

Topic: 16. DRF Searching

Speaker: Personal / Notebook: API Development using Django Framework

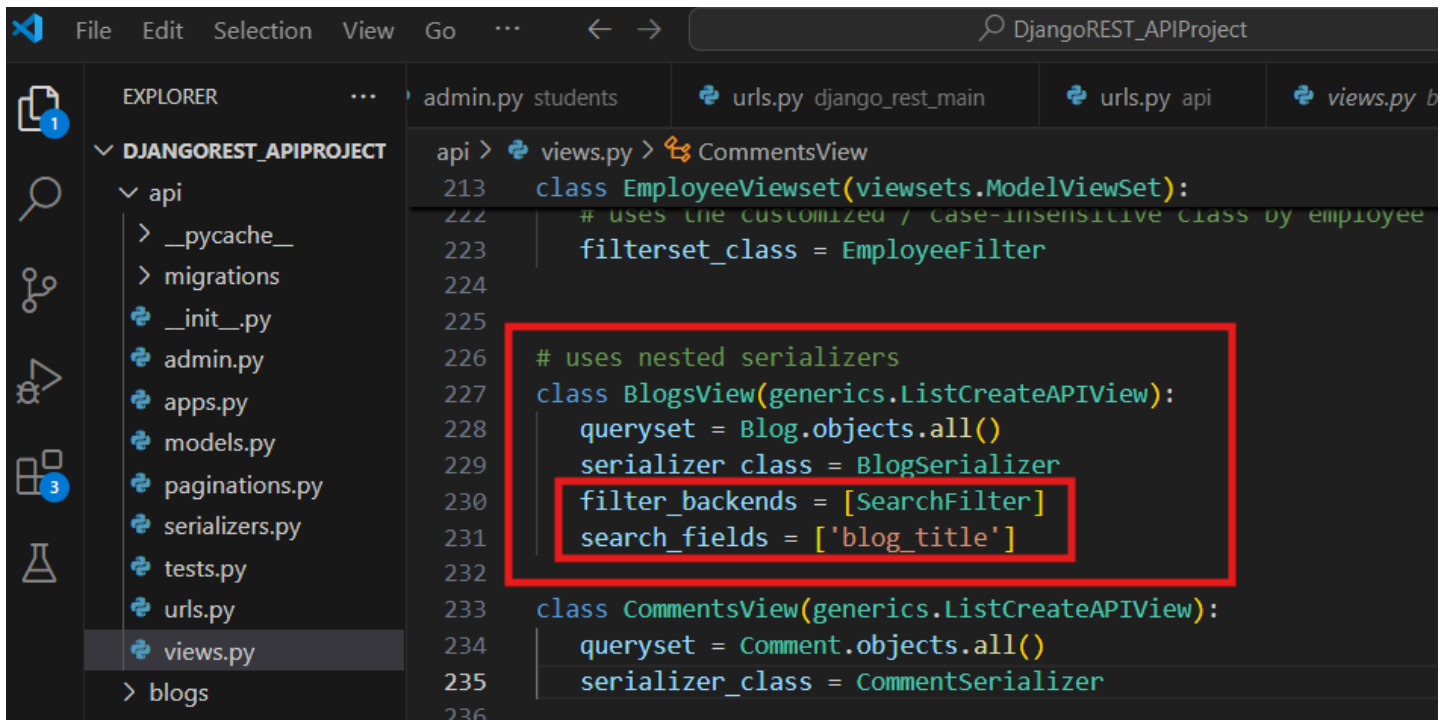


We can search through the database fields like BLOGS' title or comments.

1. Go to APIVIEWS.PY:

The screenshot shows a code editor with the Django REST API project structure on the left and the contents of `views.py` on the right. The Explorer panel on the left shows the project structure, with `views.py` selected under the `api` directory. The code in `views.py` includes imports for `render`, `get_object_or_404`, `JsonResponse`, `Student`, `Employee`, `StudentSerializer`, `EmployeeSerializer`, `Response`, `status`, `api_view`, `APIView`, `Http404`, `mixins`, `generics`, `viewsets`, `Blog`, `Comment`, `BlogSerializer`, `CommentSerializer`, `CustomPagination`, `EmployeeFilter`, and `SearchFilter`. The line `from rest_framework.filters import SearchFilter` is highlighted with a red box.

