

# Tema 1

## VEZA IZMEĐU PYTHON I PYGAME

- Iscrtam im skup veliki Python, i nekoliko malih skupova u njemu koje su njegove biblioteke, jedna od njih je Pygame, a ostale npr math.py, random.py
- Razlika u okruženju IDLEa i Visual Studio Codea - nacrtam im koji program gdje salje OUTPUT, gdje se ispisiuje print, a dodatno da IDLE gdje koristimo PYGAME pravi dodatni prozor
- Shell u idleu a Terminal u Visual studio, ali da su to iste stvari
- Uvodim i PYGAMEBG.py biblioteku u priču, kao još jedan skup u PYTHON skupu, ta biblioteka je kao i pygame, samo su njene funkcije i njen sadržaj lakši za korištenje
- Svaki put kada koristimo PYGAME mi ćemo koristiti i PYGAMEBG biblioteku
- Objašnjenje kako se ti skupovi gore međusobno vide - **import** naredba
- Objašnjenje šta je biblioteka u programiranju - fajl sa raznim funkcionalnostima. Kao stvarna biblioteka koja ima skup knjiga o raznim temama, tako biblioteka u programiranju ima skup raznih fja. Jedino pravilo je da su te funkcije istorodne, da su kao o istoj temi, kao npr da biblioteka bude samo vezana za knjige Harija Potera ili samo za knjige iz hemije, itd..
- Ekstenzija za pygame je isto PY kao za python, objasniti ekstenzije generalno

## PRAVLJENJE IGRICA

- U python smo mogli da pravimo samo terminalne igre
- U pygame možemo da pravimo 2d igrice (povući paralelu sa scratch-om)
- Za finalnu igricu moraćemo da naučimo da: (svo vreme crtam na tabli gdje se to vidi na igrici)
  - Otvaramo i zatvaramo prozor
  - Bojimo i crtamo
  - Dodajemo tekst i menjamo njegovu vrednost u toku izvršavanja programa (dinamicki)
  - Pamtim i registrujemo neki klik na tastaturi
  - Animiramo objekte na prozoru
  - Upisujemo rezultat u fajl i citamo rezultat iz fajla



## OKRUŽENJE - IDLE

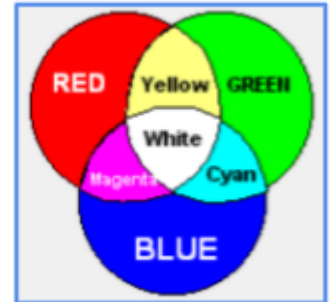
- Prodjemo kroz osnovne naredbe koje ćemo koristiti
- New file, run, pokazemo shell i pokazemo taj novi prozor u kome kuckamo kod
- Nađemo mesto gdje ćemo čuvati sve naše pygame fajlove

### PRIMER 1: Otvaranje i zatvaranje prozora

```
import pygame
import pygamebg
#setovanje dimenzija
(sirina, visina) = (400, 400)#torka, podsetiti, tip podataka u Python
#prikazivanje prozora
prozor = pygamebg.open_window(sirina, visina, "Zdravo svete!")
#zatvaranje prozora
pygamebg.wait_loop()
```

### PRIMER 2: Boje

- Istraziti rgb i heksa dekadni sistem povezanost
- otvoriti na netu color picker da tu mogu da biraju boje
- naredba fill() i nacini za bojenje prozora
- crna kao default boja pozadine prozora



```
#setap pozadine prozora
prozor.fill(pygame.Color("Red"))
#naredba preko naredbe, redosled izvorsavanja naredbi
prozor.fill((100, 100, 100))
```

### PRIMER 3: Crtanje linije - fja draw

- Da bismo mogli crtati, moramo znati orijentaciju u tom prostoru
  - Koordinatni sistem, Pikseli kao jedinice mere, Apsolutna i relativna udaljenost
  - Nacrtao prozorčić na tabli i koordinatni sistem (za online koristiti paint)
  - Napraviti paralelu sa Dekartovim sistemom u matematici i sta je isto a sta je razlicito
  - Ispisati za jedan primjer za jednu tacku npr 75,100 gdje se sve nalaze tacke koje su za 35 piksela gore dole levo i desno od nje
  - Apsolutna udaljenost od koordinatnog pocetka vs relativna udaljenost od neke tacke
- Crtanje se nalazi u delu DRAW u PYGAME
  - Razgraniciti da se funkcije za otvaranje i zatvaranje prozora nalaze u pygamebg a za bojenje i crtanje u pygame
- `pygame.draw.line(prozor, blue, (100, 60), (80, 140), 2)`
  - Objasniti šta znače svaki od ovih parametara - display, boja, početak linije, kraj linije, debljina linije

```
import pygame
import pygamebg
(sirina, visina) = (200, 200)
prozor = pygamebg.open_window(sirina, visina, "A")
prozor.fill(pygame.Color(255, 255, 0))
blue = (0, 0, 255)
pygame.draw.line(prozor, blue, (100, 60), (80, 140), 2)
pygame.draw.line(prozor, blue, (100, 60), (120, 140), 2)
pygame.draw.line(prozor, blue, (90, 100), (110, 100), 2)
pygamebg.wait_loop()
```