



Software
Engineering
School

Курсы программирования для
взрослых и детей

Делаем игры на Python

Игровой цикл

Сердце любой игры. Благодаря его непрерывной работе игра функционирует – в это время происходит много всего: ввод от пользователя, его обработка, обновление состояния игры, отрисовка экрана и так далее... Один проход цикла называется «кадром»



Скелет игры в псевдо-коде

здать_начальные_настройки_игры()

пока игра_запущена:

 обработать_ввод_пользователя()

 обновить_состояние_игры()

 отрисовать_экран()



Игровой цикл!!!

Скелет игры в Python

```
initialize() # Начальные настройки
```

```
while True:
```


```
    process_input() # Обработка ввода
```

```
    update_game_state() # Обновление состояния
```

```
    render() # Отрисовка
```

```
    if is_game_over(): # Проверка на завершенность
```

```
        break
```



Игровой
цикл!!!

Игра «Угадай число»

Привет! Я загадал число от 1 до 100...

Какое число я загадал? Asdb

Введите число!

Какое число я загадал? 24

Неа, я загадал число меньше

Какое число я загадал? 12

Вообще не то... Мое число больше

Какое число я загадал? 14

Молодец, правильно!

Задаем начальные настройки игры

```
from random import randint
```

```
def initialize():  
    LOW_BOUND = 1  
    HIGH_BOUND = 100  
    guessed_number = randint(LOW_BOUND, HIGH_BOUND)  
    return LOW_BOUND, HIGH_BOUND, guessed_number
```

Обрабатываем ввод от пользователя

```
def process_input():  
    valid_input = False  
  
    while not valid_input:  
        user_input = input("Какое число я загадал? ")  
        if not user_input.isdigit():  
            print("Введите число!")  
        else:  
            valid_input = True  
  
    return int(user_input)
```

Обновляем состояние игры

```
def update_game_state(user_input, guessed_number):  
    if user_input == guessed_number:  
        game_state = "win"  
    elif user_input < guessed_number:  
        game_state = "high"  
    elif user_input > guessed_number:  
        game_state = "low"  
  
    return game_state
```


Отрисовываем экран

```
def render(game_state):  
    if game_state == "win":  
        print("Молодец, правильно!")  
    elif game_state == "low":  
        print("Неа, я загадал число меньше")  
    elif game_state == "high":  
        print("Вообще не то... Мое число больше")
```

Проверяем завершенность игры

```
def is_game_over(game_state):  
    return game_state == "win"
```

Собираем игровой цикл

```
def game_loop():  
    low_bound, high_bound, guessed_number = initialize()  
  
    print(f"Я загадал число от {low_bound} до {high_bound}...  
    {guessed_number}")  
  
    while True:  
        user_number = process_input()  
        game_state = update_game_state(user_number, guessed_number)  
        render(game_state)  
        if is_game_over(game_state):  
            break  
  
game_loop()
```

**Как модифицировать
игру, чтобы количество
попыток угадывания было
фиксированным?**

Pygame. Как установить?

1. Проверяем, что на компьютере установлен Python:

```
> python --version  
Python 3.12.1
```

Если вывод не похож на этот, значит сначала нужно установить Python (<https://www.python.org>)

2. Устанавливаем Pygame:

```
> python -m pip install pygame
```

Начальная настройка игры

```
import pygame # Импортируем библиотеку

pygame.init() # Включаем пайгейм

# Создаем окно будущей игры с заданными размерами в пикселях
WINDOW_WIDTH, WINDOW_HEIGHT = 500, 500
WINDOW_COLOR = (255, 255, 0) # Это белый в RGB
main_window = pygame.display.set_mode((WINDOW_WIDTH,
                                         WINDOW_HEIGHT))
pygame.display.set_caption("Моя первая игра")
```

Правильная начальная настройка

```
import pygame
```

```
def initialize():  
    pygame.init()
```

```
    WINDOW_WIDTH, WINDOW_HEIGHT = 500, 500  
    main_window = pygame.display.set_mode((WINDOW_WIDTH,  
                                           WINDOW_HEIGHT))  
    pygame.display.set_caption("Моя первая игра")
```

```
    return main_window
```