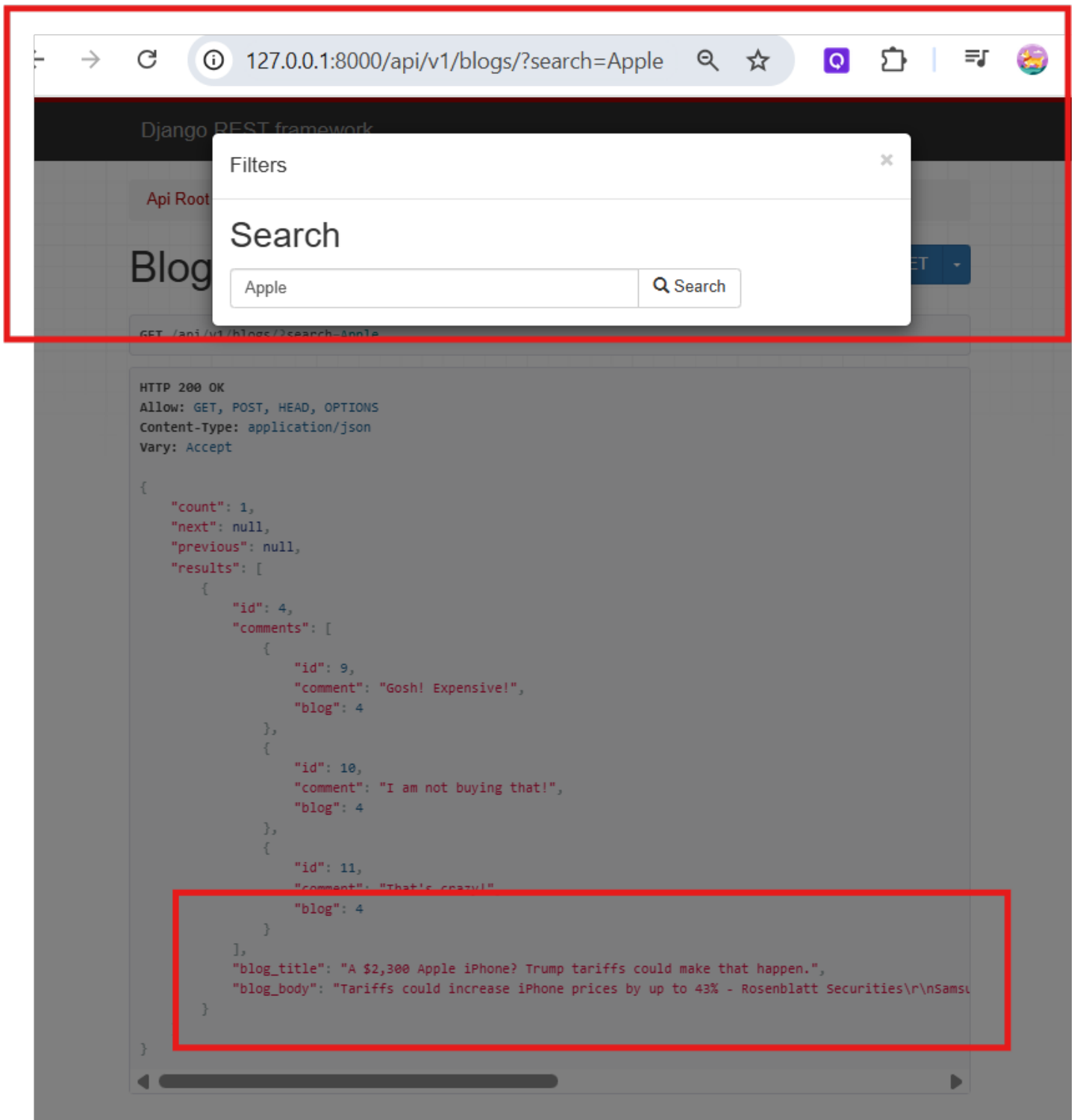
```
api > views.py > ...
1  from django.shortcuts import render, get_object_or_404
2  # from django.http import JsonResponse
3  from students.models import Student
4  from employees.models import Employee
5
6  from .serializers import StudentSerializer, EmployeeSerializer
7  from rest_framework.response 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
```



