**Пояснительная записка к игровому проекту "Sonic Runner"  
издатель: Алямовский Савва**

**Общая информация о проекте:** Проект представляет собой двухмерную бесконечную бегалку, главным героем которой является персонаж Соник. Игрок управляет Соником, избегая препятствий и набирая очки. Главная цель — набрать как можно больше очков, соревнуясь с собственным рекордом. Игра разработана на языке Python с использованием библиотеки Pygame, а для хранения результатов используется база данных SQLite.

**Основные особенности проекта:**

1. **Главный герой (Соник):**
   * Возможности персонажа: бег, прыжок, уклонение.
   * Реализованы анимации для каждого из движений.
2. **Препятствия:**
   * Три типа препятствий: низкие, высокие и летающие (драконы).
   * Динамическое создание и удаление препятствий по мере прохождения.
   * Проверка столкновений героя с препятствиями.
3. **Элементы фона:**
   * Непрерывно прокручивающийся фон для создания эффекта движения.
   * Движущиеся облака для придания реалистичности.
4. **Система очков:**
   * Нарастающая система подсчёта очков.
   * Постепенное увеличение скорости игры по мере роста очков.
5. **Интеграция базы данных:**
   * Хранение результатов в базе данных SQLite.
   * Отображение лучшего результата для соревнований.
6. **Главное меню:**
   * Удобный интерфейс для начала игры или повторного запуска.
   * Вывод текущего и лучшего результата после завершения игры.

**Структура кода:**

1. **Глобальные константы и ресурсы:**
   * Настройки окна, загрузка ресурсов (изображения, шрифты) и инициализация базы данных.
   * Подготовка изображений для Соника, препятствий, облаков и фона.
2. **Классы:**
   * SonicCharacter: Управление состояниями героя (бег, прыжок, уклонение), анимации и проверка столкновений.
   * SkyCloud: Движение и отображение облаков.
   * Obstacle и его подклассы (SmallObstacle, LargeObstacle, FlyingDragon): Типы препятствий и их поведение.
3. **Игровая логика:**
   * Основной игровой цикл main\_game, отвечающий за обновление и отрисовку всех элементов.
   * Функции для обновления очков, прорисовки фона и обработки ввода от пользователя.
4. **Операции с базой данных:**
   * save\_score\_to\_database: Сохранение текущего результата в базу данных.
   * get\_high\_score: Получение наивысшего результата из базы данных.
5. **Главное меню:**
   * Инструкции для игрока, отображение результатов и управление игрой.

**Преимущества и уникальные особенности:**

* Динамическое изменение сложности за счёт увеличения скорости игры.
* Модульный подход в реализации: отдельные классы для различных элементов игры упрощают её расширение и поддержку.
* Использование базы данных SQLite для долговременного хранения результатов.

**Инструкция по запуску игры:**

1. скопируйте директорий SonicRun
2. Установите библиотеку Pygame:
3. Выполните команду pip install pygame
4. в папке проекта двойной клик по main.py

**Возможности для дальнейшего развития:**

* Добавление игровых бонусов (например, временная неуязвимость или удвоение очков).
* Внедрение звуковых эффектов и фоновой музыки для создания более захватывающей атмосферы.

Проект демонстрирует базовые принципы разработки игр с использованием Python и Pygame, предлагая хорошую основу для дальнейшего изучения игрового программирования.