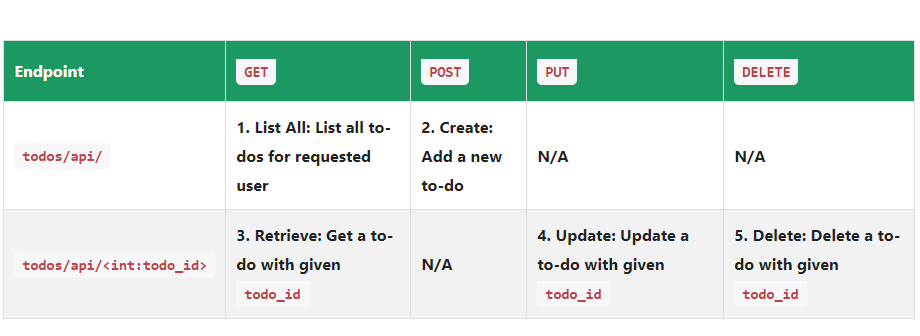
Building a RESTful API is in most cases complicated, but with the help of Django and Django REST framework, this complexity is handled extremely well.

**Django:**

Django is an open-source and python-based web framework that uses the Model-View-Template (MVT) architectural pattern.

Django has a powerful and flexible toolkit called Django REST framework (DRF) and it’s a useful tool for building Web APIs. It adopts implementations like class-based views, forms, model validators, QuerySet, and more. Its main functionality is [making serialization much easier](https://blog.logrocket.com/understanding-typescript-object-serialization/).

In a RESTful API, endpoints define the structure and usage with the GET, POST, PUT, and DELETE HTTP methods, which must be organized logically.



Models in Django are classes that contain the fields we would want our database table to contain. After creating the model, you will need to migrate it to the database using the command lines:

**python manage.py makemigrations**

**python manage.py migrate**

**Serialization:**

Serialization is such an important property built in the Django REST framework as it ensures the conversion of the object into a format that can be stored or transmitted. Therefore, after the serialized data has been transmitted or stored, you’ll be able to reconstruct the object and thus obtain the same structure as the object was originally in.

Python uses many available different formats for serialization, one common example that works across many languages is the JSON file format which is human-readable and allows us to store the dictionary and recreate it with the same structure. To convert the Model object to an API-appropriate format like JSON, the Django REST framework uses the ModelSerializer class to convert any model to serialized JSON objects. This makes serialization an important asset to Django REST Framework for web development.

