**TCP and UDP input code:**

import sys

import socket

#reading commandline arguments

option = sys.argv[1]

ip = sys.argv[2]

port = sys.argv[3]

prot = sys.argv[4]

#Listener part to listen on the user supplied host, port and protocol

#MESSAGE\_1 = "Hello Programmer!"

#if option == "-l":

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

#s.bind((ip, int(port)))

#s.listen(2)

#while True:

#c, addr = s.accept()

#print c.recv(1024)

#print "message:", MESSAGE

#Talker part to emit the user supplied host, port and protocol

#MESSAGE = "Hello Programmer!"

#if option == "-s":

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

#c.connect((ip, int(port)))

#c.send("ip = " + ip)

#c.send(" port = " + port)

#c.send(" protocol = " + prot)

#print "message:", MESSAGE

def TCP\_listener(ip, port):

#if option == "-l":

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

s.bind((ip, int(port)))

s.listen(2)

while True:

connection, address = s.accept()

while True:

c, addr = s.accept()

print c.recv(1024)

s.close()

def TCP\_talker(ip, port):

#if option == "-s":

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

c.connect((ip, int(port)))

c.send("ip = " + ip)

c.send(" port = " + port)

c.send(" protocol = " + prot)

for line in sys.stdin:

c.send(str(line))

c.close()

def UDP\_listener(ip, port):

ip = "127.0.0.1"

port = 5005

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

sock.bind((ip, port))

while True:

data, addr = sock.recvfrom(1024)

print "received message:", data

def UDP\_talker(ip, port):

ip = "127.0.0.1"

port = 5005

MESSAGE = "Hello Programmer!"

print "UDP target IP:", ip

print "UDP target port:", port

print "message:", MESSAGE

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

sock.sendto(MESSAGE, (ip, port))

if(option == "-l" or option == "-L" and prot == "tcp" or prot == "TCP"):

TCP\_listener(ip, port)

elif(option == "-s" or option == "-S" and prot == "tcp" or prot == "TCP"):

TCP\_talker(ip, port)

**OUTPUT:**

**A screenshot of a computer

Description automatically generated**