Taller de herencia y polimorfismo

Programación orientada a objetos 2023-2

**Objetivo**:

Aplicar los conceptos de herencia y polimorfismo por medio de la creación de un pequeño juego de rol (RPG) en el que diseñarán personajes y demostrarán los conceptos por medio de la implementación de los personajes y el uso de habilidades de combate de los mismos.

**Condiciones de entrega**:

Se debe entregar un archivo python **.py** que se pueda ejecutar **sin ajustes adicionales**. En caso de utilizar assets o recursos como imágenes o audio (.jpg, .png, .wav …..) asegúrese de entregar una carpeta comprimida que contenga el **.py** y los recursos necesarios.

*\* Nota: No olvide incluir en su código los comentarios que crea necesarios*

**Taller:**

1. **Diseño de Clases:**

Defina una clase base(superclase) llamada "*Personaje*" que incluya los atributos “nombre”, “nivel” y “puntos\_vida”.

Defina dos subclases o clases hijas(derived), "*Mago*" y "*Guerrero*", que hereden de la clase "*Personaje*".

Agregue atributos específicos para cada clase:

* Mago: "*mana*"
* Guerrero: "*fuerza*"

Defina un método "*morir*()" y otro "*atacar*()" en la clase "*Personaje*", los cuales reflejarán las acciones comunes de muerte y de ataque para todos los personajes.

**2. Polimorfismo en Acción**:

Implemente el método "*atacar*()" en las clases "*Mago*" y "*Guerrero*". Cada clase debe tener una implementación diferente del método para reflejar sus habilidades de combate únicas.

* Mago: Deberá lanzar bola de fuego
* Guerrero: Deberá dar un hachazo

**3. Creación de Personajes:**

En una clase principal “*MyWindow*” realice la instancia de múltiples personajes, tanto magos como guerreros, definiendo atributos diferentes para cada uno.

**4. Simulación de Combate:**

En la clase “*MyWindow*” y con las diferentes instancias de Personaje creadas realice una simulación de combate, dicha simulación puede ser implementada siguiendo turnos secuenciales de acuerdo a cada instancia realizada en el paso previo.

Cada Personaje realizará su acción de ataque por medio del método “*atacar*()” sin embargo debe ser posible evidenciar las habilidades propias y diferentes tanto de magos y guerreros aún invocando el mismo método. De igual manera se deberá consumir el método “morir()” el cual deberá retirar dicho personaje del escenario, esta última acción deberá ocurrir de la misma manera para todas las instancias de las subclases ya que no fue modificada por polimorfismo.

**4. Personalización (Opcional):**

Siempre que cumpla las condiciones establecidas previamente en las clases obligatorias (Personaje, Mago, Guerrero), los atributos definidos para la superclase y subclases, y la herencia y polimorfismo entre ellas, realice las personalizaciones que vea convenientes, nuevas subclases deseadas, interacciones de combate extra, etc.