

# | КЛАСС SPRITE.

**ЦЕЛЬ:** Вспомнить, что такое классы. Создать класс Sprite. Модифицировать код, используя экземпляр класса Sprite.

## ПЛАНИРОВАНИЕ

### 1. Что такое объекты.

Сегодня мы вспомним, что такое объектно-ориентированное программирование (ООП), а те, кто совсем забыл или не знакомы – узнают, что это такое.

#### Пример объяснения принципа ООП:

В жизни нас окружает множество объектов, сложных и простых, есть объекты статичные (например – Здания, Гора, Памятник) и есть динамические (например – капля дождя, автомобиль, человек, планета), есть составные объекты (например: компьютер, программа, солнечная система) состоящие из множества других объектов, которые тоже состоят из других объектов, а есть простые – камень например. Но в то же время и камень и компьютер на каком-то уровне состоят оба из огромного количества разных элементов: молекул и атомов. Можно сказать, что все в мире является объектами и любой объект состоит из других объектов, которые в свою очередь тоже состоят из объектов.

И у каждого объекта есть какие-то функции и какие-то свойства. Что значит – функция? Это значит, что объект может выполнять какое действие. Например, возьмем такой объект как Автомобиль. Автомобиль – это объект и у него есть функция – передвигаться, или – ехать. Мало того, у него есть и другие функции:

- перевозить пассажиров
- согревать воздух внутри салона
- изменять направление движения
- светить фарами

В программировании функции объектов – называю методами.

С функциями разобрались – это действия, которые может выполнять объект.

Что такое свойства объекта? Это его характеристики или, если сказать совсем по-простому: это какой он – объект.

Например, рассмотрим свойства нашего объекта- Автомобиля. У него есть такие свойства: цвет, размер, скорость.

У каждого свойства есть значение, например, у свойства цвет может быть значение – **БЕЛЫЙ**, а у свойства – Скорость значение уже будет числовым.

#### ! Проблемная задачи:

Какие еще есть свойства у автомобиля? Какие могут быть значения у этих свойств?

**\*\*\* работа без компьютера\*\*\***

Разбиваемся на команды (количество команд зависит от кол-во учеников в классе).

Каждой команде, необходимо придумать по 5 объектов, для каждого по 5 функций и 5 свойств.

## 2. Что такое классы?

ООП – это такое программирование, когда мы в программе создаем объекты и программируем их действия и свойства. Функции объектов называются - **методы**.

Рассмотрим еще раз такой объект из нашей жизни – автомобиль. Все себе хорошо представляют, как выглядит автомобиль. А теперь скажите, откуда берется автомобиль, как он появляется? Автомобиль изготавливают на заводе, люди и автоматизированные устройства. А откуда люди и устройства сборки знают, как делать автомобиль? Все данные об автомобиле: какой у него будет корпус, какой мотор, цвет, салон, руль – вся информация об автомобиле является своего рода его шаблоном. Чертеж автомобиля, описание деталей, материалов и всех систем – все это шаблон автомобиля. И в программировании тоже можно создавать шаблоны для объектов. И эти шаблоны называются – **классами**.

Например:

Класс Автомобиль:

Цвет: красный

Количество дверей: 4

Объем мотора: 3 литра

Коробка переключения передач: Автоматическая

**ФУНКЦИЯ** Ехать():

Установить количество оборотов двигателя (2500)

**ФУНКЦИЯ** Повернуть(сторона)

**Если** сторона = налево **то**

Крутить руль влево

**Если** сторона = направо **то**

Крутить руль вправо

**ФУНКЦИЯ** Стоп():

Установить количество оборотов двигателя (0)

Мы создали шаблон автомобиля – описали класс Автомобиль.

Теперь можем создать объект из этого класса:

Моя ласточка = Автомобиль()

После этой команды будет создан объект с именем – **Моя ласточка**. И этим объектом будет являться автомобиль красного цвета с трехлитровым мотором и коробкой-автомат. Он будет уметь ехать, поворачивать и останавливаться.

Если мы вызовем метод – Стоп():

**Моя\_ласточка.Стоп()**

Автомобиль остановиться. Или метод – повернуть(сторона):

**Моя\_ласточка.Повернуть(направо)**

Автомобиль повернет направо.

И еще, мы можем изменить характеристики автомобиля. Например, выполнить команду:

**Моя\_ласточка.цвет = черный**

После этой команды, автомобиль с названием **Моя\_ласточка** станет черного цвета)

**! Проблемная задача:**

**Придумать и записать на листах 3 примера классов из нашей жизни.**

### 3. Создаем класс Sprite

Пришло время нам создать свой первый класс. При создании игр, обычно самым главным классом является класс, который является шаблоном для создания **Спрайта** (ведь спрайт – это самый основной объект в любой игре).

Давайте опишем класс спрайт. Какие будут у него свойства?

Свойства класса спрайт:

- поверхность с загруженным из файла изображением (текстура или костюм спрайта)
- координаты спрайта – это будут две переменные: **x** и **y**
- скорость спрайта – переменная **speed**

На Python:

```
class Sprite():  
    image = pygame.image.load('Images\Player.png')  
    speed = 1  
    x = 100  
    y = 100
```

Мы описали шаблон спрайта, а теперь создадим объект (Экземпляр) нашего шаблона (класса):

**Player = Sprite()**

И перепишем нашу программу уже по-новому:

```

import pygame

class Sprite():

    image = pygame.image.load('Images\Player.png')

    speed = 1

    x = 100

    y = 100


w = pygame.display.set_mode ((1279, 700))

Player = Sprite()


game = True

while game:

    for ev in pygame.event.get ():

        if ev.type == pygame.QUIT:

            game = False

    keys = pygame.key.get_pressed()

    if keys[pygame.K_RIGHT]:

        Player.x += Player.speed

    elif keys[pygame.K_LEFT]:

        Player.x -= Player.speed


    w.fill((0, 0, 0))

    w.blit(Player.image, (Player.x, Player.y))

    pygame.display.update()


pygame.quit ()

```

**! Проблемная задача:**

**Создать один метод (функцию) в классе Sprite. Использовать ее в основной программе.**

#### **4. Рефлексия**

- Сегодня мы вспомнили ООП
- создали класс Sprite и экземпляр (объект) этого класса