NAME : GINI CHACKO

ROLL : 8942

CLASS : SE COMPS B

BATCH : B

TOPIC : PYTHON EXPERIMENT 2

SET 1 : OOPS

1. Write a Python class to find a pair of elements (indices of the two numbers) from a given array whose sum equals a specific target number. Input: numbers= [10,20,10,40,50,60,70], target=50

```
class Pair :
    def __init__(self,nums,target):
        self.nums = nums;
        self.target = target;

    def pairofele(self):
        for i in self.nums:
            if(i < self.target):
                num2 = self.target - i
                if(num2 in self.nums):
                    print("The index are ", self.nums.index(i) + 1, self.nums.index(num2) + 1
                    break
numbers = [10,20,10,40,50,60,70]
target = 50

c1 = Pair(numbers,target)
c1.pairofele()
```

```
The index are  1 4
```

2. Write a Python class to get all possible unique subsets from a set of distinct integers.

   Input : [4, 5, 6] Output : [[], [6], [5], [5, 6], [4], [4, 6], [4, 5], [4, 5, 6]]

```
class sub:
    def f1(self, s1):
        return self.f2([], sorted(s1))

    def f2(self, curr, s1):
        if s1:
            return self.f2(curr, s1[1:]) + self.f2(curr + [s1[0]], s1[1:])
        return [curr]
a = []
n = int(input("Enter number of elements of list : "))
for i in range(0,n):
    b = int(input("Enter element : "))
    a.append(b)
print("Subsets : ")
print(sub().f1(a))
```

```
Enter number of elements of list : 3
Enter element : 4
Enter element : 5
Enter element : 6
Subsets : 
[[], [6], [5], [5, 6], [4], [4, 6], [4, 5], [4, 5, 6]]
```

SET 2 : Exception handling

An interactive caluculator: Program reads an expression as input and has to calculate the value of expression. The program throws an exception if the given expression is not in expected format. Example: Assumption , the expression uses operators that are binary (a+b),

Valid Input: 3+2 , Output: 5

Input: 3+2+4 , Output: Appropriate message

```
class Calculator:
    def perform(str):
        try:
            if len(str) > 3:
                raise ValueError('Please enter 2 operands and one operator only')
            else:
                val_1 = int(str[0])
                op = str[1]
                val_2 = int(str[2])
                if op == '+':
                    ans = val_1 + val_2
                elif op == '-':
                    ans = val_1 - val_2
                elif op == '*':
                    ans = val_1 * val_2
                elif op == '/':
                    ans = val_1 / val_2
                print(ans)
        except:
            print('Entry of two operands and one operator is only allowed!')
expr = []
expr = input('Enter the expression : ')
Calculator.perform(expr)
```

```
Enter the expression : 2/3+4
Entry of two operands and one operator is only allowed!
```