

Kurs: Pajton - programiranje igara sa Pygame bibliotekom

017_018 čas: elementi_animacije

Teme: učitavanje slike, program Logo_na_sredini.py, uvod u animaciju, načini realizacije animacije, pomeranje po jednoj osi, program Levo_desno.py

001 Učitavanje slika

Modul `pygame.image` sadrži funkcije koje se koriste za učitavanje i čuvanje slika kao i za transfer Surface u formate koje mogu koristiti drugi paketi.

Kada se slika (image) učita, ona se učita kao Surface objekat.

Funkcija `pygame.image.load()` podržava sledeće formate za učitavanje u pygame: jpg, png, bmp, tga.

Čuvanje slika se može izvršiti u sledećim formatima: bmp, tga, png, jpeg.

Funkcija `pygame.image.load(naziv_fajla)` čini učitavanje slike iz izvornog fajla.

Učitana slika se smešta u isti direktorijum u kojem je i solution Pajton projekta.

Funkcija `pygame.image.load(fajl_objekat)` čini učitavanje slike iz nekog Pajton objekta.

Posle učitavanja pygame automatski određuje tip slike i kreira nov Surface objekat.

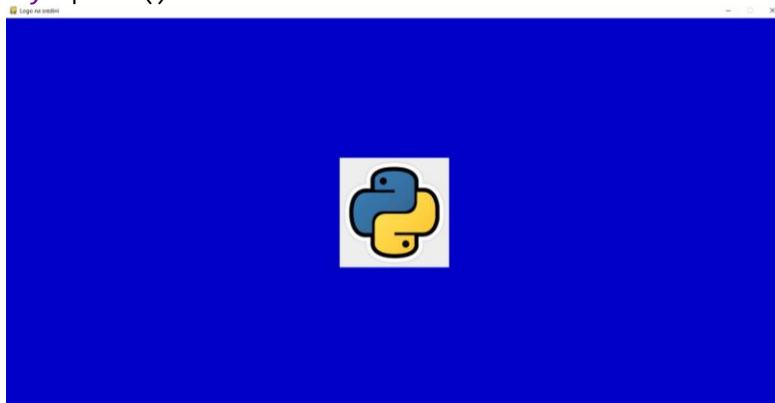
Dobijeni Surface objekat će imati iste format boje i alfa transparentnosti kao i fajl iz kojeg je učitana.

Korišćenjem `Surface.convert()` se kreira kopija koja će se brže iscrtavati na ekranu.

Logo_na_sredinu.py

```
import pygame, sys
from pygame.locals import *
pygame.init()
ekran = pygame.display.set_mode((1600, 800))
pygame.display.set_caption("Logo na sredinu")
plava = 0, 0, 200
sirina = 1600
visina = 800
while True:
    for dogadjaj in pygame.event.get():
        if dogadjaj.type in (QUIT, KEYDOWN):
            sys.exit()

    ekran.fill(plava)
    logo = pygame.image.load("python02.png")
    (x, y) = ((sirina - logo.get_width()) / 2, (visina - logo.get_height()) / 2)
    ekran.blit(logo, (x, y))
    pygame.display.update()
```



Pošto funkcija za prikaz slike na prozoru, `ekran.blit()`, ima parametre gornjeg levog ugla slike, zato je neophodno postaviti pravi centar slike u pravi centar ekrana.

Pomoću funkcija `get_width()` i `get_height()` se dobijaju prave dimenzije slike.

002 Uvod u animaciju

Animacije predstavljaju promenu oblika ili pozicije slike na ekranu računara.

Uobičajeno je u procesorski nezahtevnim kompjuterskim igrama da se slika menja 20-30 puta u sekundi i da se promena slike dešava u pravilnim vremenskim intervalima (npr, na svakih 50 milisekundi).

Svaka slika koja se koristi kao deo animacije se naziva frejm ili okvir (frame).

003 Načini realizacije animacije

Postoje dva načina realizacije animacije u kompjuterskim igrama.

Prvi način je rad glavne petlje upotrebom frejmova (`frame based loop`).

To znači da se telo petlje izvršava fiksni broj puta u sekundi (od 20 do 50).

Drugi način je rad glavne petlje zasnovan na događajima (`event based loop`).

To znači da se telo glavne petlje izvrši po jednom za svaki događaj koji se desi u igri.

