

POST http://127.0.0.1:8000/? GET http://127.0.0.1:8000/ho PUT http://127.0.0.1:8000/up +

No environment

HTTP http://127.0.0.1:8000/?city=Sakri&state=Maharashtra&country=India

Save

POST

http://127.0.0.1:8000/?city=Sakri&state=Maharashtra&country=India

Send

Params Authorization Headers (12) Body Scripts Settings

Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON

Beautify

```
1 {
2   .... "city": "Pune",
3   .... "state": "Maharashtra",
4   .... "country": "India"
5 }
```

Body Cookies Headers (10) Test Results

Status: 200 OK Time: 14 ms Size: 388 B Save as example

Pretty

Raw

Preview

Visualize

JSON

```
1 {
2   "id": 5,
3   "city": "Pune",
4   "state": "Maharashtra",
5   "country": "India"
6 }
```



http://127.0.0.1:8000/home

Save

GET

http://127.0.0.1:8000/home

Send

Params Authorization Headers (8) Body Scripts Settings

Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON

Beautify

1 {

Body Cookies Headers (10) Test Results

Status: 200 OK Time: 12 ms Size: 597 B Save as example

Pretty

Raw

Preview

Visualize

JSON



```
1 [
2   {
3     "id": 1,
4     "city": "Sakri",
5     "state": "Maharashtra",
6     "country": "India"
7   },
8   {
9     "id": 2,
10    "city": "Dhule",
11    "state": "Maharashtra",
12    "country": "India"
13  },
14  {
15    "id": 3,
16    "city": "Malegaon_city_mmante",
17    "state": "Maharashtra",
18    "country": "India"
19  },
20  {
21    "id": 5,
22    "city": "Pune",
23    "state": "Maharashtra",
```



http://127.0.0.1:8000/update/3/



Save



PUT



http://127.0.0.1:8000/update/3/

Send



Params Authorization Headers (8) Body Scripts Settings

Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON

Beautify

```
1 {  
2   ... "city": "Malegaon_MMANTC",  
3   ... "state": "Maharashtra",  
4   ... "country": "India"  
5 }
```

Body Cookies Headers (10) Test Results



Status: 200 OK

Time: 14 ms

Size: 398 B



Save as example



Pretty

Raw

Preview

Visualize

JSON



```
1 {  
2   "id": 3,  
3   "city": "Malegaon_MMANTC",  
4   "state": "Maharashtra",  
5   "country": "India"  
6 }
```

HTTP http://127.0.0.1:8000/delete/5/ Save

DELETE http://127.0.0.1:8000/delete/5/ Send

Params Authorization Headers (6) Body Scripts Settings Cookies

Query Params

	Key	Value	Description	...	Bulk Edit
	Key	Value	Description		

Body Cookies Headers (10) Test Results

Status: 200 OK Time: 24 ms Size: 367 B Save as example

Pretty Raw Preview Visualize JSON

```
1 {
2   "data": "Record Deleted Successfully."
3 }
```

EXPLORER

...

▼ SUNDAY

▼ project

▼ app

> __pycache__

> migrations

🔌 __init__.py

🔌 admin.py

🔌 apps.py

🔌 models.py

🔌 serializers.py

🔌 tests.py

🔌 urls.py

🔌 views.py

▼ project

> __pycache__

🔌 __init__.py

🔌 asgi.py

🔌 settings.py

🔌 urls.py

🔌 wsgi.py

≡ db.sqlite3

🔌 manage.py

🔌 urls.py ...\project X

🔌 settings.py

🔌 serializers.py

project > project > 🔌 urls.py > ...

1

2

3

4

5

6

7

8

9

from django.contrib import admin

from django.urls import path,include

urlpatterns = [

path('admin/', admin.site.urls),

path("",include("app.urls"))

]

settings.py

```
INSTALLED_APPS = [  
    'app',  
    'rest_framework',  
    'corsheaders',  
    'django.contrib.admin',  
    'django.contrib.auth',  
    'django.contrib.contenttypes',  
    'django.contrib.sessions',  
    'django.contrib.messages',  
    'django.contrib.staticfiles',  
]
```

```
MIDDLEWARE = [  
    'django.middleware.security.SecurityMiddleware',  
    'django.contrib.sessions.middleware.SessionMiddleware',  
    'corsheaders.middleware.CorsMiddleware',  
    'django.middleware.common.CommonMiddleware',  
    'django.middleware.csrf.CsrfViewMiddleware',  
    'django.contrib.auth.middleware.AuthenticationMiddleware',  
    'django.contrib.messages.middleware.MessageMiddleware',  
    'django.middleware.clickjacking.XFrameOptionsMiddleware',  
]
```

```
CORS_ORIGIN_ALLOW_ALL=True # for cross-origin-resource-sharing like react js, postman
```

EXPLORER

...

