Assignment 2

April 25, 2019

0.0.1 Largest number in a list

Write Python Program to find the largest number in a list

```
In [1]: numbers = []
        n = int(input("Enter the number of elements: "))
        print("Enter the elements ")
        for i in range(1, n + 1):
            ele = int(input())
            numbers.append(ele)
        l = max(numbers)
        print("The largest number is ", l)
```

```
Enter the number of elements: 4
Enter the elements
5
2
9
6
The largest number is  9
```

0.0.2 Horner’s Method

Python Program to Compute a Polynomial Equation given that the Coefficients of the Polynomial are stored in a List.

```
In [2]: def horner(poly, n, x):
            result = poly[0]
            for i in range(1, n):
                result = result * x + poly[i]
            print("The value of polynomial at x = " + str(x) + " is " + str(result))


        a0 = int(input("Enter a value for a0:"))
        a1 = int(input("Enter a value for a1:"))
        a2 = int(input("Enter a value for a2:"))
        a3 = int(input("Enter a value for a3:"))
        a4 = int(input("Enter a value for a4:"))

        poly = [a0, a1, a2, a3, a4]
        x = int(input("Enter the value for x:"))
        n = len(poly)
        horner(poly, n, x)
```

```
Enter a value for a0:5
Enter a value for a1:4
Enter a value for a2:3
Enter a value for a3:2
Enter a value for a4:1
Enter the value for x:6
The value of polynomial at x = 6 is 7465
```

0.0.3 Word frequiency using a dictionary

Python Program to Count the Frequency of Words Appearing in a String Using a Dictionary.

```
In [3]: s = input("Enter string: ")
        freq = {}
        for i in s:
            if i in freq:
                freq[i] += 1
            else:
                freq[i] = 1
        print("Frequency of all characters in \"" + s + "\" is :\n " + str(freq))
```

```
Enter string: We're in the end game now
Frequency of all characters in "We're in the end game now" is :
 {'W': 1, 'e': 5, "'": 1, 'r': 1, ' ': 5, 'i': 1, 'n': 3, 't': 1, 'h': 1, 'd': 1, 'g': 1, 'a':
```

0.0.4 Sum of elements in a list recursively

Python Program to Find the Sum of Elements in a List Recursively

```
In [4]: def sum_of_list(numbers, n):
            if n == 0:
                return 0
            else:
                return numbers[n - 1] + sum_of_list(numbers, n - 1)


        n = int(input("Enter the number of elements for list:"))
        numbers = []
        print("Enter elements :")
        for i in range(0, n):
            ele = int(input())
            numbers.append(ele)
        print("The list is:")
        print(numbers)
        s = sum_of_list(numbers, n)
        print("Sum of items in list : ", s)
```

```
Enter the number of elements for list:4
Enter elements :
4
2
6
9
The list is:
[4, 2, 6, 9]
Sum of items in list :  21
```

0.0.5 Intersection of Two Lists

Python Program to Find the Intersection of Two Lists

```
In [5]: def intersection(list1, list2):
            list3 = [value for value in list1 if value in list2]
            return list3


        list1 = [4, 9, 1, 17, 11, 26, 28, 54, 69]
        list2 = [9, 9, 74, 21, 45, 11, 63, 28, 26]
        print("Intersection of two list is ")
        print(intersection(list1, list2))
```

```
Intersection of two list is
[9, 11, 26, 28]
```

0.0.6 Seperate odd and even

Write Python Program to put the even and odd elements in list into two different lists

```
In [6]: n = int(input("Enter the number of elements for list:"))
        list1 = []
        print("Enter elements :")
        for i in range(0, n):
            ele = int(input())
            list1.append(ele)
        even_list = [value for value in list1 if value % 2 == 0]
        odd_list = [value for value in list1 if value % 2 == 1]
        print("Even list : ", even_list)
        print("Odd list : ", odd_list)
```

```
Enter the number of elements for list:5
Enter elements :
1
6
8
2
8
Even list :  [6, 8, 2, 8]
Odd list :  [1]
```

