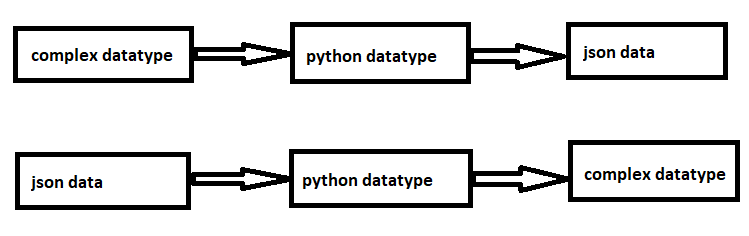
**REST API (Django Rest Framework-DRF)**

**Session-5**

**De serialization:**

* Serializers in Django REST Framework are **responsible for converting objects into data types** understandable by front-end frameworks.
* Serializers also provide deserialization, allowing parsed data to be converted back into complex types, after first validating the incoming data.

****

* The io module provides Python's main facilities for dealing with various types of I**/O.**
* Python IO module allows us to manage the file-related input and output operations.
* A stream is **basically a sequence of data**. Whatever data we use in our programming it flows through a stream.
* A stream can be thought of as a channel connecting a processor or logic unit (where data is processed according to the instructions) and input and output devices.
* BytesIO is used for **binary data.**
* BytesIO can be used in place of a file object. So you have a function that expects a file object to write to. Then you can give it that in-memory buffer instead of a file.
* BytesIO can be useful when you need to pass data to or from an API that expect to be given a file object.
* BytesIO implements read and write bytes data in memory.
* We create a BytesIO object and then write some bytes data into it.
* Please note that instead of writing a string, you write utf-8 encoded bytes with the BytesIO object.

**JSON Parser ():**

* It parses the incoming request JSON content into python content type dict.
* JSONParser. Parses **JSON request content**. **Request.data** will be populated with a dictionary of data.
* Data parsing is **the process of taking data in one format and transforming it to another format.**

**Ex: Creating or Inserting Data.**

* Create a new project and application.

**Settings.py:**

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

**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** myapp.models **import** Manager  
*# Register your models here.*@admin.register(Manager)  
**class** ManagerAdmin(admin.ModelAdmin):  
 list\_display = [**'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

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

**serializers.py:**

**from** rest\_framework **import** serializers  
**from** myapp.models **import** Manager  
  
**class** ManagerSerialzer(serializers.Serializer):  
 name=serializers.CharField(max\_length=20)  
 address=serializers.CharField(max\_length=20)  
 mail=serializers.CharField(max\_length=20)  
 age=serializers.IntegerField()  
  
 **def** create(self, validated\_data):  
 **return** Manager.objects.create(\*\*validated\_data)

**views.py:**

**from** django.shortcuts **import** render  
**import** io  
**from** rest\_framework.parsers **import** JSONParser  
**from** myapp.serializers **import** ManagerSerialzer  
**from** rest\_framework.renderers **import** JSONRenderer  
**from** django.http **import** HttpResponse  
**from** django.views.decorators.csrf **import** csrf\_exempt  
  
*# Create your views here.*@csrf\_exempt  
**def** create\_Manager(request):  
 **if** request.method==**'POST'**:  
 jsondata=request.body  
 stream=io.BytesIO(jsondata)  
 py\_data=JSONParser().parse(stream)  
 serializer=ManagerSerialzer(data=py\_data)  
 **if** serializer.is\_valid():  
 serializer.save()  
 result={**'message'**:**'Data inserted into database'**}  
 jsondata=JSONRenderer().render(result)  
 **return** HttpResponse(jsondata,content\_type=**'application/json'**)  
 jsondata=JSONRenderer().render(serializer.errors)  
 **return** HttpResponse(jsondata, content\_type=**'application/json'**)

**urls.py:**

**from** django.contrib **import** admin  
**from** django.urls **import** path  
**from** myapp **import** views  
  
urlpatterns = [  
 path(**'admin/'**, admin.site.urls),  
 path(**'createmanager/'**,views.create\_Manager),  
]

Create a new python file with the name test.py inside the application.

**test.py:**

**import** requests  
**import** json  
  
URL=**"http://127.0.0.1:8000/createmanager/"**data={  
 **'name'**:**'mohan'**,  
 **'address'**:**'hyderabad'**,  
 **'mail'**:**'mohan@gmail.com'**,  
 **'age'**:37  
}  
  
jsondata=json.dumps(data)  
r=requests.post(url=URL,data=jsondata)  
data=r.json()  
print(data)

* Now start server : python manage.py runserver
* Python test.py( in new terminal window)