

# writeup

☰ Desc	
▼ Tags	Complete

## Project Summaries

The purpose of the project in my opinion is demonstrating good OOP practices such as inheriting, encapsulating, and using polymorphism. This has been the best project I have done so far in TDQC. I just love the elegancy of being able to create classes that define the children classes. I prefer this paradigm a whole lot more than procedural.

## Challenges

My biggest challenge as always is time. I have an infant daughter who needs our constant attention. I usually don't sleep the whole 30+ hours in order to make it and I always cut it close. I just lose so much time when taking care of my daughter.

## Success

My biggest success in my opinion is my use of abstract classes to easily create unique potions without having to rewrite a ton of code. Another success if having both my hero object and monster object both inherit from the same base abstract class. This allowed me to use functions as "generics". The function didn't care who was a hero or who was a monster because they both had a guaranteed contract to support the methods required for sed function.

## Lessons Learned

My biggest lesson, and my biggest waste of time on this project was creating a list of all my consumable potions and only saving the reference to the class and not instanciating them. This caused a ton of weird and hard to narrow bugs in my code. I probably spent a good two hours on this bug.

What I needed was this:

```
CONSUMABLE_LOOT = (  
    AttackPotion(),  
    LuckySeven(),  
    HealthPotion(),  
    HeavyHand(),  
    PierceShot()  
)
```

What I had was this

```
CONSUMABLE_LOOT = (  
    AttackPotion,  
    LuckySeven,  
    HealthPotion,  
    HeavyHand,  
    PierceShot  
)
```