**Пояснительная записка**

**к проекту PyGAME**

**Крюкова Дарья**

**Оренбург**

**2024**

**СОДЕРЖАНИЕ**

[Введение 3](#_Toc159969767)

[Практическая часть 4](#_Toc159969768)

[Создание карты 4](#_Toc159969769)

[Код 7](#_Toc159969770)

[Support.py и game\_data.py 7](#_Toc159969771)

[Decoration.py, particles.py, tiles.py, ui.py, player.py, overworld.py, enemy.py 9](#_Toc159969772)

# Введение

Задачей проекты было создание игры-платформера.

Техническое задание:

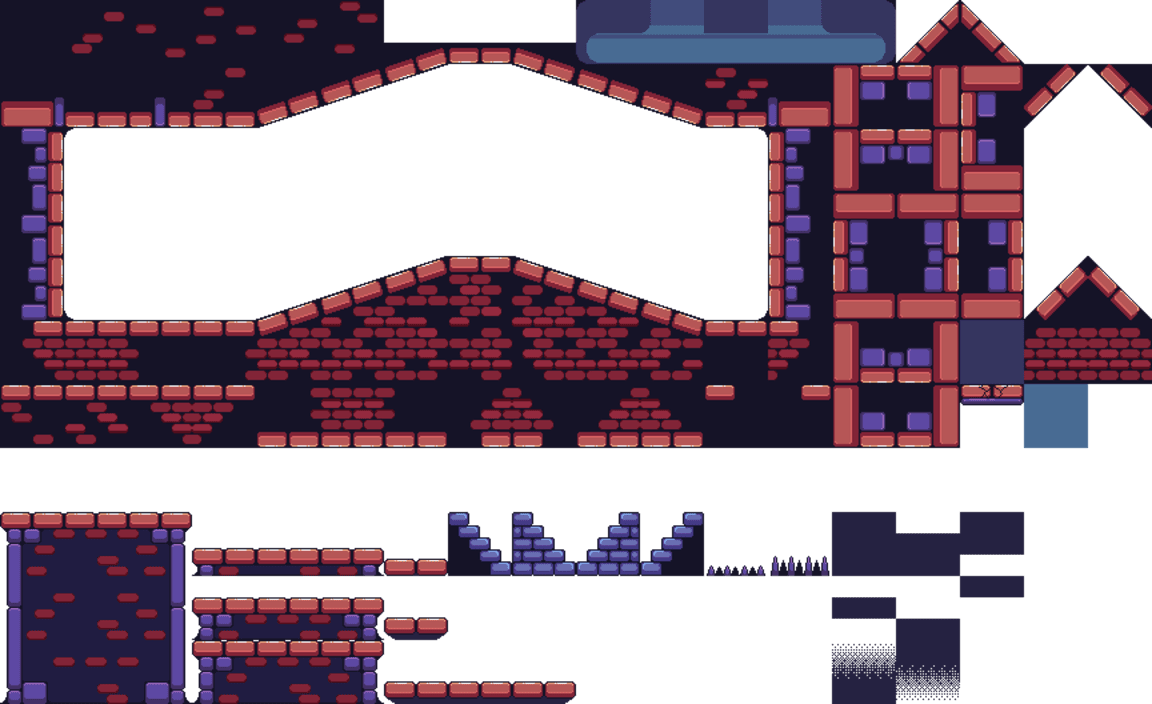
1. 3 уровня(на каждом есть острова, монеты, враги)
2. Хранение файлов в формате \*.csv
3. Спрайты персонажей, монет имеют анимации

# Практическая часть

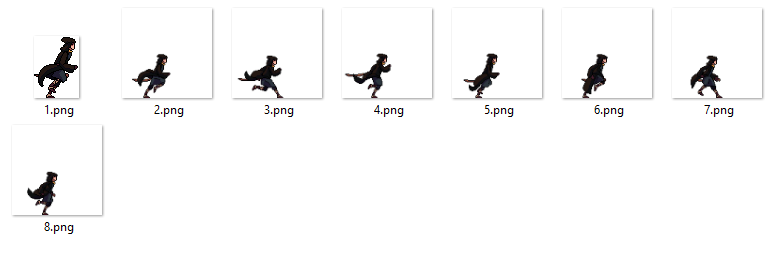
## Создание карты

В работе использовала бесплатные текстуры с сайта <https://itch.io/>

**Для карты(ящик и текстуры для островов)**

crate.png

Для главного персонажа(для примера содержание папки …\character\run)

