**REST API (Django Rest Framework-DRF)**

**Session-17**

**Working with TokenAuthentication:**

* This authentication scheme uses a simple token-based HTTP Authentication scheme.
* Token authentication is appropriate for client-server setups, such as native desktop and mobile clients.
* To use the TokenAuthentication scheme you'll need to configure the authentication classes to include TokenAuthentication, and additionally include rest\_framework.authtoken in your INSTALLED\_APPS settings.py file:

INSTALLED\_APPS = [

...

'rest\_framework.authtoken'

]

* **Note**: Make sure to run manage.py migrate after changing your settings. The rest\_framework.authtoken app provides Django database migrations.
* If successfully authenticated, TokenAuthentication provides the following credentials.
* request.user will be a Django User instance.
* request.auth will be a rest\_framework.authtoken.models.Token instance.
* Unauthenticated responses that are denied permission will result in an HTTP 401 Unauthorized response .

**Options to generate token :**

1. Using Admin Interface
2. Using manage.py command in terminal
3. By exposing API endpoint
4. Using signals

**Settings.py:**

INSTALLED\_APPS = [  
 **'django.contrib.admin'**,  
 **'django.contrib.auth'**,  
 **'django.contrib.contenttypes'**,  
 **'django.contrib.sessions'**,  
 **'django.contrib.messages'**,  
 **'django.contrib.staticfiles'**,  
 **'myapp.apps.MyappConfig'**,  
 **'rest\_framework'**,  
 **'myapp8'**,

**'rest\_framework.authtoken'**,

]

**Models.py:**

**from** django.db **import** models  
  
*# Create your models here.***class** Manager(models.Model):  
 name=models.CharField(max\_length=20)  
 address=models.CharField(max\_length=20)  
 mail = models.CharField(max\_length=20)  
 age = models.IntegerField()

**admin.py:**

**from** django.contrib **import** admin  
**from** myapp8.models **import** Manager  
*# Register your models here.*@admin.register(Manager)  
**class** ManagerAdmin(admin.ModelAdmin):  
 list\_display = [**'id'**,**'name'**,**'address'**,**'mail'**,**'age'**]

**Now go to terminal and then type the following commands**

* Python manage.py makemigrations
* Python manage.py migrate
* Python manage.py createsuperuser
* Now run server: python manage.py runserver
* Now go to browser: <http://127.0.0.1:8000/admin>
* Insert few records

Create a new python file with the name serializers.py.

**serializer.py:**

**from** rest\_framework **import** serializers  
**from** myapp8.models **import** Manager  
  
**class** ManagerSerializer(serializers.ModelSerializer):  
 **class** Meta:  
 model=Manager  
 fields=[**'id'**,**'name'**,**'address'**,**'mail'**,**'age'**]

**views.py:**

**from** django.shortcuts **import** render  
**from** myapp8.models **import** Manager  
**from** myapp8.serializers **import** ManagerSerializer  
**from** rest\_framework **import** viewsets  
**from** rest\_framework.authentication **import** TokenAuthentication  
  
*# Create your views here.***class** ManagerModelViewSet(viewsets.ModelViewSet):  
 queryset = Manager.objects.all()  
 serializer\_class = ManagerSerializer

**urls.py**:

**from** django.contrib **import** admin  
**from** django.urls **import** path,include  
**from** myapp8 **import** views  
**from** rest\_framework.routers **import** DefaultRouter  
  
*#create a router object*router=DefaultRouter()  
  
*#register your viewset with router*router.register(**'ManagerViewset'**,views.ManagerModelViewSet,basename=**'manager'**)  
  
urlpatterns = [  
 path(**'admin/'**, admin.site.urls),  
 path(**''**,include(router.urls)),  
 path(**'authentication/'**,include(**'rest\_framework.urls'**,namespace=**'rest\_framework'**))  
  
]

**1. Generating a token using admin interface**

**Note: to generate token using admin interface, first run server**

<http://127.0.0.1:8000/admin>

* Go to admin panel, click on tokens, click on Add token, select user for creating a token and then click on save.

**1. Generating a token using manage.py command**

**Note: go to terminal and type the following command**

* **python manage.py drf\_create\_token mohan**

**Note**: if user has token then it will return that, if user don’t have any token then it will create a token.