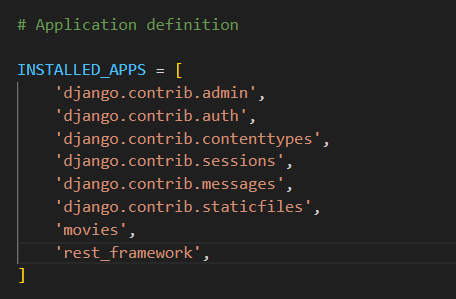
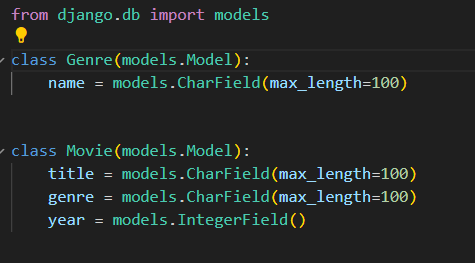
## Unidad 3. Desarrollo de un servidor Django para una aplicación de películas

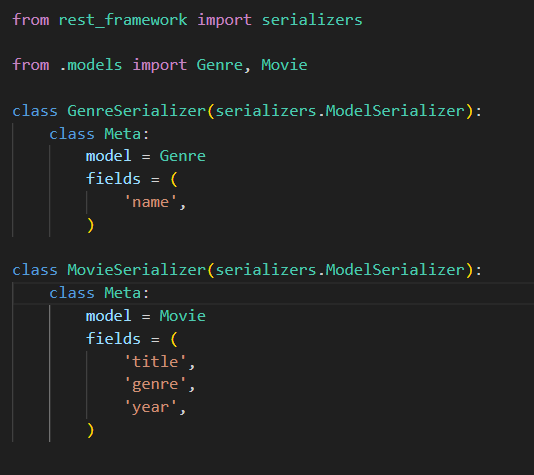
1. Creamos un proyecto nuevo “unidad3”, creamos una nueva aplicación “movies” dentro de un proyecto Django
2. Instalamos nuestra app y rest\_framework en unidad3/settings.py



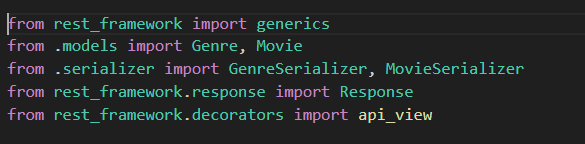
1. Generamos dos modelos: Genre y Movie en unidad3/movies/models.py



