**1.FACTORS OF A NUMBER :**

**TCP:server**

import socket

Host='127.0.0.1'

Port=12345

s=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

s.bind((Host,Port))

s.listen(5)

conn,adr=s.accept()

print(adr)

no=conn.recv(2048)

print("the received number is",no)

x=no.decode('ASCII')

y=int(x)

factor=[]

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

if y%i==0:

factor.append(i)

conn.send(str(factor).encode('ASCII'))

print("factor sent to the client")

s.close()

**TCP:CLIENT:**

import socket

Host='127.0.0.1'

Port=12345

s=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

s.connect((Host,Port))

print("connection established successfully")

n=(input("enter a number: "))

s.send(n.encode('ASCII'))

data=s.recv(2048)

result=data.decode('ASCII')

print("factors are ",result)

s.close()

**UDP:SERVER**

import socket

Host='127.0.0.1'

Port=12345

s=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

s.bind((Host,Port))

no,adr=s.recvfrom(2048)

print("the received number is",no)

x=no.decode('ASCII')

y=int(x)

factor=[]

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

if y%i==0:

factor.append(i)

s.sendto(str(factor).encode('ASCII'),adr)

print("factor sent to the client")

s.close()

**UDP:CLIENT;**

import socket

Host='127.0.0.1'

Port=12345

s=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

n=(input("enter a number: "))

s.sendto((n.encode('ASCII')),(Host,Port))

data,adr=s.recvfrom(2048)

result=data.decode('ASCII')

print("factors are ",result)

s.close()

**2.SUBSTRING:**

**TCP:CLIENT:**

import socket

import sys

sock=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM) # Create a TCP/IP socke

host='127.0.0.1' # The server's hostname or IP address

port=10000 # The port used by the server

server\_Address=('127.0.0.1',10000)

print("connecting to",(host,port))

sock.connect(server\_Address) # Connect the socket to the port where the server is listening

msg=sock.recv(1024) ## calls sock.recv() to read the server’s reply

print(msg.decode("utf-8"))

string=input("enter a string")

sock.sendall(bytes(string,'utf-8')) ## calls sock.sendall() to send its message

print("string sent to server")

substring=input("enter substring")

sock.sendall(bytes(substring,'utf-8'))

print("substring sent to server")

result=sock.recv(1024)

print(result.decode("utf-8"))

print("closing socket")

sock.close()

**TCP:SERVER:**

import socket

import sys

host='127.0.0.1' # Standard loopback interface address (localhost)

port=10000 # Port to listen on (non-privileged ports are > 1023)

sock=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM) # Create a TCP/IP socket

## AF\_INET refers to the address family ipv4

## SOCK\_STREAM means connection oriented TCP protocol.

server\_Address=('127.0.0.1',10000)

sock.bind(server\_Address) # Bind the socket to the port

sock.listen()

print('waiting for connection')

while True:

connection,address=sock.accept()

##accept() wait for incoming connection

print("connection from",address)

connection.sendall(bytes("connection established","utf-8")) ##connection.sendall sends the msg in bytes to client

string=connection.recv(1024) ## connection.recv() recieves the msg from client

print("The string recieved from client is:")

print(string.decode("utf-8"))##decode("utf-8") is used to decode from bytes to string

substring=connection.recv(1024) ## connection.recv() recieves the msg from client

print("The substring recieved from client is:")

print(substring.decode("utf-8"))

a=string.find(substring)

result=str(a)

if(result !=-1):

connection.sendall(bytes(result,"utf-8"))

else:

connection.sendall(bytes("The substring is not present","utf-8"))

**UDP:serever**

import socket

import sys

host='127.0.0.1'

port=10000

sock=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

server\_Address=('127.0.0.1',10000)

sock.bind(server\_Address)

print('waiting for receive msg')

while True:

string,address=sock.recvfrom(1024)

print("The string recieved from client is:")

print(string.decode("utf-8"))

substring,address=sock.recvfrom(1024)

print("The substring recieved from client is:")

print(substring.decode("utf-8"))

a=string.find(substring)

b=str(a)

result=bytes(b,'utf-8')

if(result !=-1):

sock.sendto(result,address)

else:

connection.sendall(bytes("The substring is not present","utf-8"))

**CLIENT:**

import socket

import sys

sock=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

host='127.0.0.1'

port=10000

server\_Address=('127.0.0.1',10000)

string=input("enter a string")

stringmsg=bytes(string,"utf-8")

sock.sendto(stringmsg,server\_Address)

print("string sent to server")

substring=input("enter substring")

substringmsg=bytes(substring,"utf-8")

sock.sendto(substringmsg,server\_Address)

print("substring sent to server")

result,server=sock.recvfrom(1024)

print(result.decode("utf-8"))

print("closing socket")

sock.close()

**3.PALINDROME :**

**TCP:Server;**

import socket

server=socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

server.bind((&#39;127.0.0.1&#39;,5884))

server.listen(1)

conn,addr=server.accept()

print(addr)

data=conn.recv(1024)

string=str(data)

string=string.split(&quot;&#39;&quot;)

string=string[1]

if(string==string[::-1]):

conn.sendall(b&quot;The string is a palindrome&quot;)

else:

conn.sendall(b&quot;Not a palindrome&quot;)

server.close()

