

# JMeter.

О программе, создание простого запроса.

# JMeter

- Инструмент нагрузочного тестирования
- Кроссплатформенный
- Бесплатный



# Инструменты



LoadComplete



The Grinder

# Преимущества в JMeter.

- Графический интерфейс
- Кроссплатформенность
- Бесплатно весь функционал
- Генерация отчетов
- Работа в консоли
- Возможность расширения и тонкой настройки

# Возможности

- Эмуляция одновременной работы пользователей
- Распределенная нагрузка
- Снятие метрик

**JMeter - это не браузер!**

# Установка и настройка

100% Java приложение - установить java и готов к использованию.

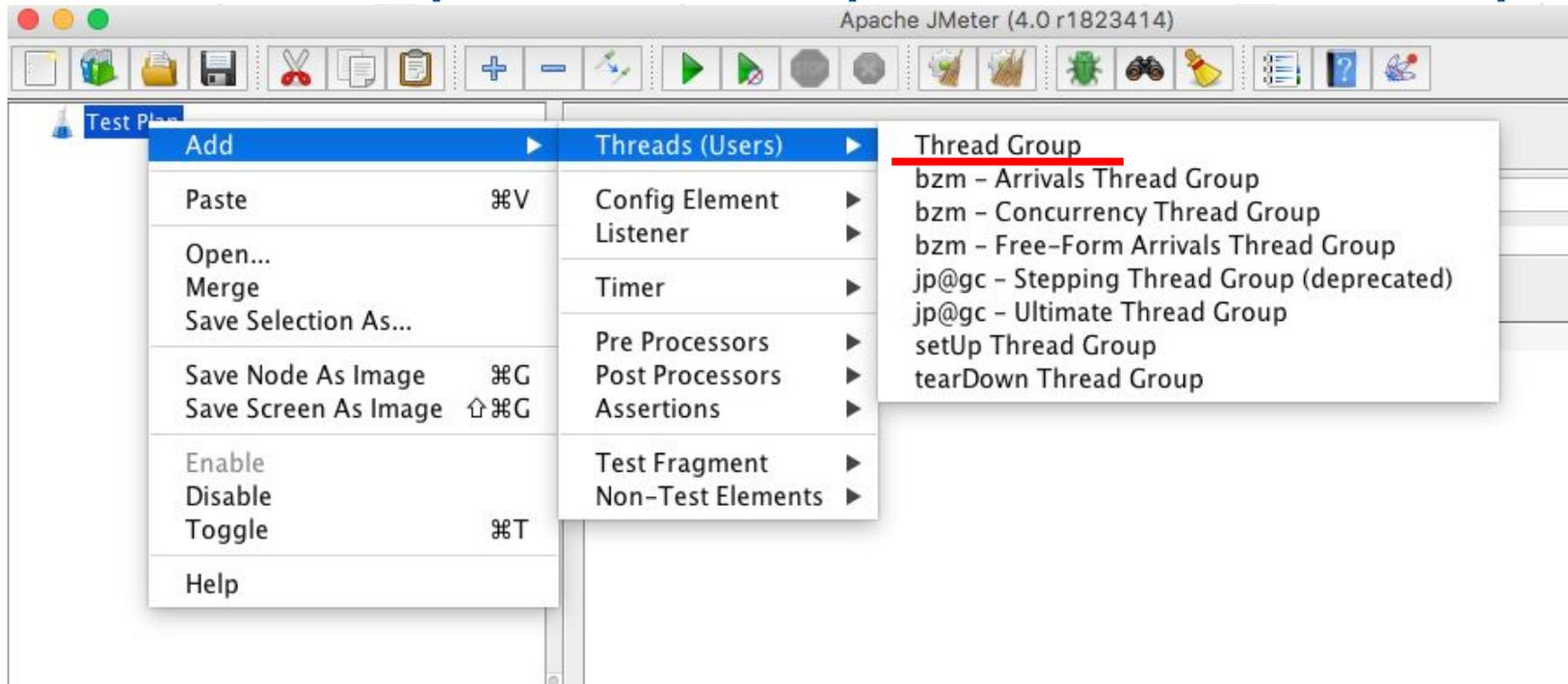
<https://www.java.com/ru/download/>

[http://jmeter.apache.org/download\\_jmeter.cgi](http://jmeter.apache.org/download_jmeter.cgi)

