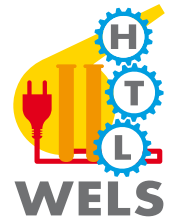




MULTITHREADED SERVER (MAIN SERVER)



```
// we use a cached thread pool for performance
// ClientHandler will not be created only once!
ExecutorService executorService = Executors.newCachedThreadPool();

// open server socket
try (ServerSocket serverSocket = new ServerSocket(port)){

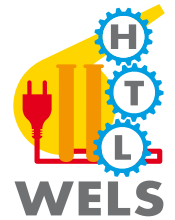
    while (true) {
        // listening for new clients -> get a client socket
        Socket client = serverSocket.accept();

        // start thread for THIS new connection (client socket),
        // where communication is handled
        executorService.execute(new ClientHandler(client));
    }

} catch (IOException e) {
    e.printStackTrace();
}
```



MULTITHREADED SERVER (CLIENT HANDLER)



```
public class ClientHandler implements Runnable {

    private Socket client;

    public ClientHandler(Socket client) {
        this.client = client;
    }

    @Override
    public void run() {
        // get input + output streams for sending/receiving,
        // then communicate...

        ...
        finally {
            try {
                if (client!=null) client.close();
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
    }
}
```