**client:**

import socket

client=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

client.connect((&#39;127.0.0.1&#39;,5884))

string=input((&quot;Enter a string:&quot;))

string=bytes(string,&#39;utf-8&#39;)

client.sendall(string)

res=client.recv(7024)

print(res)

client.close()

**Udp server;**

import socket

server=socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

server.bind((&#39;127.0.0.1&#39;,5884))

data,addr=server.recvfrom(1024)

print(addr)

string=str(data)

string=string.split(&quot;&#39;&quot;)

string=string[1]

if(string==string[::-1]):

server.sendto(b&quot;The string is a palindrome&quot;,addr)

else:

server.sendto(b&quot;Not a palindrome&quot;,addr)

server.close()

**Udp client;**

import socket

client=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

string=input((&quot;Enter a string:&quot;))

string=bytes(string,&#39;utf-8&#39;)

client.sendto(string,(&#39;127.0.0.1&#39;,5884))

res,addr=client.recvfrom(7024)

print(res)

client.close()

**4.STRING CANCADENATE:**

**TCP:CLIENT**:

import socket

s=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

s.connect((&#39;127.0.0.1&#39;,14000))

msg=s.recv(1024)

print(msg.decode(&quot;utf-8&quot;))

s1=input(&quot;Enter 1st string&quot;)

s2=input(&quot;Enter 2nd string&quot;)

s.sendall(bytes(s1,&quot;utf-8&quot;))

print(&quot;1st string sent to server&quot;)

s.sendall(bytes(s2,&quot;utf-8&quot;))

print(&quot;2nd string sent to server&quot;)

result=s.recv(1024)

print(result.decode(&quot;utf-8&quot;))

s.close()

**SERVER:**

import socket

s=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

s.bind((&#39;127.0.0.1&#39;,14000))

s.listen()

print(&quot;Waiting for connection...&quot;)

while True:

clt,adr=s.accept()

clt.sendall(bytes(&quot;connection established...&quot;,&quot;utf-8&quot;))

print(&quot;1st string received from client&quot;)

string1=clt.recv(1024)

print(string1.decode(&quot;utf-8&quot;))

print(&quot;2nd string received from client&quot;)

string2=clt.recv(1024)

print(string2.decode(&quot;utf-8&quot;))

a=string1+string2

result=str(a)

clt.sendall(bytes(result,&quot;utf-8&quot;))

**UDP:CLIENT:**

import socket

s=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

server\_address=(&#39;127.0.0.1&#39;,14000)

n=input(&quot;Enter a string:&quot;)

n1=input(&quot;Enter a string:&quot;)

data1=bytes(n,&#39;utf-8&#39;)

s.sendto(data1,server\_address)

data2=bytes(n1,&#39;utf-8&#39;)

s.sendto(data2,server\_address)

result,server\_address=s.recvfrom(1024)

print(result.decode(&#39;utf-8&#39;))

s.close()

**Server:UDP**

import socket

s=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

s.bind((&#39;127.0.0.1&#39;,14000))

print(&quot;Waiting for message...&quot;)

while True:

data1,address=s.recvfrom(1024)

print(data1.decode(&#39;utf-8&#39;))

data2,address=s.recvfrom(1024)

print(data2.decode(&#39;utf-8&#39;))

a=data1+data2

b=str(a)

c=bytes(b,&#39;utf-8&#39;)

s.sendto(c,address)

print(&quot;Result sent to client&quot;)

**5.SWAP TWO NUMBERS:**

**TCP:SERVER**

import socket

Host='127.0.0.1'

Port=12345

s=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

s.bind((Host,Port))

s.listen()

while True:

conn,addr=s.accept()

print("Connected successfully with: ",addr)

a1 = conn.recv(4100)

print("Decoding the received data")

a1 = list(a1)

print("Swapping the numbers")

print("First number: ",a1[0])

print("Second number: ",a1[1])

temp=a1[0];

a1[0]=a1[1];

a1[1]=temp;

conn.send(bytes(a1))

print("The result succesfully sent to client!")

conn.close()

**CLIENT:**

import socket

Host='127.0.0.1'

Port=12345

s=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

s.connect((Host,Port))

print("Connection established successfully!")

a1 = []

for i in range(2):

a1.append(int(input('Enter a number: ')))

s.sendall(bytes(a1))

a2 = s.recv(4100)

print(list(a2))

s.close()

**UDP:CLIENT**

import socket

import pickle

client=socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

host='127.0.0.1'

sockett\_add=(host,12345)

n1=input("Enter the first number: ")

n2=input("Enter the second number: ")

n=[n1,n2]

MESSAGE = pickle.dumps(n)

sent=client.sendto(MESSAGE,sockett\_add)

info,address = client.recvfrom(4096)

ans=pickle.loads(info)

print(str(ans))

client.close()

**SERVER:**

import socket

import pickle

server=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

host = '127.0.0.1'

sockett\_add=(host,12345)

server.bind(sockett\_add)

print("Receiving input ; The swapped numbers are : ")

info, address = server.recvfrom(4096)

data= pickle.loads(info)

print(data)

ans=[data[1],data[0]]

msg=pickle.dumps(ans)

send=server.sendto(msg,address)

server.close()

