

Pràctica I – Programació II

Programació orientada a objectes

En aquest document s'expliquen tots els mètodes i classes utilitzats per resoldre l'exercici proposat a classe. Els atributs de les classes no s'han fet privats perquè es van crear aquestes els primers dies de classe i encara no s'havia explicat que s'havia de fer d'aquesta manera. Per extreure o modificar qualsevol valor d'un atribut d'una classe es fa mitjançant sets i gets. En cas que s'hagués de fer, es podrien modificar totes les variables del programa, però s'ha pensat que la funcionalitat d'aquest programa serà la mateixa tant si les variables són públiques o privades. Per a futures pràctiques ja es començarà amb tots els atributs privats.

Per fer la importació de les partides s'ha d'importar el fitxer d'elements i el de jugadors. A dins del directori com es mostra a l'exemple. Els fitxers de jugadors han d'estar a dins de la carpeta de jugadors i els elements a dins de la d'elements. Aquets es mostraran per pantalla directament i l'usuari tria el que vulgui.

Ha faltat la part de guardar la partida, però s'ha volgut prioritzar la part d'assolir els conceptes donats i no voler fer feina extra.

Classe Partida:

__init__

- Jugadors: Diccionari on es guarda el nom de cada jugador en una key i el valor d'aquesta és la classe del jugador.
- Elements: Diccionari on es guarda el nom de cada element que hi ha actualment a la partida/mapa (menys el que tenen els jugadors), cada key és el nom de l'element i la variable d'aquesta conté la classe de l'element i la quantitat d'elements iguals.
- Partides: llista de llistes de fitxers de jugador i elements. Només es el nom dels fitxers . [['jugadors.txt',...], ['elements.txt',...]]

delet_player

- Entrades: Nom del jugador
- Sortida: ---
- Funció: Si existeix el jugador en la partida l'elimina

add_player

- Entrades: Nom i qualitats del jugador
- Sortida: ---
- Funció: Afegir un nou jugador a la partida (si no n'existeix un amb el mateix nom) i amb les qualitats assignades.

add_element

- Entrades: Nom, preu, fungible i quantitat de l'element.
- Sortida: ---

Roger Colomer

Arnau Tomasa

Enginyeria mecatrònica

- Funció: Afegir un nou element a la partida (si no existeix) amb les qualitats entrades.

dec_salut

- Entrades: Nom del jugador i salut a disminuir.
- Sortida: ---
- Funció: Es comprova la quantitat de vida del jugador seleccionat i si la resta amb la vida que es vol disminuir és superior a 0 es resta la vida. Si la vida resultant és 0 o inferior aquest s'elimina de la partida.

buy_element

- Entrades: Nom del jugador i element que es vol comprar.
- Sortida: ---
- Funció: Si existeix el jugador a la partida, té suficients diners i queden elements lliures del que vol comprar, el jugador incorporarà l'element comprat.

use_element

- Entrades: Nom del jugador i element a utilitzar.
- Sortida: ---
- Funció: Si existeix el jugador i l'element, i el jugador té aquest element l'utilitza. Seguidament es comprova si l'element és fungible o no, en cas que ho sigui s'elimina del jugador.

comp_player

- Entrades: Nom del jugador
- Sortides: Variable booleana
- Funció: Comprova si existeix el jugador entrat a la partida. En cas que existeixi es retorna una variable True.

comp_element

- Entrades: Element
- Sortides: Variable booleana
- Funció: Comprova si existeix l'element entrat a la partida. En cas que existeixi es retorna una variable True.

read_file

- Entrades: Nom de la ruta
- Sortides: Llista amb totes les dades llegides
- Funció: Llegir els fitxers de jugadors i elements

read_elements

- Entrades: Nom del fitxer d'elements que es vol llegir
- Sortides: ---
- Funció: crida a la funció read_file, i crea les classes per cada element, guardant les qualitats i la quantitat.

read_players

Roger Colomer

Arnau Tomasa

Enginyeria mecatrònica

- Entrades: Nom del fitxer de jugadors que es vol llegir
- Sortides: --
- Funció crida a la funció read_file, i crea les classes per cada jugador, guardant les qualitats

