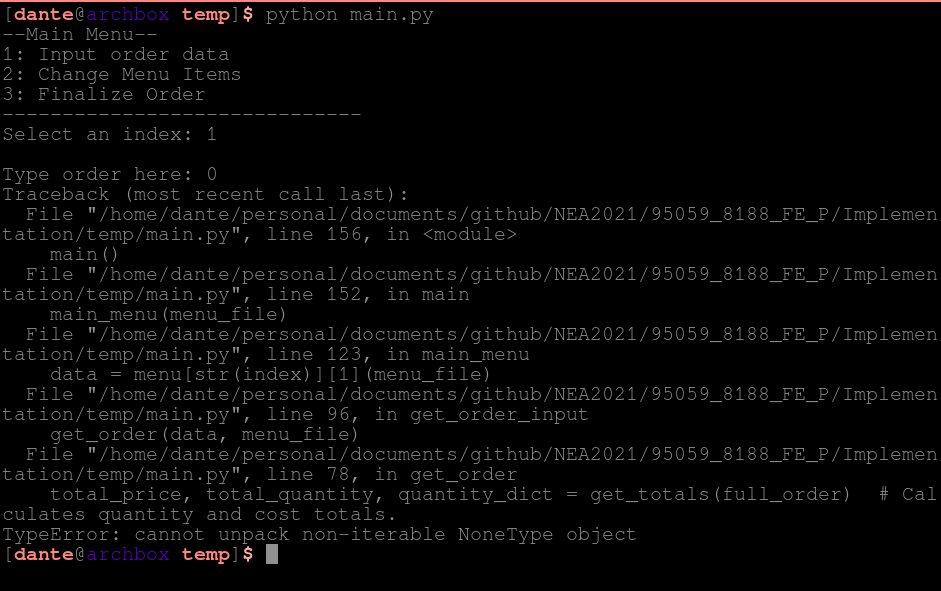
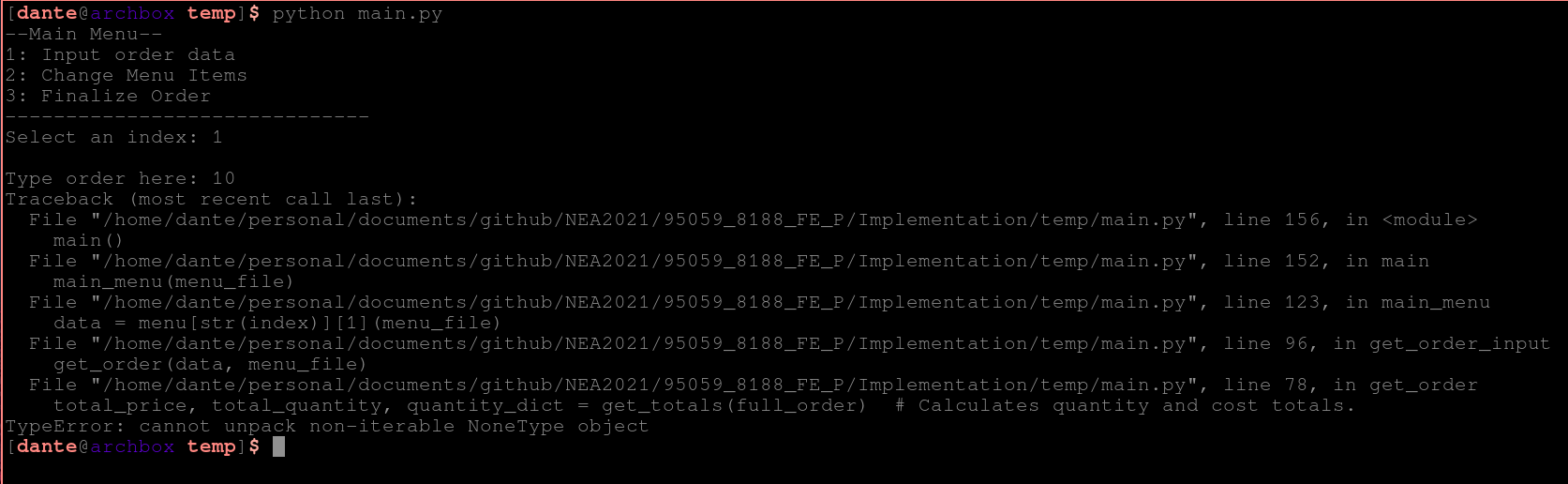
**­Debugging**