<http://jmeter-plugins.org/downloads/all/>

<https://github.com/search?utf8=%E2%9C%93&q=Jmeter>

# Создание простого запроса. Thread Group



# Создание простого запроса. Thread Group

**Test Plan** - состоит из одной или нескольких групп потоков Thread Group, конфигурационных элементов, проверок и отчетов. Все тесты пишутся внутри тест плана, тест план является наивысшей точкой от которой тест "растет" вниз.

**Thread Group** - тестовый поток, в JMeter исполняется то, что находится внутри Thread Group, в тесте может быть несколько Thread Group, при запуске теста все Thread Group будут выполняться независимо и одновременно, при определенных настройках возможен последовательный запуск Thread Group находящихся внутри Test Plan.

# Создание простого запроса. Thread Group

## Thread Properties

- Number of Threads (users): 1 **Number of Threads** - это количество юзеров которые будут заходить на сайт.
- Ramp-Up Period (in seconds): 1 **Rump-Up period** - это период за который все наши юзеры зайдут на сайт.
- Loop Count:  Forever 1 **Loop Count** - это кол-во циклов исполнения юзерами действий в Thread Group.

То есть, если установить:

Number of Threads = 10  
Rump-Up period = 15 sec  
Loop Count = 100

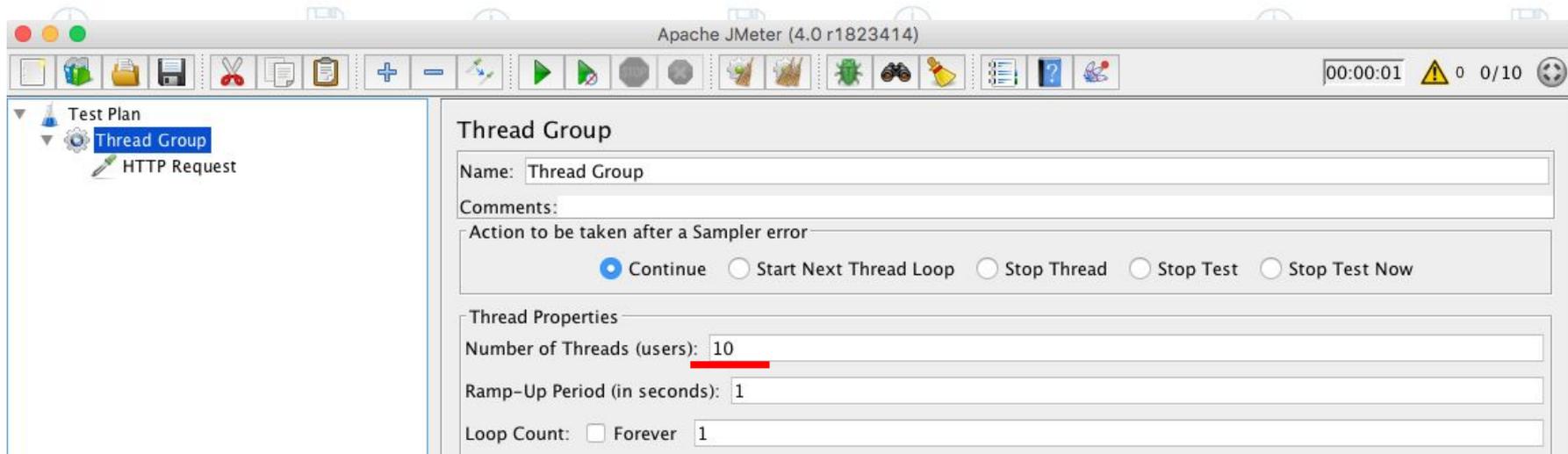
Total requests = Number of Threads \* Loop Count = 1000

Total time = (Rump-Up period / Number of Threads) \* Loop Count = 150 sec

То есть выходит, что будет осуществлено 100 повторений запросов, одновременно работающих 10 юзеров за 15 секунд. 1 юзер = 1.5 сек \* 100 повторений = 1000 запросов за 150 сек.

