

Пояснительная записка.

Проект: “Нашествие монстров”

Автор: Калинин Кирилл

Введение

Проект представляет собой окно с игрой, в которой главный герой стоит на одной из сторон экрана и отражает набеги монстров, используя огнестрельное оружие.

Цель данной работы:

- научиться писать игры на Python
- научиться пользоваться библиотекой pygame
- научиться работать с анимацией и спрайтами с помощью библиотеки pygame

Основные возможности программы:

- игровая логика:
 - герой находится в левой части экрана, с помощью кнопок W/S/UP/DOWN выполняется его перемещение
 - при нажатии кнопки “ПРОБЕЛ” выполняется выстрел из оружия героя
 - монстры появляются в правой части экрана
 - в зависимости от уровня сложности, с различной скоростью перемещаются справа налево
 - при столкновении пули и монстра, монстр погибает
 - при столкновении монстра и героя, герой теряет жизнь
 - у героя кол-во жизней от 3-х до 1 в зависимости от уровня сложности
- анимированный вывод персонажей (с помощью спрайтов)
- ведение и хранение статистики в базе данных
- вывод таблицы рейтинга игроков

Реализация

Программа реализована с помощью библиотек:

- pygame - для реализации основной логики игры
- pygame_gui - для отображения элементов интерфейса
- sqlite3 - для работы с СУБД SQLite

В проекте реализованы следующие модули:

- bullet.py - класс Bullet для вывода и перемещения пули
- db.py - функции для работы с БД (сохранение и получение данных)
- game.py - класс игры Game (основной геймплей, управление героями, описание уровней и т.д.)
- main.py - запуск игры (точка входа)
- monster.py - класс Monster для вывода и перемещения монстров
- player.py - класс Player для вывода и перемещения героя и его оружия
- settings.py - настройки игры
- ui.py - класс Controls для управления кнопками (показать/спрятать)
- data - директория, где хранятся картинки монстров, игрока, оружия и шрифт

Из состава библиотеки pygame используются следующие классы:

- pygame.sprite.Sprite - реализация собственного спрайта
- pygame.sprite.Group - для работы со спрайтами (отрисовка группы спрайтов)
- pygame.Surface - место вывода графики
- pygame_gui.elements.UIButton - кнопки интерфейса
- pygame_gui.windows.UIConfirmationDialog - диалоговые окна интерфейса

Из библиотеки sqlite3 используются функции:

- connect - для подключения к базе данных (возвращает объект connection)
- connection.cursor.execute - для выполнения запросов к базе данных

- fetchall - для получения результатов запросов

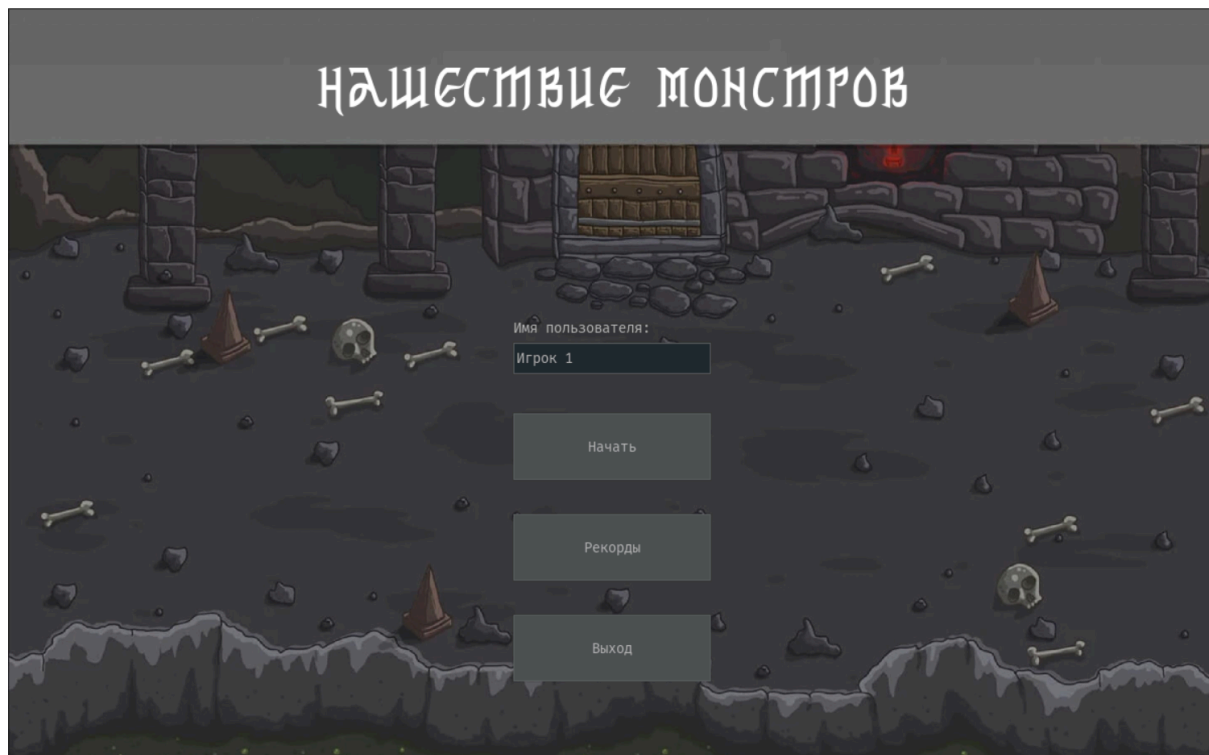
Содержимое базы данных находится в файле db.sqlite (указывается в settings.DB_PATH).