**6.SUM OF N DIGITS:**

**TCP:CLIENT**

n=input("enter the number")

n=bytes(n,'utf-8')

import socket

socket1 = socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

add\_server = ('localhost',12345)

socket1.connect(add\_server)

socket1.sendall(n)

data=socket1.recv(1024)

print(str(data))

socket1.close()

**SERVER:**

import socket

import numpy as np

sockett = socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

add\_server = ('localhost',12345)

sockett.bind(add\_server)

sockett.listen(1)

conn,addr=sockett.accept()

data=conn.recv(1024)

data=int(data)

data1=list(map(int,str(data)))

data2=np.array(data1)

total=np.sum(data2)

conn.sendall(bytes(str(total),'utf-8'))

**UDP:SERVER**

import socket

import numpy as np

server=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

host = '127.0.0.1'

sockett\_add=(host,12345)

server.bind(sockett\_add)

info, address = server.recvfrom(4096)

data=int(info)

data1=list(map(int,str(data)))

data2=np.array(data1)

total=np.sum(data2)

send=server.sendto(bytes(str(total),'utf-8'),address)

server.close()

**CLIENT:**

import socket

client=socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

host='127.0.0.1'

sockett\_add=(host,12345)

n=input("enter the number")

n=bytes(n,'utf-8')

sent=client.sendto(n,sockett\_add)

info,address = client.recvfrom(4096)

print(info)

client.close()

**7.PASCAL'S TRIANGLE:**

**TCP:SERVER:**

import socket

import threading

host = '127.0.0.1'

port = 55555

server = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

server.bind((host, port))

server.listen()

print("waiting for connection")

conn,addr = server.accept()

print("connected by",addr)

while True:

data = conn.recv(2048)

a = data.decode('ASCII')

rows = int(a)

def fact(n):

res=1

for c in range(1,n+1):

res = res\*c

return res

for i in range(0, rows):

for j in range(1, rows-i):

print(" ", end="")

for k in range(0, i+1):

coff = int(fact(i)/(fact(k)\*fact(i-k)))

print(" ", coff, end="")

print()

**CLIENT:**

import socket

host='127.0.0.1'

port= 55555

server = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

server.connect((host,port))

print("connection established")

rows =(input("Enter the number of rows : "))

server.send(rows.encode('ASCII'))

server.close()

**UDP:SERVER:**

import socket

sock=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

sock.bind(('127.0.0.1',55555))

while True:

data,addr=sock.recvfrom(4096)

print(str(data))

message="Hello I am UDP Server"

sock.sendto(str.encode(message),addr)

def fact(n):

res=1

for c in range(1,n+1):

res = res\*c

return res

rows =int(input("Enter the number of rows : "))

for i in range(0, rows):

for j in range(1, rows-i):

print(" ", end="")

for k in range(0, i+1):

coff = int(fact(i)/(fact(k)\*fact(i-k)))

print(" ", coff, end="")

print()

**CLIENT:**

import socket

client\_socket=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

msg="hello UDP Server"

client\_socket.sendto(str.encode(msg),('127.0.0.1',55555))

data,addr=client\_socket.recvfrom(4096)

print("Server Says")

print(str(data))

client\_socket.close()

**8.ODD OR EVEN:**

**TCP:SERVER**

import socket

host = '127.0.0.1'

port = 14100

s = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

s.bind((host,port))

s.listen()

conn , addr = s.accept()

with conn:

print("Connected by ", addr)

while True:

data = conn.recv(2048)

if not data:

break

print("\nDecoding the received data...")

x = data.decode('ASCII')

y = int(x)

print("\nChecking the result...")

if y % 2 == 0:

result = "Even Number"

conn.send(result.encode('ASCII'))

else :

result = "Odd Number"

conn.send(result.encode('ASCII'))

print("\nResult sent to client...")

**CLIENT:**

import socket

host = '127.0.0.1'

port = 14100

s = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

s.connect((host,port))

print("Connection Established...")

#getting the number from user

n = (input("\n Enter the number to check whether it is odd or even : "))

s.send(n.encode('ASCII'))

data = s.recv(2048)

result = data.decode('ASCII')

print(n , " is an " , result)

s.close()

**UDP:SERVER;**

import socket

s = socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

host = '127.0.0.1'

port = 14200

sock\_add = (host,port)

s.bind(sock\_add)

data,add = s.recvfrom(2048)

print("\n Message is being received from " , add)

x = data.decode('ASCII')

y = int(x)

print("\n Checking the result ")

if y%2 == 0:

result = "Even Number"

s.sendto(result.encode('ASCII'), add)

else:

result = "Odd Number"

s.sendto(result.encode('ASCII'), add)

print("\n Result sent to client...")

s.close()

**CLIENT:**

import socket

c = socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

host = "127.0.0.1"

port = 14200

sock\_add = (host,port)

#getting the input from user

n = input("\n Enter the number to check whether it is odd or even : ")

c.sendto(n.encode('ASCII'), sock\_add)

print("Message sent !!! ")

data, add = c.recvfrom(2048)

result = data.decode('ASCII')

print(n , " is an " , result)

