Name: Kashyap Sorathiya

Batch: 06_Sept_Python

Assignment Module 6 (REST Framework)

- Create Django REST API using below instruction.
- Create Book Model
    - Title
    - Author
    - Isbn
    - Publisher
    - Using Serialize, display all data in REST API
    - Also Using REST API, perfume CRUD Operation

Ans. Add 'rest_framework' to the INSTALLED_APPS list in your project's settings.py file.

Define the Book model in your app's models.py file:

```
from django.db import models

class Book(models.Model):
    title = models.CharField(max_length=255)
    author = models.CharField(max_length=255)
    isbn = models.CharField(max_length=13)
    publisher = models.CharField(max_length=255)
```

Create a serializer class in your app's serializers.py file

```
from rest_framework import serializers
from .models import Book

class BookSerializer(serializers.ModelSerializer):
    class Meta:
        model = Book
        fields = '__all__'
```

Create views for the CRUD operations using class-based views and the ModelViewSet class provided by the Django REST framework. In views.py,

```
from rest_framework import viewsets
from .models import Book
from .serializers import BookSerializer

class BookViewSet(viewsets.ModelViewSet):
    queryset = Book.objects.all()
    serializer_class = BookSerializer
```

Define the URL routes for the API urls.py file:

```
from django.urls import path, include
from rest_framework import routers
from .views import BookViewSet

router = routers.DefaultRouter()
router.register(r'books', BookViewSet)

urlpatterns = [
    path('', include(router.urls)),
]
```

Finally, run the Django server and the API will be available at http://localhost:8000/books/, we can use this API to create, read, update, and delete Book objects using standard HTTP methods (POST, GET, PUT, DELETE).