

JMeter

1	도구 개요
2	도구 설치 방법
3	도구 기능 소개
4	도구 활용 예제
5	FAQ
6	도구 평가
7	용어집

1. 도구개요

목차

- 1. 도구 개요
- 2. 도구 설치 방법
 - 2.1 JMeter 다운 받기
 - 2.2 JMeter 설치 확인하기
- 3. 도구 기능 소개
 - 3.1 JMeter 주요 기능
 - 3.2 예제 소개
 - 3.3 웹 페이지 접속 가능 여부 확인 테스트 기능
 - 3.4 웹 페이지 부하 테스트 기능
 - 3.5 Response Assertion 테스트 기능
 - 3.6 Duration Assertion 테스트 기능
 - 3.7 Size Assertion 테스트 기능
- 4. 도구 활용
 - 4.1 예제 설명
 - 4.2 테스트 플랜 작성하기
 - 4.3 Thread Group 작성하기
 - 4.4 Http Request Defaults 작성하기
 - 4.5 Http Request 작성하기
 - 4.6 Duration Assertion 작성하기
 - 4.7 Size Assertion 작성하기
 - 4.8 Listener 추가하기
 - 4.9 테스트 실행하기
 - 4.10 테스트 결과 보기
- 5. FAQ
- 6. 도구 평가
- 7. 용어집

1. 도구 개요

소개	Apache JMeter는 ASF (Apache Software Foundation)에서 부하 테스트 및 성능 측정을 위해서 개발된 100% 순수 자바 애플리케이션 입니다. Jmeter는 HTTP를 사용하여 사이트의 부하를 테스트 할 수 있으므로 정적이거나 동적인 자원(파일, 서블릿, 자바 객체들, 데이터 베이스등)에 대해서 테스트하는데 사용할 수 있습니다.		
주요 기능	웹 페이지 부하 테스트 기능, 정적이거나 동적인 자원들에 대한 테스트 기능		
카테고리	Testing	세부 카테고리	테스팅 실행
커버리지	Testing	도구 난이도	하급
라이선스 형태 / 비용	Apache Software License / 무료	사전 설치 도구	JDK 1.3 이상 버전
사용환경	운영체제	OS Independent	
	Eclipse 환경	지원 안 함	
특징	<ul style="list-style-type: none">• Http, Ftp 서버를 로드하고 테스트 해볼 수 있을 뿐만 아니라, JDBC를 통해 임의의 데이터 베이스 쿼리를 테스트 할 수 있습니다.• 완벽한 Multi Threading 프레임워크를 자원해 다수의 Thread가 동시에 테스트 데이터를 추출할 수 있도록 해주며 다른 Thread 그룹들로 하여금 다른 기능들을 동시에 테스트하여 좀 더 정확한 시간 체크가 가능 합니다.• 통계 데이터의 개인화 뿐만 아니라 최대 확장을 허용합니다.		
적용 회사 / 프로젝트	-		
관련 도구	-		
제작사	ASF (Apache Software Foundation)		
공식 홈페이지	http://jakarta.apache.org/jmeter/index.html		
개발자	Stefano Mazzocchi		

2. 도구 설치 방법

2. 도구 설치 방법

세부 목차

- 2.1 JMeter 다운 받기
- 2.2 JMeter 설치 확인하기

2. 도구 설치 방법

2.1 JMeter 다운받기 (1/4)

- 다음의 사이트에서 JMeter를 다운 받습니다.
 - <http://jakarta.apache.org/>
- 왼쪽 Subprojects에서 JMeter를 선택합니다.



The Apache Jakarta Project
[http:// jakarta.apache.org/](http://jakarta.apache.org/)

Support

- [Downloads](#)
- [Subversion](#)
- [Mailing Lists](#)
- [Bug Database](#)
- [Jakarta Wiki](#)

Subprojects

- [BCEL](#)
- [BSF](#)
- [Cactus](#)
- [ECS](#)
- [JCS](#)
- **[JMeter](#)**
- [ORO](#)
- [Regexp](#)
- [Taglibs](#)

Ex-Jakarta

- [Ant](#)
- [Avalon](#)
- [Commons](#)
- [DB](#)
- [Excalibur](#)
- [Gump](#)
- [HiveMind](#)
- [HttpComponents](#)
- [James](#)
- [Logging](#)
- [Lucene](#)
- [Maven](#)
- [POI](#)
- [Portals](#)
- [Struts](#)
- [Tapestry](#)

Welcome to The Jakarta Project

The Jakarta Project offers a diverse set of open source Java solutions and is a part of [The Apache Software Foundation](#) (ASF) which encourages a collaborative, consensus-based development process under an open [software license](#).

Jakarta is organized into subprojects, summarized in the [section below](#). The ASF contains other Java projects, some of which were a part of Jakarta, that are detailed on the [ASF Project](#) page.

It is recommended that users of Jakarta products [subscribe](#) to the Jakarta [announcements mailing list](#).

News

Latest Jakarta News

- 14 June 2008 - [Apache JMeter 2.3.2 Released](#)
- 11 April 2008 - [Apache Cactus 1.8.0 Released](#)
- 30 November 2007 - [Apache JMeter 2.3.1 final Released](#)
- 15 November 2007 - [HttpComponents TLP move](#)
- 07 November 2007 - [HttpComponents HttpClient 4.0-alpha2 Released](#)
- 03 November 2007 - [Slide is retired](#)
- 09 October 2007 - [HttpComponents HttpCore 4.0-alpha6 Released](#)
- 29 September 2007 - [Apache JMeter 2.3 final Released](#)
- 05 September 2007 - [Apache JMeter 2.3RC4 Released](#)
- 22 August 2007 - [Jakarta Commons HttpClient 3.1 Released](#)
- 20 July 2007 - [HttpComponents HttpClient 4.0-alpha1 Released](#)
- 11 July 2007 - [Apache JMeter 2.3RC3 Released](#)
- 08 July 2007 - [Commons CLI 1.1 Released](#)
- 04 July 2007 - [HttpComponents HttpCore 4.0-alpha5 Released](#)
- 02 July 2007 - [Commons IO 1.3.2 Released](#)
- 18 June 2007 - [Jakarta Commons JCI 1.0 Released](#)
- 08 June 2007 - [POI TLP move](#)
- 07 June 2007 - [Turbine TLP move](#)
- 06 June 2007 - [Jakarta JCS 1.3 Released](#)

About Jakarta

- [Welcome](#)
- [News](#)
- [Contacting Us](#)
- [Who We Are](#)
- [Charter](#)
- [Quarterly Reports](#)
- [Retired Projects](#)
- [Jakarta Guidelines](#)

About Apache

- [ASF Website](#)
- [Sponsors of the ASF](#)
- [Sponsorship program](#)
- [How The ASF Works](#)
- [Developer Resources](#)
- [Get Involved](#)

Search Jakarta

Unaffiliated Links

- [Apache News](#)
- [FreeBSD Java Ports](#)
- [JPackage](#)
- [Planet Apache](#)

Unaffiliated Translations

- [Apache-Korea \(Korean\)](#)
- [Ja-Jakarta \(Japanese\)](#)
- [Jakarta.JP \(Japanese\)](#)

2. 도구 설치 방법

2.1 JMeter 다운받기 (2/4)

- 왼쪽 Download에서 Download Releases를 클릭합니다.





About

- [Overview](#)
- [Changes](#)
- [Known Bugs](#)
- [License](#)
- [Contributors](#)

Download

- [Download Releases](#)
- [Developer \(Nightly\) Builds](#)

Documentation

- [User Manual](#)
- [Javadocs](#)
- [Localisation \(Translator's Guide\)](#)
- [Building JMeter and Add-Ons](#)
- [JMeter Wiki](#)
- [FAQ \(Wiki\)](#)

Tutorials (PDF format)

- [Distributed Testing](#)
- [Recording Tests](#)
- [JUnit Sampler](#)
- [Access Log Sampler](#)
- [Extending JMeter](#)

Community

Apache JMeter

Apache JMeter is a 100% pure Java desktop application designed to load test functional behavior and measure performance. It was originally designed for testing Web Applications but has since expanded to other test functions.

What can I do with it?

Apache JMeter may be used to test performance both on static and dynamic resources (files, Servlets, Perl scripts, Java Objects, Data Bases and Queries, FTP Servers and more). It can be used to simulate a heavy load on a server, 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.

What does it do?

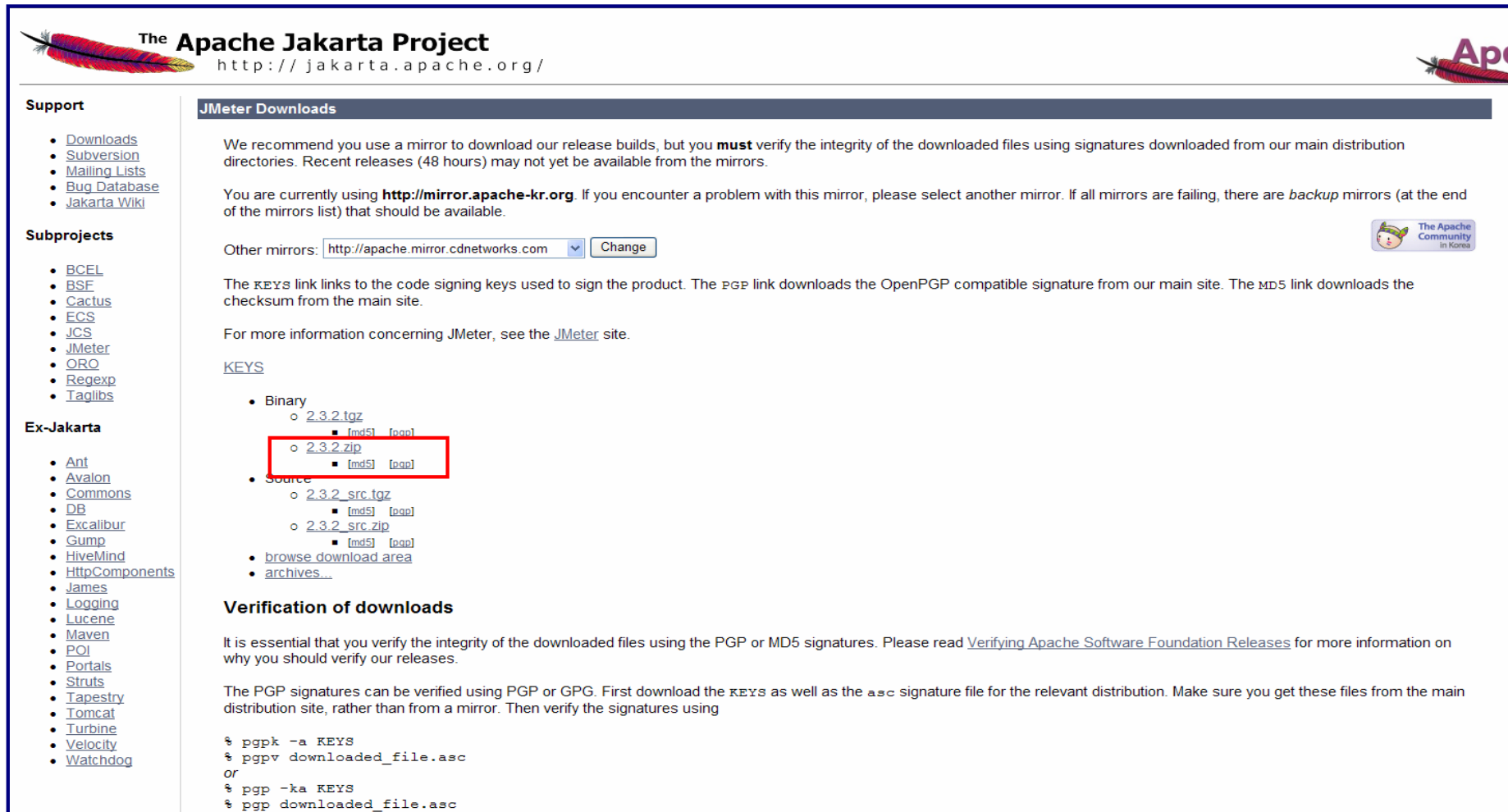
Apache JMeter features include:

- Can load and performance test many different server types:
 - Web - HTTP, HTTPS
 - SOAP
 - Database via JDBC
 - LDAP
 - JMS
 - Mail - POP3
- Complete portability and 100% Java purity .
- Full Swing and lightweight component support (precompiled JAR uses packages javax.swing.*).
- Full multithreading framework allows concurrent sampling by many threads and simultaneous sampling of different functions by separate thread groups.
- Careful GUI design allows faster operation and more precise timings.
- Caching and offline analysis/replaying of test results.

2. 도구 설치 방법

2.1 JMeter 다운받기 (3/4)

- Binary 부분에서 Windows용 JMeter를 다운 받습니다.
 - 본 매뉴얼에서는 Windows용 JMeter 2.3.2 버전을 로컬 디스크에 다운받도록 하겠습니다.



The Apache Jakarta Project
http://jakarta.apache.org/

Support

- [Downloads](#)
- [Subversion](#)
- [Mailing Lists](#)
- [Bug Database](#)
- [Jakarta Wiki](#)

Subprojects

- [BCEL](#)
- [BSF](#)
- [Cactus](#)
- [ECS](#)
- [JCS](#)
- [JMeter](#)
- [ORO](#)
- [Regexp](#)
- [Taglibs](#)

Ex-Jakarta

- [Ant](#)
- [Avalon](#)
- [Commons](#)
- [DB](#)
- [Excalibur](#)
- [Gump](#)
- [HiveMind](#)
- [HttpComponents](#)
- [James](#)
- [Logging](#)
- [Lucene](#)
- [Maven](#)
- [POI](#)
- [Portals](#)
- [Struts](#)
- [Tapestry](#)
- [Tomcat](#)
- [Turbine](#)
- [Velocity](#)
- [Watchdog](#)

JMeter Downloads

We recommend you use a mirror to download our release builds, but you **must** verify the integrity of the downloaded files using signatures downloaded from our main distribution directories. Recent releases (48 hours) may not yet be available from the mirrors.

You are currently using <http://mirror.apache-kr.org>. If you encounter a problem with this mirror, please select another mirror. If all mirrors are failing, there are *backup* mirrors (at the end of the mirrors list) that should be available.

Other mirrors:

The [keys](#) link links to the code signing keys used to sign the product. The [pgp](#) link downloads the OpenPGP compatible signature from our main site. The [md5](#) link downloads the checksum from the main site.

For more information concerning JMeter, see the [JMeter](#) site.

KEYS

- Binary
 - [2.3.2.tgz](#)
 - [\[md5\]](#) [\[pgp\]](#)
 - [2.3.2.zip](#)
 - [\[md5\]](#) [\[pgp\]](#)
- Source
 - [2.3.2_src.tgz](#)
 - [\[md5\]](#) [\[pgp\]](#)
 - [2.3.2_src.zip](#)
 - [\[md5\]](#) [\[pgp\]](#)
- [browse download area](#)
- [archives...](#)

Verification of downloads

It is essential that you verify the integrity of the downloaded files using the PGP or MD5 signatures. Please read [Verifying Apache Software Foundation Releases](#) for more information on why you should verify our releases.

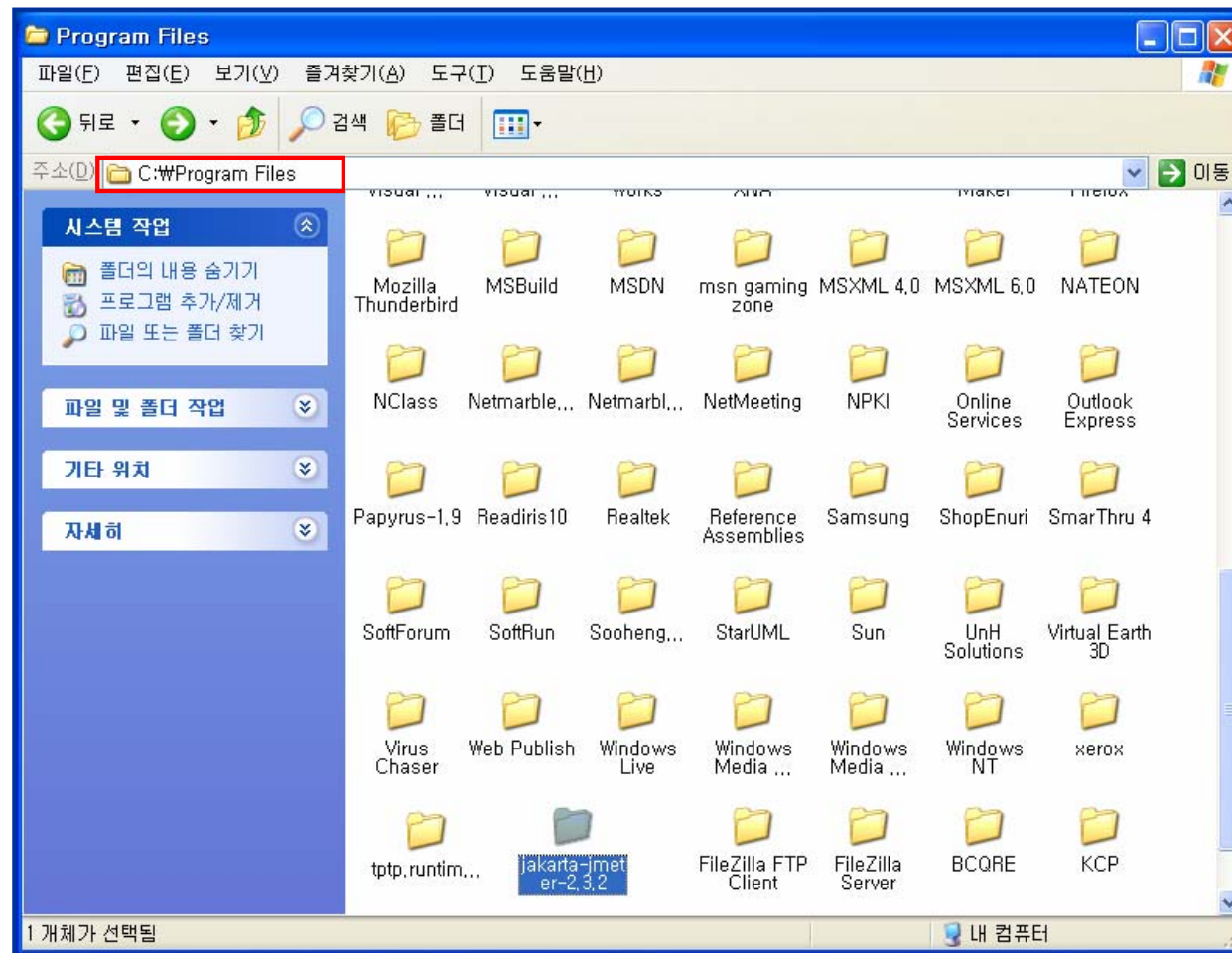
The PGP signatures can be verified using PGP or GPG. First download the [keys](#) as well as the [asc](#) signature file for the relevant distribution. Make sure you get these files from the main distribution site, rather than from a mirror. Then verify the signatures using

```
% pgpk -a KEYS
% pgpv downloaded_file.asc
or
% gpg -ka KEYS
% gpg downloaded_file.asc
```

2. 도구 설치 방법

2.1 JMeter 다운받기 (4/4)

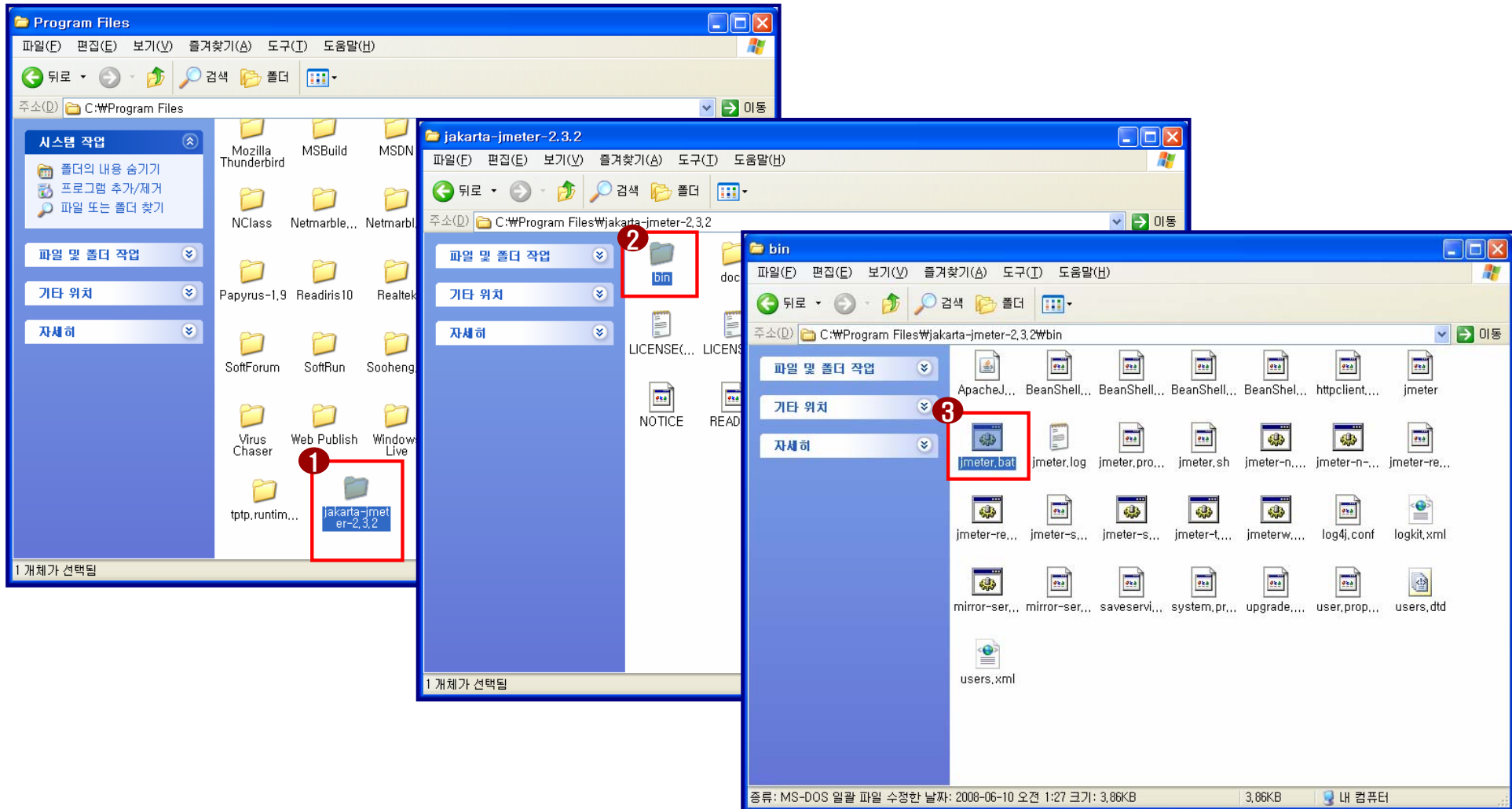
- 파일 다운로드 창에서 저장을 눌러 로컬디스크(C:)에 저장 합니다.
- 다운 받은 jakarta-jmeter-2.3.2 파일의 압축을 풉니다.
 - 본 매뉴얼은 Program Files 폴더 안에 압축을 해제하였습니다.



2. 도구 설치 방법

2.2 JMeter 설치 확인하기 (1/2)

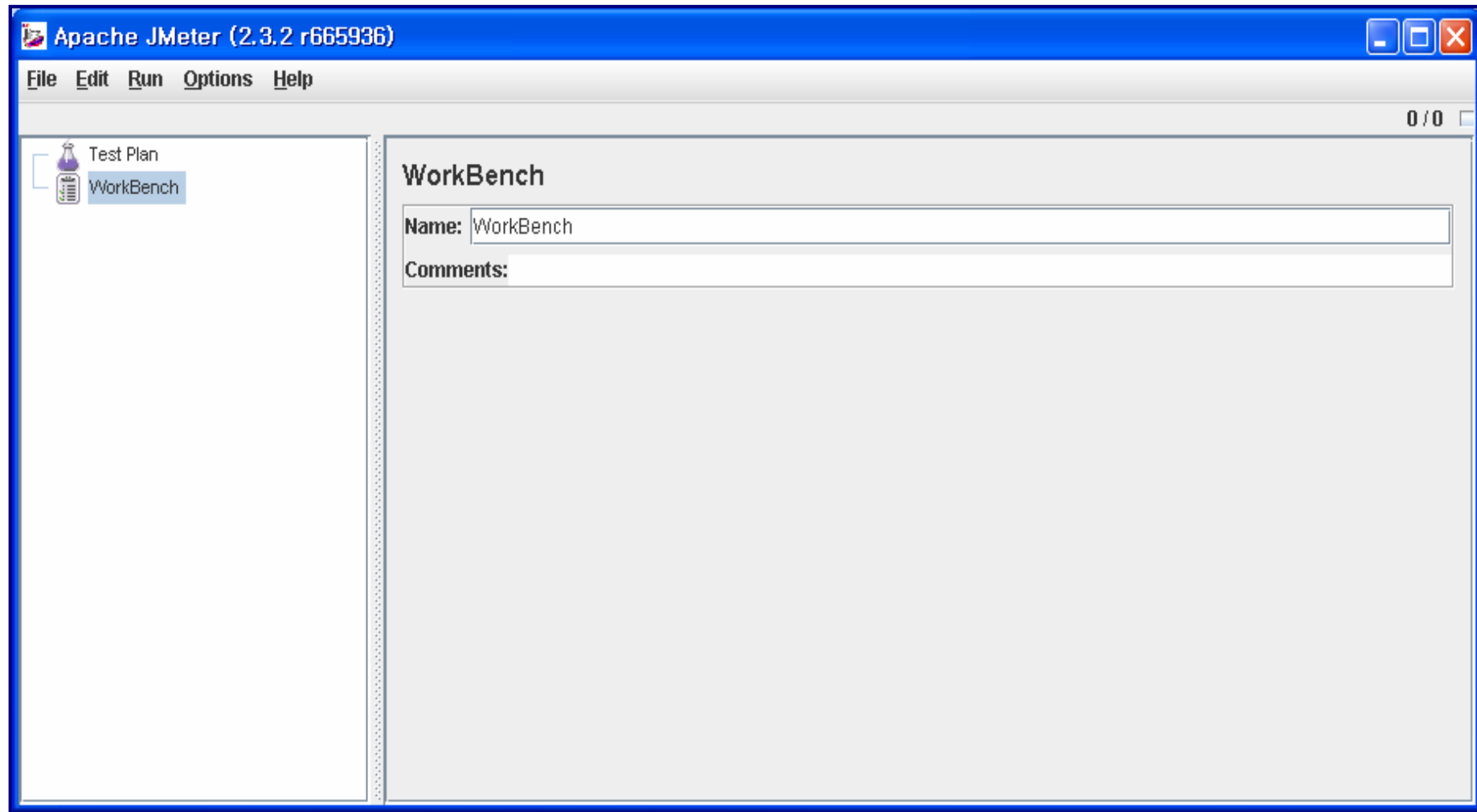
- JMeter를 실행합니다.
 - jakarta-jmeter-2.3.2 폴더에서 bin 폴더로 이동합니다.
 - jmeter.bat 파일을 클릭합니다.



2. 도구 설치 방법

2.2 JMeter 설치 확인하기 (2/2)

- jmeter.bat 파일을 클릭하면 다음과 같이 JMeter의 메인 화면이 보이게 됩니다.



3. 도구 기능 소개

3. 도구 기능 소개

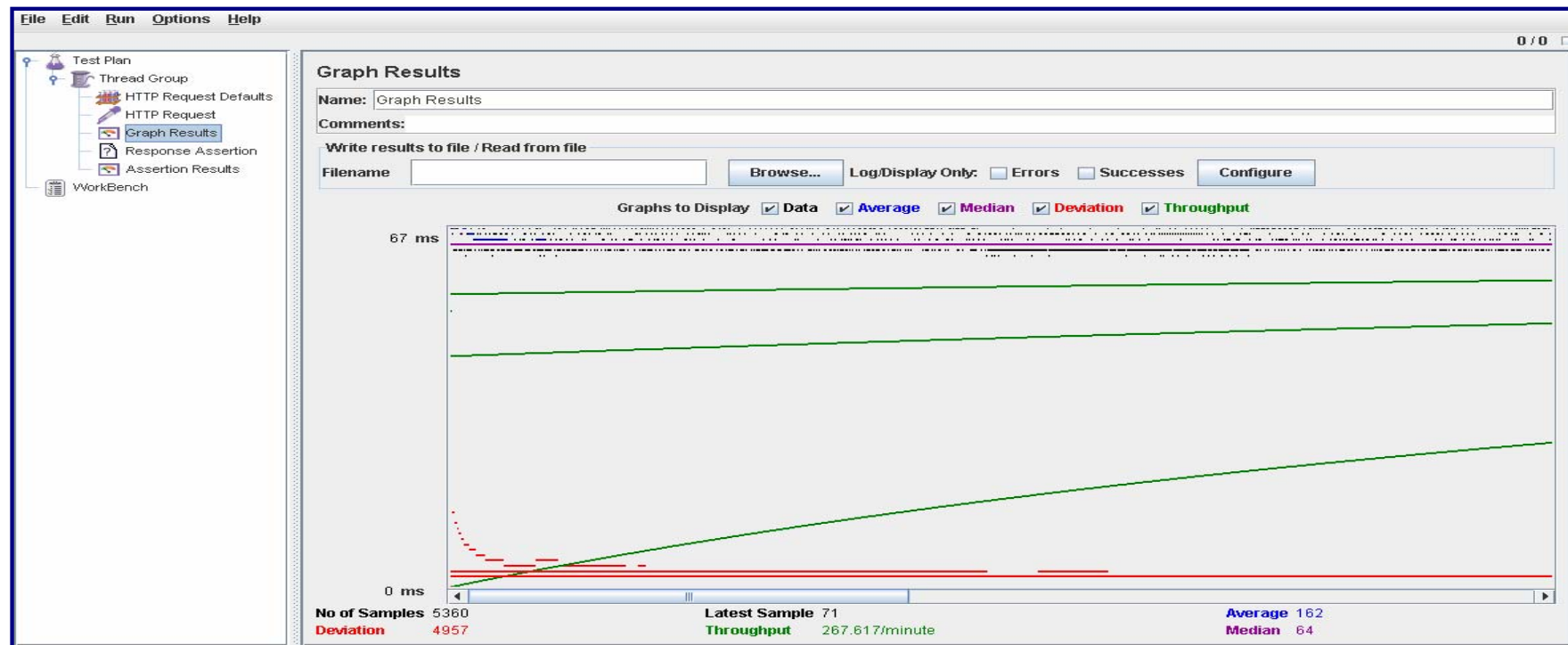
세부 목차

- 3.1 JMeter 주요 기능
- 3.2 예제 소개
- 3.3 웹 페이지 접속 가능 여부 확인 테스트 기능
- 3.4 웹 페이지 부하 테스트 기능
- 3.5 Response Assertion 테스트 기능
- 3.6 Duration Assertion 테스트 기능
- 3.7 Size Assertion 테스트 기능

3. 도구 기능 소개

3.1 JMeter의 주요기능

- JMeter의 주요기능은 다음과 같습니다.
 - 웹사이트 접속 가능 테스트 기능
 - 부하 테스트 기능
 - Response Assertion 테스트 기능
 - Duration Assertion 테스트 기능
 - Size Assertion 테스트 기능



[Graph Result의 결과 화면]

3. 도구 기능 소개

3.2 예제 소개 [1/2]

- 간단한 HTML 페이지를 사용하여 아래의 기능을 소개 하겠습니다.
 - 웹 페이지 접속 가능 여부 확인 테스트 기능
 - 웹 페이지 부하 테스트 기능
 - 응답 결과를 검증하기 위한 Assertion 기능
- 테스트를 위해 로컬 컴퓨터에 Apache 웹 서버를 구축한 후 테스트용 웹 페이지를 Apache 서버에 올려 테스트를 진행하였습니다.
- 테스트 웹 페이지는 Restaurant에 대한 정보를 입력하는 페이지이며 웹 페이지는 다음과 같이 구성되어 있습니다.
 - Restaurant Name
 - Restaurant Type
 - Location
 - Credit Cards사용 가능 유/무

Restaurant Name:

Restaurant Type: ☒ Chinese ☐ Tex/Mex ☐ Italian

Location:

☒ Accepts Credit Cards

[테스트용 웹 페이지]

3. 도구 기능 소개

3.2 예제 소개 [2/2]

- 테스트용 웹 페이지의 HTML 코드를 보면 다음과 같습니다.
 - 파일명은 form.html입니다.

```

1  <TABLE border="1" cellpadding="3">
2      <TR>
3          <TD>
4              <FORM>
5  Restaurant Name:
6              <INPUT name="Name" type="text" value="Park Ho-Jin"></INPUT>
7              <BR></BR>
8
9  Restaurant Type:
10             <INPUT checked="" name="Food" type="radio" value="Korean"></INPUT>
11             Chinese
12
13             <INPUT name="Food" type="radio" value="Chinese"></INPUT>
14             Tex/Mex
15
16             <INPUT name="Food" type="radio" value="Italian"></INPUT>
17             Italian
18             <BR></BR>
19
20 Location:
21             <SELECT name="Location">
22                 <OPTION value="Seoul">Seoul</OPTION>
23                 <OPTION value="Kyonggi">Kyonggi</OPTION>
24                 <OPTION selected="" value="Suwon">Suwon</OPTION>
25                 <OPTION value="Busan">Busan</OPTION>
26             </SELECT>
27
28             <BR></BR>
29
30             <INPUT checked="" name="CreditCard" type="checkbox"></INPUT>
31             Accepts Credit Cards
32             <BR></BR>
33             <BR></BR>
34             <INPUT type="submit" value="Update"></INPUT>
35 </FORM>
36 </TD>
37 </TR>
38 </TABLE>

```

[테스트용 웹 페이지의 HTML 코드]

3. 도구 기능 소개

3.3 웹 페이지 접속 가능 여부 확인 테스트 기능 (1/14)

- 테스트 플랜을 작성합니다.
 - 테스트 플랜은 JMeter가 수행할 일련의 실행 코드 집합을 말합니다.
 - JMeter를 사용한 테스트는 이 실행 코드 집합들을 알맞게 조립하는 것으로 표현됩니다.