# Создание простого запроса. Thread Group

Но мы пока что установим 10 одновременно работающих юзеров



# Создание простого запроса. Thread Group

**Thread Group**

Name:  **1 название группы**

Comments:

Action to be taken after a Sampler error **2 что делать с неуспешно пройденной группой**

Continue  Start Next Thread Loop  Stop Thread  Stop Test  Stop Test Now

Thread Properties

Number of Threads (users):

Ramp-Up Period (in seconds):

Loop Count:  Forever

Delay Thread creation until needed **3 активирует задержку**

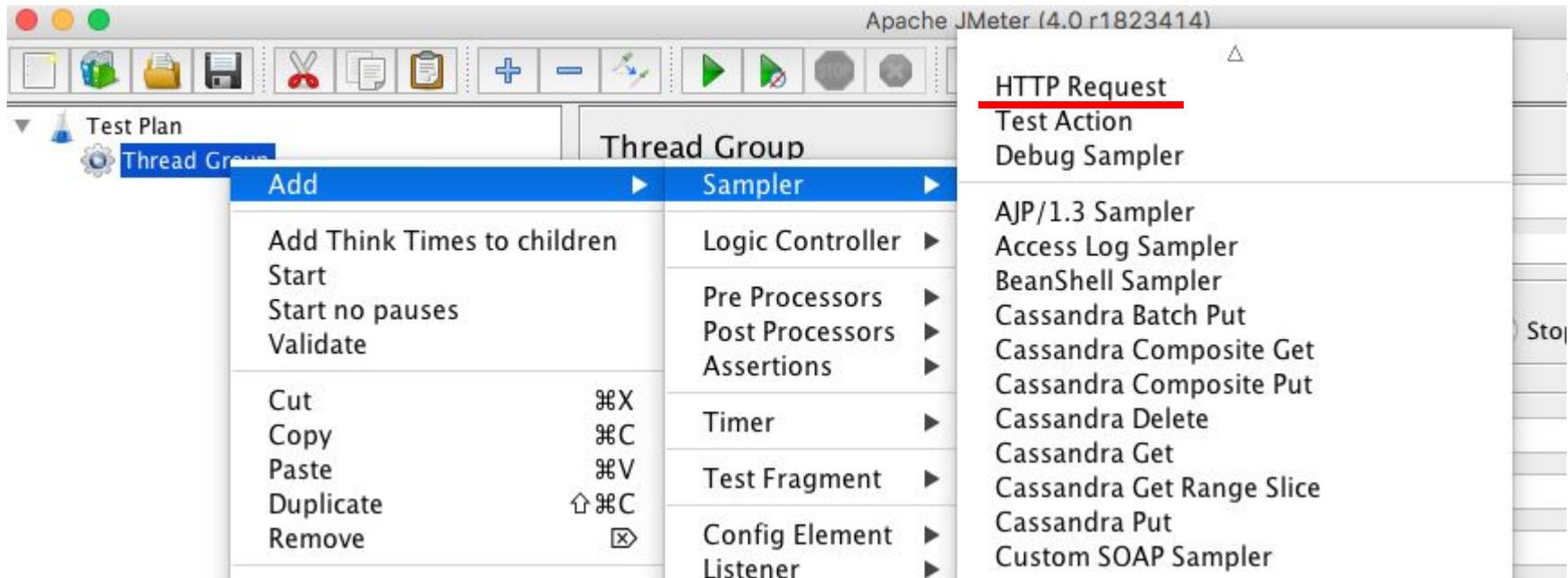
Scheduler **4 активирует планировщик**

Scheduler Configuration

Duration (seconds)  **5 время выполнения тестов**

Startup delay (seconds)  **6 время задержки перед стартом**

# Создание простого запроса. Sampler



# Создание простого запроса. Sampler

HTTP Request

Name:

