# Game reset function:

When the game is over, for ease of access there must be a button to completely reset the entire game to the beginning. This is so the player can restart the game without manually having to close processing and restart the entire code or reloading the webpage.

Reasoning:

* I discussed with Kit before about the two flappy bird games, and that the one that succeeded was the one that had an easy way of restarting the game.
  + Players will always find ways to be lazy in games, and by providing an easy and effective way of restarting players will want to play the game more.

How to solve this issue

* Make every array instead of setting index like array [] = 1 or array [] = 2
  + Use array [] = array [] + 1; / array[] +=1;
    - This way when I reset each array to the value of 0 the game will still follow the order which it is meant to be.
* Create a function called gameReset() and add all the arrays original index and original variable values to the function
  + Call this function in the setup() to cheat initialize the game
    - Call the function when the game is over, and you click retry
      * GameReset();
    - Call the function every time the game reset function is needed (ie. Retry in game over, retry in esc menu)
  + Arrays include
    - Master[] = 0
    - Bosshp = full;
    - Text1[] = 0
    - Etcetera...

# Turn based Battle:

During the single boss fight of the game, I want to emulate the same feeling you get when you play the game like an rpg. This would include actual turn-based combat, which would feel like what a jrpg is. Before my previous assignments had combat, but they lacked that turn based feeling players wish to get from jrpgs.

Reasoning:

* Although I have created a combat system in the last assignment, it does not follow turn based order and I wish to create that turn based feeling in this assignment
* My last assignment had players feeling like it was just a clicker game instead of an actual fight, which was largely attributed to the fact that the game did not have actual turn-based combat.

Goals and how to achieve

* Have the player be the odd number index, have the boss be an even number index
  + Set conditionals with if statements which use % as that can get the remainder. If array index %2 = 0 then its boss turn if array index% != 2 then it is a odd number which means it is the players turn
* Have a set of moves the boss will use
  + Make a move set of 4 for the boss and assign each to a index number between 1 and 4
  + Shift the array each time a move is used and create conditionals for the more damaging moves
    - Big damage move will only be used when the boss in under a 4th hp
      * If (boss(hp < bosshp/4))
        + Bigdmgmove();
* Have a fight button the player can press and do a different amount of dmg for each as well.

# Interactable NPCs and text boxes

You can’t have a jrpg without npcs that give dialogue and pop-up text boxes on the screen. This is the one defining feature of JRPGS and rpgs that cannot and will not change in every single iteration. Without these it is hard to showcase the story, and it then lacks the role-playing aspect which players love.

Reasoning:

* I have experimented with different ways to prompt the player on how to play the game, however all of them fall short as nothing beats just telling them
* Text boxes add story and atmosphere to the game which players enjoy
* Dialogue is one of the biggest reasons as to why players feel immersion when playing games like RPGS

Goals and how to achieve:

* Have the NPCS have an interaction window when clicking on them to initiate talking
  + Use an ismousover() function and use mousepressed() for the functionality
    - It is the same logic as buttons but masked underneath NPCs
* Set each type of dialogue under an index of an NPC [] array.
  + Each index value will show a different text box (since I cannot use more than 2 text functions I will just use multiple PImages of text boxes which shift based on array index)
  + By clicking on the screen the index of the array will change and progress the text
    - Have a conditional to make sure the text array does not happen when the text function is not active with the npc.
    - Use a isInDialogue Boolean to ensure there are no issues.
      * In the initialization the Boolean is set to false.