Event Countdown Timer using Django

Last Updated : 18 Mar, 2024

To create anticipation and excitement for these events, one useful tool is a countdown timer that displays the time remaining until the event starts. In this article, we'll explore how to build an event countdown timer using Django, a popular Python web framework.

Event Countdown Timer using Django

An event countdown timer using Django in [Python](#) can be created by storing the event's datetime in a [Django](#) model and using [JavaScript](#) to display a live countdown on a web page based on the current time and the event's datetime.

Setting Up the Project Folder

To start the project use this command

```
django-admin startproject core
cd core
```

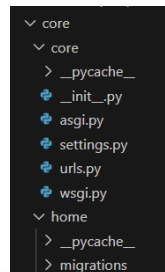To start the app use this command

```
python manage.py startapp home
```

Now add this app to the 'settings.py'
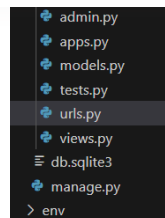
```
INSTALLED_APPS = [
    "django.contrib.admin",
    "django.contrib.auth",
    "django.contrib.contenttypes",
    "django.contrib.sessions",
    "django.contrib.messages",
    "django.contrib.staticfiles",
```

```
        "home",
    ]
```

File Structure

Setting Necessary Files

core/models.py: Here we have created an Event table with name, and **event_date** field in the table.

Python3

```python
1   from django.contrib import admin
2   from .models import Event
3
4   # Register your models here.
5   admin.site.register(Event
```

core/views.py : In below code view fetches the first Event object, calculates the time remaining until its event date, and formats it into hours, minutes, and

seconds. If no event exists, it renders 'No Event' with all time values set to zero. Finally, it passes the data to the 'myapp.html' template for rendering.

Python3

```python
from django.shortcuts import render

# Create your views here.
from django.utils import timezone
from .models import Event  # Import the Event model from your app

def countdown_timer(request):
    event = Event.objects.first()  # Retrieve the first Event object
    if event:
        time_remaining = event.event_date - timezone.now()  # Calculate time remaining
        hours = time_remaining.seconds // 3600
        minutes = (time_remaining.seconds % 3600) // 60
        seconds = time_remaining.seconds % 60
        data = {
            'name': event.name,
            'hours': hours,
            'minutes': minutes,
            'seconds': seconds
        }
    else:
        data = {
            'name': "No Event",
            'hours': 0,
            'minutes': 0,
            'seconds': 0
        }
    return render(request, 'myapp.html', {'data': data})
```

core/urls.py : Below Django urlpatterns configuration sets up routes for accessing the admin interface and includes routes defined in the 'home' app.

Python3

```python
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('home.urls'))

]
```

home/urls.py : Below urlpatterns configuration maps the root URL to the countdown_timer view function from the current application, with the name 'countdown_timer'.

Python3

```python
from django.urls import path
from . import views

urlpatterns = [
    path('', views.countdown_timer, name='countdown_timer'),
]
```

Creating GUI our Project

home/templates/myapp.html: This page will show the timer and event name.

HTML

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Event Countdown Timer</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            text-align: center;
```

```
                background-color: #f4f4f4;
11          }
12
13          h1 {
14              color: #333;
15          }
16
17          #timer {
18              background-color: #fff;
19              border-radius: 5px;
20              box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
21              padding: 20px;
22              margin: 20px auto;
23              max-width: 300px;
24          }
25
26          #countdown {
27              font-size: 24px;
28              font-weight: bold;
29              color: #e44d26;
30          }
31
32          #event {
33              font-size: 34px;
34              font-weight: bold;
35              color: #221d1c;
36          }
37      </style>
38  </head>
39  <body>
40      <h1>Event Countdown Timer</h1>
41      <div id="timer">
42          <h3 id="event">{{ data.name }}</h3>
43          <h2>Time Remaining:</h2>
44          <div id="countdown">
45              {{ data.hours|stringformat:"02d" }} : {{
    data.minutes|stringformat:"02d" }} : {{
    data.seconds|stringformat:"02d" }}
46          </div>
47      </div>
48
49      <script>
50          // Function to update the countdown timer
51          function updateTimer() {
```

