# Testing your REST Server with Apache JMeter

By Henry Chan
June, 2015
hchan@apache.org
Download: https://github.com/hchan/jmeterDemo

## What is JMeter good for?

- http://jmeter.apache.org/
- **What can I do with it?**

- Apache JMeter may be used to test performance both on static
- and dynamic resources (Files, Web dynamic languages - PHP, Java,
- ASP.NET, etc. -, Java Objects, Data Bases and Queries, FTP Servers
- and more).
- It can be used to simulate a heavy load on a server, group of servers,
- network or object to test its strength or to analyze overall performance
- under different load types. You can use it to make a graphical analysis
- of performance or to test your server/script/object behavior under heavy concurrent load.

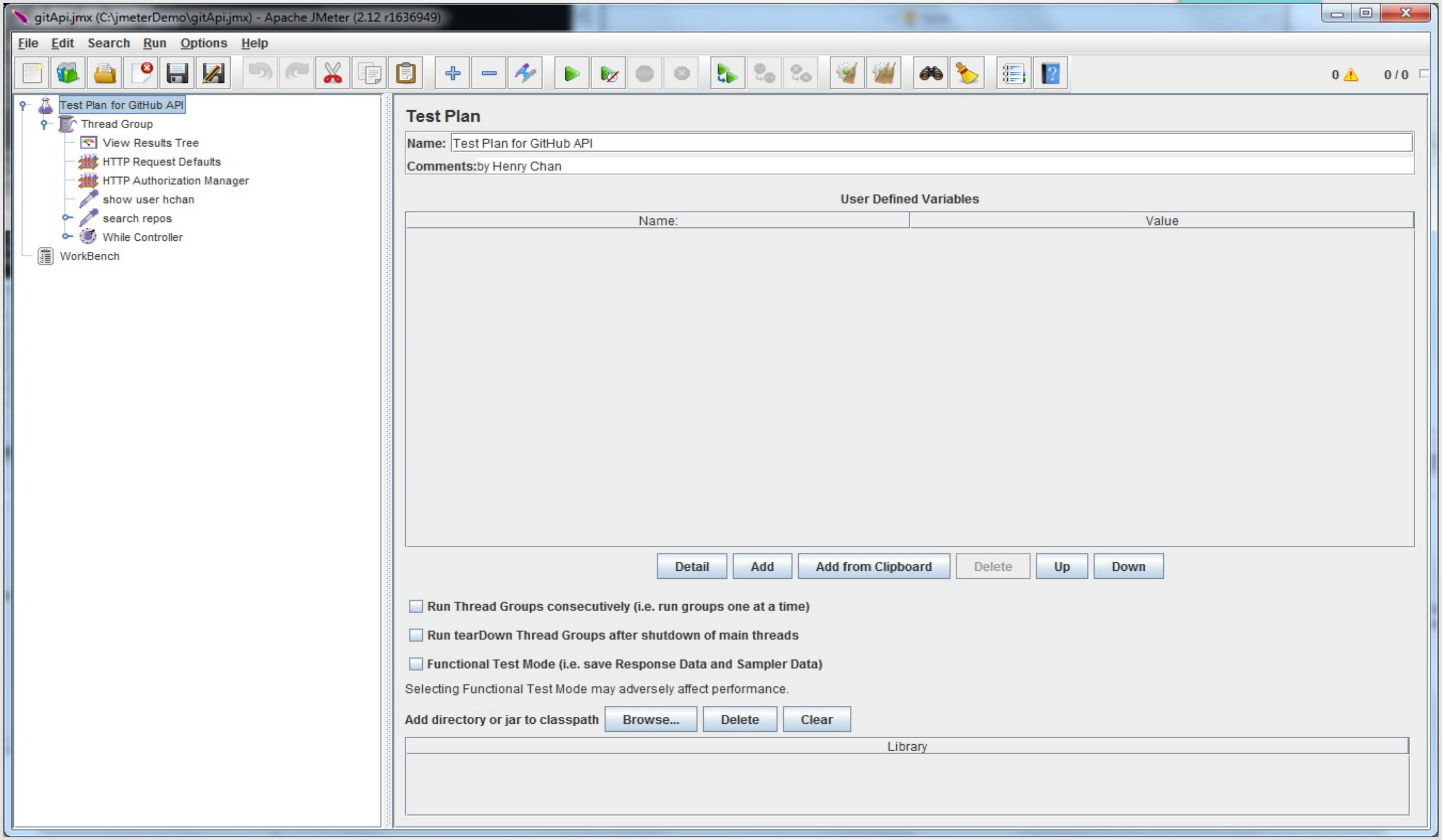
## Why use JMeter to test your REST Server?

- Two parts to this question
- a) Why test your REST Server?
  - To make sure it is up like a ping and check validity of response.
  - Can be wrapped in a cronjob (with the 'headless') option (-n)
  - Stress testing
- b) Why JMeter when there are so many other tools?
  - Headless mode
  - can do reports
  - Stress testing
  - Open Source
  - Other goodies besides REST (i.e. DBSampler, Java Sampler)
  - Comes with a Drag and Drop GUI/IDE to help create TestCases
    - And saves the final script as an XML (.jmx)

## Example – Let's test the GitHub API

- The following JMeter script https://github.com/hchan/jmeterDemo/ ->gitApi.jmx
- will consume a few REST Services from GitHub
- https://api.github.com
- It will parse the JSON request (to get all the projects in GITHub projects with
- the search string 'java' and iterate through each item) response to
- 'visit' the owner of that project's home_url

## JMeter comes with a GUI/IDE


gitApi.jmx (C:\jmeterDemo\gitApi.jmx) - Apache JMeter (2.12 r1636949)

File Edit Search Run Options Help

- Test Plan for GitHub API
  - Thread Group
    - View Results Tree
    - HTTP Request Defaults
    - HTTP Authorization Manager
    - show user hchan
    - search repos
    - While Controller
- WorkBench

**Test Plan**

Name: Test Plan for GitHub API

Comments: by Henry Chan

User Defined Variables

| Name: | Value |
|---|---|

Detail | Add | Add from Clipboard | Delete | Up | Down

☐ Run Thread Groups consecutively (i.e. run groups one at a time)