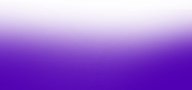


Для монет(изумруды и монеты)

1.png1.png

Для заднего фона

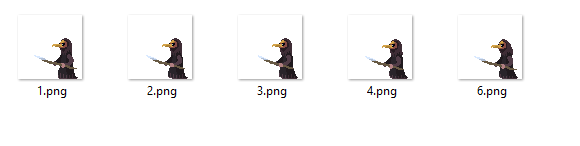
Вода



Картинка для уровня



Враг(на примере анимация бега)

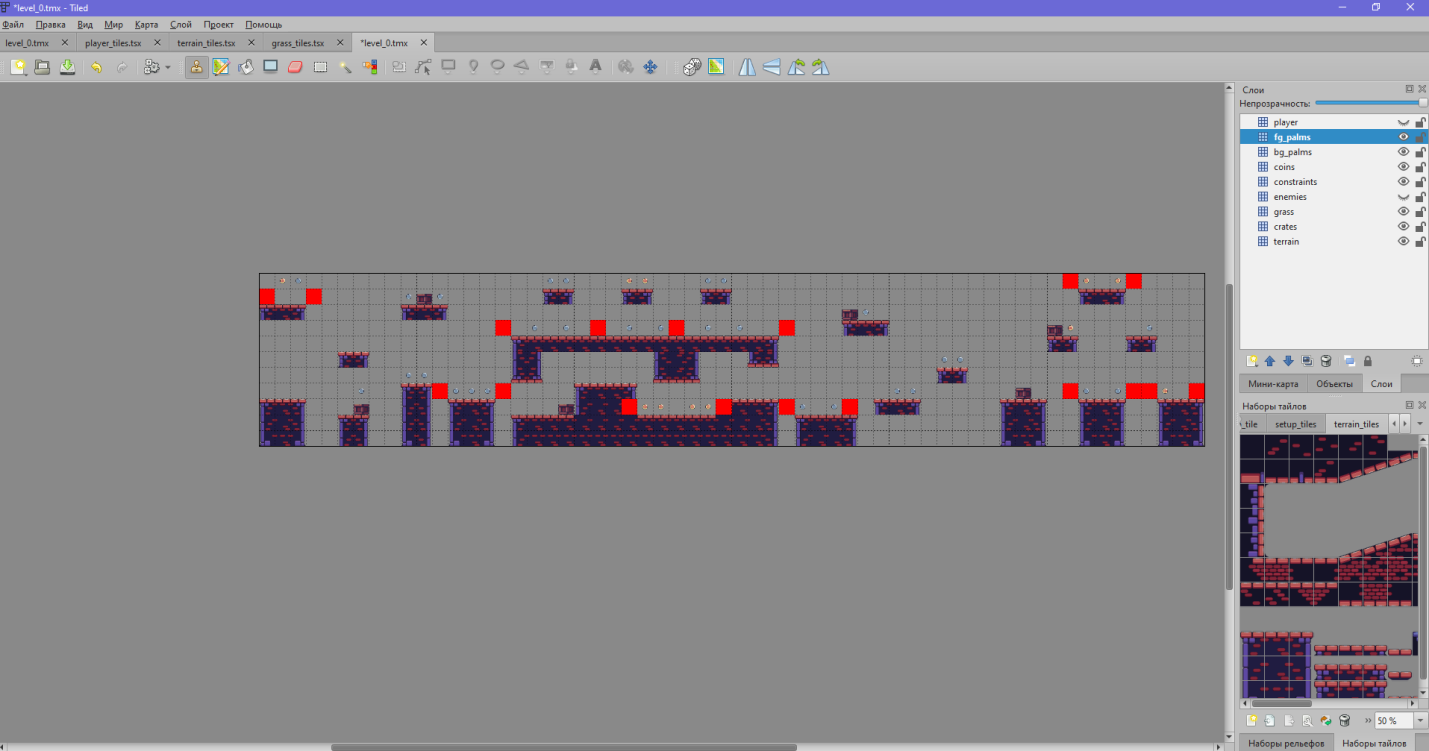


Для создания карты я использовала TILED(отлично подходит для создания пиксельных карт)

