


 c0a22155e2 / ProjExD_05



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

ProjExD_05 / tyari.py 

c0a22155e2 反転モード完了

21 minutes ago



79 lines (70 loc) · 2.52 KB

Code Blame Raw Copy Download Edit More

```
1  import sys
2  import random
3  import pygame as pg
4
5  WIDTH = 1297
6  HEIGHT = 744
7
8
9  class tyari():
10
11  def __init__(self,num: int ,xy:tuple[int,int]):
12      img0 = pg.transform.rotozoom(pg.image.load(f"ex05/figs/{num}.png"), 0, 1.0)
13      img = pg.transform.flip(img0, True, False) # デフォルトの自転車 (右向き)
14      self.imgs = { # 0度から反時計回りに定義
15          0: img, # 右
16          1: pg.transform.rotozoom(img, 45, 1.0), # 右上
17          -1: pg.transform.rotozoom(img, -45, 1.0), # 右下
18      }
19      self.img = self.imgs[0]
20      self.rct = self.img.get_rect()
21      self.rct.center = xy
22
23  def change_img(self, num: int, screen: pg.Surface):
24      self.img = pg.transform.rotozoom(pg.image.load(f"ex05/fig/{num}.png"), 0, 2.0)
25      screen.blit(self.img, self.rct)
26
27  def update(self, screen: pg.Surface):
28      self.rct.move_ip(0,0)#自転車を描画
29      screen.blit(self.img, self.rct)
30
31
32  def main():
33      pg.display.set_caption("チャリ走DX")
34      screen = pg.display.set_mode((WIDTH, HEIGHT))#スクリーンを描画
35      clock = pg.time.Clock()
36      bg_img = pg.image.load("ex05/figs/bg.png")
37      bg_img2 = pg.transform.flip(bg_img, True, False)
38      bird = tyari(1,(200,HEIGHT *0.68))#自転車を描画
39      reverse = False#反転
40      tmr = 0
41      x = tmr
42      while True:
43          for event in pg.event.get():
44              if event.type == pg.QUIT: return
45              if not reverse:#通常状態
46                  screen.blit(bg_img, [-2594-x, 0])
47                  screen.blit(bg_img2, [-1297-x, 0])
48                  screen.blit(self.img, [0-x, 0])
```

```
48     screen.blit(bg_img, [0, 0])
49     screen.blit(bg_img2, [1297-x, 0])
50     screen.blit(bg_img, [2594-x, 0])
51     x += 5
52     if x > 2594:
53         x = 0
54     else: #反転状態
55         screen.blit(bg_img, [2594-x, 0])
56         screen.blit(bg_img2, [1297-x, 0])
57         screen.blit(bg_img, [0-x, 0])
58         screen.blit(bg_img2, [-1297-x, 0])
59         screen.blit(bg_img, [-2594-x, 0])
60         x -= 5
61         if x < -2594:
62             x = 0
63     bird.update(screen)
64     pg.draw.rect(screen, (255, 255, 255), (0, HEIGHT*0.8, WIDTH, HEIGHT)) #じめんを描写
65     pg.display.update()
66     tmr += 5
67     clock.tick(1000)
68     for event in pg.event.get():
69         if event.type == pg.KEYDOWN and event.key == pg.K_SPACE: #スペースで反転
70             if reverse:
71                 reverse = False
72             else:
73                 reverse = True
74
75 if __name__ == "__main__":
76     pg.init()
77     main()
78     pg.quit()
79     sys.exit()
```