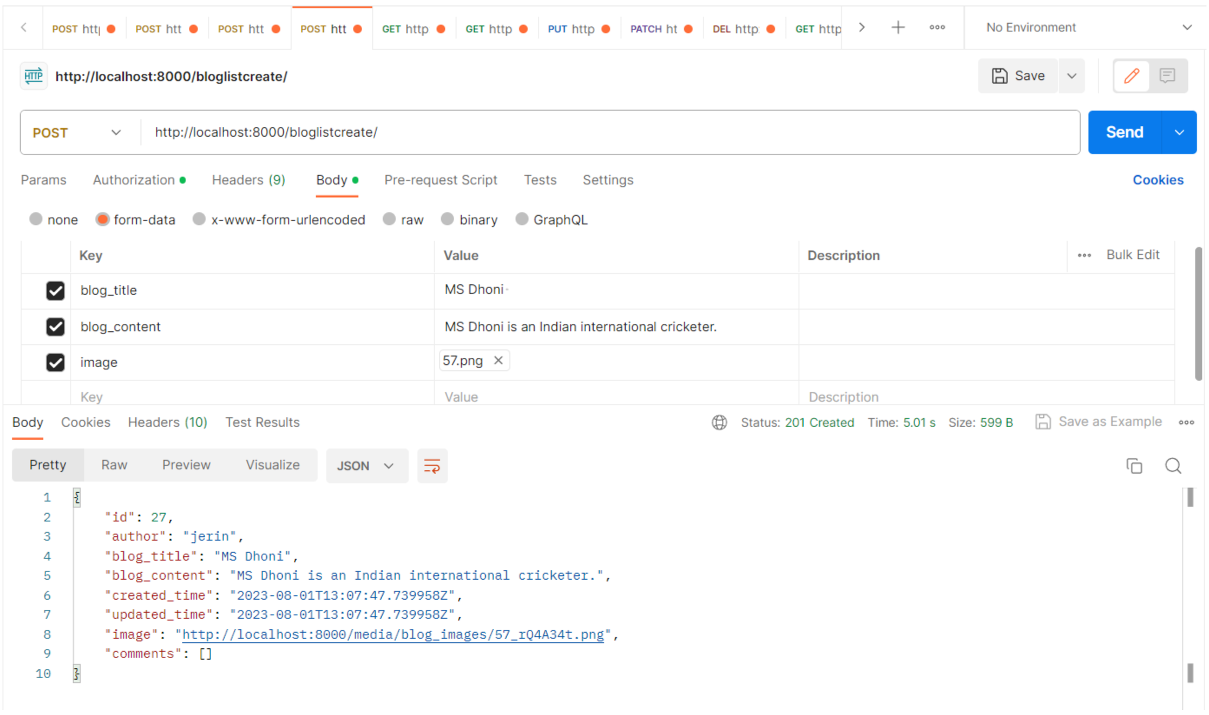
Blog Creation



http://localhost:8000/bloglistcreate/

POST http://localhost:8000/bloglistcreate/

none [✓ form-data] x-www-form-urlencoded raw binary GraphQL

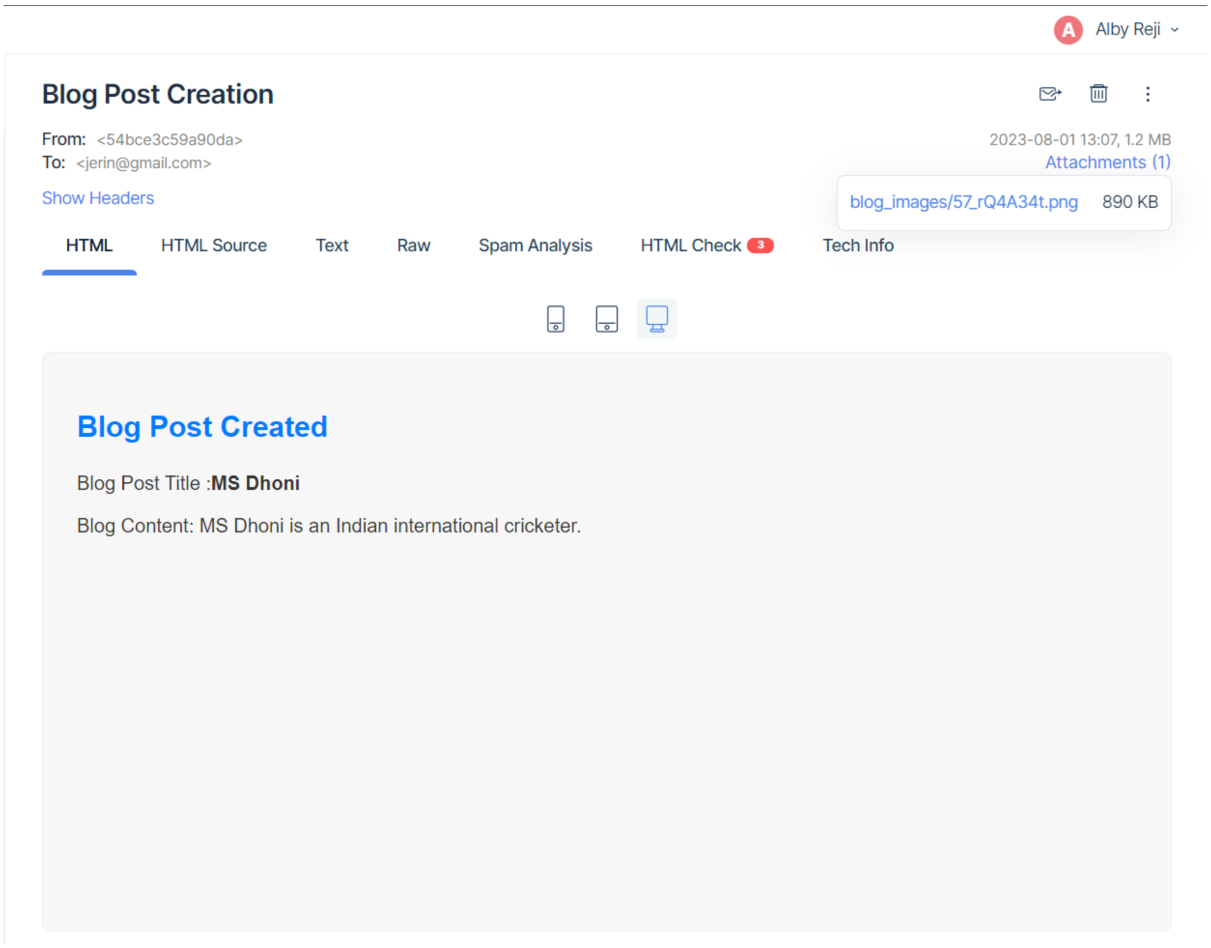
	Key	Value	Description
✓	blog_title	MS Dhoni	
✓	blog_content	MS Dhoni is an Indian international cricketer.	
✓	image	57.png	
	Key	Value	Description

Body Cookies Headers (10) Test Results

Status: 201 Created Time: 5.01 s Size: 599 B

```
{
    "id": 27,
    "author": "jerin",
    "blog_title": "MS Dhoni",
    "blog_content": "MS Dhoni is an Indian international cricketer.",
    "created_time": "2023-08-01T13:07:47.739958Z",
    "updated_time": "2023-08-01T13:07:47.739958Z",
    "image": "http://localhost:8000/media/blog_images/57_rQ4A34t.png",
    "comments": []
}
```

