RPG Battle Simulator

Introduction

For my final project, I created a turn-based RPG battle simulator in C++. I chose this project because I'm interested in gaming and wanted to build something personal but still useful for practicing programming. I also wanted a project that I could keep working on after the course, either by expanding the mechanics or adding a user interface.

Overview

The RPG Battle Simulator is a console based game where the player and an enemy take turns attacking each other. The player can choose to attack or use an item. The enemy always attacks. The match ends when one character's health drops to zero. A log of the battle is saved to a text file.

Key C++ Concepts

This project uses the following core topics:

- Classes and Inheritance:

Character is an abstract base class. Player and Enemy inherit from it and override the takeTurn function.

- Pointers:

Character interactions are handled using pointers. When one character attacks another, a pointer to the target is passed.

- Streams:

The game writes a full battle log to a file (battle_log.txt) using ofstream.

- Error Handling:

The code checks for invalid input (e.g. letters instead of numbers when choosing an action).

- Modular Design:

All logic is organized into clear, focused classes and functions.

Challenges

The biggest challenge was designing the interaction between characters using pointers while keeping the code clean and readable. I also had to learn how to structure the project into multiple files and set up basic unit tests. Once the base code worked, adding features like item use and file logging was more straightforward.

Testing

I wrote a separate test file (test/tests.cpp) that verifies:

1. Characters are created correctly.
2. Damage calculations work as expected.
3. A simulated battle ends with the correct result.
4. All tests passed and helped confirm the code works reliably.

GitHub Repository

https://github.com/mohharara/RPG-Battle-Simulator-

Conclusion

This project helped me apply the most important concepts from this course in a way that felt relevant to my interests. I now understand how to use inheritance, file streams, and dynamic memory more confidently. I plan to keep improving the project by adding more enemy types and saving player progress!