The screenshot shows the JMeter 'Test Plan' configuration window. The window is divided into several sections. The top section contains a 'Name' field with the value 'Test Plan' and a 'Comments' field. Below this is a 'User Defined Variables' section with a table for 'Name' and 'Value'. At the bottom of the window, there are checkboxes for 'Run Thread Groups consecutively' and 'Functional Test Mode', a section for 'Add directory or jar to classpath' with 'Browse...', 'Delete', and 'Clear' buttons, and a 'Library' list.

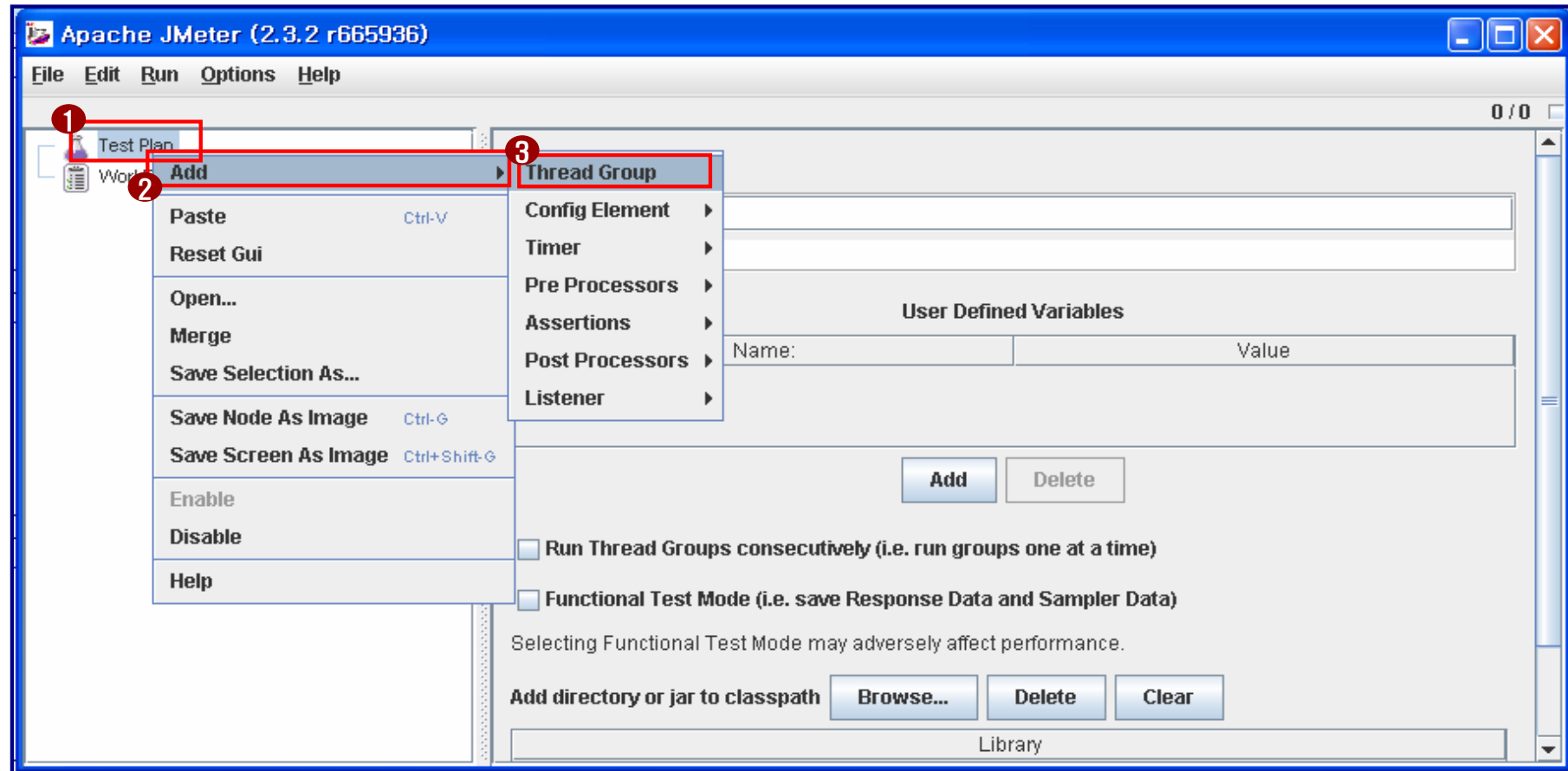
Annotations in the image include:

- A red box around the 'Test Plan' icon in the left sidebar.
- A red box around the 'Name' field containing 'Test Plan'.
- A callout box pointing to the 'Name' field: "Test Plan의 이름을 입력합니다."
- A red box around the 'Add' button in the 'User Defined Variables' section.
- A callout box pointing to the 'Add' button: "사용자 정의 변수가 있다면 Add 버튼을 누른 후 입력해 줍니다."
- A red box around the 'Add directory or jar to classpath' section.
- A callout box pointing to this section: "Test Plan 수행 중 필요한 jar파일이 있다면 이곳에서 추가해 줍니다."

3. 도구 기능 소개

3.3 웹 페이지 접속 가능 여부 확인 테스트 기능 (2/14)

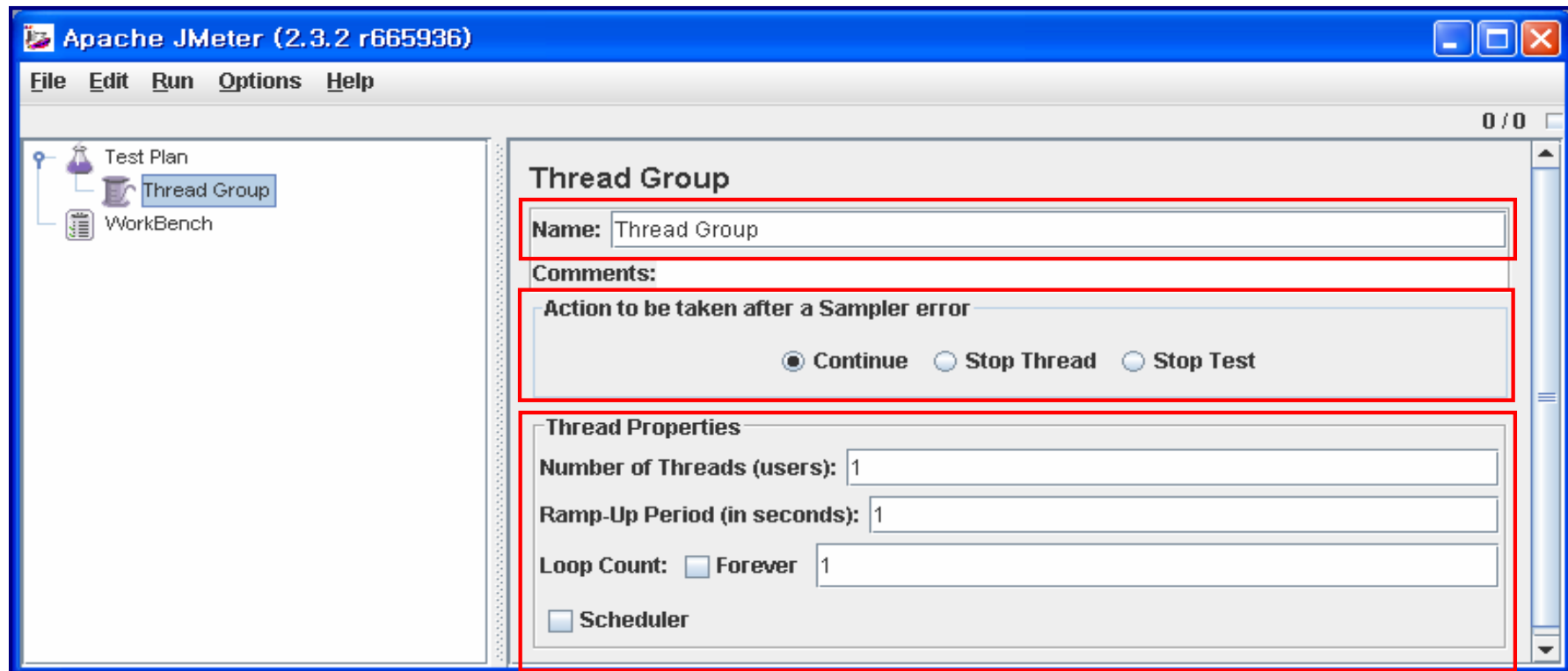
- Thread Group을 만듭니다.
 - Thread Group이란 테스트 플랜이 수행할 하나의 Group을 말합니다.
 - 왼쪽 Test Plan에서 오른쪽 버튼 -> Add -> Thread Group 선택



3. 도구 기능 소개

3.3 웹 페이지 접속 가능 여부 확인 테스트 기능 (3/14)

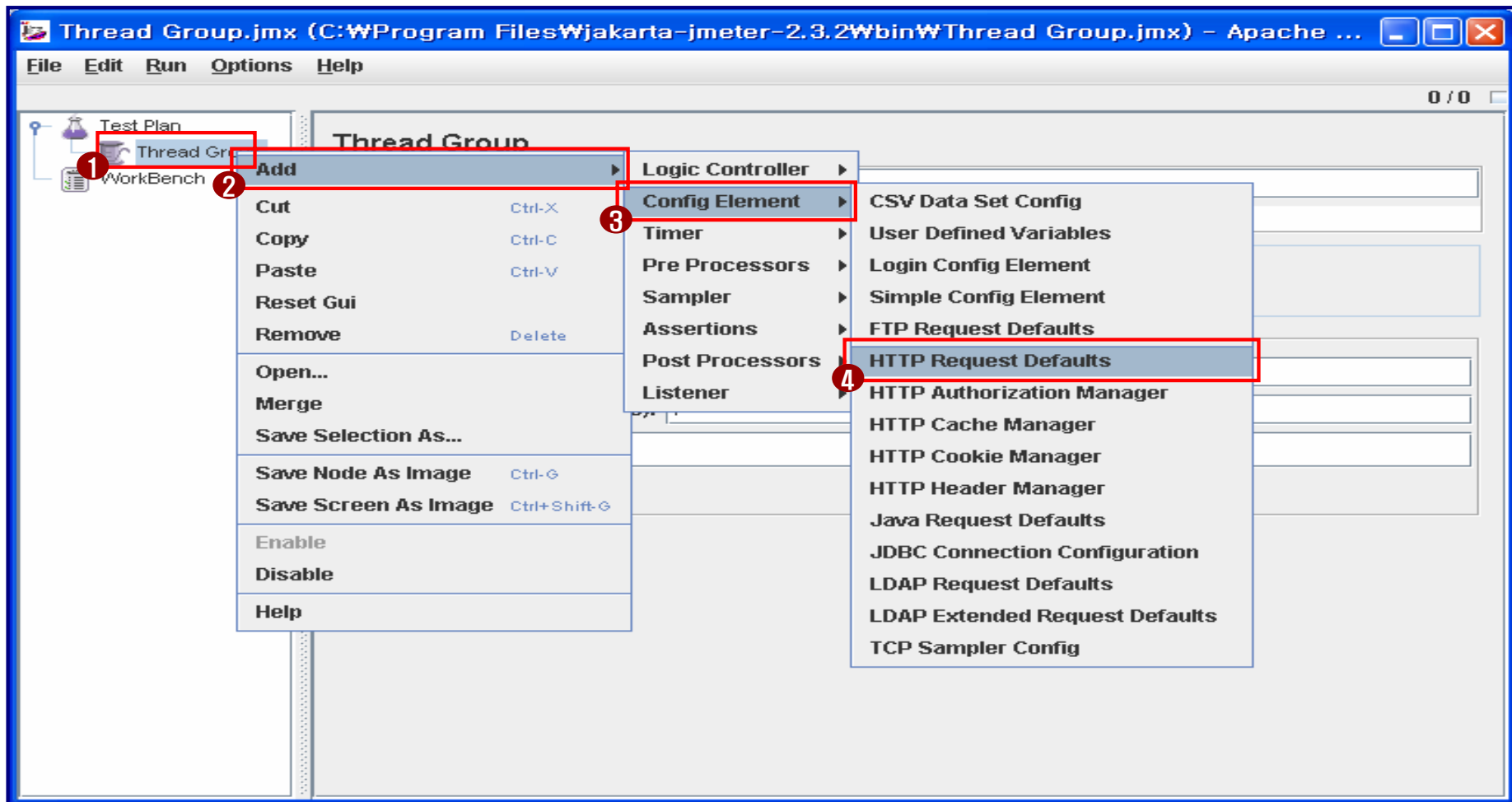
- Thread Group 생성 후 오른쪽에 나오는 항목들을 작성 합니다.
 - Name : Thread Group의 이름을 입력합니다.
 - Action to be taken after a sampler error : 테스트 수행 중 에러가 발생할 때의 상황을 설정합니다.
 - Thread Properties
 - Number of Threads : 이 Thread Group에 생성될 Thread의 개수를 지정합니다.
 - Ramp-up Period : 한 Thread가 시작한 후 다음 Thread가 시작될 때까지의 대기 시간을 지정합니다.
 - Loop Count : 각 Thread가 Thread Group에 속한 작업의 반복 횟수를 지정합니다.
 - Scheduler : Thread Group의 시작 및 종료 스케줄을 설정할 것인지 체크합니다.



3. 도구 기능 소개

3.3 웹 페이지 접속 가능 여부 확인 테스트 기능 (4/14)

- Http Request Defaults를 추가 합니다.
 - Thread Group에서 마우스 오른쪽 버튼 클릭 -> Add -> Config Element -> Http Request Defaults



3. 도구 기능 소개

3.3 웹 페이지 접속 가능 여부 확인 테스트 기능 (5/14)

- Http Request Defaults는 Config Element에 속한 요소로서 관련된 Http Requests가 사용할 기본 정보를 지정합니다.
 - Http Request Defaults는 그 자체적으로 어떤 기능도 수행하지 않습니다.
 - Http Request Defaults는 관련 영역에 있는 모든 Http Requests 요소에 영향을 줍니다.

HTTP Request Defaults

Name: HTTP Request Defaults

Comments:

Web Server

Server Name or IP: localhost Port Number:

HTTP Request

Protocol (default http): Content encoding:

Path: form.html

Send Parameters With the Request:

Name:	Value	Encode?	Include Equ...

Add Delete

☐ Retrieve All Embedded Resources from HTML Files

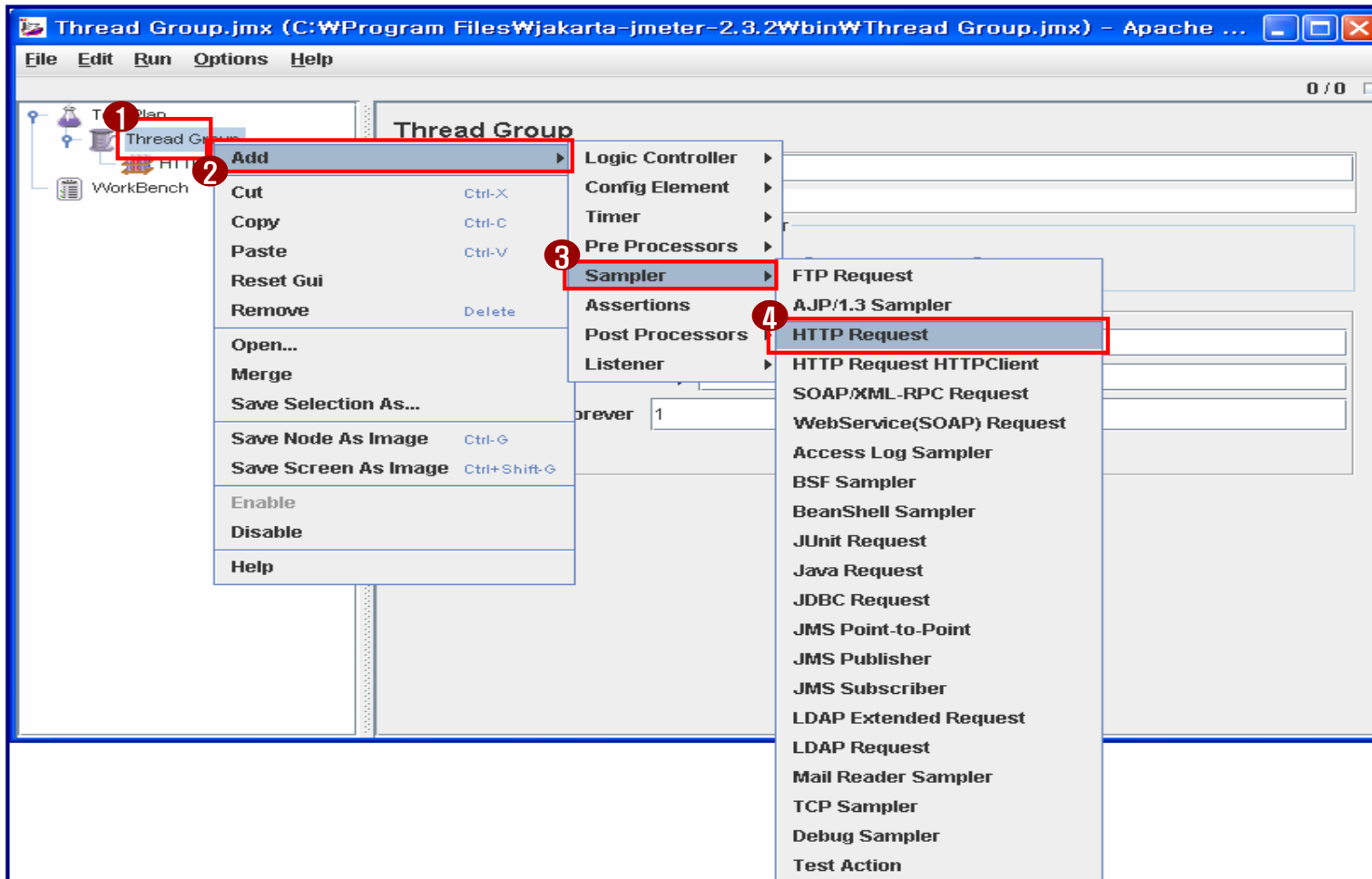
- Name : Http Request Defaults의 이름을 입력합니다.
- Server Name or IP : 서버의 이름 또는 IP주소를 입력합니다.
- Port Number : 접속 시 사용할 포트 번호를 입력합니다.
- Protocol : 사용할 프로토콜을 입력합니다.
(주로 HTTP를 사용합니다.)
- Content encoding : 인코딩 방식을 입력합니다.
- Path : 접속할 경로를 입력합니다.

기본적으로 전송할 파라미터 목록 및 값을 입력합니다.

3. 도구 기능 소개

3.3 웹 페이지 접속 가능 여부 확인 테스트 기능 (6/14)

- Http Request를 추가 합니다.
- Http Request 요소는 HTTP 요청을 서버에 전송하여 그 결과를 받아오는 부분입니다.
 - Thread Group에서 오른쪽 버튼 -> Add -> Sampler -> HTTP Request 선택



3. 도구 기능 소개

3.3 웹 페이지 접속 가능 여부 확인 테스트 기능 (7/14)

- Http Request의 입력 정보는 다음과 같습니다.
 - 입력하지 않은 정보는 앞에서 작성한 Http Request Defaults에 작성된 내용이 반영되게 됩니다.

HTTP Request

Name: HTTP Request

Comments:

Web Server

Server Name or IP: Port Number:

HTTP Request

Protocol (default http): Method: GET Content encoding:

Path:

☒ Redirect Automatically ☐ Follow Redirects ☒ Use KeepAlive ☐ Use multipart/form-data for HTTP POST

Send Parameters With the Request:

Name:	Value	Encode?	Include Equ...

Add Delete

Send Files With the Request:

File Path:	Parameter...	MIME Type:

Add Browse... Delete

Optional Tasks

☐ Retrieve All Embedded Resources from HTML Files ☐ Use as Monitor ☐ Save response as MD5 hash?

Embedded URLs must match:

- Name : Http Request의 이름을 입력합니다.
- Server Name or IP : 서버의 이름 또는 IP주소를 입력합니다.
- Port Number : 접속 시 사용할 포트 번호를 입력합니다.
- Protocol : 사용할 프로토콜을 입력합니다.
(주로 HTTP를 사용합니다.)
- Method : 전송방식을 선택합니다.
- Content encoding : 인코딩 방식을 입력합니다.
- Path : 접속할 경로를 입력합니다.

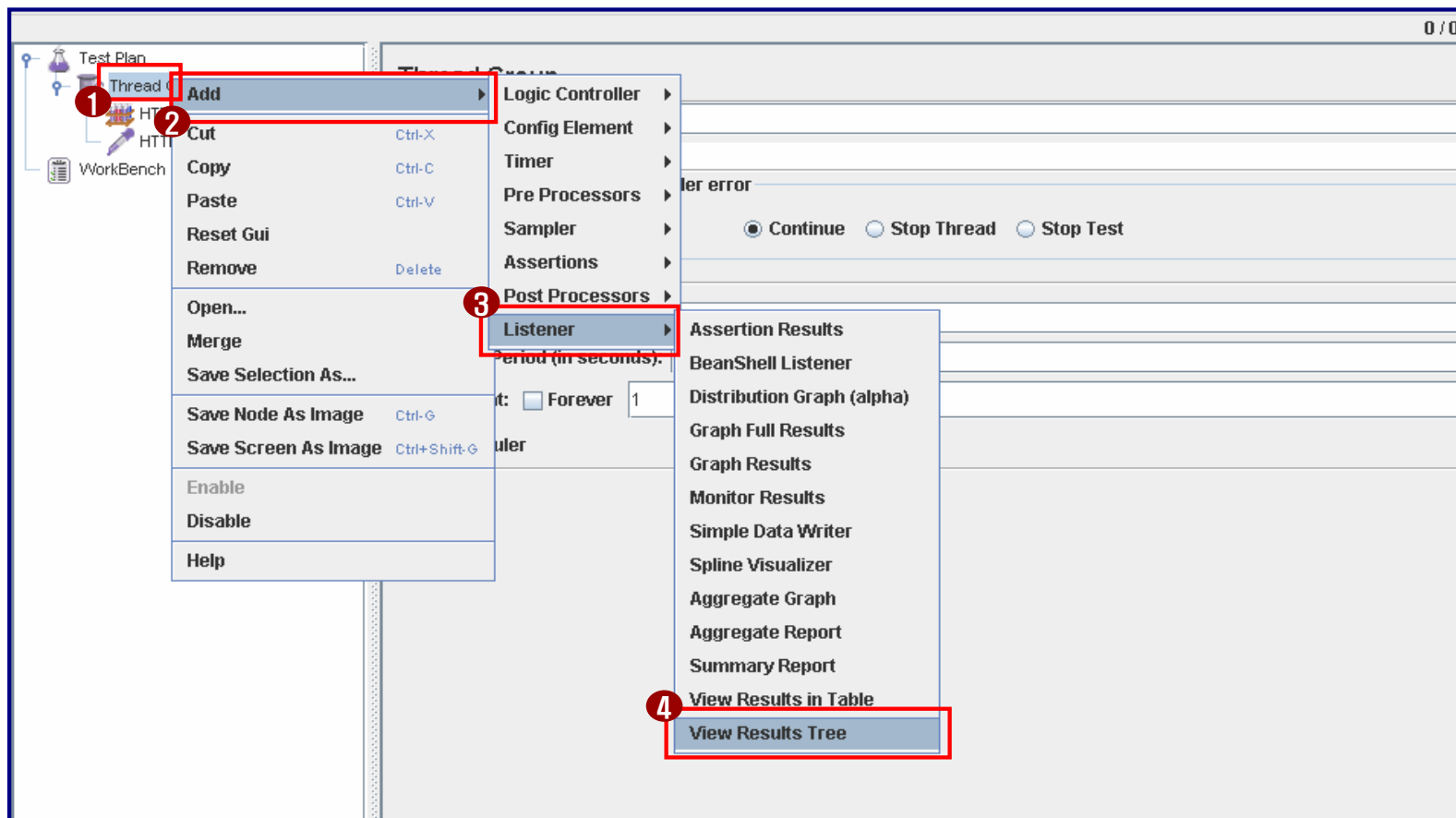
- 기본적으로 전송할 파라미터 목록 및 값을 입력합니다.

- 파일을 전송할 경우 Add버튼을 눌러서 파일을 선택하며 Parameter Name에는 파일과 관련된 파라미터의 이름을 입력합니다. MIME type에는 전송하는 파일의 MIME 타입을 입력합니다.

3. 도구 기능 소개

3.3 웹 페이지 접속 가능 여부 확인 테스트 기능 (8/14)

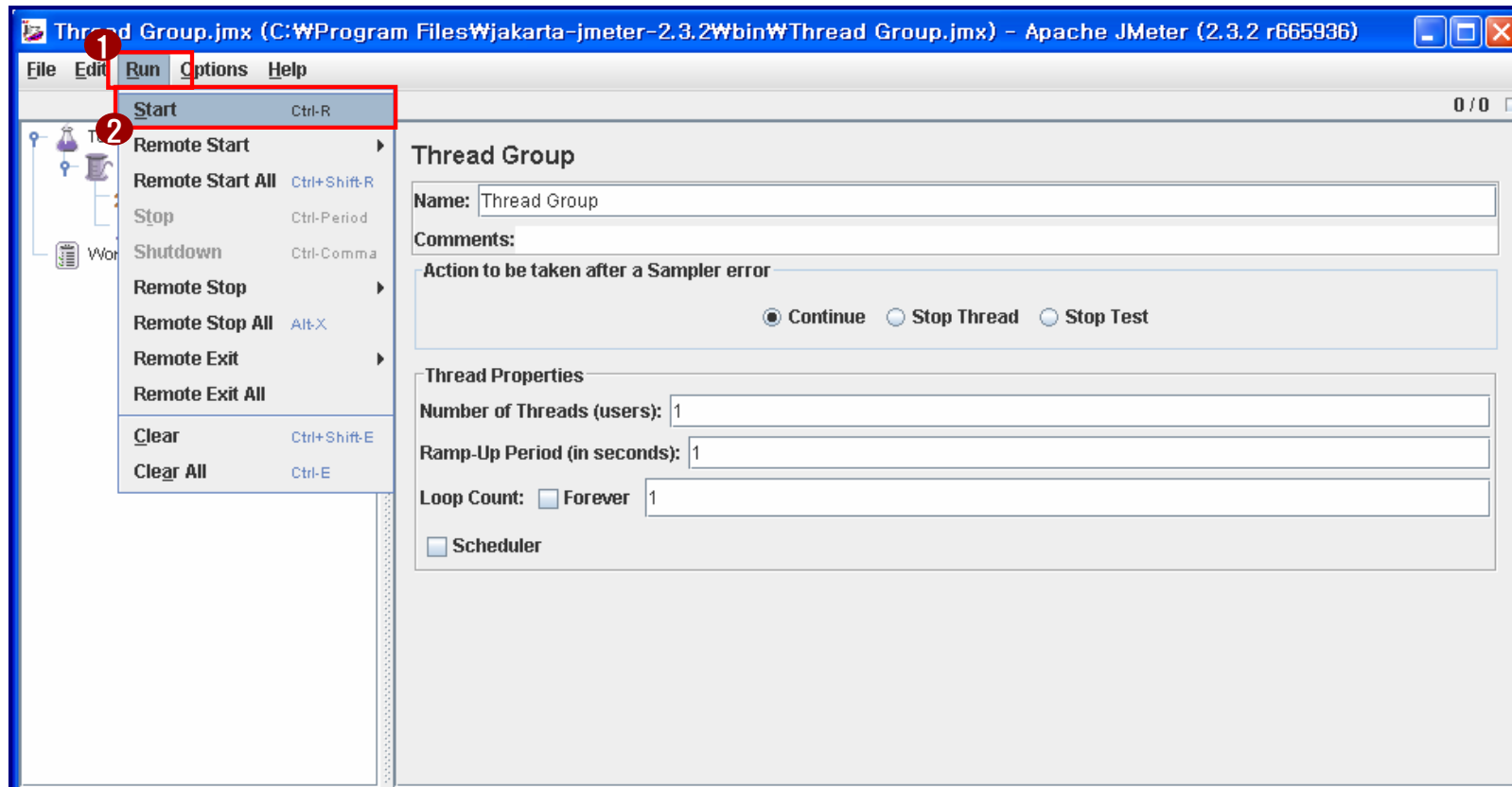
- 테스트 결과를 출력해 주는 Listener를 추가 합니다.
 - 테스트 결과를 출력하는 Listener로서 View Results Tree를 사용할 것입니다.
 - Thread Group선택 후 마우스 오른쪽 버튼 -> Add -> Listener -> View Results Tree 선택



3. 도구 기능 소개

3.3 웹 페이지 접속 가능 여부 확인 테스트 기능 (9/14)

- 마지막으로 해야 할 일은 테스트를 실행 하는 것입니다.
 - 메뉴 바의 Run -> Start



3. 도구 기능 소개

3.3 웹 페이지 접속 가능 여부 확인 테스트 기능 (10/14)

- 앞에서 Start를 클릭한 후 왼쪽 트리에서 View Results Tree를 선택한 다음 오른쪽에서 HTTP Request를 클릭하면 테스트 결과가 출력되게 됩니다.

