SET 1

1) In a given list of elements, all elements are equal except the one.Write a code to find the odd man out (Stray number)

numbers = [2, 2, 2, 2, 3, 2, 2, 2, 2, 2, 2]  
key = numbers[0]  
if key==numbers[1] or key==numbers[2]:  
 for i in range(0, len(numbers)):  
 if key==numbers[i]:  
 pass  
 else:  
 print(numbers[i])  
 break  
else:  
 print(key)

2) In a given list of elements, find the elements which is close to its mean

numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]  
mean=0  
for i in numbers:  
 mean+=i/len(numbers)  
print(mean)  
key=abs(numbers[0]-mean)  
means=[0, 0, 0, 0, 0, 0, 0, 0, 0, 0]  
i=0  
for j in numbers:  
 means[i]=abs(j-mean)  
 i=i+1  
 if abs(j-mean)<key:  
 key=abs(j-mean)  
for i in range(0, len(numbers)):  
 if means[i]==key:  
 break  
print(numbers[i])

3) Find the average speed of vehicle, given the distance travelled for fixed time intervals, e.g. [0, 0.1, 0.25, 0.45, 0.55, 0.7, 0.9, 1.0]

speed= [0, 0.1, 0.25, 0.45, 0.55, 0.7, 0.9, 1.0]  
average=0  
for i in speed:  
 average+=i/len(speed)  
print(average)

4) Find the no. of people in a bus, given the data of people onboarding & alighting at each station

number\_onboard=[5, 6, 12, 15]  
number\_alighting=[0, 2, 3, 3,]  
number\_bus=[0, 0, 0, 0]  
for i in range(0, len(number\_onboard)):  
 number\_bus[i]=number\_onboard[i]-number\_alighting[i]  
total=0  
for i in number\_bus:  
 total=total+i  
print('no. of people in the bus are:', total)

5) Find the missing number, given the original list and modified one

numbers=[1, 2, 3, 4, 5, 6, 7, 8, 9]  
mod\_no=[1, 3, 4, 5, 6, 7, 8, 9]  
add=0  
add1=0  
for i in numbers:  
 add+=i  
for i in mod\_no:  
 add1+=i  
print('the missing number is',add-add1)

6. Find the difference between two lowest numbers in the list

l = []  
x = int(input("Enter the number of elements in the list: "))  
print("Enter the list elements")  
for i in range(1, x+1):  
 y = int(input())  
 l.append(y)  
l.sort()  
d = l[1] - l[0]  
print(d)

7. In a given list, count no.of elements smaller than their mean

add = 0  
count = 0  
x = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]  
for i in x:  
 add = add + i  
 mean = add / len(x)  
for i in x:  
 if i < mean:  
 count = count + 1  
print(count)

Set 2

1. Correct the malformed time string, for e.g "5:70:65" to "6:11:05"

time = '05:70:65'  
sec=int(time[6:9])  
min=int(time[3:5])  
hrs=int(time[:2])  
if sec>60:  
 sec=sec%60  
 min+=1  
if min>60:  
 min=min%60  
 hrs+=1  
print(hrs, ':', min, ':', sec)

1. Correct the malformed date string, for e.g. "45/8/2018" to "14/9/2018"

date = '45/08/2018'  
m=int(date[3:5])  
d=int(date[:2])  
y=int(date[6:11])  
if d>31:  
 d=d%31  
 m+=1  
print(d, ':', m, ':', y)

1. Convert ip address from "a.b.c.d" format into integer and vice versa
2. def is\_valid(ip):  
    # Splitting by "."  
     
    ip = ip.split(".")  
     
