


 c0c24020ae / ProjExD_Group05



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

 177e18b ▾ ProjExD_Group05 / README.md 

Go to file t ...

 c0a24017 微修正

177e18b · last week 

39 lines (33 loc) · 1.12 KB

Preview Code Blame

Raw     

スーパーこうかтонブラザーズ

実行環境の必要条件

- python >= 3.10
- pygame >= 2.1

ゲームの概要

- 主人公のこうかтонがさらわれたこうか姫を助けるゲーム

ゲームの遊び方

- 十字キーでキャラを操作する
- スペースキーでジャンプ

ゲームの実装

共通基本機能



- 背景画像と主人公キャラクターの描画
- 背景がキャラとともに動く
- 十字キーでキャラを動かす
- 地形を作る
- ゴールでゲーム終了
- 地形に乗ることができる
- ジャンプできる

分担追加機能




- 敵を踏んで倒せる (り)
- 穴に落ちたら死ぬ (り)
- 敵の出現 (加藤)
- 敵が壁に反射 (加藤)
- ダッシュ (河辺)
- 敵に当たったら死ぬ (河辺)
- ブロックを叩ける(体が大きければ壊せる) (束田)
- アイテムの出現 (束田)
- 食べると体が大きくなるアイテム (波部)








ToDo


- キャラクターの変数名の統一
- 敵キャラの変数名決め
- 登場キャラの決定
- アイテム決定




c0c24020ae /
ProjExD_Group05





<> Code  Pull requests  Actions  Projects  Wiki  Security  Insights  Settings



 177e18b ▾

ProjExD_Group05 / super_koukaton.py



 Go to file

t

...






 c0a24017 微修正

177e18b · last week



120 lines (99 loc) · 3.45 KB

Code Blame

Raw     

```
1  import pygame
2
3  # 画面のサイズ
4  SCREEN_WIDTH = 1100
5  SCREEN_HEIGHT = 700
6  FPS = 60
7
8  # 色
9  BLACK = (0, 0, 0)
10 WHITE = (255, 255, 255)
11 RED = (255, 0, 0)
12 GREEN = (0, 255, 0)
13
14 # ブロックの上端のy座標（背景画像に合わせて調整）
15 GROUND_Y = 610 # 必要に応じて微調整してください
16
17 class Bird(pygame.sprite.Sprite):
18     def __init__(self):
19         super().__init__()
20         self.image = pygame.Surface([30, 50])
21         self.image.fill(RED)
22         self.rect = self.image.get_rect()
23         self.rect.x = 50
24         self.rect.bottom = GROUND_Y # ブロックの上に乗せる
25         self.speed_x = 0
26         self.speed_y = 0
27         self.gravity = 1
28         self.jump_power = -20
29         self.is_jumping = False
30         self.world_x = 50 # ワールド座標
31
32     def update(self):
33         # 左右移動
34         self.world_x += self.speed_x
35         # プレイヤーの画面上のx座標は後で調整
36
37         # 重力
38         self.speed_y += self.gravity
39         self.rect.y += self.speed_y
40
41         # 地面との衝突判定（ブロックの上）
```

```
42         if self.rect.bottom > GROUND_Y:
43             self.rect.bottom = GROUND_Y
44             self.speed_y = 0
45             self.is_jumping = False
46
47     def jump(self):
48         if not self.is_jumping:
49             self.speed_y = self.jump_power
50             self.is_jumping = True
51
52
53     def main():
54         pygame.init()
55         screen = pygame.display.set_mode((SCREEN_WIDTH, SCREEN_HEIGHT))
56         pygame.display.set_caption("スーパーこうかとんブラザーズ")
57         clock = pygame.time.Clock()
58
59         # 背景画像のロード
60         bg_img = pygame.transform.rotozoom(pygame.image.load("ex5/fig/pg_bg.png").convert(), 0, 2.92)
61         bg_width = bg_img.get_width()
62         bg_height = bg_img.get_height()
63
64         all_sprites = pygame.sprite.Group()
65         bird = Bird()
66         all_sprites.add(bird)
67
68         scroll_x = 0 # 背景のスクロール量
69
70         running = True
71         while running:
72             for event in pygame.event.get():
73                 if event.type == pygame.QUIT:
74                     running = False
75                 if event.type == pygame.KEYDOWN:
76                     if event.key == pygame.K_LEFT:
77                         bird.speed_x = -5
78                     if event.key == pygame.K_RIGHT:
79                         bird.speed_x = 5
80                     if event.key == pygame.K_SPACE:
81                         bird.jump()
82                 if event.type == pygame.KEYUP:
83                     if event.key == pygame.K_LEFT and bird.speed_x < 0:
84                         bird.speed_x = 0
85                     if event.key == pygame.K_RIGHT and bird.speed_x > 0:
86                         bird.speed_x = 0
87
88         # ゲームループ
89         all_sprites.update()
90
91         # プレイヤーが画面中央より右に行ったら背景をスクロール
92         center_x = SCREEN_WIDTH // 2
93         if bird.world_x > center_x:
94             scroll_x = bird.world_x - center_x
95         else:
96             scroll_x = 0
97         # 背景の範囲外に行かないように制限
98         max_scroll = bg_width - SCREEN_WIDTH
99         if scroll_x > max_scroll:
```

```
100         scroll_x = max_scroll
101     if scroll_x < 0:
102         scroll_x = 0
103
104     # プレイヤーの画面上のx座標を調整
105     if bird.world_x > center_x:
106         bird.rect.x = center_x
107     else:
108         bird.rect.x = bird.world_x
109
110     # 描画
111     screen.blit(bg_img, (-scroll_x, 0))
112     all_sprites.draw(screen)
113     pygame.display.flip()
114
115     clock.tick(FPS)
116
117     pygame.quit()
118
119 if __name__ == '__main__':
120     main()
```