Thread Group.jmx (C:\WProgram Files\Jakarta-jmeter-2.3.2\bin\Thread Group.jmx) - Apache JMeter (2.3.2 r665936)

File Edit Run Options Help

Test Plan
Thread Group
HTTP Request Defaults
HTTP Request
View Results Tree
Workbench

View Results Tree
Name: View Results Tree
Comments:
Write results to file / Read from file
Filename: Browse... Log/Display Only: ☐ Errors ☐ Successes

2 HTTP Request

Sampler result Request Response data

Thread Name: Thread Group 1-1
Sample Start: 2008-09-10 11:45:05 KST
Load time: 1
Latency: 1
Size in bytes: 1168
Sample Count: 1
Error Count: 0
Response code: 200
Response message: OK

Response headers:
HTTP/1.1 200 OK
Date: Wed, 10 Sep 2008 02:45:05 GMT
Server: Apache/2.2.8 (Win32) DAV/2 mod_ssl/2.2.8 OpenSSL/0.9.8g mod_autoindex_color PHP/5.2.5
Last-Modified: Thu, 21 Aug 2008 00:27:25 GMT
ETag: "b000000029940-490-454ed62a0d072"
Accept-Ranges: bytes
Content-Length: 1168
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html

- Thread Name : 수행한 Thread Group의 이름
- Sample Start : 테스트를 수행한 시간
- Load time : 로딩 횟수
- Latency : 대기 횟수
- Size in Bytes : 로딩한 페이지의 크기
- Sample Count : 수행한 테스트의 수
- Error Count : 에러 발생 수
- Response code : 웹 페이지 요청 상태 코드
- Response message : 응답 메시지 대한 결과를 보여줌

● 이 부분은 응답 헤더(Response headers)에 대한 정보를 보여주고 있습니다.

3. 도구 기능 소개

3.3 웹 페이지 접속 가능 여부 확인 테스트 기능 (11/14)

- 테스트 결과는 다음과 같습니다.

이 부분을 통해서 이 테스트가 성공적으로 수행되었다는 것을 알 수 있습니다.
그 이유는 우선 에러의 수가 없으며 웹 페이지 요청이 성공적으로 이루어졌기 때문입니다.

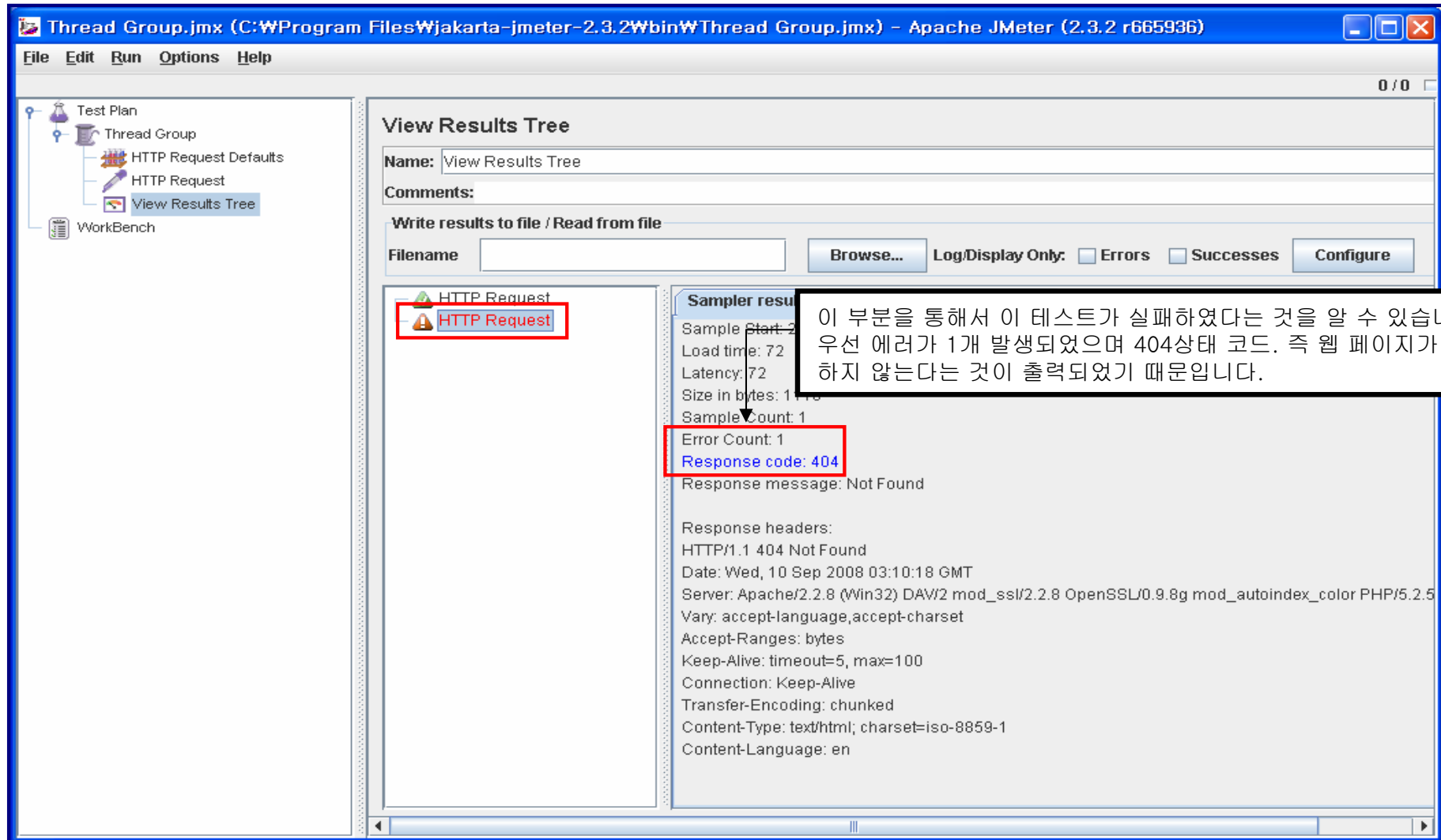
이 테스트의 결과는 다음과 같습니다.
테스트는 2008년 9월 10일 11시 45분에 수행했으며 로딩 횟수는 1번, 대기 횟수는 1번입니다. 요청한 웹 페이지의 크기는 1168바이트이며 수행한 테스트 수는 1번입니다.
에러는 발생되지 않았으며 응답한 코드의 수는 200개입니다.
마지막으로 응답 메시지는 성공적으로 받았다는 것을 Response message 부분에서 볼 수 있습니다. 즉 웹 페이지에 대한 접속 가능 여부 또는 존재여부를 테스트 한 결과 에러가 발생되지 않았으며 또한 응답 메시지도 성공적으로 받았기 때문에 웹 페이지는 존재 하며 접속이 가능하다는 것을 알 수 있습니다.

View Results Tree
Name: View Results Tree
Comments:
Write results to file / Read from file
Filename:
2 **HTTP Request**
Sampler result
Thread Name: Thread Group 1-1
Sample Start: 2008-09-10 11:45:05 KST
Load time: 1
Latency: 1
Size in bytes: 1168
Sample Count: 1
Error Count: 0
Response code: 200
Response message: OK
Response headers:
HTTP/1.1 200 OK
Date: Wed, 10 Sep 2008 02:45:05 GMT
Server: Apache/2.2.8 (Win32) DAW/2 mod_ssl/2.2.8 OpenSSL/0.9.8g mod_autoindex_color PHP/5.2.5
Last-Modified: Thu, 21 Aug 2008 00:27:25 GMT
ETag: "b000000029940-490-454ed62a0d072"
Accept-Ranges: bytes
Content-Length: 1168
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html

3. 도구 기능 소개

3.3 웹 페이지 접속 가능 여부 확인 테스트 기능 (12/14)

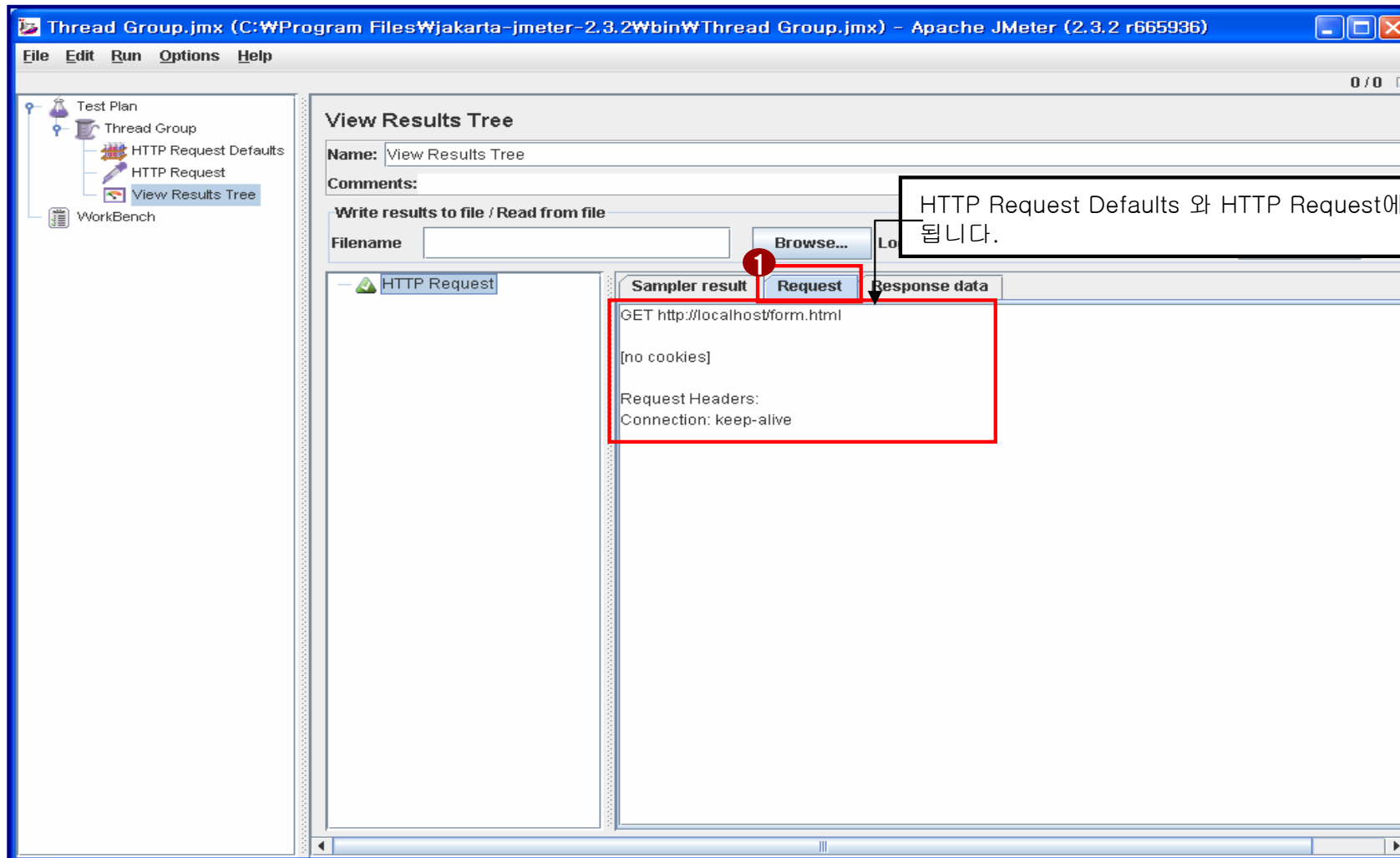
- 테스트 실패 시에는 다음과 같은 결과가 출력됩니다.



3. 도구 기능 소개

3.3 웹 페이지 접속 가능 여부 확인 테스트 기능 (13/14)

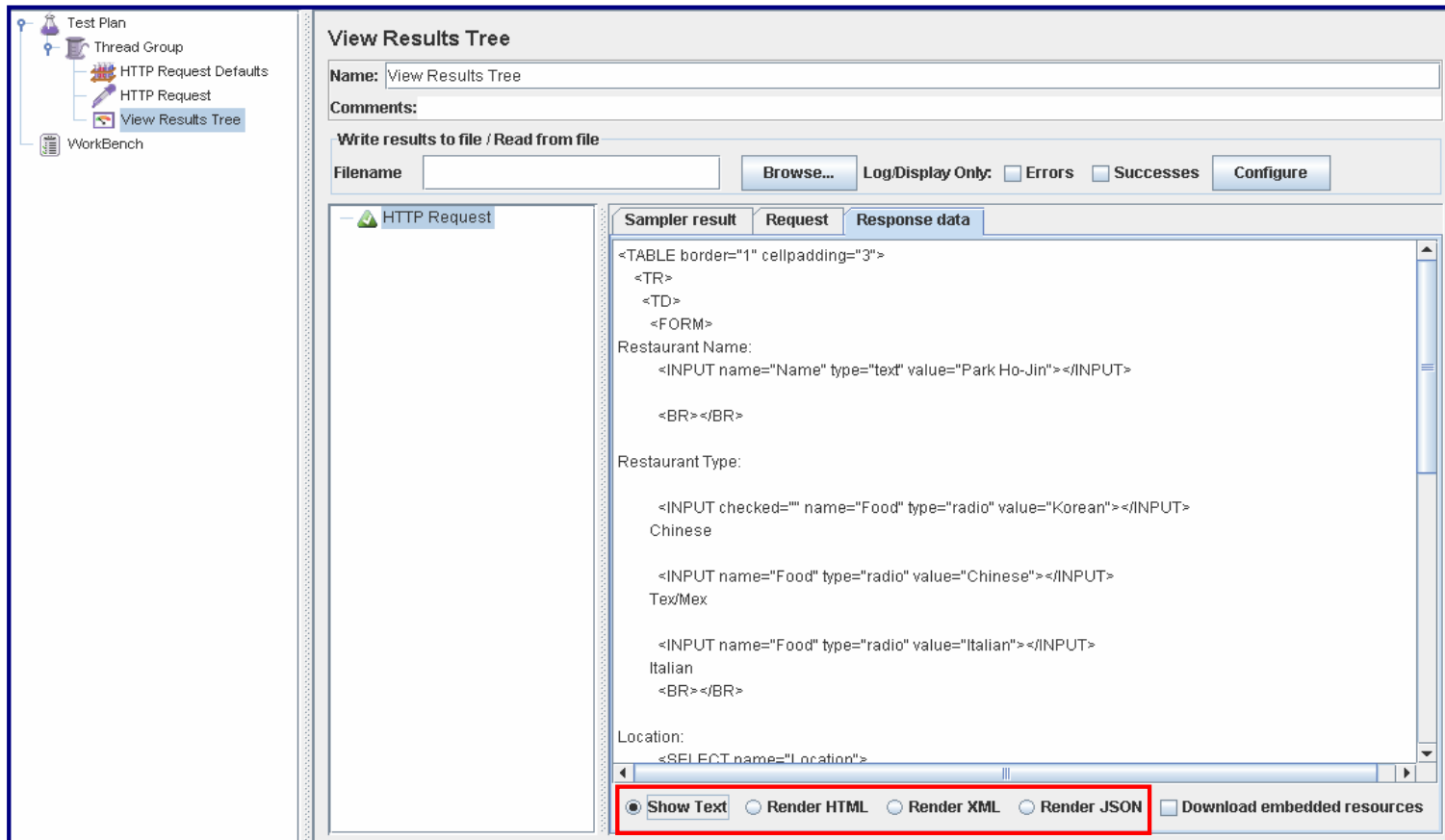
- Request탭을 누르면 다음과 같은 화면이 표시가 됩니다.
 - Request탭은 사용자가 테스트를 위해서 요청한 내용이 표시가 됩니다.



3. 도구 기능 소개

3.3 웹 페이지 접속 가능 여부 확인 테스트 기능 [14/14]

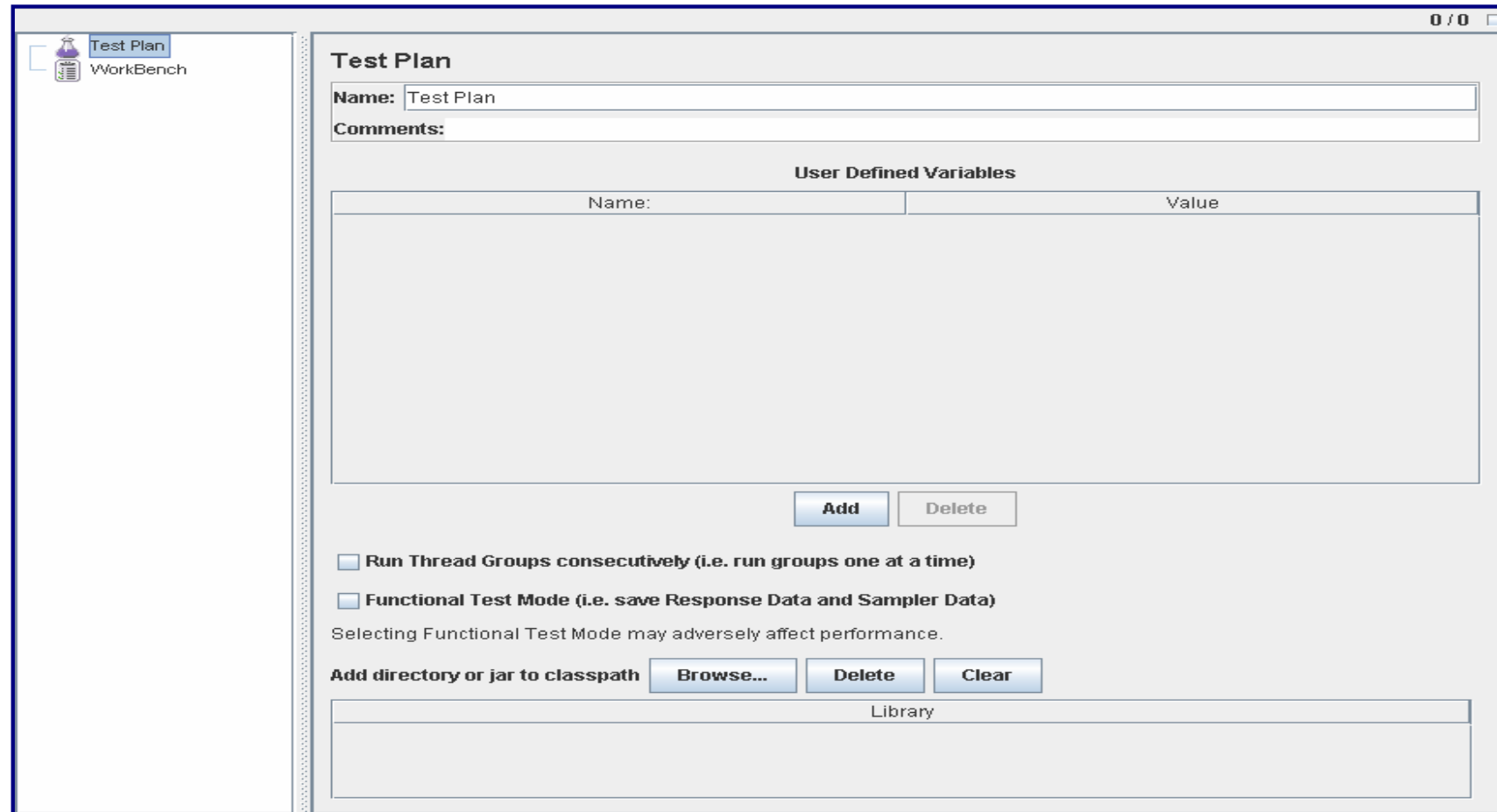
- Response data탭을 누르면 다음과 같은 화면이 표시가 됩니다.
 - Response data탭은 응답한 data가 표시가 됩니다.
 - Show Text : 텍스트 형식으로 응답데이터를 보여줍니다.
 - Render HTML : HTML 형식으로 응답데이터를 보여줍니다.
 - Render XML : XML 형식으로 응답데이터를 보여줍니다.
 - Render JSON : JSON 형식으로 응답데이터를 보여줍니다.



3. 도구 기능 소개

3.4 웹 페이지 부하 테스트 기능 [1/12]

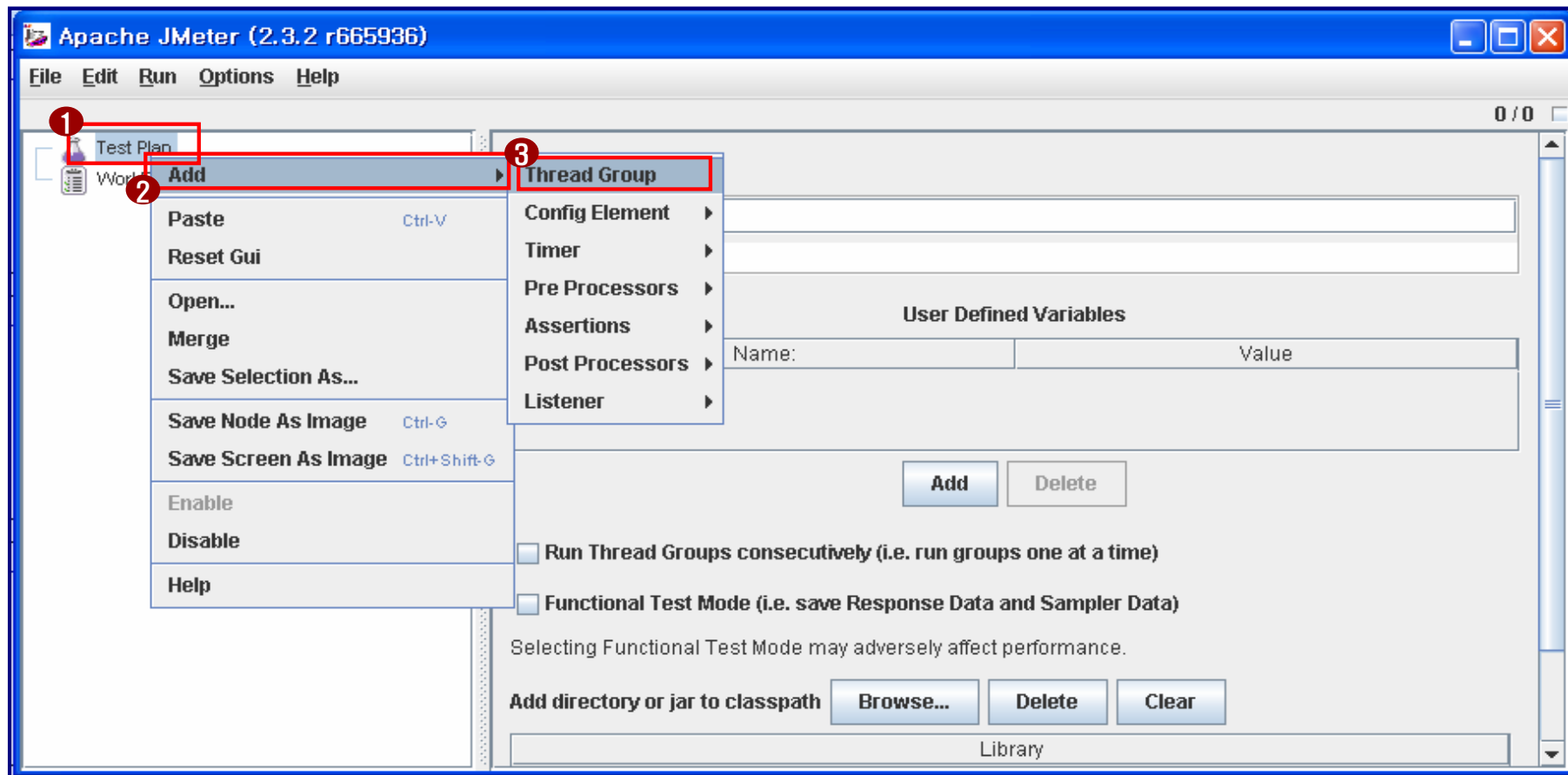
- 부하 테스트란, 서비스가 작동될 서버에 의도적으로 일정량의 트래픽을 발생시켜 서버가 다운되지 않고 정상적으로 작동되는지를 테스트 하는 것입니다.
- 처음에 할 일은 테스트 플랜을 작성하는 것입니다.
 - 16 페이지를 참조합니다.



3. 도구 기능 소개

3.4 웹 페이지 부하 테스트 기능 [2/12]

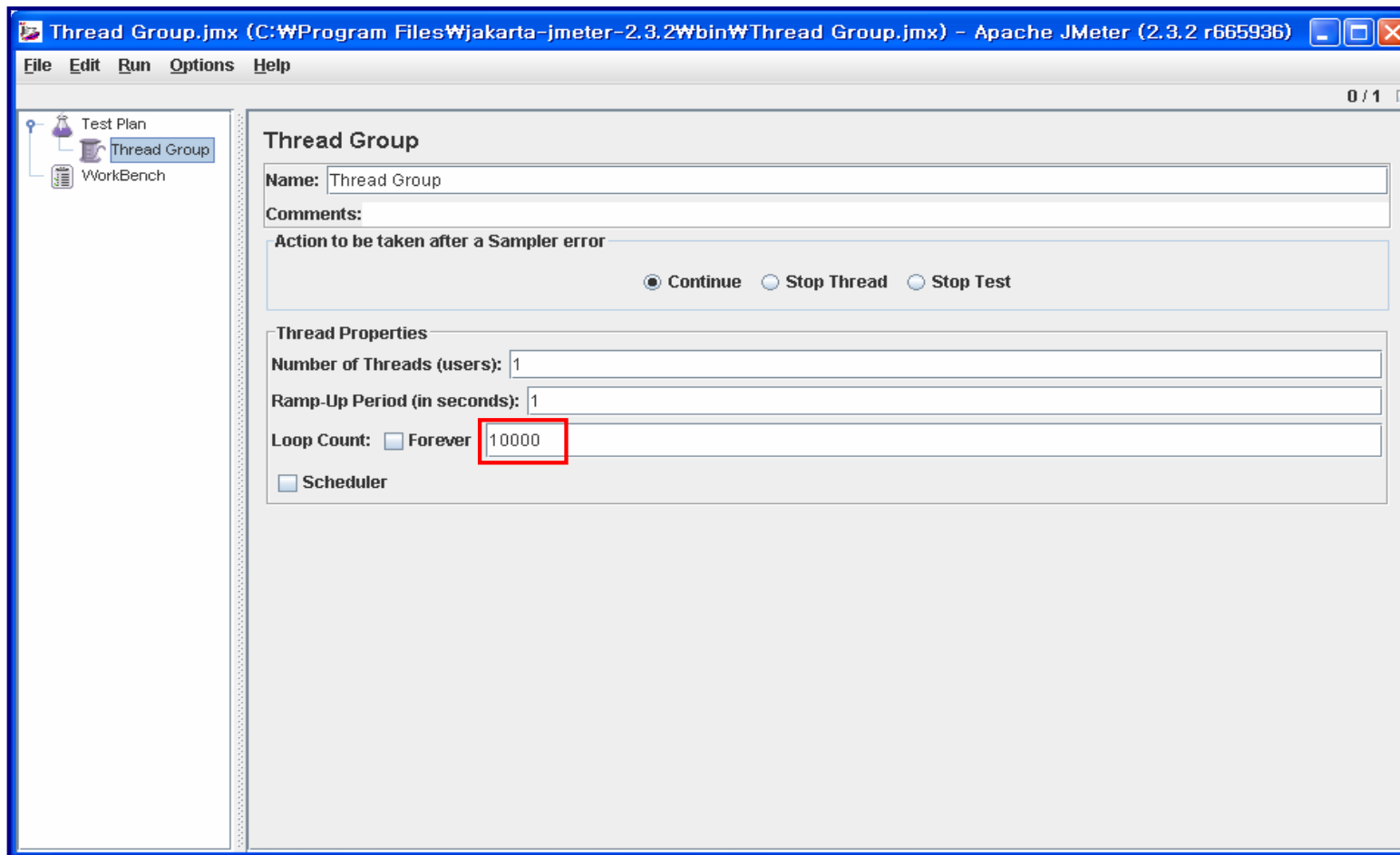
- Thread Group을 만듭니다.
- Thread Group이란 테스트 플랜이 수행할 하나의 Group을 말합니다.
 - 왼쪽 Test Plan에서 오른쪽 버튼 -> Add -> Thread Group을 선택



3. 도구 기능 소개

3.4 웹 페이지 부하 테스트 기능 [3/12]

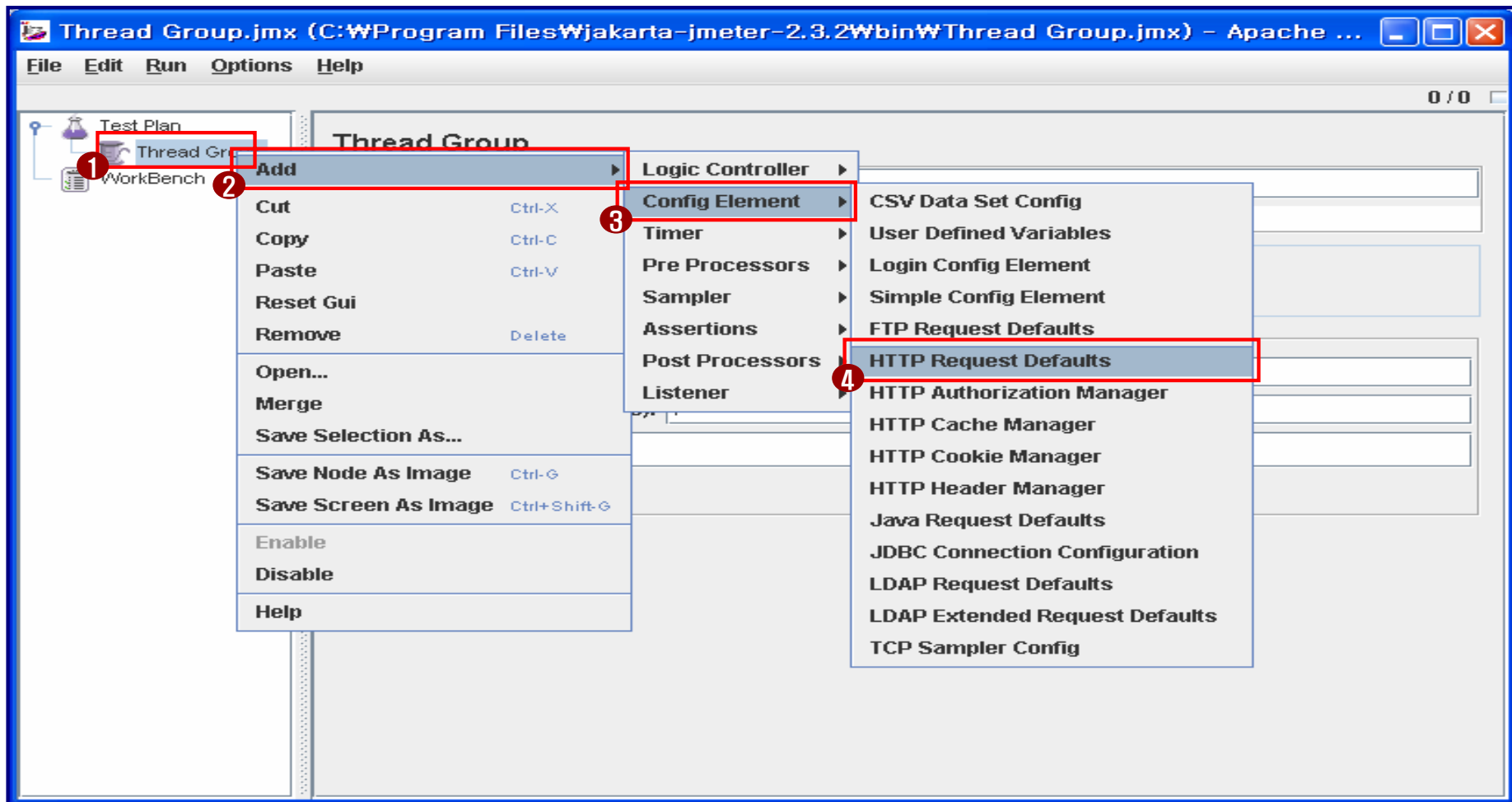
- 오른쪽에 나오는 항목들을 작성합니다.
 - 부하 테스트를 위해서 Loop Count 부분에서 10000을 입력합니다.
 - Loop Count는 웹 페이지 접속횟수를 뜻합니다.
 - 본 예제에서는 Loop Count에 10000을 입력하였으므로, 웹 페이지에 10000번 접속한다는 것입니다.
 - 나머지 부분은 18페이지를 참고하여 입력 합니다.



