Pokemon Walk Simulator. Documentation

Table of contents

Program description

The program was created for the entertainment of the player, a fan of the Pokemon universe.

It is a turn-based game in which one player with a team of Pokemons fights against another player (or computer) with his team of Pokemons.

Before the fight begins, players choose Pokemon from the suggested list. During the game, the player makes various moves, to choose from: defense boost, normal attack, special attack, replacement of the current active Pokemon.

The player wins when all the opponent's pokemons leave the fight (their hp = 0).

Program structure

The program is written in python using OOP. Accordingly, there is a division into classes:

- **Simulator**

The heart of the program. It supports the player's movements, defines the order of movements, how they are handled and related things. It is possible to set the game mode: player vs. player or player vs. computer, by specifying the option from the file config.py. It also counts down the number of rounds, controls the list of pokemons available to the player.

- **Player**

Player. The same player class that is responsible for storing and handling player data. It is possible to add pokemons to a team, delete them, set a move, comment on the movement, set the current active pokemon.

- **Computer** (inherited from the player)

The opponent is the computer. It emulates the player's behavior (currently performs random actions). He can randomly choose a move, choose a Pokemon for the team.

- **Pokemon**

Combat unit. Describes a pokemon. As static methods, it has defined functions for retrieving Pokemon data from a json file, calculating attack damage, downloading all Pokemon, downloading Pokemon by name. There is also an option to retrieve Pokemon image data from a third-party api (https://pokeapi.co/).

- **Move**

A class that describes the player's movement. Specify the type of traffic, if necessary additional data.

- **PokemonFightSimulatorWindow**

GUI window class.

There is also a config.py file where you can specify the path to a music file that can be used in the program, or to a json file with Pokemon data.

Instructions for use

To run the program, you must run the main.py file from the root directory in the python interpreter. Note that python 3.10 or higher must be used to run the program.

Work done

The project has been in development for 3 weeks. An analysis of the Pokemon theme, experiences from previous projects and gameplay mechanics was carried out. Work was carried out to rethink the architecture, structure and organization of the project. I attach diagrams of the structure of the project.

Most of the time, of course, took to write the code and test it. Unfortunately, I still have a grudge against the written code, it still needs to be optimized, which I may do in the future, gaining more experience in writing programs.

The content of the project's tasks also included data on Pokemon, which were taken into account during the implementation.

In general, all the initial requirements of the project have been met. I liked the end result, plus I gained experience with the GUI, which I find very cool.

Possible functionality to add

1. Online multiplayer
2. Animation of pokemon movements.
3. Personal profile system, collecting Pokemon, upgrading them, exchanging them, etc.
4. Sound effects
5. Improved algorithm or artificial intelligence of the opponent

Design diagrams