To implemnet Socket Programming.

GreetingServer.java

```
import java.net.*;
import java.io.*;

public class GreetingServer extends Thread {

  private ServerSocket serverSocket;

  public GreetingServer(int port) throws IOException {
    serverSocket = new ServerSocket(port);
    //amount of time the server waits for any connection
    serverSocket.setSoTimeout(10000);
  }

  public void run() {
    while (true) {
      try {
        System.out.println("Waiting for client on port "
                + serverSocket.getLocalPort() + "...");
        Socket server = serverSocket.accept();
        System.out.println("Just connected to "
                + server.getRemoteSocketAddress());
        //read in a string from the client
        DataInputStream in
                = new DataInputStream(server.getInputStream());
        System.out.println(in.readUTF());
        //write out a string to the client
        DataOutputStream out
                = new DataOutputStream(server.getOutputStream());
        out.writeUTF("Thank you for connecting to "
                + server.getLocalSocketAddress() + "\nGoodbye!");
        server.close();
      } catch (SocketTimeoutException s) {
        System.out.println("Socket timed out!");
        break;
      } catch (IOException e) {
        e.printStackTrace();
        break;
      }
    }
  }
  public static void main(String[] args) {
    //assign command line arguments to port variables,
    //should be a valid port number and be available
    int port = Integer.parseInt(args[0]);
    try {
      Thread t = new GreetingServer(port);
      t.start();
    } catch (IOException e) {
      e.printStackTrace();
    }
  }
}
```

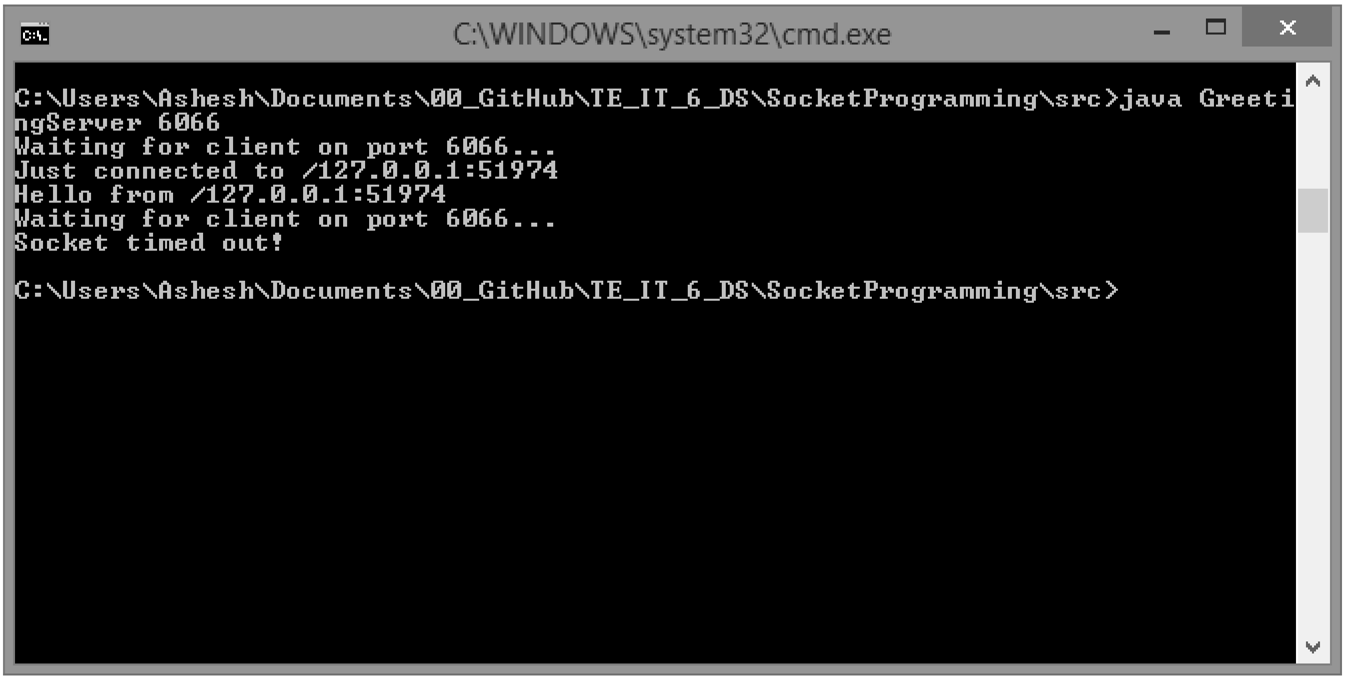
GreetingClient.java

```
import java.net.*;
import java.io.*;

public class GreetingClient {

  public static void main(String[] args) {
    //assign command line arguments to variables
    String serverName = args[0];
    int port = Integer.parseInt(args[1]);
    try {
      System.out.println("Connecting to " + serverName
              + " on port " + port);
      Socket client = new Socket(serverName, port);
      System.out.println("Just connected to "
              + client.getRemoteSocketAddress());
      //write out a string to the server
      OutputStream outToServer = client.getOutputStream();
      DataOutputStream out
              = new DataOutputStream(outToServer);
      out.writeUTF("Hello from "
              + client.getLocalSocketAddress());
      //read in a string from the server
      InputStream inFromServer = client.getInputStream();
      DataInputStream in
              = new DataInputStream(inFromServer);
      System.out.println("Server says " + in.readUTF());
      //closes the socket,
      //which makes this Socket object no longer capable of connecting
      client.close();
    } catch (IOException e) {
      e.printStackTrace();
    }
  }
}
```