3. 도구 기능 소개

3.4 웹 페이지 부하 테스트 기능 [4/12]

- Http Request Defaults를 추가 합니다.
 - Thread Group 선택 후 마우스 오른쪽 버튼 -> Add -> Config Element -> Http Request Defaults



3. 도구 기능 소개

3.4 웹 페이지 부하 테스트 기능 (5/12)

- 아래의 항목들을 작성합니다.
 - 20페이지를 참조합니다.

HTTP Request Defaults

Name: HTTP Request Defaults

Comments:

Web Server

Server Name or IP: localhost Port Number:

HTTP Request

Protocol (default http): Content encoding:

Path: form.html

Send Parameters With the Request:

Name:	Value	Encode?	Include Equ...

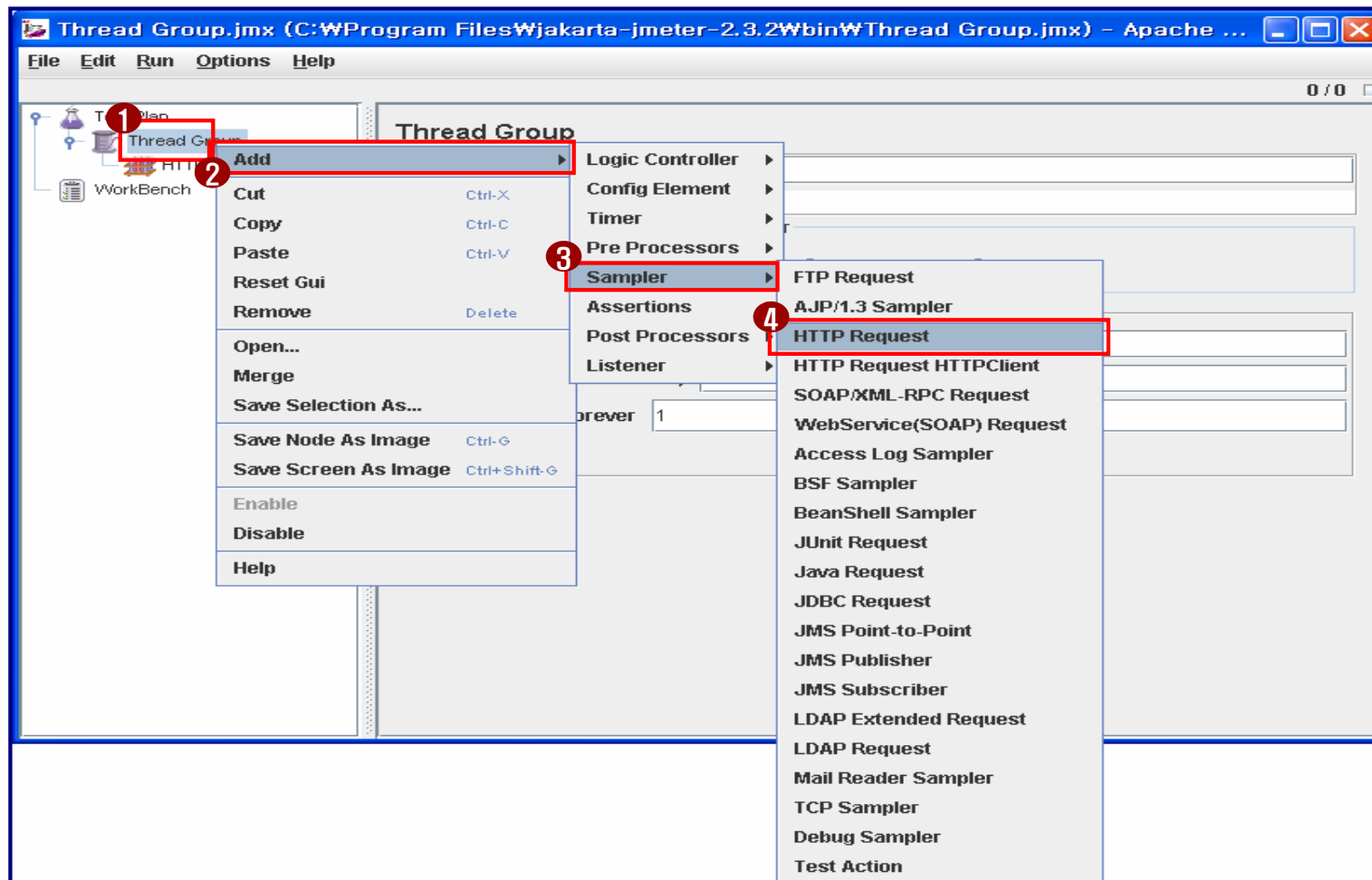
Add Delete

☐ Retrieve All Embedded Resources from HTML Files

3. 도구 기능 소개

3.4 웹 페이지 부하 테스트 기능 [6/12]

- Http Request를 추가 합니다.
 - Thread Group에서 오른쪽 버튼 -> Add -> Sampler -> HTTP Request 선택



3. 도구 기능 소개

3.4 웹 페이지 부하 테스트 기능 [7/12]

- Http Request의 입력 정보는 다음과 같습니다.
 - 22페이지를 참조합니다.

HTTP Request

Name:

Comments:

Web Server

Server Name or IP: Port Number:

HTTP Request

Protocol (default http): Method: Content encoding:

Path:

☒ Redirect Automatically ☐ Follow Redirects ☒ Use KeepAlive ☐ Use multipart/form-data for HTTP POST

Send Parameters With the Request:

Name:	Value	Encode?	Include Equ...

Send Files With the Request:

File Path:	Parameter ...	MIME Type:

Optional Tasks

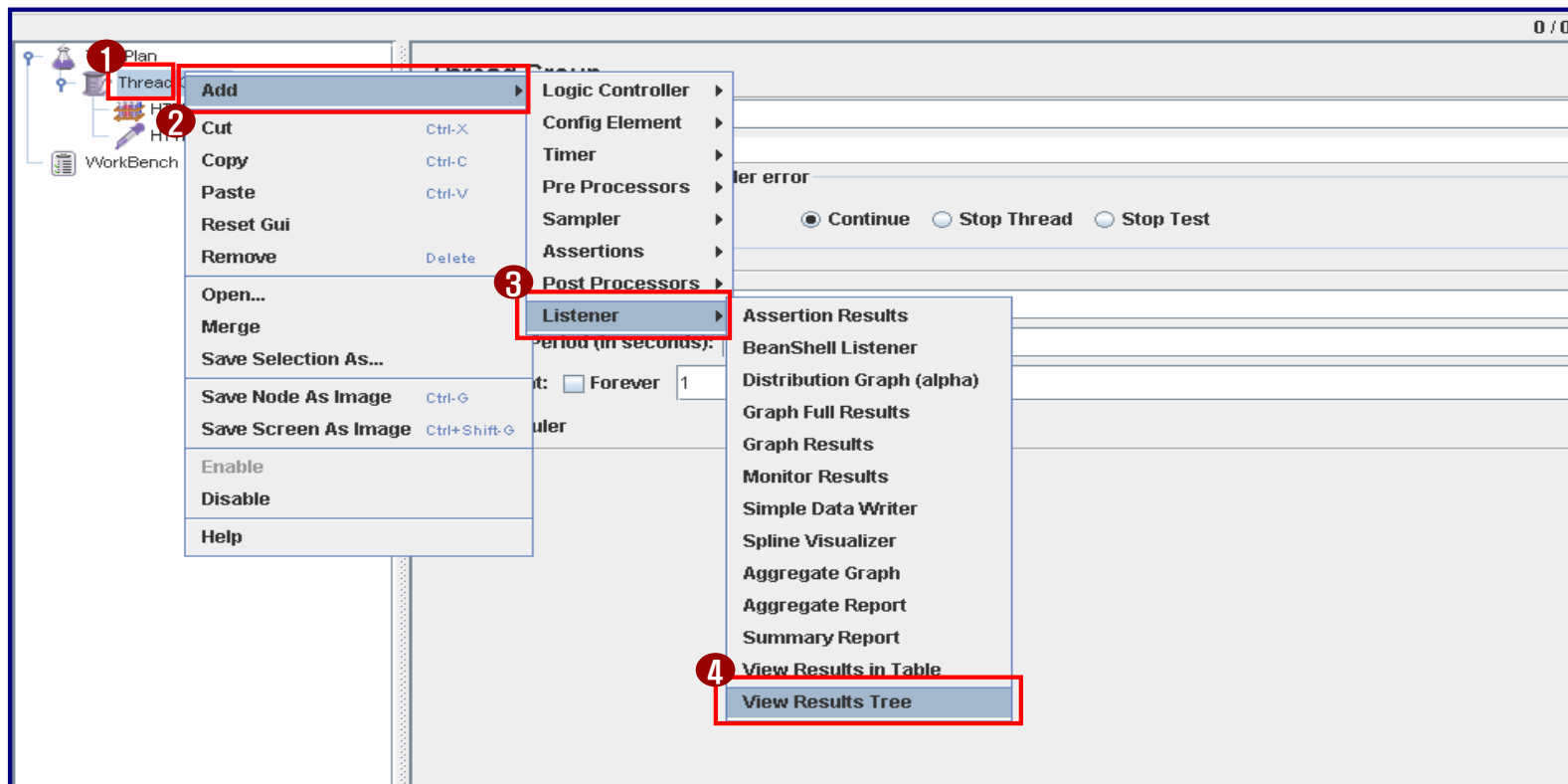
☐ Retrieve All Embedded Resources from HTML Files ☐ Use as Monitor ☐ Save response as MD5 hash?

Embedded URLs must match:

3. 도구 기능 소개

3.4 웹 페이지 부하 테스트 기능 [8/12]

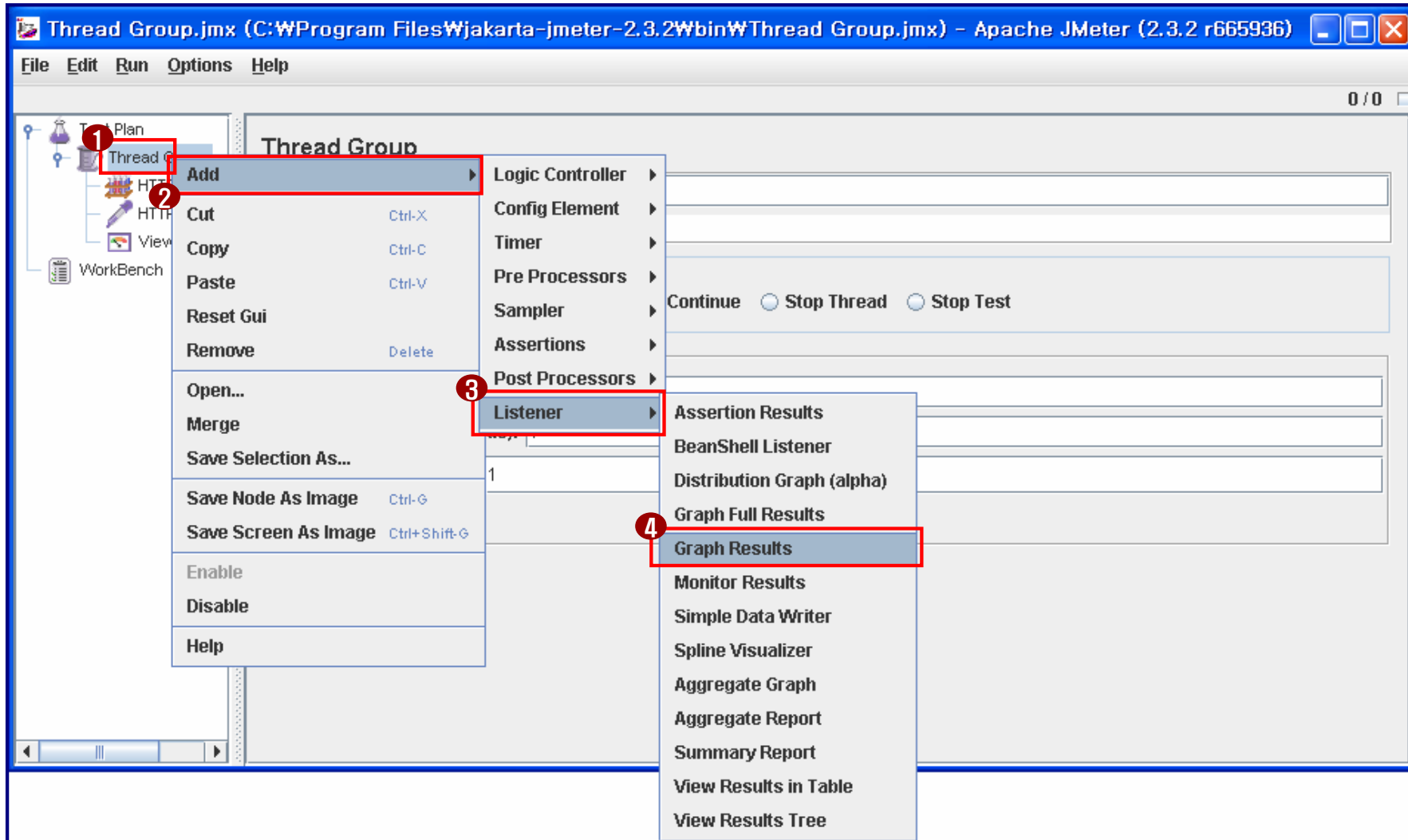
- 테스트 결과를 출력해 주는 Listener를 추가 합니다.
 - 2개의 Listener를 추가할 것입니다.
 - View Results Tree
 - Graph Results
 - Thread Group 선택 후 마우스 오른쪽 버튼 -> Add -> Listener -> View Results Tree 선택



3. 도구 기능 소개

3.4 웹 페이지 부하 테스트 기능 [9/12]

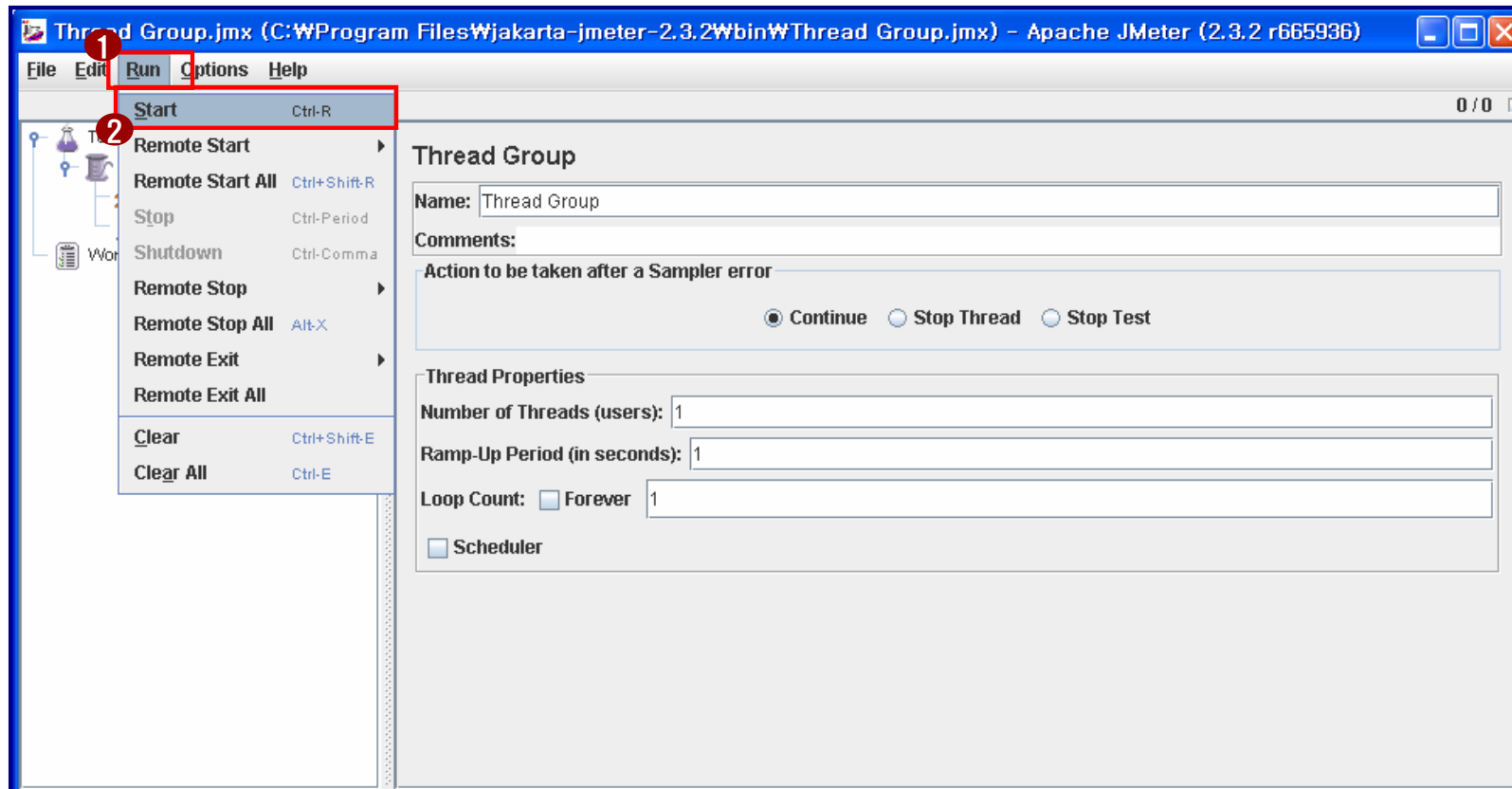
- Thread Group 선택 후 마우스 오른쪽 버튼 -> Add -> Listener -> Graph Results 선택



3. 도구 기능 소개

3.4 웹 페이지 부하 테스트 기능 [10/12]

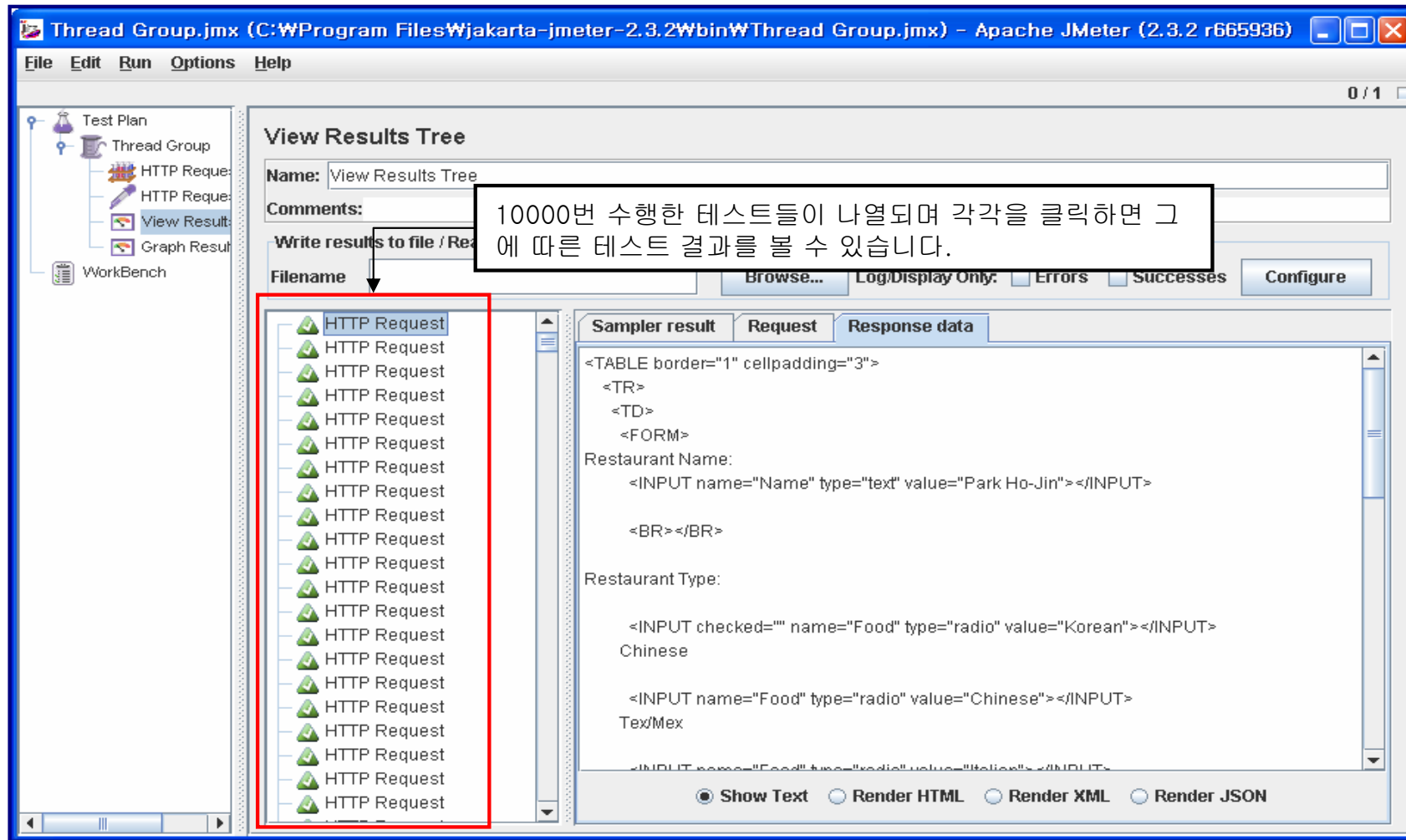
- 마지막으로 해야 할 일은 테스트를 실행 하는 것입니다.
 - 메뉴 바의 Run 선택 -> Start



3. 도구 기능 소개

3.4 웹 페이지 부하 테스트 기능 (11/12)

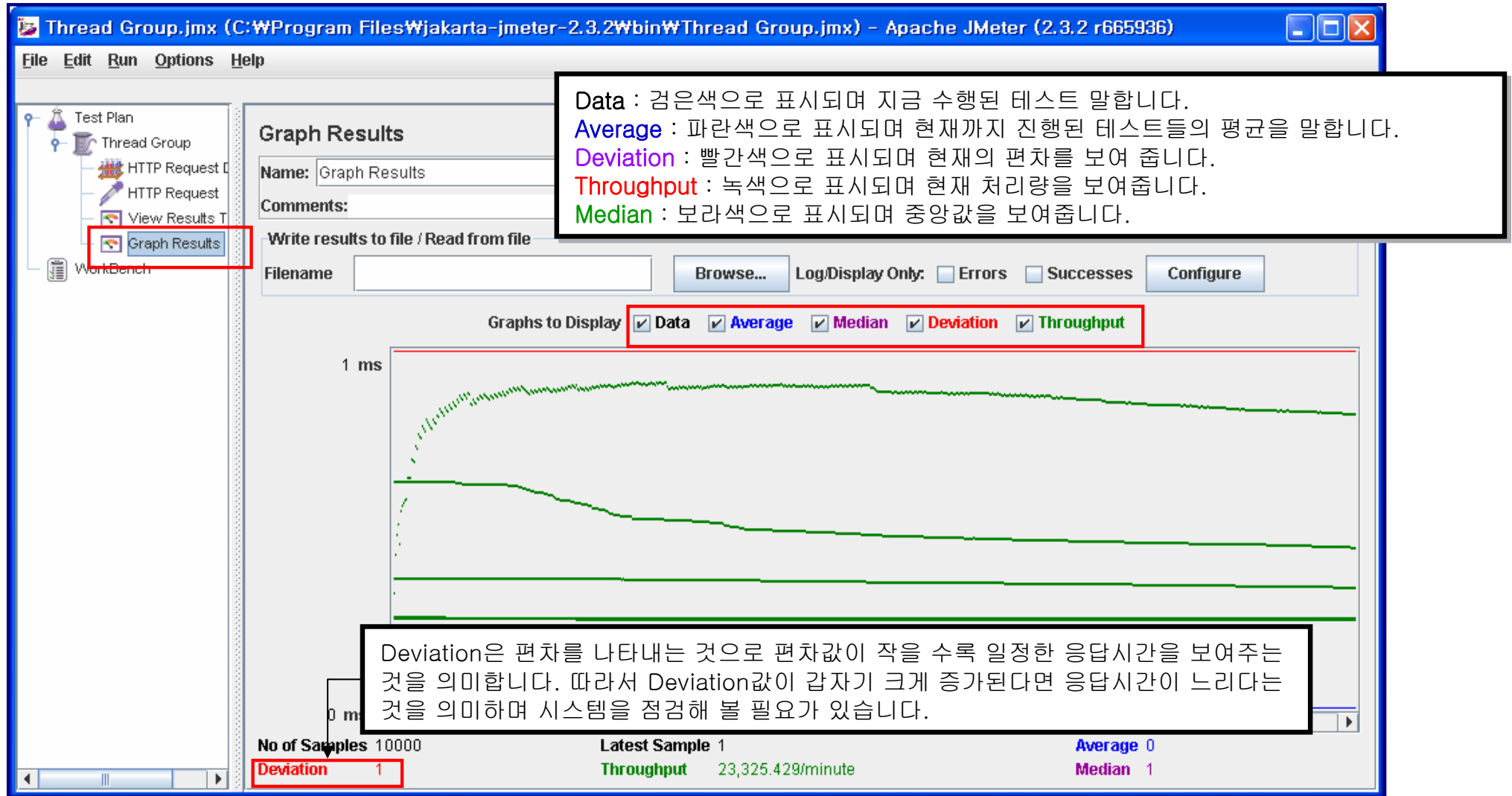
- Start 클릭 후 왼쪽 트리에서 View Results Tree 를 선택합니다.
- 선택 후 오른쪽에서 HTTP Request를 클릭하면 테스트 결과가 출력 됩니다.



3. 도구 기능 소개

3.4 웹 페이지 부하 테스트 기능 [12/12]

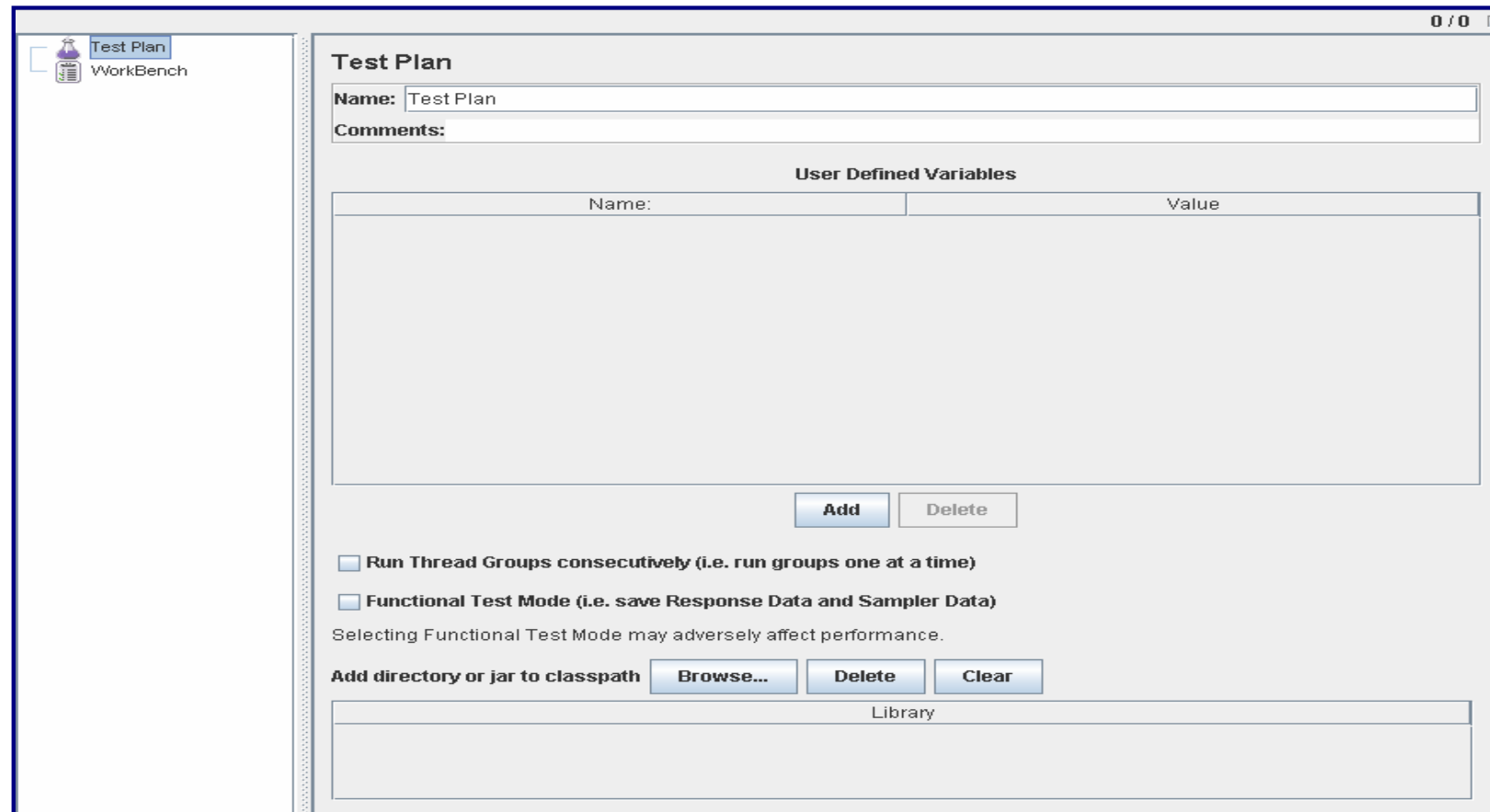
- 왼쪽 트리에서 Graph Results를 선택하면 아래와 같은 화면을 볼 수 있습니다.



