



---

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

---

# Pygame – Input Handling

Last Updated : 03 Mar, 2022

---

**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 with the Python programming language.

The **sys** module in Python provides various functions and variables that are used to manipulate different parts of the Python runtime environment. It allows operating on the interpreter as it provides access to the variables and functions that interact strongly with the interpreter.

## Handling Keyboards Inputs

Basic steps to handle keyboard input:

- Import required libraries.
- Create a display surface object using `display.set_mode()` method of pygame.
- Load the image/object.
- Create a click event i.e., KEYDOWN
- Define all the events keys and perform task.
- Create a pause event i.e., KEYUP
- Copying the Text surface object to the display surface object using `blit()` method of pygame display surface object.
- Show the display surface object on the pygame window using the `display.update()` method of pygame.

Example:

---

## Python3

```
# importing all the required libraries
import pygame
from pygame.locals import *
from sys import exit
```

```
# initiating pygame library to use it's
# functions
pygame.init()

# declaring windows/surface width and height
size = width, height = 740, 480
screen = pygame.display.set_mode(size)

# loads a new image from a file and convert()
# will create a copy of image on surface
img = pygame.image.load("char.png").convert()

# declaring value to variables
x, y = 0, 0
move_x, move_y = 0, 0

while True:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            # pygame.QUIT deactivates pygame
            exit()
            # exit() is sys function used to
            # kill the program

        # KEYDOWN event will be triggered everytime
        # we press a button
        if event.type == KEYDOWN:

            if event.key == K_LEFT:
                move_x = -0.3 # object moves -0.3 towards x axis
                print("pressed LEFT")
            elif event.key == K_RIGHT:
                move_x = +0.3 # object moves 0.3 towards x axis
                print("pressed RIGHT")
            elif event.key == K_UP:
                move_y = -0.3 # object moves -0.3 towards y axis
                print("pressed UP")
            elif event.key == K_DOWN:
                move_y = +0.3 # object moves 0.3 towards y axis
                print("pressed DOWN")

        # K_LCTRL event will be triggered everytime we
        # press left CTRL button
        elif event.key == K_LCTRL:

            # declaring new image file to update image
            # everytime left CTRL is pressed
            img = pygame.image.load("char1.png")
            pygame.display.update() # update image
```

```

elif event.key == K_BACKSPACE:

    # this the default file we declared in start
    # and it will restore it everytime we press
    # backspace
    img = pygame.image.load("char.png")
    pygame.display.update() # update image

# it will get triggered when left key is released
if event.type == KEYUP:
    if event.key == K_LEFT:
        move_x = 0 # movement stops
    elif event.key == K_RIGHT:
        move_x = 0 # movement stops
    elif event.key == K_UP:
        move_y = 0 # movement stops
    elif event.key == K_DOWN:
        move_y = 0 # movement stops
    """KEYUP event will be triggered when the release the keys
    and x,y coordinates will not change anymore"""

x += move_x
y += move_y
# updating coordinate values of x,y
screen.fill((255, 255, 255))

# the function will fill the background with white color
screen.blit(img, (x, y))

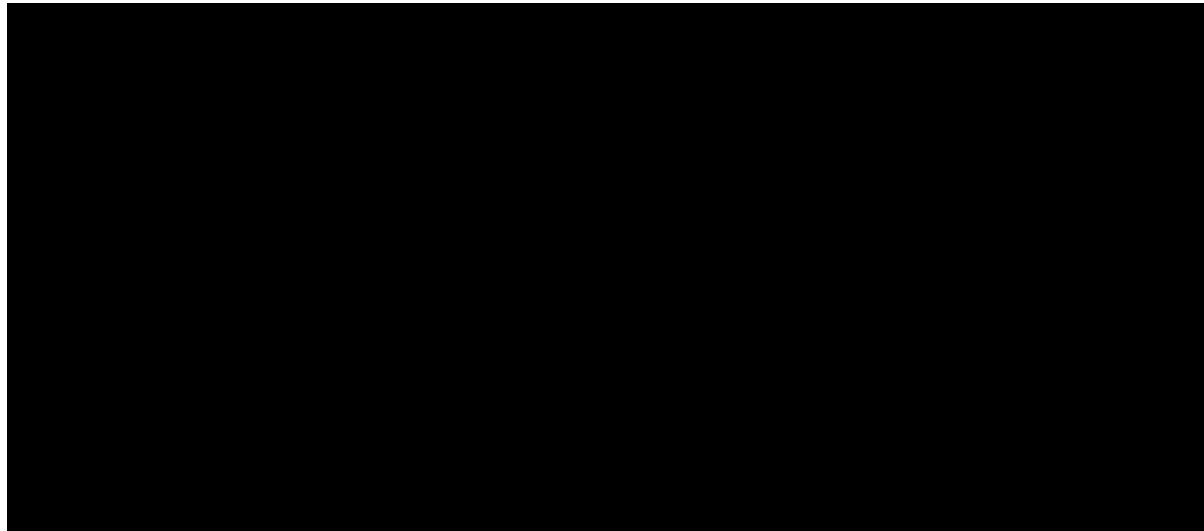
# blit() function will copy image file to x,y coordinates.
pygame.display.update()
# draw the objects on screen