Игровой цикл в Pygame

Представим, что тут идет код с прошлого слайда...

```
while True:
```

```
    # «Ввод данных» от пользователя
```

```
    events = pygame.event.get()
```

```
    for event in events:
```

```
        # Обработка ввода и обновление состояния игры идут тут...
```

```
        # Выход из игры
```

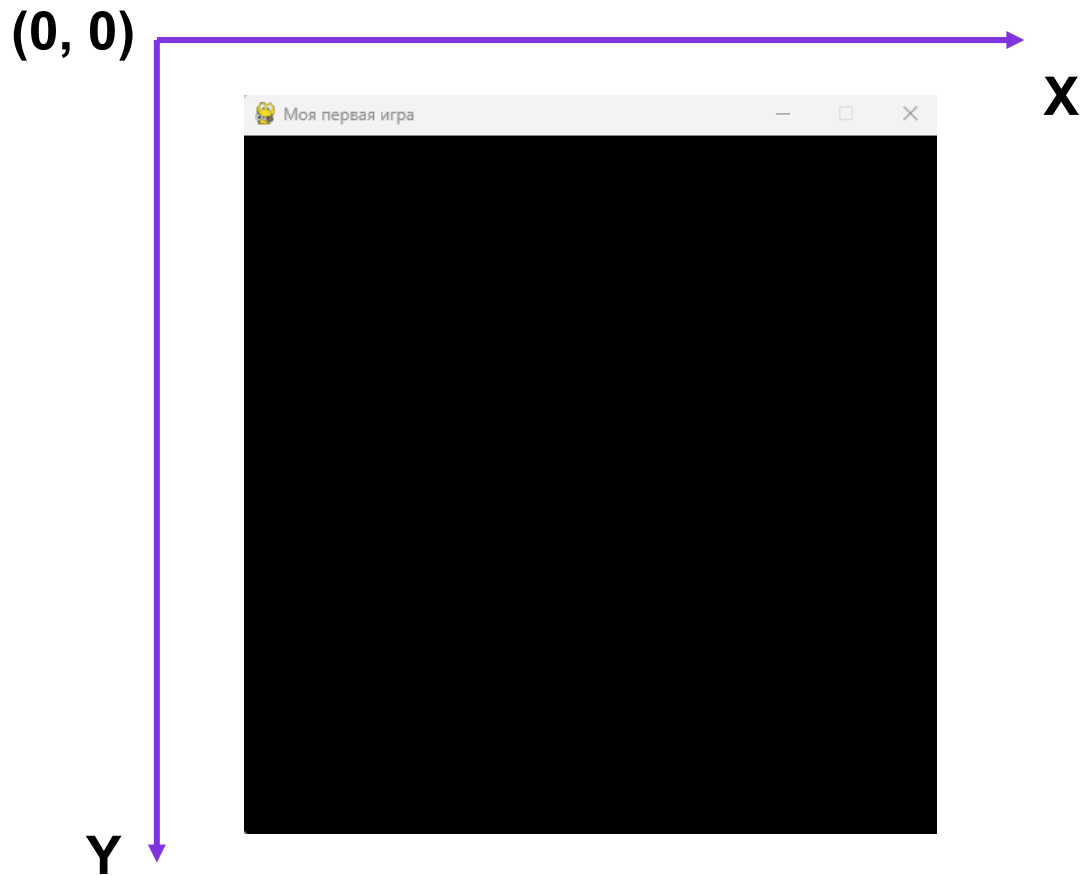
```
        if event.type == pygame.QUIT:
```

```
            break
```

```
    pygame.display.update() # Отрисовываем экран
```

```
pygame.quit() # Не забываем выключить пайгейм в конце игры!
```


Как выглядит окно игры



Рисуем прямоугольники

Во-первых, прямоугольник надо создать:

```
left_top_x, left_top_y, width, height = 0, 0, 200, 300  
rectangle = pygame.Rect(left_top_x, left_top_y, width, height)
```

