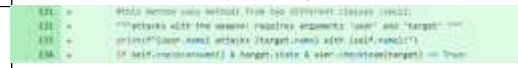
Drew And Matt - Drew Pickett & Matthew Tong	Playability: 9/10 Readability: 8/10 Quality: 8/10 Understanding of Python: 9/10 Overall: 34/40			Success Team	Playability: 8/10 Readability: 7/10 Quality: 8/10 Understanding of Python: 8/10 Overall: 31/40	
1		You could potentially make a basic enemy class in order to avoid repeating the same code if more enemies are added then move the name, enemy class, health, attack, and loot logic to it. This class could be accessed for future enemy types just by calling it rather than retyping everything again		1		
2		Rather than going through and checking for each class the save data is for, you could put the classes in a class map to be accessed when the game is loaded. This would allow for future class additions or subtractions without needing to write out the code to check for each individual class.		2		
3		For line 89, you might consider moving the floor generation logic out of the "def __init__" in case you wanted to modify how the rooms were generated in the future. This could be for things like adding extra functionality to the rooms depending on things happening in the game or maybe changing how the rooms are generated in different difficulties beyond just making one of the random floors the boss room		3		

	Python's Wild Ride	Playability: 6/10 Readability: 7/10 Quality: 7/10 Understanding of Python: 7/10 Overall: 27/40	
	1		On line 134, make sure to use "and" instead of "&" to allow python to properly handle the logic
	2		
	3		