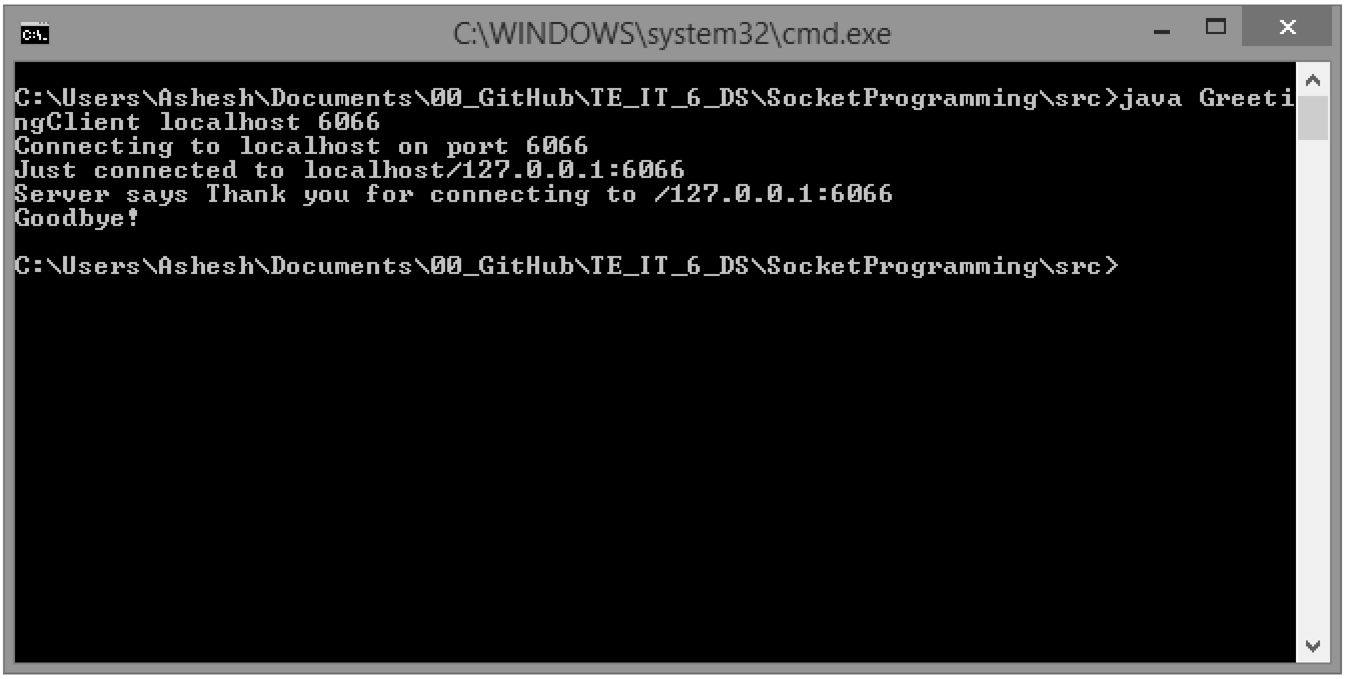
Server Side



```
C:\Users\Ashesh\Documents\00_GitHub\TE_IT_6_DS\SocketProgramming\src>java Greeti
ngServer 6066
Waiting for client on port 6066...
Just connected to /127.0.0.1:51974
Hello from /127.0.0.1:51974
Waiting for client on port 6066...
Socket timed out!

C:\Users\Ashesh\Documents\00_GitHub\TE_IT_6_DS\SocketProgramming\src>
```

Client Side



```
C:\Users\Ashesh\Documents\00_GitHub\TE_IT_6_DS\SocketProgramming\src>java Greeti
ngClient localhost 6066
Connecting to localhost on port 6066
Just connected to localhost/127.0.0.1:6066
Server says Thank you for connecting to /127.0.0.1:6066
Goodbye!

C:\Users\Ashesh\Documents\00_GitHub\TE_IT_6_DS\SocketProgramming\src>
```