1. Creamos el archivo unidad3/movies/ serializer.py

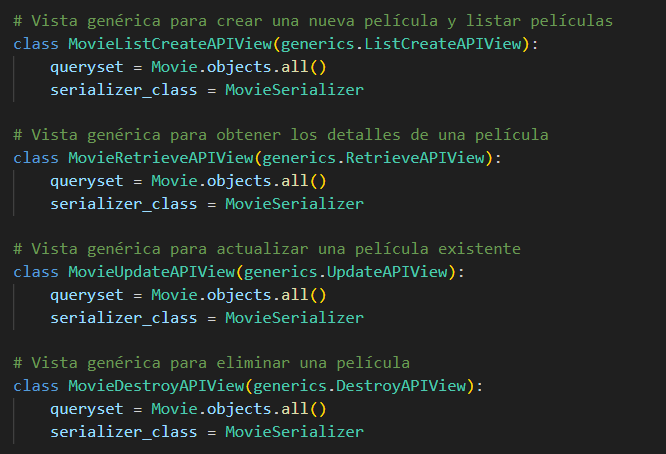


1. En unidad3/movies/views.py hacemos los importaciones necesarios

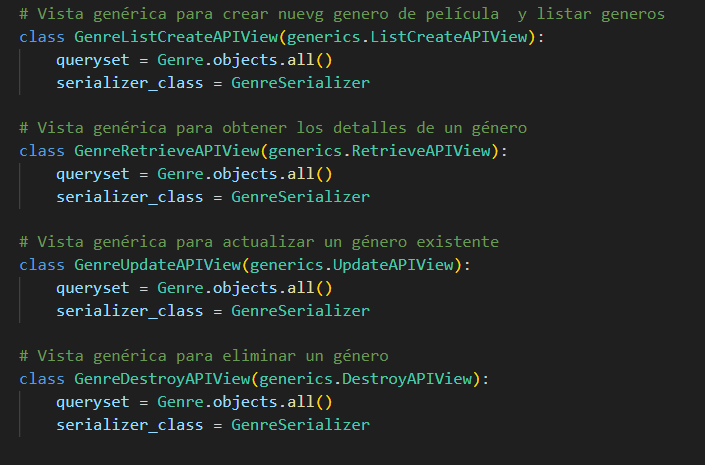


1. Creamos 4 vistas genéricas distintos para cada modelo

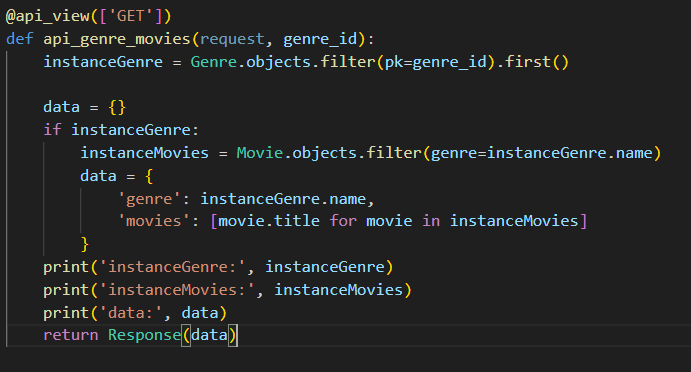
Para el modelo Movie:



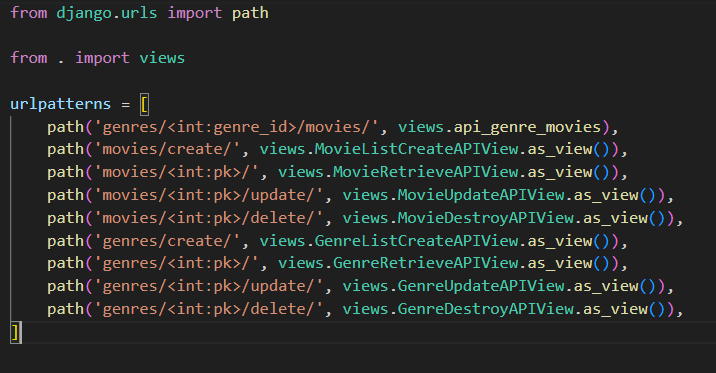
Para el modelo Genre:



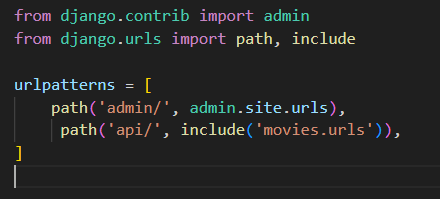
1. Creamos API personalizada que enlace modelos: una vista que enlace los modelos Genre y Movie y muestre las películas de un género específico



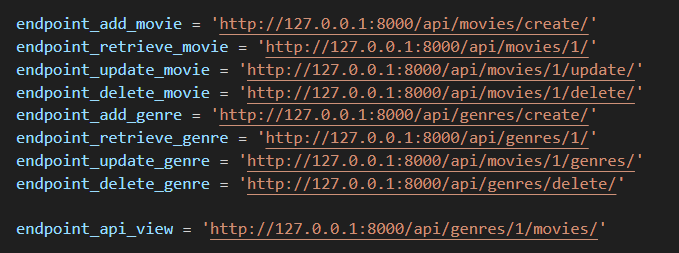
1. Configuramos las URL en el archivo unidad3/movies/urls.py



1. Configuramos las URL en el archivo unidad3/ /urls.py, incluyendo los urls de unidad3/movies/urls.py



1. Hacemos la migración
2. Creamos la carpeta unidad3/clients y el archivo unidad3/clients/test.py
3. Probamos las peticiones a las API de tu aplicación de películas:
   1. Creamos los endpoints