3. 도구 기능 소개

3.5 Response Assertion 테스트 기능 (1/12)

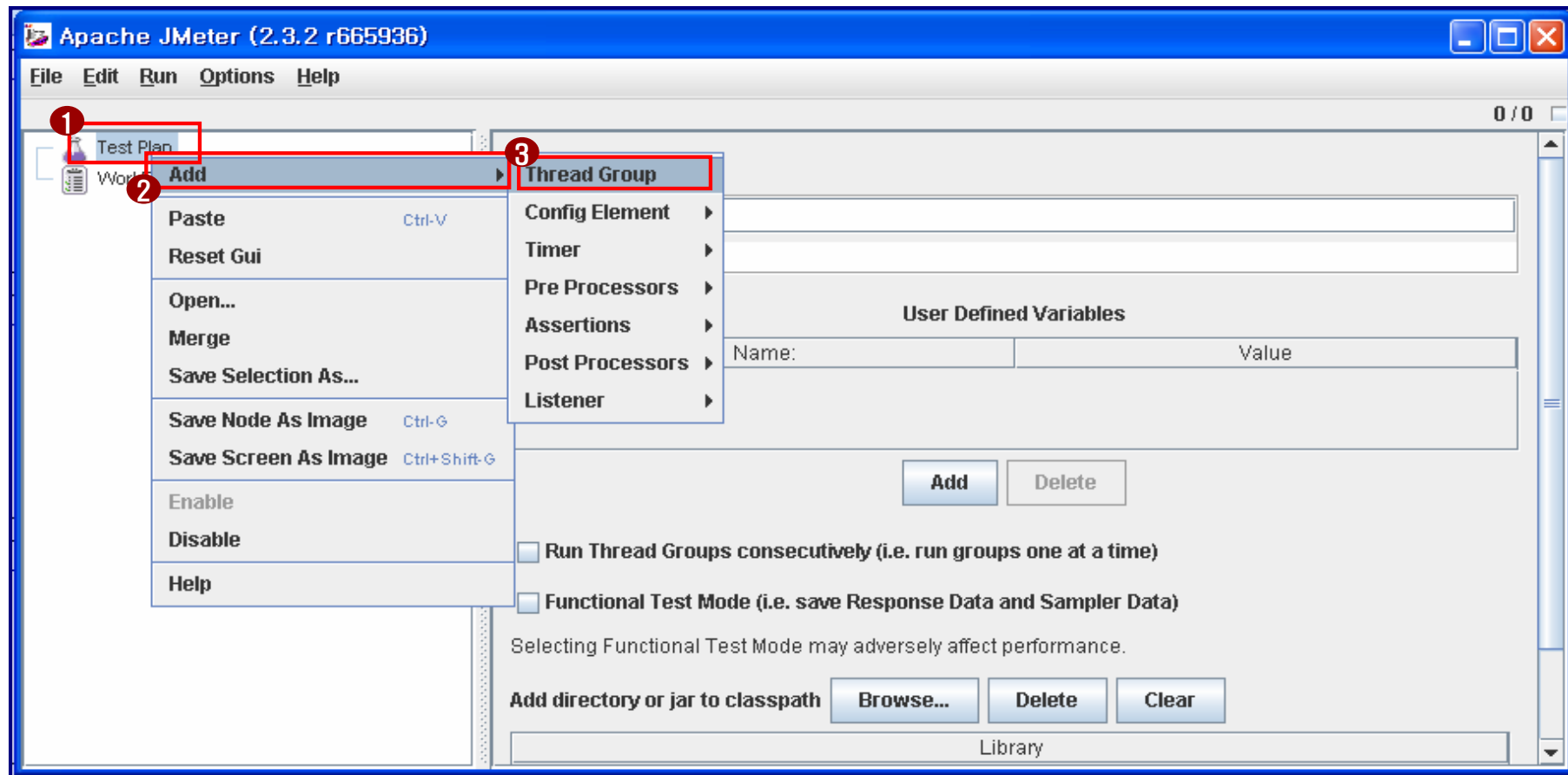
- Response Assertion이란 해당 요청의 응답 결과가 지정한 문자열을 포함하고 있는지의 여부를 검증할 때 사용하는 Assertion입니다.
- 처음에 할 일은 테스트 플랜을 작성하는 것입니다.
 - 16페이지를 참조합니다.



3. 도구 기능 소개

3.5 Response Assertion 테스트 기능 (2/12)

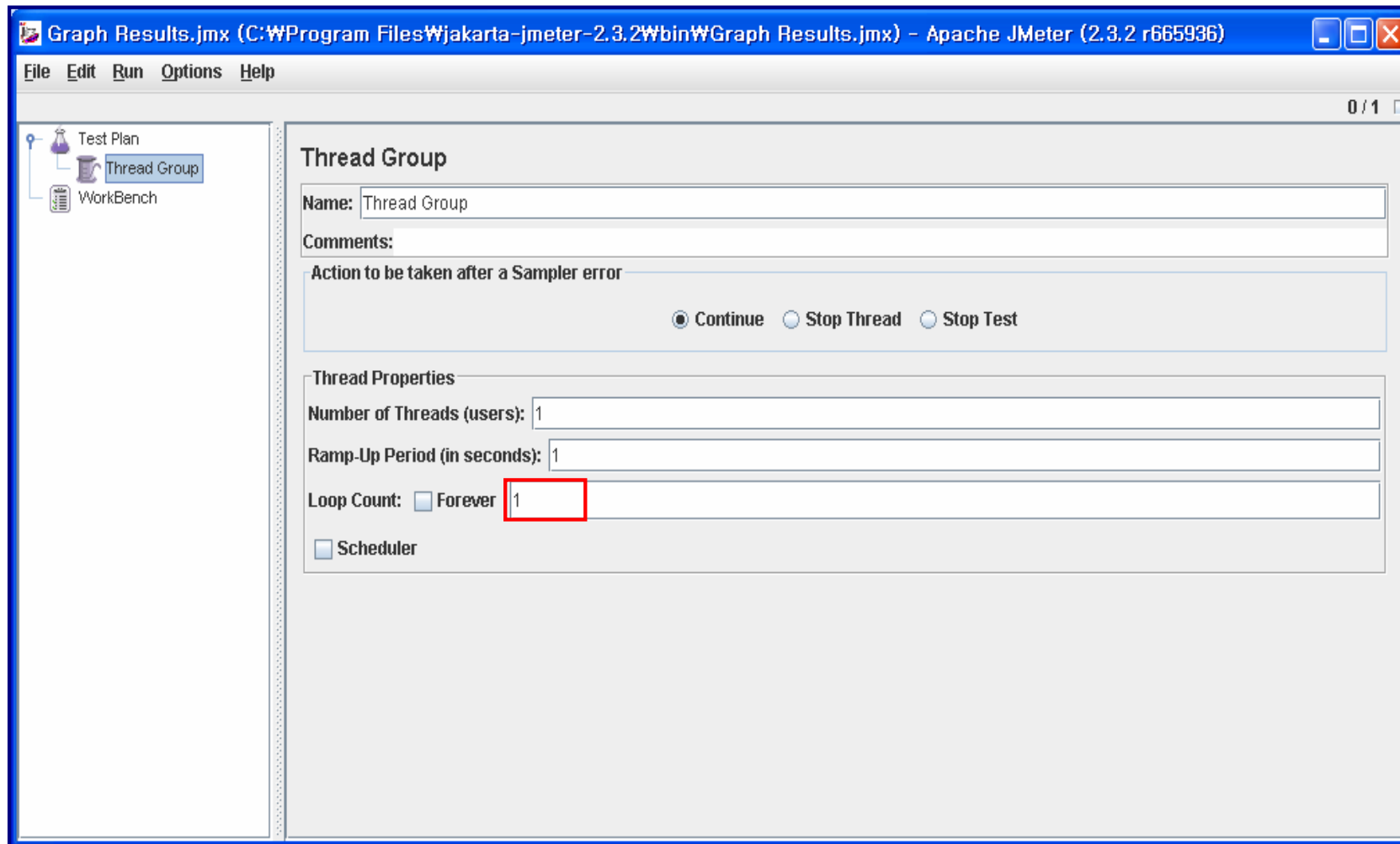
- Thread Group을 만듭니다.
 - Test Plan 선택 후 마우스 오른쪽 버튼 -> Add -> Thread Group 선택



3. 도구 기능 소개

3.5 Response Assertion 테스트 기능 (3/12)

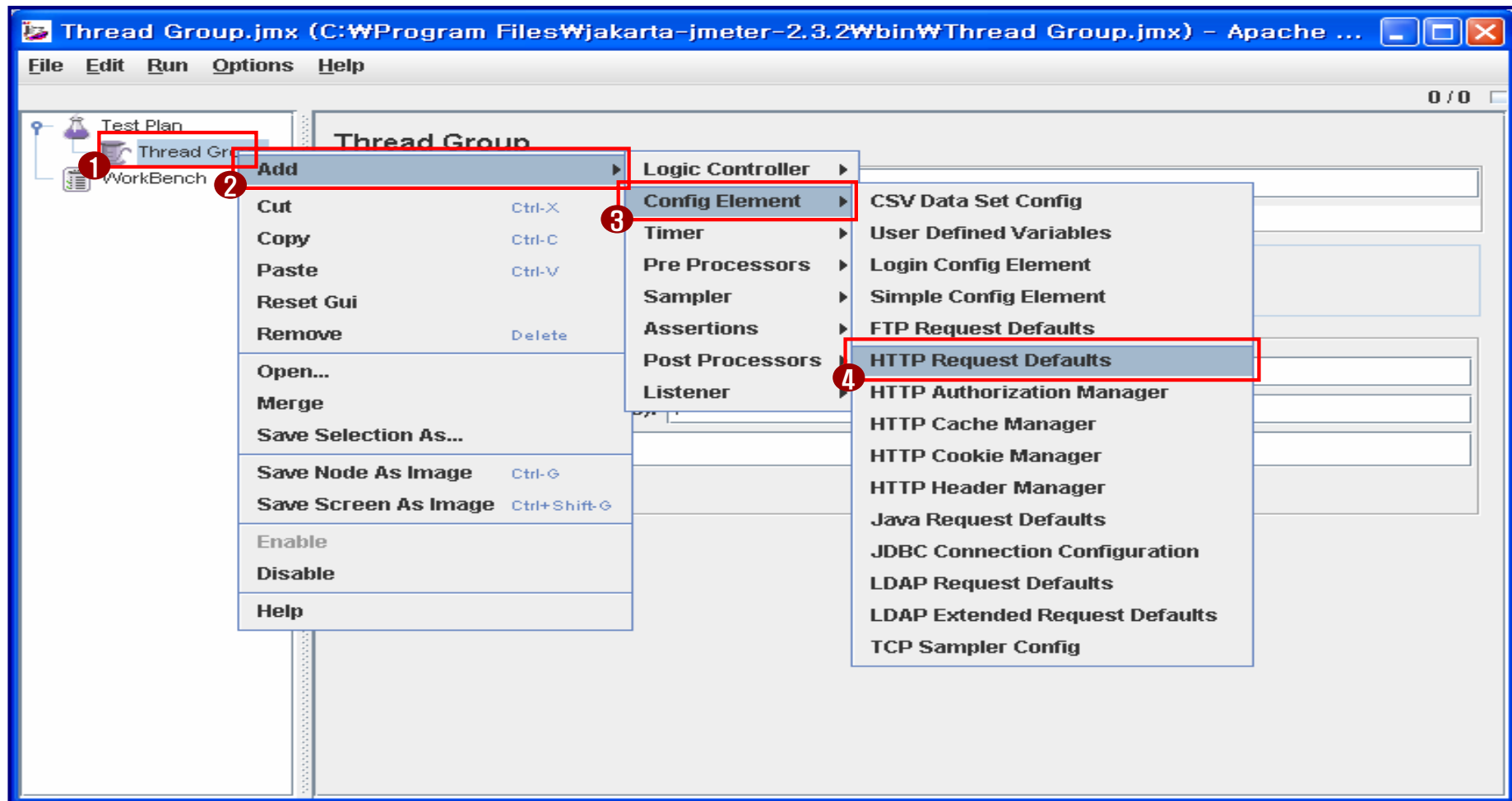
- 오른쪽에 나오는 항목들을 작성 합니다.
 - Loop Count 부분에서 1을 입력합니다.
 - 나머지 부분은 18페이지를 참조하여 작성합니다.



3. 도구 기능 소개

3.5 Response Assertion 테스트 기능 (4/12)

- Http Request Defaults를 추가 합니다.
 - Thread Group 선택 후 오른쪽 버튼 -> Add -> Config Element -> Http Request Defaults



3. 도구 기능 소개

3.5 Response Assertion 테스트 기능 (5/12)

- 아래의 항목들을 20페이지를 참고하여 작성합니다.

HTTP Request Defaults

Name: HTTP Request Defaults

Comments:

Web Server

Server Name or IP: localhost Port Number:

HTTP Request

Protocol (default http): Content encoding:

Path: form.html

Send Parameters With the Request:

Name:	Value	Encode?	Include Equ...

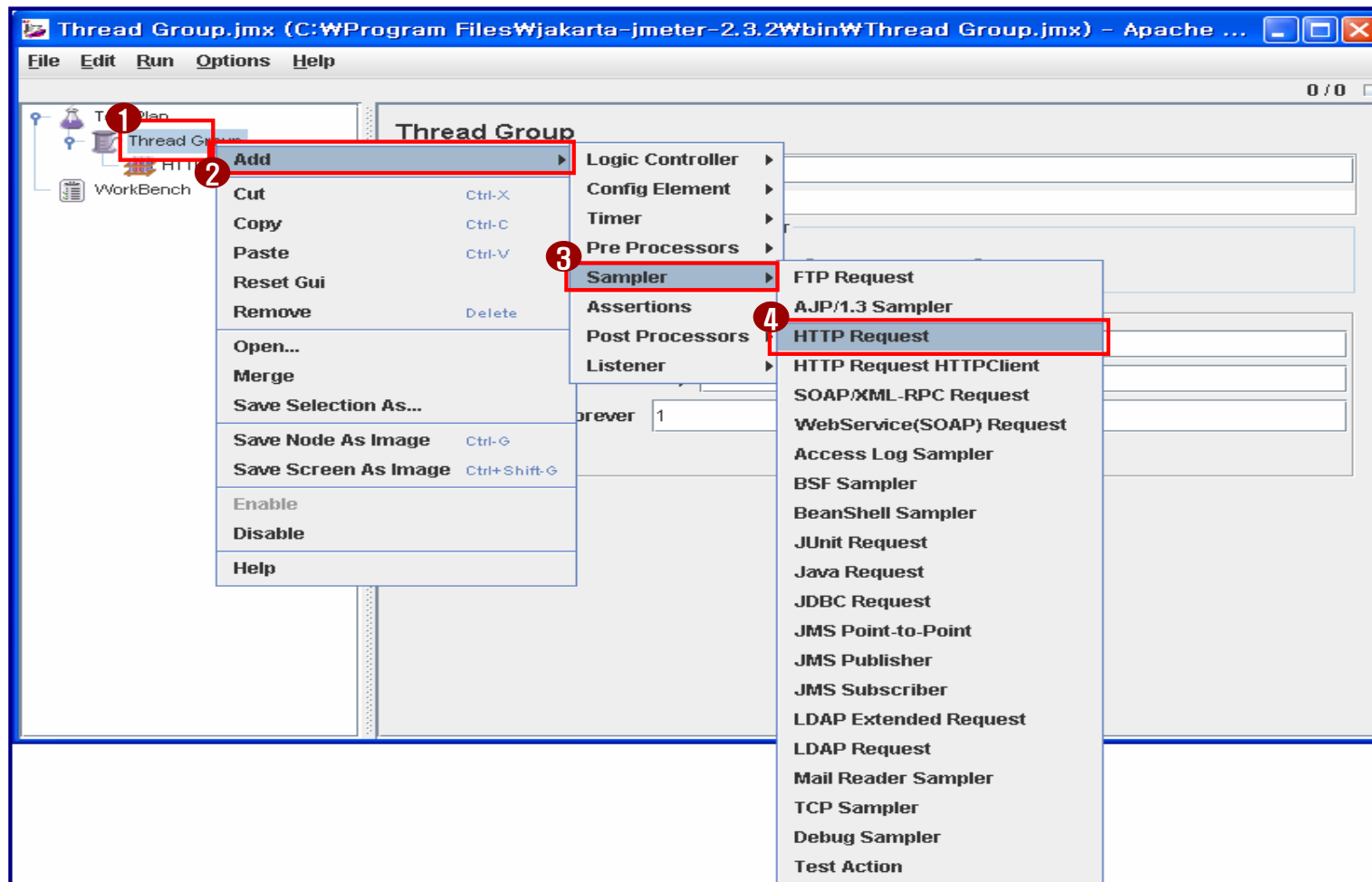
Add Delete

☐ Retrieve All Embedded Resources from HTML Files

3. 도구 기능 소개

3.5 Response Assertion 테스트 기능 (6/12)

- Http Request를 추가 합니다.
 - Thread Group선택 후 마우스 오른쪽 버튼 -> Add -> Sampler -> HTTP Request 선택



3. 도구 기능 소개

3.5 Response Assertion 테스트 기능 (7/12)

- 22페이지를 참고하여 Http Request를 작성합니다.
 - Http Request에 작성되지 않은 부분은 자동으로 Http Request Defaults에 작성된 내용으로 대체됩니다.

HTTP Request

Name: HTTP Request

Comments:

Web Server

Server Name or IP: Port Number:

HTTP Request

Protocol (default http): Method: GET Content encoding:

Path:

☒ Redirect Automatically ☐ Follow Redirects ☒ Use KeepAlive ☐ Use multipart/form-data for HTTP POST

Send Parameters With the Request:

Name:	Value	Encode?	Include Equ...

Add Delete

Send Files With the Request:

File Path:	Parameter ...	MIME Type:

Add Browse... Delete

Optional Tasks

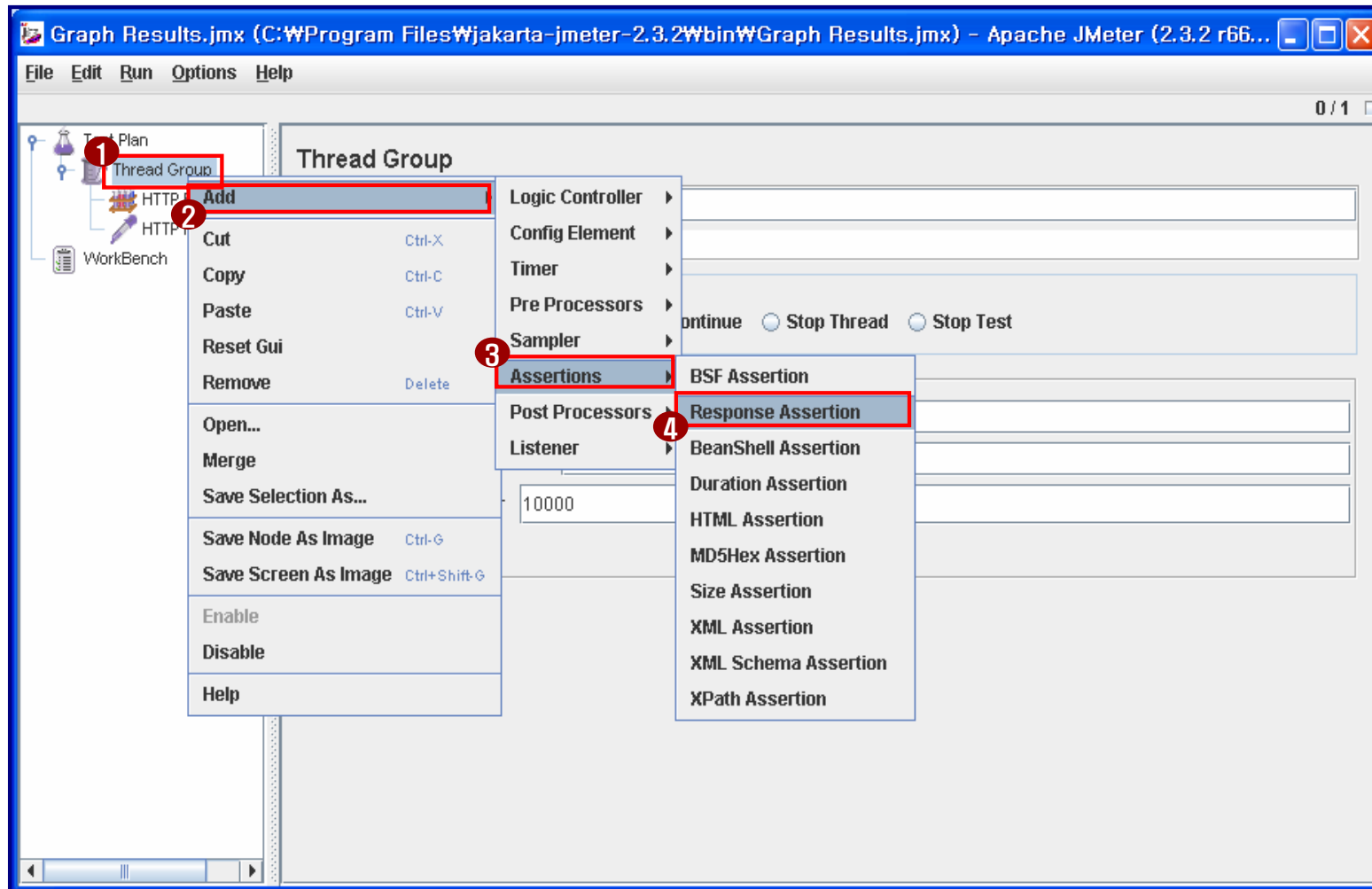
☐ Retrieve All Embedded Resources from HTML Files ☐ Use as Monitor ☐ Save response as MD5 hash?

Embedded URLs must match:

3. 도구 기능 소개

3.5 Response Assertion 테스트 기능 (8/12)

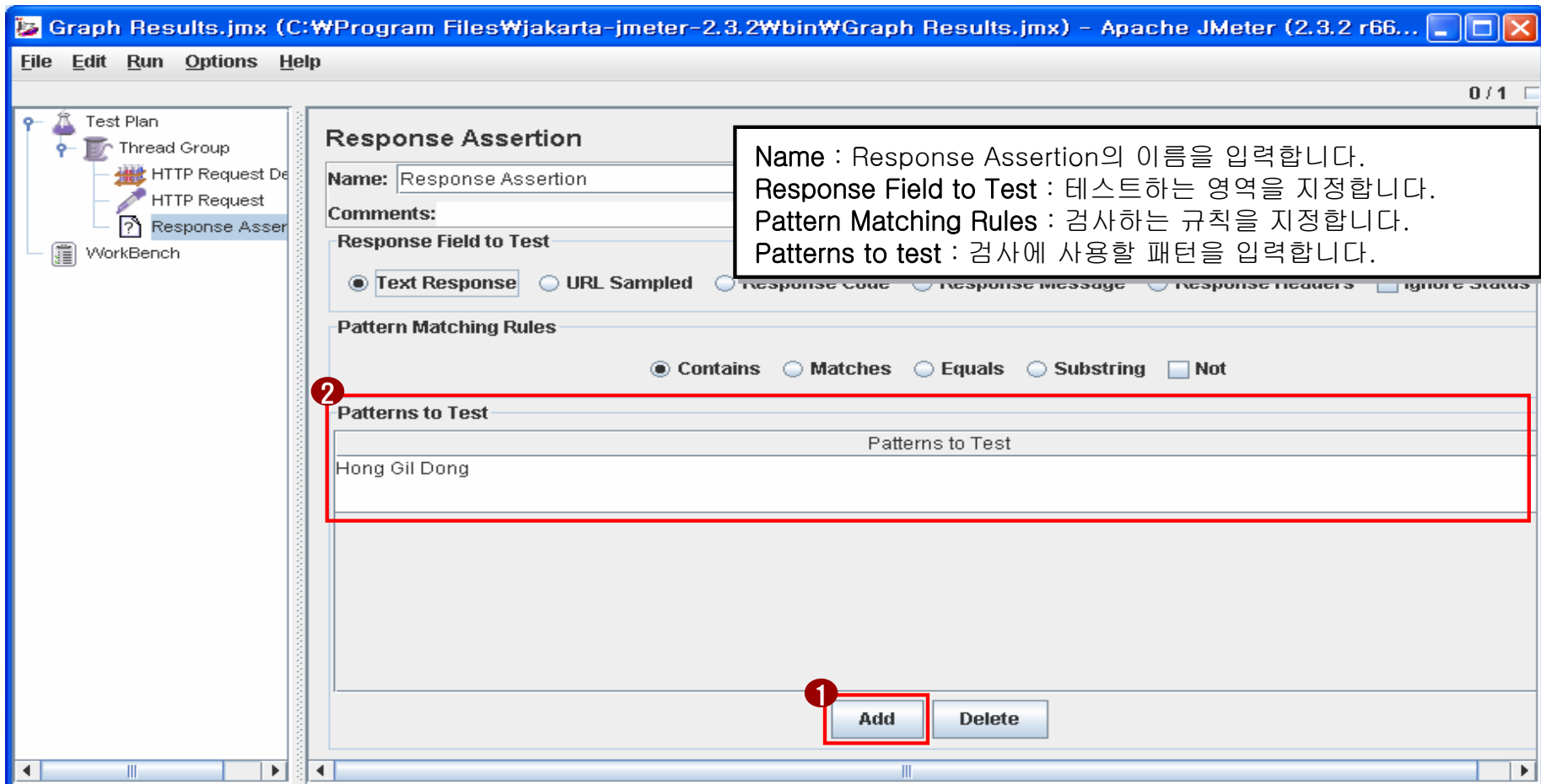
- Response Assertion을 추가합니다.
 - Thread Group 선택 후 마우스 오른쪽 버튼 -> Add -> Assertions -> Response Assertion 수행



3. 도구 기능 소개

3.5 Response Assertion 테스트 기능 (9/12)

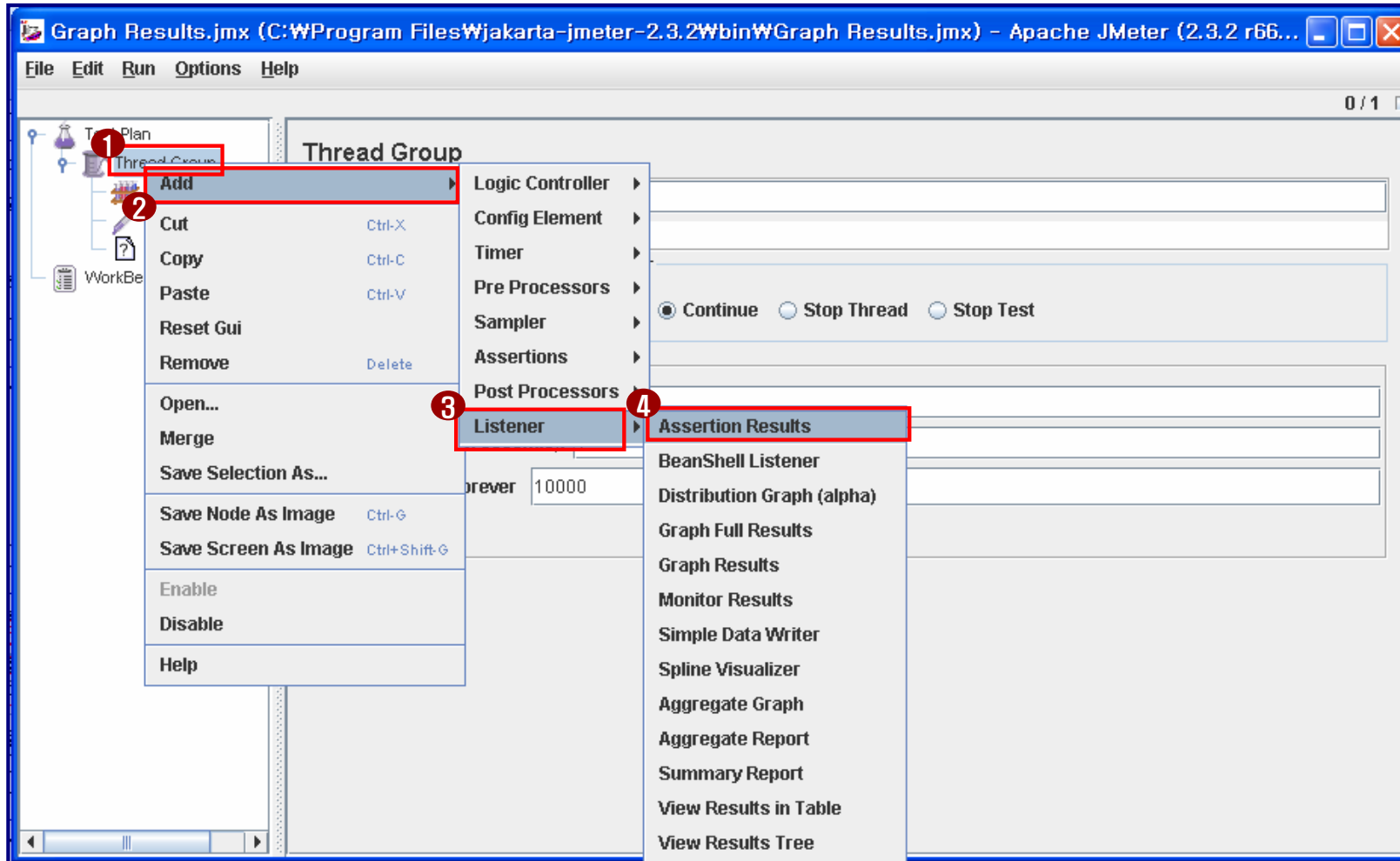
- Response Assertion 항목들을 입력합니다.
- 검사할 문자열을 입력합니다.
 - Add 버튼 클릭 -> Pattern to Test에 검사할 문자열 입력
 - 본 매뉴얼에서는 테스트 웹 페이지에 Hong Gil Dong이라는 단어가 존재하는 지 검사할 것입니다.



3. 도구 기능 소개

3.5 Response Assertion 테스트 기능 (10/12)

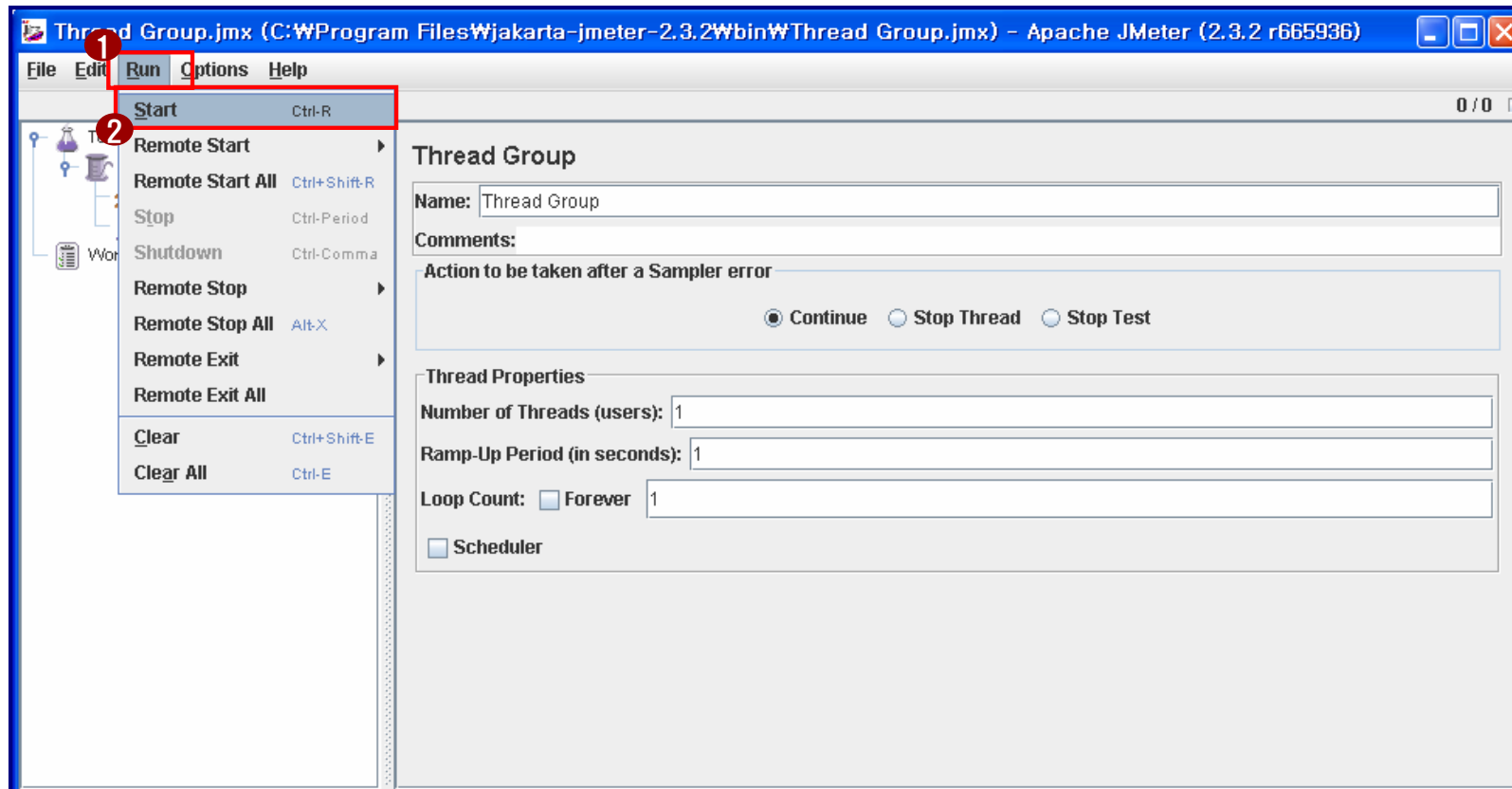
- Assertion 테스트 결과를 보기 위해서 Listener를 추가 합니다.
 - Thread Group선택 후 마우스 오른쪽 버튼 -> Add -> Listener -> Assertion Results 선택



3. 도구 기능 소개

3.5 Response Assertion 테스트 기능 [11/12]

- 마지막으로 해야 할 일은 테스트를 실행 하는 것입니다.
 - 메뉴 바의 Run -> Start



3. 도구 기능 소개

3.5 Response Assertion 테스트 기능 [12/12]

- 테스트 실행 후 왼쪽 Assertion Results를 클릭하면 다음과 같은 화면이 출력 됩니다.

