

Topic: 16. DRF Searching

Speaker: Personal / Notebook: API Development using Django Framework



We can search through the database fields like BLOGS' title or comments.

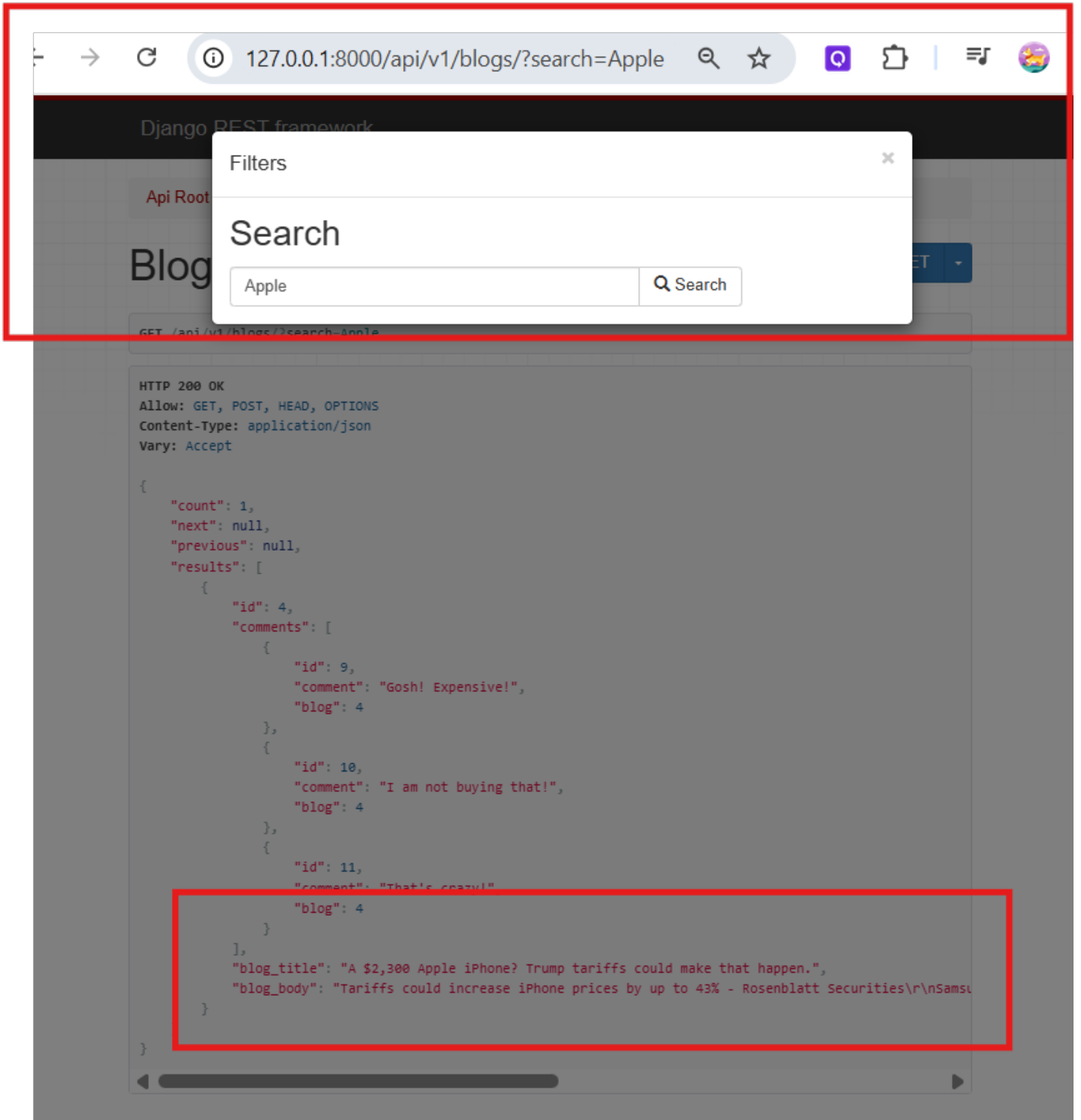
1. Go to APIVIEWS.PY:

```
File Edit Selection View Go ... DjangoREST_APIProject  
admin.py students urls.py django_rest_main urls.py api views.py blogs  
DJANGOREST_APIPROJECT  
  api  
    > __pycache__  
    > migrations  
    & _init_.py  
    & admin.py  
    & apps.py  
    & models.py  
    & paginations.py  
    & serializers.py  
    & tests.py  
    & urls.py  
    & views.py  
  > blogs  
  > django_rest_main  
  > employees  
  > env  
  > students  
  db.sqlite3  
api > views.py > ...  
1 from django.shortcuts import render, get_object_or_404  
2 # from django.http import JsonResponse  
3 from students.models import Student  
4 from employees.models import Employee  
5  
6 from .serializers import StudentSerializer, EmployeeSerializer  
7 from rest_framework.response import Response  
8 from rest_framework import status  
9 from rest_framework.decorators import api_view  
10 from rest_framework.views import APIView  
11 from django.http import Http404  
12  
13 from rest_framework import mixins, generics, viewsets  
14 from blogs.models import Blog, Comment  
15 from blogs.serializers import BlogSerializer, CommentSerializer  
16 from .paginations import CustomPagination  
17 from employees.filters import EmployeeFilter  
18 from rest_framework.filters import SearchFilter  
19  
20  
21
```