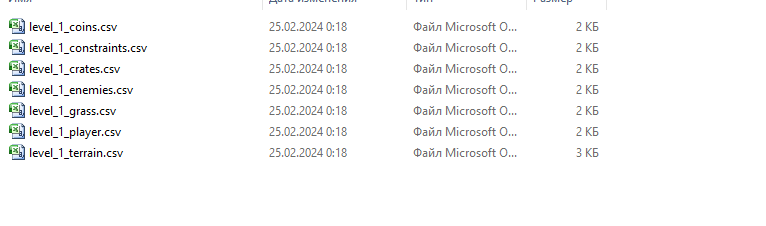
Весь принцип работы:

* Определение png картинки, как tmx файла и текстуры
* Создание карты и размещение на ней необходимых квадратов текстур
* Рапределение каждого типа текстур(terrain, coins, player и т.д.) по своим слоям

к примеру устройство последнего уровня:

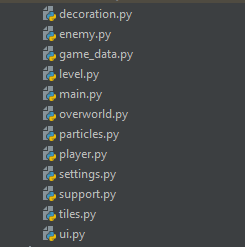


В итоге, экспортируем все слои в виде csv файла, содержание папки уровня будет таким



## Код

Содержание проекта



### Support.py и game\_data.py

game\_data.py – словари с адресом файлов для каждого уровня, его позицией на стартовом экране

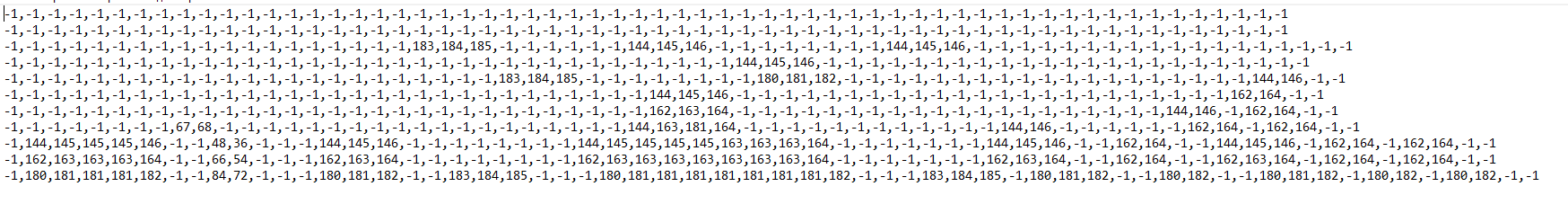
Пример для первого уровня:



