

Ryomacob21083 / ProjExD Public

[Code](#) [Issues 2](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#)

main ▾

...

[ProjExD](#) / [ex04](#) / [dodge_bomb.py](#) / [Jump to ▾](#)

Ryomacob21083 空行の削除 #9

[History](#)

1 contributor

101 lines (85 sloc) 3 KB

...

```
1 import pygame as pg
2 import sys
3 from random import randint
4 import tkinter as tk
5 import tkinter.messagebox as tkm
6
7
8 def check_bound(obj_rct, scr_rct):
9     #obj_rct: こうか とんrect または 爆弾rect
10    #scr_rct: スクリーンrect
11
12    yoko, tate = +1, +1
13    if obj_rct.left < scr_rct.left or scr_rct.right < obj_rct.right:
14        yoko = -1
15    if obj_rct.top < scr_rct.top or scr_rct.bottom < obj_rct.bottom:
16        tate = -1
17    return yoko, tate
18
19
20 def main():
21     root = tk.Tk()
22     root.withdraw()
23     pg.display.set_caption("逃げろ！こうか とん")    ##タイトルバーに「初めての…」を表示
24     scrn_sfc = pg.display.set_mode((1600, 900)) ## 800×600の画面surfaceを生成
25     scrn_rct = scrn_sfc.get_rect()
26     bg_sfc = pg.image.load("fig/pg_bg.jpg")
27     bg_rct = bg_sfc.get_rect()
28
29     tori_sfc = pg.image.load("fig/6.png")
30     tori_sfc = pg.transform.rotozoom(tori_sfc, 0, 2.0)
31     tori_rct = tori_sfc.get_rect()
32     tori_rct.center = 900, 400
33
34     bomb_sfc = pg.Surface((20, 20))
35     bomb_sfc.set_colorkey((0, 0, 0))
36     pg.draw.circle(bomb_sfc, (255, 0, 0), (10, 10), 10)
37     bomb_rct = bomb_sfc.get_rect()
38     bomb_rct.centerx = randint(0, scrn_rct.width)
```

```
39     bomb_rct.centery = randint(0, scrn_rct.height)
40
41     vx, vy = +1, +1
42     clock = pg.time.Clock()
43     while True:
44         scrn_sfc.blit(bg_sfc, bg_rct)
45         for event in pg.event.get():
46             if event.type == pg.QUIT:
47                 return
48
49         key_states = pg.key.get_pressed()
50         if key_states[pg.K_UP]:
51             tori_rct.centery -= 5
52         if key_states[pg.K_DOWN]:
53             tori_rct.centery += 5
54         if key_states[pg.K_LEFT]:
55             tori_rct.centerx -= 5
56         if key_states[pg.K_RIGHT]:
57             tori_rct.centerx += 5
58         yoko, tate = check_bound(tori_rct, scrn_rct)
59         if yoko == -1:
60             if key_states[pg.K_LEFT]:
61                 tori_rct.centerx += 5
62             if key_states[pg.K_RIGHT]:
63                 tori_rct.centerx -= 5
64         if tate == -1:
65             if key_states[pg.K_UP]:
66                 tori_rct.centery += 5
67             if key_states[pg.K_DOWN]:
68                 tori_rct.centery -= 5
69
70         scrn_sfc.blit(tori_sfc, tori_rct)
71         yoko, tate = check_bound(bomb_rct, scrn_rct)
72         vx *= yoko
73         vy *= tate
74
75         if vx < 0:
76             vx -= 0.001
77         else:
78             vx += 0.001
79         if vy < 0:
80             vy -= 0.001
81         else:
82             vy += 0.001
83
84         bomb_rct.move_ip(vx, vy)
85         scrn_sfc.blit(bomb_sfc, bomb_rct)
86
87         if tori_rct.colliderect(bomb_rct):
88             tkm.showinfo("GAME OVER", "また挑戦してね")
89             return
90         if pg.time.get_ticks() >= 3000:
91             tkm.showinfo("GAME CREAR", "おめでとう！！")
92
93         pg.display.update()
94         clock.tick(1000)
95
96
```

```
97  if __name__ == "__main__":
98      pg.init()          ##モジュールを初期化
99      main()
100     pg.quit()           ##モジュールの初期化を解除
101     sys.exit()          ##プログラムを終了
```