Comments:

Basic Advanced

Web Server

Protocol [http]:  Server Name or IP:  Port Number:

HTTP Request

Method:  Path:  Content encoding:

Redirect Automatically  Follow Redirects  Use KeepAlive  Use multipart/form-data for POST  Browser-compatible headers

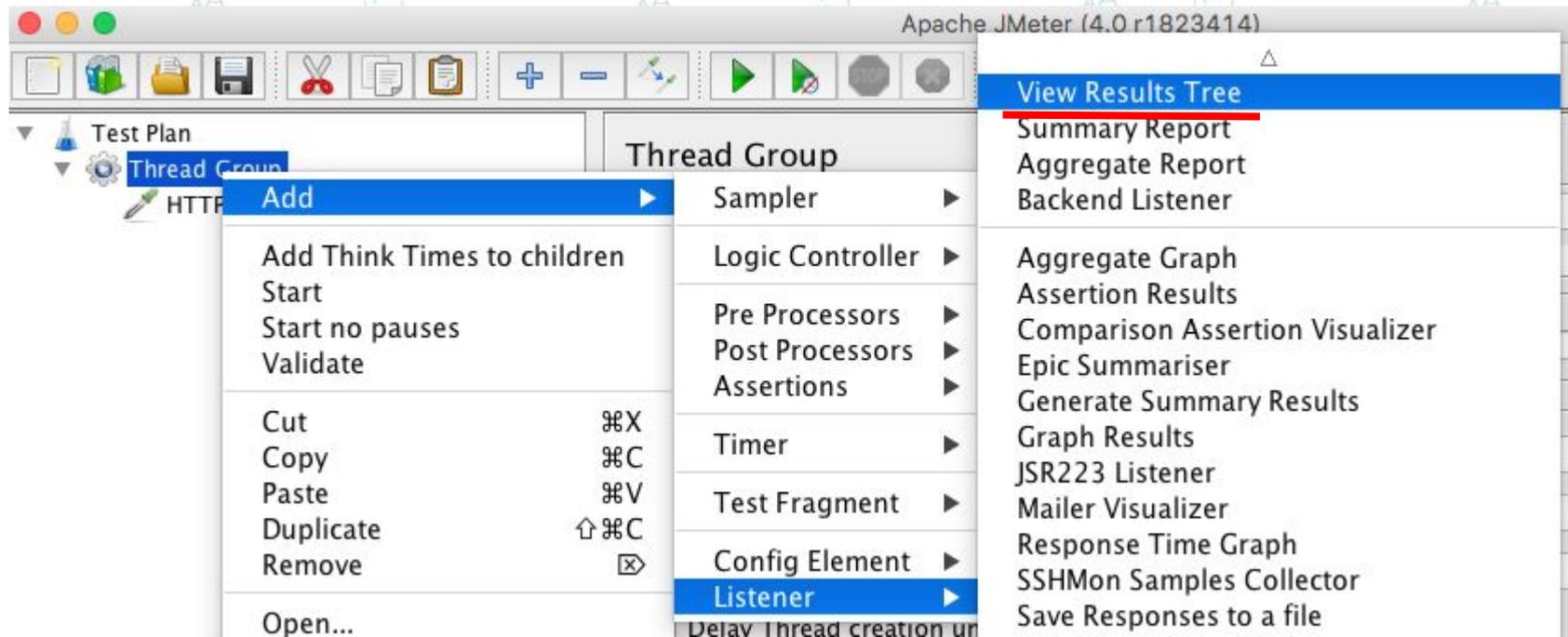
Parameters Body Data Files Upload

# Создание простого запроса. Sampler

Sampler - образцы (элементы теста)

HTTP Request - используется для отправки HTTP/HTTPS запросов, позволяет отправлять файлы различных типов.

