

Разработка игры на Python: создание персонажа игрока

Беликов Андрей Геннадьевич

Приамурский государственный университет имени Шолом-Алейхема

Студент

Аннотация

В данной статье был рассмотрен процесс создания персонажа игрока на языке Python. В исследовании использовалась программа PyCharm. В результате работы был разработан персонаж игрока для игры на языке Python.

Ключевые слова: Геймдев, PyCharm, Python

Python game development: creating a player character

Belikov Andrey Gennadievich

Sholom-Aleichem Priamursky State University

Student

Abstract

This article reviewed the process of creating a player character in Python. The PyCharm program was used in the study. As a result of the work, a player character was developed for the game in Python.

Keywords: Game Dev, PyCharm, Python

Актуальность создания игры на языке Python обусловлена тем, что данный процесс позволяет связать изучение языка и развлечение.

Цель данной статьи, создать персонажа игрока для игры на языке Python.

Для создания проекта была рассмотрена статья М.И. Мялова в которой рассмотрен процесс создания и разработки игры на языке программирования Python [1]. Так же Плас Дж. Вандер описал возможности языка программирования Python [2]. А. Д. Тулегулов рассмотрел и показал примеры и методы обучения и реализации на языке программирования Python [3]

Для создания персонажа будем использовать сервис для рисования в пиксельной графике Pixilart. Сервис является бесплатным и прост в использовании. В этом сервисе нарисуем пушку на прозрачном фоне чтобы использовать её как полноценного персонажа. Выбираем цвет и размер кисти и рисуем подходящую модель (Рис.1).

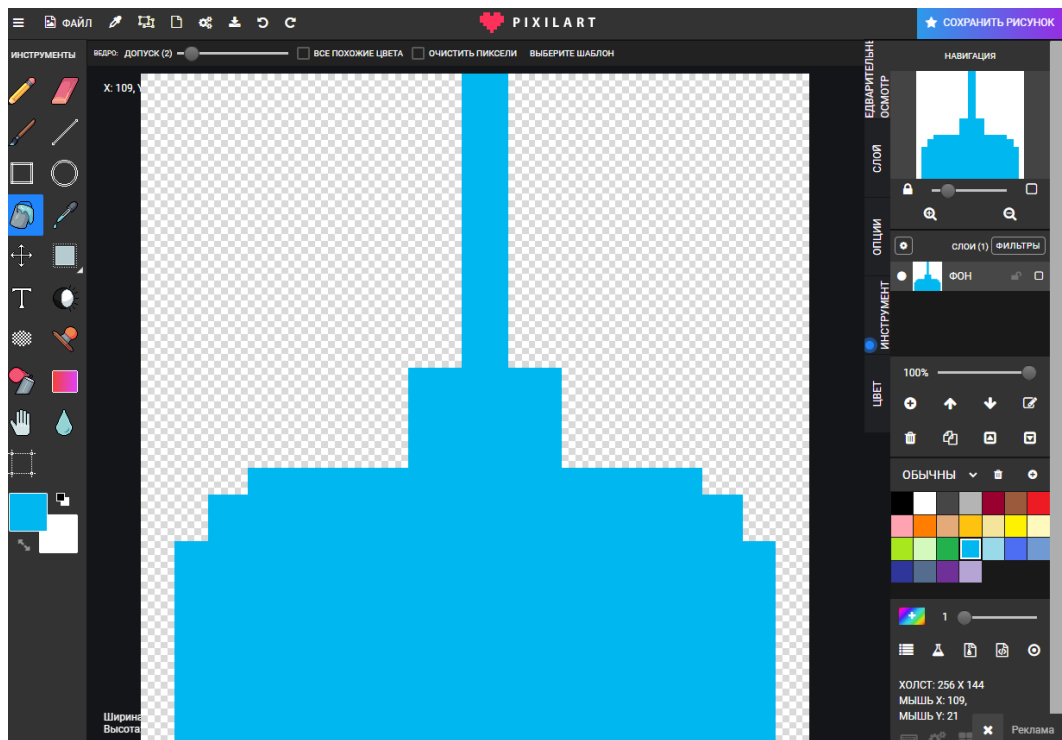


Рисунок 1 – Готовое изображение пушки

После готовое изображение импортируем в PyCharm в формате png в отдельную папку с изображениями, где в последующем будут сохраняться все созданные образы (Рис.2).

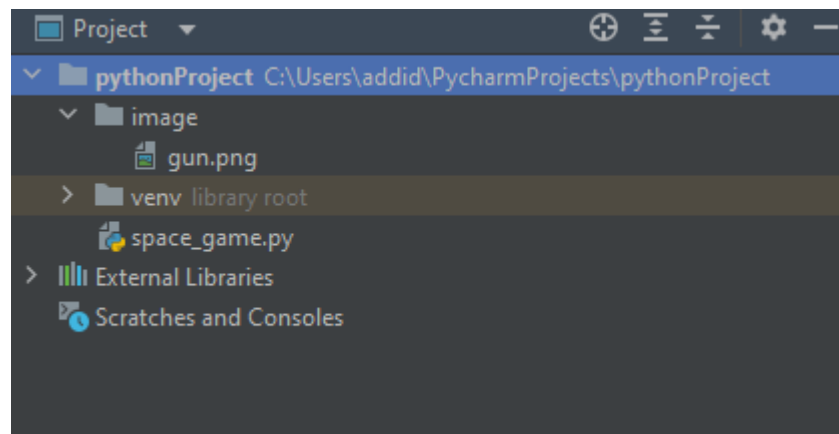


Рисунок 2 – Папка с изображениями

Далее в эту папку перетаскиваем изображение с пушкой и создаём новую папку под названием «gun» там будет прописываться поведение игрового персонажа, а именно его передвижение по игровому полю (Рис.3).