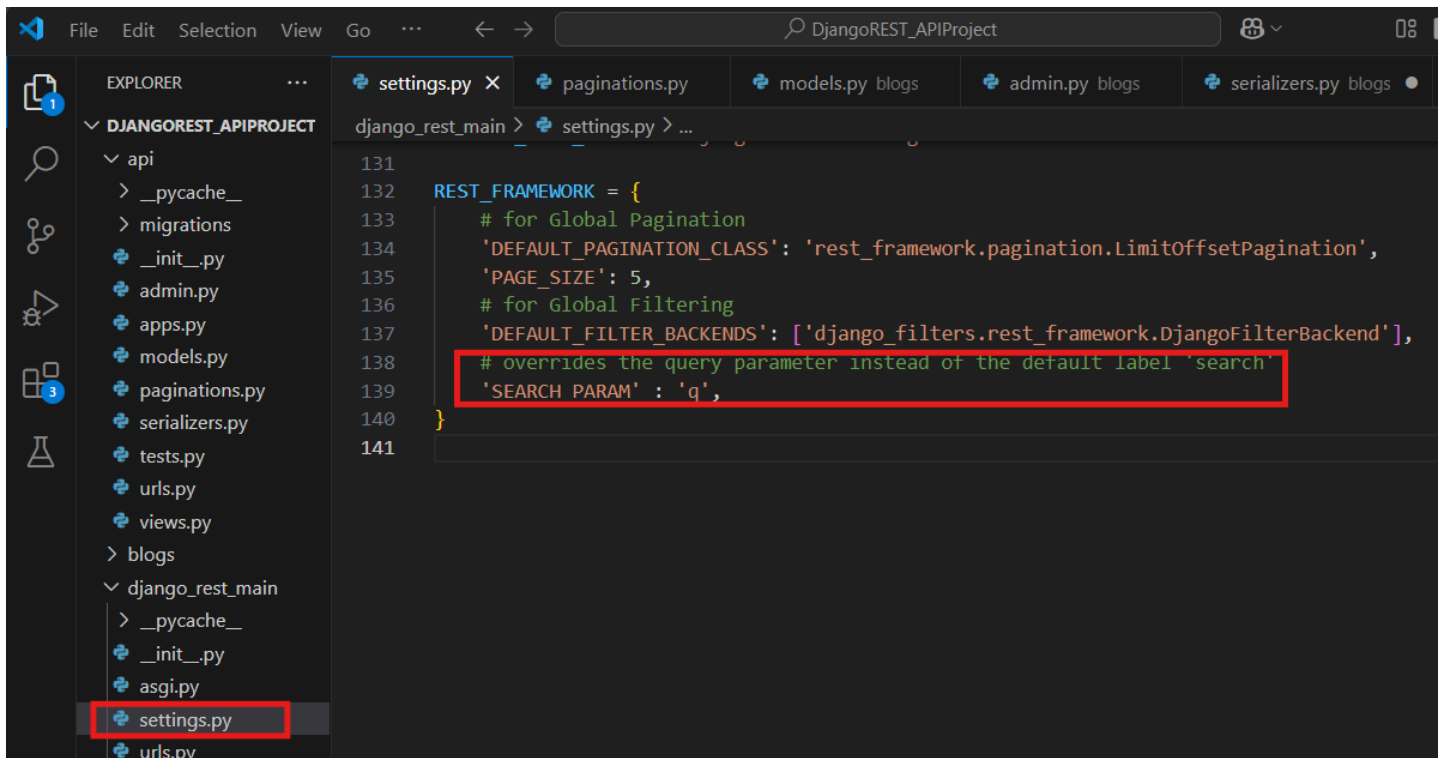
The screenshot shows a code editor with the Explorer sidebar on the left. The Explorer sidebar is expanded to show the 'api' directory, which contains files like __pycache__, migrations, __init__.py, admin.py, apps.py, models.py, paginations.py, serializers.py, tests.py, urls.py, and views.py. The 'views.py' file is selected. The main editor area shows the code for the 'EmployeeViewSet' and 'BlogAPIView' classes. The 'EmployeeViewSet' class is defined on line 213, and the 'BlogAPIView' class is defined on line 227. The 'BlogAPIView' class has a 'filter_backends' attribute set to '[SearchFilter]' and a 'search_fields' attribute set to '['blog_title']'. These two lines are highlighted with a red box. The 'CommentsView' class is also visible, starting on line 233.

```
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 BlogAPIView(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
```

