

Objektovo orientované programovanie *- pokročilí*

Učiteľ:
Ing. Jozef Wagner PhD.

Učebnica:
<https://oop.wagjo.com/>

OPGP

Pokročilí 14

1. Pygame
2. Práca s myšou
3. Vykreslovanie obrázkov a textu
4. Tlačítka

Spracovanie myši

Jednorázové udalosti – **eventy**

Stavy, v ktorých sa myš nachádza (tlačítko je stlačené, myš sa nachádza na pozícii X, Y, ...)

Udalosti si vieme odchytiť pomocou `pygame.event.get()`, tak ako pri klávesnici.

Stav myši vieme zistiť pomocou `pygame.mouse.get_pressed()` alebo `pygame.mouse.get_pos()`.

Spracovanie myši

```
for event in pygame.event.get():
    if event.type == pygame.MOUSEBUTTONDOWN:
        print("Klik!", event.button, "na pozícii", event.pos)
    if event.type == pygame.MOUSEBUTTONUP:
        print("Myš pustená!", event.button)
    if event.type == pygame.MOUSEWHEEL:
        print("Koliesko otočené:", event.y)
# Aktuálna pozícia myši (každý frame)
mys_x, mys_y = pygame.mouse.get_pos()
# Ktoré tlačidlá sú práve stlačené (ľavé, stredné, pravé)
tlacidla = pygame.mouse.get_pressed()
# tlacidla je tuple: (ľavé, stredné, pravé)
if tlacidla[0]:
    print(f"Držím ľavé tlačidlo na [{mys_x}, {mys_y}]")
```

Tlačítka

V Pygame vieme použiť **Rect** ako jednoduché tlačítka - buttons, na ovládanie aplikácie. Postup je nasledovný:

- Vykreslíme **Rect** na pozícii, kde sa má tlačítko nachádzať
- Vypíšeme label tlačítka
- V hernej slučke budeme sledovať kliknutia myšou. Ak bolo kliknuté na pozícii tlačítka, vykonáme akciu

Tlačítka

```
button_rect = pygame.Rect(POZICIA)
button_label = "..."
```

```
pygame.draw.rect(screen, FARBA_POZADIA, button_rect)
btn_text = font.render(button_label, True, FARBA_TEXTU)
screen.blit(btn_text, POZICIA)
```

```
...
```

```
if event.type == pygame.MOUSEBUTTONDOWN:
if event.button == 1: # Left click
    if button_rect.collidepoint(event.pos):
        button_action()
```

Vykreslovanie obrázku

Modul `pygame.image`. Podporuje štandardné formáty ako GIF, JPEG, PNG a WEBP

Obrázok si viem načítať pomocou metódy

```
image = pygame.image.load("filename")
```

Po načítaní vieme zmeniť veľkosť obrázka pomocou

```
image = pygame.transform.scale(image, (sirka, vyska))
```

Samotné vykreslenie obrázku na obrazovku sa realizuje pomocou

```
screen.blit(image, rect)
```

Nie je dobré obrázok načítavať každý frame. Obrázok načítame raz a pamäťame si načítané dátu. Vo frame sa už obrázok iba vykreslí.

Hra Pong

Dnes si pre našu adventúru vytvoríme grafické rozhranie.
Postup je nasledovný:

- Do `assets` pridáme novú mapu a sadu obrázkov miestností
- Upravíme triedu `Room` o nový atribút s obrázkom
- Upravíme akciu `use`, aby sa v miestnosti mohol meniť obrázok
- Vytvoríme novú triedu `PGameEngine` s grafickým enginom