# Лабораторная работа №5

## Автоматизация тестирования при помощи Apache JMeter, часть 2

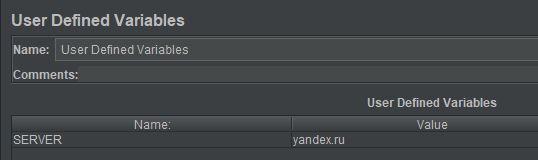
**Цель** **лабораторной работы:** изучение функций Apache JMeter.

В первой части лабораторной работы вы познакомились:

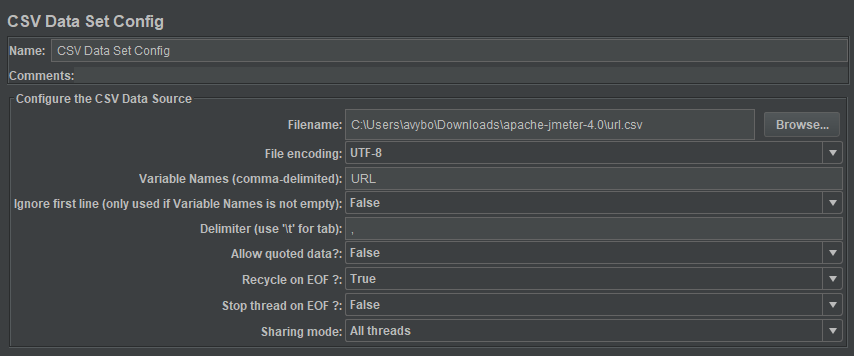
* C возможностью записи запросов и ответов к web-приложению через встроенный в JMeter proxy-сервер (элемент **HTTP(S) Test Script Recorder**, где настраивается и включается proxy-сервер, и элемент **Recording controller**, куда складываются записанные вопросы и ответы).
* С элементами **Configuration elements** в которых можно задавать настойки, распространяющиеся на весь тест план (переменные, дефолтные настройки HTTP-запросов, управление cookies).
* Элементом **Thread Group**, который объединяет в себе в себе действия, которые будут выполнять виртуальные «пользователи» во время проведения тестирования.
* Логическими контроллерами **Once only controller**, **Interleave Controller** и другими (<https://jmeter.apache.org/usermanual/test_plan.html>), которые позволяют сформировать логическую структуру действий пользователя (какие действия повторять единожды за тест, какие при каждом повторении, какие выбирать случайным образом из списка и т.д.).
* Элементами **Sampler**, которые содержат собственно действия виртуальных «пользователей» (например, в виде HTTP-запросов к web-приложению).
* Типичной структурой теста в JMeter:
* Test Plan
  + Thread Group
    - Once Only Controller
      * Login Request (an [HTTP Request](https://jmeter.apache.org/usermanual/component_reference.html#HTTP_Request))
    - Load Search Page (HTTP Sampler)
    - Interleave Controller
      * Search "A" (HTTP Sampler)
      * Search "B" (HTTP Sampler)
      * HTTP default request (Configuration Element)
    - HTTP default request (Configuration Element)
    - Cookie Manager (Configuration Element)

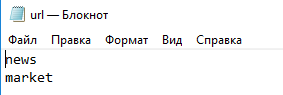
Следует отметить, что никакой более или менее серьезный тест не обходится без введения переменных и параметризации (создание одного параметра, содержащего несколько значений, для однотипных действий, отличающихся только этим параметром ).