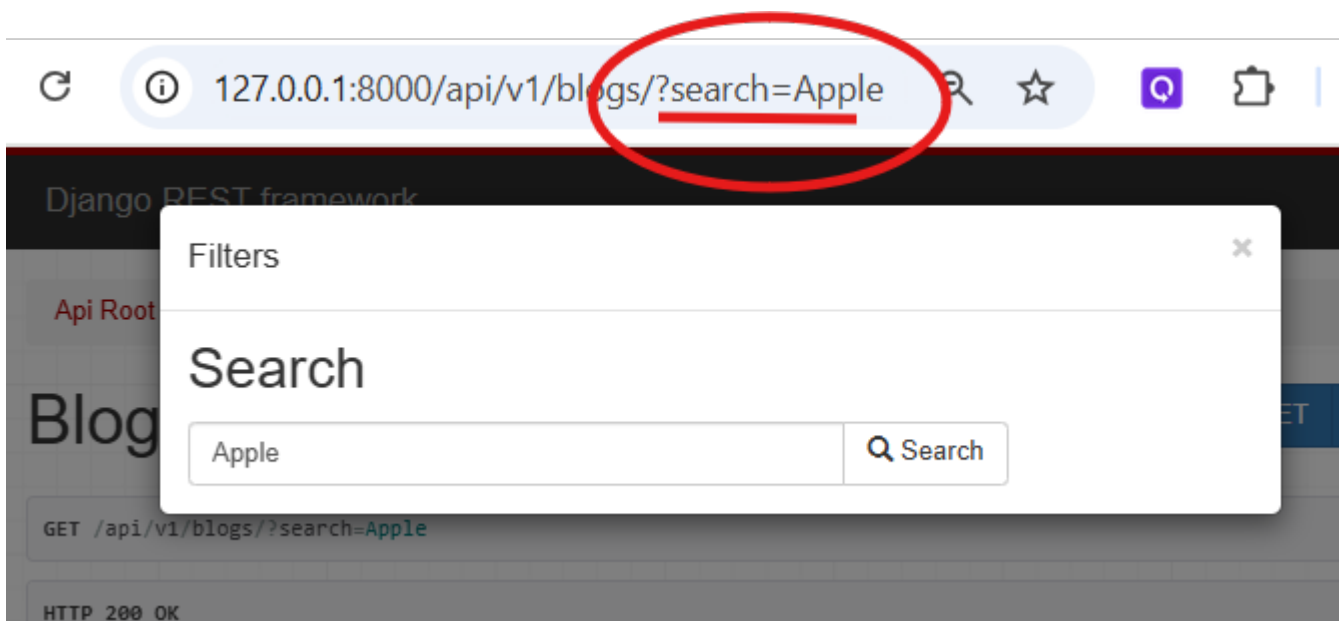
2. Now search thru BLOGS model:



