**# 1 Table**

num = int(input("enter the number for get a table of the number:"))

for i in range(1,11):

    print(num, "\*", i, "=", num\*i)

#--------------------------------------------------------------

**# 2 Prime**

num = int(input("enter the number for check whether given number is prime or not: "))

prime = True

for i in range(2,num):

    if(num % i == 0):

        prime = False

        break

if prime:

    print("Prime")

else:

    print("Not a Prime")

#--------------------------------------------------------------

**#3. factorial of a Number**

def factorial(n):

    if n == 0 or n == 1:

        return 1

    else:

        return n \* factorial(n-1)

num = int(input("Enter the number: "))

print('Result:', factorial(num))

**# 2.a**

list1 = [1,2,3,4,5]

list2 = ['a','b','c','d','e','f']

# Nested

nested\_list = [list1, list2]

print(nested\_list)

# Length

print(len(list1))

print(len(list2))

# Concatenation

con\_cat = list1 + list2

print(con\_cat)

# membership

print(1 in list1)

print('a' in list2)

print('z' in list2)

# Iteration

for i in list1:

    print(i, end="")

    print()

# Indexing

print(list1[3])

print(list2[2])

# Slicing

print(list1[1:4])

print(list2[1:4])

# list1.append, .insert, .extend, .remove, .pop, .clear, .index, .count, .sort, .reverse, .copy

#----------------------------------------------------------------------------------------------

**# 2.b**

my\_list = [1,2,3,4,5,6,7,8,9,10]

print("Original List:")

print(my\_list)

# Insert

my\_list.insert(3,11)

print("List after inserting 11 at index 3:")

print(my\_list)

# Append

my\_list.append(12)

print("List after appending 12:")

print(my\_list)

# Extend

my\_list.extend([13,14,15])

print("List after extending with [13,14,15]:")

print(my\_list)

# Delete

my\_list.remove(11)  # Corrected index

print("List after removing 11:")

print(my\_list)

**# 3. Chat-Bot**

def chatbot():

    conversion = {

        "1":"NS LONI-1",

        "2":"NS LONI-2",

        "3":"NS LONI-3"

    }

    print("Hello I am Chatbot ask me anything")

    while True:

        user\_input = input("you:").lower()

        if user\_input in conversion:

            print("Chatbot: ", conversion[user\_input])

        if user\_input  == "good by":

            break

        else:

            print("Chatbot: Oh I am not Understand")

chatbot()

**# 4. Set Operation**

**# Set Operation's**

**setA = {1,2,3,4,5}**

**setB = {4,5,6,7,8,9}**

**# Union**

**union\_result = setA.union(setB) # {1,2,3,4,5,6,7,9}**

**print(union\_result)**

**# Inter-Section**

**intersection\_result = setA.intersection(setB) # {4,5}**

**print(intersection\_result)**

**# Difference**

**difference\_result = setA.difference(setB) # {1,2,3}**

**print(difference\_result)**

**# Sysmetric Difference**

**Sysmetric\_Difference\_result = setA.symmetric\_difference(setB) #{1,2,3,6,7,8,9}**

**print(Sysmetric\_Difference\_result)**

**# Check of set is subset or not**

**sub\_set\_result = setA.issubset(setB) # False**

**print(sub\_set\_result)**

**# Super Set**

**Super\_set = setA.issubset(setB)  # False**

**print(Super\_set)**

**# Tw0 set are Disjoint**

**disjoint\_set = setA.isdisjoint(setB) # False**

**print(disjoint\_set)**

**# 5.a.String Based**

**def count\_occurence(s1, s2):**

**count = s2.count(s1)**

**return count**

**s1=input("Enter the substring to find(s1): ")**

**s2=input("Enter the Main string: ")**

**occurence = count\_occurence(s1, s2)**

**print(f"The substring of{s1} Occurs {occurence} time(s) in the String{s2}")**

**# 5.bString Based**