**Пояснительная записка к проекту pygame: Игра «За двоих»**

**Автор:** Демьянов Антон Павлович

**Описание:**

Проект – это игра, написанная на языке Python. При запуске пользователь попадает в основное меню программы или же главный экран, где имеются кнопки и «Правила игры» с подсказкой. Далее он по нажатию кнопок производит действия в игре. Он может начать новую игру, загрузить игру или выйти из игры (имеется в виду закрыть всю программу).

Если игрок начнет новую игру, то ему предложат ввести имя, после чего он попадет в начало игры, где ему нужно будет заходить в порталы и перемещаться на уровень выше и выше, чтобы пройти игру. В «игре» у пользователя будет показываться имя пользователя, которое он выбрал при старте, номер уровня и количество шагов, которые он сделал.

При нажатии “ESCAPE” откроется “внутри игровое меню” в котонром пользователь сможет по нажатию выйти в «главное меню», загрузить или сохранить игру а также выйти из программы.

Если пользователь прошел все 5 уровней, то появится Меню «Конец игры», где пользователь узнает, насколько хорош его результат, также он сможет увидеть результаты самых быстрых прохождений.

Также, после прохождения игрок сможет выйти в главное меню или выйти из программы.

На протяжении всей игры пользователя будут сопровождать информативные сообщения, если он будет сохранять или загружать игру(если будет делать это правильно или неправильно), а также при попытке ввести слишком длинное имя.

**Реализация:**

В коде содержится достаточное количество комментариев, с мини-пояснениями.(P.s.)

В начале - создание screen – экрана, задание его ширины высоты. Создание экрана для самой игры – он немного больше экранов заставки и концовки.

Задаются переменные x1, y1, x2, y2 – глобальные переменные для отслеживания передвижения спрайтов игровых персонажей.

Clock – для времени как ни странно

FPS – для частоты отрисовки экранов, ну это и есть фреймс пер секонд.

nLevel=0 – номер карты

level\_map=None -сюда вернут карту (в виде списка из точек решеток и тд)

username=’’ – имя пользователя

steps=0 – счетчик шагов

start\_move- флаг для отслеживания момента с движением одного из двух, когда другой уперся в стену

menu\_option=0 – флаг, скоро скажу зачем

toptk – создается окошко ткинтера для вызова его при создании игры – задать имя пользователя.

pygame.init() – инициализируем модули pygame

задаем группы спрайтов:

all\_sprites – все спрайты

tiles\_group – спрайты-блоки (ландшафт)

player\_group – спрайты двух персонажей пользователя

После этих “организационных моментов” запускается функция start\_screen(), которая отвечает за “MAIN MENU” или же главное меню, в котором пользователь начинает свое взаимодействие с игрой. Текст с правилами в переменной intro\_text, картинка фона, а также другие картинки и спрайты подгружаются с помощью функции load\_image(), которая загружает картинку из папки data, лежащей рядом с файлом программы. Функция load\_image() вернет картинку, удалив у нее фон, если это необходимо (надо просто передать в нее второе значение = -1).

Далее в start\_screen() задаются параметры для текста в переменную font и запускается бесконечный цикл, внутри которого будет задаваться положения текста, кнопок, текста на кнопках. После этого идет проверка на нажатия пользователя, если он нажал на крестик окна игры, то игра закроется, так же и после нажатия кнопки escape, а если он нажал на кнопку мыши внутри окна игры, то мы поднимаем флаг click. Если клик == тру и координаты его мыши (берем их в переменные mx и my) в момент щелчка в координате кнопки, то вызовется соответствующая функция, иначе ничего не произойдет.

1)Пользователь нажал на кнопку new game:

Запускается цикл для ввода имени пользователем, реализованный через ткинтер, после прохождения цикла – указываются значения в переменные steps, отвечает за подсчет шагов персонажа, nLevel, отвечает за номер карты(уровня) и запускается функция game()(о ней далее).

2)Пользователь нажал на кнопку load game:

Запускается функция load\_game()(она используется только 2 раза и оба при нажатии кнопки load game – один раз в главном и один раз во внутренне игровом меню, соотв. Опишу только один раз)

Функция подразумевает собой загрузку файла, который был создан именно в результате сохранения пользователя в ходе ИМЕННО ЭТОЙ ИГРЫ (кнопка save game во внутренне игровом меню), поэтому по сути эта функция считает файл и, если данные в нем корректны (по ним можно загрузить игру), то данные передадутся в глобальные переменные, а флаг menu\_optionполучит значение 3(значение 3 означает, что игра была загружена, это нужно во избежание рекурсии при загрузке игры из внутренне игрового меню p.s. старая игра бы просто не закончилась). Далее вызывается функция game(), как раз при проверке флага menu\_option.

3)Пользователь нажал на кнопку EXIT- вызывается функция terminate() – по сути просто завершение работы программы.

После всех этих кнопок происходит отрисовка – отображение всего заданного на screen и задается указанная частота кадров для отображения, потом все повторяется.

Функция game() – основная функция игры:

**Программы и библиотеки, использованные для работы:**

Для создания баз данных использовалась программа ‘SQLite studio’

Для создания кода на языке Python использовалась программа “PyCharm”

Для запуска проекта необходимы библиотеки: 