

Class MainServer

java.lang.Object
MainServer

```
public class MainServer
extends Object
```

This class is run to start the chat server, which is a multi-threaded web server. The class starts the UDP and TCP/IP server. This class handles keeping track of all of the UDP and TCP clients. It keeps a client list which is all of the currently connected clients. It also handles forwarding client messages to all of the clients on the client list.

Field Summary

Fields

Modifier and Type	Field and Description
private int	clientID The current ID number to be assigned, it increments as each client connects
private ArrayList<Socket>	clientsTCP the list of TCP clients connected to the server
private ArrayList<ServerThread>	clientsUDP the list of UDP client ports on sever
private PrintWriter	out create the object to send the message
protected static String	removeMe the special message sent when the disconnect button is hit by the client to to remove a client from the client
private ServerGUI	serverGUI the object to access the GUI for the server to update the client list
private ServerSocket	serverSocket create a TCP socket for server for clients to connect to

Constructor Summary

Constructors

Constructor and Description

`MainServer()`

Constructor starts the GUI and starts the TCP and UDP servers

Method Summary

All Methods Static Methods Instance Methods Concrete Methods

Modifier and Type	Method and Description
void	addClient (ServerThread clientThread) add a client to the client list
void	closeSocket () Closes the current socket.
int	getClientID () Returns the client ID which is a unique number
static void	main (String[] args) Calls the constructor to create a new instance of the main server
void	removeClient (ServerThread clientThread) remove a client from the client list
void	sendToAll (String message) Sends a message to all of the UDP and TCP/IP clients connected to the server

Methods inherited from class java.lang.Object

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`,
`toString`, `wait`, `wait`, `wait`

Field Detail

serverGUI

```
private ServerGUI serverGUI
```

the object to access the GUI for the server to update the client list

clientsTCP

```
private ArrayList<Socket> clientsTCP
```

the list of TCP clients connected to the server

clientsUDP

```
private ArrayList<ServerThread> clientsUDP
```

the list of UDP client ports on sever

removeMe

```
protected static String removeMe
```

the special message sent when the disconnect button is hit by the client to to remove a client from the client

clientID

```
private int clientID
```

The current ID number to be assigned, it increments as each client connects

serverSocket

```
private ServerSocket serverSocket
```

create a TCP socket for server for clients to connect to

out

```
private PrintWriter out
```

create the object to send the message

Constructor Detail

MainServer

```
public MainServer()
```

Constructor starts the GUI and starts the TCP and UDP servers

Method Detail

sendToAll

```
public void sendToAll(String message)
```

Sends a message to all of the UDP and TCP/IP clients connected to the server

Parameters:

message - the message that is being sent

getClientID

```
public int getClientID()
```

Returns the client ID which is a unique number

Returns:

the client ID number

removeClient

```
public void removeClient(ServerThread clientThread)
```

remove a client from the client list

Parameters:

clientThread - the client being removed from the list

addClient

```
public void addClient(ServerThread clientThread)
```

add a client to the client list

Parameters:

`clientThread` - the client being added

closeSocket

```
public void closeSocket()
```

Closes the current socket. Is called when the client exits the window.

main

```
public static void main(String[] args)
```

Calls the constructor to create a new instance of the main server

Parameters:

`args` - not used