# Создание простого запроса. Listener



# Создание простого запроса. Listener

Listeners - слушатели (отчеты), собраны различные варианты отчетов

View Results Tree - показывает дерево ответов на все семплы. В дополнение к ответам можно увидеть, которое понадобилось для получения ответа и код ответа.

# Создание простого запроса. Result

The image shows a web browser's developer tools interface. On the left, a list of network requests is displayed, with the first one selected. A red box highlights the 'HTML' content type dropdown menu. The main pane on the right shows the 'Response data' tab, also highlighted with a red box. The response content is a screenshot of the Google search page in Ukrainian, featuring a search bar and buttons for 'Пошук Google' and 'Мені пощастить'.

HTML

- HTTP Request

Sampler result Request Response data

Пошук Зображення Карти YouTube Новини Gmail Диск Календар Інше

Історія веб-пошуку | Налаштування | Увійти

Розширені пошукМовні інструменти

Пошук Google Мені пощастить

РекламаПошук для бізнесуВсе про GoogleGoogle.com

# Простой Test

---

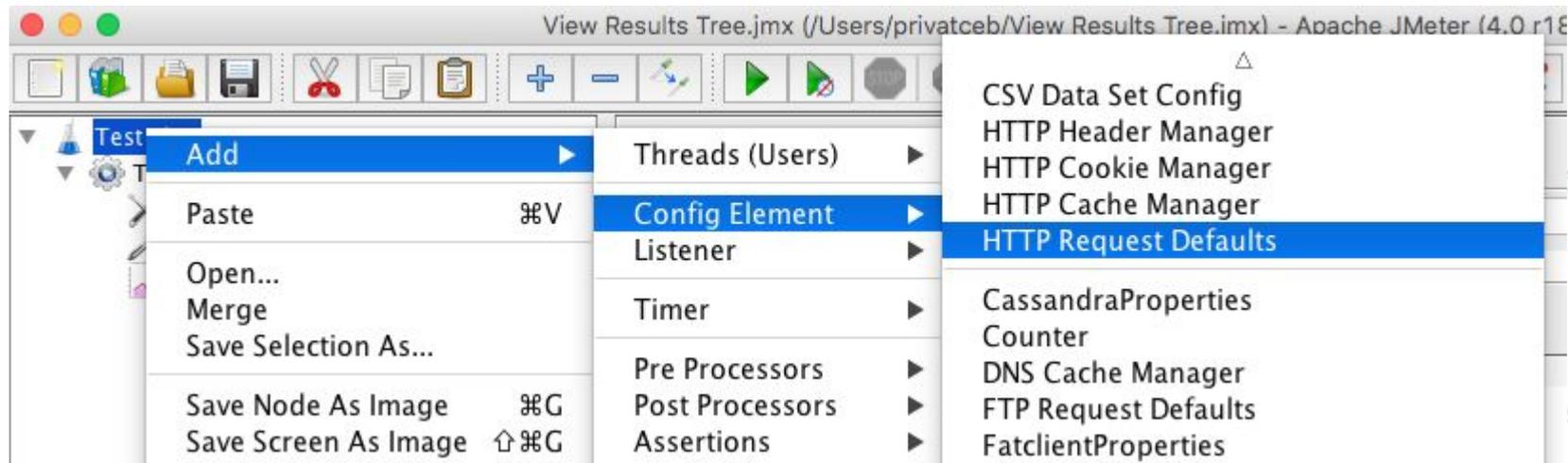
## JSONPlaceholder

Fake Online REST API for Testing and Prototyping

Powered by [JSON Server](#) and [LowDB](#)

<https://jsonplaceholder.typicode.com>

# Простой Test



# Простой Test

## HTTP Request Defaults

Если вы собираетесь отправить несколько запросов на один веб сервер, рекомендуется использовать HTTP Request Defaults, не придется вводить одну и ту же информацию для каждого запроса HTTP.