```

## Output:

00:00

00:10



me.

method of pygame display surface object.

- Show the display surface object on the pygame window using the display.update() method of pygame.

### Example:

---

## Python3

```
# importing all the required libraries
import pygame
from pygame.locals import *
from sys import exit

# initiating pygame library to use it's functions
pygame.init()

# declaring windows/surface width and height
size = width, height = 740, 480
screen = pygame.display.set_mode(size)

# loads a new image from a file and convert()
# will create a copy of image on surface
img = pygame.image.load("char.png").convert()

# declaring value to variables
clicking = False
right_clicking = False
middle_click = False
```

```
while True:
    mx, my = pygame.mouse.get_pos() # gets mouse x,y coordinates
    location = [mx, my]
    for event in pygame.event.get():
        if event.type == pygame.QUIT:

            # pygame.QUIT deactivates pygame
            exit()
            # exit() is sys function used to kill the program

    # MOUSEBUTTONDOWN event is triggered when a button is pressed
    if event.type == pygame.MOUSEBUTTONDOWN:

        # returns true when mouse left button is clicked
        if event.button == 1:
            clicking = True

            # declaring new image file to update image
            # everytime left button clicking is true
            img = pygame.image.load("char1.png")
            pygame.display.update() # update image

        # returns true when mouse right button is clicked
        if event.button == 3:
            right_clicking = True

            # declaring new image file to update image
            # everytime right button is clicked
            img = pygame.image.load("char.png")
            pygame.display.update() # update image

        # returns true when mouse middle button is clicked
        if event.button == 2:
            middle_click = middle_click

            # rescale image when middle button clicking is true
            img = pygame.transform.scale(img, (100, 100))
            pygame.display.update() # update image

    # MOUSEBUTTONUP is triggered when mouse button
    # is released(not clicked)
    if event.type == pygame.MOUSEBUTTONUP:
        if event.button == 1:
            clicking = False

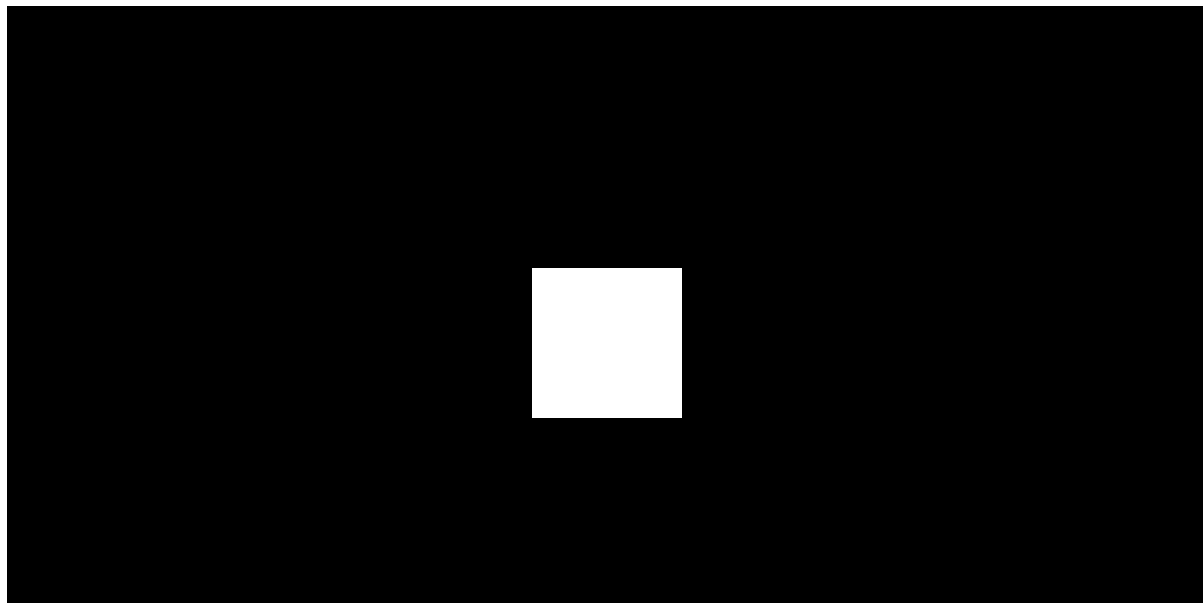
    screen.fill((255, 255, 255))

    # the function will fill the background
```

```
# with white color
screen.blit(img, (location[0], location[1]))

# blit() function will copy image file
# to x,y coordinates.
pygame.display.update()
# draw the objects on screen
```

