

# pygame

José Humberto Abril García

[jose.abril@unison.mx](mailto:jose.abril@unison.mx)

662 417 3146



[/humabril](https://twitter.com/humabril)




[/humabril](https://www.instagram.com/humabril)

# pygame

pygame.org/news

← → ↻ https://www.pygame.org/news

pip install  Projects ▾ News About Getting Started Docs Info ▾ Development ▾

## News

New here?

pygame 2.2.0 - 🐛 — 28 Feb, 2023

```
python -m pip install -U pygame==2.2.0 --user
```

[release notes](#)

Please file an issue if you notice a problem:  
<https://github.com/pygame/pygame/issues>

pygame 2.1.3 - make it count — 14 Feb, 2023

```
python -m pip install -U pygame==2.1.3 --user
```

[release notes](#)

Spanish translation update — 3 Feb, 2023

We are making progress on the Spanish translation of pygame materials. Estefania (who is also a teacher) has in addition been providing feedback on the English versions of documents. The workflow is still improving, and we hope to make progress on fine tuning this process as we go through more of the Spanish translations.


pygame in all the languages — 16 May 2022

New members signup

Log In


## Recent Releases

27 Feb, 2023




MetalArbiter - .2

3 Feb, 2023



\$STONKS simulator - 1.2

23 Jan, 2023



Backpack: 145/250  
Sell 0/100  
Shop  
To Surface

# \$ pip list

```
|                                     % pip3 list
Package                             Version
-----
certifi                             2022.9.24
charset-normalizer                   2.1.1
click                                8.1.3
Flask                                 2.1.2
idna                                  3.4
importlib-metadata                   4.11.3
itsdangerous                         2.1.2
Jinja2                               3.1.2
MarkupSafe                           2.1.1
Pillow                               9.1.0
pip                                  22.0.4
pygame                               2.1.2
pyserial                             3.5
pyusb                                 1.2.1
requests                             2.28.1
setuptools                           60.5.0
tk                                    0.1.0
urllib3                              1.26.12
Werkzeug                             2.1.2
wheel                                 0.37.1
zipp                                  3.8.0
```

WARNING: You are using pip version 22.0.4; however, version 23.0.1 is available.

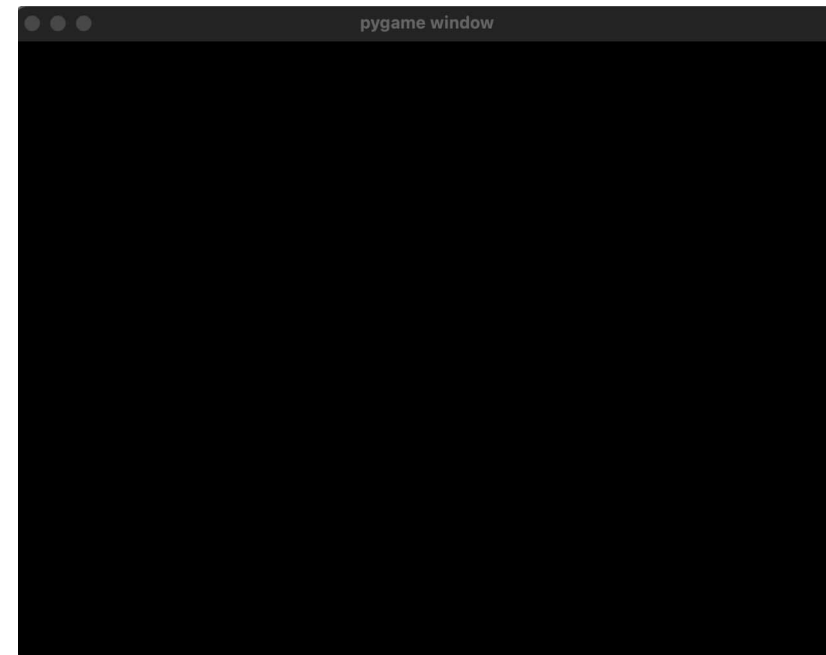
You should consider upgrading via the '/opt/homebrew/opt/python@3.9/bin/python3.9 -m pip install --upgrade pip' command.

```
% █
```

