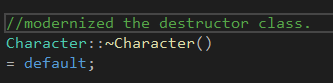
1. Namespaces missing or variable types not correctly declared (c2061)
   1. Error and fix shown below:



1. The Character class was missing some properties, so I added them manually.
   1. Name was missing.
   2. Duplicated gridbox property fixed, removed one of them and fixed the missing Types class reference.
   3. Removed duplicated reference to target Property.
   4. Cleaned unused #includes
2. Inconsistent code formatting, there was no standardization on case type, sometimes camelCase was used, other times Pascal case.
   1. Referred to this [documentation](https://lefticus.gitbooks.io/cpp-best-practices/content/03-Style.html) to make the code refactor to camelCase and snake\_case.
   2. There were also spelling mistakes:Texto

      Descrição gerada automaticamente com confiança média
   3. I have gone on every class fixing spelling mistakes and code conventions.
   4. BattfleField class was misspelled, the right name should be Battlefield.
3. Removed namespace refs from header files, it is a good practice to avoid unwanted conflicts.
4. Removed the unnecessary properties of the battlefield class, there is no need to have a pointer to each player when we have a list of all players that we could iterate as needed.
5. The character function TakeDamage was receiving a variable that was not being used, I fixed the issue by using that variable as the value of damage the player will be inflicted.
6. Repeated includes on the character class, I removed the repeated ones as there was no need for 2 includes of the same class.
7. Where applicable I modernized the code, on this case I modernized the destructor of the Character class. 
8. Each class should have responsibilities only to itself, so there is no sense in the Battlefield class having a return random integer to other classes, this is a principle of clean code.
   1. That is also valid for the Create Character and create enemy character, also both o them will be refactored in a create player function se we can re-use the same function every time.
   2. Created a turn handler so we can segregate responsibilities from each class and organize better how the game will flow.
   3. Removed Drawn Battlefield from Grid class, as that should be a function of battlefield.
   4. Removed turn handling from battlefield, so we can give that responsibility to a class that is designed to handle turns.
9. Removed the Shared Pointer from player class, it is an unnecessary use of this type of pointer, we don’t need to know how many references we have to these pointers, and we don’t need to use memory for a control block.
10. Refactored how the player management work, so we can have multiple players in the future.
11. There were many functions that could be simplified and deleted, the allocate enemy character and allocate player character on battlefield were deleted and I created a AlocatePlayer, this made the code cleaner, and we didn’t repeat code unnecessarily.
12. In the character class, the switch used to define what was going to be the char class was compering strings, that is not optimal because the switch case needs a number to work properly.
13. Updated some recursive functions to improve performance, recursive functions usually create a stack that while repetition loops don’t.
14. One of the things that were not being used is pass by reference, this way we can greatly reduce the memory cost and it is easier to read and use than pointers