Implement a function that takes as input three variables, and returns the largest of the three. Do this without using the Python max() function!

The goal of this exercise is to think about some internals that Python normally takes care of for us. All you need is some variables and if statements!

This code does not give us correct answer all time

```
def max_of_three(a,b,c):
     max_3=0
     if a>b:
         max_3=a
         if a>c:
             max_3=c
         else:
             max_3=a
     else:
          if b>c:
             max_3=b
          else:
             max_3=c
     return max_3
```

Now call the function

```
a = int(input("Enter a Number"))
b = int(input("Enter a Number"))
c = int(input("Enter a Number"))

solution = max_of_three(a,b,c)
```

```
Enter a Number12
Enter a Number1
Enter a Number109
```

Now print the solution

```
print(solution)
```

```
12
```

But we can do this using max() function which will give us correct answer all cases

```
a = int(input("Enter a Number"))
b = int(input("Enter a Number"))
c = int(input("Enter a Number"))
```

```
solution2 = max(a,b,c)
print(solution2)
```

```
Enter a Number12
Enter a Number1
Enter a Number109
109
```

Now if we want to take the input as array

```
x = [int(i) for i in input().split()]
print (x)
```

```
12 1 22 109 77 23 10 4 7 64 3
[12, 1, 22, 109, 77, 23, 10, 4, 7, 64, 3]
```

```
solution3 = max(x)
print(solution3)
```

```
109
```

Take an array and find if there is a duplicate number!!!

```
y = [int(i) for i in input().split()]
print(y)
```

```
12 1 22 109 23 33 21 67 56 98 45 67 23 29 77 84
[12, 1, 22, 109, 23, 33, 21, 67, 56, 98, 45, 67, 23, 29, 77, 84]
```

Now rest of the code..!!!

```
def checkIfDuplicates_3(y):
    ''' Check if given list contains any duplicates '''
    for elem in y:
        if y.count(elem) > 1:
            return True
    return False


result = checkIfDuplicates_3(y)
if result:
    print('Yes, list contains duplicates')
else:
    print('No duplicates found in list')
```

```
Yes, list contains duplicates
```

Now, we want to find out how many times they occur.

```
from collections import Counter
```

```
Counter(y)
```

```
Counter({1: 1,
         12: 1,
         21: 1,
         22: 1,
         23: 2,
         29: 1,
         33: 1,
         45: 1,
         56: 1,
         67: 2,
         77: 1,
         84: 1,
         98: 1,
         109: 1})
```

Now find out the sum of all numbers

```
SUM = sum(y)
print(SUM)
```

```
767
```

Now if there is strings then lets see what happens..

```
y = [i for i in input().split()]
from collections import Counter
Counter(y)
```

```
2 3 12 14 5  6 67 56 99 87 65 56 43 221 12 102 13 22 45 67 44 3 98 5 21 12 56 77
Counter({'102': 1,
         '12': 3,
         '13': 1,
         '14': 1,
         '2': 1,
         '21': 1,
         '22': 1,
         '221': 1,
         '3': 2,
         '43': 1,
         '44': 1,
         '45': 1,
         '5': 2,
         '56': 3,
         '6': 1,
         '65': 1,
         '67': 2,
         '77': 1,
         '87': 1,
         '98': 1,
         '99': 1})
```

Now take an strings as input and count the alphabate in that sentence

```
z = [i for i in input().split()]
from collections import Counter
Counter(z)
```

```
'sunny' 'Badhon' 23 44 56 12 12.3 12.3
Counter({"'Badhon'": 1,
         "'sunny'": 1,
         '12': 1,
         '12.3': 2,
         '23': 1,
         '44': 1,
         '56': 1})
```

Insert value between values in the array

```
z1 = [int(i) for i in input().split()]
print(z1)
```

```
6 8 3 23 10 3 4 44 8 78 67 45 99 102 304  44 54 10 22 33
[6, 8, 3, 23, 10, 3, 4, 44, 8, 78, 67, 45, 99, 102, 304, 44, 54, 10, 22, 33]
```

Now lets insert value 80 at position 3

```
z1.insert(3, 80)
print(z1)
```

```
[6, 8, 3, 80, 23, 10, 3, 4, 44, 8, 78, 67, 45, 99, 102, 304, 44, 54, 10, 22, 33]
```

Now reverse the value in the array

```
z1.reverse()
print(z1)
```

```
[33, 22, 10, 54, 44, 304, 102, 99, 45, 67, 78, 8, 44, 4, 3, 10, 23, 80, 3, 8, 6]
```

Find the first non-consecutive number in an array

```
def first_non_consecutive(arr):
    prev = arr[0]
    first = None
    for num in arr[1:]:
        if (prev + 1) != num:
            first = num
            break
        prev += 1
    return first


arr = [1,2,3,4,6,7,8,9]
Result = first_non_consecutive(arr)
```

```
print(Result)
```

```
6
```

Lets work with 2D array

```
T = [[11, 12, 5, 2], [15, 6,10], [10, 8, 12, 5], [12,15,8,6]]
print(T)
print(T[1])
print(T[1][2])
```

```
[[11, 12, 5, 2], [15, 6, 10], [10, 8, 12, 5], [12, 15, 8, 6]]
[15, 6, 10]
10
```

To print out the entire two dimensional array we can use python for loop as shown below. We use end of line to print out the values in different rows.

```
for r in T:
    for c in r:
        print(c,end = " ")
    print()
```

```
11 12 5 2
15 6 10
10 8 12 5
12 15 8 6
```