2К24

То, что вы так давно ждали и просили. То, от чего волосы встают дыбом и мурашки пробегают по коже. Это наконец свершилось и имя ему…. ЧИТАТЬ ДАЛЕЕ

АРКАНОИД

Ver. 1488

Авторы проекта

1. Святослав Пальцев
2. Денис Кочебин
3. Екатерина Федорова

**Идея проекта "Арканоид"Проект представляет собой игру, в которой уничтожение блоков крутящимся шариком становится метафорой для борьбы с жизненными вызовами и преодоления препятствий, символизирующих неопределенность и случайность.Описание игры:Игрок управляет платформой, отбивая мяч и разрушая блоки, которые представляют собой обстоятельства и проблемы, возникающие в жизни. Чем больше блоков игрок разрушает, тем больше он приближается к победе, но каждый промах мяча напоминает о непредсказуемости и переменчивости судьбы.Скрытый смысл:Уничтожение блоков символизирует преодоление препятствий в жизни. Каждый блок, разбитый мячом, представляет собой преодоленное препятствие, а каждое удачное попадание мяча - шаг к достижению цели. Но при этом игра напоминает, что жизнь полна неопределенности и случайностей: иногда даже малейшая ошибка может повлечь за собой серьезные последствия. Таким образом, игра становится метафорой танца с неизвестностью и вызовом управления своей судьбой в условиях неопределенности.**

**"Мы не просто играем в игры - игры играют с нами." - Платон**

**"Великие изменения в истории обычно начинаются с маленьких действий, выполненных людьми, а не со сложившихся обстоятельств." - Отто фон Бисмарк**

**"Люди строят собственные тюрьмы страхом и жаждой уверенности, а затем прячутся в них." - Владимир Ленин**

**\*\*Описание реализации:\*\***

**Данный проект представляет собой реализацию классической аркадной игры "Арканоид" с использованием библиотеки Pygame в языке программирования Python. Основными классами игры являются `Ball`, `Platform`, `Block`, `Button`, `MainMenu`, `DeathScreen` и некоторые вспомогательные классы для меню и настроек.**

**Класс `Ball` отвечает за поведение мяча в игре, его движение, анимацию и взаимодействие с другими игровыми объектами. Класс `Platform` представляет собой платформу, которой управляет игрок, и определяет ее движение по горизонтали. `Block` создает и управляет блоками на игровом поле.**

**Класс `Button` используется для создания интерактивных кнопок в меню игры. `MainMenu` реализует главное меню с кнопками для начала игры, просмотра рекордов, настроек и выхода из игры. `DeathScreen` представляет экран смерти, который появляется при потере всех жизней в игре.**

**Интересным приемом является использование SQLite базы данных для сохранения и отображения рекордов игроков. Также стоит отметить использование механизма обработки событий в игре, что позволяет корректно реагировать на действия игрока и обновлять игровой мир в соответствии с ними.**

**\*\*Технологии:\*\***

**Для разработки данной игры использовались следующие технологии:**

**- Python: язык программирования для написания игровой логики и взаимодействия с Pygame.**

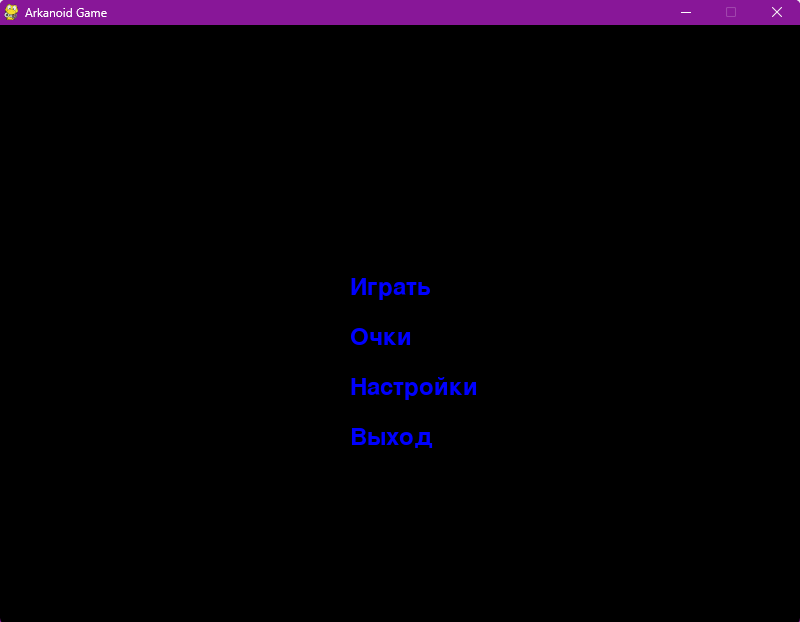
**- Pygame: библиотека для разработки 2D игр в Python, обеспечивающая работу с графикой, звуком и управлением.**

