**Prepare a Chat Application using Client-Server.**

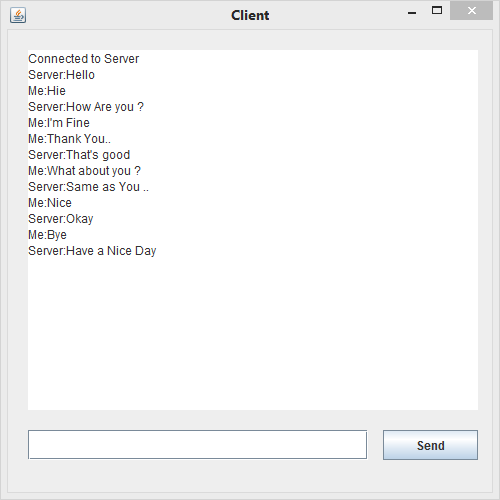
**serverChatform.java**

1. package chatapp;
2. import java.awt.event.\*;
3. import java.io.\*;
4. import java.net.\*;
5. import javax.swing.\*;
6. public class serverChatform extends JFrame implements ActionListener
7. {
8. static ServerSocket server;
9. static Socket conn;
10. JPanel panel;
11. JTextField NewMsg;
12. JTextArea ChatHistory;
13. JButton Send;
14. DataInputStream dis;
15. DataOutputStream dos;
16. public serverChatform() throws UnknownHostException, IOException
17. {
18. panel = new JPanel();
19. NewMsg = new JTextField();
20. ChatHistory = new JTextArea();
21. Send = new JButton("Send");
22. this.setSize(500, 500);
23. this.setVisible(true);
24. panel.setLayout(null);
25. this.add(panel);
26. ChatHistory.setBounds(20, 20, 450, 360);
27. panel.add(ChatHistory);
28. NewMsg.setBounds(20, 400, 340, 30);
29. panel.add(NewMsg);
30. Send.setBounds(375, 400, 95, 30);
31. panel.add(Send);
32. this.setTitle("Server");
33. Send.addActionListener(this);
34. server = new ServerSocket(888);
35. ChatHistory.setText("Waiting for Client");
36. conn = server.accept();
37. ChatHistory.setText(ChatHistory.getText() + '\n' + "Client Found");
38. while (true)
39. {
40. try
41. {
42. DataInputStream dis = new DataInputStream(conn.getInputStream());
43. String string = dis.readUTF();
44. ChatHistory.setText(ChatHistory.getText() + '\n' + "Client:"+ string);
45. }
46. catch (Exception e1)
47. {
48. ChatHistory.setText(ChatHistory.getText() + '\n'+ "Message sending fail:Network Error");
49. }
50. }
51. }
52. @Override
53. public void actionPerformed(ActionEvent e)
54. {
55. if ((e.getSource() == Send) && (NewMsg.getText() != ""))
56. {
57. ChatHistory.setText(ChatHistory.getText() + '\n' + "ME:" + NewMsg.getText());
58. try
59. {
60. DataOutputStream dos = new DataOutputStream(
61. conn.getOutputStream());
62. dos.writeUTF(NewMsg.getText());
63. }
64. catch (Exception e1)
65. {
66. }
67. NewMsg.setText("");
68. }
69. }
70. public static void main(String[] args) throws UnknownHostException,IOException
71. {
72. new serverChatform();
73. }
74. }

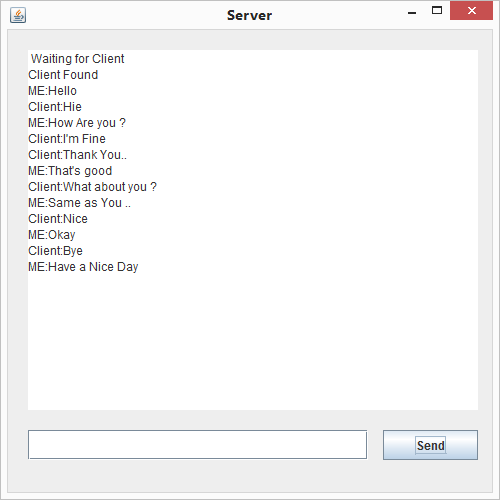
**ClientChatForm.java**

1. package chatapp;
2. import java.awt.event.\*;
3. import java.io.\*;
4. import java.net.\*;
5. import javax.swing.\*;
6. public class ClientChatForm extends JFrame implements ActionListener
7. {
8. static Socket conn;
9. JPanel panel;
10. JTextField NewMsg;
11. JTextArea ChatHistory;
12. JButton Send;
13. public ClientChatForm() throws UnknownHostException, IOException
14. {
15. panel = new JPanel();
16. NewMsg = new JTextField();
17. ChatHistory = new JTextArea();
18. Send = new JButton("Send");
19. this.setSize(500, 500);
20. this.setVisible(true);
21. panel.setLayout(null);
22. this.add(panel);
23. ChatHistory.setBounds(20, 20, 450, 360);
24. panel.add(ChatHistory);
25. NewMsg.setBounds(20, 400, 340, 30);
26. panel.add(NewMsg);
27. Send.setBounds(375, 400, 95, 30);
28. panel.add(Send);
29. Send.addActionListener(this);
30. conn = new Socket("localhost", 888);
31. ChatHistory.setText("Connected to Server");
32. this.setTitle("Client");
33. while (true)
34. {
35. try
36. {
37. DataInputStream dis = new DataInputStream(conn.getInputStream());
38. String string = dis.readUTF();
39. ChatHistory.setText(ChatHistory.getText() + '\n' + "Server:" + string);
40. }
41. catch (Exception e1)
42. {
43. ChatHistory.setText(ChatHistory.getText() + '\n' + "Message sending fail:Network Error");
44. }
45. }
46. }
47. @Override
48. public void actionPerformed(ActionEvent e)
49. {
50. if ((e.getSource() == Send) && (NewMsg.getText() != ""))
51. {
52. ChatHistory.setText(ChatHistory.getText() + '\n' + "Me:" + NewMsg.getText());
53. try
54. {
55. DataOutputStream dos = new DataOutputStream(
56. conn.getOutputStream());
57. dos.writeUTF(NewMsg.getText());
58. }
59. catch (Exception e1)
60. {
61. ChatHistory.setText(ChatHistory.getText() + '\n' + "Message sending fail:Network Error");
62. }
63. NewMsg.setText("");
64. }
65. }
66. public static void main(String[] args) throws UnknownHostException, IOException
67. {
68. ClientChatForm chatForm = new ClientChatForm();
69. }
70. }

**OUTPUT**

****

**Client**

****

**Server**