☐ Run tearDown Thread Groups after shutdown of main threads

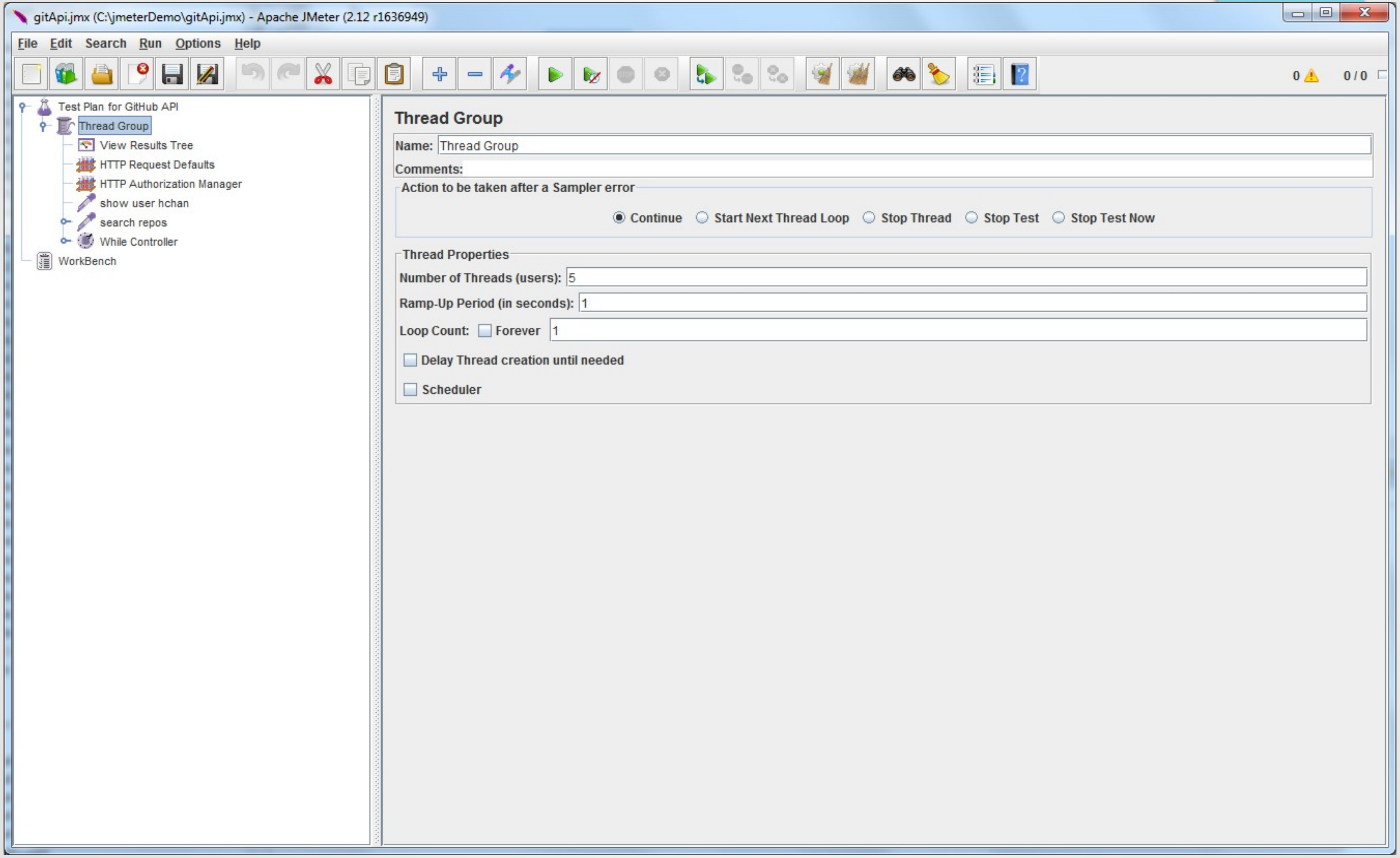
☐ Functional Test Mode (i.e. save Response Data and Sampler Data)

Selecting Functional Test Mode may adversely affect performance.

Add directory or jar to classpath Browse... Delete Clear

| Library |
|---|

## Can create Thread Groups for load testing


gitApi.jmx (C:\jmeterDemo\gitApi.jmx) - Apache JMeter (2.12 r1636949)

Thread Group

Name: Thread Group

Comments:

Action to be taken after a Sampler error

[✓ Continue] Start Next Thread Loop Stop Thread Stop Test Stop Test Now

Thread Properties

Number of Threads (users): 5

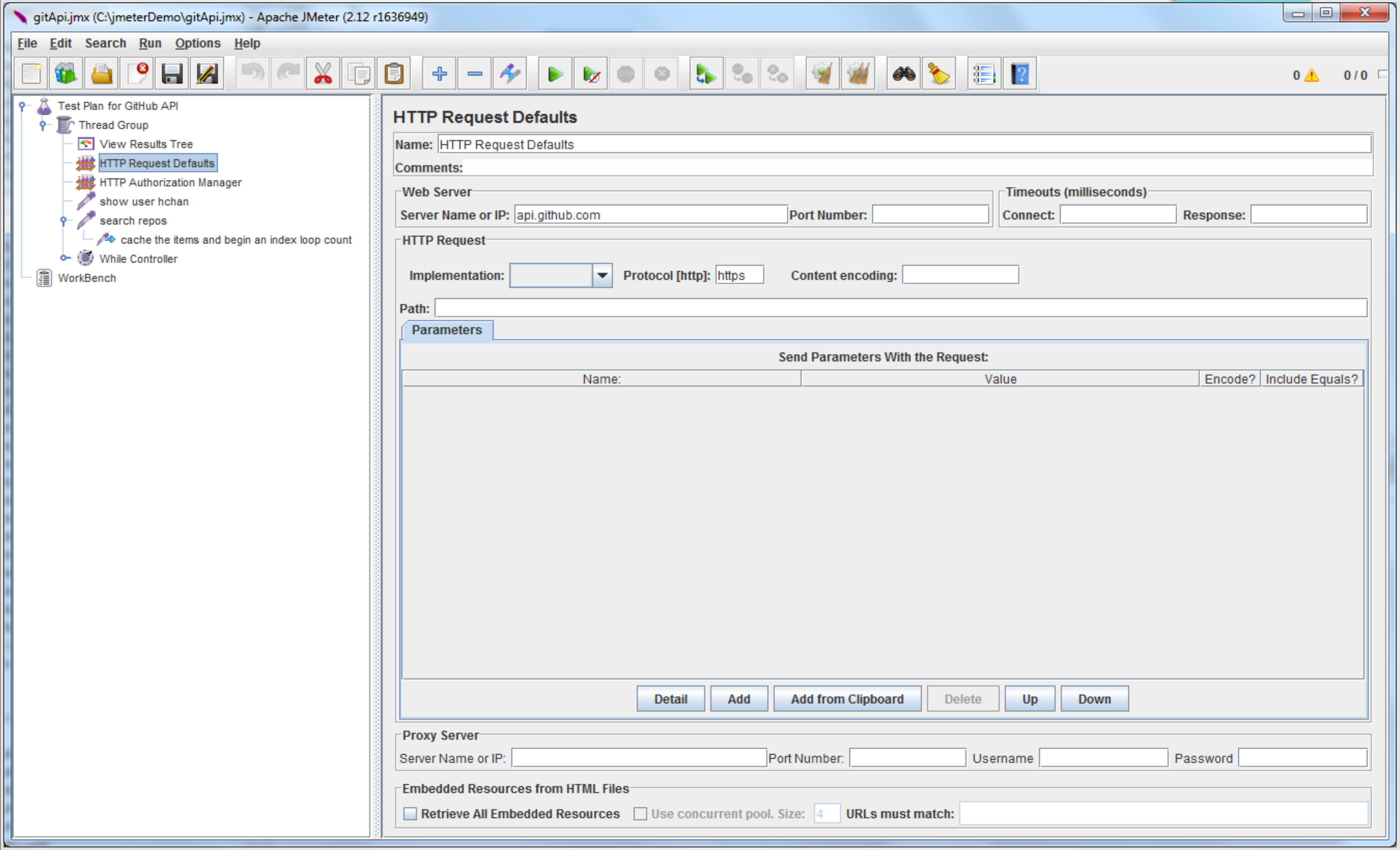
Ramp-Up Period (in seconds): 1

Loop Count: ☐ Forever 1

☐ Delay Thread creation until needed

☐ Scheduler

## Define your HTTP Request Defaults



gitApi.jmx (C:\jmeterDemo\gitApi.jmx) - Apache JMeter (2.12 r1636949)

- Test Plan for GitHub API
  - Thread Group
    - View Results Tree
    - HTTP Request Defaults
    - HTTP Authorization Manager
    - show user hchan
    - search repos
      - cache the items and begin an index loop count
    - While Controller
- WorkBench

**HTTP Request Defaults**

Name: HTTP Request Defaults

Comments:

Web Server

Server Name or IP: api.github.com

Port Number:

Timeouts (milliseconds)

Connect:

Response:

HTTP Request

Implementation:

Protocol [http]: https

Content encoding:

Path:

Parameters

Send Parameters With the Request:

| Name: | Value | Encode? | Include Equals? |
|---|---|---|---|

Proxy Server

Server Name or IP:

Port Number:

