

Рассмотрим библиотеку PYGAME

Pygame — это «игровая библиотека», набор инструментов, помогающих программистам создавать игры. К ним относятся:

Графика и анимация Звук (включая музыку) Управление (мышь, клавиатура, геймпад и так далее)

Игровой цикл В сердце каждой игры лежит цикл, который принято называть «игровым циклом». Он запускается снова и снова, делая все, чтобы работала игра. Каждый цикл в игре называется кадром.

В каждом кадре происходит масса вещей, но их можно разбить на три категории:

- 1. Обработка ввода (события)
- 2. Обновление игры
- 3. Рендеринг (прорисовка)

Еще один важный аспект игрового цикла — скорость его работы. Многие наверняка знакомы с термином FPS, который расшифровывается как Frames Per Second (или кадры в секунду).

In [4]:

```
# Установка библиотеки
!pip install pygame
```

Requirement already satisfied: pygame in d:\prg\python367\lib\site-packages (1.9.6)

In [3]:

```
# Pygame шаблон - скелет для нового проекта Pygame
import pygame
import random

WIDTH = 360 # ширина игрового окна
HEIGHT = 480 # высота игрового окна
FPS = 30 # частота кадров в секунду
white = (255,255,255) # RED = 255, GREEN = 255, BLUE = 255
black = (0,0,0)
green = (0, 200, 64)
yellow = (225, 225, 0)
pink = (230, 50, 230)
```

Нарисовать линии pygame.draw.lines(screen, color, closed, pointlist, thickness)

Чтобы нарисовать прямоугольник pygame.draw.rect(screen, color, (x,y,width,height), thickness)

Нарисовать круг pygame.draw.circle(screen, color, (x,y), radius, thickness)

In [4]:

```
# создаем игру и окно
pygame.init() # это команда, которая запускает pygame
pygame.mixer.init() # для звука
screen = pygame.display.set_mode((WIDTH, HEIGHT)) # окно программы, которое создается, когда мы задаем его размер
pygame.display.set_caption("Игра 1") # Задаем наименование окна
clock = pygame.time.Clock() # Управление временем.

# Цикл игры
running = True
while running:
    # Обновление
    screen.fill(white) # Заливка экрана заданным цветом

    pygame.draw.line(screen, black, [0,0], [100,100], 3) # Рисуем линию толщиной 3.
    # pygame.draw.Lines(screen, black, True, [[0,0], [0,100], [150,10]], 3) # Рисуем ломаную линию толщиной 3.
    # Рисуем полигон
    # pygame.draw.polygon(screen, black, [[150, 10], [180, 50], [90, 90], [30, 30]])
    # Рисуем круги
    # pygame.draw.circle(screen, yellow, (100, 100), 50)
    # pygame.draw.circle(screen, pink, (200, 100), 50, 10)

    # Визуализация (сборка)
    # после отрисовки всего, переворачиваем экран
    pygame.display.flip() # OR pygame.display.update()

    # Ввод процесса (события)
    # Проверим, какие события произошли
    for event in pygame.event.get():
        # проверить закрытие окна
        if event.type == pygame.QUIT:
            running = False

    # держим цикл на правильной скорости
    clock.tick(FPS)

pygame.quit()
```

In []:

In [1]:

```
# Управление квадратом
import pygame
import time

WIDTH = 400
HEIGHT = 300
FPS = 3 # частота кадров в секунду

# создаем игру и окно
pygame.init()
dis = pygame.display.set_mode((WIDTH, HEIGHT))
red = (255,0,0)
green = (0,255,0)
blue = (0,0,255)
white = (255,255,255)

x1 = 200
y1 = 150
dx1 = 0
dy1 = 0

pygame.display.update()
pygame.display.set_caption('Пробная игра')

clock = pygame.time.Clock()

game_end = False
while not game_end:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            game_end = True
        # print(event)
        if event.type == pygame.KEYDOWN: # Проверить что произошло событие нажатия кнопки
            if event.key == pygame.K_LEFT:
                dx1 = -10
                dy1 = 0
            if event.key == pygame.K_RIGHT:
                dx1 = 10
                dy1 = 0
            if event.key == pygame.K_UP:
                dx1 = 0
                dy1 = -10
            if event.key == pygame.K_DOWN:
                dx1 = 0
                dy1 = 10
        if x1 >= WIDTH or x1 < 0 or y1 >= HEIGHT or y1 < 0:
            game_end = True
        x1 = x1 + dx1
        y1 += dy1
        dis.fill(white)
        pygame.draw.rect(dis, red, [x1,y1,20,20])
        pygame.display.update()
    # print(x1, ' - ', y1)
    clock.tick(FPS)

pygame.quit()
quit()

pygame 1.9.6
Hello from the pygame community. https://www.pygame.org/contribute.html
```