The image shows a Visual Studio Code editor window for a Django REST API project. The Explorer sidebar on the left shows the project structure under 'DJANGOREST_APIPROJECT', with 'api' expanded to show 'views.py'. The main editor area displays the code for 'views.py', specifically the 'CommentsView' class. A red box highlights the 'BlogsView' class definition, which includes a comment '# uses nested serializers' and attributes for 'serializer_class', 'filter_backends', and 'search_fields'. The code is as follows:

```
213 class EmployeeViewset(viewsets.ModelViewSet):
222     # uses the customized / case-insensitive class by employee
223     filterset_class = EmployeeFilter
224
225
226     # uses nested serializers
227     class BlogsView(generics.ListCreateAPIView):
228         queryset = Blog.objects.all()
229         serializer_class = BlogSerializer
230         filter_backends = [SearchFilter]
231         search_fields = ['blog_title']
232
233     class CommentsView(generics.ListCreateAPIView):
234         queryset = Comment.objects.all()
235         serializer_class = CommentSerializer
236
```

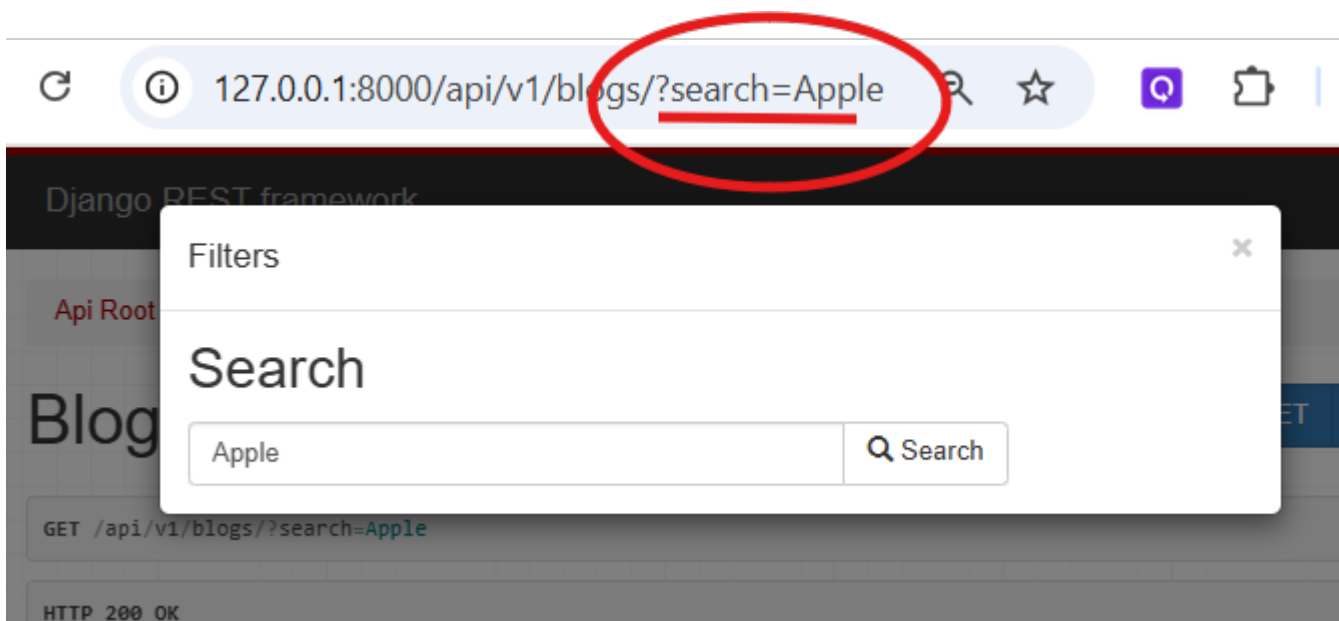
2. Now search thru BLOGS model:



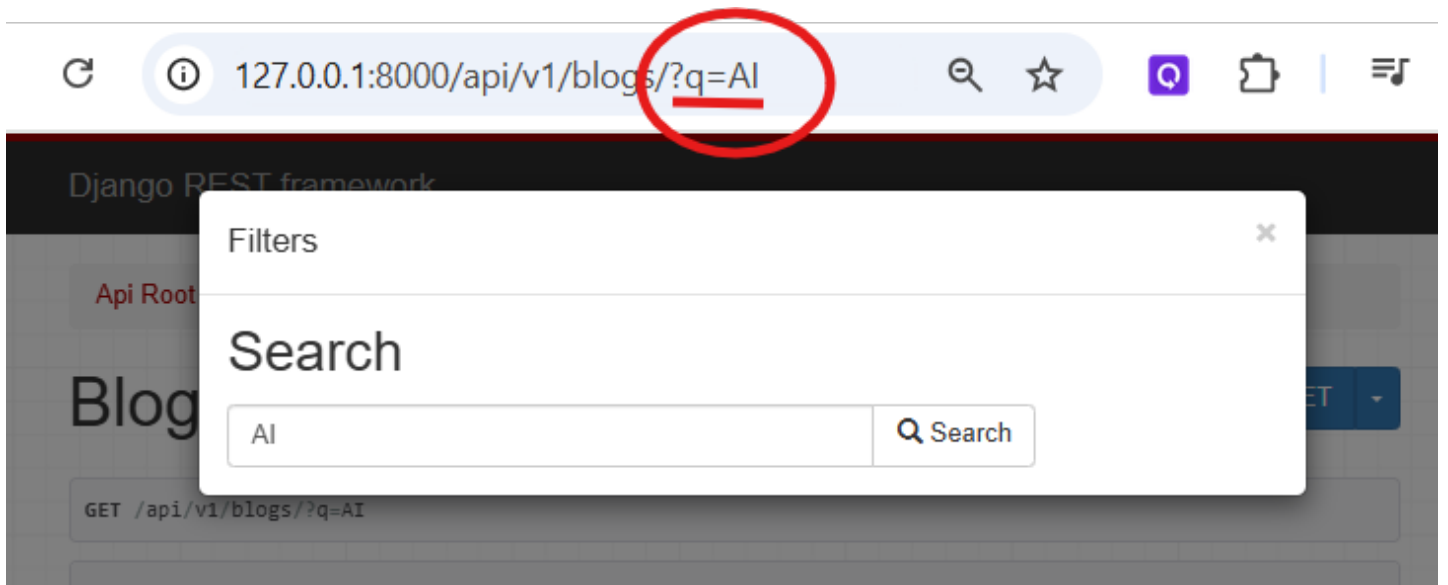
3. If you want to change the search keyword 'search' to 'q,' then update the SETTINGS.PY

```
131
132 REST_FRAMEWORK = {
133     # for Global Pagination
134     'DEFAULT_PAGINATION_CLASS': 'rest_framework.pagination.LimitOffsetPagination',
135     'PAGE_SIZE': 5,
136     # for Global Filtering
137     'DEFAULT_FILTER_BACKENDS': ['django_filters.rest_framework.DjangoFilterBackend'],
138     # overrides the query parameter instead of the default label 'search'
139     'SEARCH_PARAM' : 'q',
140 }
141
```