SUNDAY

project

app

> __pycache__

> migrations

__init__.py

admin.py

apps.py

models.py

serializers.py

tests.py

urls.py

views.py

project

> __pycache__

__init__.py

asgi.py

settings.py

urls.py

wsgi.py

db.sqlite3

manage.py

urls.py ...\project

settings.py

serializers.py

urls.py ...\app

project > app > models.py > cityModel > __str__

1 from django.db import models

2

3 # Create your models here.

4

5

6 class cityModel(models.Model):

7 id = models.AutoField(primary_key=True)

8 city = models.CharField(max_length=500)

9 state = models.CharField(max_length=500)

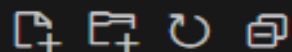
10 country = models.CharField(max_length=500)

11

12 def __str__(self) -> str:

13 return super().__str__()

SUNDAY



project

app

> __pycache__

> migrations

__init__.py

admin.py

apps.py

models.py

serializers.py

tests.py

urls.py

views.py

project

> __pycache__

__init__.py

asgi.py

settings.py

urls.py

wsgi.py

db.sqlite3

manage.py

urls.py ...\project

settings.py

serializers.py X

urls.py ...\app

project > app > serializers.py > citySerializer > Meta

1 from rest_framework import serializers

2 from .models import cityModel

3

4

5 class citySerializer(serializers.ModelSerializer):

6 class Meta:

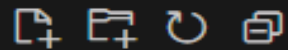
7 model = cityModel

8 fields = "__all__"

EXPLORER



SUNDAY



project

app

> __pycache__

> migrations

urls.py

admin.py

apps.py

models.py

serializers.py

tests.py

urls.py

views.py

project

> __pycache__

urls.py

asgi.py

settings.py

urls.py

wsgi.py

db.sqlite3

manage.py

urls.py ...\project

settings.py

serializers.py

urls.py ...\app X

project > app > urls.py > ...

1 from django.urls import path

2 from . import views

3

4 urlpatterns =

5 path("", views.create),

6 path("home/", views.list),

7 path("home/<int:id>/", views.list_detail),

8 path("delete/<int:id>/", views.delete_detail),

9 path("update/<int:id>/", views.update)

10]

SUNDAY

project

app

> __pycache__

> migrations

__init__.py

admin.py

apps.py

models.py

serializers.py

tests.py

urls.py

views.py

project

> __pycache__

__init__.py

asgi.py

settings.py

urls.py

wsgi.py

db.sqlite3

manage.py

project > app > views.py > list

```
1 from django.shortcuts import get_object_or_404, redirect, render
2 from .models import cityModel
3 from .serializers import citySerializer
4 from rest_framework.response import Response
5 from rest_framework.decorators import api_view
6 # Create your views here.
7
8 # new record
9 @api_view(['POST'])
10 def create(request):
11     if request.method == "POST" or None:
12         serializer = citySerializer(data=request.data)
13         if serializer.is_valid():
14             serializer.save()
15             return Response(serializer.data)
16         return Response(serializer.errors)
17
18 @api_view(['GET'])
19 def list(request):
20     instance = cityModel.objects.all()
21     serializer = citySerializer(instance=instance, many=True)
22     return Response(serializer.data)
23
24 @api_view(['GET'])
25 def list_detail(request, id):
26     instance = get_object_or_404(cityModel, id=id)
27     serializer = citySerializer(instance=instance)
28     if serializer is not None:
29         return Response(serializer.data)
30
31 @api_view(['DELETE'])
32 def delete_detail(request, id):
33     instance = get_object_or_404(cityModel, id=id)
34     if instance is not None:
35         instance.delete()
36         return Response({"data": "Record Deleted Successfully."})
37
38 @api_view(['PUT'])
39 def update(request, id):
40     instance = get_object_or_404(cityModel, id=id)
41     serializer = citySerializer(instance=instance, data=request.data)
42     if serializer.is_valid():
43         serializer.save()
44         return Response(serializer.data)
45     else:
46         serializer = citySerializer(instance=instance)
47         return Response(serializer.data)
48
```