● Write results to file/read from file : 결과를 파일로 작성하거나 기존에 있던 결과를 읽어들이는 부분입니다.

● Log/Display only : 결과창의 출력형식을 지정합니다.(Errors/Successes)

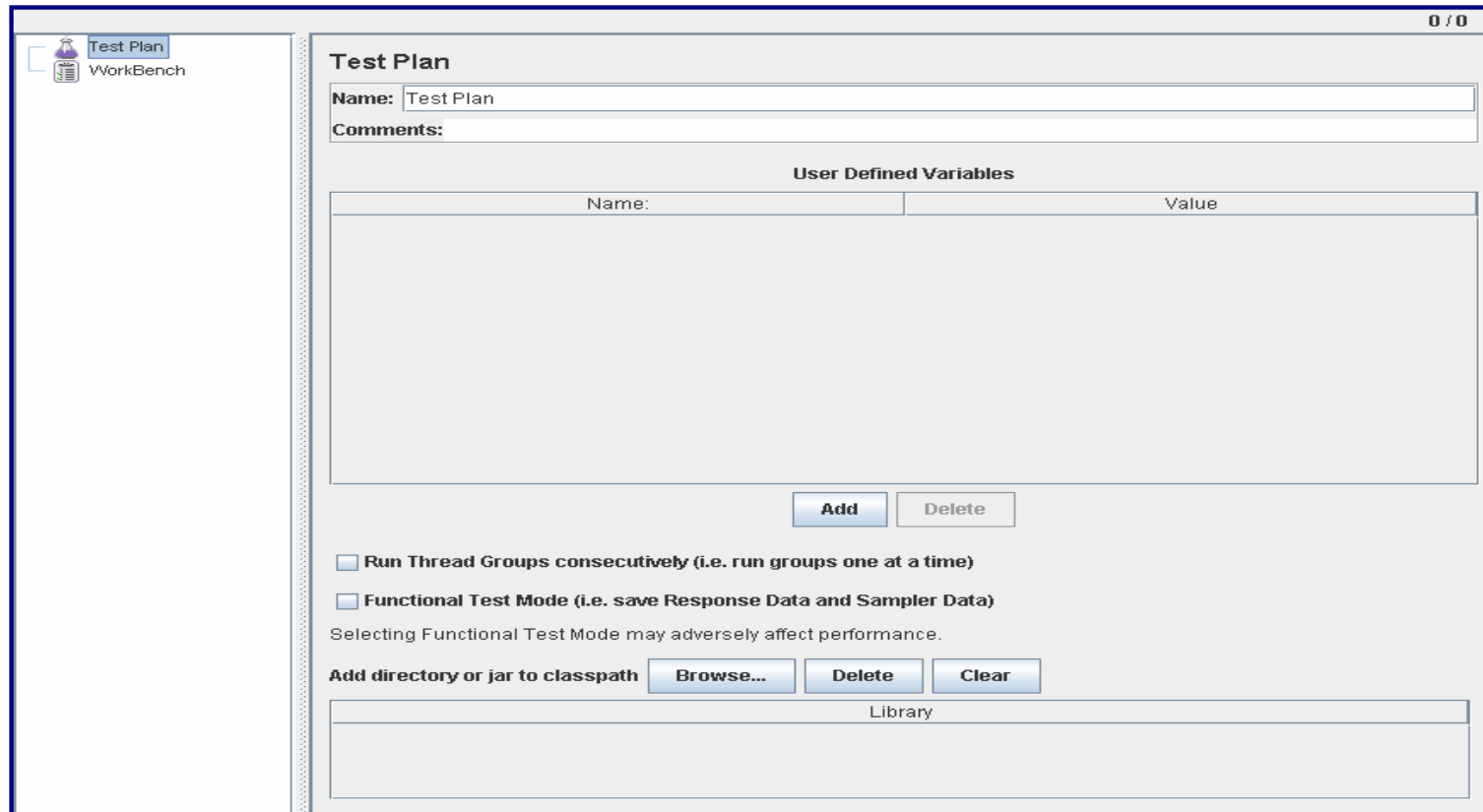
● Configure : 결과창의 출력형식에 대해서 좀 더 자세하게 보기 위해서는 이 부분에서 환경설정을 합니다.

테스트 결과를 보여주는 창입니다.
지금 출력된 부분을 설명하면 테스트 웹 페이지에는 요청한 Hong Gil Dong이라는 문자열이 없다는 것을 보여주고 있습니다.

3. 도구 기능 소개

3.6 Duration Assertion 테스트 기능 (1/12)

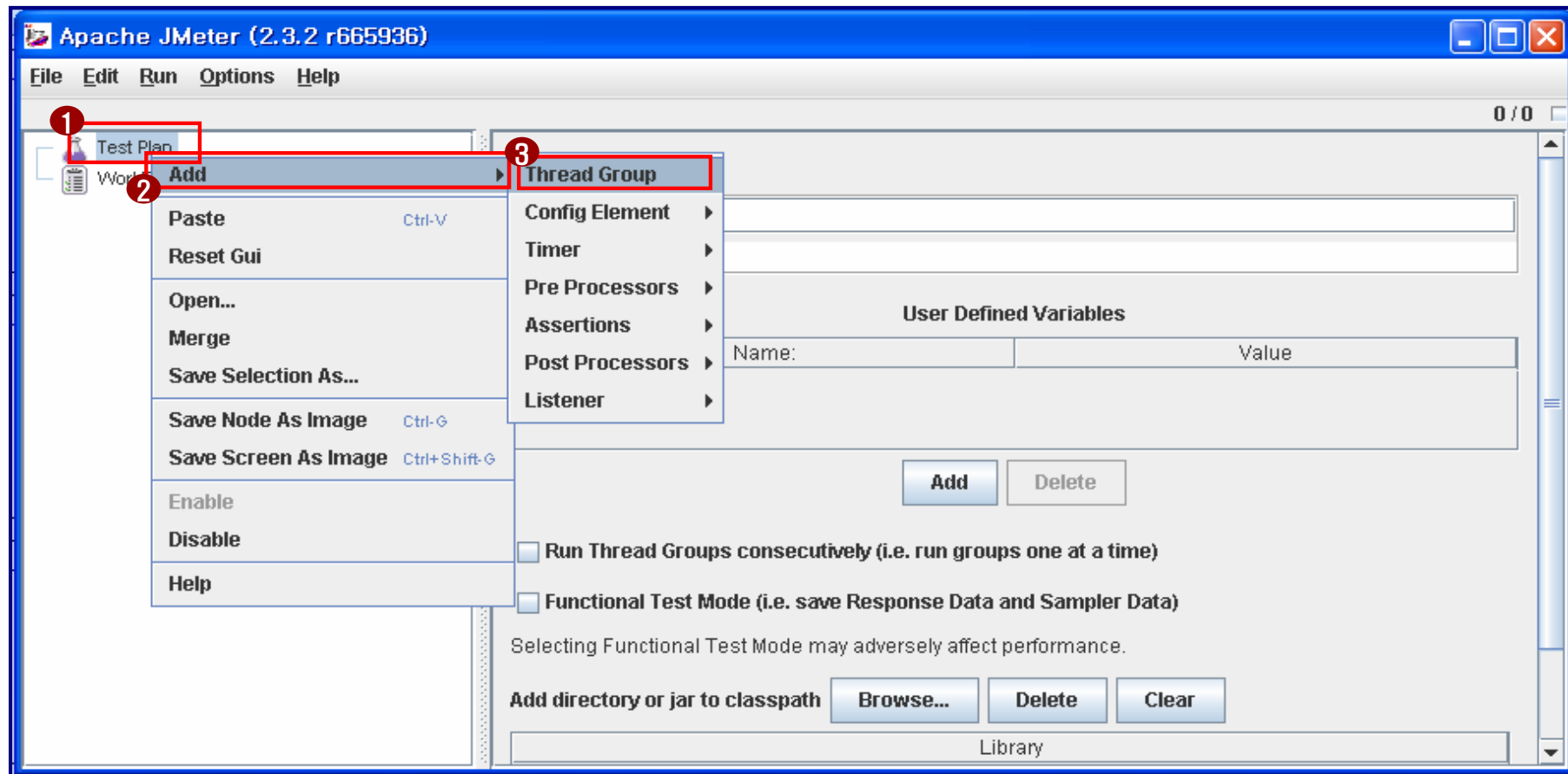
- Duration Assertion이란 Http Request가 지정한 시간 내에 응답 결과를 리턴 하는지의 여부를 검증해 줍니다.
- 처음에 할 일은 테스트 플랜을 작성하는 것입니다.
 - 16페이지를 참고하여 테스트 플랜을 작성합니다.



3. 도구 기능 소개

3.6 Duration Assertion 테스트 기능 (2/12)

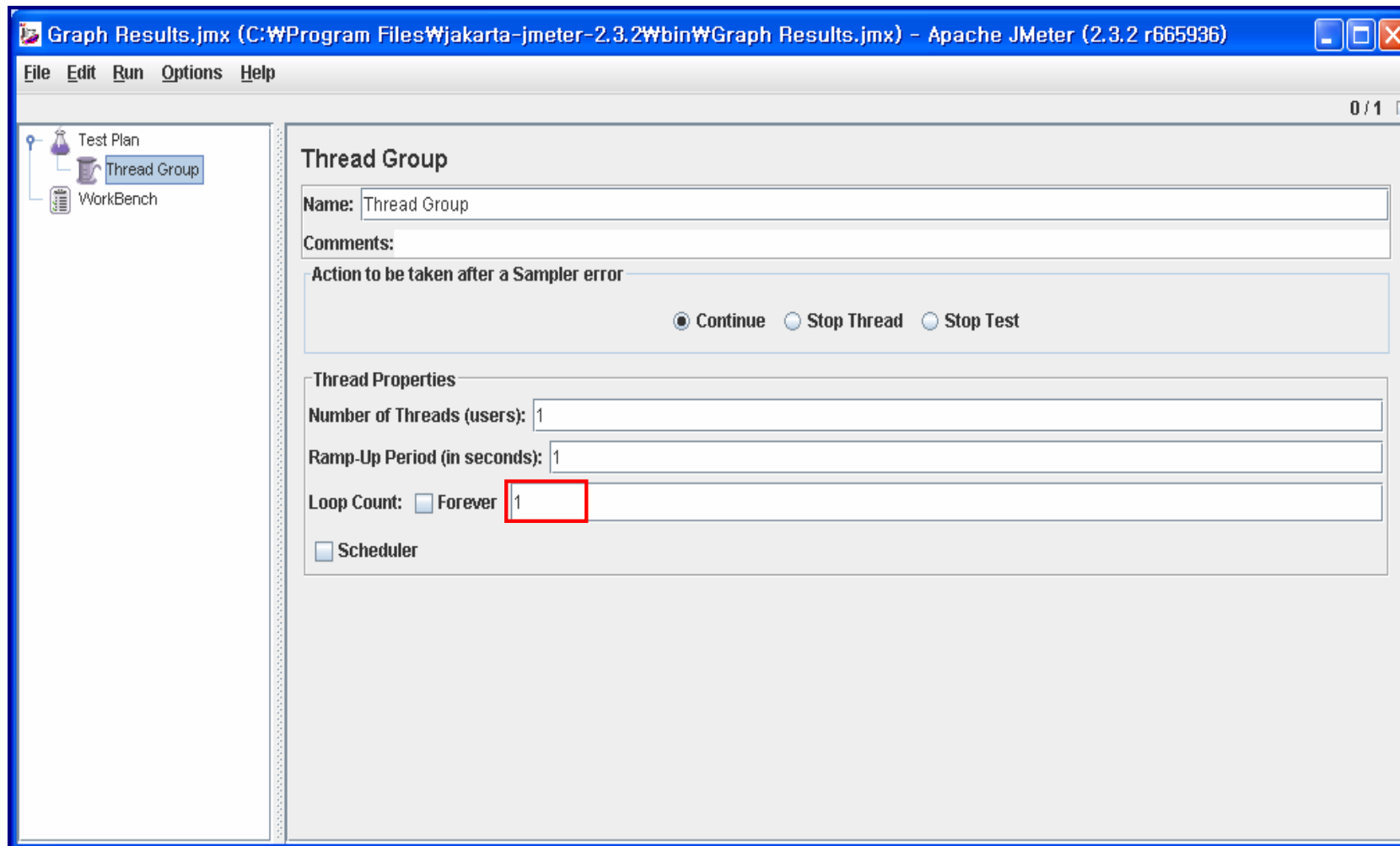
- Thread Group을 만듭니다.
 - 왼쪽 Test Plan 선택 후 마우스 오른쪽 버튼 -> Add -> Thread Group 선택



3. 도구 기능 소개

3.6 Duration Assertion 테스트 기능 (3/12)

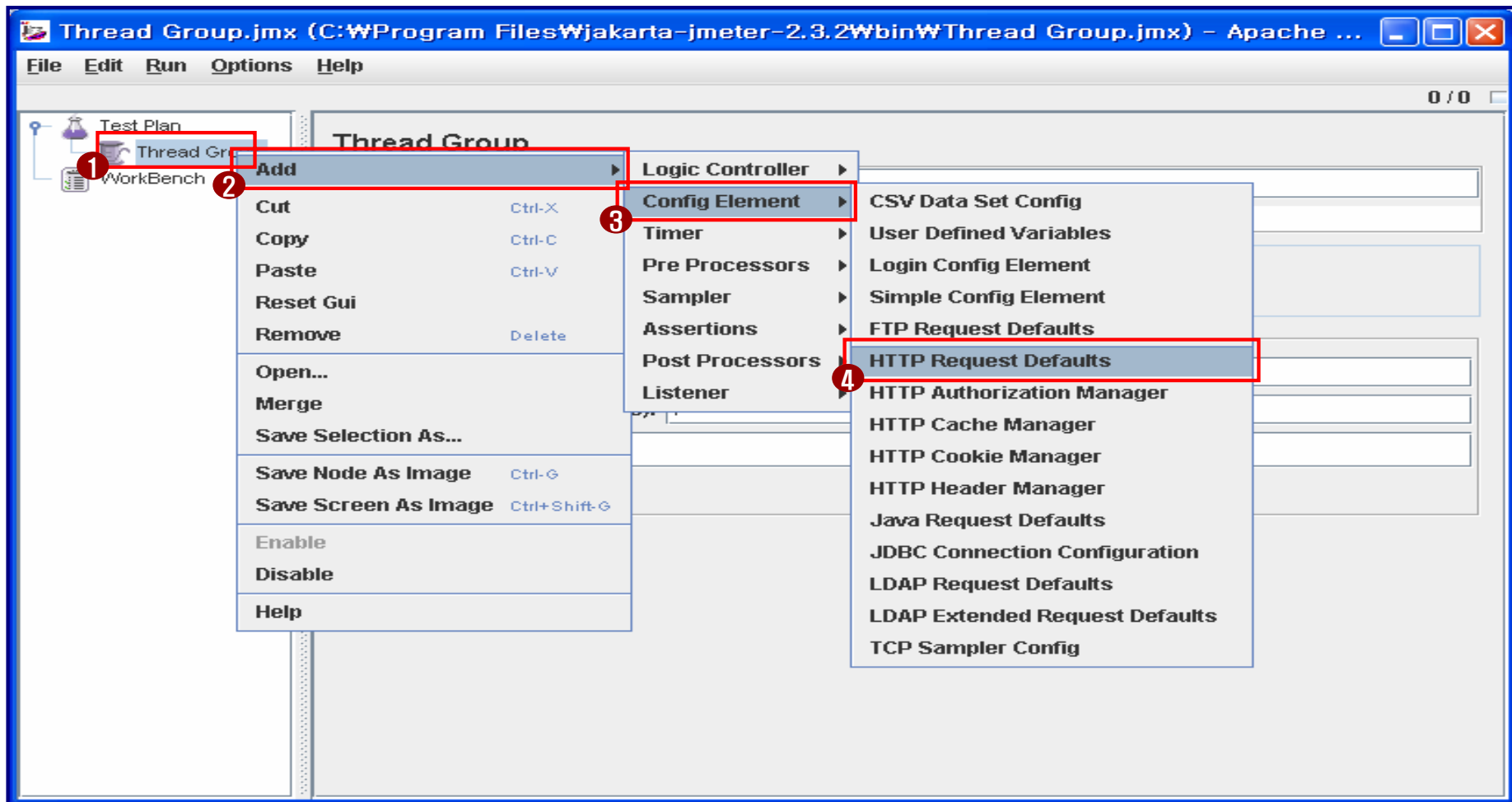
- 오른쪽에 나오는 항목들을 작성합니다.
 - 테스트를 한번 수행한다는 의미로 Loop Count 부분에 1을 입력합니다.
 - 나머지 항목은 18페이지를 참고하여 작성 합니다.



3. 도구 기능 소개

3.6 Duration Assertion 테스트 기능 (4/12)

- Http Request Defaults를 추가 합니다.
 - Thread Group 선택 후 마우스 오른쪽 버튼 -> Add -> Config Element -> Http Request Defaults



3. 도구 기능 소개

3.6 Duration Assertion 테스트 기능 (5/12)

- 아래의 항목들을 작성합니다.
 - 20페이지를 참고하여 아래의 항목들을 작성합니다.

HTTP Request Defaults

Name:

Comments:

Web Server

Server Name or IP: Port Number:

HTTP Request

Protocol (default http): Content encoding:

Path:

Send Parameters With the Request:

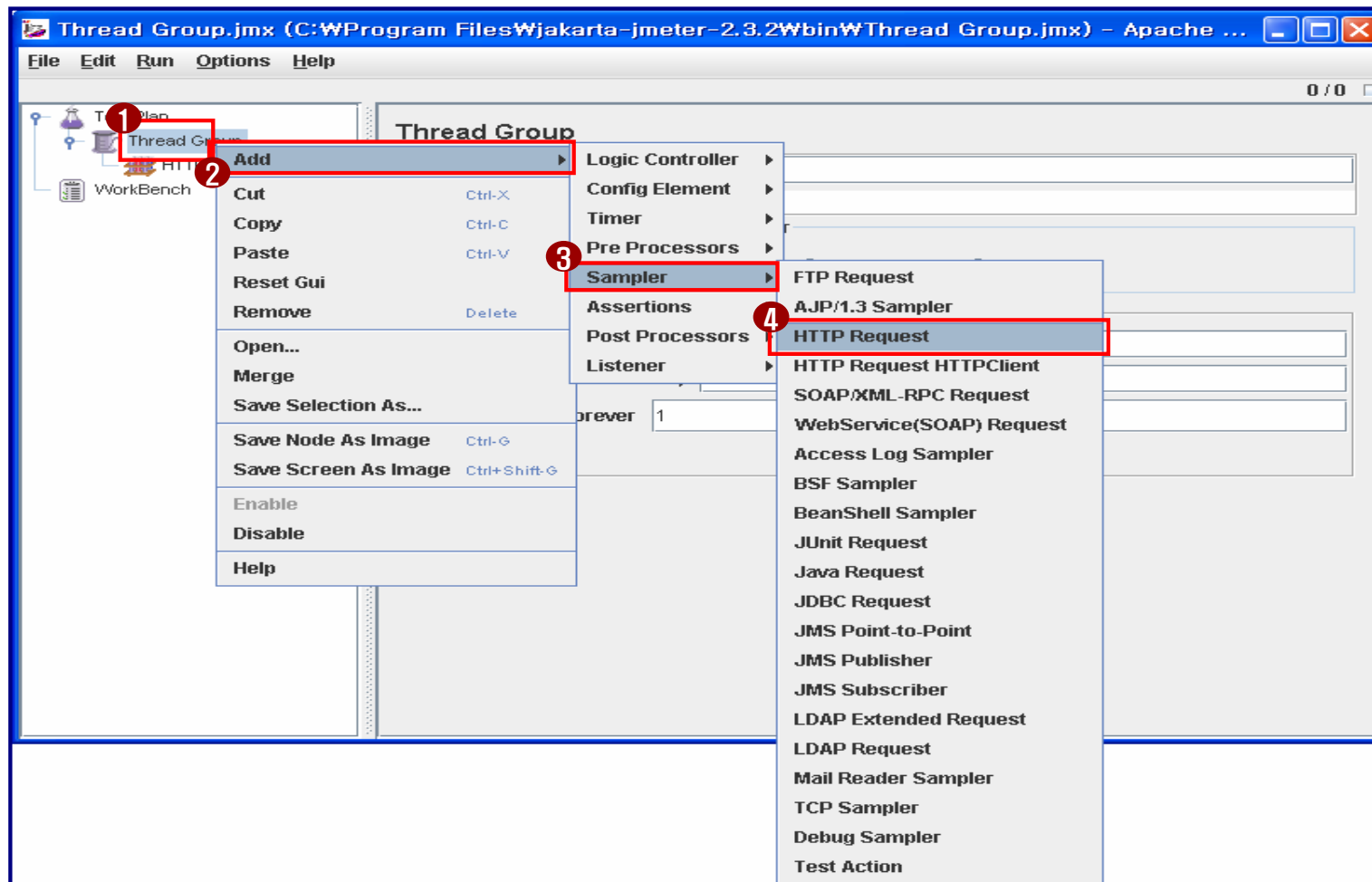
Name:	Value	Encode?	Include Equ...

☐ Retrieve All Embedded Resources from HTML Files

3. 도구 기능 소개

3.6 Duration Assertion 테스트 기능 (6/12)

- Http Request를 추가합니다.
 - Thread Group 선택 후 마우스 오른쪽 버튼 -> Add -> Sampler -> HTTP Request 선택



3. 도구 기능 소개

3.6 Duration Assertion 테스트 기능 (7/12)

- Http Request의 입력 정보는 다음과 같습니다.
 - 22페이지를 참고하여 Http Request를 작성합니다.

HTTP Request

Name:

Comments:

Web Server

Server Name or IP: Port Number:

HTTP Request

Protocol (default http): Method: Content encoding:

Path:

☒ Redirect Automatically ☐ Follow Redirects ☒ Use KeepAlive ☐ Use multipart/form-data for HTTP POST

Send Parameters With the Request:

Name:	Value	Encode?	Include Equ...

Send Files With the Request:

File Path:	Parameter ...	MIME Type:

Optional Tasks

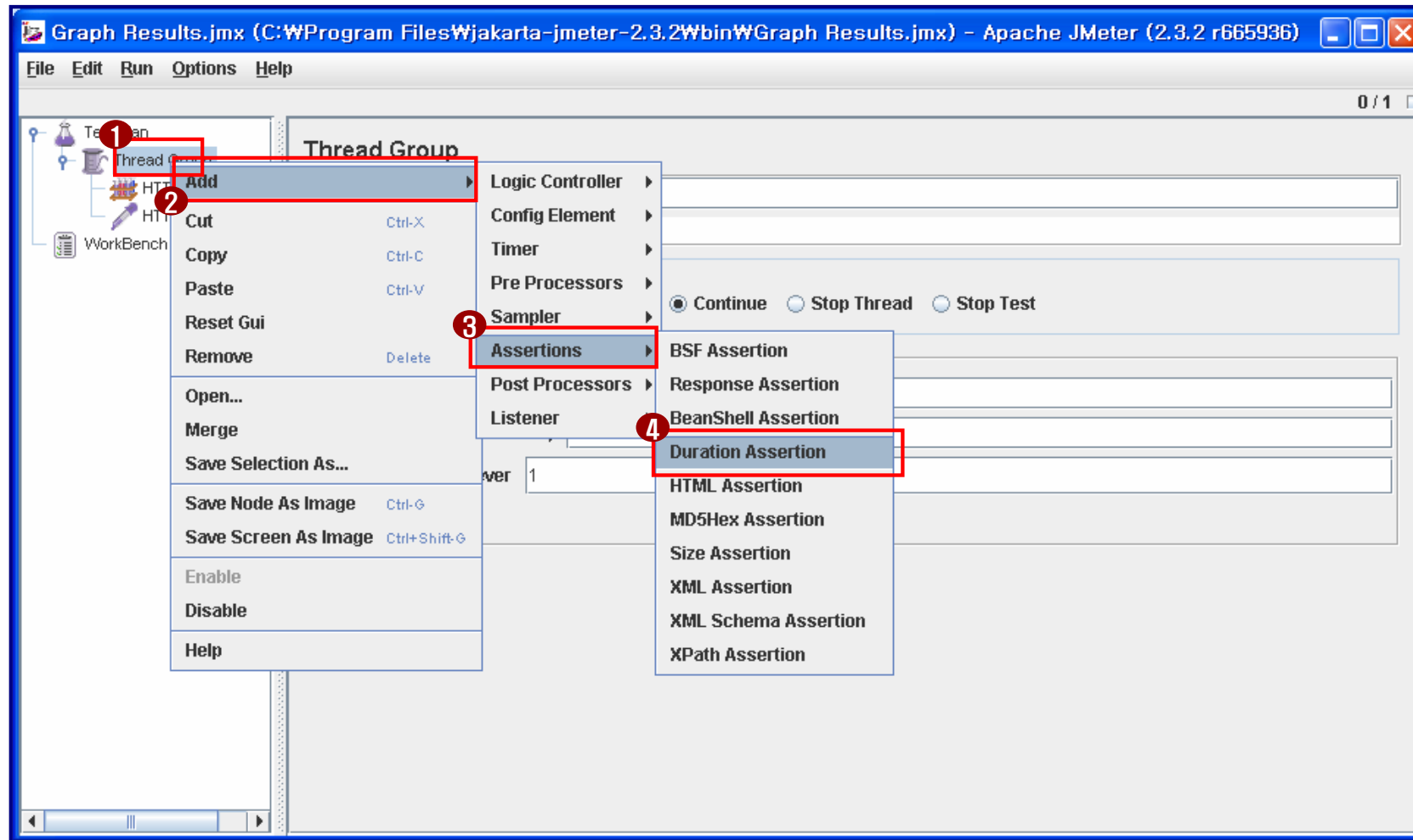
☐ Retrieve All Embedded Resources from HTML Files ☐ Use as Monitor ☐ Save response as MD5 hash?

Embedded URLs must match:

3. 도구 기능 소개

3.6 Duration Assertion 테스트 기능 (8/12)

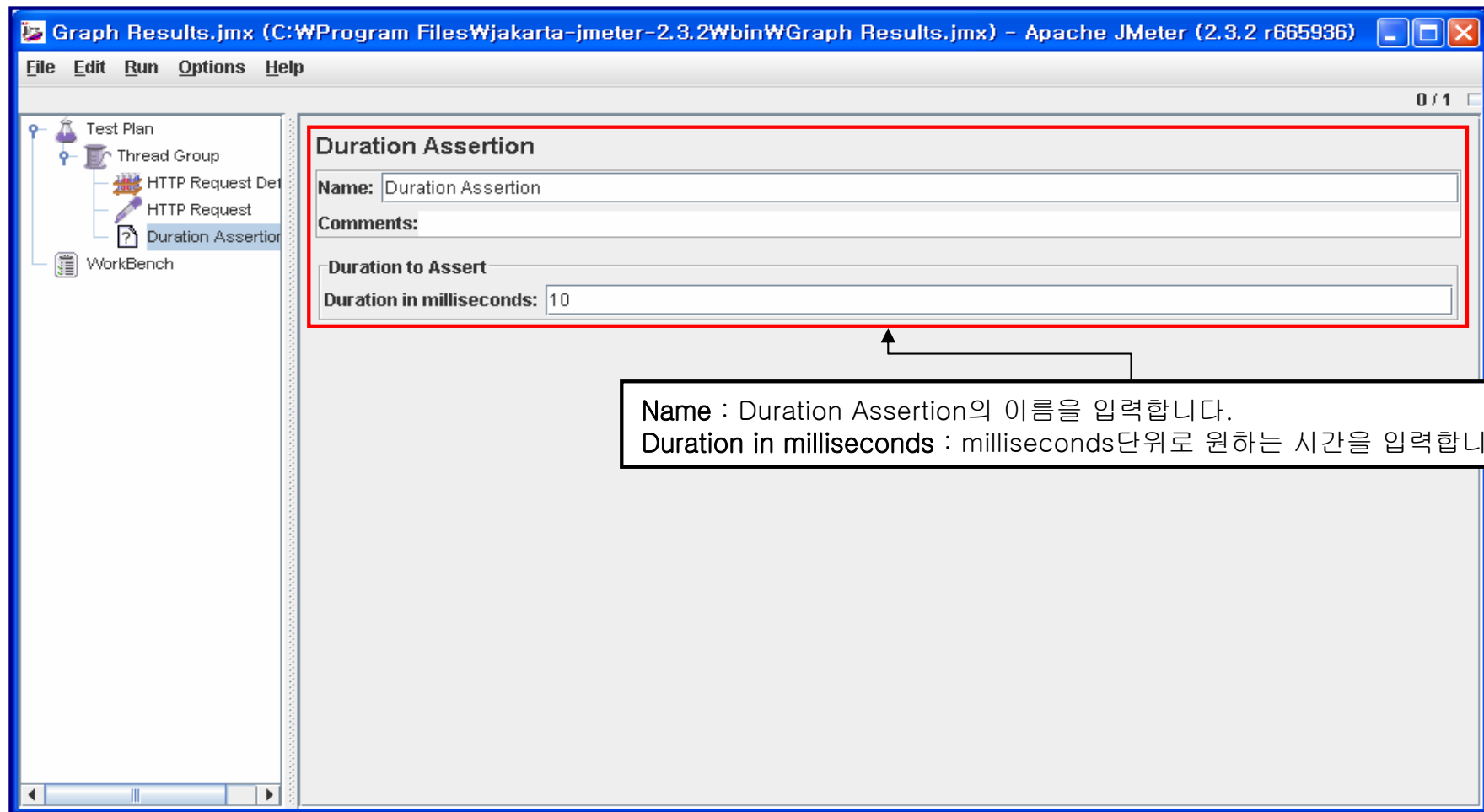
- Duration Assertion을 추가합니다.
 - Thread Group 선택 후 마우스 오른쪽 버튼 -> Add -> Assertions -> Duration Assertion 수행



3. 도구 기능 소개

3.6 Duration Assertion 테스트 기능 (9/12)

- Duration Assertion 항목을 입력 합니다.
- Duration to Assert 부분에 원하는 시간을 Milliseconds 단위로 입력 합니다.
 - 본 매뉴얼에서는 원하는 시간으로 10 milliseconds로 입력하였습니다.

