

— How To Create —

Comment Section for Your Django Blogs

In just 5 steps



1

Install Django Comments XTD

Open a terminal and execute this command to
install django-comments-xtd

```
pip install django-comments-xtd
```

2

Add in settings.py

Open settings.py and add following:

```
INSTALLED_APPS = [  
    'django.contrib.admin',  
    'django.contrib.auth',  
    'django.contrib.sites',  
    'django.contrib.contenttypes',  
    'django.contrib.sessions',  
    'django.contrib.messages',  
    'django.contrib.staticfiles',  
  
    # django_comments_xtd and django_comments  
    # order should be same  
    'django_comments_xtd',  
    'django_comments',  
]
```

```
SITE_ID = 1  
  
COMMENTS_APP = 'django_comments_xtd'  
COMMENTS_XTD_MAX_THREAD_LEVEL = 2  
COMMENTS_XTD_CONFIRM_EMAIL = False  
  
EMAIL_HOST = "smtp.mail.com"  
EMAIL_PORT = "587"  
EMAIL_HOST_USER = ''  
EMAIL_HOST_PASSWORD = ''  
DEFAULT_FROM_EMAIL = ""
```

3

Create Blog Model and Views

```
class BlogPost(models.Model):
    title = models.CharField(
        _("Blog Title"), max_length=250,
        null=False, blank=False
    )
    slug = models.SlugField(unique=True)
    body = models.TextField()
    date_published = models.DateTimeField(null=True, blank=True,
                                           default=timezone.now
                                           )

    def __str__(self):
        return self.title

    def get_absolute_url(self):
        return reverse("blog:blog_detail",
                      kwargs={"slug": self.slug})
```

models.py

```
class BlogListView(ListView):
    queryset = BlogPost.objects.all()
    context_object_name = 'blog_posts'
    template_name = 'blog/blog_list.html'

class BlogDetailView(DetailView):
    model = BlogPost
    context_object_name = 'blog_detail'
    template_name = 'blog/blog_detail.html'
```

urls.py

```
urlpatterns = [
    path('', BlogListView.as_view(), name="blog"),
    path('<str:slug>/', BlogDetailView.as_view(), name="blog_detail"),
]
```

4

Add this in your main urls.py

```
path('comments/', include('django_comments_xtd.urls')),
```

Then run



```
python manage.py migrate
```

5

Add the following in your blog_detail.html

```
{% extends 'base.html' %}  
{% load static %}  
<!-- for comments --&gt;<br/>{% load comments %}  
{% load comments_xtd %}
```

**Display the total number of
comments
made on the blog post.**

```
{% get_comment_count for object as comment_count %}  
    <div class="py-4 text-center">  
        &nbsp;&sdot;&nbsp;  
        {{ comment_count }} comment{{ comment_count|pluralize }}  
        ha{{ comment_count|pluralize:"s,ve" }} been posted.  
    </div>
```

5

CONTINUE

Add the following in your blog_detail.html

```
{% if comment_count %}  
    <ul class="media-list shadow p-3 mb-5 bg-white rounded container">  
        {% render_xtdcomment_tree for object %}  
    </ul>  
{% endif %}
```

List the already posted comments.

Show the comment form, so that comments can be posted

```
<div class="comment mt-3 shadow p-3 mb-5 bg-white rounded container">  
    <h4 class="text-center mb-4">Your comment</h4>  
    <div class="card pt-4">  
        {% render_comment_form for object %}  
    </div>  
</div>
```

Do You Want Complete Article + Code?

<https://www.letscodemore.com/blog/how-to-create-a-comment-section-for-django-blog-in-5-steps/>



www.letscodemore.com