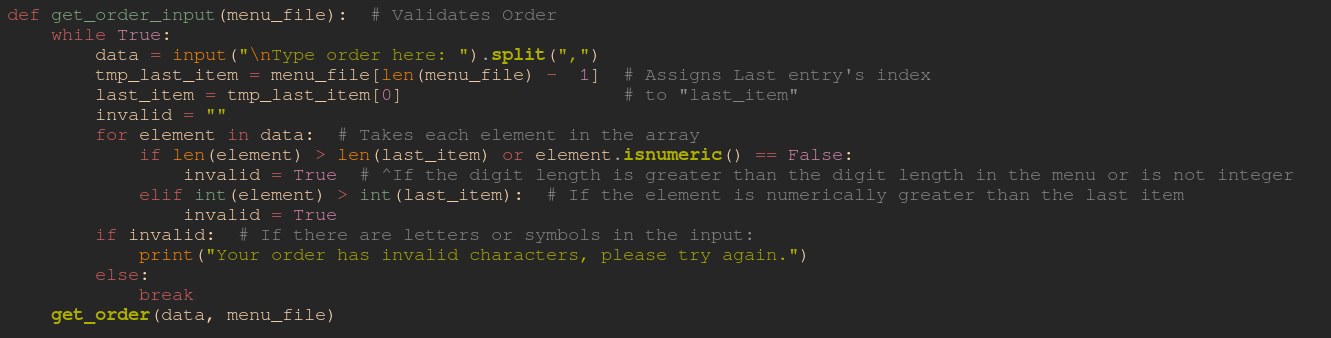
* **Bug #1: Fixing Order Input Validation in get\_order\_input() *(Runtime Error)***
  + When entering a single value for the table number that is equal to or less than 10 the program would raise **TypeError**
    - Examples: “0” would raise **TypeError**, as well as entering “10”