Во-вторых, прямоугольник надо отобразить:

```
color = (255, 0, 0) # Цвета задаются в RGB, это красный  
pygame.draw.rect(main_window, color=color, rect=rectangle)
```

Рисуем прямоугольники

Во-первых, прямоугольник надо создать:

```
left_top_x, left_top_y, width, height = 0, 0, 200, 300  
rectangle = pygame.Rect(left_top_x, left_top_y, width, height)
```

Во-вторых, прямоугольник надо отобразить:

```
color = (255, 0, 0) # Цвета задаются в RGB, это красный  
pygame.draw.rect(main_window, color=color, rect=rectangle)
```

Рисуем прямые линии

Во-первых, создадим начальную и конечную точку для линии:

```
start_point = (0, 0)
end_point = (100, 100)
```

Во-вторых, отображаем линию:

```
color = (0, 255, 0) # Цвета задаются в RGB, а это зеленый
pygame.draw.line(main_window, color=color,
                  start_pos=start_point, end_pos=end_point)
```

Рисуем круги

Во-первых, зададим координаты центра и радиус:

```
center = (200, 200)
```

```
radius = 50
```

Во-вторых, отобразим сам круг:

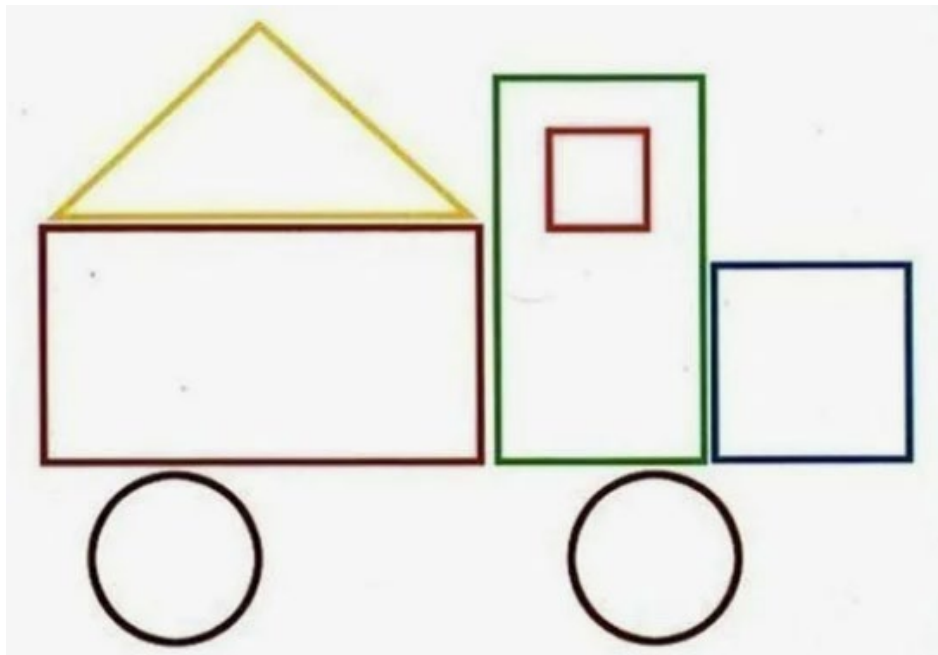
```
color = (0, 255, 0) # Цвета задаются в RGB, а это зеленый
```

```
pygame.draw.circle(main_window, color=color, center=center,  
                    radius=radius)
```

Выводим на экран фигуры

```
# Представим, что тут идет код с прошлого слайда...
left_top_x, left_top_y, width, height = 0, 0, 200, 300
rectangle = pygame.Rect(left_top_x, left_top_y, width, height)
color = (255, 0, 0) # Цвета задаются в RGB, это красный
running = True
while running:
    # «Ввод данных» от пользователя
    events = pygame.event.get()
    for event in events:
        # Выход из игры
        if event.type == pygame.QUIT:
            running = False
    pygame.draw.rect(main_window, color=color, rect=rectangle)
    pygame.display.update() # Отрисовываем экран
```

Задача...



Наш репозиторий

https://github.com/samedit66/pygame_2024/tree/main