004 Pomeranje po jednoj osi

Korišćenjem `event based loop`, može se pomerati objekat po jednoj osi na ekranu.

Ako je omogućeno da se ponove događaji pritiska na isto dugme tastature, dugmad koja se pritisnu generišu višestruki `pygame.KEYDOWN` događaj.

Sa funkcijom `pygame.key.set_repeat(odlaganje, interval)` se kontroliše kako se ponavlja pritisak na istu dugmad tastature.

`Odlaganje` je broj milisekundi pre slanja prvo ponovljenog `pygame.KEYDOWN` događaja.

Posle toga sledeći `pygame.KEYDOWN` događaj će biti poslat za svaki `interval` milisekundi.

Ako se ne pojavi nijedan argument u zagradi, tako se onemogućava reakcija na ponovljeno pritiskanje istog dugmeta tastature.

Sa `pygame.key.set_repeat(50, 25)` postavlja se da se prvi put događaj generiše posle 50 milisekundi a svaki sledeći događaj se generiše posle sledećih 25 milisekundi.

Promenljive `x` i `y` su tekuće koordinate centra objekta koji se pomera i one su definisane kao globalne promenljive, što znači da ako im se promeni vrednost u bilo kojem delu koda, ta promena važi za ceo kod.

Promenljiva `brzina_x` je pomeraj objekta po x-osi tokom kretanja.

Funkcija `crtanje_objekta()`:

```
def crtanje_objekta():
    ekran.fill(pygame.Color("white"))
    pygame.draw.rect(ekran, pygame.Color("blue"), (x, y, 100, 50), 0)
```

radi samo crtanje pravougaonika sa gornjim levim uglom na `x, y` poziciji.

Funkcija `obrada_dogadjaja()` kao izlaz daje `True` ako se desio pritisak leve ili desne strelice kursora, ili daje `False` ako se bilo koji od ta dva događaja nije desio.

Ako se bilo koji od ta dva događaja nije desio, nema potrebe za ponovnim crtanjem pravougaonika na nekoj novoj poziciji jer se nije pomerilo sa stare pozicije.

Promenljiva `pomerio_se` dobija izlaz funkcije `obrada_dogadjaja()` i time se pokreće ili ne ponovno crtanje pravougaonika na ekranu.

Levo_desno.py

```
import pygame, random
pygame.init()
pygame.display.set_caption("Levo Desno")
ekran_sirina = 1600
ekran_visina = 800
ekran = pygame.display.set_mode((ekran_sirina, ekran_visina))
pygame.key.set_repeat(50, 25)
```

```

(x, y) = (ekran_sirina // 2, ekran_visina // 2)
brzina_x = 10
def crtanje_objekta():
    ekran.fill(pygame.Color("white"))
    pygame.draw.rect(ekran, pygame.Color("blue"), (x, y, 100, 50), 0)

def obrada_dogadjaja(dogadjaj):
    global x
    if dogadjaj.type == pygame.KEYDOWN:
        if dogadjaj.key == pygame.K_LEFT:
            x -= brzina_x
            return True
        if dogadjaj.key == pygame.K_RIGHT:
            x += brzina_x
            return True
    return False

pomerio_se = True
kraj = False
while not kraj:
    if pomerio_se:
        crtanje_objekta()
        pygame.display.update()
        pomerio_se = False

    dogadjaj = pygame.event.wait()
    if dogadjaj.type == pygame.QUIT:
        kraj = True
    else:
        pomerio_se = obrada_dogadjaja(dogadjaj)

pygame.quit()

```

Zadaci

013) Izmeniti kod programa Logo_na_sredinu.py tako da izabrana slika se postavi na sva cetiri ugla ekrana bilo koje dimenzije.



014) Izmeniti kod programa Levo_desno.py tako da nacrtani objekat ne može izaći sa ekrana.

015) Modifikovati kod programa Levo_desno.py tako da nacrtani objekat se kreće samo gore i dole posle pritiska na dugmad kursora i da ne može izaći sa ekrana

016) Napisati kod igre Setnja_po_ivici.py. Nacrtani objekat može da se kreće samo po ivicama prozora.

017) Napisati kod igre Setnja.py. Nacrtani objekat može da se kreće po celom ekranu do granica ekrana.