**Problem Solving**

**# intro loops and conditional statements**

**# 1. elif program**

n=int(input("Enter A number"))

# n=10

if n>10:

    print(n," is greater than 10")

elif n<10:

    print(n," is less than 10")

else:

    print("Not A Number")

**# 2. even or odd**

n=15

if n%2==0:

    print(n, "is even")

else:

    print(n,"is even")

# 3. prime numbers

n=11

c=0

for x in range(2,n-1,1):

    if n%x==0:

        c+=1

if c==0:

    print(n,"is a prime")

else:

    print("not a prime")

**# 4. sum of individual numbers**

n="1234"

sum=0

for i in n:

    sum+=int(i)

print(sum)

**# 5. sum of odd individual numbers present in given number**

n="237"

sum=0

for i in n:

    if int(i)%2 !=0:

        sum+=int(i)

print(sum)

**# 6. difference b/w even and odd**

n="2376"

esum=0

osum=0

for i in n:

    if int(i)%2 ==0:

        esum+=int(i)

    else:

        osum+=int(i)

if esum >osum:

    print(esum-osum)

else:

    print(osum-esum)

**# 7. which is largest sum even or odd**

n="2376"

esum=0

osum=0

for i in n:

    if int(i)%2 ==0:

        esum+=int(i)

    else:

        osum+=int(i)

if esum >osum:

    print(esum,"even sum is greater")

else:

    print(osum,"odd sum is greater")

**2. prime numbers,not prime, maximum and minimum**

**# for loop(practice)**

**# 1. practice**

for i in "banana":

    print(i)

**# 2. practice**

for i in range(1,25,2):

    print(i)

**# 3. practice**

x= int(input())

for i in range(1,x,1):

    print(i)

**# 1. Sum of prime numbers in a given number**

n="123456789"

sum=0

nsum=0

for i in n:

    x=int(i)

    # prime number

    c=0

    for i in range(2,x-1,1):

        if x%i==0:

            c+=1

    if c==0:

        sum+=x

    if c!=0:

        nsum+=x

print(sum, "is sum of prime numbers")

print(nsum, "is sum of non prime numbers")

**# 2 largest prime in given number**

n="46537129"

y=0

for i in n:

    x=int(i)

    # prime

    c=0

    for j in range(2,x-1,1):

        if x%j==0:

            c+=1

    if c==0:

        if x>y:

            y=x

print(y)

**# 3. two largest prime numbers**

n="46537129"

y=0

z=0

for i in n:

    x=int(i)

    # prime

    c=0

    for j in range(2,x-1,1):

        if x%j==0:

            c+=1

    if c==0:

        if x>y:

            z=y

            y=x

print(y)

print(z)

**# 4. minimum prime number in given number**

n="4653729"

max=0

for i in n:

    x=int(i)

    # prime

    c=0

    for j in range(2,x-1,1):

        if x%j==0:

            c+=1

    if c==0:

        if x>max:

            max=x

        # min

        min=max

        if min>x:

            min=x

print(max)

print(min)

**# 5. sum of largest and smallest prime number in given digits**

n="4653729"

max=0

sum=0

for i in n:

    x=int(i)

    # prime

    c=0

    for j in range(2,x-1,1):

        if x%j==0:

            c+=1

    if c==0:

        if x>max:

            max=x

        # min

        min=max

        if min>x:

            min=x

print(max)

print(min)

sum=max+min

print(sum, "is sum of largest and smallest prime num")

**# 6.  largest non prime in given number**

n="46537129"

y=0

for i in n:

    x=int(i)

    # prime

    c=0

    for j in range(2,x-1,1):

        if x%j==0:

            c+=1

    if c!=0:

        if x>y:

            y=x

print(y)

**3. Fibonacci Series, factorial and prime numbers**

**# 1. fibonacci series**

n=int(input("Enter range"))

i=1

a=0

b=1

while i<=n:

    c=a+b

    print(a)

    a=b

    b=c

    i+=1

**# 2. sum of given numbers**

n="123454"

sum=0

for i in n:

    sum+=int(i)

print(sum)

**# 3. Factorial of given number**

n=int(input("Enter no:"))

fact=1

for i in range(1,n+1,1): # 5! 1\*2\*3\*4\*5=120

    fact=fact\*i

print("Factorial of {n} is:",fact)