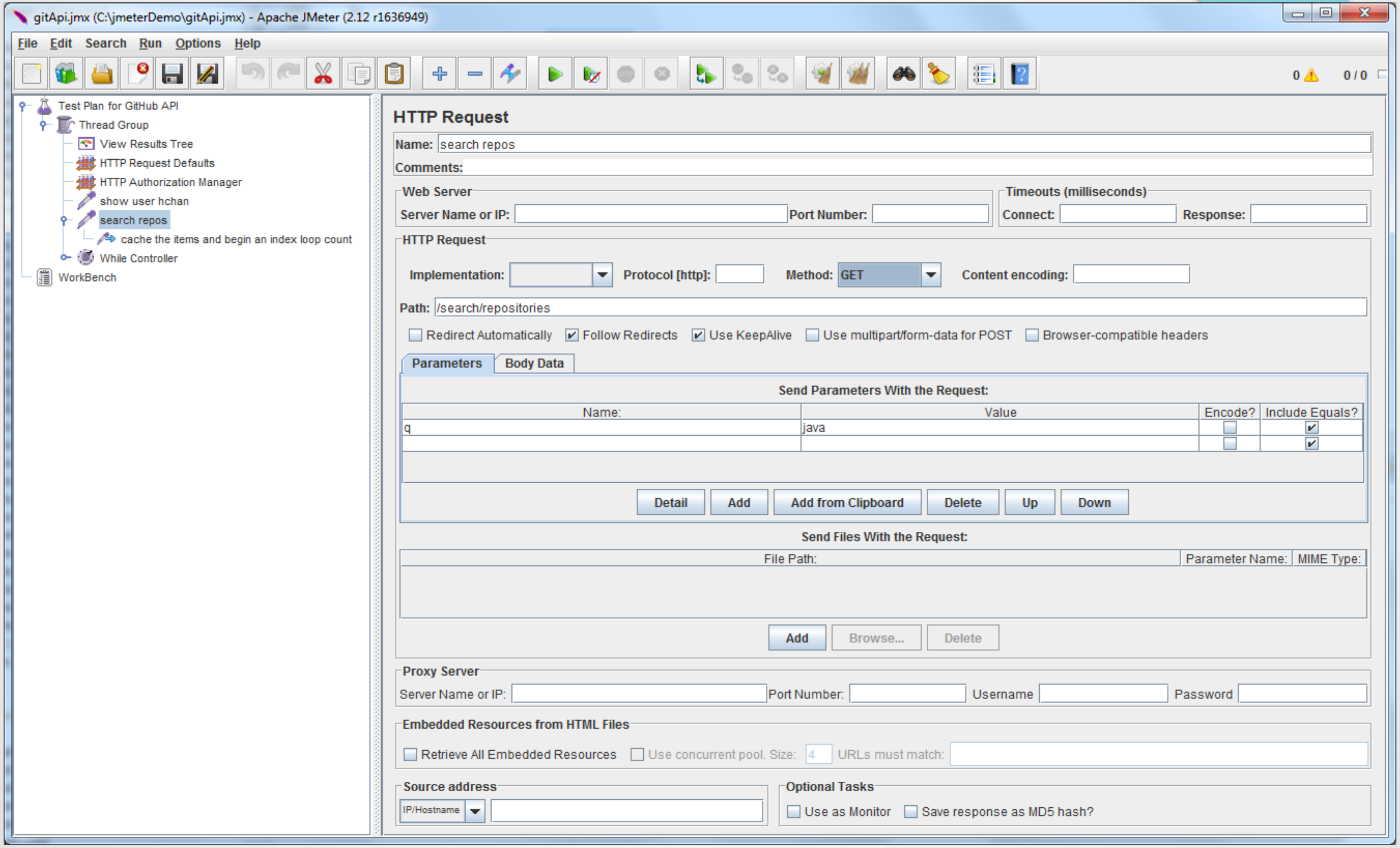
Username

Password

Embedded Resources from HTML Files

☐ Retrieve All Embedded Resources ☐ Use concurrent pool. Size: 4 URLs must match:

# Create your Test Cases (GET, PUT, DELETE, POST)



gitApi.jmx (C:\jmeterDemo\gitApi.jmx) - Apache JMeter (2.12 r1636949)

- Test Plan for GitHub API
  - Thread Group
    - View Results Tree
    - HTTP Request Defaults
    - HTTP Authorization Manager
    - show user hchan
    - search repos
      - cache the items and begin an index loop count
    - While Controller
- WorkBench

**HTTP Request**

Name: search repos

Comments:

Web Server — Server Name or IP: | Port Number:

Timeouts (milliseconds) — Connect: | Response:

HTTP Request — Implementation: | Protocol [http]: | Method: GET | Content encoding:

Path: /search/repositories

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

Parameters | Body Data

Send Parameters With the Request:

| Name: | Value | Encode? | Include Equals? |
|---|---|---|---|
| q | java | ☐ | ☑ |
|  |  | ☐ | ☑ |

Send Files With the Request:

| File Path: | Parameter Name: | MIME Type: |
|---|---|---|

Proxy Server — Server Name or IP: | Port Number: | Username | Password

Embedded Resources from HTML Files — ☐ Retrieve All Embedded Resources ☐ Use concurrent pool. Size: 4 URLs must match:

Source address — IP/Hostname

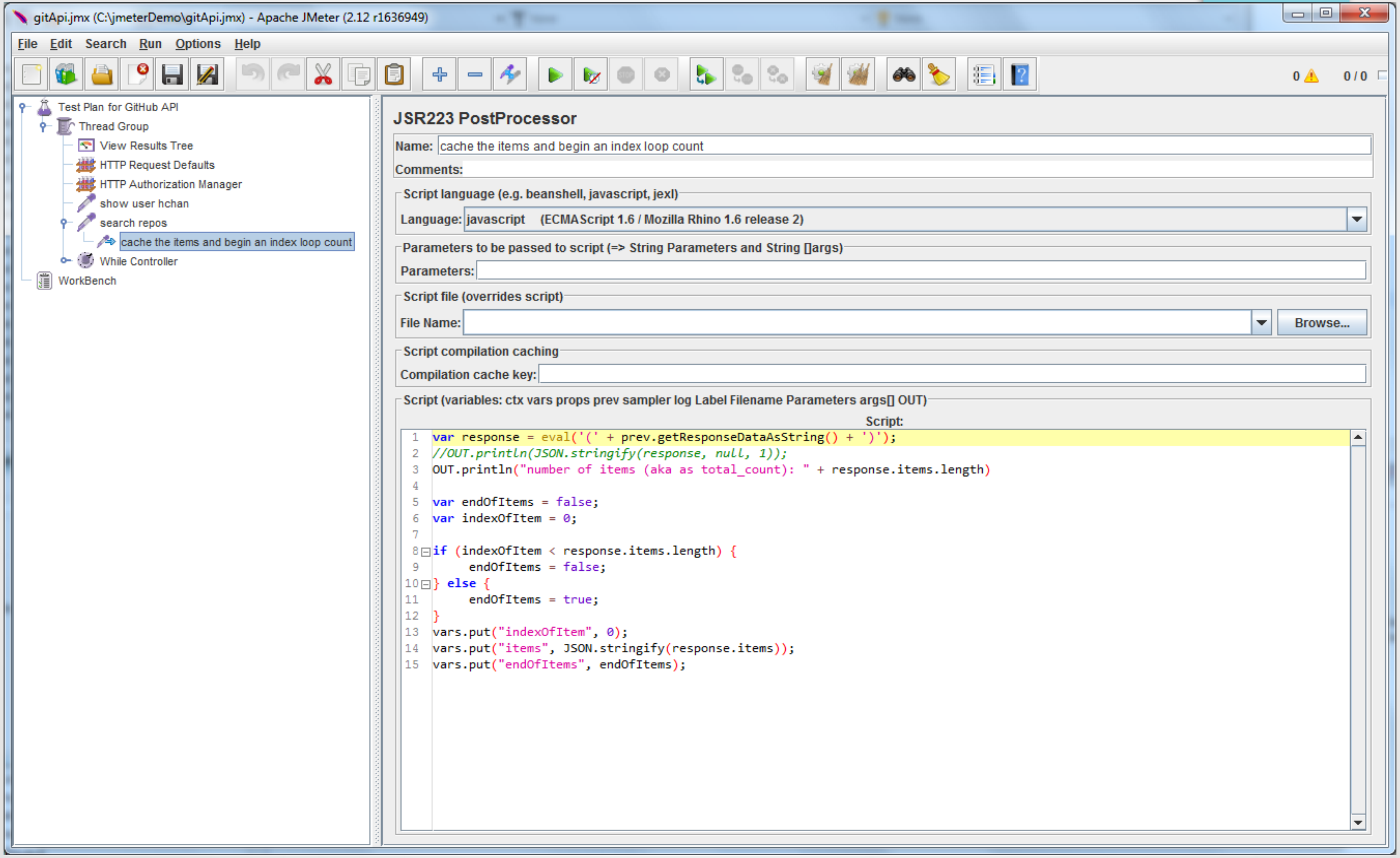
Optional Tasks — ☐ Use as Monitor ☐ Save response as MD5 hash?

