Topic: 5. Serializing / Deserializing JSON Data (GET/POST)

Speaker: Personal | Notebook: API Development using Django Framework



To see more details about serializers, view this Youtube clip

1. To view the JSON file in a formatted style, we added the Google Chrome extension, JSON FORMATTER:

BEFORE:

Г	← → C ③ 127.0.0.1:8000/api/v1/students/	\$	Q
L	retty-print 🗌		
L	("id":1,"student_id":"S001","name":"Rosilie","branch":"Computer Science"),{"id":2,"student_id":"S002","name":"Yuri","branch":"Engineering"),{"i	":3,"student_id":"S003","name":"Xeria","branch":"Hote	l Mgt"}]

AFTER:

÷	\rightarrow	G	i	127.0.0.1 :80)00/api/v1/students/	
Pretty	-prin	t 🗹				
[{ "i "s "n }, { "i }, { "i }, { "i s "n } }]	d": 1 tuden ame": oranch d": 2 tuden ame": oranch ame": oranch	t_id" "Ros ": "C t_id" "Yur ": "E , t_id" "Xer ": "H	: "S0 ilie" omput : "S0 i", ngine : "S0 ia", otel	01", er Science" 02", ering" 03", Mgt"		

2. Previously, we manually used serializers to convert our query set into a list. The code is below to show the output in Step 1.



3. In Django, we can use serializer tools. In the API app folder, create a new file SERIALIZERS.PY:

×	=	$\leftarrow \rightarrow$,	oREST_APIProject	8~	8~ 0: 🛛 🖵 🗍		
L)	EXPLORER		🕏 views.py api	🕏 serializers.py 🔍	🕏 models.py	🕏 admin.py api		
ρ	✓ DJANGOREST_APIPROJECT ✓ api		api > 🕈 serializers. 1 from res	py > 😤 StudentSerializ t_framework impor	t serializers			
* °°°	 _pycache migrations _initpy admin.py 		2 from stu 3 4 class St 5 clas 6	dents.models impo udentSerializer(so s Meta: model = Student	rt Student erializers.Mode	lSerializer):		
æ~	 apps.py models.py serializers.py 		7	fields = "all	n			

4. Update our API\VIEWS.PY:

FROM manual serialization:



TO:

×	File Edit Selection View Go Run	Terminal	Help $\leftarrow \rightarrow$		>> DjangoRES1_	APIProjec
ر ل	EXPLORER ····	. py student	ts 🏾 🇬 urls.py django_rest_main	🗬 urls.py api	🍨 views.py api 🗙	🅏 ser
	✓ DJANGOREST_APIPROJECT	api > 🦸	🕈 views.py > 🛇 studentsView			
Q	∨ api		<pre># from django.shortcuts impo</pre>	rt render		
/~	>pycache		# from django.http import Js	onResponse		
90	> migrations		<pre>from students.models import</pre>	Student		
_₽ ₂	🔹 init .pv		from serializers import Stu	dentSerializer		
~	admin.pv		from rest_framework.response	1mport Response		
_ a >	apps py		from rest_framework import s	lalus re import ani vi	OW	
	 apps.py models py 	8		is import api_vi	CW	
П ^О		9	<pre>@api view(['GET'])</pre>			
	tests py	10	<pre>def studentsView(request):</pre>			
π	e tests.py	11	if request.method == 'GET'			
A	😴 urls.py	12	# get all the data from	the Student tabl		
	🐨 views.py		<pre>students = Student.objec</pre>	ts.all()		
	✓ django_rest_main	14	<pre># serialize or translate</pre>	the query set i	nto Json	
	>pycache	15	serializer = StudentSeri	alizer <mark>(students,</mark>	many=True)	
	🕏initpy	16	return Response(serializ	er.data, status=	status.HTTP_200_0	()
	🕏 asgi.py					
	a cottings py					

So, when you run the URL path again:

http://127.0.0.1:8000/api/v1/students/

← → C ① 127.0.0.1:8000/api/v1/students/	☆		Q	ជ		≡î	3
Django REST framework				api_o	djang	joadi	min
Students							
Students		0	PTIC	NS		GET	•
GET /api/v1/students/							
<pre>HTTP 200 OK Allow: GET, OPTIONS Content-Type: application/json Vary: Accept [</pre>							

5. Now, if you update your database model for a new record and use the GET button from Step 4, you will be able to use GET button to get the latest added records or you can simply refresh your page and that will be considered as a GET method.