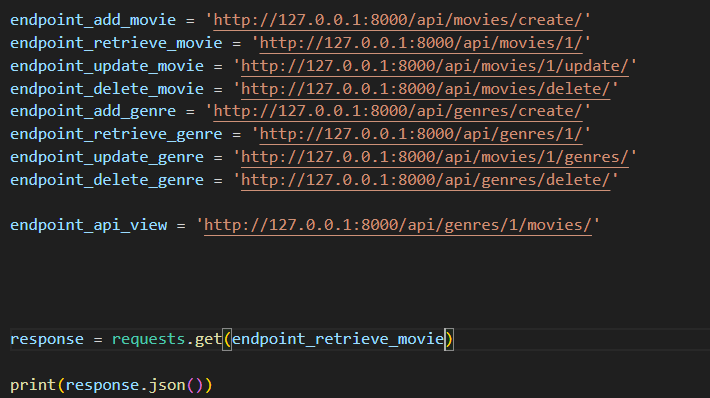


* 1. Creamos nuevas películas





* 1. Obtenemos los detalles de la película con id=1





* 1. Modificamos los datos de la película con id=1

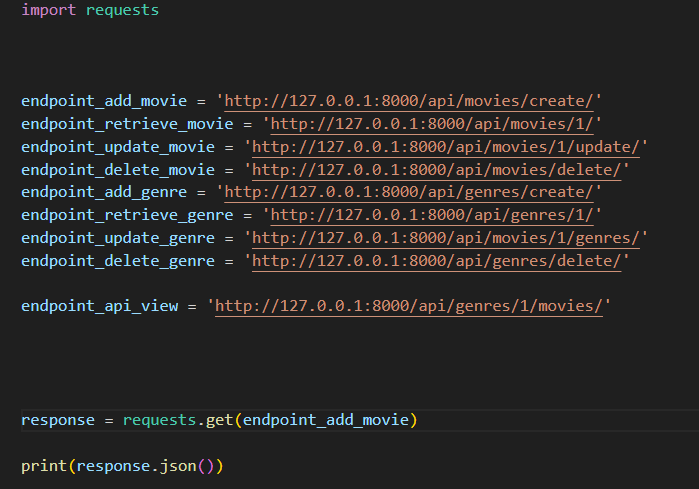


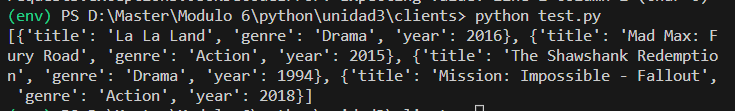


* 1. Borramos la película con id=1

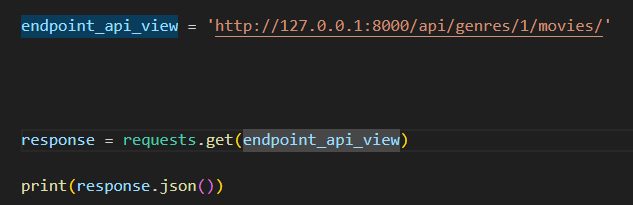


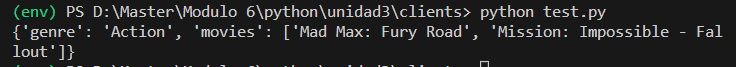
* 1. Obtenemos la lista de todas las peliculas





* 1. Obtenemos la lista de las peliculas de genero con id=1 (Action)





## Observaciones:

1. Se puede establecer relación uno a muchos entre Genre y Movie (utilizando el campo ForeignKey en el modelo Movie)
2. Se puede definir una lista de opciones predefinidas para el campo genre para limitar las opciones válidas para este campo
3. En lugar de tener vistas genéricas separadas para cada operación (crear, obtener detalles, actualizar, eliminar), se puede utilizar las vistas genéricas RetrieveUpdateDestroyAPIView y ListCreateAPIView juntas para combinar las operaciones correspondientes y reducir el código