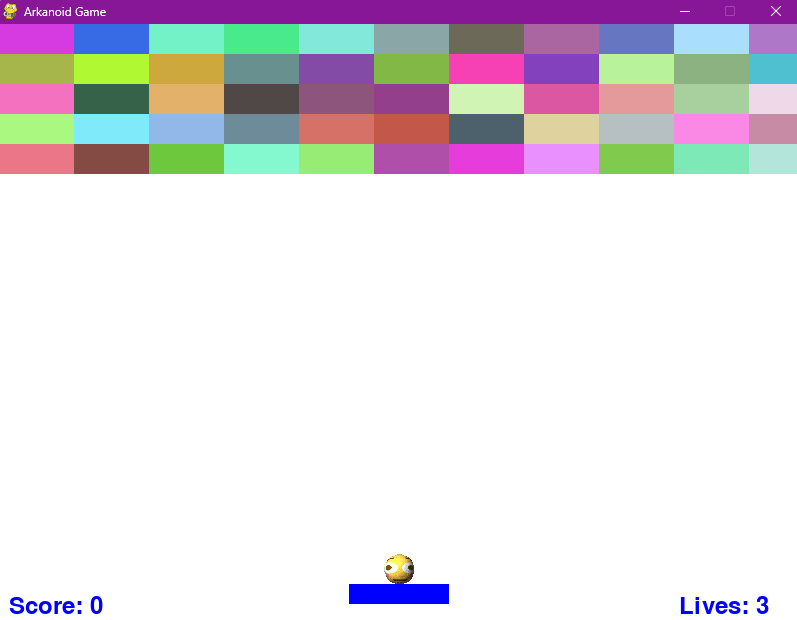
**- SQLite: встроенная реляционная база данных, используемая для хранения рекордов игроков.**

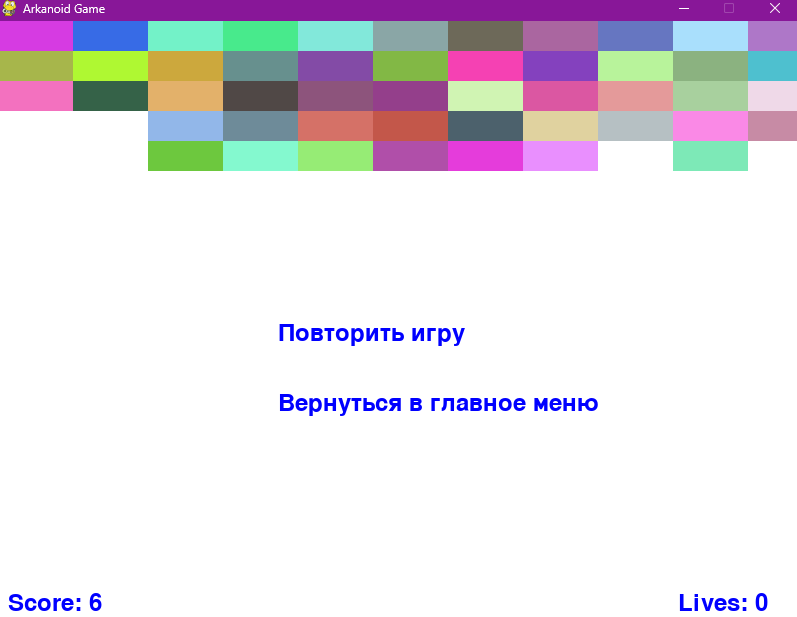
**- Git: система контроля версий для отслеживания и управления изменениями в коде проекта.**

**Для запуска игры необходимо наличие Python и установленной библиотеки Pygame. Код проекта может быть запущен из среды разработки или терминала, после чего игра будет открыта в новом окне Pygame.**

**Скриншоты игры –**

****

****

****