## Output:



00:00

00:11



vina...



14

## Next Article

[Pygame - Event Handling](#)

## 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

## Pygame - Event Handling

An event is an action that is performed by the user in order to get the desired result. For instance, if a user clicks a button then it is known as a click event. No...

6 min read

## How to create a text input box with Pygame?

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 c...

3 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

## Python VLC MediaPlayer – Enabling Mouse Input Handling

In this article we will see how we can enable mouse input handling the MediaPlayer object in the python vlc module. VLC media player is a free and...

2 min read

## Python Arcade - Handling Keyboard Input

In this article, we will discuss how to handle keyboard inputs in Python arcade module. In Arcade, you can easily check which keyboard button is pressed and...

4 min read

## How to add moving platforms in PyGame

Prerequisite: Drawing in Pygame In this article, we will learn how we can add moving platforms to our game using PyGame in Python. Creating a Platform We...

5 min read

## Pygame - Surface

When using Pygame, surfaces are generally used to represent the appearance of the object and its position on the screen. All the objects, text, images that we...

6 min read

## Python | Display images with PyGame

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...

2 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

### Article Tags :

[Geeks Premier League](#)[Python](#)[Geeks-Premier-League-2022](#)[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](#)

### Languages

[Python](#)[Java](#)[C++](#)[PHP](#)[GoLang](#)[SQL](#)[R Language](#)



[GeeksforGeeks Community](#)[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](#)

## Web Technologies

[HTML](#)[CSS](#)[JavaScript](#)[TypeScript](#)[ReactJS](#)[NextJS](#)[Bootstrap](#)[Web Design](#)

## Computer Science

[Operating Systems](#)[Computer Network](#)[Database Management System](#)[Software Engineering](#)[Digital Logic Design](#)[Engineering Maths](#)[Software Development](#)[Software Testing](#)

## System Design

[High Level Design](#)[Low Level Design](#)[UML Diagrams](#)[Interview Guide](#)[Design Patterns](#)[OOAD](#)[System Design Bootcamp](#)[Interview Questions](#)

## School Subjects

[Mathematics](#)[Physics](#)[Chemistry](#)[Biology](#)

## Data Science & ML

[Data Science With Python](#)[Data Science For Beginner](#)[Machine Learning](#)[ML Maths](#)[Data Visualisation](#)[Pandas](#)[NumPy](#)[NLP](#)[Deep Learning](#)

## Python Tutorial

[Python Programming Examples](#)[Python Projects](#)[Python Tkinter](#)[Web Scraping](#)[OpenCV Tutorial](#)[Python Interview Question](#)[Django](#)

## DevOps

[Git](#)[Linux](#)[AWS](#)[Docker](#)[Kubernetes](#)[Azure](#)[GCP](#)[DevOps Roadmap](#)

## Interview Preparation

[Competitive Programming](#)[Top DS or Algo for CP](#)[Company-Wise Recruitment Process](#)[Company-Wise Preparation](#)[Aptitude Preparation](#)[Puzzles](#)

## GeeksforGeeks Videos

[DSA](#)[Python](#)[Java](#)[C++](#)

Social Science	Web Development
English Grammar	Data Science
Commerce	CS Subjects
World GK	

@GeeksforGeeks, Sanchhaya Education Private Limited, All rights reserved