**Error Messages:**

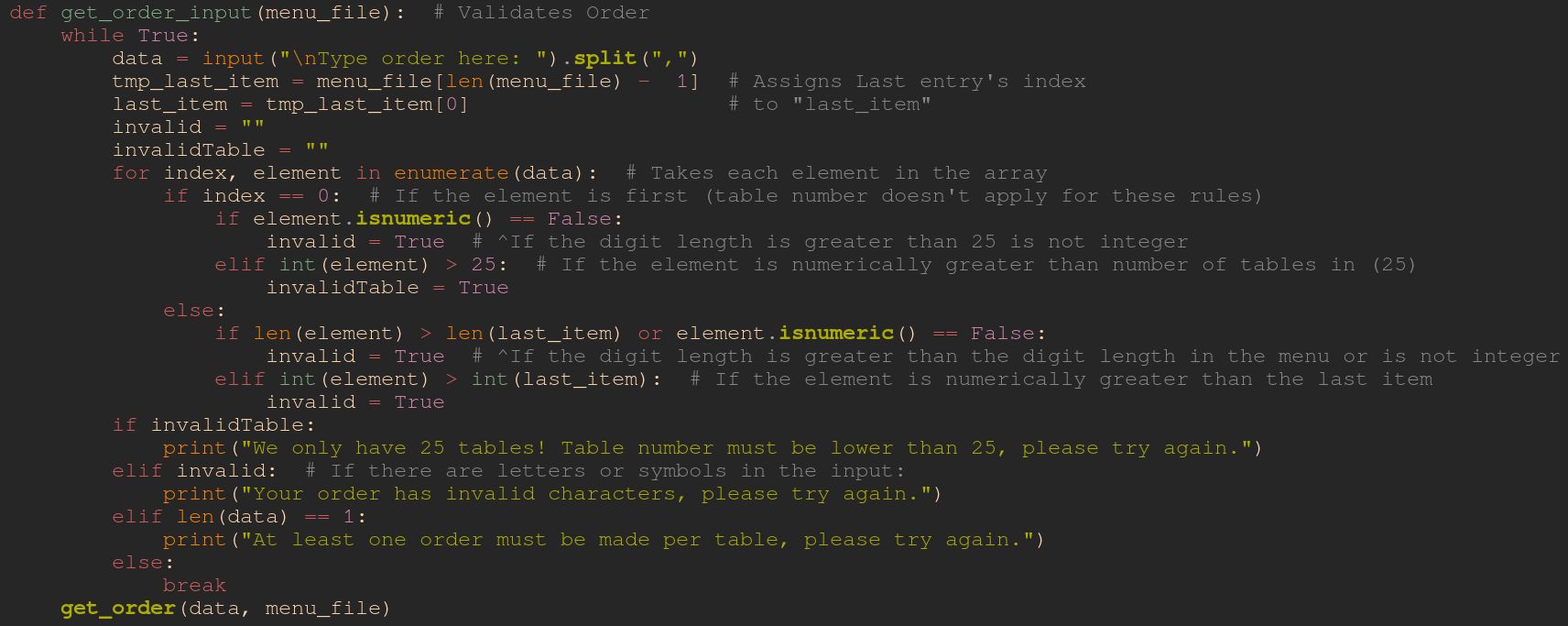




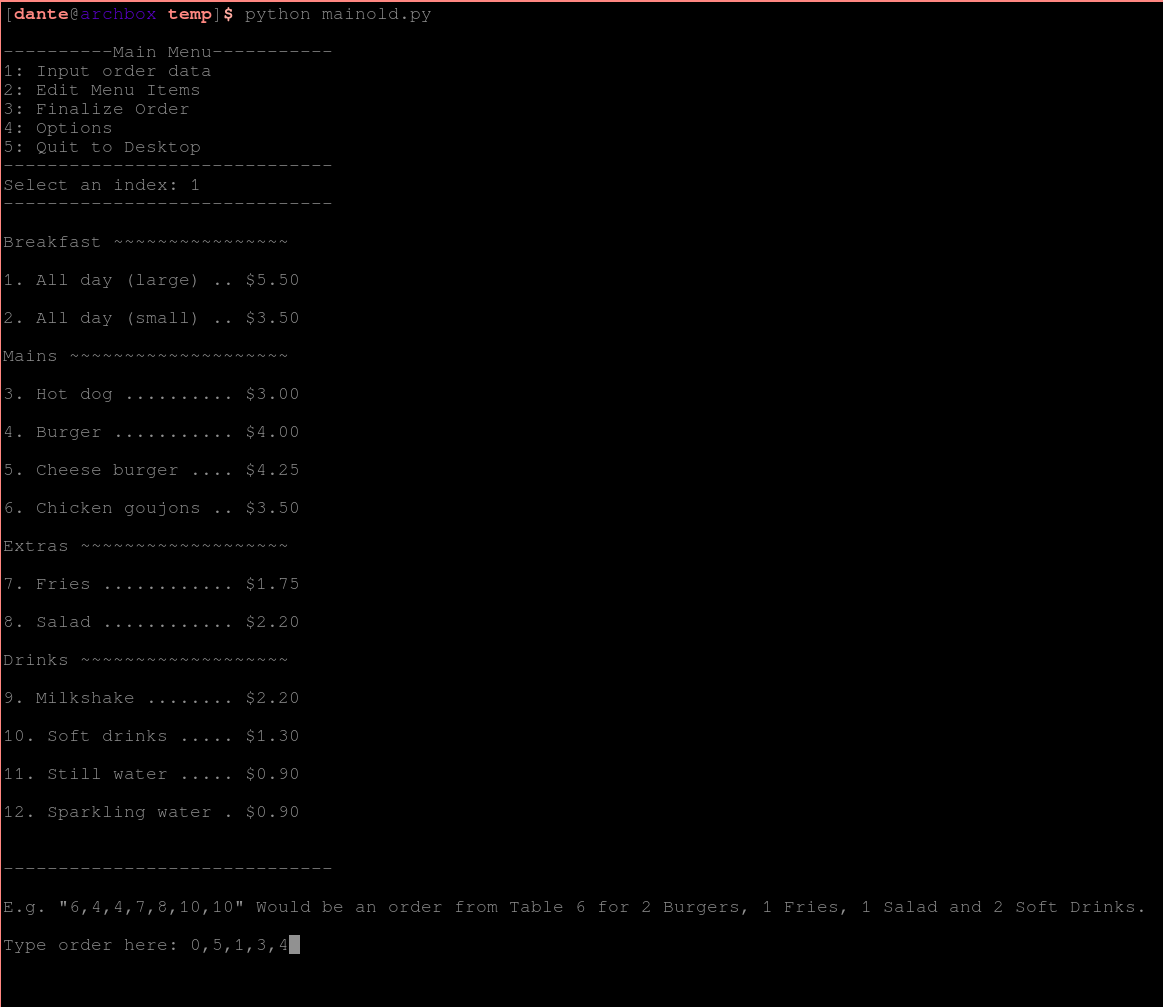
**Original Code:**



**Fixed Code:**



* + How was the bug fixed?
    - The code will loop through each element in the user’s input
    - The first element (the table number) will be checked to check to see:
      * IF: it is a numeric string
      * IF: The element is less than 25 *(at the time I was not aware that the maximum amount of tables was 10 and this was fixed later in the production of the program)*
* **Bug #2: Fixing Table number bug when entering “0” as a table number *(Logic Error)***
  + When entering the table number for the user’s order as “0”, the program would still accept it.
    - Example of entering“0” as the table number:



* + - The ouput shows order with the table number: “0”
    - The program has accepted the table number when it should’ve rejected it instead because Table #0 doesn’t exist  
        
        
        
        
      

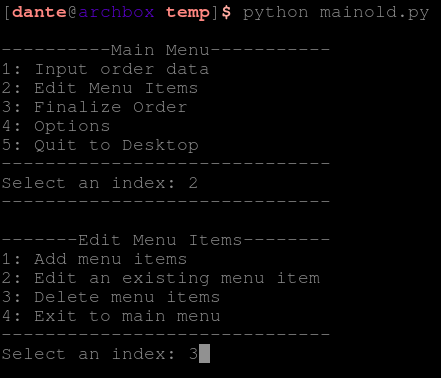
**Fixed Code:**

* + How was the bug fixed?
    - The code will loop through each element in the user’s input
    - The first element (the table number) will be checked to check to see:
      * IF it’s greater than 0 and less than or equal to 10. (Tables)
    - Originally the code would just check if it was less than 10 which would allow “0” to go through
    - **Key:**
      * ***Green*** *= Additions made in the program*
      * ***Red*** *= Deletions in the program*