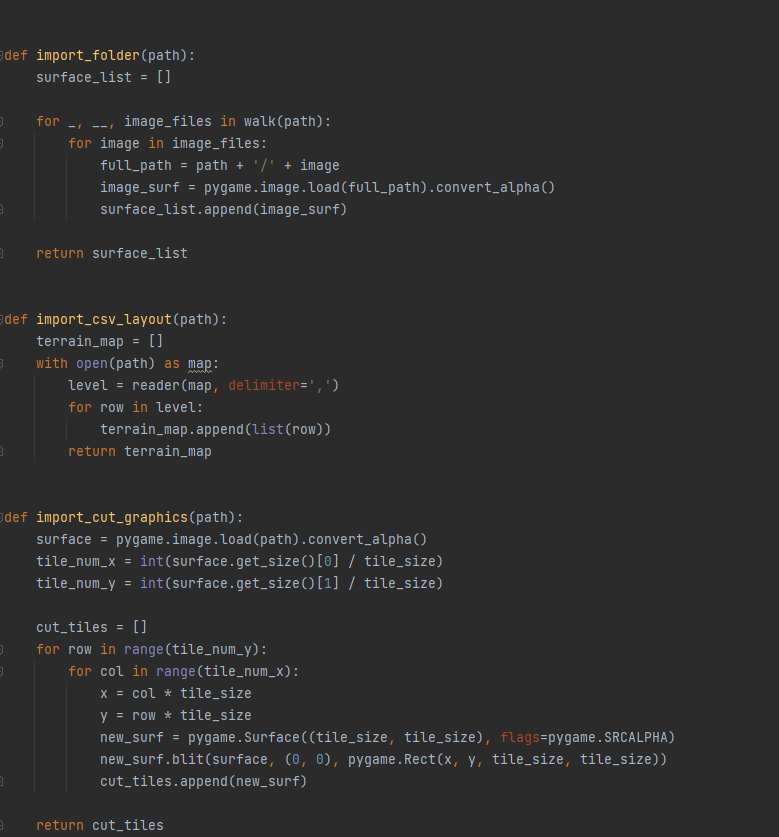
Support.py нужен для распаковки csv файлов.

Каждый csv файл содержит код элемента определенного слоя(либо -1(это пустая клетка, либо код элемента в сетке текстур))

Пример содержания:

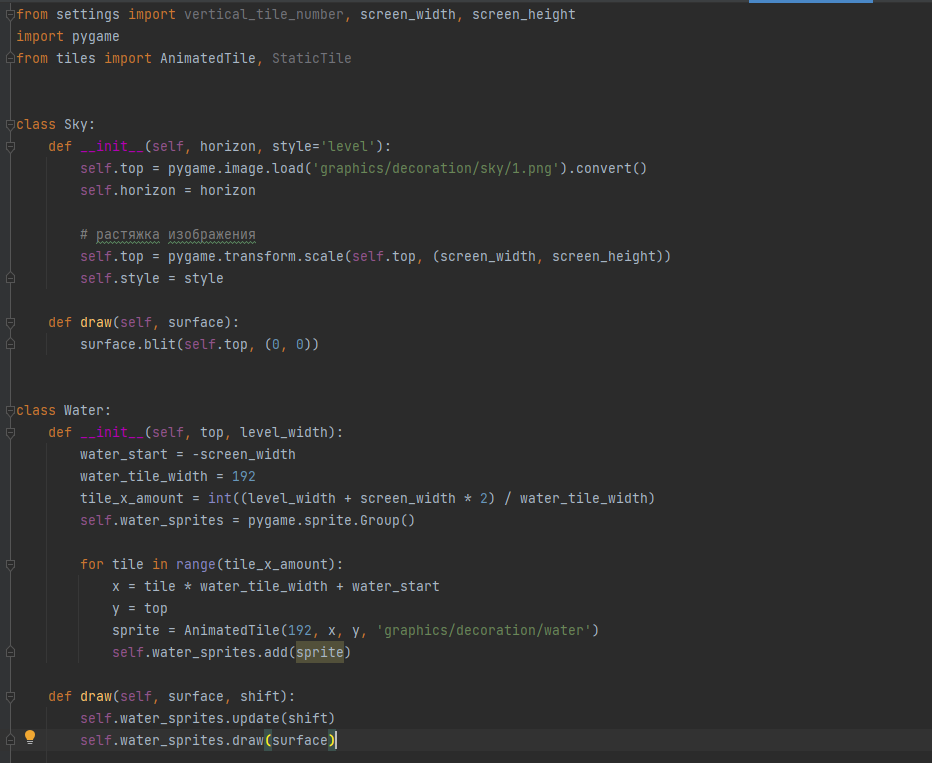


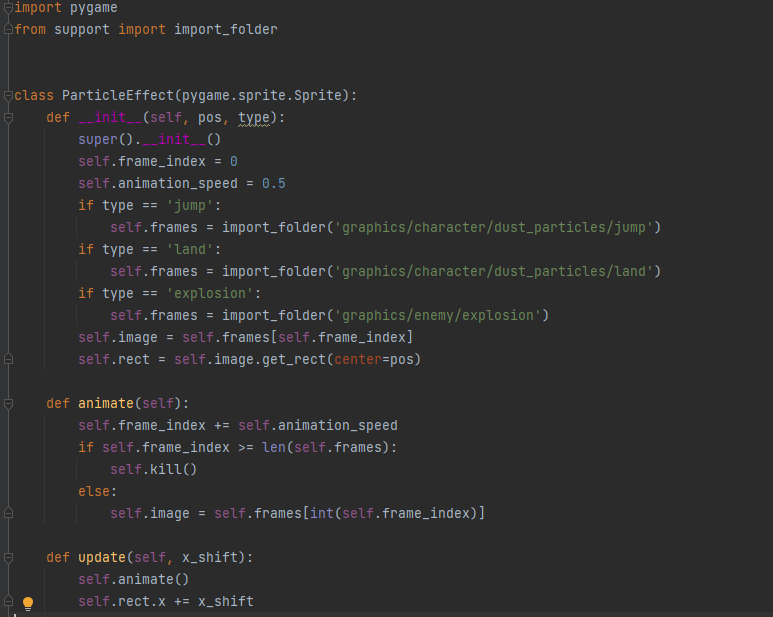
Код, преобразующий csv файлы



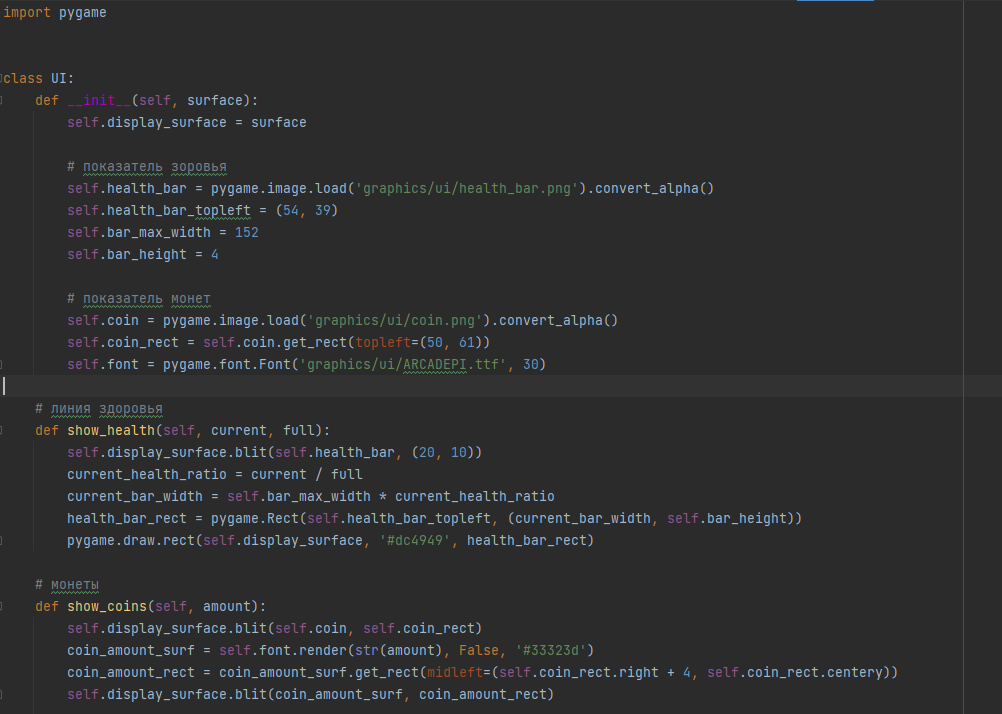
### Decoration.py, particles.py, tiles.py, ui.py, player.py, overworld.py, enemy.py

Decoration.py – заливка заднего плана изображением sky и заливка изображением water экрана снизу

