Message: holds a string, a RGB color value in a Tuple, and a count of how many times a message with the same string has been created in a row

Message
"That way is blocked": plain_text [255, 255, 255]: fg 2: count
+ full_text(): str

MessageLog: holds a list of all messages that have been sent, plus a method for rendering them onto the console.

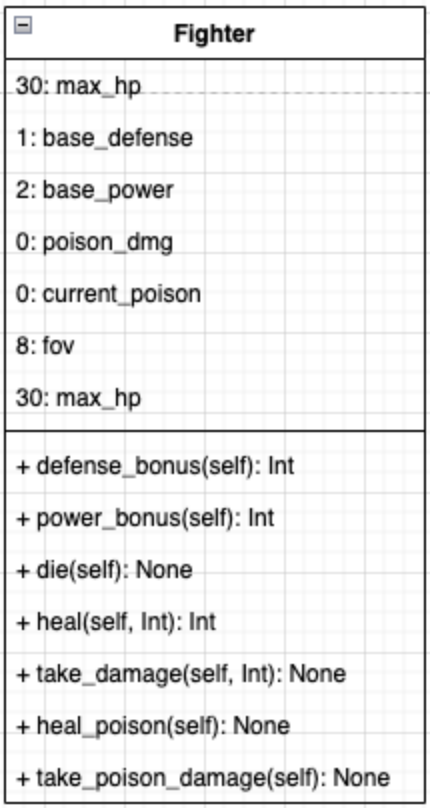
MessageLog
List[Message]: messages
+ add_message(text:str, fg: Tuple[int, int, int], *, stack: bool): None + render(console: Console, x: int, y:int,width: int, height: int): None + wrap(string:str, width: int): Iterable[str] + render_messages(console: Console, x: int, y: int, width" int, height: int, messages: Reversible[Message]): None

Fighter:

The purpose of the fighter class is to provide combat functionality to any Actor who should be capable of fighting. It allows them to track their hitpoints, take damage from various sources, heal damage from various sources, and die.

There are three specializations of Fighter, which represent different player character classes. Throughout the program, there are event hooks that fire functions like "fighter.use_consumable" and "fighter.on_enemy_hit". These functions on the Fighter specialization allow us to have custom class-specific abilities trigger depending on which class the player has selected. These abilities include the warrior's ability to inflict the "confused" condition with a melee attack or the mage's ability to occasionally recover consumed scrolls.

Fighter is not an abstract class, as it is still used directly by enemies in the game.



Entity:

Represents any object that can be placed in the game map. Those objects can be either movable or items.