

## Week 1 – Socket programming in java

### Server Programming

#### Establish a Socket Connection

To write a server application two sockets are needed.

- A ServerSocket which waits for the client requests (when a client makes a new Socket())
- A plain old Socket to use for communication with the client.

#### Communication

getOutputStream() method is used to send the output through the socket.

#### Close the Connection

After finishing, it is important to close the connection by closing the socket as well as input/output streams.

### Client-Side Programming

#### Establish a Socket Connection

To connect to another machine we need a socket connection. A socket connection means the two machines have information about each other's network location (IP Address) and TCP port. The java.net.Socket class represents a Socket.

To open a socket:

```
Socket socket = new Socket("127.0.0.1", 5000)
```

- The first argument – **IP address of Server**. ( 127.0.0.1 is the IP address of localhost, where code will run on the single stand-alone machine).
- The second argument – **TCP Port**. (Just a number representing which application to run on a server. For example, HTTP runs on port 80. Port number can be from 0 to 65535)

#### Communication

To communicate over a socket connection, streams are used to both input and output the data.

#### Closing the connection

The socket connection is closed explicitly once the message to the server is sent.

*In the program, the Client keeps reading input from a user and sends it to the server until "Over" is typed.*