## Assertions and PostProcessors

* Assertions help verify that your server under test returns the expected results

* PostProcessor – similar to assertion, but isn't necessary used to verify the response message.  JMeter PostProcessors can be written in various languages, but with testcases involving REST responses (JSON), I recommend Javascript

# Javascript PostProcessor



gitApi.jmx (C:\jmeterDemo\gitApi.jmx) - Apache JMeter (2.12 r1636949)

**JSR223 PostProcessor**

Name: cache the items and begin an index loop count

Comments:

Script language (e.g. beanshell, javascript, jexl)

Language: javascript (ECMAScript 1.6 / Mozilla Rhino 1.6 release 2)

Parameters to be passed to script (=> String Parameters and String []args)

Parameters:

Script file (overrides script)

File Name:

Script compilation caching

Compilation cache key:

Script (variables: ctx vars props prev sampler log Label Filename Parameters args[] OUT)

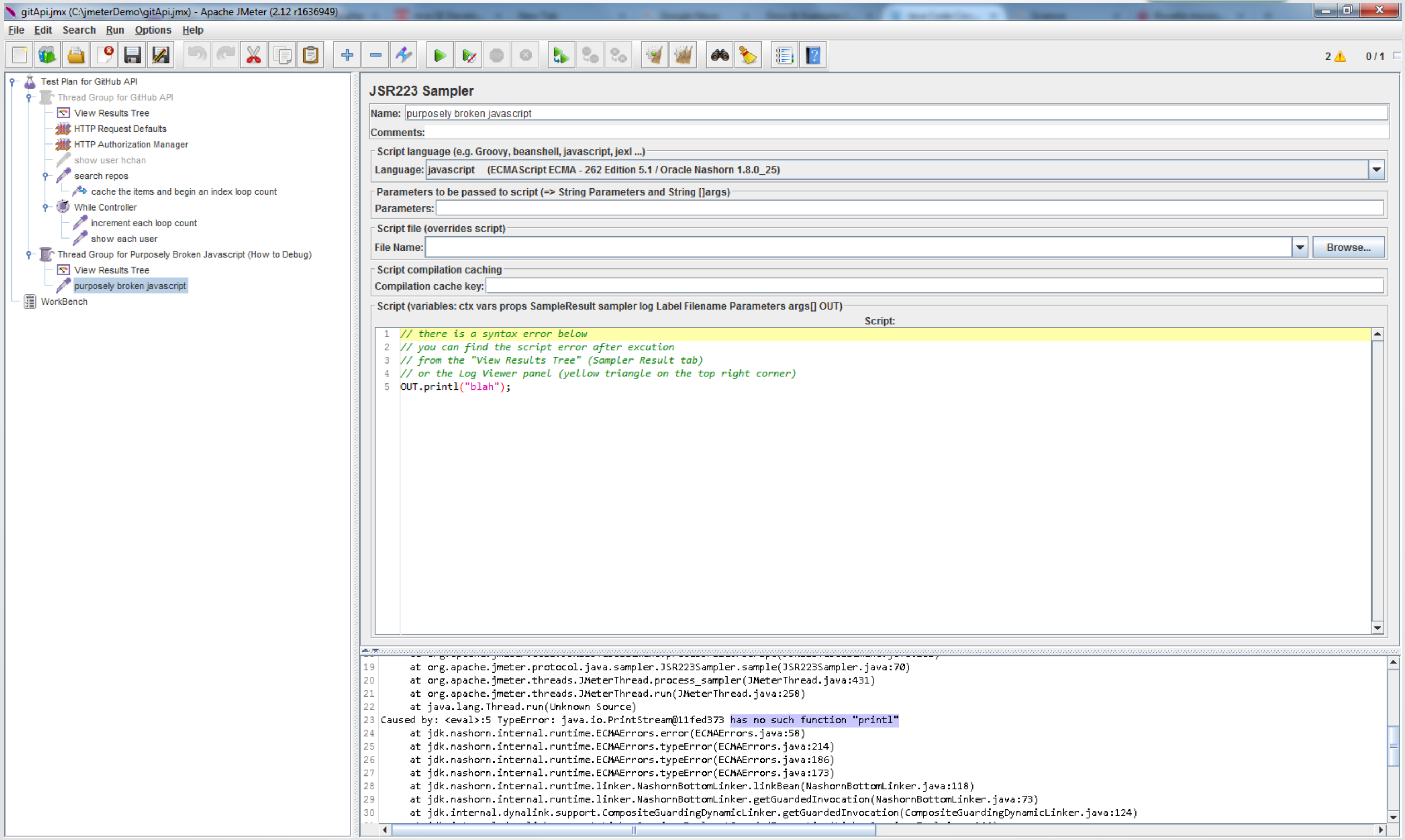
Script:

```
var response = eval('(' + prev.getResponseDataAsString() + ')');
//OUT.println(JSON.stringify(response, null, 1));
OUT.println("number of items (aka as total_count): " + response.items.length)

var endOfItems = false;
var indexOfItem = 0;

if (indexOfItem < response.items.length) {
     endOfItems = false;
} else {
     endOfItems = true;
}
vars.put("indexOfItem", 0);
vars.put("items", JSON.stringify(response.items));
vars.put("endOfItems", endOfItems);
```

## Saving variables in PostProcessors

```
// Script (variables “vars” is global)
// Other global variables include OUT, props, ctx

vars.put("indexOfItem", 0);
// FYI, I recommend you use Java8 – Nashhorn rocks!
// the JSON class is a bit … in JDK's < 8
vars.put("items", JSON.stringify(response.items));
vars.put("endOfItems", endOfItems);
```

# Debugging Scipts


gitApi.jmx (C:\jmeterDemo\gitApi.jmx) - Apache JMeter (2.12 r1636949)

File Edit Search Run Options Help
2 ⚠ 0 / 1
Test Plan for GitHub API
Thread Group for GitHub API
View Results Tree
HTTP Request Defaults
HTTP Authorization Manager
show user hchan
search repos
cache the items and begin an index loop count
While Controller
increment each loop count
show each user
Thread Group for Purposely Broken Javascript (How to Debug)
View Results Tree
purposely broken javascript
WorkBench
Browse...

**JSR223 Sampler**

Name: purposely broken javascript

Comments:

Script language (e.g. Groovy, beanshell, javascript, jexl ...)