View Results Tree.jmx (/Users/privatceb/View Results Tree.jmx) - Apache JMeter (4.0 r1823414)

Test Plan

- Test
  - HTTP Request Defaults**
  - HTTP Request
  - View Results Tree

### HTTP Request Defaults

Name: HTTP Request Defaults

Comments:

Basic Advanced

Web Server

Protocol [http]:  Server Name or IP:

View Results Tree.jmx (/Users/privatceb/View Results Tree.jmx) - Apache JMeter (4.0 r1823414)

Test Plan

- Test
  - HTTP Request Defaults
  - HTTP Request**
  - View Results Tree

Thread Properties

Number of Threads (users):

Ramp-Up Period (in seconds):

Loop Count:  Forever

### HTTP Request

Name: HTTP Request

Comments:

Basic Advanced

Web Server

Protocol [http]:  Server Name or IP:

HTTP Request

Method:  Path:

И переименуем в GET

# Простой Test

The screenshot shows the Apache JMeter 4.0 interface. The title bar reads "View Results Tree.jmx (/Users/privatceb/View Results Tree.jmx) - Apache JMeter (4.0 r1823414)". The left sidebar shows a tree view with "Test Plan" expanded to "Test", which contains "HTTP Request Defaults", "GET", and "View Results Tree" (highlighted in blue). The main panel is titled "View Results Tree" and contains the following elements:

- Name: View Results Tree
- Comments: (empty)
- Write results to file / Read from file: Includes a "Filename" input field, a "Browse..." button, and checkboxes for "Log/Display Only: Errors" and "Successes". A "Configure" button is also present.
- Search: Includes a search input field, checkboxes for "Case sensitive" and "Regular exp.", and "Search" and "Reset" buttons.
- Response data: A tabbed view showing the response of the selected sampler. The selected sampler is "Text" (with a dropdown arrow) and "GET" (with a green checkmark icon). The response is a JSON object:

```
{  "userId": 1,  "id": 1,  "title": "sunt aut facere repellat provident occaecati excepturi optio reprehenderit",  "body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et cum\nreprehenderit molestiae ut ut quas totam\nnostrum rerum est autem sunt rem veniet architecto"}
```

# Простой Test

## HTTP Request

Name: POST

Comments:

Basic Advanced

### Web Server

Protocol [http]: Server Name or IP: Port Number:

### HTTP Request

Method: POST Path: posts Content encoding:

Redirect Automatically  Follow Redirects  Use KeepAlive  Use multipart/form-data for POST  Browser-compatible headers

Parameters Body Data Files Upload

### Send Parameters With the Request:

Name:	Value	Encode?	Include Equals?	Equals?
userid	333	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
id	1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
title	Тестовый запрос Post	<input type="checkbox"/>	<input checked="" type="checkbox"/>	

View Res

## Test Plan

### Test

HTTP Request Defaults

GET

POST

View Results Tree

Detail

Add

Add from Clipboard

Delete

Up

Down

# Простой Test

The screenshot displays the Apache JMeter 4.0 interface. The title bar reads "View Results Tree.jmx (/Users/privatceb/View Results Tree.jmx) - Apache JMeter (4.0 r1823414)". The top toolbar includes icons for file operations, test execution, and search. The left sidebar shows a "Test Plan" tree with "Test" expanded to include "HTTP Request Defaults", "GET", "POST", and "View Results Tree" (highlighted in blue).