**9. FIBO NACCI:**

**TCP;SERVER**

import socket

s=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

s.bind(('127.0.0.1',17000))

s.listen()

print("waiting for connection...")

while True:

conn,adr=s.accept()

conn.sendall(bytes("connection established","utf-8"))

n=conn.recv(1024)

print(n.decode("utf-8"))

number=int(n)

a= 0

b=1

count = 0

sum=0

if(number<=0):

conn.sendall(bytes("Enter valid number",'utf-8'))

elif(number==1):

conn.sendall(bytes("0","utf-8"))

elif(count <number):

for i in range(number):

no=str(sum)

print("\n")

conn.sendall(bytes(no,'utf-8'))

a = b

b = sum

sum = a+ b

count = count+1

**CLIENT:**

import socket

s=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

s.connect(('127.0.0.1',17000))

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

number=str(n)

s.sendall(bytes(number,"utf-8"))

r=s.recv(1024)

print(r.decode("utf-8"))

for i in range(n):

result=s.recv(1024)

print(result.decode("utf-8"))

s.close()

**UDP:SERVER**

import socket

s=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

host = '127.0.0.1'

port=12345

addr=(host,port)

s.bind(addr)

data, address = s.recvfrom(4096)

dat=int(data)

a=0

b=1

sum=0

count=1

if(dat<=0):

print("enter valid number")

elif(dat==1):

print(a)

else:

while (count <= dat):

print(sum)

count=count+1

a=b

b=sum

sum=a+b

send=s.sendto(bytes(str(sum),'utf-8'),address)

s.close()

**CLIENT:**

import socket

s=socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

host='127.0.0.1'

port=12345

addr=(host,port)

n=input("enter the number")

n=bytes(n,'utf-8')

sent=s.sendto(n,addr)

data,address = s.recvfrom(4096)

s.close()

**10.DNS PROTOCOL;**

**TCP:SERVER:**

import socket

server=socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

server.bind(('127.0.0.1',4734))

server.listen(1)

conn,addr=server.accept()

data=conn.recv(1024)

url=str(data)

url=url.split("'")

url=url[1]

DNS\_DATA={"google.com":"10.0.0.1","kct.ac.in":"10.0.0.2"}

ip\_addr=DNS\_DATA[url]

conn.sendall(bytes(ip\_addr,'utf-8'))

server.close()

**CLIENT:**

import socket

client=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

client.connect(('127.0.0.1',4734))

url=input("enter the url")

url=bytes(url,'utf-8')

client.sendall(url)

ip\_addr=client.recv(1024)

print(str(ip\_addr))

client.close()

**UDP:SERVER:**

import socket

server=socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

server.bind(('127.0.0.1',4734))

data,addr=server.recvfrom(1024)

url=str(data)

url=url.split("'")

url=url[1]

DNS\_DATA={"google.com":"10.0.0.1","kct.ac.in":"10.0.0.2"}

ip\_addr=DNS\_DATA[url]

server.sendto(bytes(ip\_addr,'utf-8'),addr)

server.close()

**CLIENT:**

import socket

client=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

url=input("enter the url")

url=bytes(url,'utf-8')

client.sendto(url,('127.0.0.1',4734))

data,ip\_addr=client.recvfrom(1024)

print(str(ip\_addr))

client.close()

**11.POSSITIVE OR NEGATIVE:**

**TCP:SERVER**

import socket

host='127.0.0.1'

port=12000

s=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

s.bind((host,port))

s.listen()

conn,addr=s.accept()

with conn:

print("connected by",addr)

while True:

data=conn.recv(2048)

if not data:

break

print("\n decoding recieved data")

x=data.decode('ASCII')

y=int(x)

print("\n checking result")

if y>=0:

result="positive no"

conn.send(result.encode('ASCII'))

else:

result="negative no"

conn.send(result.encode('ASCII'))

print("\n result sent to client")

**CLIENT:TCP**

import socket

host='127.0.0.1'

port=12000

s=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

s.connect((host,port))

print("connection estabilished")

n=(input("\n enter no"))

s.send(n.encode('ASCII'))

data=s.recv(2048)

result=data.decode('ASCII')

print(n, "is a" ,result)

s.close()

**UDP:SERVER**

import socket

s=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

host='127.0.0.1'

port=12000

sock\_add=(host,port)

s.bind(sock\_add)

data,add=s.recvfrom(2048)

print("\n Message is being recieved from ", add)

x=data.decode('ASCII')

y=int(x)

print("\n Checking result")

if y>=0:

result="positive no"

s.sendto(result.encode('ASCII'),add)

else:

result="negative no"

s.sendto(result.encode('ASCII'),add)

print("\n Result sent to client")

s.close()

**UDP:CLIENT:**

import socket

c=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

host='127.0.0.1'

port=12000

sock\_add=(host,port)

n=input("\n Enter no")

c.sendto(n.encode('ASCII'),sock\_add)

print("message sent")

data,add=c.recvfrom(2048)

result=data.decode('ASCII')

print(n,"is",result)

**12.REVERSE A STRING**

**TCP:**

UDP:

**13.ARMSTRONG NUMBER:**

**TCP:SERVER**

import socket

host = '127.0.0.1'

port = 14100

s = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

s.bind((host,port))

s.listen()

conn , addr = s.accept()

with conn:

print("Connected by ", addr)

while True:

data = conn.recv(2048)

if not data:

break

print("\nDecoding the received data...")

x = data.decode('ASCII')

y = int(x)

print("\nChecking the result...")

sum = 0

temp = y

while temp > 0:

digit = temp % 10

sum += digit \*\* 3

temp //= 10

if y == sum:

result = "Armstrong Number"

conn.send(result.encode('ASCII'))

else :

result = "Not an Armstrong Number"

conn.send(result.encode('ASCII'))

print("\nResult sent to client...")