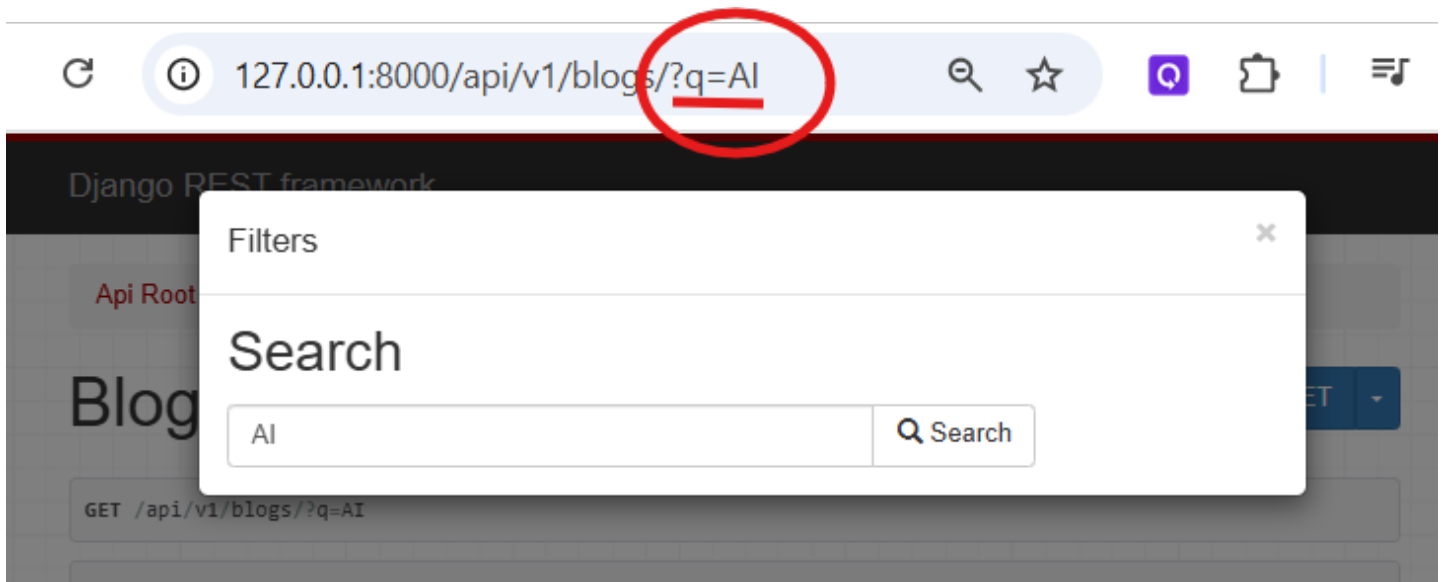
3. If you want to change the search keyword 'search' to 'q,' then update the SETTINGS.PY



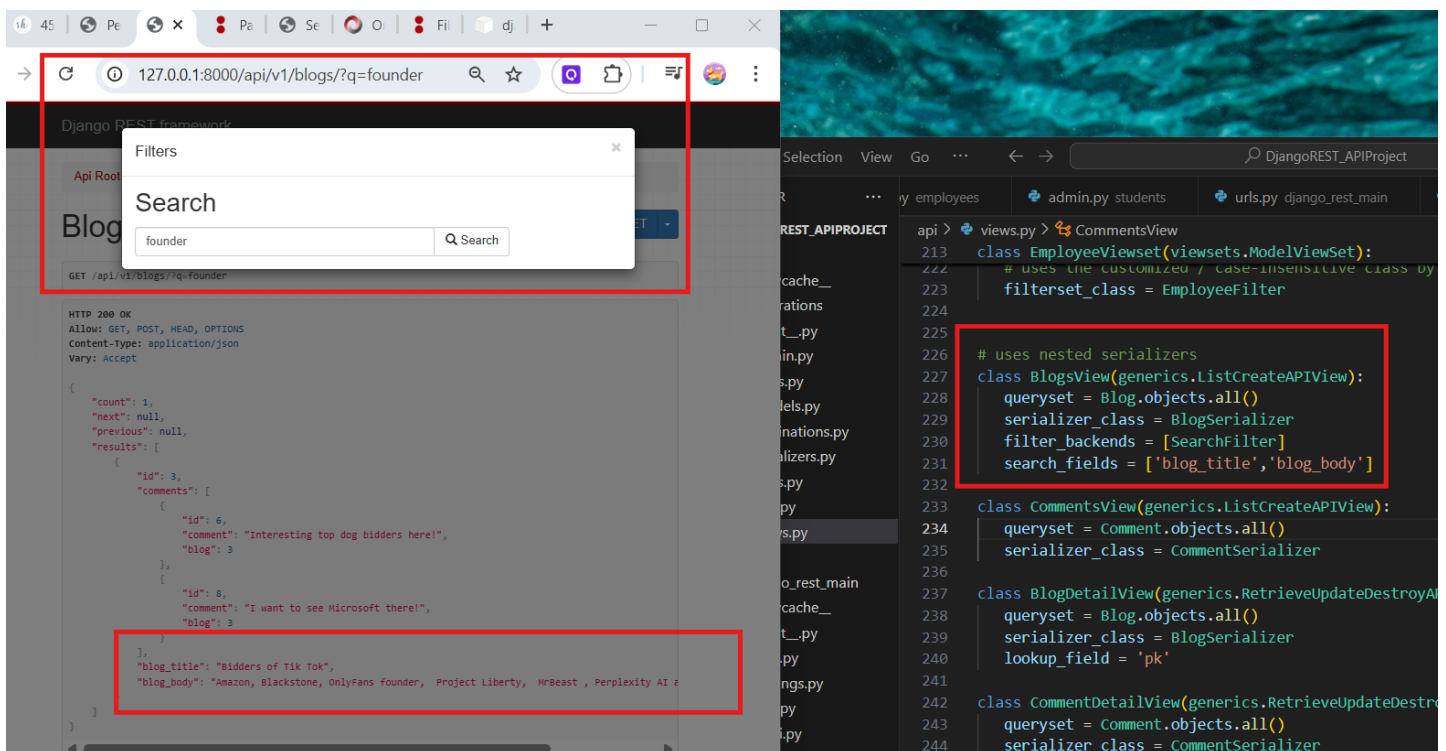
FROM:



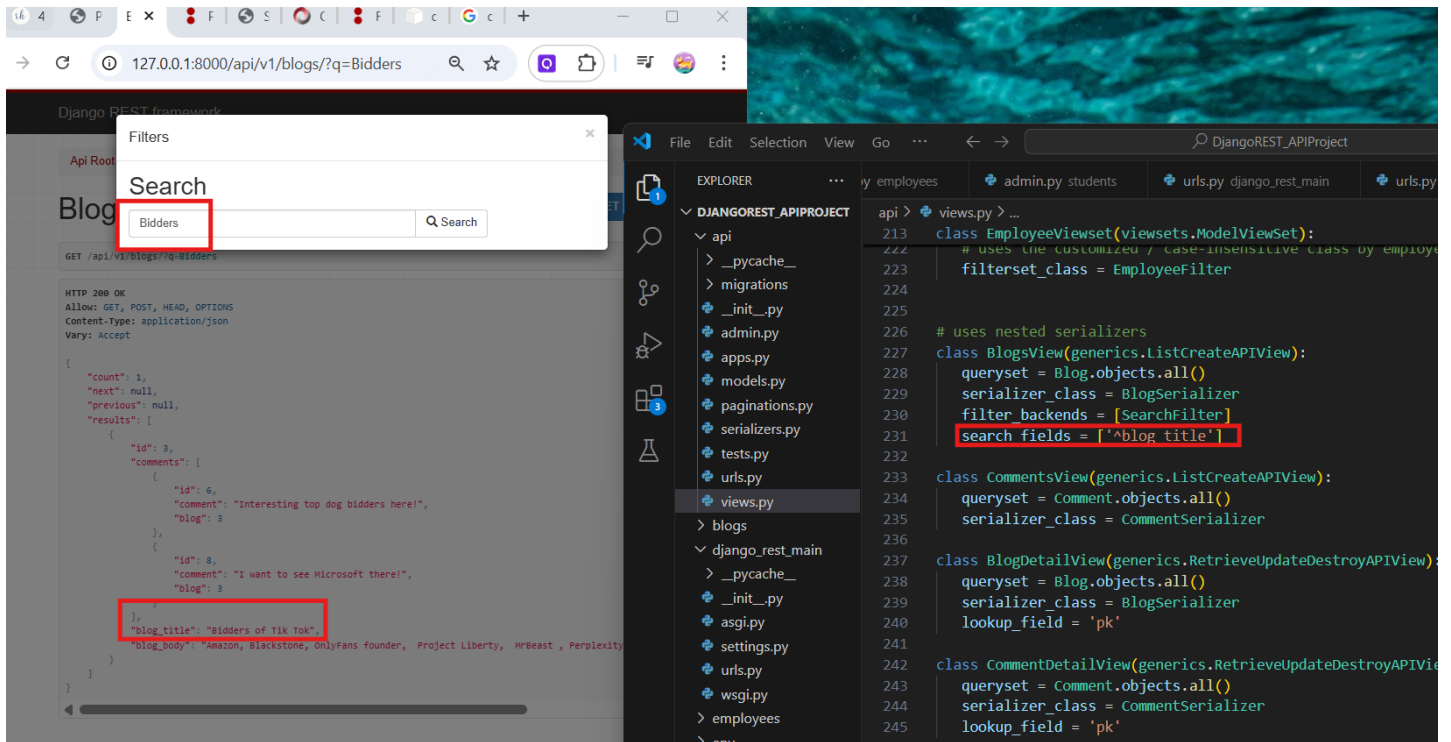
TO:



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.

