

Link do
projektu: <https://github.com/PrzemekTy/PythonProjectTextRpg>



Projekt podstawy programowanie 1

Autor: Przemysław Typa s1 INF 3(3)u

Pod patronatem mgr. Inżyniera Marcina Tracza.



Dokumentacja projektu

- Wstęp: Gra przedstawia fabułę, w której gracz jest zatrudniony przez króla do odsieczy księżniczki, która jest uwięziona przez smoka. Gracz musi pokonać różne przeszkody i bossów, aby dotrzeć do smoczej jaskini i uratować księżniczkę. Gra oferuje różne ścieżki, które mogą być obrane przez gracza, a także zbieranie przedmiotów, które mogą pomóc graczowi w pokonaniu przeciwników.
- Moduły używane w grze: 'os' i 'sys' służą do obsługi systemu operacyjnego. 'time' jest używany do opóźniania ekranów.
- Zmienne: 'a', 'b' i 'c' służą do opóźniania ekranów w module 'time'.
- Funkcje: 'prompt()' jest używana do wyświetlenia menu startowego gry. 'clear()' jest używana do wyczyszczenia ekranu.
- Mapa: 'rooms' to słownik, który przechowuje informacje o pokojach w grze, w tym położenie pomieszczeń, przedmiotów i bossów.
- 'current_room' to zmienna, która przechowuje informacje o aktualnie przebywanym pokoju przez gracza.

- Inwentarz: 'inventory' to lista, która przechowuje informacje o przedmiotach zdobytych przez gracza.
- Pętla gry: Pętla while jest używana do gry, która kończy się tylko po zakończeniu zadania przez gracza. Wewnątrz pętli są warunki, które w zależności od aktualnie przebywanego pokoju wyświetlają odpowiednie informacje i opcje dostępne dla gracza.
- Zakończenie: Gracz kończy grę po dotarciu do smoka, pokonaniu go. Gracz może także zakończyć grę przed ukończeniem zadania, decydując się na opuszczenie jaskini(Exit), lub ginąć np. przegrywając w walce z którymś z bossów.
- Implementacja: Gra jest napisana w języku Python i korzysta z konsoli do interakcji z graczem. Gracz może poruszać się po mapie, korzystając z komendy 'go' i wskazując kierunek. Gracz może także zbierać przedmioty, korzystając z komendy 'get/take' i podając nazwę przedmiotu. Gracz jest również zmuszony do rozwiązania zagadek, aby przejść dalej.

Testowanie: Gra powinna zostać przetestowana przez kilku graczy, aby upewnić się, że działa poprawnie i jest bezbłędna. Testowanie powinno obejmować sprawdzenie funkcjonalności wszystkich komend i opcji dostępnych dla gracza. Testowanie powinno również obejmować sprawdzenie poprawności zagadek i walk z bossem.

Dalsze rozwijanie: Gra może być rozwijana poprzez dodanie nowych pokoi, przedmiotów i bossów. Gra może być również rozwijana poprzez dodanie nowych funkcjonalności, takich jak system ekwipunku, rozwijanie postaci gracza poprzez dodanie punktów życia/many/czarów oraz zaimplementowanie walki łącznie z (import random). Gra może być także rozwijana poprzez dodanie grafiki i dźwięku, aby zwiększyć immersję gracza.

Wymagania systemowe: Aby uruchomić grę, wymagany jest komputer z systemem operacyjnym Windows, macOS lub Linux oraz interpreter języka Python. Gra jest przeznaczona do użytku z konsolą, więc nie wymaga specjalnego sprzętu graficznego ani dźwiękowego. Gra powinna działać na większości komputerów osobistych bez konieczności dokonywania dodatkowych ustawień.

Getting started:

Typ pliku: Python Script

Czym są pliki PY?:

Plik PY jest plikiem programu lub skryptem napisanym w języku Python, zinterpretowanym językiem programowania obiektowego. Można go tworzyć i edytować za pomocą edytora tekstowego, ale do uruchomienia potrzebny jest interpreter języka Python. Pliki PY są często używane do programowania serwerów internetowych i innych administracyjnych systemów komputerowych.

Windows:

Programy, które otwierają pliki PY:

UWAGA: Język programowania został pierwotnie stworzony pod koniec lat 80. przez Guido van Rossum z kolejnymi wersjami wydanymi w 2000 (Python 2.0) i 2008 (Python 3.0).

Python został zaprojektowany tak, aby był łatwy do odczytania i prosty w implementacji. Jest to oprogramowanie typu open source i służy do opracowywania szerokiej gamy bezpłatnych i komercyjnych aplikacji, takich jak Bazaar, Blender, Pylons i Panda3D.

GNU Emacs Microsoft Notepad ES-Computing EditPlus Richardson EditRocket Notepad++ Eclipse Sublime Text Python Software Foundation Python ActiveState Komodo Edit IDM UltraEdit Scribus

Linux:

GNU Emacs Richardson EditRocket gedit Eclipse Sublime Text Python Software Foundation Python ActiveState Komodo Edit IDM UltraEdit Scribus

Macintosh:

MacroMates TextMate MacVim GNU Emacs Richardson EditRocket Eclipse Sublime Text Python Software Foundation Python ActiveState Komodo Edit IDM UltraEdit Scribus