```
n1 = int(input("Enter start number: "))

n2 = int(input("Enter end number: "))

result = sum(range(n1, n2+1))

print("n1 =", n1)

print("n2 =", n2)

print("sum of all number from n1 to n2 is : ", result)
```

OUTPUT:

```
(P15ZZ22S126014) C:\Users\P15ZZ22S126014>python lab1.py
Enter start number: 2
Enter end number: 4
n1 = 2
n2 = 4
sum of all number from n1 to n2 is :  9
```

```
p=[]
n=[]
num = int(input("Enter the array size: "))
arr = [int(input("Enter the elements: ")) for j in range(num)]
for i in arr:
    if i >= 0:
        p.append(i)
    else:
        n.append(i)
print("All elements:", arr)
print("Positive array elements:", p)
print("Negative array elements:", n)
print("Sum of positive elements:", sum(p))
print("Sum of negative elements:", sum(n))
```

```
(P15ZZ22S126014) C:\Users\P15ZZ22S126014>python lab2.py
Enter the array size: 4
Enter the elements: -8
Enter the elements: 7
Enter the elements: 5
Enter the elements: -2
All elements: [-8, 7, 5, -2]
Positive array elements: [7, 5]
Negative array elements: [-8, -2]
Sum of positive elements: 12
Sum of negative elements: -10
```

```
def linear_search(a, n, x):
    for i in range(n):
        if a[i] == x:
            return i
    return -1
n = int(input("enter length of list: "))
a = [int(input("Enter the elements: ")) for i in range(n)]
print(a)
x = int(input("Enter element to be searched: "))
r = linear_search(a, n, x)
if r == -1:
    print("Element not found")
else:
    print("Element is found at at position: ", r + 1)
```

```
(P15ZZ22S126014) C:\Users\P15ZZ22S126014>python lab3.py
enter length of list: 5
Enter the elements: 4
Enter the elements: 6
Enter the elements: 8
Enter the elements: 7
Enter the elements: 2
[4, 6, 8, 7, 2]
Enter element to be searched: 8
Element is found at at position:  3
```

```
def binary_search(a, l, h, x):
    if h >= l:
        m = (h + l) // 2
        if a[m] == x:
            return m
        elif a[m] > x:
            return binary_search(a, l, m - 1, x)
        else:
            return binary_search(a, m + 1, h, x)
    else:
        return -1
n = int(input("Enter the length of the list: "))
a = [int(input("Enter the elements: ")) for _ in range(n)]
print(a)
x = int(input("Enter the element to be searched: "))
result = binary_search(a, 0, n - 1, x)
if result == -1:
    print("Element not found")
else:
    print("Element is found at position:", result + 1)
```

```
(P15ZZ22S126014) C:\Users\P15ZZ22S126014>python lab4.py
Enter the length of the list: 4
Enter the elements: 1
Enter the elements: 2
Enter the elements: 3
Enter the elements: 4
[1, 2, 3, 4]
Enter the element to be searched: 3
Element is found at position: 3
```

```
print("Stack operations")
a = []
n = int(input("Enter limit of array: "))
i = 1
while i:
    ch = int(input("Select your choice given below\n 1.Insertion\n 2.Deletion\n 3.Display\nEnter Choice: "))
    if ch == 1:
        if len(a) >= n:
            print("Stack is overflow")
        else:
            a.append(int(input("Enter an element: ")))
    elif ch == 2:
        if a:
            print("Deleted element is ", a.pop())
        else:
            print("Stack is empty")

    elif ch == 3:
        print("Stack elements are: ", a)
    else:
        print("Invalid choice")
        i = int(input("Do you want to continue press 1 or press 0: "))
```

```
(P15ZZ22S126014) C:\Users\P15ZZ22S126014>python lab5.py
Stack operations
Enter limit of array: 3
Select your choice given below
 1.Insertion
 2.Deletion
 3.Display
Enter Choice: 1
Enter an element: 70
Select your choice given below
 1.Insertion
 2.Deletion
 3.Display
Enter Choice: 1
Enter an element: 50
Select your choice given below
 1.Insertion
 2.Deletion
 3.Display
Enter Choice: 1
Enter an element: 20
Select your choice given below
 1.Insertion
 2.Deletion
 3.Display
Enter Choice: 2
Deleted element is  20
Select your choice given below
 1.Insertion
 2.Deletion
 3.Display
Enter Choice: 3
Stack elements are:  [70, 50]
Select your choice given below
 1.Insertion
 2.Deletion
 3.Display
Enter Choice: 0
Invalid choice
Do you want to continue press 1 or press 0: 0
```

```
def evaluate_expression(expression):
    operand = []
    operator = []
    precedence = ['+', '-', '*', '/']
    for token in expression:
        if token.isdigit():
            operand.append(int(token))
        elif token in precedence:
            operator.append(token)
    while operator:
        right_operand = operand.pop(0)
        left_operand = operand.pop(0)
        opr = operator.pop(0)
        if opr == '+':
            result = left_operand + right_operand
        elif opr == '-':
            result = left_operand - right_operand
        elif opr == '*':
            result = left_operand * right_operand
        elif opr == '/':
            result = left_operand / right_operand
        operand.append(result)
    print(eval(expression.replace(' ','')))
    return operand.pop()
expression = input("Enter an expression: ")
result = evaluate_expression(expression)
```

OUTPUT:

```
(P15ZZ22S126014) C:\Users\P15ZZ22S126014>python lab6.py
Enter an expression: (4+5)*8
72
```