Task 5

Machine Learning and Deep learning Summer Internship

Registration id -SIRSS2290

Mohammad Sharique Salman

Q1. Given a list of integers, write a function to return the sum of all prime numbers in that list.

In [1]:

```
num_list=[]
size = int(input("Enter the size of the list : "))
for i in range (size):
    x=int(input("Enter no. \n"))
    num_list.insert(i,x)
    i+=1
print(num_list)
def sprime(num_list):
    prime_list = []
    for i in num_list:
        for j in range(2, i-1 ):
            if i%j == 0:
                break
        else:
            prime_list.append(i)
    return sum(prime_list)
sprime(num_list)
```

```
Enter the size of the list : 4
Enter no.
1
Enter no.
2
Enter no.
5
Enter no.
7
[1, 2, 5, 7]
```

Out[1]:

```
15
```

Q2. Given a list of integers, write a function to check whether the list is strictly increasing or not.

In [2]:

```
num_list=[]
size = int(input("Enter the size of the list : "))
for i in range (size):
    x=int(input("Enter no. \n"))
    num_list.insert(i,x)
    i+=1
print(num_list)

if (all(i < j for i, j in zip(num_list, num_list[1:]))):
    print("Yes, the list is stritly increasing.")
```

```
else:
    print("No, the list is not strictly increasing.")
```

```
Enter the size of the list : 4
Enter no.
2
Enter no.
4
Enter no.
7
Enter no.
8
[2, 4, 7, 8]
Yes, the list is stritly increasing.
```

Q3. Write a function to check whether a given list is expanding or not (the difference between adjacent elements should keep on increasing).

In [3]:

```
arr =[]
size = int(input("Enter the size of the list : "))
for i in range (size):
    x=int(input("Enter no. \n"))
    arr.insert(i,x)
    i+=1
def is_expanding(arr):
    if all(arr[i] <= arr[i + 1] for i in range(len(arr) - 1)):
        return "List is expanding"
    elif all(arr[i] >= arr[i + 1] for i in range(len(arr) - 1)):
        return "List is expanding"
    else :
        return "List is not expanding"

is_expanding(arr)
```

```
Enter the size of the list : 6
Enter no.
43
Enter no.
56
Enter no.
78
Enter no.
175
Enter no.
345
Enter no.
566
```

Out[3]:

```
'List is expanding'
```

Q4. Write a function to calculate all permutations of a given string. (Without using itertools)

In [4]:

```
string = input("Enter a string to get all possible permutations: ")
def permutations(string):
    permutation_list = []
    if len(string) == 1:
        return [string]
    else:
        for char in string:
            [permutation_list.append(char + a) for a in permutations(string.replace(char
```

```
, "", 1))]
    return permutation_list
permutations(string)
```

```
Enter a string to get all possible permutations: car
```

Out[4]:

```
['car', 'cra', 'acr', 'arc', 'rca', 'rac']
```

Thank You