Generics:-

→ Django rest framework provide 'generics' module to providing the multiple database operations

→ This module is providing several individual classes and also combination classes to performing the required operations

Example :- individual classes

List APIView : it is used to get all the data from db

Create APIView : it is used to post the data

Retrive APIView : it is used to get any specific data based on id

Destroy APIView : it is used to delete any specific data based on id

List Create APIView :

Update API View : it is used to modify any specific data based on id

Combination class :-

List Create APIView : it is used to get all data and also post the create new data in the database (it can perform non-id operations)

Retrive Update APIView : it is used to get the specific data based on id and update that id based data.

Retrive Destroy APIView : it is used to get the specific data based on id and delete that id based data

Retrive Update Destroy APIView :- it is used to get any specific data based on the id and we can update or Delete the specific id based record

NOTE :- Here, we no need of writing the handler and action methods directly, we required only queryset & serializer_class field only with required information

Eg:-
```
def get (self, request, pk)
    return self. update (request)
```
✗ not required in generics

```
queryset = modelname. objects. all
serializer.class = modelserializer
```
} we only required this in generics

Create a project to performing CURD operations by using Generics

→ copy & paste all the files from previous program except views

Open views.py

class import

from .models import Emp

from .serializers import EmpSerializer

from rest_framework import generics

class EmpListView (generics. ListCreateAPIView):

    queryset = Emp. objects. all ()

    serializer_class = EmpSerializer

class EmpDetailView (generics. RetriveUpdateDestroyAPIView):

    queryset = Emp. objects. all ()

    serializer_class = EmpSerializer