**4.  Factorial of individual no's of given number's**

n=input("Enter no:")

for j in n:

    fact=1

    j=int(j)

    for i in range(1,j+1,1):

        fact=fact\*i

    print("Factorial of {n} is:",fact)

**# 5. removing duplicate numbers in given number**

n="324356"

unique=""

for i in n:

    if i not in unique:

        unique+=i

print(unique)

**6. removing duplicate numbers in given number and Factorial of individual no's of given number**

import math

n = "324356"

unique = ""

# Remove duplicate digits

for i in n:

    if i not in unique:

        unique += i

print("Unique digits:", unique)

# Calculate factorial of each unique digit

for j in unique:

    num = int(j)

    fact = math.factorial(num)

    print("Factorial of ",num," is: ",fact)

**# 7. write a program to print the factorial of a number in the below format**

# i/p: 5

# o/p: 1\*2\*3\*4\*5 = 120

n=int(input("Enter no:"))

fact=1

st="1"

for i in range(2,n+1,1):

    fact=fact\*i

    st+="\*"+str(i) #1\*2\*3\*4\*5

print(st, " = ", fact)

**# 8. write a program to print sum of prime numbers in fibonaci series**

import math

n = int(input("Enter range: "))

i = 1

a, b = 0, 1

prime\_sum = 0

while i <= n:

    c = a + b

    a, b = b, c

    i += 1

    # Prime Check

    if a > 1:  # 1 is not prime

        is\_prime = True

        for x in range(2, int(math.sqrt(a)) + 1):

            if a % x == 0:

                is\_prime = False

                break

        if is\_prime:

            prime\_sum += a  # Add to sum if prime

print("Sum of prime numbers in Fibonacci series:", prime\_sum)

**9. Next Prime number**

**ex: n=11  nxt= 13,17,19**

a=13

num=a

nxt=0

nxt\_prime=False

while(not nxt\_prime):

    num+=1

    count=0

    for i in range(2,num,1):

        if num%i==0:

            count+=1

            break

    if count==0:

        nxt=num

        nxt\_prime=True

print(nxt)

**10. previous prime number**

**ex: n=17  prev= 7,11,13**

a=13

num=a

prev=0

prev\_prime=False

while(not prev\_prime):

    num-=1

    count=0

    for i in range(2,num,1):

        if num%i==0:

            count+=1

            break

    if count==0:

        prev=num

        prev\_prime=True

print(prev)

**11. Nearest prime number**

**ex: n=5  prev=3  nxt=7 (same difference print null)**

a=int(input("Enter Number for neaarest prime"))

num=a

# Next prime

nxt=0

ndiff=0

nxt\_prime=False

while(not nxt\_prime):

    num+=1

    ndiff+=1

    count=0

    for i in range(2,num,1):

        if num%i==0:

            count+=1

            break

    if count==0:

        nxt=num

        nxt\_prime=True

print("next prime is: ",nxt)

# Prev prime

num=a

prev=0

pdiff=0

prev\_prime=False

while(not prev\_prime):

    num-=1

    pdiff+=1

    count=0

    for i in range(2,num,1):

        if num%i==0:

            count+=1

            break

    if count==0:

        prev=num

        prev\_prime=True

print("previous prime is: ",prev)

# comapre

if ndiff<pdiff:

    print("nearest prime is: ",nxt)

elif pdiff<ndiff:

    print("nearest prime is: ",prev)

elif pdiff==ndiff:

    print("nearest prime difference is Null ")

else:

    print("Invalid number ")

**# 12. print next 10 prime numbers of given number**

a=2

num=a

count=0

while count < 10:

    num+=1

    fact=0

    for i in range(2,num,1):

        if num%i==0:

            fact+=1

            break

    if fact==0:

        count+=1

        print(num)

**4. Duplicate Numbers**

**# 1. unique numbers in given number**

n="324356"

unique=""

for i in n:

    if i not in unique:

        unique+=i

print(unique)

**# 2. Duplicate Numbers**

n = "43723397"

unq = ""

dup = ""