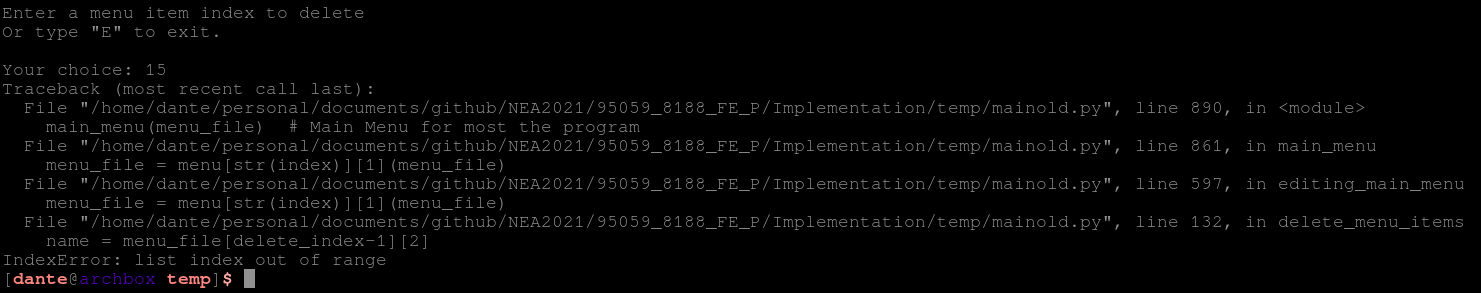


* **Bug #3: Fixing Deletion of Menu Items Validation in delete\_menu\_items() *(Runtime Error)***
  + Upon entering a menu item index number that was greater than the maximum index number (out of range), the program would crash and raise **IndexError**
    - Examples: “” would raise **IndexError**, as well as entering “15”