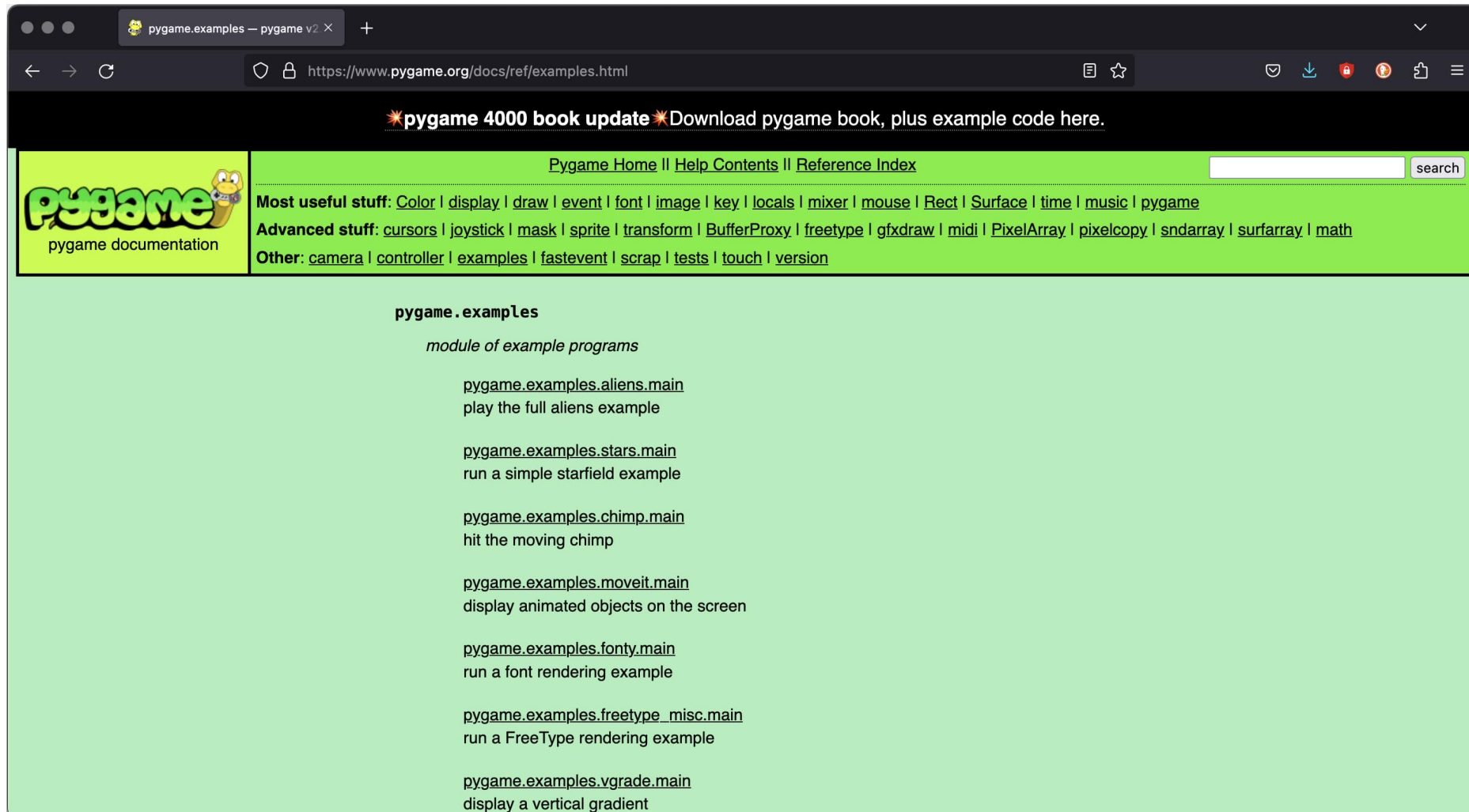
# python3 -m pygame.tests

josehumbertoabrilgarcia — Python -m pygame.tests — 80x24

```
loading pygame.tests.camera_test
loading pygame.tests.color_test
loading pygame.tests.constants_test
loading pygame.tests.controller_test
loading pygame.tests.cursors_test
loading pygame.tests.display_test
loading pygame.tests.docs_test
loading pygame.tests.draw_test
loading pygame.tests.event_test
loading pygame.tests.font_test
loading pygame.tests.freetype_test
loading pygame.tests.ftfont_test
loading pygame.tests.gfxdraw_test
loading pygame.tests.image__save_gl_surface_test
loading pygame.tests.image_test
loading pygame.tests.imageext_test
loading pygame.tests.joystick_test
loading pygame.tests.key_test
loading pygame.tests.mask_test
loading pygame.tests.math_test
loading pygame.tests.midi_test
loading pygame.tests.mixer_music_test
loading pygame.tests.mixer_test
```