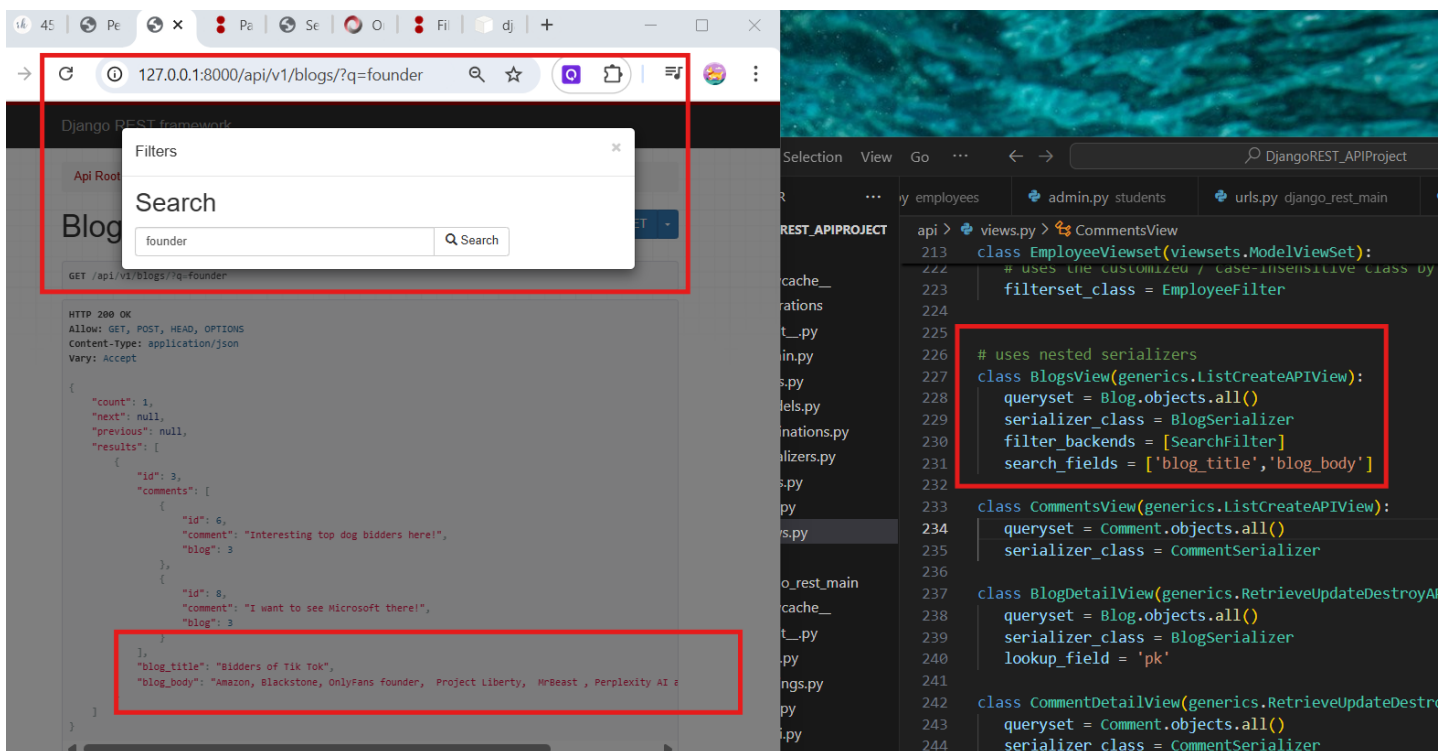
FROM:



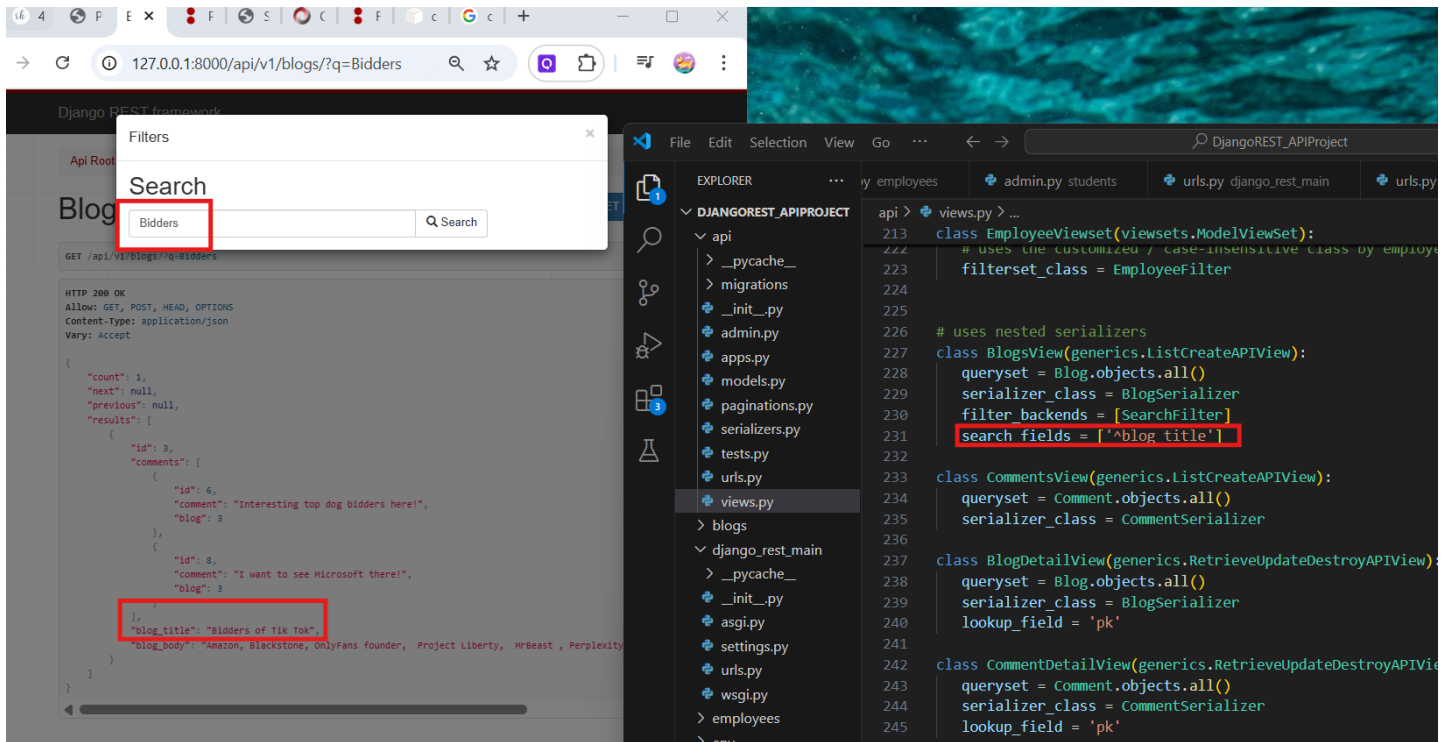
TO:



4. To search through the BLOG_BODY:



5. To search for a blog title that starts with a specific keyword, use the caret symbol (^).



But when you use a keyword, not the actual starting word, it won't show any result.

