

TO GIVE A COMMENTS FOR A SPECIFIC POST AND COUNT NUMBER OF COMMENTS  
IN VIEWS.PY

IMPORT HIZI KWANZA

```
from django.shortcuts import render, redirect, reverse, get_object_or_404
from django.urls import reverse_lazy
from django.views import generic
from django.contrib.messages.views import SuccessMessageMixin
from django.views.generic import CreateView, DetailView, DeleteView, UpdateView, ListView
```

ILA KM HAUJATUMIA HIYO HITCOUNT KWA AJILI YA KUTRACK NUMBER OF VIEWS BASI  
TUMIA ILE YA KAWAIDA INAYOITWA DETAILVIEW KWASABABU HIYO NI DETAILPAGE.

```
class PostDetail(HitCountDetailView):
    model = Post
    template_name = 'DimosoApp/postdetail.html'
    count_hit = True
    # HII NI CODES KWA AJILI YA POST DETAIL PAGE KWA KUTUMIA CLASS VIEW ZINAISHIA
    HAPA

    #SASA HIZI ZINAZOANZIA HAPA NI KWA AJILI YA COMMENT KWENYE POST HUSIKA
    form = CommentForm

    def post(self, request, *args, **kwargs):
        form = CommentForm(request.POST)
        if form.is_valid():
            post = self.get_object()
            form.instance.user = request.user
            form.instance.post = post
            form.save()
            messages.success(request, "message sent successfully")

            return redirect(reverse("PostDetail", kwargs={
                'pk':post.pk
            })))

    def get_context_data(self, **kwargs):
        new_posts = Post.objects.all().order_by('-id')[:6]
        #to count number of comment
        post_comments_count = Comment.objects.all().filter(post=self.object.id).count()
        #kwa ajili ya kudisplay comment huo mstari wa chini
        post_comments = Comment.objects.all().filter(post=self.object.id)
        #zinaendelea za kupost comment kwa admin
        context = super().get_context_data(**kwargs)
        #context["form"] = self.form
        context.update({
            'form':self.form,
            'post_comments':post_comments,
```

```

        'post_comments_count':post_comments_count,
        'new_posts':new_posts,
    })
    return context

```

KWENYE HIYO TEMPLATE YA DETAIL MAY BE POST\_DETAIL, KATENGENEZE HIYO FORM YA COMMENT NA FANYA YAFUATAYO.

```

<div class="newsletter" data-aos="fade-up" data-aos-delay="300">
    <!--HII YA CHINI NI KWA AJILI YA KUHESABU HIZO COMMENTS ACHANA NA HIYO
    YA KUHESABU VIEWS-->
    <h2 style="color: red; font-weight: bold;"> Comments:
    {{post_comments_count}}</h2>
    <h2 style="color: red; font-weight: bold;"> Views: <i class="fa fa-eye" aria-
    hidden="true"></i> {% get_hit_count for object %}</h2>

    {% if messages %}
    {% for message in messages %}

        <div class="alert alert-warning alert-dismissible fade show" role="alert">
        <strong style="">{{message}}</strong>
        <button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"
        style=""></button>
        </div>
        {% endfor %}
    {% endif %}

    <h2 style="color: black;font-weight: bold;font-family: serif;">Write a Comment</h2>

    <div class="form-element">

        <!--display content (KAMA UKITAKA KUZIDISPLAY HIZO COMMENT NDO HIZI
CODES-----
        {% for comment in post_comments %}
        {{comment.date}}
        {{comment.content}}

        {% endfor %}
        ----->

        <!--mwisho wa kudisplay content----->

```

```

<!--display form----->
<form method="POST" action="{% url 'PostDetail' object.pk %}">
    {% csrf_token %}
    {{form}}

    <button type="Submit" style="background-color: blue;color: white;font-weight:
bold;margin-top: 1cm;width: 6cm;padding: 10px;border-radius: 8px;font-size:
20px;">Submit</button>

</form>

<!--mwisho wa kudisplay FORM----->

</div>
</div>

```

IN FORMS.PY

```

class CommentForm(forms.ModelForm):
    content = forms.CharField(
        label=False,
        widget=forms.Textarea(attrs={

            'rows': '4',
            'id': 'content',
            'placeholder' : 'Write your Comment for this tutorial',
        }))
    class Meta:
        model = Comment
        fields = ('content',)

```

IN MODELS.PY

```

class Comment(models.Model):
    user = models.ForeignKey(settings.AUTH_USER_MODEL, on_delete=models.CASCADE)
    date = models.DateTimeField(auto_now_add=True)
    post = models.ForeignKey('Post', on_delete=models.CASCADE)
    content = models.TextField()

    def __str__(self):
        return self.user.email

```