0.0.7 Write Python Program to merge two lists and sort it.

```
In [7]: list1 = []
        list2 = []
        list3 = []
        n = int(input("Enter the number of elements for list 1:"))
        print("Enter elements for list 1:")
        for i in range(0, n):
            ele = int(input())
            list1.append(ele)
        n = int(input("Enter the number of elements for list 2:"))
        print("Enter elements for list 2:")
        for i in range(0, n):
            ele = int(input())
            list2.append(ele)
        list3 = list1 + list2
        print("The merged list is")
        print(list3)
        list3.sort()
        print("The sorted list is")
        print(list3)
```

```
Enter the number of elements for list 1:4
Enter elements for list 1:
4
2
1
5
Enter the number of elements for list 2:3
Enter elements for list 2:
1



        ---------------------------------------------------------------------------
```

```
    ValueError                                Traceback (most recent call last)

    <ipython-input-7-43ccb7f128a3> in <module>
     10 print("Enter elements for list 2:")
     11 for i in range(0, n):
---> 12     ele = int(input())
     13     list2.append(ele)
     14 list3 = list1 + list2


    ValueError: invalid literal for int() with base 10: ''
```

0.0.8 Python Program to swap the first and last value of a list

```
In [ ]: list1 = []
        n = int(input("Enter the number of elements for list:"))
        print("Enter elements for list:")
        for i in range(0, n):
            ele = int(input())
            list1.append(ele)
        a = list1[0]
        list1[0] = list1[n - 1]
        list1[n - 1] = a
        print("After swapping the new list is")
        print(list1)
```

0.0.9 Python Program to remove the nth index character from a non-empty string.

```
In [ ]: string = input("Enter a string : ")
        n = int(input("Enter the index of the character to remove :"))
        string = string[:n] + string[n + 1:]
        print("Modified string : ", string)
```

0.0.10 Python Program to remove the characters of odd index values in a string.

```
In [ ]: string = input("Enter a string : ")
        mod_string = ""
        for i in range(len(string)):
            if i % 2 == 0:
                mod_string = mod_string + string[i]
        print("Modified string is : ", mod_string)
```

0.0.11 Write a Python program to multiplies all the items in a list.

```
In [ ]: list1 = []
        product = 1
        n = int(input("Enter the number of elements for list:"))
        print("Enter elements for list:")
        for i in range(0, n):
            ele = int(input())
            list1.append(ele)
        for i in range(0, n):
            product = product * list1[i]
        print("Product : ", product)
```

0.0.12 Write a Python program to remove duplicates from a list.

```
In [ ]: list1 = []
        n = int(input("Enter the number of elements for list:"))
        print("Enter elements for list:")
        for i in range(0, n):
            ele = int(input())
            list1.append(ele)
        list2 = []
        for i in range(0, n):
            if list1[i] in list1 and list1[i] not in list2:
                list2.append(list1[i])
        print("After removing duplicates : ", list2)
```

0.0.13 Write a Python program to check a list is empty or not.

```
In [ ]: list1 = []
        if len(list1) == 0:
            print("The list is empty")
        else:
            print("The list is not empty")
```

0.0.14 Write a Python function that takes two lists and returns True if they have atleast one common member.

```
In [ ]: list1 = []
        list2 = []
        n = int(input("Enter the number of elements for list 1:"))
        print("Enter elements for list 1:")
        for i in range(0, n):
            ele = int(input())
            list1.append(ele)
        n = int(input("Enter the number of elements for list 2:"))
        print("Enter elements for list 2:")
        for i in range(0, n):
            ele = int(input())
            list2.append(ele)
        for i in range(0, n):
            if list1[i] in list2:
                print("TRUE")
                break
            else:
                print("FALSE")
```

0.0.15 Write a Python program to count the number of elements in a list within a specified range.

```
In [ ]: list1 = []
        n = int(input("Enter the number of elements for list 1:"))
        print("Enter elements for list 1:")
        for i in range(0, n):
            ele = int(input())
            list1.append(ele)
        print("enter the range:")
        x = int(input())
        y = int(input())
        c = 0
        for i in range(0, n):
            if list1[i] >= x and list1[i] <= y:
                c = c + 1
        print("number of elements within ", x, "and", y, "is", c)
```

0.0.16 Write a Python program to insert an element before each element of a list.

```
In [ ]: list1 = []
        n = int(input("Enter the number of elements for list:"))
        print("Enter elements for list 1:")
        for i in range(0, n):
            ele = int(input())
            list1.append(ele)
        list1 = [v for ele in list1 for v in ('a', ele)]
        print(list1)
```

0.0.17 Write a Python program to find all the values in a list are greater than a specified number.

```
In [ ]: list1 = []
        n = int(input("Enter the number of elements for list:"))
        print("Enter elements for list 1:")
        for i in range(0, n):
            ele = int(input())
            list1.append(ele)
        x = int(input("enter a number : "))
        for num in list1:
            if num > x:
                print(num)
```

0.0.18 Write a Python script to merge two Python dictionaries.

```
In [ ]: dict1 = {'a': 3, 'b': 4, 'c': 6}
        dict2 = {'d': 7, 'e': 9}
        print("before merging ")
        print(dict1)
        print(dict2)
        dict1.update(dict2)
        print("after merging ")
        print(dict1)
```

0.0.19 Write a Python program to sort a dictionary by key.

```
In [ ]: dict1 = {'c': 4, 'e': 1, 'a': 6, 'd': 5, 'b': 2}
        print("before sorting:")
        print(dict1)
        print("after sorting:")
        for i in sorted(dict1):
            print("%s: %s" % (i, dict1[i]))
```

0.0.20 Write a Python program to get the maximum and minimum value in a dictionary.

```
In [ ]: my_dict = {'x': 500, 'y': 5874, 'z': 560}

        key_max = max(my_dict.keys(), key=(lambda k: my_dict[k]))
        key_min = min(my_dict.keys(), key=(lambda k: my_dict[k]))

        print('Maximum Value: ', my_dict[key_max])
        print('Minimum Value: ', my_dict[key_min])
```

0.0.21 Write a Python program to sort a list alphabetically in a dictionary.

```
In [ ]: chars = {'n1': ['c', 'a', 'j'], 'n2': ['r', 'd', 'q'], 'n3': ['z', 'k', 'b']}
        sorted_dict = {x: sorted(y) for x, y in chars.items()}
        print(sorted_dict)
```

0.0.22 Check if list is ascending order or not

```
In [ ]: test_list = [9, 4, 5, 8, 10]
        print("Original list : " + str(test_list))


        def is_ascending(l):
            if (all(l[i] <= l[i + 1] for i in range(len(l) - 1))):
                print("Yes, List is sorted.")
            else:
                print("No, List is not sorted.")
        is_ascending(test_list)
```

0.0.23 Remove duplicates from a list in Python

```
In [ ]: mylist = [10, 20, 20, 30, 40, 50, 50]
        mylist = list(dict.fromkeys(mylist))
        print(mylist)
```

0.0.24 Check if list is empty or not

```
In [ ]: my_list = []
        if not my_list:
            print("Empty")
```

0.0.25 Count frequency of each character in input string, using dictionary

```
In [ ]: s = "Hello"
        res = {}
        for keys in s:
            res[keys] = res.get(keys, 0) + 1

        print(res)
```

0.0.26 Extract email IDs using re

```
In [ ]: import re

        s = 'Hello from shubhamg199630@gmail.com to priya@yahoo.com about the meeting @2PM'
        lst = re.findall('\S+@\S+\.\S+', s)
        print(lst)
```

0.0.27 Validate Indian phone number

```
In [ ]: import re
        ph_num = input("Enter a phone number:")
        if re.search('\+91\s[0-9]{5}\s[0-9]{5}', ph_num):
            print("Valid")
```

0.0.28 'a' followed by anything, ending in a 'b'

```
In [ ]: import re
        s = input("Enter string:")
        if re.search('a.*?b$', s):
            print("Match")
```

0.0.29 Validate USN

```
In [ ]: import re
        usn = input("Enter USN:")
        if (re.search('[0-9][A-Z]{2}1[0-9][A-Z]{2}[0-9]{3}', usn)):
            print("UG USN")
        elif (re.search('[0-9][A-Z]{2}1[0-9][A-Z]{3}[0-9]{2}', usn)):
            print("PG USN")
```

0.0.30 Rectangle class and area

```
In [ ]: class Rectangle():
            def __init__(self, l, w):
                self.length = l
                self.width = w

            def area(self):
                return self.length * self.width


        new_rectangle = Rectangle(12, 10)
        print(new_rectangle.area())
```