Описание интерфейса

Программа состоит из пяти окон:

- стартовое окно
- игровое окно
- окно просмотра статистики
- межуровневые окна (выводит правила уровня)
- финальное окно

Стартовое окно



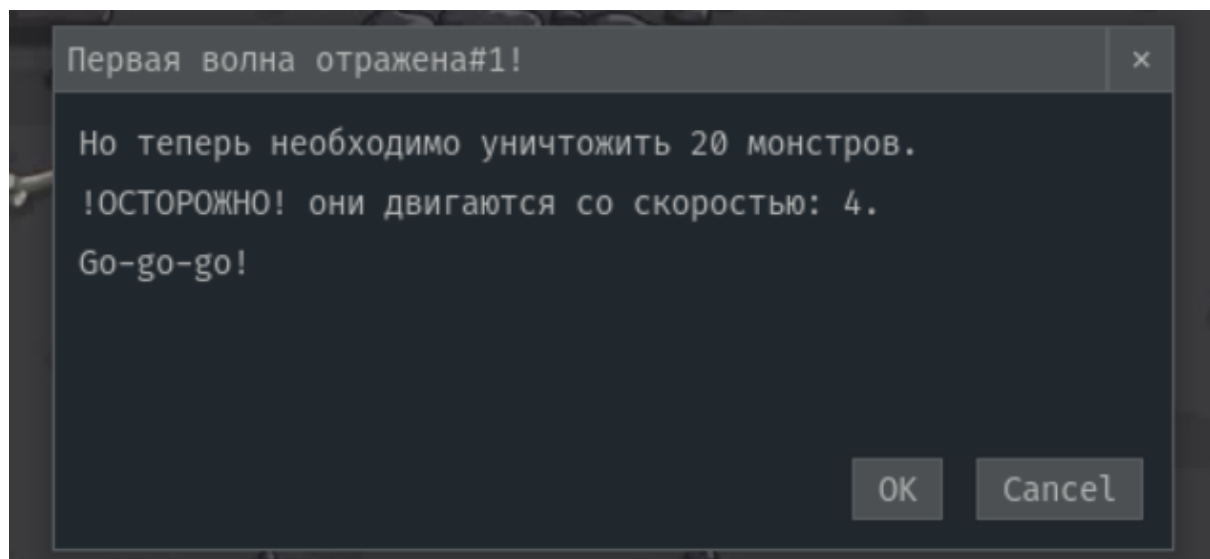
Игровое окно



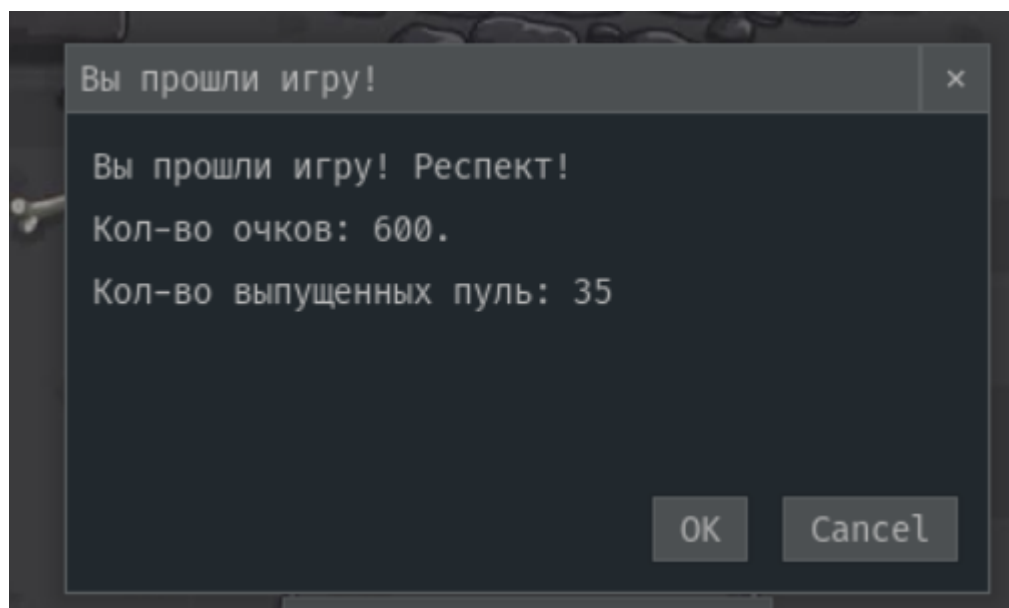
Окно просмотра статистики

id	пользователь	дата	очки	пули
1	Игрок 1	2024-02-04 21:08:09	89	1380
2	brat	2024-02-04 21:09:29	92	1110
4	simple_2	2024-02-04 21:16:01	32	520
3	simple	2024-02-04 21:14:34	27	510
5	Игрок 1	2024-02-07 23:14:17	0	0
6	Игрок 1	2024-02-07 23:19:14	0	0

Межуровневые окна



Финальное окно



Установка зависимостей и запуск

Для запуска программы необходимо установить библиотеки `pygame`, `pygame_gui`:

```
pip install -f .\requirements.txt
```


Структура базы данных

База данных содержит одну таблицу game_stat, в которой есть следующие колонки:

- id - первичный ключ
- user_name - сохранение имен пользователей
- dt - дата окончания игровой сессии
- bullets - кол-во выпущенных пуль за игру
- point - кол-во заработанных очков за игру

Команда для создания таблицы:

```
CREATE TABLE IF NOT EXISTS GAME_STAT (  
    id INTEGER PRIMARY KEY AUTOINCREMENT,  
    user_name TEXT NOT NULL,  
    dt DATETIME NOT NULL,  
    bullets INTEGER NOT NULL,  
    points INTEGER NOT NULL  
)
```