```
                   var hours =
      parseInt(document.getElementById('countdown').textContent.sp
      lit(':')[0]);
53              var minutes =
      parseInt(document.getElementById('countdown').textContent.sp
      lit(':')[1]);
54              var seconds =
      parseInt(document.getElementById('countdown').textContent.sp
      lit(':')[2]);
55
56             if (seconds > 0) {
57                 seconds--;
58             } else {
59                 if (minutes > 0) {
60                     minutes--;
61                     seconds = 59;
62                 } else {
63                     if (hours > 0) {
64                         hours--;
65                         minutes = 59;
66                         seconds = 59;
67                     } else {
68                         // Timer has reached zero, you can
      handle this case as needed
69                     }
70                 }
71             }
72
73             // Update the timer display
74             document.getElementById('countdown').textContent
      =
75                 (hours < 10 ? '0' : '') + hours + ' : ' +
76                 (minutes < 10 ? '0' : '') + minutes + ' : '
      +
77                 (seconds < 10 ? '0' : '') + seconds;
78         }
79
80         // Update the timer every second
81         setInterval(updateTimer, 1000);
82     </script>
83 </body>
84 </html>
```

admin.py: Here we are registering our models.

Python3

```
1   from django.contrib import admin
2   from .models import Event
3
4   admin.site.register(Event)
```
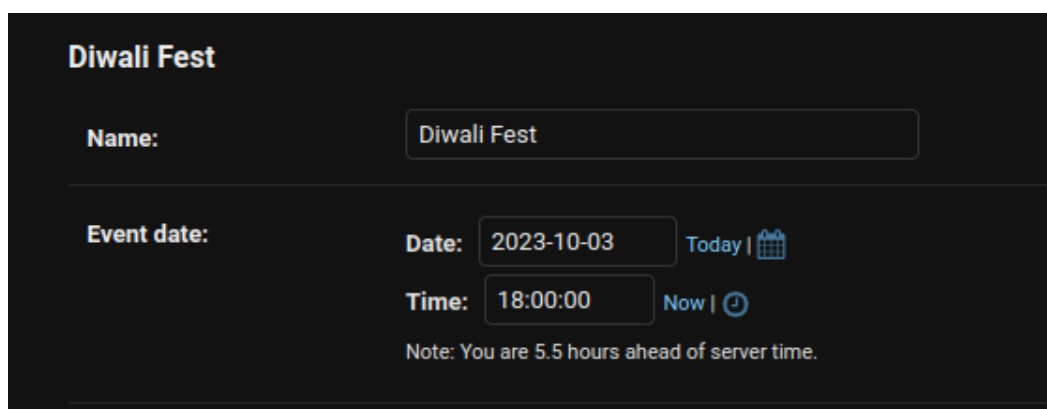
Deployement of the Project

Run these commands to apply the migrations:

```
python3 manage.py makemigrations
python3 manage.py migrate
```

Before running the development server, create a superuser and add an event in the Django admin panel:

```
python3 manage.py createsuperuser
```

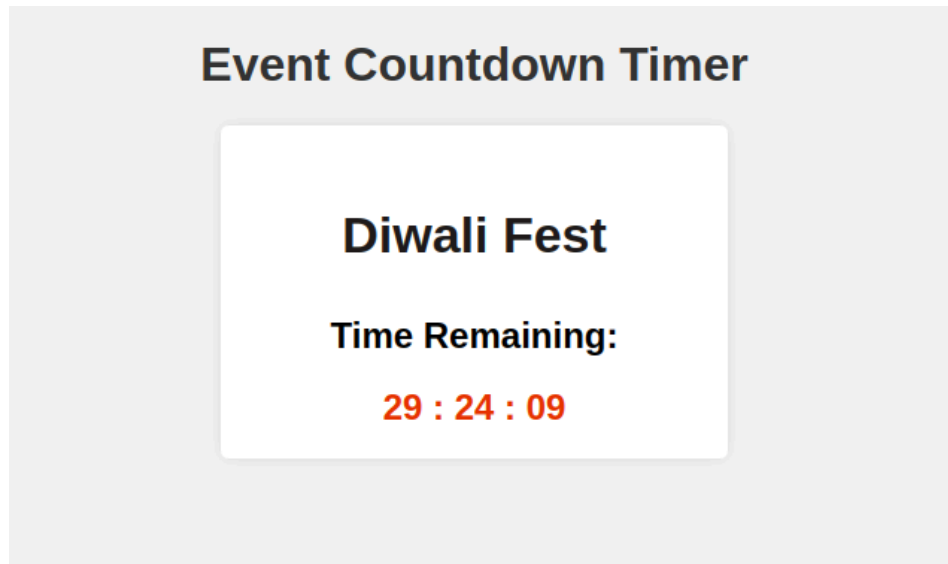After creating super user, go admin pannel and add event.

Run the server with the help of following command:

