

### Import & Initialization

```
import pygame  
pygame.init()
```

This snippet of code imports pygame module and by using init function you can initialize pygame so that you can start rocking with game dev!

### Create screen

```
pygame.display.set_mode((width,  
height))
```

Creates a window for your game, its similar to a canvas and it returns a surface. The arguments are width and height of the screen as a tuple

### Set Title

```
display.set_caption('Title of  
the window')
```

This function simply sets the argument as the title of the window.

### Update Display

```
pygame.display.update()
```

Updates the screen, basically redraws the main surface if arguments are not specified. And on the other hand, if you do happen to specify the arguments, it redraws the portions that you gave it.

### Color

```
pygame.Color(R, G, B)
```

Creates a color object with RGBA as arguments.

### fill Function

```
Surface.fill(color)
```

This function is used to fill a solid color onto your screen. Arguments should be RGBA(Red, Green, Blue, Alpha) values.

### Set Font

```
pygame.font.SysFont('Font Name',  
FontSize)
```

This function lets you choose a font for your text that appears on the screen. It takes Font name and size as its arguments and it returns a font object

### blit Function

```
Surface.blit(source, dest, area,  
speical_flags)
```

Draws one image onto another. Basically, it copies the pixels from one surface to another. It can be used to draw images to the screen.

### Time

```
pygame.ti-    Creates a clock object and  
me.Clock()    you can control the clock  
               using tick() function
```

```
pygame.ti-    Returns the clock framerate  
me.Clo-  
ck.get-  
_fps()
```

```
pygame.ti-    Returns the time used in  
me.Clo-    previous tick  
ck.get-  
_time()
```

```
pygame.ti-    Pause for time specified  
me.delay()
```

### Common Event Loop

```
for event in pygame.event.get():  
    if event.type ==  
        pygame.QUIT:  
            pygame.quit()
```

One of the most common ways of event handling, Its a loop which constantly checks for events, quits if the QUIT event is triggered and prevents your game from freezing.

### Event

```
pygame.ev-    Places a new event that you  
ent.post()    specify on the queue
```

```
pygame.ev-    Creates a new event object  
ent.Event()
```

```
pygame.ev-    Gets the event from the  
ent.get()    queue
```

```
pygame.ev-    removes all the events from  
ent.clear()    the queue
```

Events are always in a queue. Order of events does matter.

### Images

```
pygame.im-    Loads a new image from a file  
age.load()    that you specify
```

```
pygame.im-    You can save the image to  
age.save()    your drive using this function
```

### Audio

```
pygame.mi-    Initializes the mixer  
xer.init()    module
```

```
pygame.mi-    Loads the music file you  
xer.mu-    specify as the argument  
sic.load()
```

```
pygame.mi-    Plays the sound  
xer.mu-  
sic.play()
```

```
pygame.mi-    Stops the music from  
xer.mu-    being played  
sic.stop()
```

```
pygame.mi-    UnInitializes the mixer  
xer.quit()    module
```

### Exit

```
pygame.quit()    Quits the game
```



By **rootix**  
[cheatography.com/rootix/](https://cheatography.com/rootix/)

Published 4th November, 2017.

Last updated 4th November, 2017.

Page 1 of 1.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish Yours!

<https://apollopad.com>