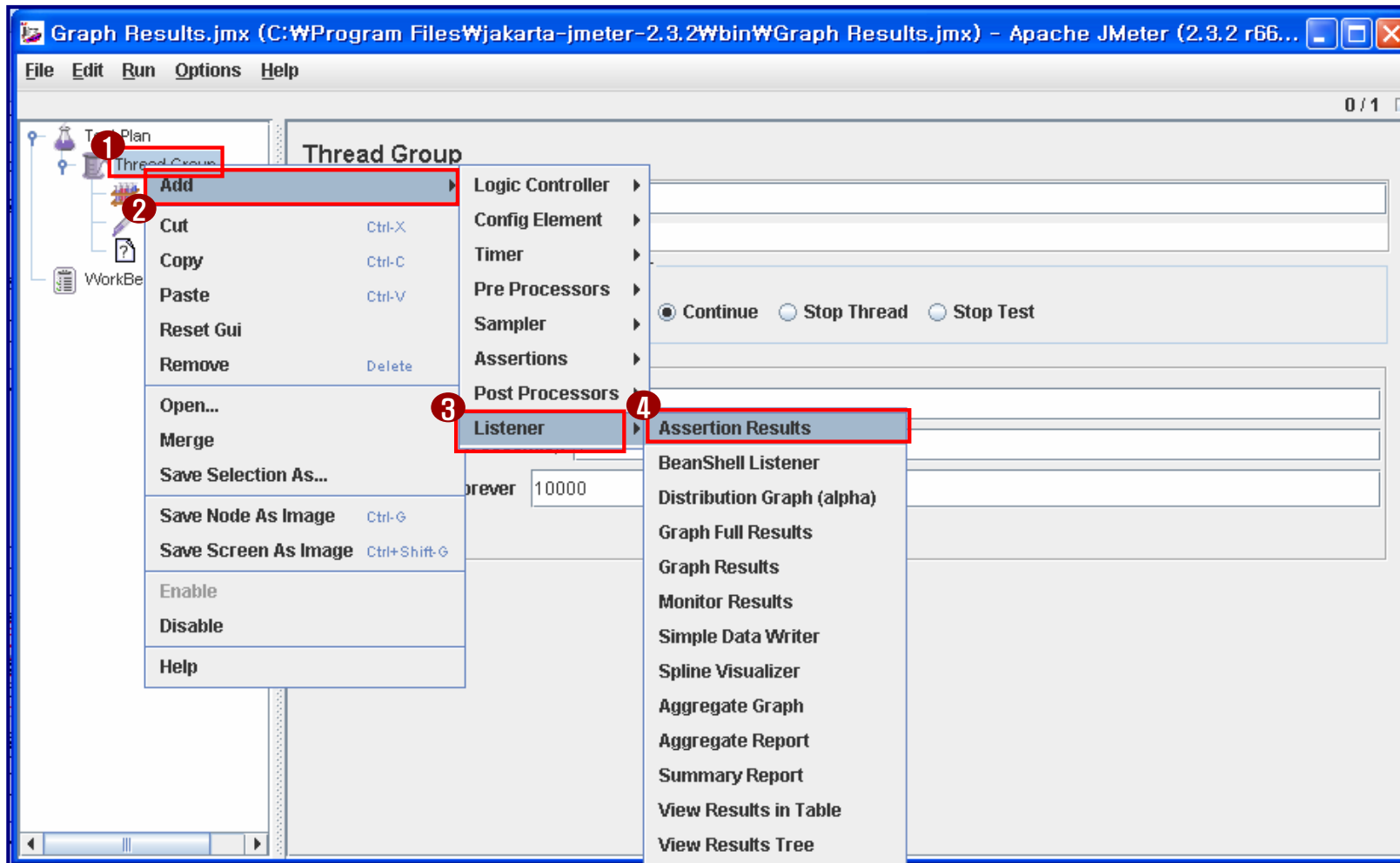


Name : Duration Assertion의 이름을 입력합니다.
Duration in milliseconds : milliseconds단위로 원하는 시간을 입력합니다.

3. 도구 기능 소개

3.6 Duration Assertion 테스트 기능 (10/12)

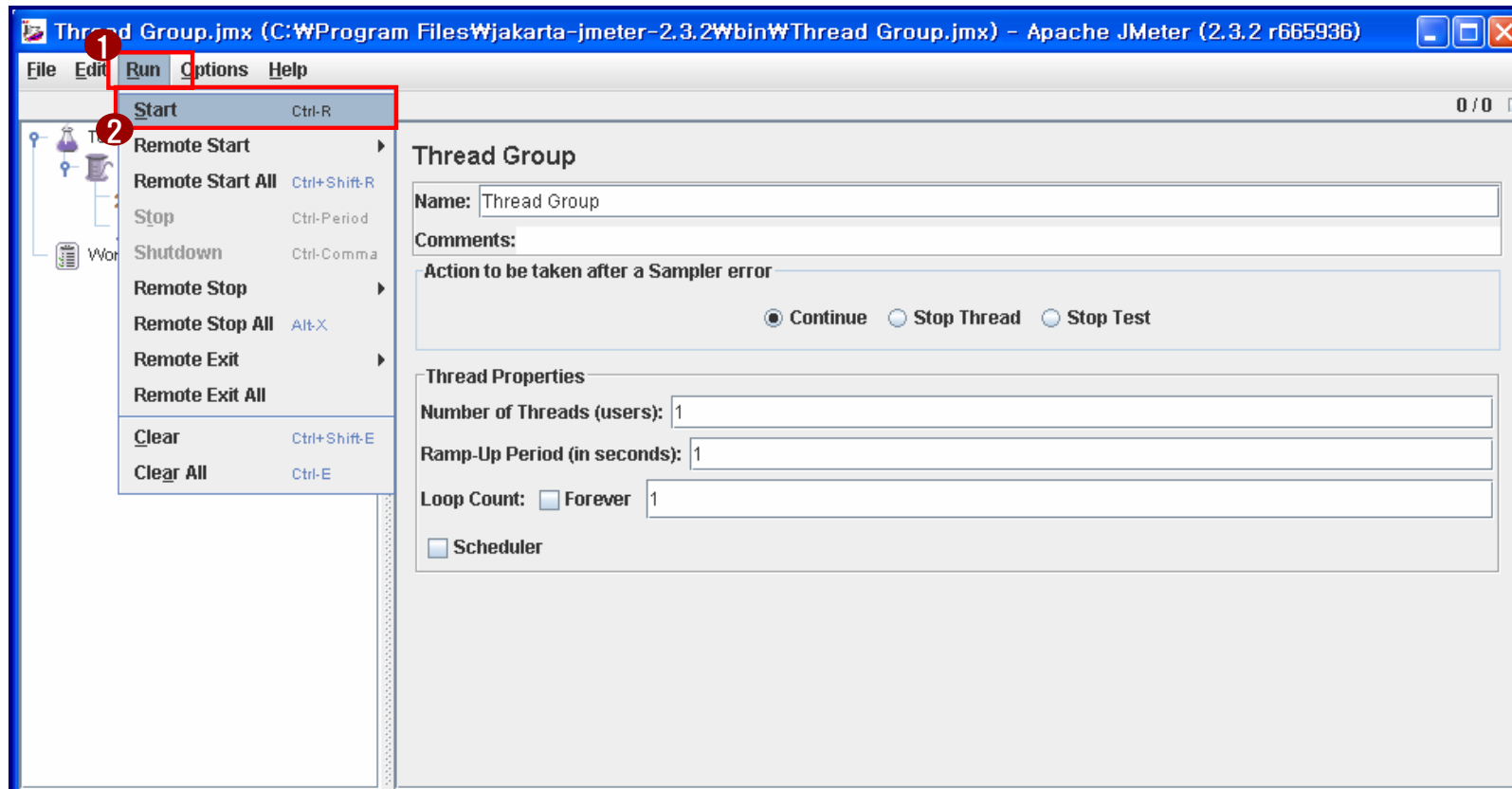
- Assertion 테스트 결과를 보기 위해 Listener를 추가 합니다.
 - Thread Group 선택 후 마우스 오른쪽 버튼 -> Add -> Listener -> Assertion Results 선택



3. 도구 기능 소개

3.6 Duration Assertion 테스트 기능 (11/12)

- 마지막으로 해야 할 일은 테스트를 실행 하는 것입니다.
 - 메뉴 바의 Run -> Start



3. 도구 기능 소개

3.6 Duration Assertion 테스트 기능 (12/12)

- 테스트 실행 후 왼쪽 Assertion Results를 클릭하면 다음과 같은 화면이 출력 됩니다.

● Write results to file/read from file : 결과를 파일로 작성하거나 기존에 있던 결과를 읽어 들이는 부분입니다.

● Log/Display only : 결과창의 출력형식을 지정합니다.(Errors/Successes)

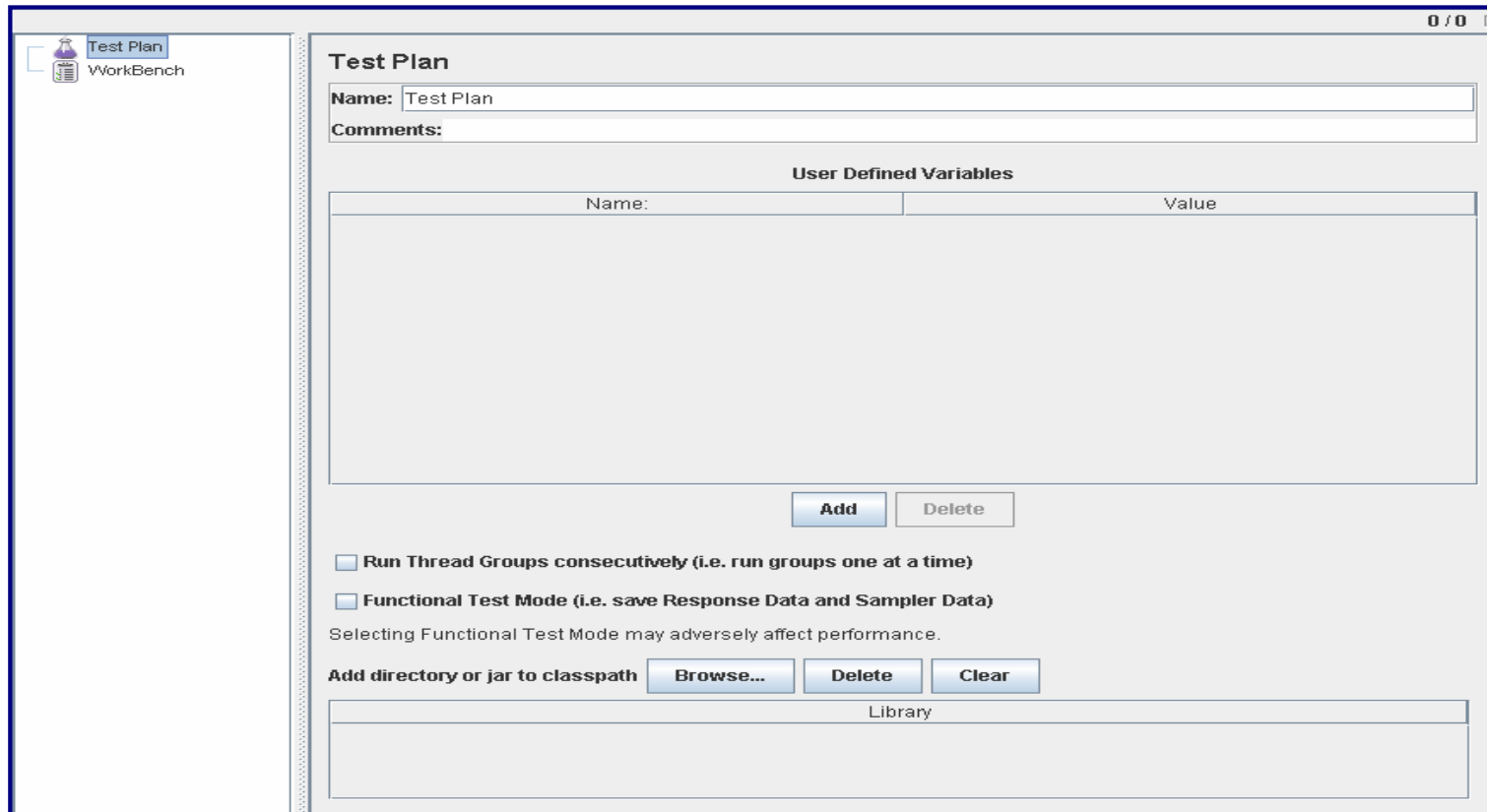
● Configure : 결과창의 출력형식에 대해서 좀 더 자세하게 보기 위해서는 이 부분에서 환경설정을 합니다.

테스트 결과를 보여주는 창입니다. 지금 출력된 부분을 설명하면 테스트 결과 설정된 시간 내에 응답결과가 성공적으로 왔다는 것을 보여주고 있습니다.

3. 도구 기능 소개

3.7 Size Assertion 테스트 기능 (1/12)

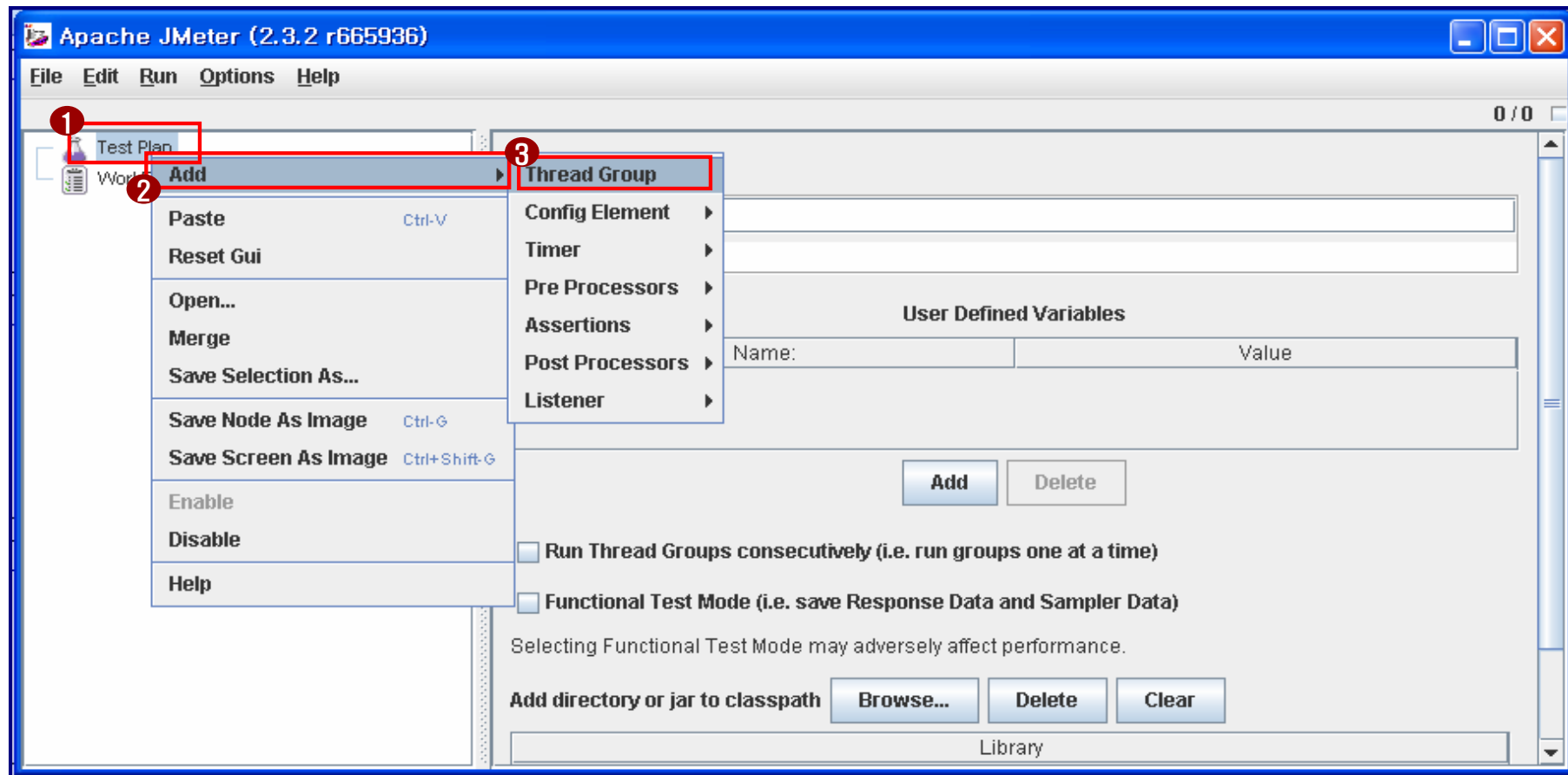
- Size Assertion이란 웹 페이지의 크기가 원래 목표한 웹 페이지의 크기를 준수하고 있는 지를 확인하는 검증입니다.
- 처음에 할 일은 테스트 플랜을 작성하는 것입니다.
 - 16페이지를 참고하여 테스트 플랜을 작성합니다.



3. 도구 기능 소개

3.7 Size Assertion 테스트 기능 (2/12)

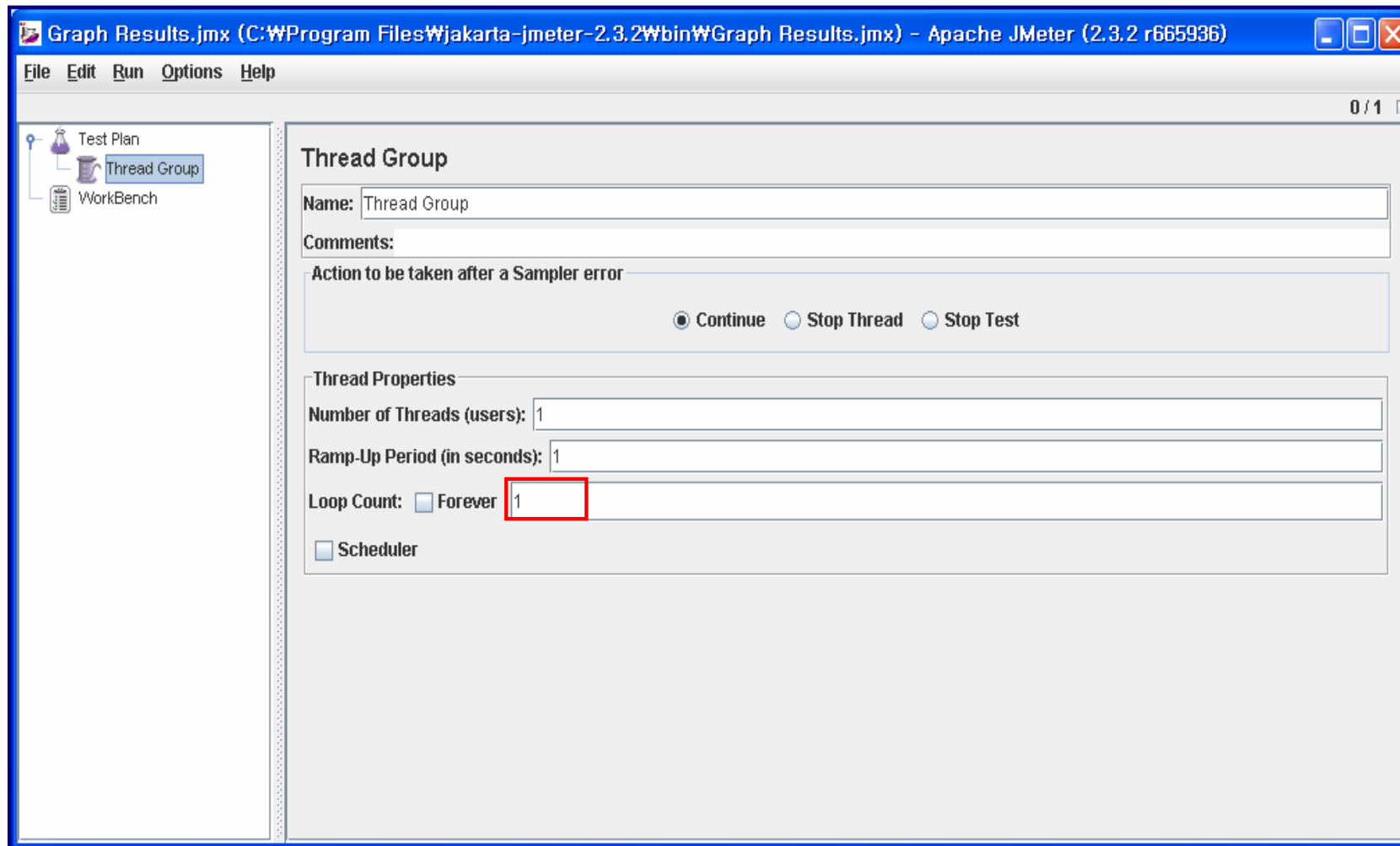
- Thread Group을 만듭니다.
 - 왼쪽 Test Plan선택 후 마우스 오른쪽 버튼 -> Add -> Thread Group 선택



3. 도구 기능 소개

3.7 Size Assertion 테스트 기능 (3/12)

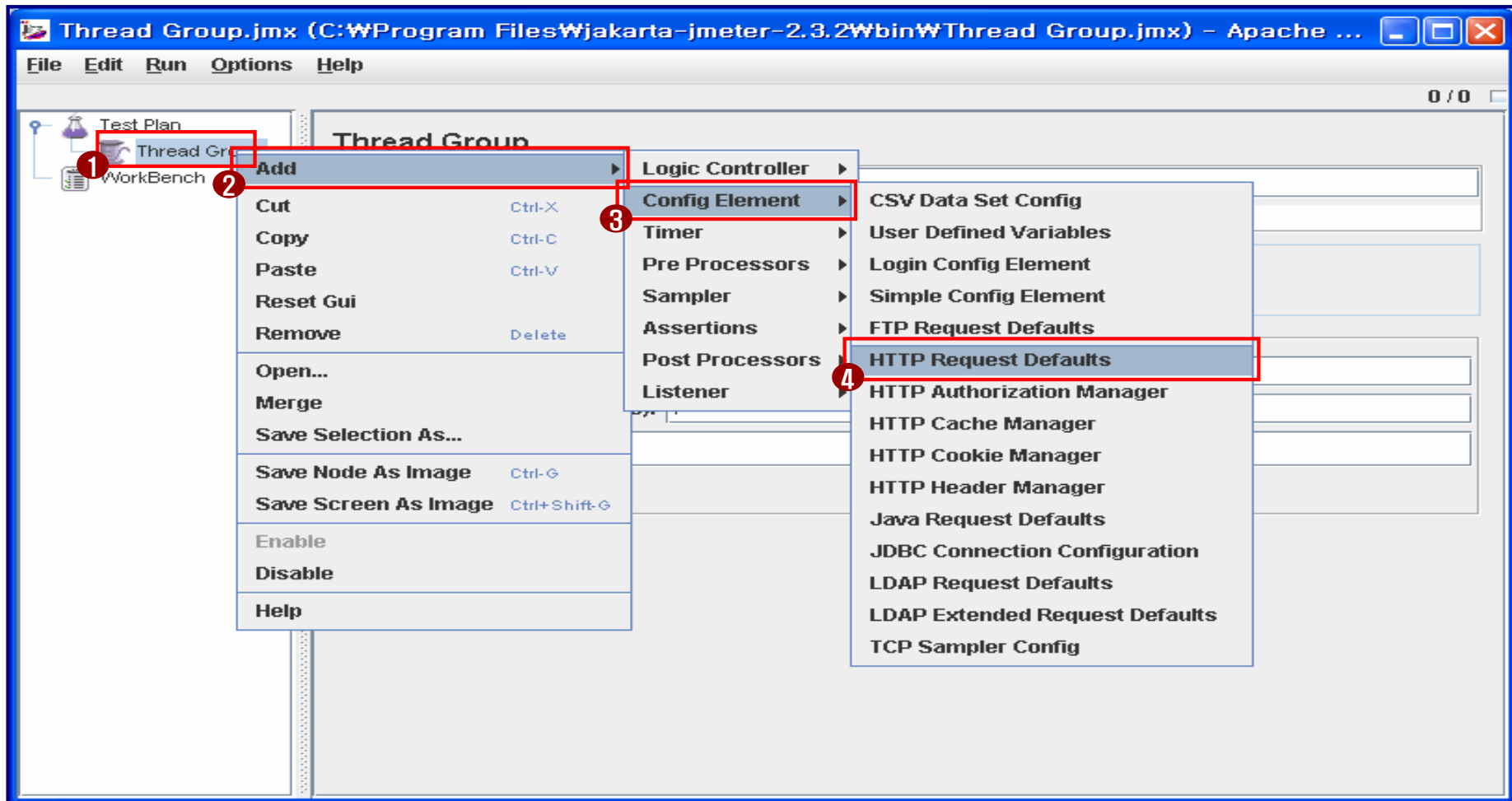
- 오른쪽에 나오는 항목들을 작성합니다.
 - 테스트를 한번만 수행한다는 의미로써 Loop Count 부분에서 1을 입력합니다.
 - 나머지 항목은 18페이지를 참고하여 작성 합니다.



3. 도구 기능 소개

3.7 Size Assertion 테스트 기능 (4/12)

- Http Request Defaults를 추가 합니다.
 - Thread Group 선택 후 마우스 오른쪽 버튼 -> Add -> Config Element -> Http Request Defaults



3. 도구 기능 소개

3.7 Size Assertion 테스트 기능 (5/12)

- 19페이지를 참고하여 아래의 항목들을 작성합니다.

HTTP Request Defaults

Name: HTTP Request Defaults

Comments:

Web Server

Server Name or IP: localhost Port Number:

HTTP Request

Protocol (default http): Content encoding:

Path: form.html

Send Parameters With the Request:

Name:	Value	Encode?	Include Equ...

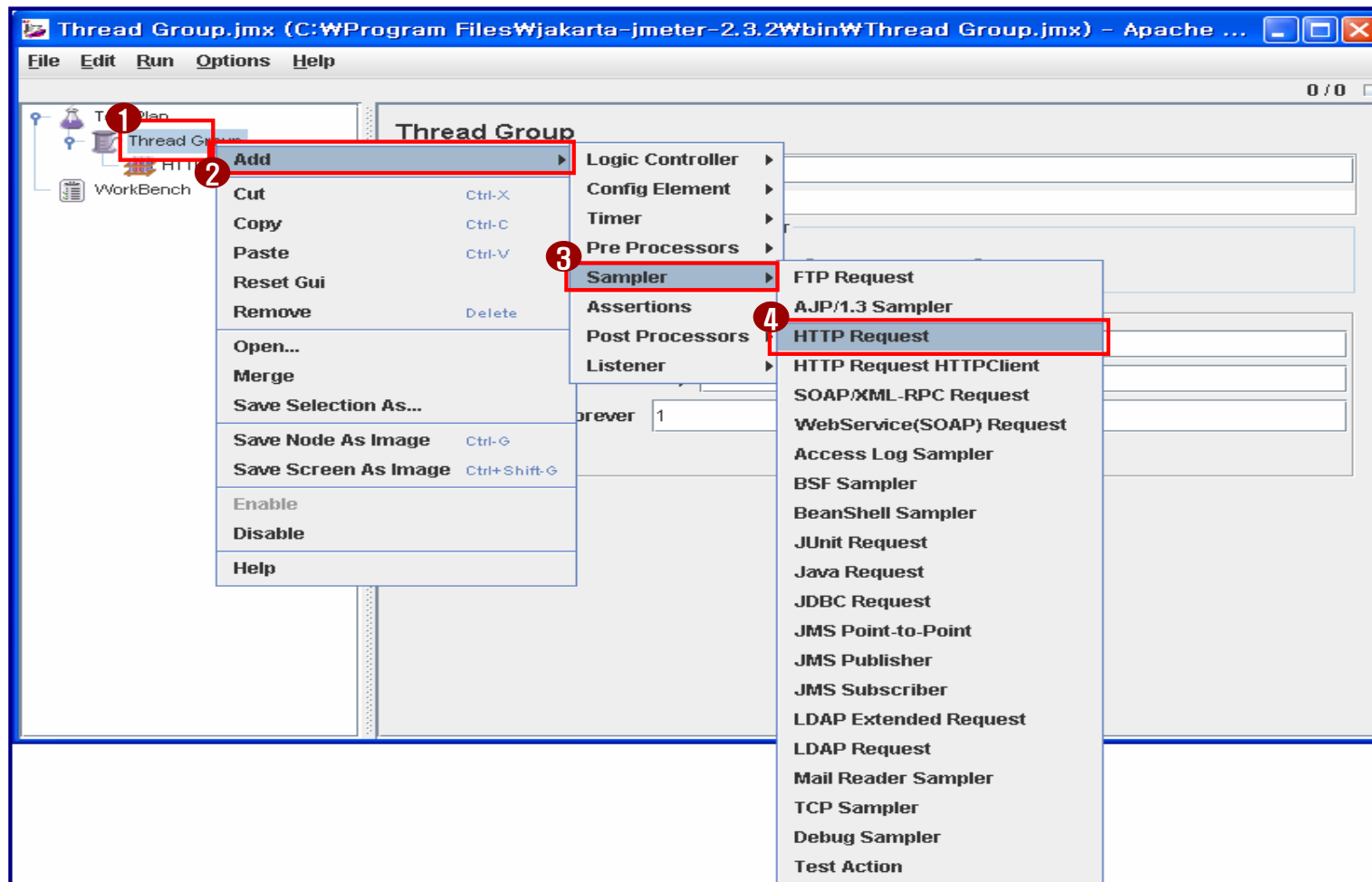
Add Delete

☐ Retrieve All Embedded Resources from HTML Files

3. 도구 기능 소개

3.7 Size Assertion 테스트 기능 (6/12)

- Http Request를 추가 합니다.
 - Thread Group 선택 후 마우스 오른쪽 버튼 -> Add -> Sampler -> HTTP Request 선택



3. 도구 기능 소개

3.7 Size Assertion 테스트 기능 (7/12)

- 22페이지를 참고하여 아래의 Http Request를 작성합니다.

HTTP Request

Name:

Comments:

Web Server

Server Name or IP: Port Number:

HTTP Request

Protocol (default http): Method: Content encoding:

Path:

☒ Redirect Automatically ☐ Follow Redirects ☒ Use KeepAlive ☐ Use multipart/form-data for HTTP POST

Send Parameters With the Request:

Name:	Value	Encode?	Include Equ...

Send Files With the Request:

File Path:	Parameter ...	MIME Type:

Optional Tasks

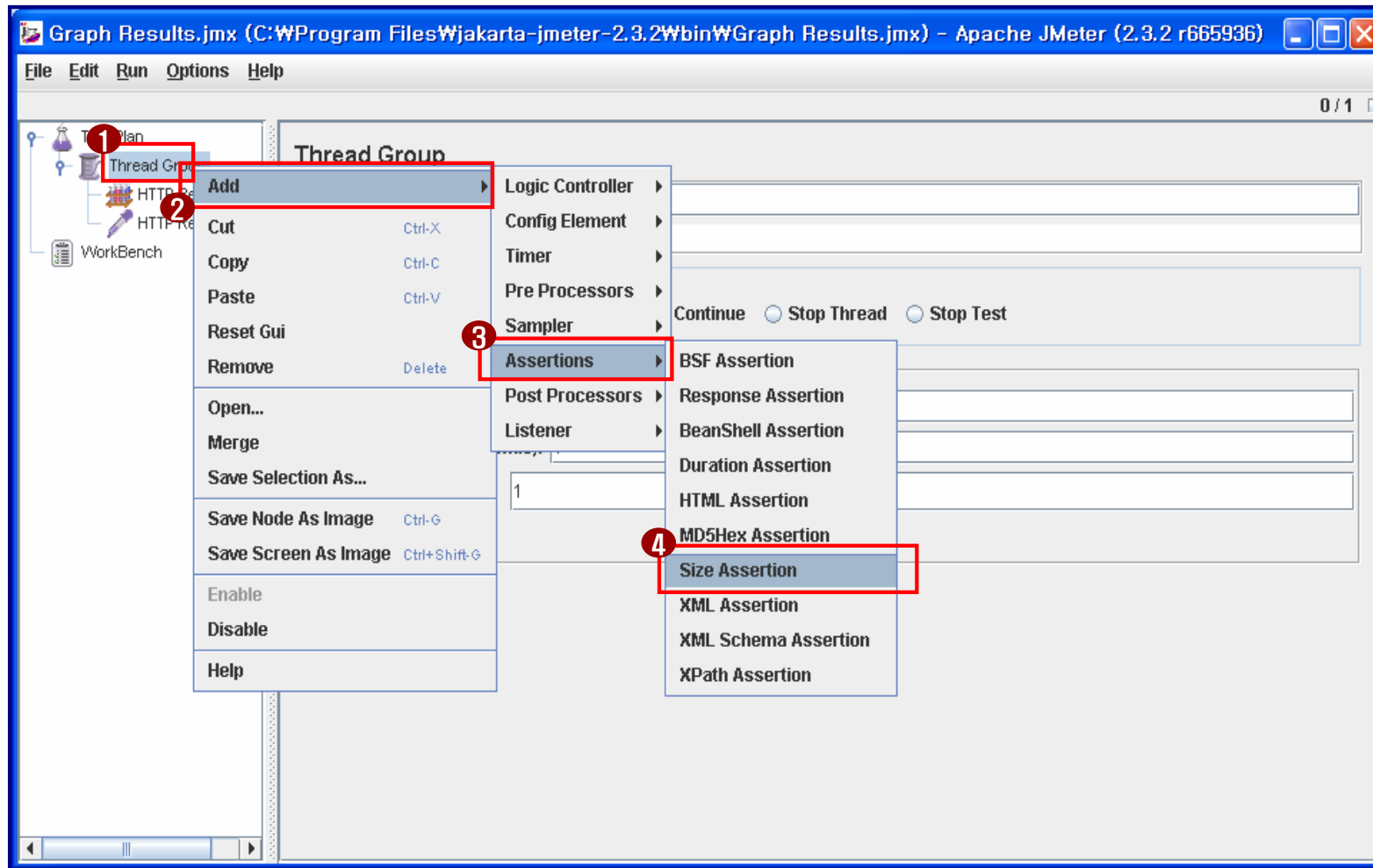
☐ Retrieve All Embedded Resources from HTML Files ☐ Use as Monitor ☐ Save response as MD5 hash?