Email With Attachment



Blog Post Creation

From: <54bce3c59a90da>
To: <jerin@gmail.com>
Show Headers

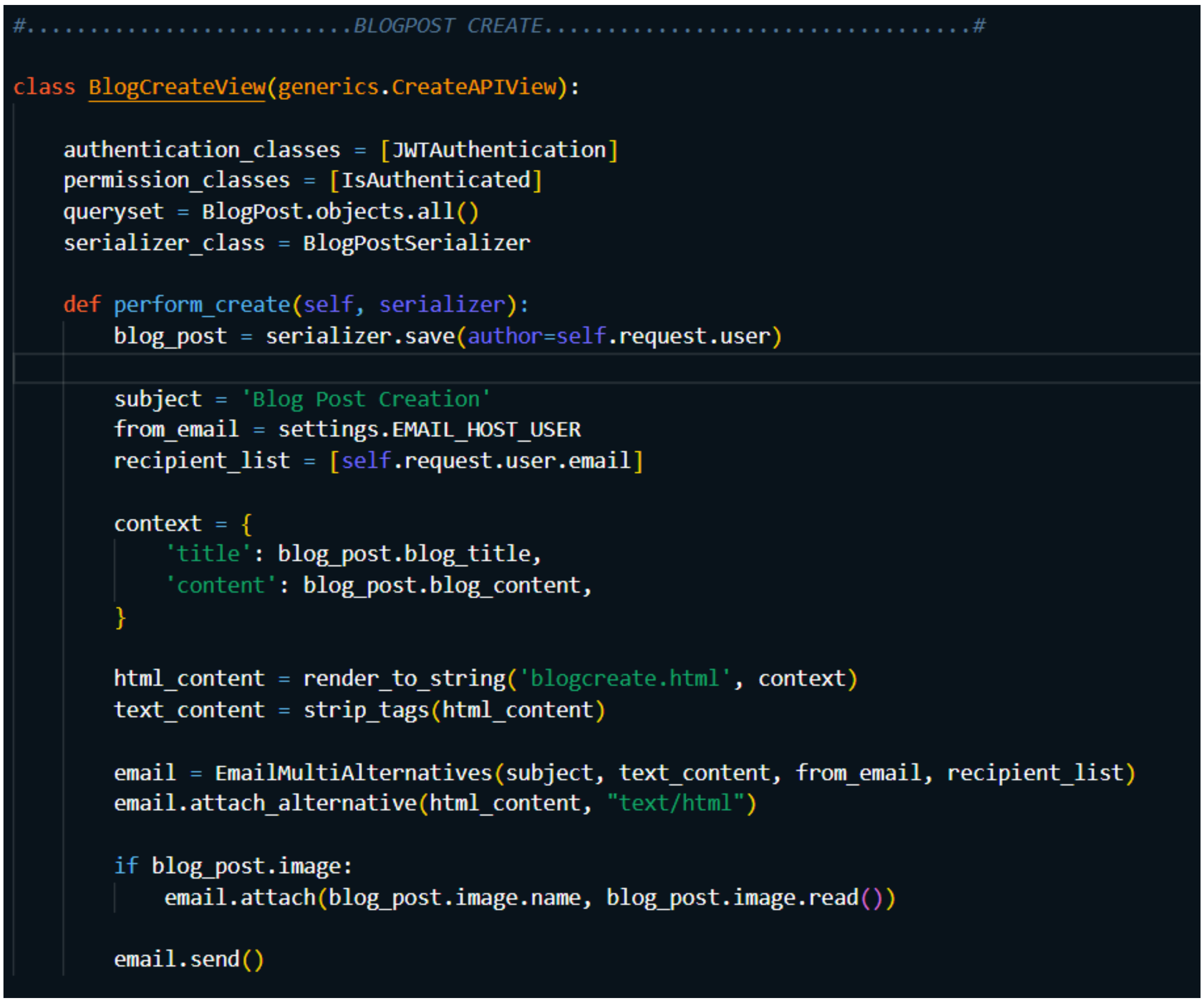
2023-08-01 13:07, 1.2 MB
Attachments (1)
blog_images/57_rQ4A34t.png 890 KB

Blog Post Created

Blog Post Title :**MS Dhoni**

Blog Content: MS Dhoni is an Indian international cricketer.

Code



```python
#........................BLOGPOST CREATE................................#

class BlogCreateView(generics.CreateAPIView):

    authentication_classes = [JWTAuthentication]
    permission_classes = [IsAuthenticated]
    queryset = BlogPost.objects.all()
    serializer_class = BlogPostSerializer

    def perform_create(self, serializer):
        blog_post = serializer.save(author=self.request.user)

        subject = 'Blog Post Creation'
        from_email = settings.EMAIL_HOST_USER
        recipient_list = [self.request.user.email]

        context = {
            'title': blog_post.blog_title,
            'content': blog_post.blog_content,
        }

        html_content = render_to_string('blogcreate.html', context)
        text_content = strip_tags(html_content)

        email = EmailMultiAlternatives(subject, text_content, from_email, recipient_list)
        email.attach_alternative(html_content, "text/html")

        if blog_post.image:
            email.attach(blog_post.image.name, blog_post.image.read())

        email.send()
```