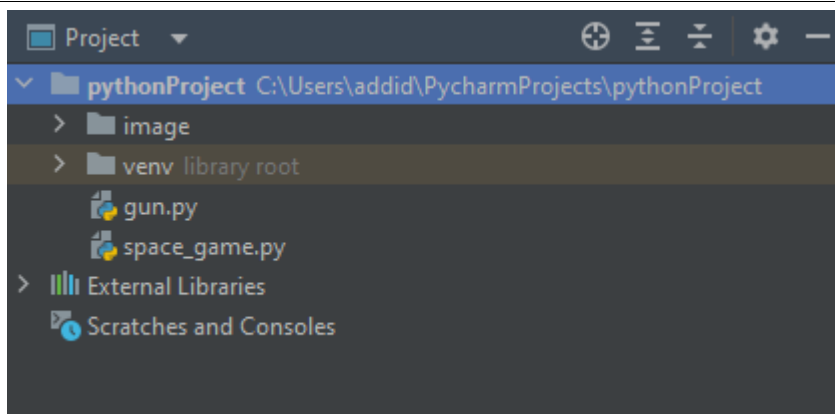


Рисунок 3 – Расположение папки с пушкой

Для того чтобы пушка появилась в игровом поле необходимо её подгрузить в Pygame и обрисовать её в графическом объекте. Для этого необходимо импортировать Pygame в файле «gun». Игровую механику и инициализацию пушки прописываем далее с помощью команды «def __init__(self, screen)». Далее присваиваем пушке изображение при этом прописываем расположение файла. Расположение файла показывается с помощью команды «self.image = pygame.load('image/gun.png')». Изображение пушки необходимо расположить на графическом объекте с помощью команды «self.rect = self.image.get_rect()». Необходимо получить объект экрана и прописываем «self.screen_rect = screen.get_rect()». Далее прописываем координаты центра пушки «self.rect.centerx = self.screen_rect.centerx» при этом центр графического изображения будет по центру экрана. Для этого приписываем нижнюю координату «self.rect.bottom = self.screen_rect.bottom» после чего прописываем функцию, которая будет выводить пушку на экран и используем следующую запись «def output(self)». Для того чтобы пушка отображалась на экране прописываем «self.screen.blit(self.image, self.rect)» (Рис.4).

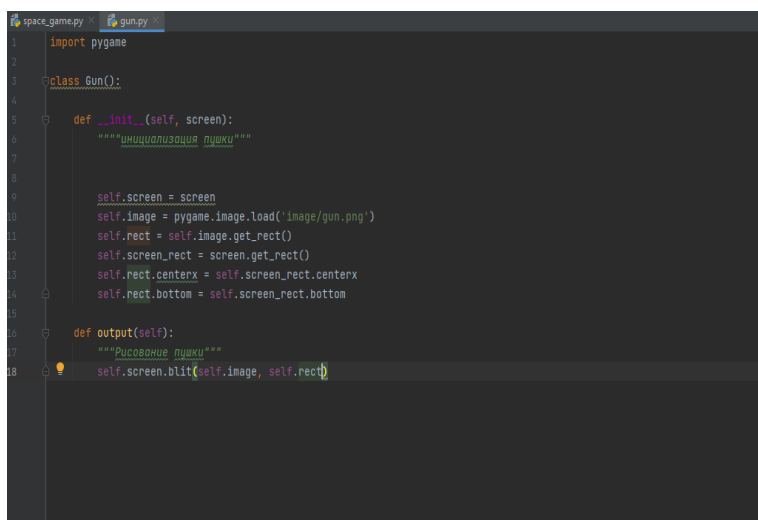
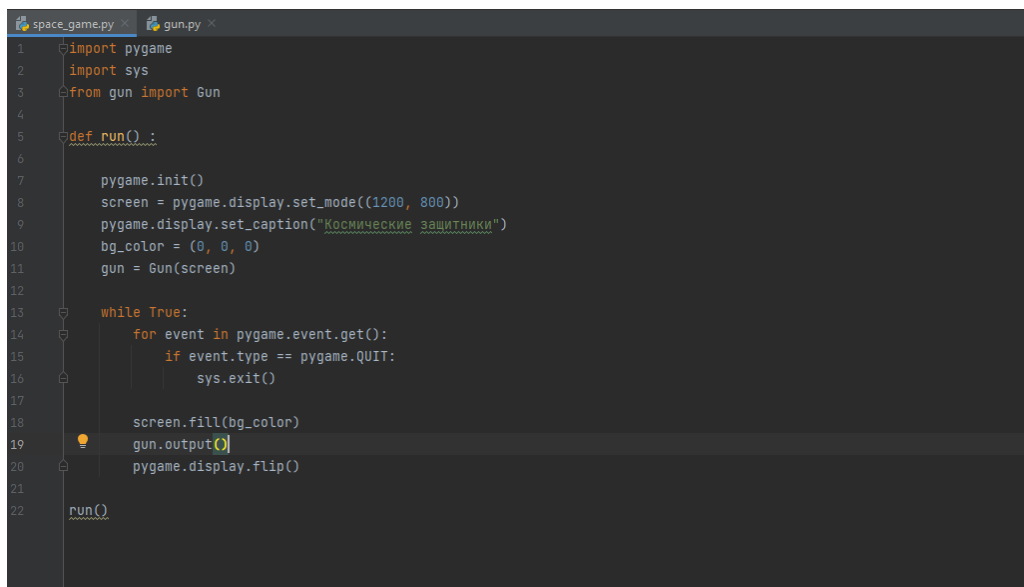