**CLIENT:TCP**

import socket

host = '127.0.0.1'

port = 14100

s = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

s.connect((host,port))

print("Connection Established...")

#getting the number from user

n = (input("\n Enter the number to check whether it is armstrong : "))

s.send(n.encode('ASCII'))

data = s.recv(2048)

result = data.decode('ASCII')

print(n , " is an " , result)

s.close()

**14.SUM OF N NUMBERS: 2 PROGRAMS ARE AVAILABLE....**

**TCP:server**

**import socket**

**sock1 = socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)**

**server\_add = (&#39;localhost&#39;,10000)**

**sock1.bind(server\_add)**

**sock1.listen()**

**connection,client\_add = sock1.accept()**

**print(client\_add)**

**data = connection.recv(1024)**

**sum1 = 0**

**n = data**

**m = int(n)**

**while(m&gt;0):**

**sum1 = sum1 + m**

**m -=1**

**connection.sendall(bytes(str(sum1),&#39;utf-8&#39;))**

**connection.close()**

**TCP:client**

**import socket**

**sock0 = socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)**

**server\_add = (&#39;localhost&#39;,10000)**

**sock0.connect(server\_add)**

**n=input(&quot;Enter your input: \n&quot;)**

**sock0.sendall(bytes(n,&#39;utf-8&#39;))**

**data = sock0.recv(1024)**

**print(str(data))**

**sock0.close()**

**UDP:server**

**import socket**

**server=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)**

**#host = &#39;127.0.0.1&#39;**

**port=10000**

**sock\_add=(&#39;localhost&#39;,port)**

**server.bind(sock\_add)**

**data, address = server.recvfrom(1024)**

**print(&quot;From :&quot;,address,&quot;data :&quot;,data)**

**sum1 = 0**

**n = data**

**m = int(n)**

**while(m&gt;0):**

**sum1 = sum1 + m**

**m -=1**

**msg = bytes(str(sum1),&#39;utf-8&#39;)**

**send=server.sendto(msg,address)**

**server.close()**

**UDP:client**

**import socket**

**client=socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)**

**#host=&#39;127.0.0.1&#39;**

**port=10000**

**sock\_add=(&#39;localhost&#39;,port)**

**n=input(&quot;Enter your input : \t&quot;)**

**msg = bytes(n,&#39;utf-8&#39;)**

**sent=client.sendto(msg,sock\_add)**

**data, address = client.recvfrom(1024)**

**print(&quot;From : server&quot;,&quot;data :&quot;,data)**

**client.close()**

**OR**

**TCP:**

PRINT('closing the connection')

**udp:client**

import socket

s=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

server\_address=('127.0.0.1',19000)

n=int(input("Enter a number:"))

n1=str(n)

input\_msg=bytes(n1,'utf-8')

s.sendto(input\_msg,server\_address)

for i in range(n):

no=int(input("Enter a number:"))

a=str(no)

b=bytes(a,'utf-8')

s.sendto(b,server\_address)

data,server=s.recvfrom(1024)

print(data.decode('utf-8'))

**serevr:'**

import socket

s=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

s.bind(('127.0.0.1',19000))

print("Waiting to receive message...")

while True:

sum=0

n,address=s.recvfrom(1024)

print(n.decode('utf-8'))

n1=int(n)

for i in range(n1):

data,address=s.recvfrom(1024)

print(data.decode('utf-8'))

a=int(data)

sum=sum+a

b=str(sum)

c=bytes(b,'utf-8')

s.sendto(c,address)

print("Result given to client")

**15.REVERSE OF A NUMBER:**

**TCP:SERVER**

import socket

sock1=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

serverdet=('127.0.0.1',54321)

sock1.bind(serverdet)

sock1.listen(1)

while True:

conn,clientdet=sock1.accept()

print(clientdet)

data=conn.recv(16)

rev=0

data1=int(data)

while(data1>0):

a=data1%10

rev=rev\*10+a

data1=data1//10

print(rev)

if not data:

print('Data not recieved from ',clientdet)

break

conn.close()

sock1.close()

**CLIENT:**

import socket

sock=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

serverdet=('127.0.0.1',54321)

sock.connect(serverdet)

num=int(input('Enter number: '))

num1=str(num)

sock.sendall(bytes(num1,'utf-8'))

sock.close()

**UDP:SEREVR**

import socket

udp\_port=54321

sock1=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

sock1.bind(('127.0.0.1',udp\_port))

print('waiting for input in client')

while True:

info,clientdet=sock1.recvfrom(1024)

print(clientdet)

data=info

rev=0

data1=int(data)

while(data1>0):

a=data1%10

rev=rev\*10+a

data1=data1//10

print(rev)

if not data:

print('Data not recieved from ',clientdet)

break

sock1.close()