# pygame examples



The screenshot shows a web browser window with the URL `https://www.pygame.org/docs/ref/examples.html`. The page has a black header with the text "pygame 4000 book update" and "Download pygame book, plus example code here." Below the header is a green navigation bar with the Pygame logo on the left and a search bar on the right. The navigation bar contains links for "Pygame Home", "Help Contents", and "Reference Index". Below the navigation bar, there are three sections of links: "Most useful stuff" (Color, display, draw, event, font, image, key, locals, mixer, mouse, Rect, Surface, time, music, pygame), "Advanced stuff" (cursors, joystick, mask, sprite, transform, BufferProxy, freetype, gfxdraw, midi, PixelArray, pixelcopy, sndarray, surfarray, math), and "Other" (camera, controller, examples, fastevent, scrap, tests, touch, version). The main content area has a light green background and is titled "pygame.examples". It lists several example programs with their module names and descriptions: "pygame.examples.aliens.main" (play the full aliens example), "pygame.examples.stars.main" (run a simple starfield example), "pygame.examples.chimp.main" (hit the moving chimp), "pygame.examples.moveit.main" (display animated objects on the screen), "pygame.examples.fonty.main" (run a font rendering example), "pygame.examples.freetype\_misc.main" (run a FreeType rendering example), and "pygame.examples.vgrade.main" (display a vertical gradient).

pygame.examples

*module of example programs*

`pygame.examples.aliens.main`  
play the full aliens example

`pygame.examples.stars.main`  
run a simple starfield example

`pygame.examples.chimp.main`  
hit the moving chimp

`pygame.examples.moveit.main`  
display animated objects on the screen

`pygame.examples.fonty.main`  
run a font rendering example

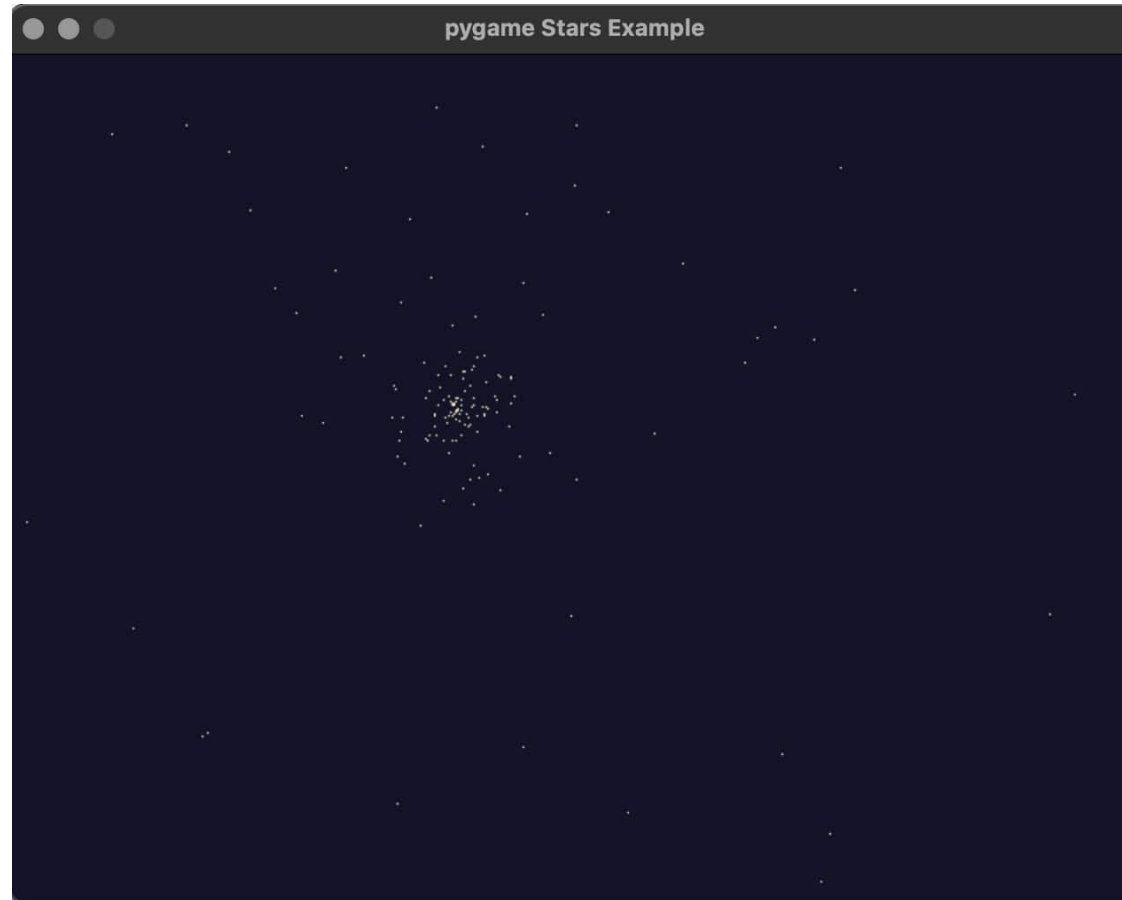
`pygame.examples.freetype_misc.main`  
run a FreeType rendering example

