Concurrent Time Server application using UDP

Aim

To implement Concurrent Time Server application using UDP to execute the pro-gram at remote server. Client sends a time request to the server, server sends its system time back to the client. Client displays the result.

Theory

UDP (User Datagram Protocol) is primarily for establishing low-latency and loss-tolerating connections between applications on the internet. **UDP** sends messages, called datagrams, and is considered a best-effort mode of communications. Itis considered a connectionless protocol because it doesn’t require a virtual circuit tobe established before any data transfer occurs.

Server- The server here waits for the client’s time request. When a request isreceived, the present system time of the server is sent to the client.

Client- The client sends the server a time request. The response from the serveris received and provided as the output

Code

udptimeerver.py

```
from socket import *
from os import fork
import sys
import datetime
myHost = ""
myPort = 5000
s=socket(AF_INET, SOCK_DGRAM)
s.bind((myHost, myPort))
while True:
 data, address = s.recvfrom(1024)
 print "UDP server:", data, "from", address
 if data:
  print "Request received"
  print "Processing is done "
  s.sendto("message is " + data + "\n Time recieved from server is:" +
str(datetime.datetime.now()), address)
 else:
```

```
  break
```

udptimeclient.py

```
import sys, time
from socket import *
serverHost = "localhost"
serverPort = 5000
s=socket(AF_INET, SOCK_DGRAM)
s.connect((serverHost, serverPort))
s.send("Hello world")
data = s.recv(1024)
print data
```

Output