Проект по PyGame «3 в ряд»

Яндекс лицей 2 курс

Элина ГЛУШАНОК, Сергей Евдокимов

Оглавление

[Общее описание 3](#_Toc158046888)

[Интерфейс программы 3](#_Toc158046889)

[Начальное окно 3](#_Toc158046890)

[Окно игры (основное окно) 4](#_Toc158046891)

[Окно окончания (результаты игры) 5](#_Toc158046892)

[P.S. 5](#_Toc158046893)

# Общее описание

«3 в ряд» – общеизвестная игра, геймплей которой заключается в том, чтобы поставить 3 одинаковых фигурки в линию (по вертикали или по горизонтали), меняя 2 соседние по горизонтали или вертикали фигурки разных типов.

В нашей интерпретации игры цель – за наименьшее количество ходов набрать фиксированное количество очков (100).

Для создания интерфейса использован библиотека PyGame, интерфейсы некоторых окон были созданы с помощью программы QtDesigner в формате \*.ui, а также библиотеки PyQT5.

Основной функция игры – game, в ней реализован цикл, который заканчивается по достижении игроком цели в 100 очков. Функция вызывается после ввода ника в начальном окне и открывается окно результатов по окончании игры.

База данных для хранения лучших результатов игроков - Results.sqlite.

В ней всего 1 таблица – result. Архитектура таблицы представлена ниже:

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Эта база данных была создана с помощью программы SQLiteStudio.

# Интерфейс программы

## Начальное окно

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

В этом окне пользователь вводит свой ник и начинает игру.

Это окно реализовано средствами библиотеки PyQ5.

## Окно игры (основное окно)

Изображение выглядит как мультфильм, текст, снимок экрана, млекопитающее

Автоматически созданное описание

В этом окне пользователь выбирает желаемый размер игрового поля из предложенных вариантов (7 на 7, 10 на 10 или 12 на 12).

После выбора формата поля в этом же окне начинается игра.

Сверху выведена актуальная информация по текущему количеству очков, ходов игрока, а также цель игры.

Для того, чтобы поменять 2 фигурки местами, необходимо поочередно нажимать на них.

Если на данный момент выбрана 1 фигурка, она будет подсвечена зеленым.

При ошибке пользователя (если он выбрал фигурки, которые не могут быть поменяны местами по правилам игры), фигурки возвращаются в состояние, которые было перед ошибочным ходом.

Пример возможной игровой ситуации:

Изображение выглядит как текст, снимок экрана, Красочность, шаблон

Автоматически созданное описание

После успешного хода:

1. Увеличивается количество совершенных ходов на 1.
2. Увеличивается количество очков игрока, на количество фигурок, пропавших с поля
3. На месте пропавших фигурок появляются звезды, постепенно летящие вниз (анимация).
4. Место пропавших фигурок занимают их соседи сверху.

Это окно реализовано средствами библиотеки PyGame.

## Окно окончания (результаты игры)

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание

В этом окне пользователь видит свой результат.

Лучший результат пользователя показан деленым в талице.

Можно завершить игру или начать заново.

Это окно реализовано средствами библиотеки PyQ5.

# P.S.

1. Для наглядного отображения данных из БД в таблицы используются SQL-запросы, включающие в себя различные команды, например, select, update, insert и delete с необходимыми инструкциями.
2. Идеи для дальнейшего развития нашего проекта можно посмотреть в презентации к нашему проекту.