

**Name:**Komal Chitnis

**Moodle Id :** 20102068

**Roll Number:** 26

**CN LAB**

**EXPERIMENT NUMBER: 8**

**Aim:** Creation of a simple Socket for basic information exchange between Server and Client.

**Code: server.py**

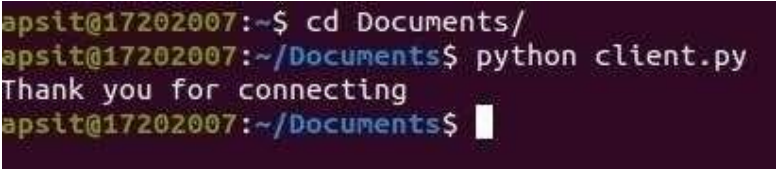
```
# first of all import the socket library
import socket
# next create a socket object s = socket.socket()
print "Socket successfully created" # reserve a port
on your computer in our # case it is 12345 but it
can be anything
port = 1234
# Next bind to the port
# we have not typed any ip in the ip field
# instead we have inputted an empty string# this makes
the server listen to requests #
coming from other computers on the networks.bind(('',
port))
print "socket binded to %s" %(port)# put the
socket into listening mode s.listen(5) print
"socket is listening"
# a forever loop until we interrupt it or# an error
occurs
while True:
# Establish connection with client.c, addr
    s.accept()
    print 'Got connection from', addr
# send a thank you message to the client.c.send('Thank you for
    connecting')
# Close the connection with the client.c.close()
```

A terminal window with a dark background and light-colored text. The prompt is 'apsit@17202007:~\$'. The user enters 'cd Documents/' and the prompt changes to 'apsit@17202007:~/Documents\$'. Then the user enters 'python server.py'. The output of the script is displayed: 'Socket successfully created', 'socket binded to 1234', 'socket is listening', and 'Got connection from ('127.0.0.1', 38682)'. A cursor is visible at the end of the last line of output.

```
apsit@17202007:~$ cd Documents/
apsit@17202007:~/Documents$ python server.py
Socket successfully created
socket binded to 1234
socket is listening
Got connection from ('127.0.0.1', 38682)
```

### client.py:

```
#Import socket
module import socket
#Create a socket
object
s=socket.socket()
#Define the port on which you want to
connect port=1234
s.connect(('127.0.0.1',port)
) print s.recv(1024)
s.close()
```

A terminal window with a dark purple background. The prompt is 'apsit@17202007:~\$'. The user enters 'cd Documents/' and the prompt changes to 'apsit@17202007:~/Documents\$'. The user then enters 'python client.py'. The output of the program is 'Thank you for connecting'. The prompt returns to 'apsit@17202007:~/Documents\$' with a white cursor.

```
apsit@17202007:~$ cd Documents/
apsit@17202007:~/Documents$ python client.py
Thank you for connecting
apsit@17202007:~/Documents$
```