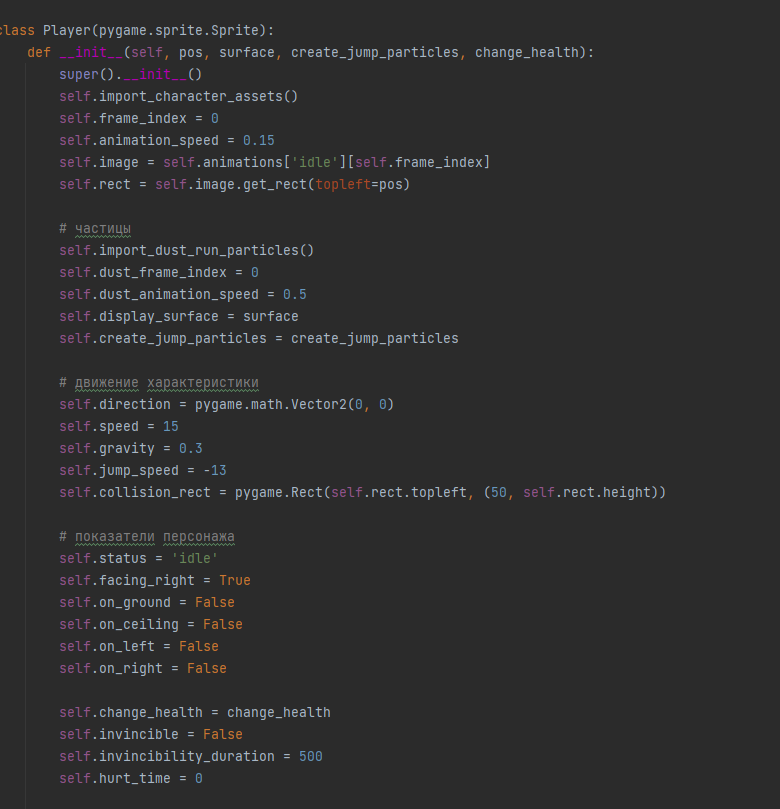


particles.py-работа с анимацией частиц для персонажа и врагов при событиях 

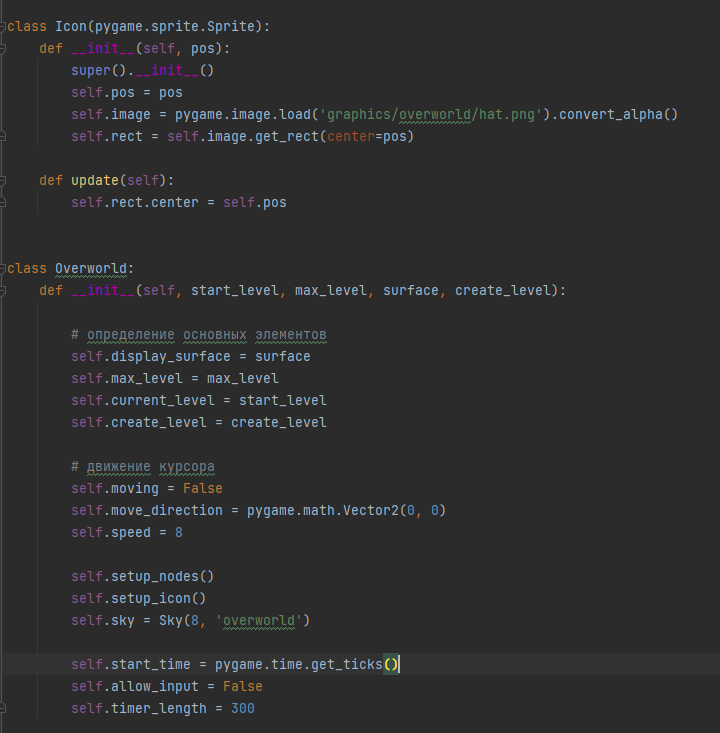
ui.py



Player.py – определение состояния модели персонажа, столкновение, характеристики и тд



overworld.py – настройка стартового окна





Enemy.py – настройка врагов



Main.py