**CLIENT:**

import socket

udp\_port=54321

sock=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

num=int(input('Enter number: '))

num1=str(num)

num2=bytes(num1,'utf-8')

sock.sendto(num2,('127.0.0.1',udp\_port))

sock.close()

**16.STRING COPY:**

**TCP:SERVER**

import socket

server=socket.socket(socket.AF\_INET ,socket.SOCK\_STREAM)

host='127.0.0.1'

port=9999

sock=(host,port)

server.bind(sock)

server.listen(1)

print("Waiting...\n")

con,address=server.accept()

string1=con.recv(1024)

#coping the string from cilent

data=str(string1)

string2=data

print(string1 ,string2)

mgs="The string "+string2+" is received and copied successfully"

data=bytes(mgs,'utf-8')

con.sendall(data)

server.close()

**CLIENT:**

import socket

string1=input("Enter the string \n::->")

data=bytes(string1,'utf-8')

client=socket.socket(socket.AF\_INET ,socket.SOCK\_STREAM)

host='127.0.0.1'

port=9999

sock=(host,port)

client.connect(sock)

client.sendall(data)

print(client.recv(1024))

client.close()

**UDP:SERVER**

import socket

server=socket.socket(socket.AF\_INET ,socket.SOCK\_DGRAM)

sock=('127.0.0.1',54321)

server.bind(sock)

print("Waiting ...\n")

data,addr=server.recvfrom(4096)

#copying the string

string=str(data)

string1=string

print("From ",addr," :",data)

msg="The string "+string1+" is copied "

data=bytes(msg,'utf-8')

server.sendto(data,addr)

server.close()

**CLIENT:**

import socket

string1=input("Enter the string\n::->")

data=bytes(string1,'utf-8')

client=socket.socket(socket.AF\_INET ,socket.SOCK\_DGRAM)

sock=('127.0.0.1',54321)

client.sendto(data,sock)

data,addr=client.recvfrom(4096)

print("FROM SERVER :",str(data))

client.close()

**17.FTP:**

**TCP:SERVER**

import socket

s=socket.socket()

host='127.0.0.1'

port=2048

s.bind((host,port))

s.listen(1)

print(host)

print("waiting for the connection.....")

conn, addr=s.accept()

print(addr,"Has connected to the srerver")

filename = input(str("Enter the filename :"))

file = open(filename, 'rb')

file\_data= file.read(1024)

conn.send(file\_data)

print("Data has been transmitted sucessfully")

**TCPE:CLIENT**

import socket

s=socket.socket()

host='127.0.0.1'

port=2048

s.connect((host,port))

print("connected")

filename = input(str("Enter the filename for upcoming file :"))

file=open(filename, 'wb')

file\_data=s.recv(1024)

file.write(file\_data)

file.close()

print("file has been received sucessfully..")

**UDP:SERVER:**

import socket

s=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

host='127.0.0.1'

port= 50000

s.bind((host , port))

print("waiting for the connection.....")

data,add = s.recvfrom(1024)

print(add,"Has connected to the srerver")

filename = input(str("Enter the filename :"))

file = open(filename, 'rb')

file\_data= file.read(1024)

send=s.sendto(file\_data, add)

print("Data has been transmitted sucessfully")

s.close()

**CLIENT:**

import socket

s=socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

host='127.0.0.1'

port=50000

addr=(host,port)

print("connected")

filename = input(str("Enter the filename for upcoming file :"))

file=open(filename, 'wb')

file\_data=s.recvfrom(1024)

file.write(file\_data)

file.close()

print("file has been received sucessfully..")

s.close()

**18.ARP:**

**TCP:SERVER**

import socket

import getmac

from getmac import get\_mac\_address as gma

host = '127.0.0.1'

port = 14105

s = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

s.bind((host,port))

s.listen()

conn , addr = s.accept()

with conn:

print("Connected by ", addr)

while True:

data = conn.recv(2048)

if not data:

break

print("\nDecoding the received data...")

x = data.decode('ASCII')

print("mac Address : " ,x )

ip=socket.gethostbyname(data)

print("Ip address of the given mac address is:",ip)

conn.send(ip.encode('ASCII'))

print("IP address is sent to client...")

**CLIENT:**

import socket

host = '127.0.0.1'

port = 14100

s = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

s.connect((host,port))

print("Connection Established...")

hostname=socket.gethostname()

ipaddress = socket.gethostbyname(hostname)

s.send(ipaddress.encode('ASCII'))

print("Data sent to server...")

data = s.recv(2048)

result = data.decode('ASCII')

print("MAC address is : " , result)

**UDP:**

**SERVER:'**

import socket

import getmac

from getmac import get\_mac\_address as gma

host = '127.0.0.1'

port = 14102

s = socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

s.bind((host,port))

while True:

data =s.recvfrom(2048)

if not data:

break

print("\nDecoding the received data...")

x = data[0].decode('ASCII')

print("IP Address : " ,x )

mac = gma()

s.sendto(mac.encode('ASCII'),data[1])

print("MAC address is sent to client...")