Language: javascript (ECMAScript ECMA - 262 Edition 5.1 / Oracle Nashorn 1.8.0_25)

Parameters to be passed to script (=> String Parameters and String []args)

Parameters:

Script file (overrides script)

File Name:

Script compilation caching

Compilation cache key:

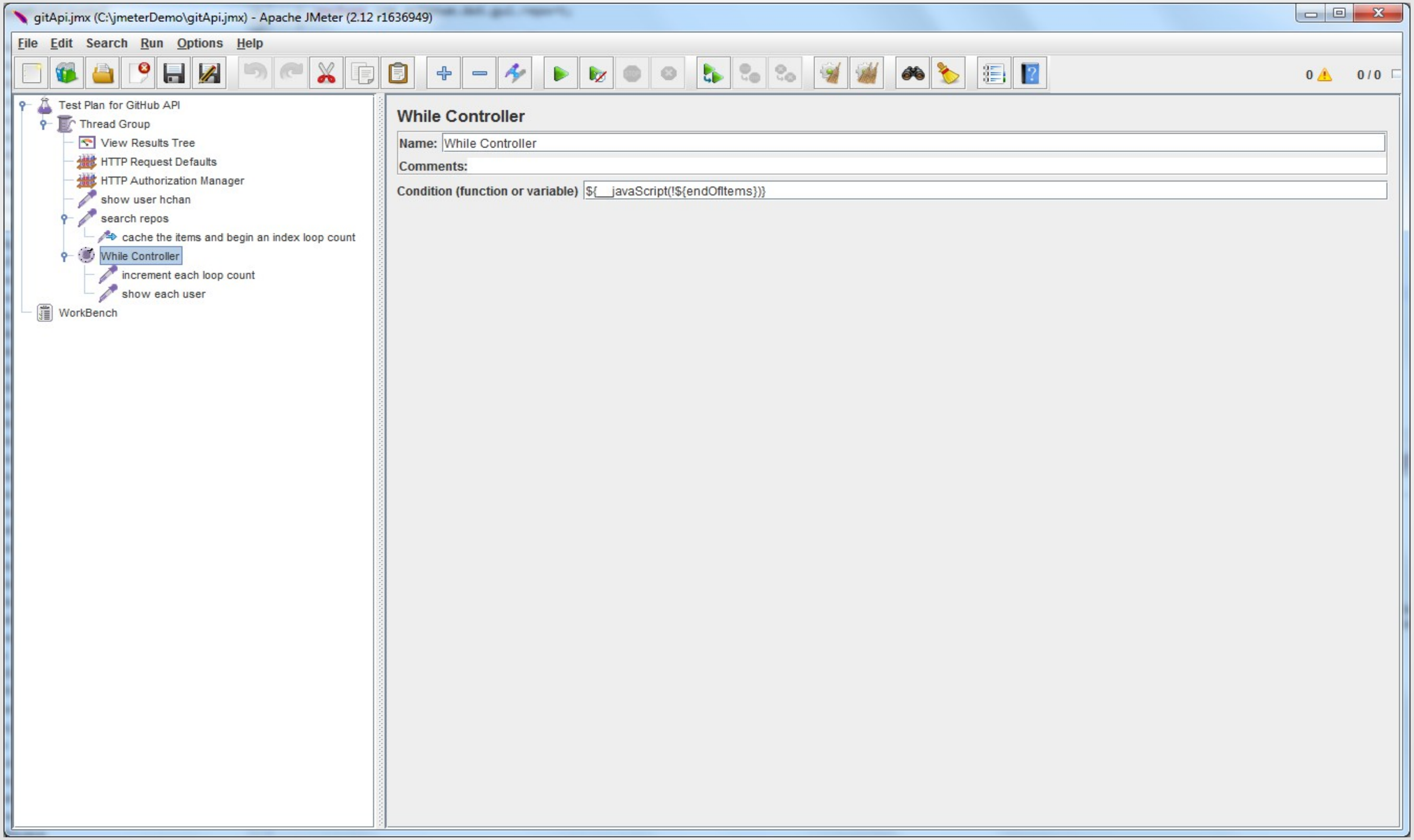
Script (variables: ctx vars props SampleResult sampler log Label Filename Parameters args[] OUT)

Script:

```
// there is a syntax error below
// you can find the script error after excution
// from the "View Results Tree" (Sampler Result tab)
// or the Log Viewer panel (yellow triangle on the top right corner)
OUT.printl("blah");
```

```
    at org.apache.jmeter.protocol.java.sampler.JSR223Sampler.sample(JSR223Sampler.java:70)
    at org.apache.jmeter.threads.JMeterThread.process_sampler(JMeterThread.java:431)
    at org.apache.jmeter.threads.JMeterThread.run(JMeterThread.java:258)
    at java.lang.Thread.run(Unknown Source)
Caused by: <eval>:5 TypeError: java.io.PrintStream@11fed373 has no such function "printl"
    at jdk.nashorn.internal.runtime.ECMAErrors.error(ECMAErrors.java:58)
    at jdk.nashorn.internal.runtime.ECMAErrors.typeError(ECMAErrors.java:214)
    at jdk.nashorn.internal.runtime.ECMAErrors.typeError(ECMAErrors.java:186)
    at jdk.nashorn.internal.runtime.ECMAErrors.typeError(ECMAErrors.java:173)
    at jdk.nashorn.internal.runtime.linker.NashornBottomLinker.linkBean(NashornBottomLinker.java:118)
    at jdk.nashorn.internal.runtime.linker.NashornBottomLinker.getGuardedInvocation(NashornBottomLinker.java:73)
    at jdk.internal.dynalink.support.CompositeGuardingDynamicLinker.getGuardedInvocation(CompositeGuardingDynamicLinker.java:124)
```

## Accessing variables … While loop


gitApi.jmx (C:\jmeterDemo\gitApi.jmx) - Apache JMeter (2.12 r1636949)
File Edit Search Run Options Help
0 0 / 0

Test Plan for GitHub API
- Thread Group
  - View Results Tree
  - HTTP Request Defaults
  - HTTP Authorization Manager
  - show user hchan
  - search repos
    - cache the items and begin an index loop count
  - While Controller
    - increment each loop count
    - show each user
WorkBench

**While Controller**

Name: While Controller

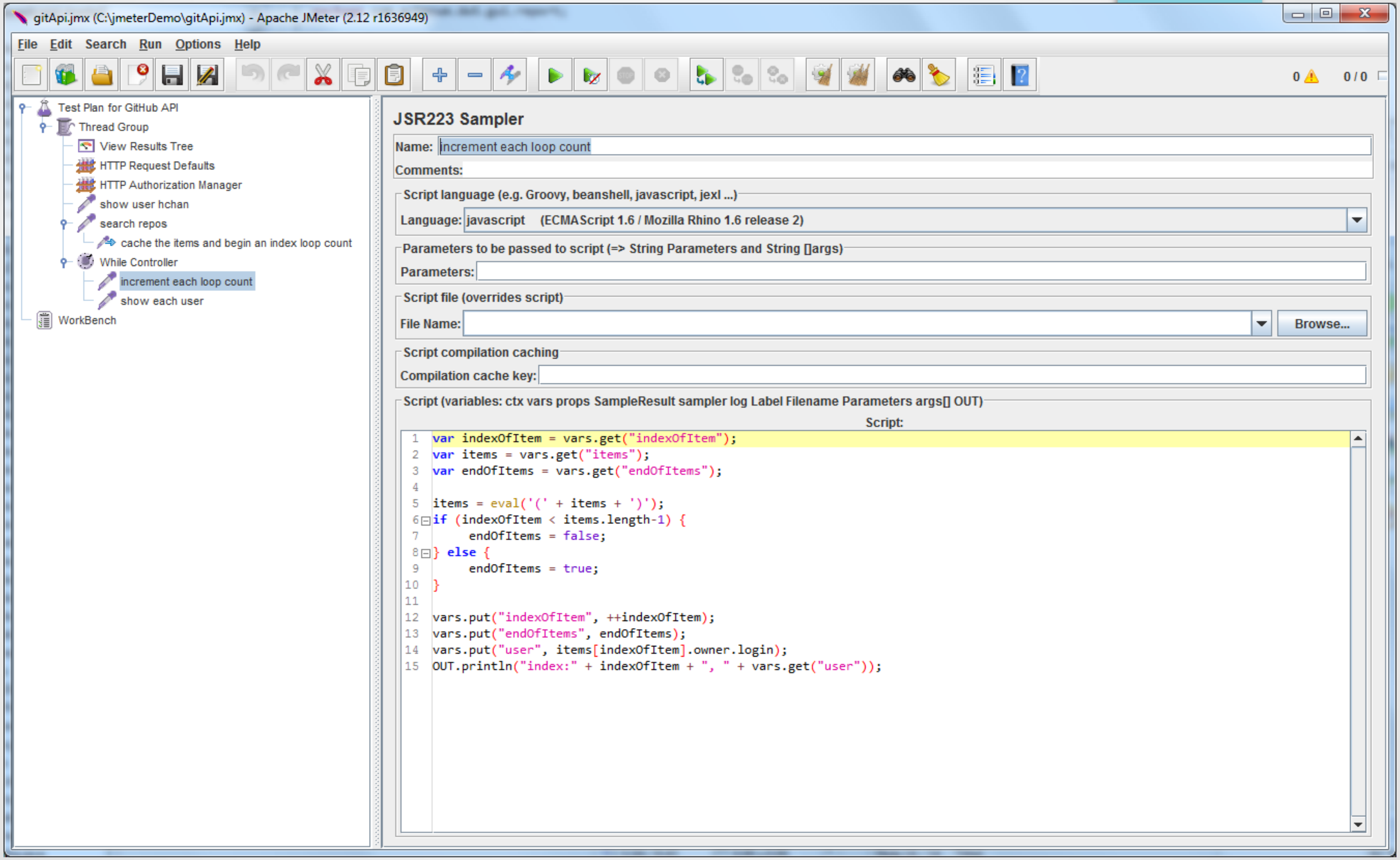
Comments:

