

```
1 # Kalkulator RGB
2 # https://www.w3schools.com/colors/colors_rgb.asp
3 #~~~~~
4
5 import pygame
6 pygame.init()
7
8 SCREEN_WIDTH = 800
9 SCREEN_HEIGHT = 600
10
11 screen_surface = pygame.display.set_mode((SCREEN_WIDTH, SCREEN_HEIGHT))
12 pygame.display.set_caption("Pierwsza gra")
13
14
15 def load_image(img_path: str, position):
16     image = pygame.image.load(img_path)
17     surface = image.convert()
18     transparent_color = (0,0,0)
19     surface.set_colorkey(transparent_color)
20
21     rect = surface.get_rect(center = position)
22     return [image, surface, rect]
23
24
25 def print_image(img_list: list):
26     image, surface, rect = img_list
27     screen_surface.blit(surface, rect)
28     pass
29
30 # 2. Zamiana pozycji obrazka gracza -----
31 def set_position_image(img_list, position):
32     image, surface, rect = img_list
33     rect = surface.get_rect(center=position)
34     return [image, surface, rect]
35 # -----
36
37 # 4. Sterowanie graczem -----
38 # * współrzędna w pygame
39 def calculate_player_movement(keys):
40     speed = 10
41     delta_x = 0
42     delta_y = 0
43
44     #(na koniec) booster prędkości
45     if keys[pygame.K_LSHIFT]:
46         speed *= 2
47
48     if keys[pygame.K_w]:
49         delta_y -= speed
50     if keys[pygame.K_s]:
51         delta_y += speed
52     if keys[pygame.K_a]:
53         delta_x -= speed
54     if keys[pygame.K_d]:
55         delta_x += speed
56
57     return [delta_x, delta_y]
58 # -----
59
60
61 # 8. Limit wyjeżdżania poza ekran -----
62 def limit_position(position):
63     x, y = position
64     x = max(0, min(x, SCREEN_WIDTH))
65     y = max(0, min(y, SCREEN_HEIGHT))
66     return [x, y]
67 # -----
```

```
69
70 player_pos = [SCREEN_WIDTH // 2, SCREEN_HEIGHT // 2]
71
72 #1. Obrazek gracza -----
73 player_path = r'C:\Users\aplac\Documents\Giganci\Projekty Python\Lekcja 12 i 13 - pierwsza
74 # aplikacja\grafiki\player.png'
75 # dodaj print_image(player) w pętli gry jesli nie mają
76 # TESTY
77 # dodać później w ramach uproszczenia:
78 # import os #importuje moduł os, który pozwala pracować z folderami i plikami
79 # os.chdir(os.path.dirname(__file__)) #zmienia bieżący folder roboczy programu na folder, w którym znajduje się ten
80 # plik .py
81 # # nie trzeba wtedy podawać pełnych ścieżek dostępu do np. plików graficznych, jesli są w tym samym folderze
82 #-----
83 player = load_image(player_path, player_pos)
84
85 # 6. Kolor tła -----
86 background_color = [178, 221, 242]
87 # -----
88
89 clock = pygame.time.Clock()
90 game_status = True
91
92
93 while game_status:
94     events = pygame.event.get()
95
96     for event in events:
97         #print(event)
98         if event.type == pygame.QUIT:
99             game_status = False
100            pass
101
102
103     # 3. Zbieranie wcisniętych klawiszy -----
104     pressed_keys = pygame.key.get_pressed()
105     # -----
106
107     # 5. Reakcja na klawisze -----
108     delta_x, delta_y = calculate_player_movement(pressed_keys)
109
110     player_pos[0] += delta_x
111     player_pos[1] += delta_y
112
113     # 9. -----
114     player_pos = limit_position(player_pos)
115     # -----
116
117     player = set_position_image(player, player_pos)
118     # TESTY
119     # - rysowanie po ekranie
120     # - wyjeżdżanie poza ekran
121     # -----
122
123     # 7. Odświeżanie -----
124     screen_surface.fill(background_color)
125     # TESTY
126     # - ładny kolor tła
127     # - pozostało wyjeżdżanie za ekran
128     # -----
129
130     print_image(player) #1.2 !!!!!!!!
131
132     pygame.display.update()
133     clock.tick(60)
134     pass
135
136     pygame.quit()
137     quit()
```