__str__

- Entrades: ---
- Sortida: valor a imprimir
- Funció: Retorna una string per imprimir tota la partida per pantalla

Classe Jugador:

__init__

- Entrades: nom i qualitats del jugador
- Atributs:
 - o Nom del jugador
 - o Salut del jugador (inicialment a 50)
 - o Riquesa del jugador (inicialment a 100)
 - o Llista d'elements que posseeix el jugador

get_sal

- Entrades: ---
- Sortides: Salut del jugador
- Funció: Retornar la salut del jugador

get_player

- Entrades: ---
- Sortides: Nom, salut, riquesa, qualitats i elements
- Funció: Retornar els elements anteriors.

get_e_player

- Entrades: ---
- Sortides: Llista d'elements d'un jugador
- Funció: Retornar els elements del jugador

get_riquesa

- Entrades: ---
- Sortides: Riquesa del jugador
- Funció: Retornar la riquesa del jugador

set_riquesa

- Entrades: Riquesa del jugador
- Sortides: ---
- Funció: Canviar la riquesa del jugador

set_salut

- Entrades: Salut del jugador

Roger Colomer

Arnau Tomasa

Enginyeria mecatrònica

- Sortides: ---
- Funció: Canviar la salut del jugador

add_e

- Entrades: Element
- Sortida: ---
- Funció: Afegir l'element al jugador.

del_e

- Entrades: Element
- Sortida: ---
- Funció: eliminar l'element del jugador.

Classe Element:

init

- Entrades: Nom, preu, fungible
- Atributs:
 - o Nom
 - o Preu
 - o Fungible

set_e

- Entrades: Nom, preu, fungible
- Sortides: ---
- Funció: canviar els valors d'un element

get_e

- Entrades: ---
- Sortides: Nom, preu, fungible
- Funció: Retornar el nom, preu i fungible de l'element.

Main:

Aquest fitxer incorpora una màquina d'estats:

- Estat 0 → inicialització
- Estat 1 → importar partida
- Estat 2 → afegir jugador
- Estat 3 → eliminar jugador
- Estat 4 → disminuir vida jugador
- Estat 5 → utilitzar element jugador
- Estat 6 → comprar element jugador
- Estat 7 → mostrar partida
- Estat 8 → sortir del joc
- Estat 9 → Mostar menú

Roger Colomer

Arnau Tomasa

Enginyeria mecatrònica

Cada un d'aquets estat és una funció. I aquets utilitzen mètodes de la classe partida. Aquestes funcions només fan la gestió d'entrades i mostrar missatges per pantalla.

Inicialització

Aquesta funció serveix per començar el joc, demana si es vol importar una partida o sortir del joc.

Importar partida

Serveix per importar partides de jugadors i d'elements. Primer de tot mostra els fitxers .txt de la carpeta jugadors del directori, i permet triar una opció. Seguidament fa el mateix però amb la carpeta elements. Un cop s'ha executat es mostra el menú principal.

Afegir un jugador

S'entren per pantalla el nom del jugador i les qualitats d'aquest, si el jugador no existeix i les qualitats estan entre 0 i 100 s'uneix el jugador a la partida. Un cop s'ha executat es torna a mostrar el menú.

Eliminar un jugador

S'entra per pantalla a el nom del jugador que es vol eliminar i si existeix a la partida s'elimina. Un cop s'ha executat es torna a mostrar el menú.

Disminuir salut d'un jugador

S'entra per pantalla el nom del jugador i la quantitat de vida que se li vol restar. Un cop s'ha executat es torna a mostrar el menú.

Utilitzar element

S'entra per pantalla el nom del jugador i l'element que vol utilitzar. Un cop s'ha executat es torna a mostrar el menú.

Comprar element

S'entra per pantalla el nom del jugador i l'element que es vol comprar. Un cop s'ha executat es torna a mostrar el menú.

Mostrar partida

Es mostra per pantalla en l'estat que es troba la partida en aquest precís moment.

A part el programa principal també té algunes funcions alternatives que utilitzen algunes de les funcions anteriors.

Show files

Mostra els arxius .txt a dins del directori seleccionat per la crida de la funció.

Comp int

Comprovador que el valor entrat és un enter i està comprès entre 0 i 100.