Condition (function or variable) `${__javaScript(!${endOfItems})}`

## More scripting … samplers

Javascript Samplers are used create logic.
Although there are various other Jmeter components
To help with logging, creating custom made HTTP
Requests, accessing the DB, samplers give you
The full power.  Let's see a sampler that increments
An index in previous example's While Loop
${__javaScript(!${endOfItems})}

# Javascript Sampler


gitApi.jmx (C:\jmeterDemo\gitApi.jmx) - Apache JMeter (2.12 r1636949)
File Edit Search Run Options Help
Test Plan for GitHub API
Thread Group
View Results Tree
HTTP Request Defaults
HTTP Authorization Manager
show user hchan
search repos
cache the items and begin an index loop count
While Controller
increment each loop count
show each user
WorkBench

JSR223 Sampler

Name: increment each loop count

Comments:

Script language (e.g. Groovy, beanshell, javascript, jexl ...)

Language: javascript (ECMAScript 1.6 / Mozilla Rhino 1.6 release 2)

Parameters to be passed to script (=> String Parameters and String []args)

Parameters:

Script file (overrides script)

File Name:

Script compilation caching

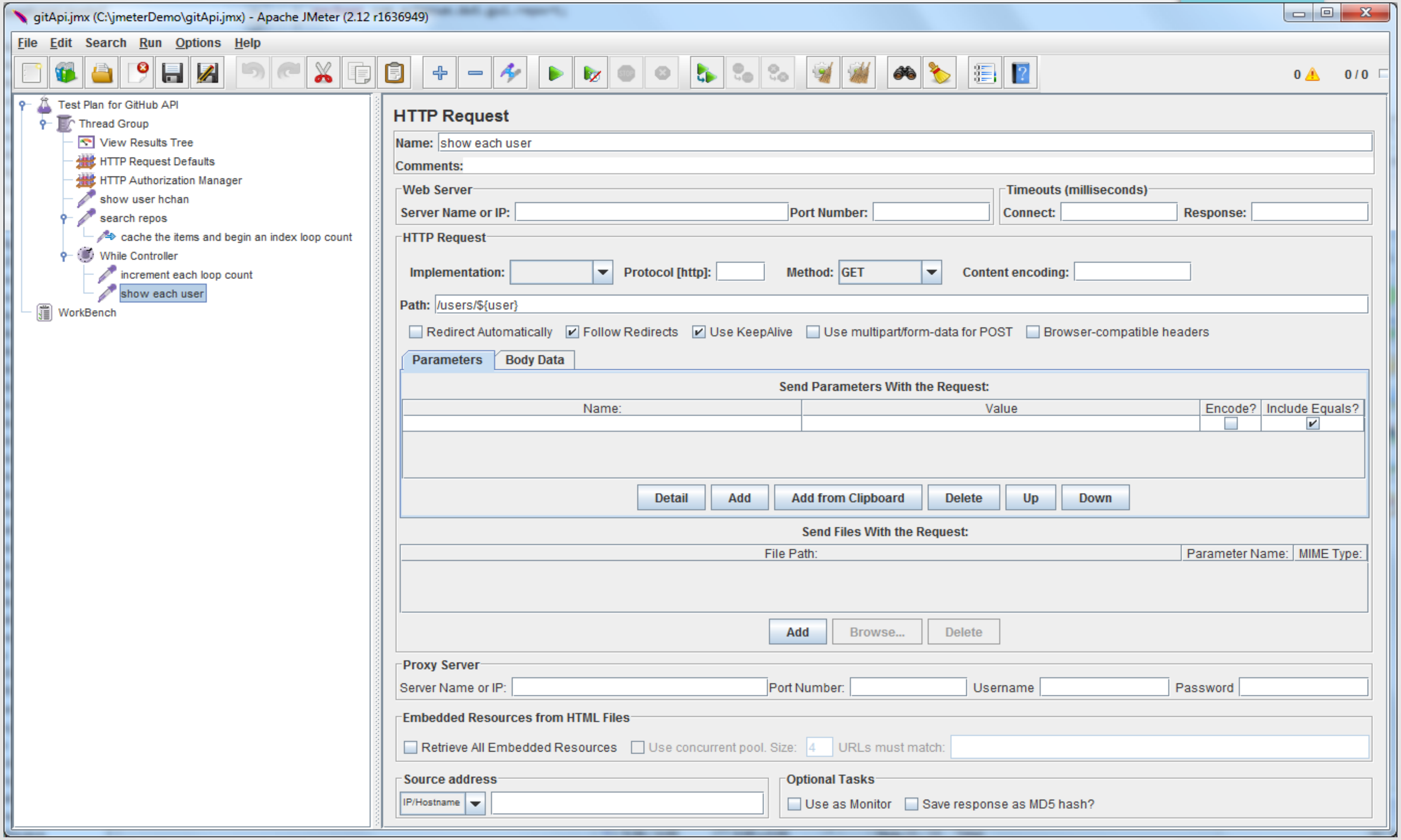
Compilation cache key:

Script (variables: ctx vars props SampleResult sampler log Label Filename Parameters args[] OUT)

Script:

```javascript
var indexOfItem = vars.get("indexOfItem");
var items = vars.get("items");
var endOfItems = vars.get("endOfItems");

items = eval('(' + items + ')');
if (indexOfItem < items.length-1) {
    endOfItems = false;
} else {
    endOfItems = true;
}

vars.put("indexOfItem", ++indexOfItem);
vars.put("endOfItems", endOfItems);
vars.put("user", items[indexOfItem].owner.login);
OUT.println("index:" + indexOfItem + ", " + vars.get("user"));
```

# HTTP Request ${variable}



gitApi.jmx (C:\jmeterDemo\gitApi.jmx) - Apache JMeter (2.12 r1636949)

## HTTP Request

Name: show each user

Comments:

Web Server — Server Name or IP: | Port Number:

Timeouts (milliseconds) — Connect: | Response:

HTTP Request — Implementation: | Protocol [http]: | Method: GET | Content encoding:

Path: /users/${user}

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

Parameters | Body Data

Send Parameters With the Request:

| Name: | Value | Encode? | Include Equals? |
|---|---|---|---|
| | | ☐ | ☑ |

Send Files With the Request:

| File Path: | Parameter Name: | MIME Type: |
|---|---|---|

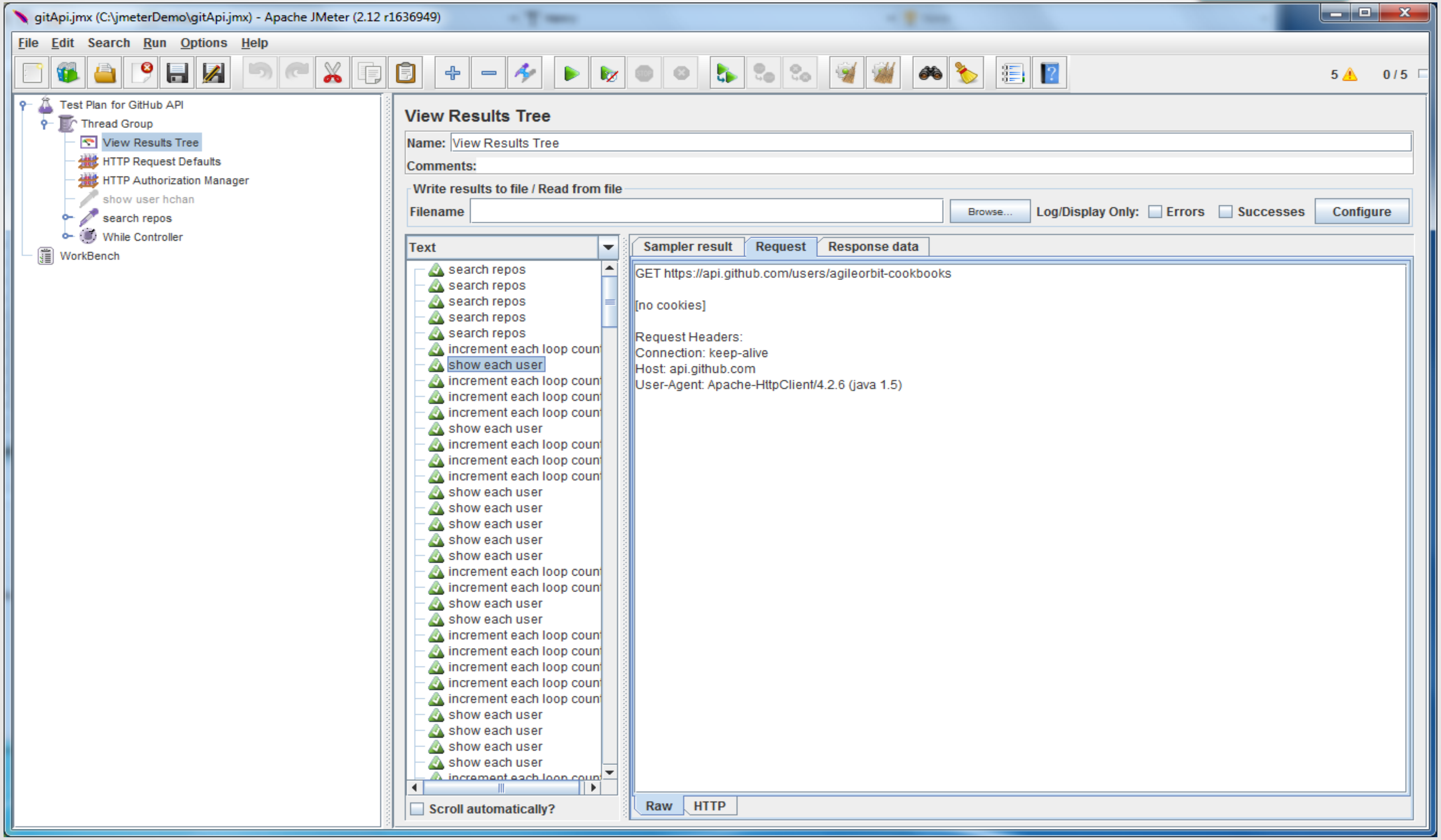
Proxy Server — Server Name or IP: | Port Number: | Username | Password

Embedded Resources from HTML Files — ☐ Retrieve All Embedded Resources ☐ Use concurrent pool. Size: 4 URLs must match:

Source address — IP/Hostname

Optional Tasks — ☐ Use as Monitor ☐ Save response as MD5 hash?

# Results Tree (request)



gitApi.jmx (C:\jmeterDemo\gitApi.jmx) - Apache JMeter (2.12 r1636949)

File Edit Search Run Options Help

- Test Plan for GitHub API
  - Thread Group
    - View Results Tree
    - HTTP Request Defaults
    - HTTP Authorization Manager
    - show user hchan
    - search repos
    - While Controller
- WorkBench

View Results Tree

Name: View Results Tree

Comments:

Write results to file / Read from file

Filename | Browse... | Log/Display Only: ☐ Errors ☐ Successes | Configure

Text

- search repos
- search repos
- search repos
- search repos
- search repos
- increment each loop coun
- show each user
- increment each loop coun
- increment each loop coun
- increment each loop coun
- show each user
- increment each loop coun
- increment each loop coun
- increment each loop coun
- show each user
- show each user
- show each user
- show each user
- show each user
- increment each loop coun
- increment each loop coun
- show each user
- show each user
- increment each loop coun
- increment each loop coun
- increment each loop coun
- increment each loop coun
- increment each loop coun
- show each user
- show each user
- show each user
- show each user

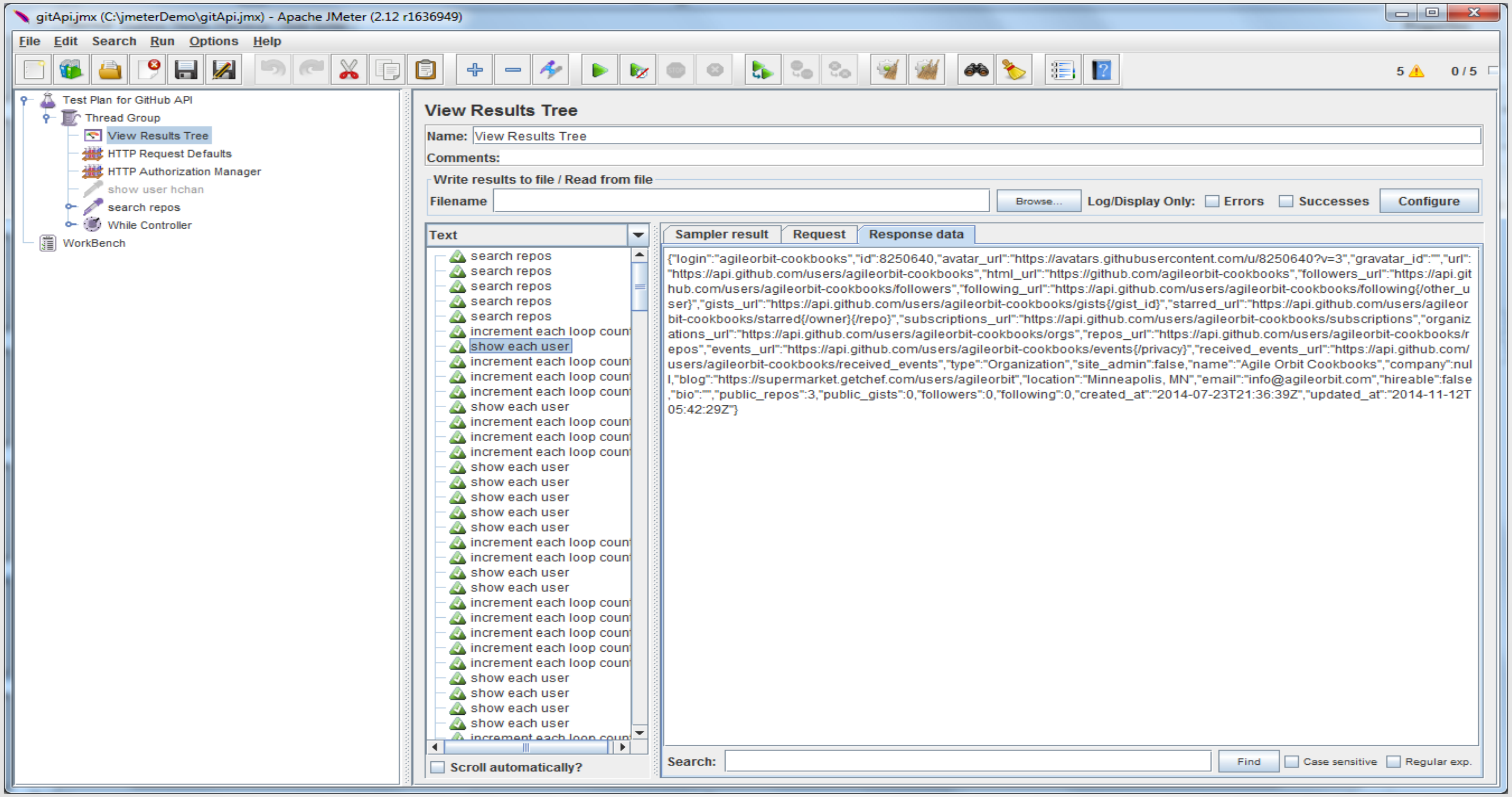
☐ Scroll automatically?