Рисунок 4 – Отображение пушки

Далее следует импортировать пушку к экрану игры для этого прописываем «import pygame», «import sys» и указываем путь изображения «from gun import Gun». Далее объект пушку прописываем и обрисовываем его на графическом объекте «gun = Gun(screen)». Дальше в главном цикле программы добавляем «gun.output()» (Рис.5).



```
1 import pygame
2 import sys
3 from gun import Gun
4
5 def run() :
6
7     pygame.init()
8     screen = pygame.display.set_mode((1200, 800))
9     pygame.display.set_caption("Космические защитники")
10    bg_color = (0, 0, 0)
11    gun = Gun(screen)
12
13    while True:
14        for event in pygame.event.get():
15            if event.type == pygame.QUIT:
16                sys.exit()
17
18        screen.fill(bg_color)
19        gun.output()
20        pygame.display.flip()
21
22    run()
```

Рисунок 5 – Импорт пушки на игровое окно

Далее нажимаем на кнопку «Run» и получаем изображение нашей пушки на игровом поле (Рис.5)

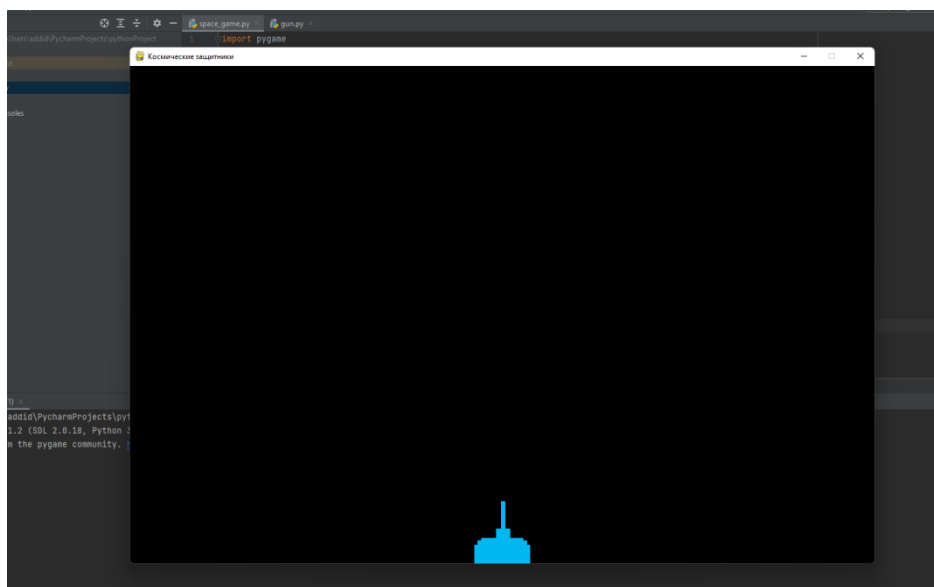


Рисунок 6 –Пушка в игровом окне

В данной статье был рассмотрен процесс создания персонажа игрока на языке Python. В исследовании использовалась программа PyCharm. В результате работы был разработан персонаж игрока для игры на языке Python.

Библиографический список

1. Мялова М.И, Пазин М.А. Разработка пользовательского интерфейса на основе Python для автоматического выбора алгоритма кластеризации данных. В сборнике: Цифровые инфокоммуникационные технологии. сборник научных трудов. Ростов-на-Дону, 2021. С. 77-81.
2. Вандер П. Дж. Python для сложных задач: наука о данных и машинное обучение. СПб.: Питер, 2018. 576 с.
3. Тулегулов А. Д., Ешпанов В. С., Исмаилов А. Практический опыт обучения методам интеллектуального анализа на платформе Python Anaconda // Цифровое образование: новая реальность: Сборник материалов Всероссийской научно-методической конференции с международным участием, Чебоксары, 16 ноября 2020 года. Чебоксары: Общество с ограниченной ответственностью «Издательский дом «Среда», 2020. С. 197-200.