**Program Running with Error Message:**



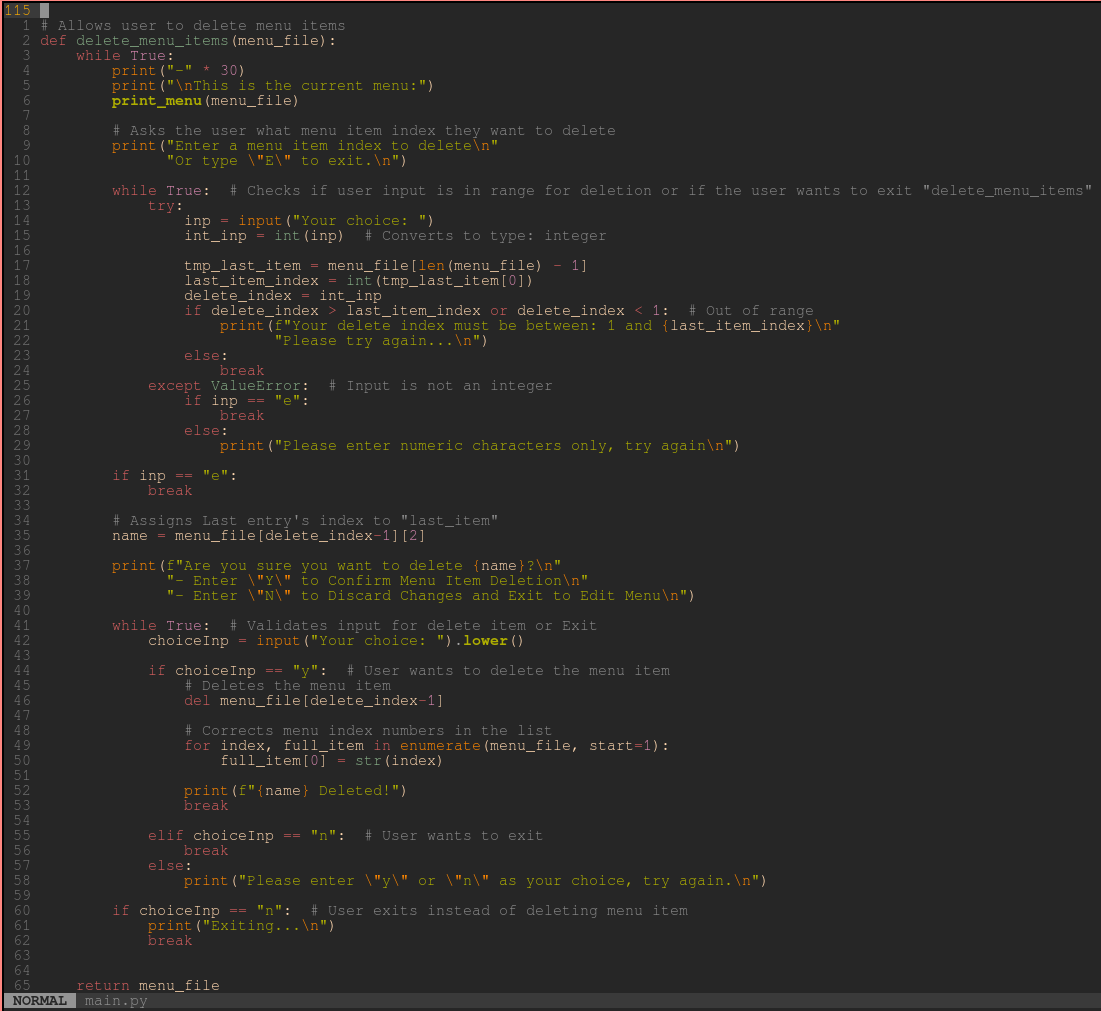




**Original Code:**

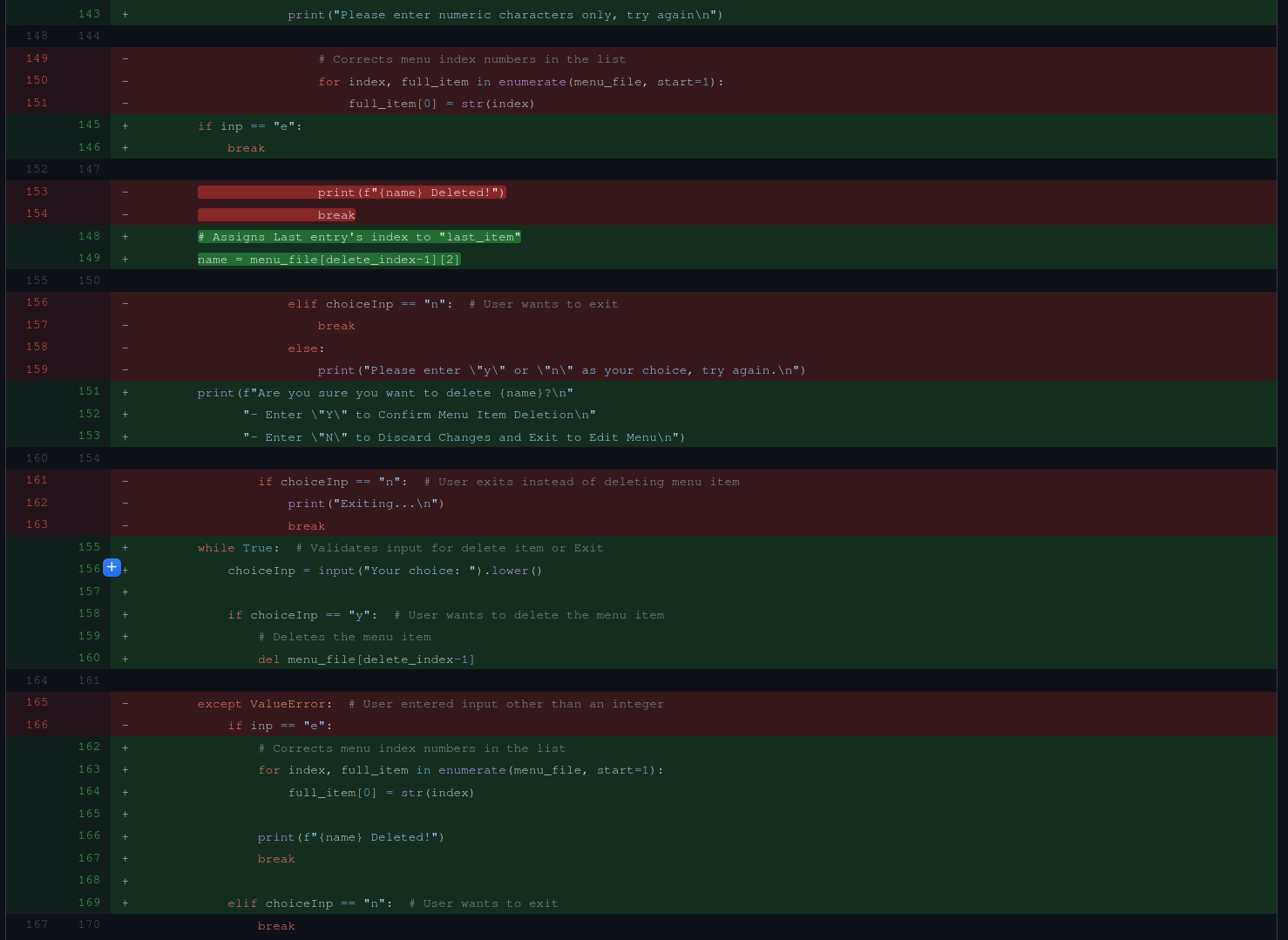


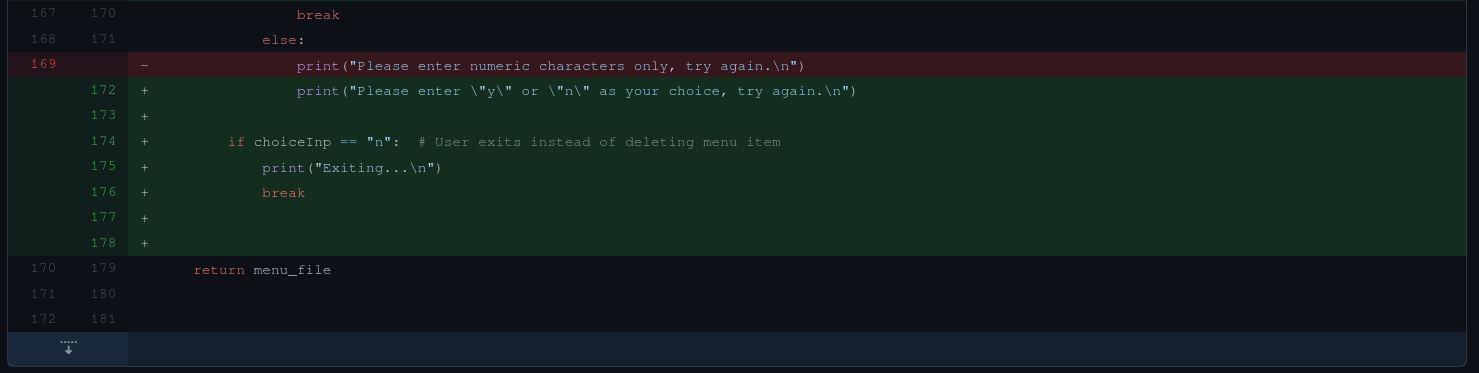
**Fixed Code:**

****

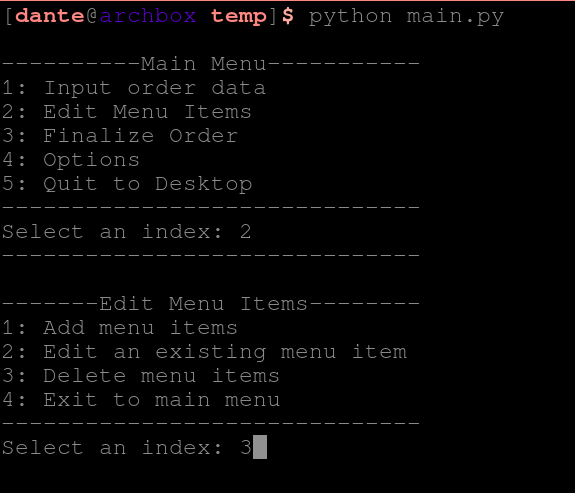
* + How was the bug fixed?
    - The program will initialize the variable *last\_item\_index* (the menu item in the menu with the greatest index number)
    - Then the program will evaluate if the user’s input is greater than value of *last\_item\_index.*
      * IF the delete\_index (deletion index that the user entered) is greater than the *last\_item\_index* less than 1 THEN: ***Ask to enter another value again.***
    - *Originally the code would raise* ***IndexError*** *upon the user entering an invalid index because the user specified index doesn’t exist.*
    - **Key:**
      * ***Green*** *= Additions made in the program*
      * ***Red*** *= Deletions in the program*







***Fixed Code Running without Errors:***







**The program doesn’t raise any errors and can only take user input that is valid.**