Sampler result | Request | Response data

```
GET https://api.github.com/users/agileorbit-cookbooks

[no cookies]

Request Headers:
Connection: keep-alive
Host: api.github.com
User-Agent: Apache-HttpClient/4.2.6 (java 1.5)
```

Raw | HTTP

# Results Tree (response)


gitApi.jmx (C:\jmeterDemo\gitApi.jmx) - Apache JMeter (2.12 r1636949)

File Edit Search Run Options Help

- Test Plan for GitHub API
  - Thread Group
    - View Results Tree
    - HTTP Request Defaults
    - HTTP Authorization Manager
    - show user hchan
    - search repos
    - While Controller
- WorkBench

View Results Tree

Name: View Results Tree

Comments:

Write results to file / Read from file

Filename

Log/Display Only: Errors Successes

Text

```
search repos
search repos
search repos
search repos
search repos
increment each loop coun
show each user
increment each loop coun
increment each loop coun
increment each loop coun
show each user
increment each loop coun
increment each loop coun
increment each loop coun
show each user
show each user
show each user
show each user
show each user
increment each loop coun
increment each loop coun
show each user
show each user
increment each loop coun
increment each loop coun
increment each loop coun
increment each loop coun
increment each loop coun
show each user
show each user
show each user
show each user
```

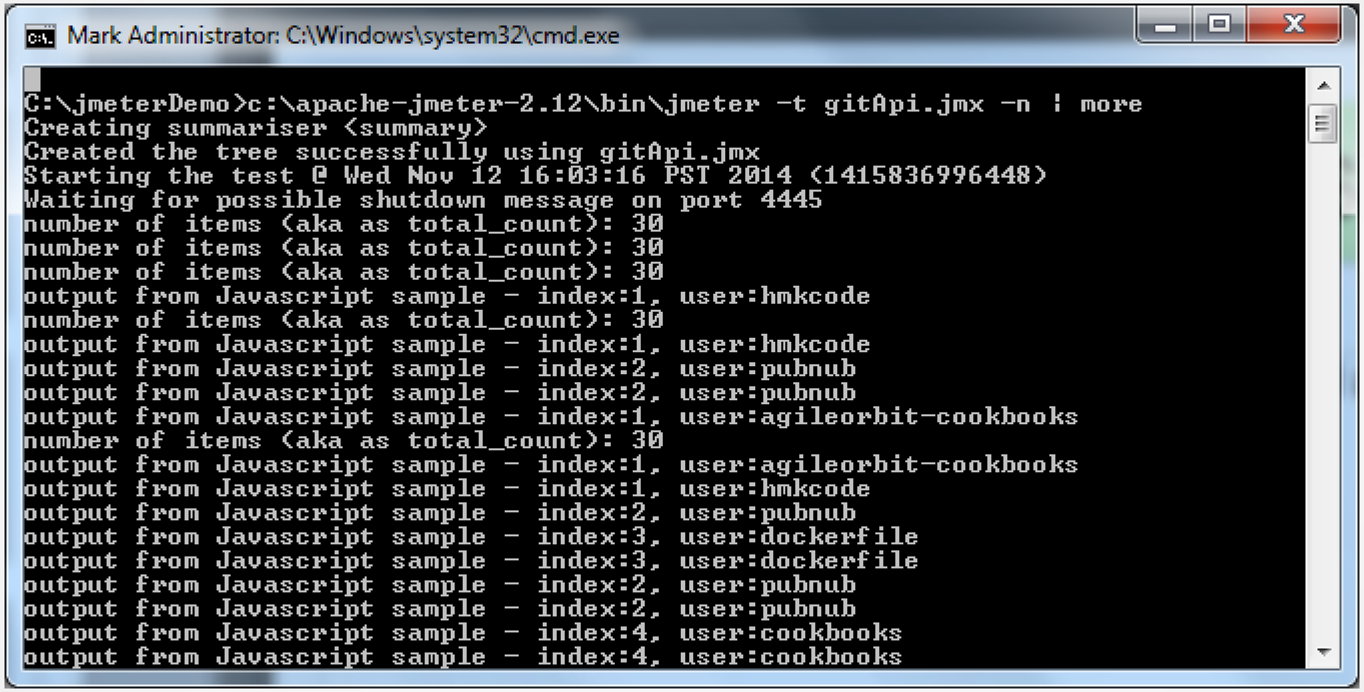
Sampler result | Request | Response data

```
{"login":"agileorbit-cookbooks","id":8250640,"avatar_url":"https://avatars.githubusercontent.com/u/8250640?v=3","gravatar_id":"","url":"https://api.github.com/users/agileorbit-cookbooks","html_url":"https://github.com/agileorbit-cookbooks","followers_url":"https://api.github.com/users/agileorbit-cookbooks/followers","following_url":"https://api.github.com/users/agileorbit-cookbooks/following{/other_user}","gists_url":"https://api.github.com/users/agileorbit-cookbooks/gists{/gist_id}","starred_url":"https://api.github.com/users/agileorbit-cookbooks/starred{/owner}{/repo}","subscriptions_url":"https://api.github.com/users/agileorbit-cookbooks/subscriptions","organizations_url":"https://api.github.com/users/agileorbit-cookbooks/orgs","repos_url":"https://api.github.com/users/agileorbit-cookbooks/repos","events_url":"https://api.github.com/users/agileorbit-cookbooks/events{/privacy}","received_events_url":"https://api.github.com/users/agileorbit-cookbooks/received_events","type":"Organization","site_admin":false,"name":"Agile Orbit Cookbooks","company":null,"blog":"https://supermarket.getchef.com/users/agileorbit","location":"Minneapolis, MN","email":"info@agileorbit.com","hireable":false,"bio":"","public_repos":3,"public_gists":0,"followers":0,"following":0,"created_at":"2014-07-23T21:36:39Z","updated_at":"2014-11-12T05:42:29Z"}
```

Scroll automatically?

Search: Case sensitive Regular exp.

## Running Headless Mode



```
C:\jmeterDemo>c:\apache-jmeter-2.12\bin\jmeter -t gitApi.jmx -n | more
Creating summariser <summary>
Created the tree successfully using gitApi.jmx
Starting the test @ Wed Nov 12 16:03:16 PST 2014 (1415836996448)
Waiting for possible shutdown message on port 4445
number of items (aka as total_count): 30
number of items (aka as total_count): 30
number of items (aka as total_count): 30
output from Javascript sample - index:1, user:hmkcode
number of items (aka as total_count): 30
output from Javascript sample - index:1, user:hmkcode
output from Javascript sample - index:2, user:pubnub
output from Javascript sample - index:2, user:pubnub
output from Javascript sample - index:1, user:agileorbit-cookbooks
number of items (aka as total_count): 30
output from Javascript sample - index:1, user:agileorbit-cookbooks
output from Javascript sample - index:1, user:hmkcode
output from Javascript sample - index:2, user:pubnub
output from Javascript sample - index:3, user:dockerfile
output from Javascript sample - index:3, user:dockerfile
output from Javascript sample - index:2, user:pubnub
output from Javascript sample - index:2, user:pubnub
output from Javascript sample - index:4, user:cookbooks
output from Javascript sample - index:4, user:cookbooks
```

## The End

Thanks
for
listening