

Create Post Form Design

1. Design a form to create a Post for Author which contains following Fields

Step 1: Define the Post Model

In models.py:
from django.db import models

```
class Post(models.Model):
    title = models.CharField(max_length=200)
    content = models.TextField()
```

```
    def __str__(self):
        return self.title
```

Step 2: Create the PostForm

from django import forms
from .models import Post

```
class PostForm(forms.ModelForm):
    class Meta:
        model = Post
        fields = ['title', 'content']
        widgets = {
            'title': forms.TextInput(attrs={'class': 'form-control', 'placeholder': 'Enter post title'}),
            'content': forms.Textarea(attrs={'class': 'form-control', 'placeholder': 'Enter content'}),
        }
```

2. Design a suitable url and view to access the form with GET Request.

from django.shortcuts import render, redirect
from .forms import PostForm

```
def create_post(request):
    if request.method == 'POST':
        form = PostForm(request.POST)
        if form.is_valid():
            form.save()
            return redirect('post_success') # or any other view after saving
    else:
        form = PostForm()
    return render(request, 'create_post.html', {'form': form})
```

3. Design view to save the data received from the form into Model Post.

```
<!DOCTYPE html>
<html>
<head>
    <title>Create Post</title>
</head>
<body>
```

```
<h2>Create New Post</h2>
<form method="POST">
    {% csrf_token %}
    {{ form.as_p }}
    <button type="submit">Submit</button>
</form>
</body>
</html>
Add the URL pattern
from django.urls import path
from .views import create_post

urlpatterns = [
    path('create-post/', create_post, name='create_post'),
]
```