The main "View Results Tree" panel is configured as follows:

- Name: View Results Tree
- Comments: (empty)
- Write results to file / Read from file:
  - Filename: (empty)
  - Browse... button
  - Log/Display Only:  Errors  Successes
  - Configure button
- Search: (empty)  Case sensitive  Regular exp. Search Reset

At the bottom, the "Text" tab is selected, showing a list of samplers on the left and the "Response data" tab on the right. The samplers list includes "GET" and "POST" (highlighted in blue). The "Response data" tab displays the following JSON response:

```
{
  "title": "foo",
  "body": "bar",
  "userId": "1",
  "id": 101
}
```

# Простой Test

Web Server

Protocol [http]:  Server Name or IP:

HTTP Request

Method: PUT Path: posts/1

Redirect Automatically  Follow Redirects  Use KeepAlive  Use multipart/form-data

Parameters Body Data Files Upload

```
1 {  
2   id: 1,  
3   title: 'foo',  
4   body: 'bar',  
5   userId: 1  
6 }
```

Test Plan

- Test
  - HTTP Request Defaults
  - GET
  - POST
  - PUT
  - DELETE
  - View Results Tree

Web Server

Protocol [http]:  Server Name or IP:

HTTP Request

Method: DELETE Path: posts/1

Redirect Automatically  Follow Redirects  Use KeepAlive  Use multipart/form-data

Parameters Body Data Files Upload

```
1 {id: 1}
```

Test Plan

- Test
  - HTTP Request Defaults
  - GET
  - POST
  - PUT
  - DELETE
  - View Results Tree

# Простой Test

The screenshot displays the Apache JMeter 4.0 interface. The main window title is "View Results Tree.jmx (/Users/privatceb/View Results Tree.jmx) - Apache JMeter (4.0 r1823414)". The interface includes a toolbar with various icons for file operations and execution. On the left, a tree view shows the test plan structure:

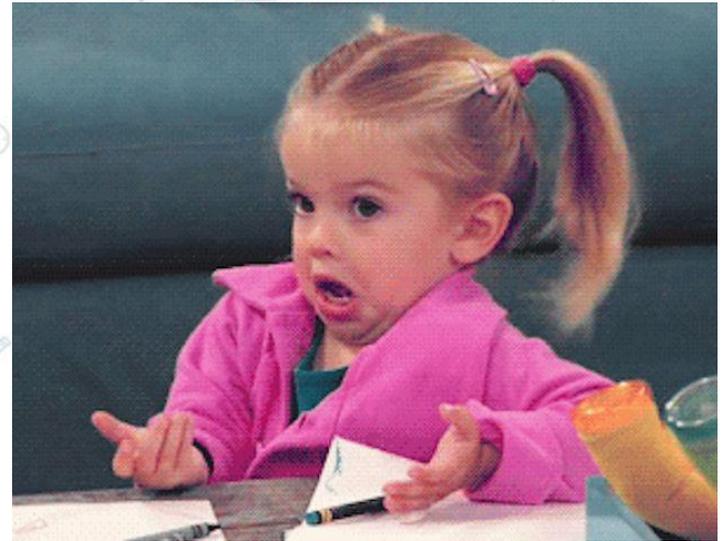
- Test Plan
  - Test
    - HTTP Request Defaults
    - GET
    - POST
    - PUT
    - DELETE
    - View Results Tree** (selected)

The main area is titled "View Results Tree" and contains the following fields and controls:

- Name: View Results Tree
- Comments:
- Write results to file / Read from file
  - Filename:
  - Browse...
  - Log/Display Only:  Errors  Successes
  - Configure
- Search:   Case sensitive  Regular exp.
- Text: 
  - GET
  - POST
  - PUT
  - DELETE**
- Sampler result | Request | Response data
  - Thread Name: Test 1-1
  - Sample Start: 2018-07-25 00:01:00 EEST
  - Load time: 726
  - Connect Time: 0

# Простой Test

Но что это за тест без проверки результата?



