An Exercise on Apache JMeter (http://jakarta.apache.org/jmeter/)

For the latest version, please check the Moodle course „Software Quality (Tarkvara kvaliteet)" (or tepandi.ee)

1. Getting started (to ensure that JMeter is up and working)
    - See http://jakarta.apache.org/jmeter/usermanual/get-started.html. If JMeter is not already downloaded, then follow the instructions on http://jmeter.apache.org/download_jmeter.cgi to download JMeter
    - Start JMeter - run jmeter.bat from the folder where the unzip was done; for example, may be in C:\Program Files\jMeter\…\bin\jmeter.bat)
    - Open a sample test plan (eg File -> Open –> Examples -> CSVSample.jmx)
    - Select "View Results in Table" (in the left hand window tree), then Run –> Start (or click Start icon)
    - Compare the Jmeter results table and the Thread Group elements. Go through the left window tree with right clicks => help. Explain some results (see also point 3 below).
    - Close JMeter
2. First tests: Basic web test plan in GUI mode (to have a basic understanding of web testing with JMeter)
    - Follow http://jmeter.apache.org/usermanual/build-web-test-plan.html to build a basic test plan (minimum: test plan -> thread group, HTTP request, listener View results in table). Save the test plan
    - Run the thread group (Run->Start)
    - Add listener "View results in table". Understand the Graph Results view:
        - How is the value "No of samples" computed?
        - How many dots, why?
        - Select different items to be displayed on the graph.
        - Why is average decreasing or increasing (hint: see table)?
        - Is the "Average" value reasonable? Median?
        - What does "Latest sample" mean (hint: see the table)?
        - Is the Throughput value reasonable (hint: use the table to find total time and compare this with the number of requests)?
    - Add the tree listener. Clear the graph and table results (rightclick on the listener in the tree). Compare the latest sample value in the graph, tree and table.
    - Sort the results table by "Start Time" values. Are these values reasonable? (hint: Ramp-up period from the Thread Group / the Number of Threads)
    - In what sequence are the requests executed (hint: see Thread Names in the table)? Why this sequence? If you insert Ramp-up period=0, why the ordering changes?
    - Which of the sites loads faster? What may be the reason?

- Why so large differences in load times, even if the sizes are not so different? Why might the load times change in the sequence?... etc
- Remark 1: The GUI mode is used for creating and debugging a test plan. For actual load testing, this test plan should be used in the Command-line (Non-GUI) mode.
- Remark 2: Be careful when repeating test runs for the same website. It may be a good idea to reduce the number of requests (or address a different website).

3. Experiment with a safe test (to understand JMeter deeper)
    - Open a sample test plan as in Section 1 (CSVSample.jmx). See associated csv files (hint: see the examples folder in …\bin\examples\...)
    - How is this test operating? Why is it safe to experiment?
    - How the while loop works? (hint: https://commons.apache.org/proper/commons-jexl/reference/syntax.html#Language_Elements)
    - Run, see the results in the table. Why u2 is tested? Why no actions for u2?
    - Delete the table and tree contents (Run-> Clear)
    - Change the thread parameters (eg number of threads), run, explain the results
    - How to cycle the test for more requests? (hint: Get user details…Recycle on EOF)
    - Add Graph Results and other listeners (to the top node)
    - Let the test run in cycle, interpret the Graph Results. Are the Data values converging? The others? Explain.
4. JMeter and others: analysis
    - Opne the CSVSample.jmx file in a code editor. Analyse its contents, compare with the test plan in GUI mode. Change and save the test plan, follow changes in the file.
    - Analyse the Test.jmx test plan from the JM extras folder.
    - Could Jmeter be used to run performance tests of desktop applications? When does this make sense? Hint: see eg https://www.blazemeter.com/blog/how-run-performance-tests-desktop-applications-using-jmeter.
    - Install the RESTful Stress App. Compare with Jmeter.
    - Try recording tests, compare (http://jmeter.apache.org/usermanual/jmeter_proxy_step_by_step.html). Is it easier than to write tests?
    - Could JMeter be used as a functional testing tool? Selenium as a load testing tool?
5. See other materials, eg JMeter wiki (https://wiki.apache.org/jmeter/JMeterFAQ).
6. Use JMeter for your own project. NB! Be careful with load testing not agreed with the site owner.
    - See also http://jakarta.apache.org/jmeter/usermanual/boss.html etc.
    - Please see also remarks on non-functional testing in http://tepandi.ee/Jaak_Tepandi_Sw_QS_org.pdf.