# Extract unique digits

for i in n:

    if i not in unq:

        unq += i

# Find duplicate digits

for i in unq:  # Loop through unique digits

    count = 0

    for j in n:  # Count occurrences in n

        if i == j:

            count += 1

    if count > 1:  # If digit appears more than once, add to dup

        dup += i

print("Duplicate digits:", dup)

**(or)**

n = "4372339"

seen = set()

duplicates = set()

for digit in n:

    if digit in seen:

        duplicates.add(digit)

    seen.add(digit)

print("Duplicate digits:", duplicates if duplicates else "No duplicates")

**#3. count how many times each digit appears in this string.**

n="122345546"

unq=""

for i in n:

    if i not in unq:

        unq+=i

print(unq)

for i in unq:

    count=0

    for j in n:

        if i==j:

            count+=1

    print(i, count)

**#4. count duplicate values of each number present in given number**

n="122345546"

unq=""

for i in n:

    if i not in unq:

        unq+=i

print(unq)

for i in unq:

    count=0

    for j in n:

        if i==j:

            count+=1

    if count>1:

        print(i, count)

**#5. Minimum and maximum number in duplicate**

n = "43575235397"

unq = ""

dup = ""

# Extract unique digits

for i in n:

    if i not in unq:

        unq += i

# Find duplicate digits

for i in unq:

    count = 0

    for j in n:

        if i == j:

            count += 1

    if count > 1:

        dup += i

print("Duplicate digits:", dup)

min=dup[0]

max=dup[0]

for i in dup:

    if i < min:

        min=i

    if i>max:

        max=i

print(min)

print(max)

**#6. sum of min numbers of unique and duplicate Number**

n = "43575235397"

unq = ""

dup = ""

# Extract unique digits

for i in n:

    if i not in unq:

        unq += i

# Find duplicate digits

for i in unq:

    count = 0

    for j in n:

        if i == j:

            count += 1

    if count > 1:

        dup += i

munq=unq[0]

mdup=dup[0]

for i in unq:

    if i < munq:

        munq=i

for i in dup:

    if i<mdup:

        mdup=i

print("Duplicate digits:", dup)

print("Unique digits:", unq)

print("minimum duplicate digits:", mdup)

print("Minimum Unique digits:", munq)

sum=int(munq)+int(mdup)

print("sum of minimum number of unq and dup",sum)

**# 7. reverse of individual prime numbers**

**#Example: i/p: 3 5 7 9 11 13 15   o/p: 3 5 7 9 11 31 51**

num=20

for n in range(1,num):

    if n>1:

        c=0

        for i in range(2,n):

            if n%i==0:

                c+=1

                break

        if c==0:

            t=str(n)

            t1=t[::-1]

            print(t1, end=" ")

**LCM**

**#1. LCM of two numbers**

num1 = 12

num2 = 18

max\_num = max(num1, num2)

while True:  # Infinite loop to find LCM

    if max\_num % num1 == 0 and max\_num % num2 == 0:

        print("LCM:", max\_num)

        break  # Exit loop once LCM is found

    max\_num += 1  # Increment and check next number

**#2. Calculate power of Number(using loop only)**

num=5

power=3

sol=1

for i in range(0,power,1):

    sol\*=num

print(sol)

print(5\*\*3)

**#3. take one integer n and loop till n pass to the function.**

**#create a function that takes one integer parameter multiple two with integer.**

num=5

s=0

for i in range(1,num+1,1):

    s=i\*2

    print(s)