`pygame.examples.vgrade.main`  
display a vertical gradient

```
python3 -m pygame.examples.aliens
```



```
python3 -m pygame.examples.stars
```



```
python3 -m pygame.examples.chimp
```





# Paso 1: cargar imagen de fondo y cerrar ventana.

control.py ×

```
1  import pygame
2
3  pygame.init()
4
5  clk = pygame.time.Clock()
6
7  size = width, height = 389, 187
8  screen = pygame.display.set_mode(size)
9  background_image = pygame.image.load('1.png').convert()
10 frameRect = pygame.Rect((0, 0), (width, height))
11
12 while True:
13     pygame.event.pump()
14
15     screen.blit(background_image, (0, 0))
16
17     for events in pygame.event.get():
18         if events.type == pygame.QUIT:
19             pygame.quit()
20             exit()
21
```

# Paso 1: cargar imagen de fondo y cerrar ventana.

```
control.py ×  
11  
12 while True:  
13  
14     pygame.event.pump()  
15  
16     screen.blit(background_image, (0, 0))  
17  
18     for events in pygame.event.get():  
19         if events.type == pygame.QUIT:  
20             pygame.quit()  
21             exit()  
22  
23     pygame.display.flip()  
24  
25     clk.tick(40)
```

Paso 1: cargar imagen de fondo y cerrar ventana.



## Paso 2: identificar botones.

 control.py ×

```
15
16     screen.blit(background_image, (0, 0))
17
18     Keys=pygame.key.get_pressed()
19
20     if Keys[pygame.K_x]: print("X")
21
22     if Keys[pygame.K_a]: print("A")
23
24     for events in pygame.event.get():
25         if events.type == pygame.QUIT:
26             pygame.quit()
27             exit()
```

## Paso 3: identificar flechas.

control.py ×

```
19
20     if Keys[pygame.K_x]: print("X")
21
22     if Keys[pygame.K_a]: print("A")
23
24     x = "LEFT" if Keys[pygame.K_LEFT] else "RIGHT" if Keys[pygame.K_RIGHT] else ""
25
26     y = "UP" if Keys[pygame.K_UP] else "DOWN" if Keys[pygame.K_DOWN] else ""
27
28     print(x)
29
30     print(y)
```

control.py ×

```
15
16     screen.blit(background_image, (0, 0))
17
18     Keys=pygame.key.get_pressed()
19
20     if Keys[pygame.K_x]: print("X")
21
22     if Keys[pygame.K_a]: print("A")
23
24     x = "LEFT" if Keys[pygame.K_LEFT] else "RIGHT" if Keys[pygame.K_RIGHT] else ""
25
26     y = "UP" if Keys[pygame.K_UP] else "DOWN" if Keys[pygame.K_DOWN] else ""
27
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

Python + ▾ ... ^ ×

```
A
A
X
X
X
X
█
```

## Paso 3: pygame.Surface

```
12 crosshair = pygame.surface.Surface((10, 10))
13 pygame.draw.circle(crosshair, pygame.Color("black"), (5,5), 5, 0)
14
15 crosshairb = pygame.surface.Surface((10, 10))
16 pygame.draw.circle(crosshairb,pygame.Color("red"), (5,5), 5, 0)
17
18 while True:
```

## Paso 4: blit()

```
25  
26     if Keys[pygame.K_x]: screen.blit(crosshair, (298, 70) )  
27  
28     if Keys[pygame.K_a]: screen.blit(crosshair, (335, 98) )  
29
```



## Paso 5: blit()

```
39     screen.blit(crosshairb, ((x*20)+80-5, (y*20)+105-5))
40
41     pygame.display.flip()
42
43     clk.tick(40)
```

# Resultado final.

