EX : 10 - Ping to test server connectivity using sockets

Date:17.10.2024

Installing Python Ping

pip install pythonping

in windows python get-pythonping.py [in run command prompt]

Python Ping (`pythonping`) is a public repository you can find on PyPI.

from pythonping import ping

ping('8.8.8.8')

simply ping Google. you won't see anything in your console if you just run this script. This is because our ping is **silent by default**, and does not print anything to screen.

If we want to see everything on-screen, we can simply use the `verbose` flag.

ping('8.8.8.8', verbose=**True**)

Ping to test server connectivity

How to ping a website in python

```
from os import system
print('1. Ping Google')
print('2. Ping Yahoo')
print('3. Ping custom URL')
while True:
    key = int(input('Input your choice: '))
    if key == 1:
        system("ping www.google.com")
    elif key == 2:
        system("ping www.yahoo.com")
    elif key == 3:
        url = input('Enter URL: ')
        system("ping " + url)
    else:
        print("Invalid Option!")
```

```
import os
os.system("ping google.com")
```

```
import os
os.system('ping 127.0.0.1')
```

PING TO TEST SERVER CONNECTIVITY USING SOCKETS

AIM:

To develop ping program to test server connectivity using sockets.

ALGORITHM:

Server.py

1. Import the socket package
2. Initialize local IP address and local port.
3. Create a socket using socket() function
4. Bind the IP address and port number.
5. Accept client request for connection.
6. Print the received connection details
7. Send reply message to the client.
8. Close the connection.

Client.py

1. Import the socket package
2. Initialize server IP address and local port.
3. Create a socket using socket() function.
4. Start the timer.
5. Send message to the server.
6. The reply message of the server is received.
7. The timer is stopped.
8. Print the round trip time statistics.

Ping to test server connectivity using sockets

Client code:

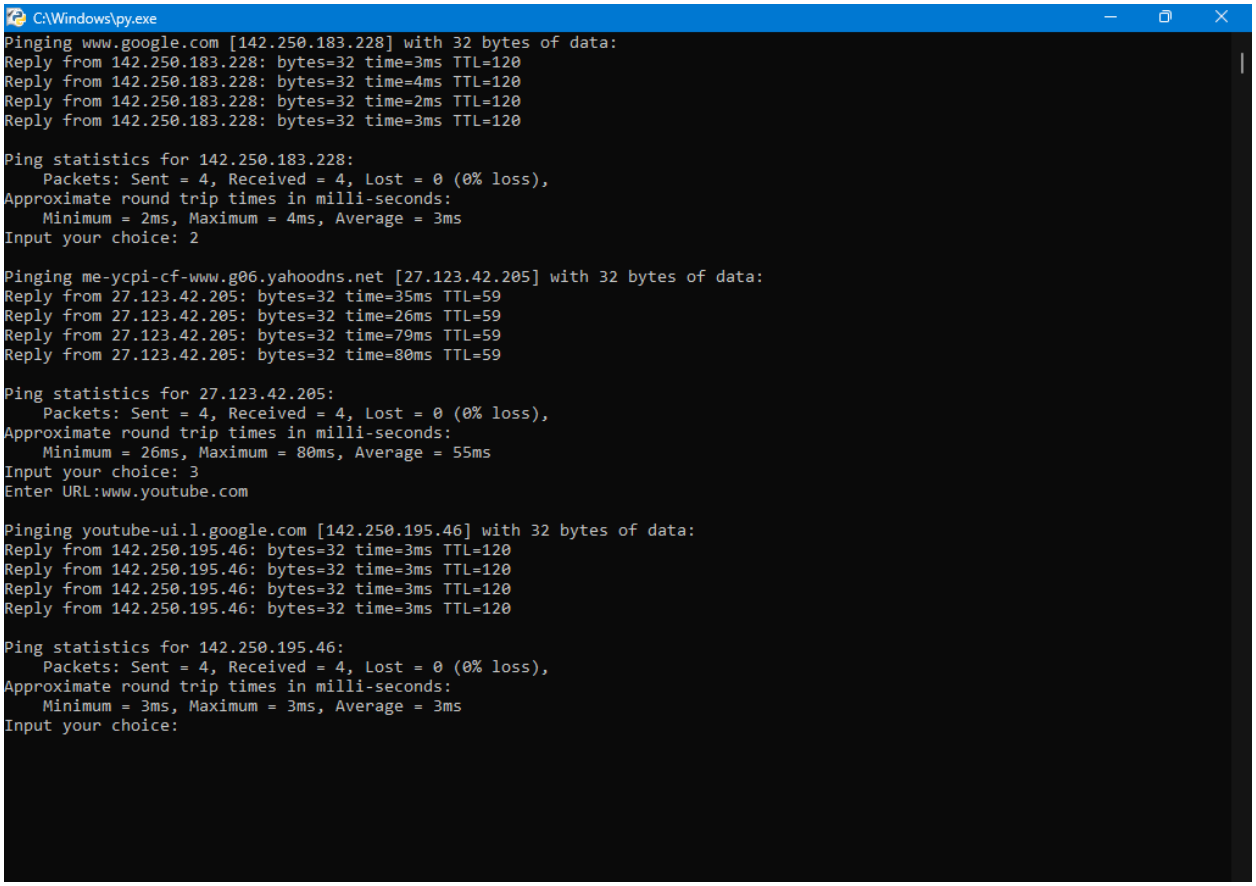
```
from socket import *
from os import system
```

```
s = socket(AF_INET, SOCK_STREAM)
s.connect(("127.0.0.1",8000)) # Connect
op='connect'
s.send(op.encode('utf-8')) # Send request
data = s.recv(100).decode()# Get response
print(data)
system("ping "+ gethostname())
s.close()
```