0.0.31 Modified rectangle class and area

Construct a program to create a class called rectangle with the help of a cornerpoint, width and height. Write following methods and demonstrate their working: a.To find and display center of rectangle b.To display point as an ordered pair c.To resize the rectangle

```
In [8]: class Rectangle(object):
            def __init__(self, l=0, w=0, cp=(0, 0)):
                self.length = l
                self.width = w
                self.bottom_left_corner = cp
                self.bottom_right_corner = (cp[0] + l, cp[1])
                self.top_left_corner = (cp[0], cp[1] + w)
                self.top_right_corner = (cp[0] + l, cp[1] + w)
                self.center = (cp[0] + (w / 2), cp[1] + (l / 2))

            def display_center(self):
                print(f"Center is: {self.center}")

            def __str__(self):
                return f"Length: {self.length}, Width: {self.width}, Bottom Left: {self.bottom_

            def resize(self):
                print(
                    "With the same left corner point, pick a width and height to modify:"
                )
                self.length = int(input("Enter the new length:"))
                self.width = int(input("Enter the new width:"))
                self.__init__(l=self.length, w=self.width, cp=self.bottom_left_corner)
                print(self)


        r = Rectangle(5, 6, (0, 0))
        print(r)
        r.display_center()
        r.resize()
```

```
Length: 5, Width: 6, Bottom Left: (0, 0), Bottom Right: (5, 0), Top Left: (0, 6), Top Right: (5
Center is: (3.0, 2.5)
With the same left corner point, pick a width and height to modify:
Enter the new length:8
Enter the new width:9
Length: 8, Width: 9, Bottom Left: (0, 0), Bottom Right: (8, 0), Top Left: (0, 9), Top Right: (8
```

0.0.32 Circle class and area

```
In [ ]: class Circle():
            def __init__(self, r):
                self.radius = r

            def area(self):
                return self.radius**2 * 3.14

            def perimeter(self):
                return 2 * self.radius * 3.14


        new_circle = Circle(8)
        print(new_circle.area())
        print(new_circle.perimeter())
```

0.0.33 Python class accepts string and prints it

```
In [ ]: class IOString():
            def __init__(self):
                self.str1 = ""

            def get_String(self):
                self.str1 = input()

            def print_String(self):
                print("Printing:" + self.str1)
```

```
        str1 = IOString()
        str1.get_String()
        str1.print_String()
```

0.0.34 Basic Calculator class

```
In [ ]: class Cal():
            def __init__(self, a, b):
                self.a = a
                self.b = b

            def add(self):
                return self.a + self.b

            def mul(self):
                return self.a * self.b

            def div(self):
                return self.a / self.b

            def sub(self):
                return self.a - self.b


        a = int(input("Enter first number: "))
        b = int(input("Enter second number: "))
        obj = Cal(a, b)
        choice = 1
        while choice != 0:
            print("0. Exit")
            print("1. Add")
            print("2. Subtraction")
            print("3. Multiplication")
            print("4. Division")
            choice = int(input("Enter choice: "))
            if choice == 1:
                print("Result: ", obj.add())
            elif choice == 2:
                print("Result: ", obj.sub())
            elif choice == 3:
                print("Result: ", obj.mul())
            elif choice == 4:
                print("Result: ", round(obj.div(), 2))
            elif choice == 0:
                print("Exiting!")
            else:
                print("Invalid choice!!")
```

0.0.35 Point class and distance

```
In [ ]: import math


        class Point():
            def __init__(self, init_x, init_y):
                self.x = init_x
                self.y = init_y

            @staticmethod
            def distance(p1, p2):
                """Returns the distance between two points in 2d space."""
                delta_x = p2.x - p1.x
                delta_y = p2.y - p1.y
                return math.sqrt(delta_x**2 + delta_y**2)


        p = Point(0, 0)
        q = Point(6, 0)
        print(Point.distance(p, q))
```

0.0.36 Move rectangle

```
In [2]: class Rectangle(object):
            def __init__(self, l=0, w=0, cp=(0, 0)):
                self.length = l
                self.width = w
                self.corner_point = cp

            def move_rectangle(self):
                dx = int(input("Enter the value for dx:"))
                dy = int(input("Enter the value for dy:"))
                old_x = self.corner_point[0]
                old_y = self.corner_point[1]
                self.corner_point = (old_x + dx, old_y + dy)
            def __str__(self):
                return f"Length: {self.length} Width: {self.width} Corner Point: {self.corner_

        r = Rectangle(5, 7, (0, 0))
        print(r)
        r.move_rectangle()
        print(r)
```

```
Length: 5 Width: 7 Corner Point: (0, 0)
Enter the value for dx:3
Enter the value for dy:4
Length: 5 Width: 7 Corner Point: (3, 4)
```

```
In [ ]:
```