В значения переменных удобно записывать адреса серверов, логины, пароли и другую информацию, которая может присутствовать во многих местах теста, и при этом в некоторых случаях требует изменения во всех этих местах (например, сменился IP-адрес тестируемого web-сервера). Переменные можно задавать в элементе **User Defined Variables:**

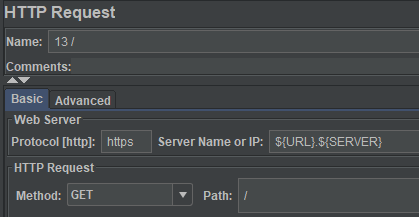


Параметризовать удобно те действия, которые отличаются друг от друга одним признаком. Например, если в ходе теста требуется перейти по ряду однотипных ссылок и проверить, что все страницы по ссылкам существуют и открываются, то все URL, по которым надо пройти (или их меняющаяся часть), могут быть записаны в csv файл и подставляться оттуда в процессе тестирования по очереди. Такая возможность настраивается в элементе **CSV Data Set Config**, например, для файла со списком URL:

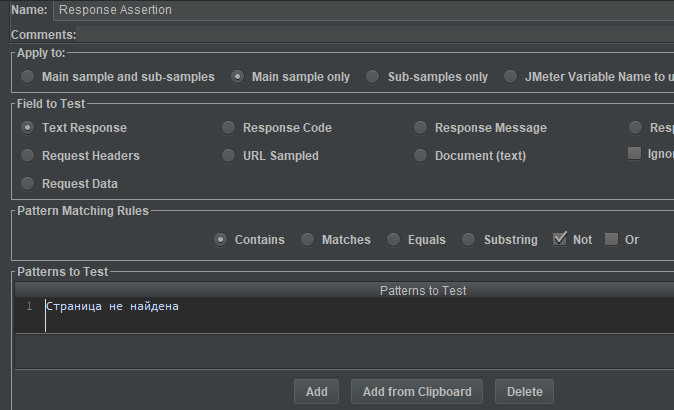




Для подстановки имени переменной вместо конкретного значения используется конструкция **${Имя переменной}**:



Для проверки того, что полученный на HTTP-запрос ответ соответствует определенным критериям или содержит определённую информацию, используются элементы **Assertion** (подробнее – <https://jmeter.apache.org/usermanual/component_reference.html#assertions> ). Так, например, настройка, приведенная ниже, проверяет, что в пришедшем HTTP-ответе нет слов «Страница не найдена»:



Для сохранения, а также удобного представления результатов теста служат элементы **Listener**. Например, элемент **Aggregate Report** используется для табличного представления времени ответа на запросы, отправляемые в процессе тестирования, **Aggregate Graph** – для визуального представления той же информации, **Simple Data Writer** – для записи в файл. Подробнее о разных типах Listeners: <https://jmeter.apache.org/usermanual/component_reference.html#listeners>

Для имитации пауз, которые делает пользователь в процессе работы, могут использоваться элементы **Timers**. Подробнее о них: <https://jmeter.apache.org/usermanual/component_reference.html#timers>

**Задание на лабораторную работу**:

В записанном в процессе лабораторной работы №4 тесте необходимо:

1. Добавить переменную Server с именем тестируемого сервера (yandex.ru) и в HTTP Samplers заменить yandex.ru на имя переменной.
2. Создать файл со списком URL (или их частей), по которым вы переходили в предыдущей лабораторной работе. В элементе **CSV Data Set Config** создать переменную, берущую информацию из этого файла, затем заменить несколько соответствующих HTTP Samplers на 1-2 с указанием созданной переменной вместо конкретного URL.
3. При помощи элемента **Response Assertion** проверить, что на всех страницах, которые открываются в ответ на HTTP-запросы в тесте, есть слово «Яндекс».
4. Добавить несколько таймеров, имитирующих паузы, которые делает реальный пользователь.
5. Добавить несколько различных Listeners и посмотреть, какую информацию и в каком виде они выдают.

**Содержание отчета:** Созданный тест план необходимо сохранить и прислать на электронную почту [a.vybornova@gmail.com](mailto:a.vybornova@gmail.com)

**Дополнительная литература:**

1. <https://jmeter.apache.org/usermanual/index.html>
2. <https://wiki.apache.org/jmeter>
3. <https://habrahabr.ru/post/140310/>