SocketSever

|  |
| --- |
| import java.io.\*; |
|  | import java.net.\*; |
|  | import java.util.ArrayList; |
|  |  |
|  | // Server class |
|  | class Server { |
|  |  |
|  | ServerSocket server = null; |
|  | ArrayList<User> userList = new ArrayList<User>(100); |
|  |  |
|  | public void listenSocket(int port) |
|  | { |
|  |  |
|  | String hostname = "unknown"; |
|  |  |
|  | try { |
|  |  |
|  | InetAddress address; |
|  | address = InetAddress.getLocalHost(); |
|  | hostname = address.getHostName(); |
|  |  |
|  | } catch(UnknownHostException ex) { |
|  |  |
|  | System.out.println("Hostname can not be resolved"); |
|  | ex.printStackTrace(); |
|  |  |
|  | } |
|  |  |
|  | try |
|  | { |
|  | server = new ServerSocket(port); |
|  | System.out.println("Server is running on " + hostname + " : " + port + |
|  | "," + " use ctrl-C to end"); |
|  | } |
|  | catch (IOException e) |
|  | { |
|  | System.out.println("Error creating socket"); |
|  | System.exit(-1); |
|  | } |
|  |  |
|  | while(true) |
|  | { |
|  | ClientWorker w; |
|  | try |
|  | { |
|  | // accept() method returns a Socket object. |
|  | w = new ClientWorker(server.accept(), userList); |
|  | Thread t = new Thread(w); |
|  | t.start(); |
|  | } |
|  | catch (IOException e) |
|  | { |
|  | System.out.println("Accept failed"); |
|  | System.exit(-1); |
|  | } |
|  | } |
|  | } |
|  |  |
|  | protected void finalize() |
|  | { |
|  | try |
|  | { |
|  | server.close(); |
|  | } |
|  | catch (IOException e) |
|  | { |
|  | System.out.println("Could not close socket"); |
|  | System.exit(-1); |
|  | } |
|  | } |
|  |  |
|  | public static void main(String[] args) |
|  | { |
|  | if (args.length != 1) |
|  | { |
|  | System.out.println("Usage: java Server port"); |
|  | System.exit(1); |
|  | } |
|  |  |
|  | Server server = new Server(); |
|  | int port = Integer.valueOf(args[0]); |
|  | server.listenSocket(port); |
|  |  |
|  | } |
|  |  |
|  | } |