← → C ① 127.0.0.1:8000/api	/v1/students/		☆	Q	ጏ	l =1	3
Django REST framework					api_dj	angoad	min
Students		_					
Students				ΟΡΤΙΟ	ONS	GET	•
GET /api/v1/students/							
<pre>HTTP 200 OK Allow: GET, OPTIONS Content-Type: application/json Vary: Accept [</pre>	e"						

6. Now using **POSTMAN**, you can copy the same API link paste it into the POSTMAN search bar and use GET method. It should return all records from the database. Simply click on + and add the same path we used from the browser. To use POSTMAN, this must be installed in your device.



7. To store data using the Django Rest Framework, update the VIEWS.PY to allow for POST method.

≺	\equiv	$\leftarrow \rightarrow$	DjangoREST_APIProject B □ □							
d,	EXPLORER ····	students	🗬 urls.py django_rest_main 🛛 🕏 urls.py api 🖉 views.py api 🗙 🌩 serializers							
	\vee DJANGOREST_APIPROJECT	api 🖂	Iviews.py >							
Q	∨ api		# from django.http import JsonResponse							
/-	>pycache		<pre>from students.models import Student</pre>							
90	> migrations		from .serializers import StudentSerializer							
حک	a init nv		<pre>from rest_framework.response import Response</pre>							
	admin.nv		from rest_framework import status							
		7	from rest_framework.decorators import api_view							
~	apps.py	8	Appining ([log] poct]]							
пQ	e models.py	10	<pre>(dof_ctudenteView(peruect))</pre>							
	🗬 serializers.py	if request method 'CET':								
	🕏 tests.py	12	# get all the data from the Student table							
Ä	🕏 urls.py	13	students = Student.objects.all()							
	🕏 views.py	14	# serialize or translate the query set into Json							
	✓ django_rest_main	15	serializer = StudentSerializer(students.manv=True)							
	>pycache	16	return Response(serializer.data, status=status.HTTP 200 OK)							
	🚔 init .pv	17	elif request.method == 'POST':							
		18	# saves data into our database							
	asgi.py	19	<pre>serializer = StudentSerializer(data=request.data)</pre>							
	e settings.py	20	<pre>if serializer.is_valid():</pre>							
	e uris.py	21	serializer.save()							
	🗬 wsgi.py	22	return Response(serializer.data, status=status.HTTP_201_CREATED)							
	> env	23	# if data are not valid							
	✓ students	24	print(serializer.errors)							
	>pycache	25	return Response(serializer.errors,status.HTTP_400_BAD_REQUEST)							
	> migrations									

When you reload your page, then you can insert a new post:

• sk	12 Stc 🚷 Pers	Sol Select	S 127.0.	s ×	G how c	🗘 Se	ettin	+			
$\leftarrow \rightarrow$	C 🛈 12	27.0.0.1:8000/api,	/v1/students/			Q	☆	Q	Ď	≡ſ	3
Djang	o REST framew	ork							api_d	jangoadn	nin
Stude	ents										
Stu	idents							OP	TIONS	GET	•
GET /a	pi/v1/students/										
HTTP 2 Allow: Conten Vary:	200 OK POST, OPTIONS, GET it-Type: application Accept	/json									
[{ }, { }, { }, { }, { }, } { }, } { }, } }	<pre>"id": 1, "student_id": "S0 "name": "Rosilie" "branch": "Comput "id": 2, "student_id": "S0 "name": "Yuri", "branch": "Engine "id": 3, "student_id": "S0 "name": "Xeria", "branch": "Hotel "id": 4, "student_id": "S0 "name": "Russell" "branch": "Veteri</pre>	01", er Science" 02", ering" 03", Mgt" 04", , nary"									
	Media type: Content:	application/json { "student_id": "{ "name": "Mary "branch": "Eng }	S005", Ann", ineering"],						~	
										C C	<i>4</i>

•	🀠 12 Sto 🌖 Perso	Select	3 127. 0.	S 🗙	G how	c 🏟 Settin	+	_	
÷	→ C ① 127	7.0.0.1:8000/ap	i/v1/student	s/		୧ 🕁	. Q	۲ ا	=1 😌
	Django REST framew	ork						api_dj	angoadmin
	Students								
	Students							OPTIONS	GET -
	POST /api/v1/students/								
	<pre>HTTP 201 Created Allow: POST, OPTIONS, GET Content-Type: application Vary: Accept { "id": 5, "student_id": "S005", "name": "Mary Ann", "branch": "Engineerin }</pre>	/json g"							
	Media type:	application/json							~
	Content:								POST

8. To use POSTMAN, add the path again. Select BODY, then RAW, then JSON. Add your records then select select the SEND method.

$r \equiv \leftarrow \rightarrow$	Home Workspaces ~ API Network	ĸ	Q Search Postman	°,- Invite	\$\$ \$ 🚫	Upgrade ~ -
A My Workspace	New Import	🐼 Overview	POST http://127.0.0.1:8000/ai ● +	-		✓ No environment ✓
Collections +		http://127.0.0.1:8000)/api/v1/students/			🖺 Save 🗸 Share
Environments	Your collecton • + •••	POST ~ http:	//127.0.0.1:8000/api/v1/students/			Send ~
ic oC	Your collection	Params Authorization	Headers (8) Body Scrip	ots Settings		Cookies
Flows	Authorization	🔿 none 🔿 form-data	O x-www-form-urlencoded	raw 🔿 binary 🔿 GraphQL	JSON 🗸	Beautify
History	Туре лякну 🗸	1 {"student_id":	"S006","name":"Ziggy","branc	h":"Business"}		
87	Create a collection for your requests					
	A collection lets you group related	Body Cookies Headers	s (10) Test Results 🕠		201 Created	• 18 ms • 391 B • 🖨 🛛 •••
	authorization, tests, scripts, and variables for all requests in it.	<pre>{} JSON ∨ ▷ Prev 1 { 2 "id": 6, 3 "student_ 4 "name": " 5 "branch": 6 }</pre>	iew ∛ð Visualize ∨ id": "S006", Ziggy", "Business"			≂ © Q 0

9. Now, to see the newly inserted record, use the GET method. You will then see the newly added record.

\equiv \leftarrow \rightarrow	Home Workspaces ~ API Network y Workspace New mport Overview er http://127.0.01.8000/api.++ © Upgrade ~ - • y Workspace New mport © Overview er http://127.0.01.8000/api.++ © No environment ~ y Workspace New mport © Overview er http://127.0.01.8000/api.v1/students/ © Save ~ Share isse + = = = = = = = = = = = = = = = = = = =						
A My Workspa	Ace New Import	🔗 Overview	GET http://127.0.0.1:8000/api	• +		~	No environment V
 ⇒ Home Workspace API Net My Workspace New Imperiod My Workspace New Imperiod Collections Collections Collections Collections Collection History Create a collection for your requests A collection lets you group related requests and easily set common authorization, tests, scripts, and 		http://127.0.0.1:8000	/api/v1/students/				🖹 Save 🗸 Share
Environments C Flows History Create a collect requests and east authorization, te	Vour collection • + •••• Your collection Authorization • Type Answy ·	GE1 nttp:// Params Authorization O none form-data 1 {"student_id":"	Headers (8) Body • > x-www-form-urlencodec "S006", "name": "Ziggy", "	Scripts Settings) GraphQL JSON	~	Cookies Beautify
	Create a collection for your related requests and easily set common authorization, tests, scripts, and variables for all requests in it. Create Collection	Body Cookies Headers {} JSON ✓ ▷ Previ 23 "name' 24 "brand 25 }, 26 { 27 "id": 28 "study 29 "brand 30 "brand 31 2 33 "id": 34 "study 35 "study 36 "brand 37 38]	s(10) Test Results ew ∛⊘ Visualize : "Russell", ch": "Veterinary" 5, ant_id": "S005", ": "Mary Ann", ch": "Engineering" 6, ant_id": "S006", ": "Ziggy", ch": "Business"	S Posthy	nt آت Bunner دیک	200 OK	6 ms 732 B D 000