```
python3 manage.py runserver
```

Output

A amit…   1

Previous Article   Next Article

Python Compiler Using Django

Similar Reads

Event Countdown Timer Using React

Event Countdown Timer is an application where you can create an event with the name, date, and category of the event and it will create a card that will show yo…

5 min read

Task Pomodoro Timer using Django

This article explores the development of a Task Pomodoro Timer using Django. Users can set, stop, pause, and delete time intervals for tasks. To engage these…

12 min read

Create a Countdown using Bootstrap 5

The countdown timer displays the remaining time from a set future date, showing days, hours, minutes, and seconds until the specified event or deadline. In this…

3 min read

Create Timer Based Phone Silencer App using React-Native

In this project, we will create a mobile application using React Native that acts as a timer-based phone silencer. The app allows users to set a timer, and when the...

3 min read

Create a Quiz App using React Hooks and Timer-Based State Management

Quiz Application is a web-based application built using React Hooks and Timer-Based State Management. It provides users with a series of multiple-choice...

5 min read

Building a Pomodoro Timer with VueJS

A Pomodoro Timer is a time management tool that helps individuals break work into the intervals traditionally 25 minutes in the length separated by the short...

3 min read

Event Management System Using Python Django

In today's fast-paced world, businesses, schools, and groups need to organize events well. An Event Management System (EMS) makes planning and managin...

13 min read

Create a new Django project in Pycharm using Pycharm Terminal

PyCharm is one of the most popular Python-IDE developed by JetBrains used for performing scripting in Python language. PyCharm provides many useful featur...

2 min read

A Guide to Sending Data Using Cache in Django

In this article, we aim to comprehensively explore the functionality of sending data using cache in Django by providing a well-structured example...

6 min read

Blog Post Recommendation using Django

In this article, we will guide you through the creation of a blog post recommendation system using Django. Our article covers the integration of a...

10 min read

Article Tags :   Django   Geeks Premier League   Project   Geeks Premier League 2023

GeeksforGeeks

Corporate & Communications Address:-
A-143, 9th Floor, Sovereign Corporate
Tower, Sector- 136, Noida, Uttar Pradesh
(201305) | Registered Address:- K 061,
Tower K, Gulshan Vivante Apartment,
Sector 137, Noida, Gautam Buddh
Nagar, Uttar Pradesh, 201305

GET IT ON Google Play   Download on the App Store

Company

About Us

Legal

In Media

Contact Us

Advertise with us

GFG Corporate Solution

Placement Training Program

GeeksforGeeks Community

Languages

Python

Java

C++

PHP

GoLang

SQL

R Language

Android Tutorial

Tutorials Archive

DSA

Data Structures

Algorithms

DSA for Beginners

Basic DSA Problems

DSA Roadmap

Top 100 DSA Interview Problems

DSA Roadmap by Sandeep Jain

All Cheat Sheets

Data Science & ML

Data Science With Python

Data Science For Beginner

Machine Learning

ML Maths

Data Visualisation

Pandas

NumPy

NLP

Deep Learning

Web Technologies

HTML

CSS

JavaScript

TypeScript

ReactJS

NextJS

Bootstrap

Web Design

Python Tutorial

Python Programming Examples

Python Projects

Python Tkinter

Web Scraping

OpenCV Tutorial

Python Interview Question

Django

Computer Science

Operating Systems

Computer Network

Database Management System

Software Engineering

Digital Logic Design

Engineering Maths

Software Development

Software Testing

DevOps

Git

Linux

AWS

Docker

Kubernetes

Azure

GCP

DevOps Roadmap

System Design

High Level Design

Low Level Design

UML Diagrams

Interview Guide

Design Patterns

OOAD

System Design Bootcamp

Interview Questions

Inteview Preparation

Competitive Programming

Top DS or Algo for CP

Company-Wise Recruitment Process

Company-Wise Preparation

Aptitude Preparation

Puzzles

School Subjects

Mathematics

Physics

Chemistry

Biology

Social Science

English Grammar

Commerce

World GK

GeeksforGeeks Videos

DSA

Python

Java

C++

Web Development

Data Science

CS Subjects