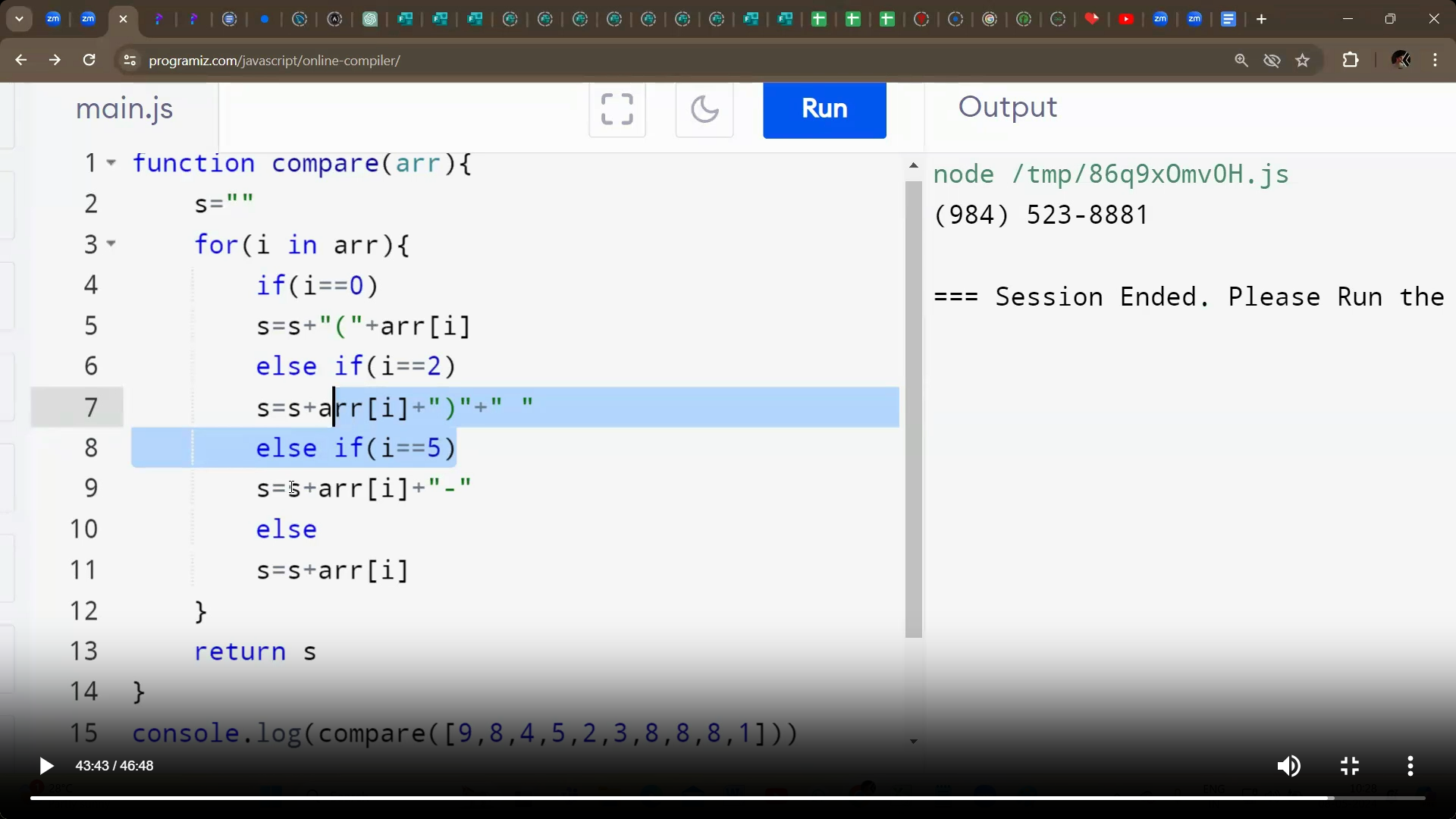
**# 4. create a function that takes a array of 10 numbers and formated as a phone number**

def format\_phone\_number(numbers):

    return "({}{}{}) {}{}{}-{}{}{}{}".format(\*numbers)

numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 0]

print(format\_phone\_number(numbers))



**Arrays**

**#1. Sort method**

numbers = [5, 2, 3, 7, 202]

for i in range(len(numbers)):

    for j in range(i + 1, len(numbers)):  # Compare all pairs

        if numbers[i] > numbers[j]:  # Swap if the order is wrong

            numbers[i], numbers[j] = numbers[j], numbers[i]

print(numbers)  # Sorted List

**#2. print True if given array is increasing order**

num = [1, 2, 3, 7, 8]

c=0

for i in range(0,len(num)-1):

    if num[i+1]>num[i]:

        c+=1

if c==len(num)-1:

    print("true")

**#3. Decreacing order**

numbers = [5, 2, 3, 7, 202]

for i in range(len(numbers)):

    for j in range(i + 1, len(numbers)):

        if numbers[i] < numbers[j]:

            numbers[i], numbers[j] = numbers[j], numbers[i]

print(numbers)

