3. Simple Ping Applicaton

Functionality

Given some hosts in the config:

- jasmin.com
- oranum.com

Ping with ICMP protocol

Ping command: *ping -n 5 HOST*

We should constantly ping these hosts. It should be scheduled with a given delay (the delay should be **configured**) we do the ping command. The ping command should be **configured**, and the HOST is the dynamic part.
If a current ping is still runs for a given host, then we do nothing, an another scheduled run will occur. **We have to store the last icmp ping result for the given host and when it happened.** The result should contain the lines received from command.
The hosts should be checked parallel.
If we have connection timeout during ping or any error (lets say the pockets sent and received do not match, or the packet loss is not 0) then we call the **Report** functionality for the given host (see below).

Ping with TCP/IP protocol

We should constantly ping these addresses. It should be scheduled with a given delay (the delay should be **configured**) we do HTTP request with timeout (the timeout should be **configured**) to the given url. The request can be anything which do the given job. If a current request still runs for a given url, then we do nothing, an another scheduled run will occur. **We have to store the last tcp ping result for the given host and when it happened.** This result should contain which url, and how much was the response time (milliseconds) and what was the response http status.
The addresses should be checked parallel.
If we have connection timeout, or host not reachable, or the response time was above a defined value, (the value should be **configured**) during ping then we call the **Report** functionality for the given host (see below).

Trace route

We should constantly trace route these hosts. It should be scheduled, and with a given delay (the delay should be **configured**) we do a *tracert HOST*. The trace command should be **configured**, and the HOST is the dynamic part. If a current trace route is still runs for a given host, then we do nothing, an another scheduled run will occur. **We have to store the last trace route result for the given host and when it happened.** The result should contain the lines received from command.
The hosts should be checked parallel.

Report

Calling the report we have to send data with POST to a given url (the url should be **configured**) in json format. The POST body should have the json value.
The data should contain 4 parts for the reporting host: the host, last icmp ping result for the host, last tcp ping result for the host, last trace route result for the host.
Example structure:

```
{"host":"the given host", "icmp_ping":"result lines of the last icmp ping command",
"tcp_ping":"result lines of the last tcp ping command", "trace":"result lines of the
last trace command"}
```

We have to log these reports too. The log should go into the local file, and the log should be able to **configured**. The log level should be warning.

The aim

We'd like to check the user connections - status, time to the given host.

Restriction

Java version: 1.8
Use Threads for parallel jobs.
The program should run on windows or/and on linux. It's not determined but should run at least on one of it.
Maven project
Runnable big jar file with dependency (libs if any) embedded.
Use core java, no frameworks.
Unit tests