**CLIENT:**

import socket

host = '127.0.0.1'

port = 14102

s = socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

hostname=socket.gethostname()

ipaddress = socket.gethostbyname(hostname)

s.sendto(ipaddress.encode('ASCII'),(host,port))

print("Data sent to server...")

data = s.recvfrom(2048)

result = data[0].decode('ASCII')

print("MAC address is : " , result)

**19.HTTP:**

**TCP:SERVER**

import socket

host = '127.0.0.1'

port = 14100

server = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

server.bind((host,port))

server.listen()

conn,addr=server.accept()

data=conn.recv(1024)

web\_name=data.decode('ASCII')

web\_link={"google":"www.google.com","facebook":"www.facebook.com","youtube":"www.youtube.com"}

a=web\_link[web\_name]

conn.sendall(a.encode('ASCII'))

print("Web link send to client...")

server.close()

**CLIENT:**

import socket

host = '127.0.0.1'

port = 14100

c = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

c.connect((host,port))

print("Connection Established...")

a=input("enter the website name : ")

c.send(a.encode('ASCII'))

web=c.recv(1024)

print("Website link is : " , web.decode('ASCII'))

c.close()

**UDP:**

**SERVER**:

import socket

import sys

sock = socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

host='127.0.0.1'

port=10045

server\_address = (host,port)

sock.bind(server\_address)

a1,addr = sock.recvfrom(4096)

web\_name=a1.decode("utf-8")

web\_link={"google":"[www.google.com](http://www.google.com)","facebook":"[www.facebook.com](http://www.facebook.com)","youtube":"[www.youtube.com](http://www.youtube.com)"}

a=web\_link[web\_name]

sent = sock.sendto(bytes(a,"utf=8"), addr)

print("Web link send to client...")

sock.close()

**CLIENT:**

import socket

import sys

sock = socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

server\_address = ('127.0.0.1',10045)

a=input("enter the website name : ")

a1=bytes(a,"utf-8")

sent = sock.sendto(a1,server\_address)

a,server = sock.recvfrom(4096)

print("Website link is : " , a.decode('utf-8'))

sock.close()

**20.SLIDING WINDOW**

**TCP:SERVER:**

import socket

host = '127.0.0.1'

port = 14100

s = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

s.bind((host,port))

s.listen()

conn , addr = s.accept()

with conn:

print("Connected by ", addr)

while True:

data = conn.recv(2048)

if not data:

break

print("\nDecoding the received data...")

x = data.decode('ASCII')

print("Data Received is : " ,x )

conn.send("ACK".encode('ASCII'))

print("ACK send to client...")

**CLIENT:**

import socket

host = '127.0.0.1'

port = 14100

s = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

s.connect((host,port))

print("Connection Established...")

window\_size = int(input("Enter the size of buffer window " ))

n=1

while(n <= window\_size):

print("Enter the data ",n," : " )

data = input()

s.send(data.encode('ASCII'))

print("Data sent to server...")

data = s.recv(2048)

result = data.decode('ASCII')

print("From Server : " , result)

n = n+1

print("Window size Exceeded...")

s.close()

**UDP:**

**SERVER:**

import socket

import sys

sock = socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

host='127.0.0.1'

port=10045

server\_address = (host,port)

sock.bind(server\_address)

while True:

data,addr = sock.recvfrom(4096)

if not data:

break

print("\nDecoding the received data...")

x = data.decode('ASCII')

print("Data Received is : " ,x )

sent = sock.sendto("ACK".encode('ASCII'), addr)

print("ACK send to client...")

**CLIENT:**

import socket

import sys

sock = socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

server\_address = ('127.0.0.1',10045)

print("Connection Established...")

window\_size = int(input("Enter the size of buffer window " ))

n=1

while(n <= window\_size):

print("Enter the data ",n," : " )

data = input()

sent = sock.sendto(data.encode('ASCII'),server\_address)

print("Data sent to server...")

data,server = sock.recvfrom(4096)

result=data.decode('ASCII')

print("From Server : " , result)

n = n+1

print("Window size Exceeded...")

sock.close()

**21.RARP:**

**TCP:SERVER**

import socket

import threading

# Connection Data

host = '127.0.0.1'

port = 55555

# Starting Server

server = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

server.bind((host, port))

server.listen()

# Lists For Clients and Their Nicknames

clients = []

nicknames = []

# Sending Messages To All Connected Clients

def broadcast(message):

for client in clients:

client.send(message)

# Handling Messages From Clients

def handle(client):

while True:

try:

# Broadcasting Messages

message = client.recv(1024)

broadcast(message)

except:

# Removing And Closing Clients

index = clients.index(client)

clients.remove(client)

client.close()

name = nicknames[index]

broadcast('{} left!'.format(name).encode('ascii'))

nicknames.remove(name)

break

hostname=socket.gethostname()

ipaddress = socket.gethostbyname(hostname)

print("Hostname:",{hostname})

print("IP Address:",{ipaddress})

# Receiving / Listening Function

def receive():

while True:

# Accept Connection

client, address = server.accept()

print("Connected with {}".format(str(address)))

# Request And Store Nickname

client.send('NICK'.encode('ascii'))

nickname = client.recv(1024).decode('ascii')

nicknames.append(nickname)

clients.append(client)

