



[Turtle](#) [Tkinter](#) [Matplotlib](#) [Python Imaging Library](#) [Pyglet](#) [Python](#) [Numpy](#) [Pandas](#) [Python Database](#)

How to create a text input box with Pygame?

Last Updated : 26 Mar, 2021

In this article, we will discuss how to create a text input box using PyGame.

Installation

Before initializing pygame library we need to install it. This library can be installed into the system by using *pip* tool that is provided by Python for its library installation. Pygame can be installed by writing these lines into the terminal.

We can install Pygame using command :

```
pip install pygame
```

Pygame can be used to create a text input box which will be explained step by step further in this article.

Approach

1. Use `pygame.init()` which will initialize all imported modules.
2. Set screen size.
3. Set font of the text which user will type.
4. Create a condition according to user key.
5. Also, declare two variable which will contain color name which will be further used for input color.
6. Also, store input in a variable to display on screen.
7. Now draw rectangle and pass argument which should be on screen.
8. Also, set the size of screen to be rendered.
9. Use `clock.tick()` which means that for every second at most given frames should be passed.

Functions Used

Function	Description
<code>clock.tick()</code>	It is used to refresh the frame in given second
<code>pygame.exit()</code>	It is used to quit game
<code>pygame.init()</code>	It is used to initialize all imported module
<code>pygame.font.Font</code>	Create a new Font object from a file
<code>pygame.display.flip()</code>	It will update only a portion of the screen to updated, not full area
<code>screen.fill((r, g, b, a))</code>	It will set the background color of the screen. The range is between 0 and 255.

Implementation

Python3

```
# import sys module
import pygame
import sys

# pygame.init() will initialize all
# imported module
pygame.init()

clock = pygame.time.Clock()

# it will display on screen
screen = pygame.display.set_mode([600, 500])

# basic font for user typed
base_font = pygame.font.Font(None, 32)
user_text = ''

# create rectangle
input_rect = pygame.Rect(200, 200, 140, 32)

# color_active stores color(lightskyblue3) which
# gets active when input box is clicked by user
```

```
color_active = pygame.Color('lightskyblue3')

# color_passive store color(chartreuse4) which is
# color of input box.
color_passive = pygame.Color('chartreuse4')
color = color_passive

active = False

while True:
    for event in pygame.event.get():

        # if user types QUIT then the screen will close
        if event.type == pygame.QUIT:
            pygame.quit()
            sys.exit()

        if event.type == pygame.MOUSEBUTTONDOWN:
            if input_rect.collidepoint(event.pos):
                active = True
            else:
                active = False

        if event.type == pygame.KEYDOWN:

            # Check for backspace
            if event.key == pygame.K_BACKSPACE:

                # get text input from 0 to -1 i.e. end.
                user_text = user_text[:-1]

            # Unicode standard is used for string
            # formation
            else:
                user_text += event.unicode

        # it will set background color of screen
        screen.fill((255, 255, 255))

        if active:
            color = color_active
        else:
            color = color_passive

        # draw rectangle and argument passed which should
        # be on screen
        pygame.draw.rect(screen, color, input_rect)

        text_surface = base_font.render(user_text, True, (255, 255, 255))
```

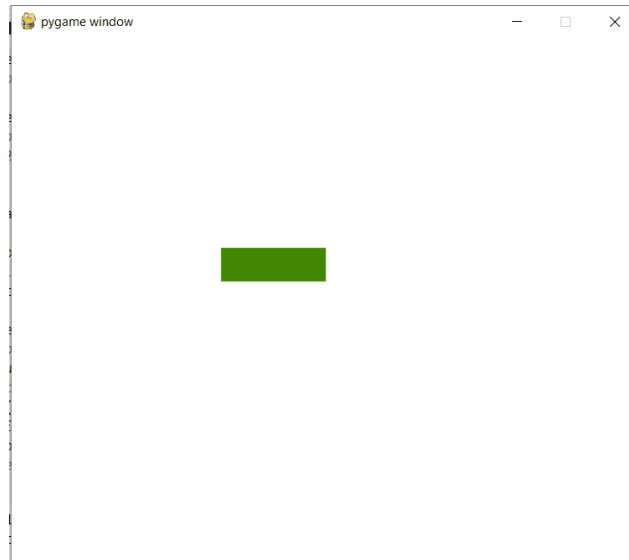
```
# render at position stated in arguments
screen.blit(text_surface, (input_rect.x+5, input_rect.y+5))

# set width of textfield so that text cannot get
# outside of user's text input
input_rect.w = max(100, text_surface.get_width()+10)

# display.flip() will update only a portion of the
# screen to updated, not full area
pygame.display.flip()

# clock.tick(60) means that for every second at most
# 60 frames should be passed.
clock.tick(60)
```

Output:



Looking to dive into the world of programming or sharpen your Python skills? Our [Master Python: Complete Beginner to Advanced Course](https://www.geeksforgeeks.org/master-python-complete-beginner-to-advanced-course/) is your ultimate guide to becoming proficient in Python. This course covers everything you need to build a solid foundation from fundamental programming concepts to advanced techniques. With **hands-on projects**, real-world examples, and expert guidance, you'll gain the confidence to tackle complex **coding challenges**. Whether you're starting from scratch or aiming to enhance your skills, this course is the perfect fit. Enroll now and master Python, the language of the future!



14

Previous Article

Python | Display text to PyGame window

Next Article

Python | Display images with PyGame

Similar Reads

Adding Collisions Using `pygame.Rect.colliderect` in Pygame

Prerequisite: Drawing shapes in Pygame, Introduction to pygame In this article, we are going to use `pygame.Rect.colliderect` for adding collision in a shape usin...

3 min read

Working with Input box/Test Box in Selenium with Python

Selenium is an effective device for controlling an internet browser through the program. It is purposeful for all browsers, works on all fundamental OS and its...

1 min read

How to get keyboard input in PyGame ?

While using pygame module of Python, we sometimes need to use the keyboard input for various operations such as moving a character in a certain direction. To...

3 min read

Pygame - Input Handling

Pygame is a cross-platform set of Python modules designed for writing video games. It includes computer graphics and sound libraries designed to be used...

5 min read

Text Input box with a verification button in kivy

Kivy is a platform-independent GUI tool in Python. As it can be run on Android, IOS, linux and Windows etc. It is basically used to develop the Android...

3 min read

Text Input box with a verification button in kivy (using .kv file)

Kivy is a platform-independent GUI tool in Python. As it can be run on Android, IOS, linux and Windows etc. It is basically used to develop the Android...

3 min read

How to Get the Input From Tkinter Text Box?

Tkinter Text box widget is used to insert multi-line text. This widget can be used for messaging, displaying information, and many other tasks. The important task...

1 min read

Python | Display text to PyGame window

Pygame is a cross-platform set of Python modules designed for writing video games. It includes computer graphics and sound libraries designed to be used...

6 min read

Pygame - Working with Text

In this article, we will see how to play with texts using the Pygame module. We will be dealing here with initializing the font, rendering the text, editing the text...

5 min read

How to wrap text within Tkinter Text Box?

In this article, we will see that how can we wrap the text in the TKinter Text-Box using the Tkinter module Called textWrap Module. The textwrap module can be...

2 min read

Article Tags : [Python](#) [Python-PyGame](#)

Practice Tags : [python](#)



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



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

DevOps

Git
Linux
AWS
Docker
Kubernetes

Engineering Maths
Software Development
Software Testing

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

@GeeksforGeeks, Sanchhaya Education Private Limited, All rights reserved