**For Creating Front end UI with just 2 text boxes**

**2. Prompt**

create django front end for displaying two text boxes , guide me through the installations

**Answer**

Sure, here's a step-by-step guide to creating a Django front-end with two text boxes, including the installation process:

Step 1: Install Python and Virtual Environment Make sure you have Python installed on your system. You can download it from the official Python website:<https://www.python.org/downloads/>

After installing Python, open a terminal or command prompt and install virtualenv to create a virtual environment for your Django project:

bash

pip install virtualenv

Step 2: Create a Virtual Environment and Install Django Create a new directory for your Django project and navigate to it in the terminal. Then, create a virtual environment and activate it:

bash

# Create a new directory and navigate to it

mkdir my\_django\_project

cd my\_django\_project

# Create and activate a virtual environment

virtualenv venv

source venv/bin/activate # On Windows, use: venv\Scripts\activate

Next, install Django within the virtual environment:

bash

pip install django

Step 3: Create a Django Project and App Inside your project directory, create a new Django project and navigate to it:

bash

django-admin startproject myproject .

cd myproject

Now, create a new Django app within your project:

bash

python manage.py startapp myapp

Step 4: Create the HTML Template Inside your app's templates folder (create it if it doesn't exist), create a new HTML file named textboxes.html:

html

<!-- myapp/templates/textboxes.html -->

<!DOCTYPE html>

<html>

<head>

<title>Two Text Boxes</title>

</head>

<body>

<h1>Two Text Boxes Example</h1>

<form method="post">

{% csrf\_token %}

<label for="textbox1">Text Box 1:</label>

<input type="text" id="textbox1" name="textbox1"><br>

<label for="textbox2">Text Box 2:</label>

<input type="text" id="textbox2" name="textbox2"><br>

<button type="submit">Submit</button>

</form>

</body>

</html>

Step 5: Create a View In your app's views.py, create a view function to render the textboxes.html template:

python

# myapp/views.py

from django.shortcuts import render

def textboxes\_view(request):

return render(request, 'textboxes.html')

Step 6: Set Up URL Routing In your app's urls.py, set up a URL pattern to map to the textboxes\_view:

python

# myapp/urls.py

from django.urls import path

from . import views

urlpatterns = [

path('textboxes/', views.textboxes\_view, name='textboxes'),

]

Step 7: Start the Django Development Server Navigate back to the project directory and start the Django development server:

bash

python manage.py runserver

Open your web browser and access http://127.0.0.1:8000/myapp/textboxes/ to see the page with two text boxes.

This guide should help you create a Django front-end with two text boxes. Remember to replace 'myproject', 'myapp', and 'my\_django\_project' with appropriate names according to your project.

**Prompt**

I'm getting error from this above code, which is "Page not found (404) Request Method: GET Request URL:http://127.0.0.1:8000/myapp/textboxes/" how to correct this

**Answer**

I apologize for any confusion. The error you're encountering is likely due to a misconfiguration in your URL routing. Let's troubleshoot this step by step.

First, please make sure you've followed the previous steps correctly, including creating the project, app, views, and templates as described. If you've made any typos or mistakes, please correct them.

Here's a summary of the steps again:

1. Create a new project:  
   bash

django-admin startproject myproject .

Create a new app:  
bash

python manage.py startapp myapp

Define the view in myapp/views.py:  
python

from django.shortcuts import render

def textboxes\_view(request):

return render(request, 'textboxes.html')

Set up URL routing in myapp/urls.py:  
python

from django.urls import path

from . import views

urlpatterns = [

path('textboxes/', views.textboxes\_view, name='textboxes'),

]

Create the textboxes.html template in myapp/templates/textboxes.html.

Run the development server:  
bash

python manage.py runserver

1. Access the page at http://127.0.0.1:8000/myapp/textboxes/ in your browser.

If you've followed these steps and are still encountering the 404 error, double-check the following:

1. Verify that the myapp app is listed in the INSTALLED\_APPS setting in myproject/settings.py.
2. Make sure you've correctly set up the URL routing in myproject/urls.py to include the myapp.urls module.
3. Ensure that your project's directory structure is accurate and matches the steps provided.

If you're still having trouble, please provide more details about your project's directory structure, the contents of myproject/settings.py, and any additional relevant code. This will help me provide more targeted assistance to identify and resolve the issue.

**Prompt**

what should I write in the myproject/urls.py

**Answer:**

In the myproject/urls.py file, you need to include the URL patterns from your app's urls.py module. This is done using the include function from Django's urls module. Here's how you should configure the myproject/urls.py file:

Assuming your app is named myapp and you want to include its URL patterns:

python

from django.contrib import admin

from django.urls import path, include

urlpatterns = [

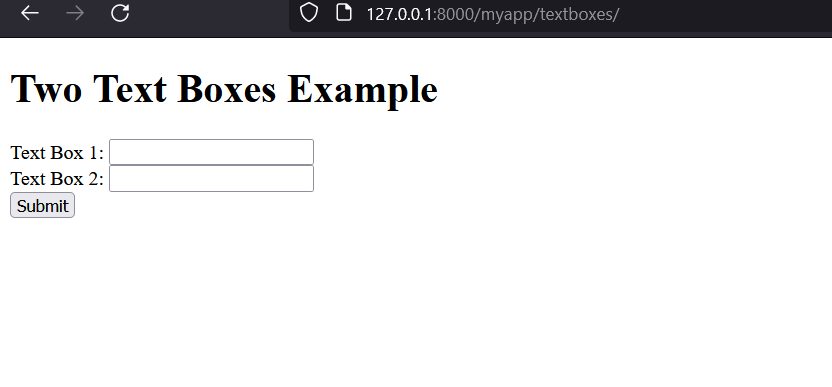
path('admin/', admin.site.urls),

path('myapp/', include('myapp.urls')), # Include your app's URL patterns

]

In the above code, the line path('myapp/', include('myapp.urls')) includes the URL patterns from your myapp app. It means that any URL starting with http://127.0.0.1:8000/myapp/ will be handled by your app's URLs. In your case, you've defined a URL pattern for /textboxes/ in your myapp/urls.py, so the complete URL for accessing the text boxes page will be http://127.0.0.1:8000/myapp/textboxes/.

Make sure that you have this configuration in your myproject/urls.py file, and it should correctly route requests to your app's views.



**Requirements**

python>=3.9

Django==3.2.8