



Project Explorer

- > Clock
- > clock.java
- > ClockApplication
- > ecommerce
- > LibraryCatalogProject
- > OnlineChatApp
 - > JRE System Library [JavaSE-23]
 - > src
 - > client
 - > ChatClient.java
 - > package-info.java
 - > server
 - > ChatServer.java
 - > package-info.java
 - > module-info.java

```
1 package server;
2
3 import java.io.*;
4 import java.net.*;
5 import java.util.*;
6
7 public class ChatServer {
8     private static Set<ClientHandler> clientHandlers = new HashSet<>();
9     private static int userId = 0;
10
11     public static void main(String[] args) throws IOException {
12         ServerSocket serverSocket = new ServerSocket(1234);
13         System.out.println("Server is running...");
14
15         while (true) {
16             Socket clientSocket = serverSocket.accept();
17             userId++;
18             ClientHandler clientHandler = new ClientHandler(clientSocket, userId);
19             clientHandlers.add(clientHandler);
20             new Thread(clientHandler).start();
21         }
22     }
23
24     static void broadcastMessage(String message, ClientHandler sender) {
25         for (ClientHandler client : clientHandlers) {
26             if (client != sender) {
27                 client.sendMessage(message);
28             }
29         }
30     }
31
32     static void removeClient(ClientHandler clientHandler) {
33         clientHandlers.remove(clientHandler);
34     }
35 }
```

Problems Javadoc Declaration Console Coverage

ChatClient (1) [Java Application] C:\Users\Yahya\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_23.0.2.v20250131-0604\jre\bin\javaw.exe (Mar 13, 2025, 7:53:40 a.m. elapsed: 0:01:11) [pid: 1208]

You are connected. Type your messages:

Welcome! You are User#2

I am good. You tell..

User#1: I am good too. How is life goin?



Project Explorer

- > Clock
- > clock.java
- > ClockApplication
- > ecommerce
- > LibraryCatalogProject
- > OnlineChatApp
 - > JRE System Library [JavaSE-23]
 - src
 - client
 - ChatClient.java
 - package-info.java
 - server
 - ChatServer.java
 - package-info.java
 - module-info.java

package-info.java

package-info.java

ChatServer.java

ChatClient.java

```
1 package client;
2
3 import java.io.*;
4 import java.net.*;
5
6 public class ChatClient {
7     private Socket socket;
8     private BufferedReader in;
9     private PrintWriter out;
10
11     public ChatClient(String serverAddress) throws IOException {
12         socket = new Socket(serverAddress, 1234);
13         in = new BufferedReader(new InputStreamReader(socket.getInputStream()));
14         out = new PrintWriter(socket.getOutputStream(), true);
15         new Thread(new IncomingMessagesHandler()).start();
16     }
17
18     public void sendMessage(String message) {
19         out.println(message);
20     }
21
22     private class IncomingMessagesHandler implements Runnable {
23         public void run() {
24             try {
25                 String incomingMessage;
26                 while ((incomingMessage = in.readLine()) != null) {
27                     System.out.println(incomingMessage);
28                 }
29             } catch (IOException e) {
30                 e.printStackTrace();
31             }
32         }
33     }
34 }
```

Problems Javadoc Declaration Console Coverage

ChatClient (1) [Java Application] C:\Users\Yahya\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_23.0.2.v20250131-0604\jre\bin\javaw.exe (Mar 13, 2025, 7:53:40 a.m. elapsed: 0:01:26) [pid: 1208]

You are connected. Type your messages:

Welcome! You are User#2

I am good. You tell..

User#1: I am good too. How is life goin?


```
// ChatServer.java
```

```
package server;
```

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

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

```
import java.util.*;
```

```
public class ChatServer {
```

```
    private static Set<ClientHandler> clientHandlers = new HashSet<>();
```

```
    private static int userId = 0;
```

```
    public static void main(String[] args) throws IOException {
```

```
        ServerSocket serverSocket = new ServerSocket(1234);
```

```
        System.out.println("Server is running...");
```

```
        while (true) {
```

```
            Socket clientSocket = serverSocket.accept();
```

```
            userId++;
```

```
        ClientHandler clientHandler = new ClientHandler(clientSocket, userId);

        clientHandlers.add(clientHandler);

        new Thread(clientHandler).start();

    }

}
```

```
static void broadcastMessage(String message, ClientHandler sender) {

    for (ClientHandler client : clientHandlers) {

        if (client != sender) {

            client.sendMessage(message);

        }

    }

}
```

```
static void removeClient(ClientHandler clientHandler) {

    clientHandlers.remove(clientHandler);

}

}
```

```
class ClientHandler implements Runnable {

    private Socket socket;

    private PrintWriter out;

    private BufferedReader in;

    private int userId;

    public ClientHandler(Socket socket, int userId) {

        this.socket = socket;

        this.userId = userId;

    }

    @Override

    public void run() {

        try {

            in = new BufferedReader(new InputStreamReader(socket.getInputStream()));

            out = new PrintWriter(socket.getOutputStream(), true);
```

```
        out.println("Welcome! You are User#" + userId);

        String message;

        while ((message = in.readLine()) != null) {

            System.out.println("User#" + userId + ": " + message);

            ChatServer.broadcastMessage("User#" + userId + ": " + message, this);

        }

    } catch (IOException e) {

        e.printStackTrace();

    } finally {

        try {

            socket.close();

        } catch (IOException e) {

            e.printStackTrace();

        }

        ChatServer.removeClient(this);

    }

}
```

```
    public void sendMessage(String message) {  
  
        out.println(message);  
  
    }  
  
}
```

```
// ChatClient.java
```

```
package client;
```

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

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

```
public class ChatClient {
```

```
    private Socket socket;
```

```
    private BufferedReader in;
```

```
    private PrintWriter out;
```

```
    public ChatClient(String serverAddress) throws IOException {
```

```
        socket = new Socket(serverAddress, 1234);
```



```
in = new BufferedReader(new InputStreamReader(socket.getInputStream()));

out = new PrintWriter(socket.getOutputStream(), true);

new Thread(new IncomingMessagesHandler()).start();

}
```

```
public void sendMessage(String message) {

    out.println(message);

}
```

```
private class IncomingMessagesHandler implements Runnable {

    public void run() {

        try {

            String incomingMessage;

            while ((incomingMessage = in.readLine()) != null) {

                System.out.println(incomingMessage);

            }

        } catch (IOException e) {

            e.printStackTrace();

        }

    }

}
```

```
    }  
    }  
}
```

```
public static void main(String[] args) throws IOException {  
  
    ChatClient client = new ChatClient("localhost");  
  
    BufferedReader userInput = new BufferedReader(new InputStreamReader(System.in));  
  
    System.out.println("You are connected. Type your messages:");  
  
    String message;  
  
    while ((message = userInput.readLine()) != null) {  
  
        client.sendMessage(message);  
  
    }  
  
}  
  
}
```