    # Checking for the corner cases  
     
    for i in ip:  
     
    if (len(i) > 3 or int(i) < 0 or  
     
    int(i) > 255):  
    return False  
     
    if len(i) > 1 and int(i) == 0:  
    return False  
     
    if (len(i) > 1 and int(i) != 0 and  
     
    i[0] == '0'):  
    return False  
     
    return True  
     
     
   # Function converts string to IP address  
     
   def convert(s):  
    sz = len(s)  
     
    # Check for string size  
     
    if sz > 12:  
    return []  
     
    snew = s  
     
    l = []  
     
    # Generating different combinations.  
     
    for i in range(1, sz - 2):  
     
    for j in range(i + 1, sz - 1):  
     
    for k in range(j + 1, sz):  
     
    snew = snew[:k] + "." + snew[k:]  
     
    snew = snew[:j] + "." + snew[j:]  
     
    snew = snew[:i] + "." + snew[i:]  
     
    # Check for the validity of combination  
     
    if is\_valid(snew):  
    l.append(snew)  
     
    snew = s  
     
    return l  
     
     
   # Driver code  
     
   A = "25525511135"  
     
   B = "25505011535"  
     
   print(convert(A))  
     
   print(convert(B))
3. Check whether givenstring is isogram or not

def is\_isogram(word):  
  
 # Convert the word or sentence in lower case letters.  
 clean\_word = word.lower()  
  
 # Make an empty list to append unique letters  
 letter\_list = []  
  
 for letter in clean\_word:  
  
 # If letter is an alphabet then only check  
 if letter.isalpha():  
 if letter in letter\_list:  
 return False  
 letter\_list.append(letter)  
  
 return True  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 print(is\_isogram("Machine"))  
 print(is\_isogram("isogram"))  
 print(is\_isogram("GeeksforGeeks"))  
 print(is\_isogram("Alphabet "))

1. Given a string, find the mexican wave

s='hello'  
new=[]  
for i, val in enumerate(s[:]):  
 up=s[i].upper()  
 c=s[:i] + up + s[i+1:]  
 new.append(c)  
print(new)

6. Given a number, find the largest number by deleting single digit (order of digits will remain same)

def maxnumber(n, k):  
 # Function to return the  
 # largest number possible  
  
 for i in range(0, k):  
 # Generate the largest number  
 # after removal of the least K digits  
 # one by one  
 ans = 0  
 i = 1  
  
 while n // i > 0:  
 # Remove the least digit  
 # after every iteration  
 temp = (n // (i \* 10)) \* i + (n % i)  
 i \*= 10  
 # Store the numbers formed after  
 # removing every digit once  
  
 # Compare and store the maximum  
 if temp > ans:  
 ans = temp  
 n = ans  
  
 # Return the remaining number  
 # after K removals  
 return ans;  
  
  
n = 6358  
k = 1  
print(maxnumber(n, k))

7. Given a number, find the largest number by shuffling the digits

def printMaximum(inum):  
  
 # Hashed array to store count of digits  
 count = [0 for x in range(10)]  
  
 # Connverting given number to string  
 string = str(num)  
  
 # Updating the count array  
 for i in range(len(string)):  
 count[int(string[i])] = count[int(string[i])] + 1  
  
 # Result stores final number  
 result = 0  
 multiplier = 1  
  
 # traversing the count array  
 # to calculate the maximum number  
  
 for i in range(10):  
 while count[i] > 0:  
 result = result + ( i \* multiplier )  
 count[i] = count[i] - 1  
 multiplier = multiplier \* 10  
  
 # return the result  
 return result  
  
# Driver code  
num = 38293367  
print(printMaximum(num))

8. Compute the word frequency in given message

# Python code to find frequency of each word  
def freq(str):  
 # break the string into list of words  
 str = str.split()  
 str2 = []  
  
 # loop till string values present in list str  
 for i in str:  
  
 # checking for the duplicacy  
 if i not in str2:  
 # insert value in str2  
 str2.append(i)  
  
 for i in range(0, len(str2)):  
 # count the frequency of each word(present  
 # in str2) in str and print  
 print('Frequency of', str2[i], 'is :', str.count(str2[i]))  
  
  
def main():  
 str = 'apple mango apple orange orange apple guava mango mango'  
 freq(str)  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 main() # call main function

9. RGB to Hex conversion and vice versa, e.g. (255,0,255) into OxFF00FF

def rgb\_to\_hex(rgb):  
 return '%02x%02x%02x' % rgb  
print(rgb\_to\_hex((255, 0, 255)))

10. Generate accumulated strings.e.g. abcd ==> A Bb-Ccc-Dddd

n = int(input("Enter number of rows: "))  
  
a = 97  
  
for i in range(1,n+1):  
 for j in range(1, i+1):  
 print("%c" %(a), end="")  
 a +=1  
 print()