#Server Code:

```
from socket import *
from os import system
s = socket(AF_INET,SOCK_STREAM)
s.bind(("",8000))
s.listen(5)
while True:
        c,a = s.accept()
        print("Received connection from", a)
        data=c.recv(100).decode()
        print(data)
        c.send(data.encode('utf-8'))
        system("ping "+ a)


c.close()
```


C:\Windows\py.exe
Pinging www.google.com [142.250.183.228] with 32 bytes of data:
Reply from 142.250.183.228: bytes=32 time=3ms TTL=120
Reply from 142.250.183.228: bytes=32 time=4ms TTL=120
Reply from 142.250.183.228: bytes=32 time=2ms TTL=120
Reply from 142.250.183.228: bytes=32 time=3ms TTL=120
Ping statistics for 142.250.183.228:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 2ms, Maximum = 4ms, Average = 3ms
Input your choice: 2
Pinging me-ycpi-cf-www.g06.yahoodns.net [27.123.42.205] with 32 bytes of data:
Reply from 27.123.42.205: bytes=32 time=35ms TTL=59
Reply from 27.123.42.205: bytes=32 time=26ms TTL=59
Reply from 27.123.42.205: bytes=32 time=79ms TTL=59
Reply from 27.123.42.205: bytes=32 time=80ms TTL=59
Ping statistics for 27.123.42.205:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 26ms, Maximum = 80ms, Average = 55ms
Input your choice: 3
Enter URL:www.youtube.com
Pinging youtube-ui.l.google.com [142.250.195.46] with 32 bytes of data:
Reply from 142.250.195.46: bytes=32 time=3ms TTL=120
Reply from 142.250.195.46: bytes=32 time=3ms TTL=120
Reply from 142.250.195.46: bytes=32 time=3ms TTL=120
Reply from 142.250.195.46: bytes=32 time=3ms TTL=120
Ping statistics for 142.250.195.46:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 3ms, Maximum = 3ms, Average = 3ms
Input your choice: