Experiment No. 13

Aim: Develop a chatting application using networking concepts.

Problem Statement:

1. Write java socket programming for Client and Server communication.
2. Develop a chatting application using networking concepts.
3. Develop an application to send and receive DatagramPacket by using DatagramSocket.

Theory:

Java Socket programming-

- It is used for communication between the applications running on different JRE.
- Can be connection-oriented or connection-less.
- Socket used for Client and server programming
- Client should know two information:
    - IP Address of Server
    - Port number.
- Socket programming is connection-oriented
    - Socket class
    - ServerSocket class
- The java.net package provides a class, Socket
    - java.net.Socket class

Socket class

- A socket is simply an endpoint for communications between the machines.
- The Socket class can be used to create a socket.
- Important methods-
    - public InputStream getInputStream()
    - public OutputStream getOutputStream()
    - public void close()

ServerSocket class

- The ServerSocket class can be used to create a server socket.
- This object is used to establish communication with the clients.
- Important methods
    - public Socket accept()
    - public void close()

Steps for Client Side Programming

- Establish a Socket Connection
    - To open a socket:
    - Socket socket = new Socket("127.0.0.1", 6000)
- Communication
    - getOutputStream() to send output to the server socket
    - getInputStream()

- Closing the data stream and connection
    - input.close();
    - out.close();
    - socket.close()

Steps for Server Programming

- Establish a Socket Connection
    - ServerSocket which waits for the client requests.
    - A plain Socket to use for communication with the client (after accepting connection)
    - ServerSocket ss=new ServerSocket(6666);
    - Socket s=ss.accept();//bind connection and waits for client
- Communication
    - getOutputStream() method to send the output through the socket.
    - getInputStream() method to receive the input through the socket.
- Close the i/o stream and Connection
    - Close() method.

Conclusion:

Students successfully studied and developed chatting application for client and server.