# Print And Broadcast Nickname

print("User’s name is {}".format(nickname))

broadcast("{} joined!".format(nickname).encode('ascii'))

client.send('Connected to server!'.encode('ascii'))

# Start Handling Thread For Client

thread = threading.Thread(target=handle, args=(client,))

thread.start()

receive()

**CLIENT:**

import socket

import threading

# Choosing Nickname

name = input("Enter User’s name: ")

# Connecting To Server

client = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

client.connect(('127.0.0.1', 55555))

# Listening to Server and Sending Nickname

def receive():

while True:

try:

# Receive Message From Server

# If 'NICK' Send Nickname

message = client.recv(1024).decode('ascii')

if message == 'NICK':

client.send(name.encode('ascii'))

else:

print(message)

except:

# Close Connection When Error

print("An error occured!")

client.close()

break

# Sending Messages To Server

def write():

while True:

message = '{}: {}'.format(name, input(''))

client.send(message.encode('ascii'))

# Starting Threads For Listening And Writing

receive\_thread = threading.Thread(target=receive)

receive\_thread.start()

write\_thread = threading.Thread(target=write)

write\_thread.start()

**UDP:SERVER;**

import socket

sock=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

sock.bind(('127.0.0.1',55555))

while True:

data,addr=sock.recvfrom(4096)

print(str(data))

message="This is your IP address"

sock.sendto(str.encode(message),addr)

hostname=socket.gethostname()

ipaddress = socket.gethostbyname(hostname)

print("Hostname:",{hostname})

print("IP Address:",{ipaddress})

**CLIENT:**

import socket

client\_socket=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

msg=("hello please find my IP address")

client\_socket.sendto(str.encode(msg),('127.0.0.1',55555))

data,addr=client\_socket.recvfrom(4096)

print("Server Says")

print(str(data))

client\_socket.close()

**22.PRIME NUMBERS**

**TCP:SERVER**

import socket

sock1 = socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

server\_add = (&#39;localhost&#39;,10000)

sock1.bind(server\_add)

sock1.listen()

connection,client\_add = sock1.accept()

print(client\_add)

data = connection.recv(1024)

msg = &quot; is neither prime nor composite number&quot;

msg1 = &quot; is not a prime number&quot;

msg2 = &quot; is a prime number&quot;

sum1 = 0

n = data

m = int(n)

if (m==0):

connection.send(msg.encode(&#39;ASCII&#39;))

elif (m==1):

connection.send(msg1.encode(&#39;ASCII&#39;))

elif (m==2):

connection.send(msg2.encode(&#39;ASCII&#39;))

elif (m&gt;1):

for i in range(2,m):

if (m%i) == 0:

connection.send(msg1.encode(&#39;ASCII&#39;))

break

else:

connection.send(msg2.encode(&#39;ASCII&#39;))

break

connection.close()

**TCP:client**

import socket

sock0 = socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

server\_add = (&#39;localhost&#39;,10000)

sock0.connect(server\_add)

n=input(&quot;Enter your input: \n&quot;)

sock0.sendall(bytes(n,&#39;utf-8&#39;))

data = sock0.recv(1024)

print(str(data))

sock0.close()

UDP:

**23.DOWNLOAD A FILE**

**TCP:client**

import socket

client=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

client.connect((&#39;127.0.0.1&#39;,5774))

fn=input(&quot;enter the file name&quot;)

fn1=fn

fn=bytes(fn,&#39;utf-8&#39;)

client.sendall(fn)

file=client.recv(7024)

print(str(file))

fp=open(fn1+&quot; server.txt&quot;,&#39;w&#39;)

file=str(file)

file=file.split(&quot;&#39;&quot;)

file=file[1]

fp.write(file)

fp.close()

client.close()

**TCP-server**

import socket

server=socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

server.bind((&#39;127.0.0.1&#39;,5774))

server.listen(1)

conn,addr=server.accept()

data1=conn.recv(1024)

data1=str(data1)

data1=data1.split(&quot;&#39;&quot;)

data1=data1[1]

fp=open(data1,&#39;r&#39;)

data=fp.read()

conn.sendall(bytes(data,&#39;utf-8&#39;))

fp.close()

server.close()

**UDP-client**

import socket

client=socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

sock\_addr=((&#39;127.0.0.1&#39;,7774))

fn=input(&quot;enter the file name&quot;)

fn1=fn

fn=bytes(fn,&#39;utf-8&#39;)

client.sendto(fn,sock\_addr)

file,addr=client.recvfrom(4096)

print(str(file))

fp=open(fn1,&#39;w&#39;)

file=str(file)

file=file.split(&quot;&#39;&quot;)

file=file[1]

fp.write(file)

fp.close()

client.close()

**UDP-server**

import socket

server=socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

server.bind((&#39;127.0.0.1&#39;,7774))

data1,addr=server.recvfrom(1024)

data1=str(data1)

data1=data1.split(&quot;&#39;&quot;)

data1=data1[1]

fp=open(data1,&#39;r&#39;)

data=fp.read()

server.sendto(bytes(data,&#39;utf-8&#39;),addr)

print(addr)

fp.close()

server.close()