# Простой Test. Assertions - проверки

View Results Tree.jmx (/Users/privatceb/View Results Tree.jmx) - Apache JMeter (4.0 r1823414)

Test Plan

- Test
  - HTTP Request Defaults
    - GET
      - Add
        - Assertions
          - Response Assertion**
            - JSON Assertion
            - Size Assertion
            - JSR223 Assertion
            - XPath Assertion
            - Binary Response Assertion
            - Compare Assertion
            - Duration Assertion**
              - HTML Assertion
              - MD5Hex Assertion
              - SMIME Assertion
              - SOAP XPath Assertion
              - XML Assertion
              - XML Schema Assertion
              - jp@gc - JSON Path Assertion
              - BeanShell Assertion
        - Timer
        - Pre Processors
        - Post Processors
        - Config Element
        - Listener
      - Insert Parent
      - Cut ⌘X
      - Copy ⌘C
      - Paste ⌘V
      - Duplicate ⇧⌘C
      - Remove ⌘X
      - Open...
      - Merge
      - Save Selection As...
      - Save as Test Fragment
      - Save Node As Image ⌘G
      - Save Screen As Image ⇧⌘G
      - Enable
      - Disable

HTTP Request

Name: GET

Advanced

Redirect Automatically

Use multipart/form-data

Data Files Upload

With the Request: Value

# Простой Test. Assertions - проверки

View Results Tree.jmx (/Users/privatceb/View Results Tree.jmx) - Apache JMeter (4.0 r1823414)

Test Plan

- Test
  - HTTP Request Defaults
  - GET
    - Response Assertion
    - Duration Assertion**
    - POST
    - PUT
    - DELETE
  - View Results Tree

**Duration Assertion**

Name: Duration Assertion

Comments:

Apply to:
   
 Main sample and sub-samples
   
 Main sample only
   
 Sub-samples only

Duration to Assert

Duration in milliseconds: 100

Text

- GET
- Duration Assertion**
- POST
- PUT
- DELETE

**Assertion result**

Assertion error: false  
 Assertion failure: true  
 Assertion failure message: The operation lasted too long: It took 313 milliseconds, but should not have lasted longer than 100 milliseconds.

# Простой Test. Assertions - проверки

Response Assertion

Name: Response Assertion

Comments:

Apply to:

Main sample and sub-samples  Main sample only  Sub-samples only  JMeter Variable Name to use

Field to Test

Text Response  Response Code  Response Message  Response Headers

Request Headers  URL Sampled  Document (text)  Ignore Status

Request Data

Pattern Matching Rules

Contains  Matches  Equals  Substring  Not  Or

Patterns to Test

	Patterns to Test
1	userId1

- Text
- ✖ GET
  - ✖ **Response Assertion**
  - ✔ POST
  - ✔ PUT
  - ✔ DELETE

Assertion result

```
Assertion error: false
Assertion failure: true
Assertion failure message: Test failed: text expected to contain /userId1/
```

Text

- GET
- Response Assertion
- POST
- PUT
- DELETE

Assertion result

Assertion error: false  
Assertion failure: true  
Assertion failure message: Test failed: code expected to equal /

\*\*\*\*\* received : [[[2]]]00  
\*\*\*\*\* comparison: [[[3]]]00

main sample and sub-samples     main sample only     sub-samples only     JMeter variable name to use

Field to Test

- Text Response
- Response Code
- Response Message
- Response Headers
- Request Headers
- URL Sampled
- Document (text)
- Ignore Status
- Request Data

Pattern Matching Rules

- Contains
- Matches
- Equals
- Substring
- Not
- Or

Patterns to Test

Patterns to Test	
1	300

Простой Test. Assertions - проверки

# Домашнее задание

Добавить 5 запросов для <https://api.nasa.gov/> запросов с assertions. Сохранить Test Plan, файл будем иметь расширение .jmx

Читать и практиковать:

[JMeter. Основные элементы.](#)

[Введение в JMeter](#)

[Приручаем JMeter](#)