Embedded URLs must match:

3. 도구 기능 소개

3.7 Size Assertion 테스트 기능 (8/12)

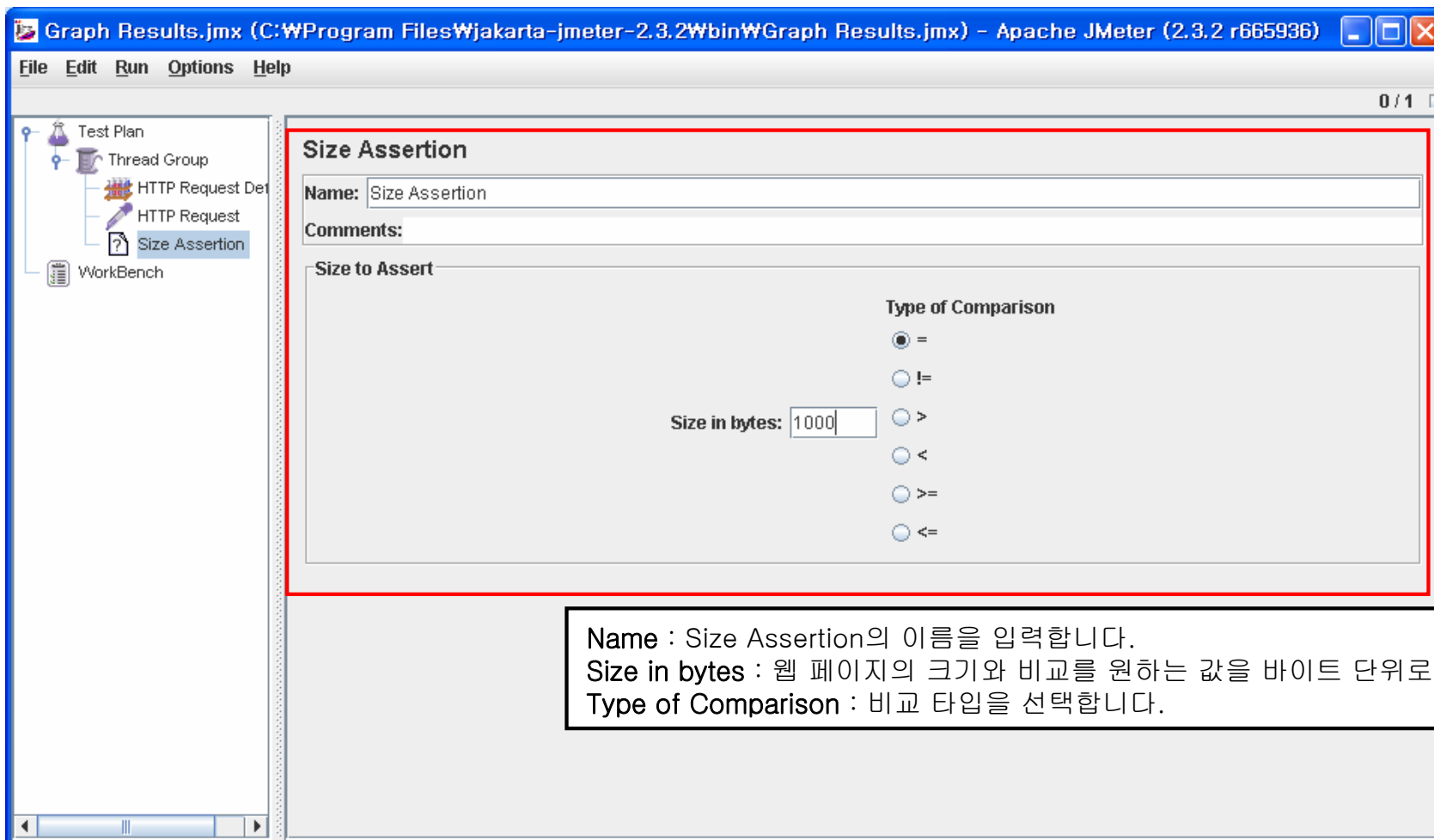
- Size Assertion을 추가합니다.
 - Thread Group 선택 후 마우스 오른쪽 버튼 -> Add -> Assertions -> Size Assertion 수행



3. 도구 기능 소개

3.7 Size Assertion 테스트 기능 (9/12)

- Size Assertion 항목들을 입력합니다.
- Size in bytes 부분에 원하는 바이트 단위로 값을 입력한 후 Type of comparison에서 비교타입을 선택합니다.
 - 본 매뉴얼에서는 Size in bytes : 1000 을 입력하고 Thpe of comparison은 “=”를 선택하였습니다.

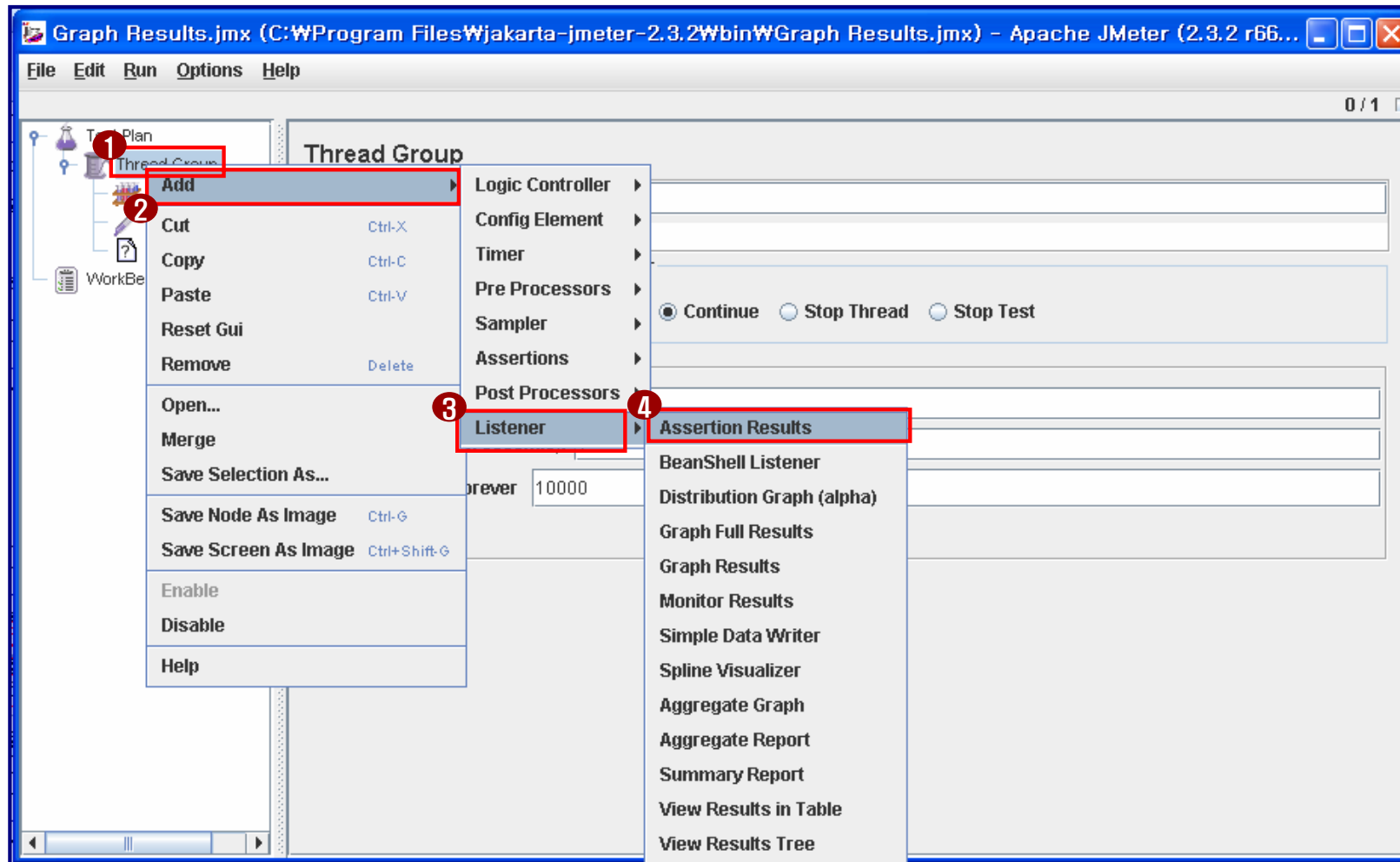


Name : Size Assertion의 이름을 입력합니다.
 Size in bytes : 웹 페이지의 크기와 비교를 원하는 값을 바이트 단위로 입력합니다.
 Type of Comparison : 비교 타입을 선택합니다.

3. 도구 기능 소개

3.7 Size Assertion 테스트 기능 (10/12)

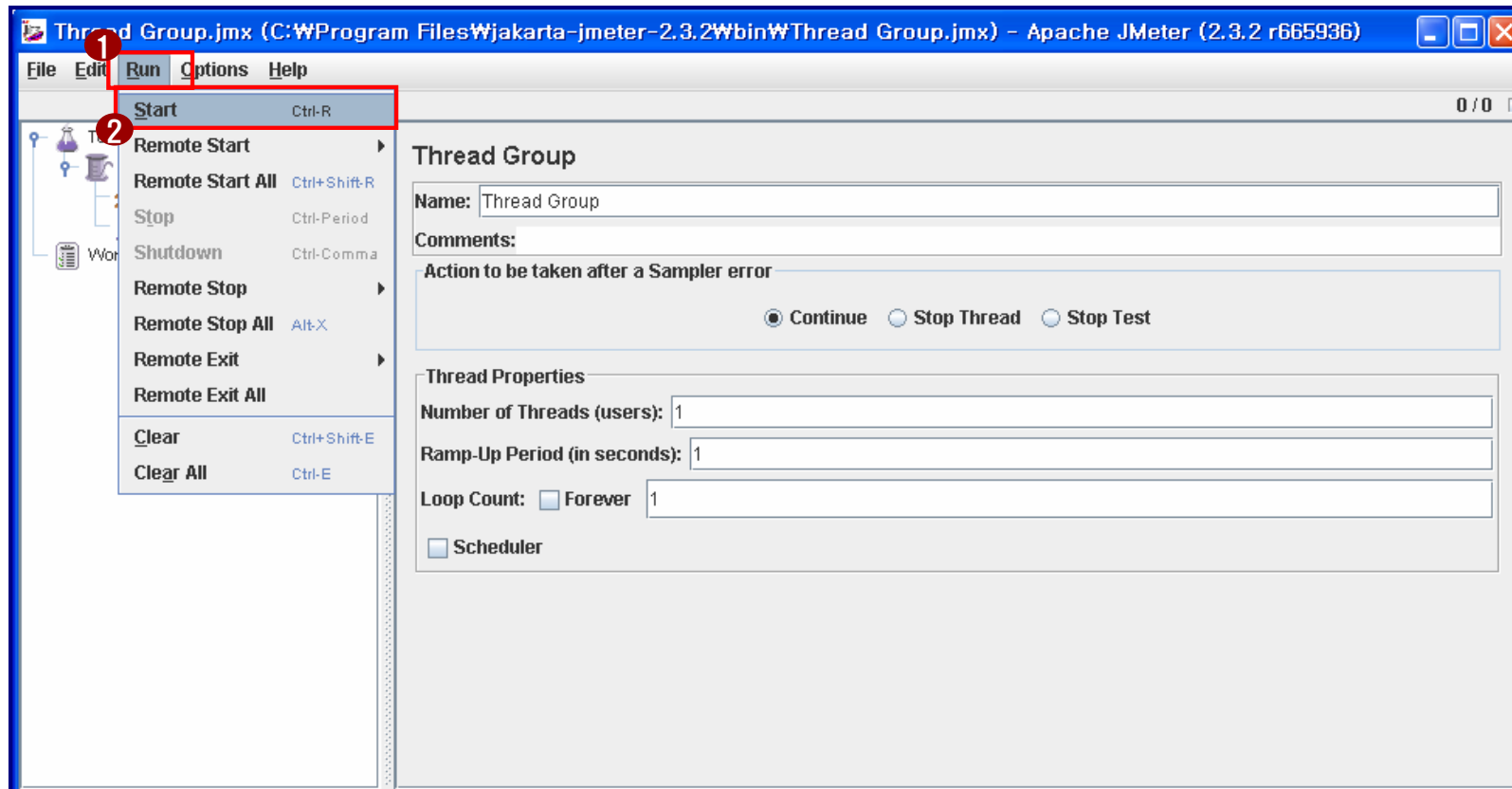
- Assertion 테스트 결과를 보기 위해 Listener를 추가합니다.
 - Thread Group 선택 후 마우스 오른쪽 버튼 -> Add -> Listener -> Assertion Results 선택



3. 도구 기능 소개

3.7 Size Assertion 테스트 기능 [11/12]

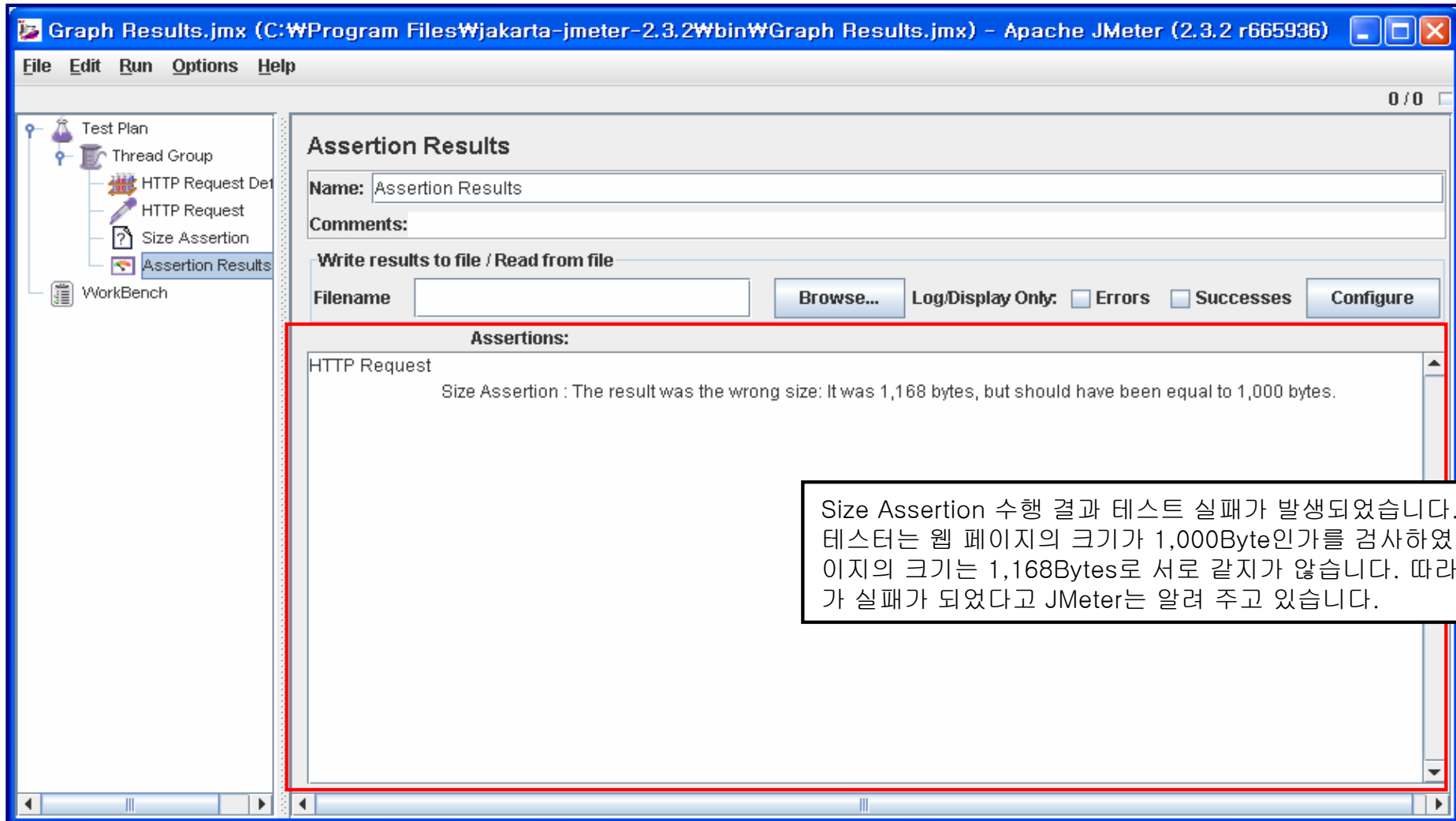
- 마지막으로 해야 할 일은 테스트를 실행 하는 것입니다.
 - 메뉴 바의 Run -> Start 클릭



3. 도구 기능 소개

3.7 Size Assertion 테스트 기능 [12/12]

- 테스트 실행 후 왼쪽 Assertion Results를 클릭하면 다음과 같은 화면이 출력 됩니다.



Size Assertion 수행 결과 테스트 실패가 발생되었습니다.
 테스트는 웹 페이지의 크기가 1,000Byte인가를 검사하였지만 웹 페이지의 크기는 1,168Bytes로 서로 같지가 않습니다. 따라서 테스트가 실패가 되었다고 JMeter는 알려 주고 있습니다.

4. 도구 활용 예제

4. 도구 활용 예제

세부 목차

- 4.1 예제 설명
- 4.2 테스트 플랜 작성하기
- 4.3 Thread Group 작성하기
- 4.4 Http Request Defaults 작성하기
- 4.5 Http Request 작성하기
- 4.6 Duration Assertion 작성하기
- 4.7 Size Assertion 작성하기
- 4.8 Listener 추가하기
- 4.9 테스트 실행하기
- 4.10 테스트 결과 보기

4. 도구 활용 예제

4.1 예제 설명 [1/2]

- 예제 개요



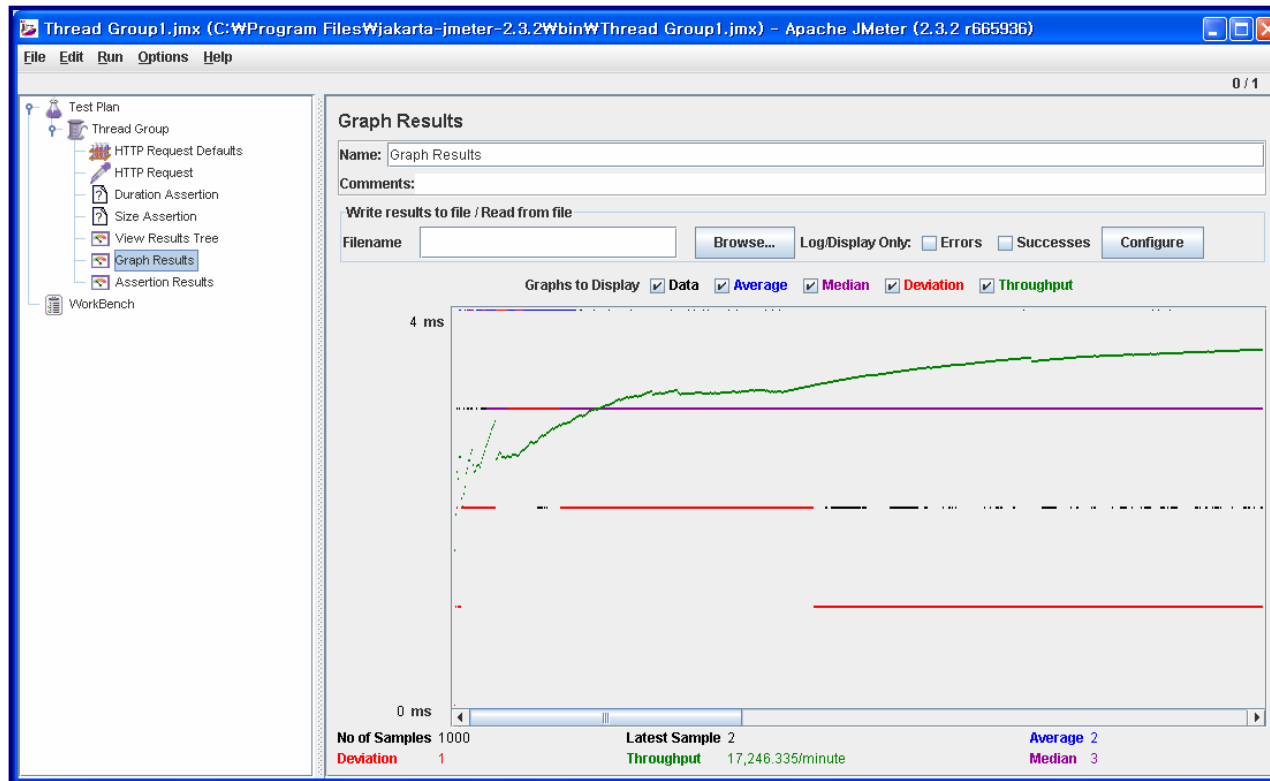
예제 설명

- 경기대학교 웹 페이지를 사용하는 학생들은 학교 측에 계속해서 웹 페이지를 개선할 것을 요구하였습니다.
- 학생들의 요구사항은 다음과 같습니다.
 - 웹 페이지 접속에 너무 많은 시간이 걸림
 - 몇몇 웹 페이지는 존재하지 않음
 - 수강신청 기간에는 웹 페이지가 열리지 않음
- 경기대학교 웹 관리자인 홍길동씨는 자신의 관리하는 웹에 대해서 몇 가지의 테스트를 수행하기로 결정하였습니다.
 - 웹 페이지에 대한 부하 테스트 기능
 - 응답 결과를 검증하기 위한 Assertion 기능 중 Duration Assertion 테스트 기능
 - 응답 결과를 검증하기 위한 Assertion 기능 중 Size Assertion 테스트 기능

4. 도구 활용 예제

4.1 예제 설명 [2/2]

- 관리자 홍길동씨의 테스트 계획은 다음과 같습니다.
 - 부하 테스트 수행 시 반복 횟수는 1000번으로 설정한다.
 - Duration Assertion 테스트 수행 시 지정한 시간은 10Millisecond으로 설정한다.
 - Size Assertion 테스트 수행 시 테스트할 웹 페이지 크기가 2,000Bytes보다 작을 것이라고 설정한다.
- 홍길동씨는 테스트 결과를 바탕으로 웹 페이지 수정 작업을 할 것입니다.



[부하 테스트 결과를 보여주는 Graph Results 화면]

3. 도구 기능 소개

4.2 테스트 플랜 작성하기

- 처음으로 홍길동씨가 해야 할 일은 테스트 플랜을 작성하는 것입니다.
 - 16페이지를 참고하여 작성합니다.

Test Plan

Name: Test Plan

Comments:

User Defined Variables

Name:	Value

Add Delete

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

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

Selecting Functional Test Mode may adversely affect performance.

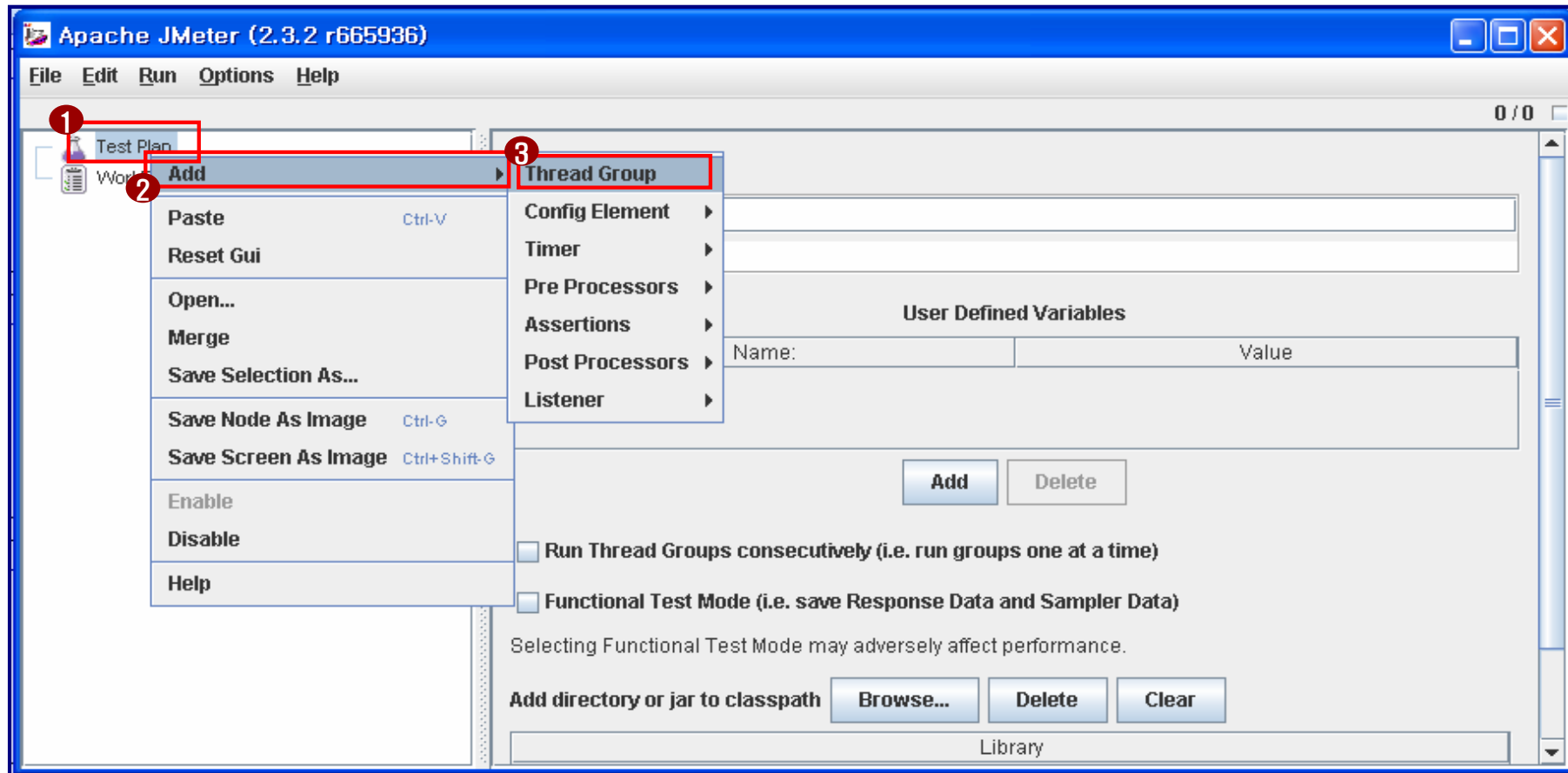
Add directory or jar to classpath Browse... Delete Clear

Library

3. 도구 기능 소개

4.3 Thread Group 작성하기 (1/2)

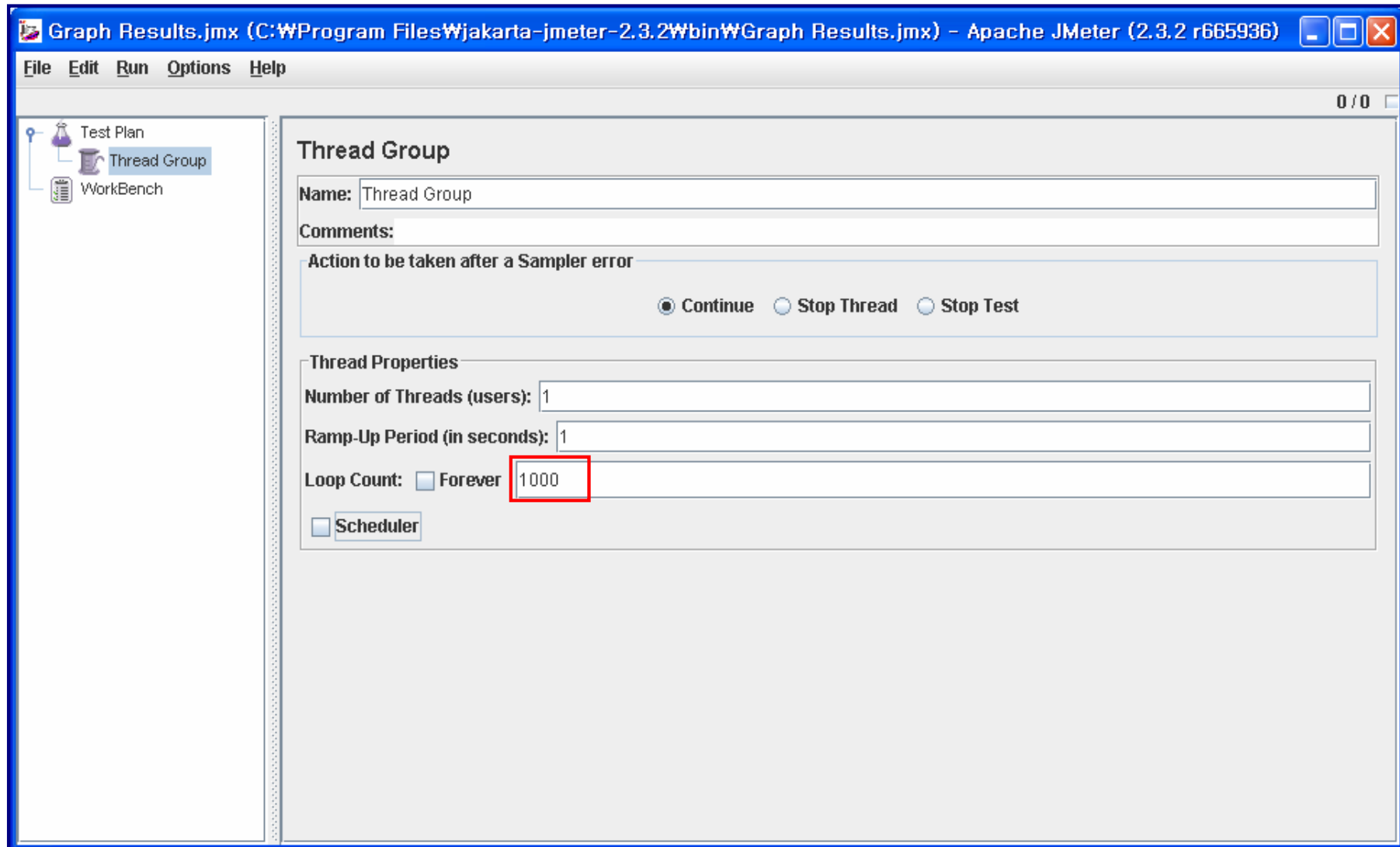
- Thread Group을 만듭니다.
 - Test Plan 선택 후 마우스 오른쪽 버튼 -> Add -> Thread Group 선택



3. 도구 기능 소개

4.3 Thread Group 작성하기 (2/2)

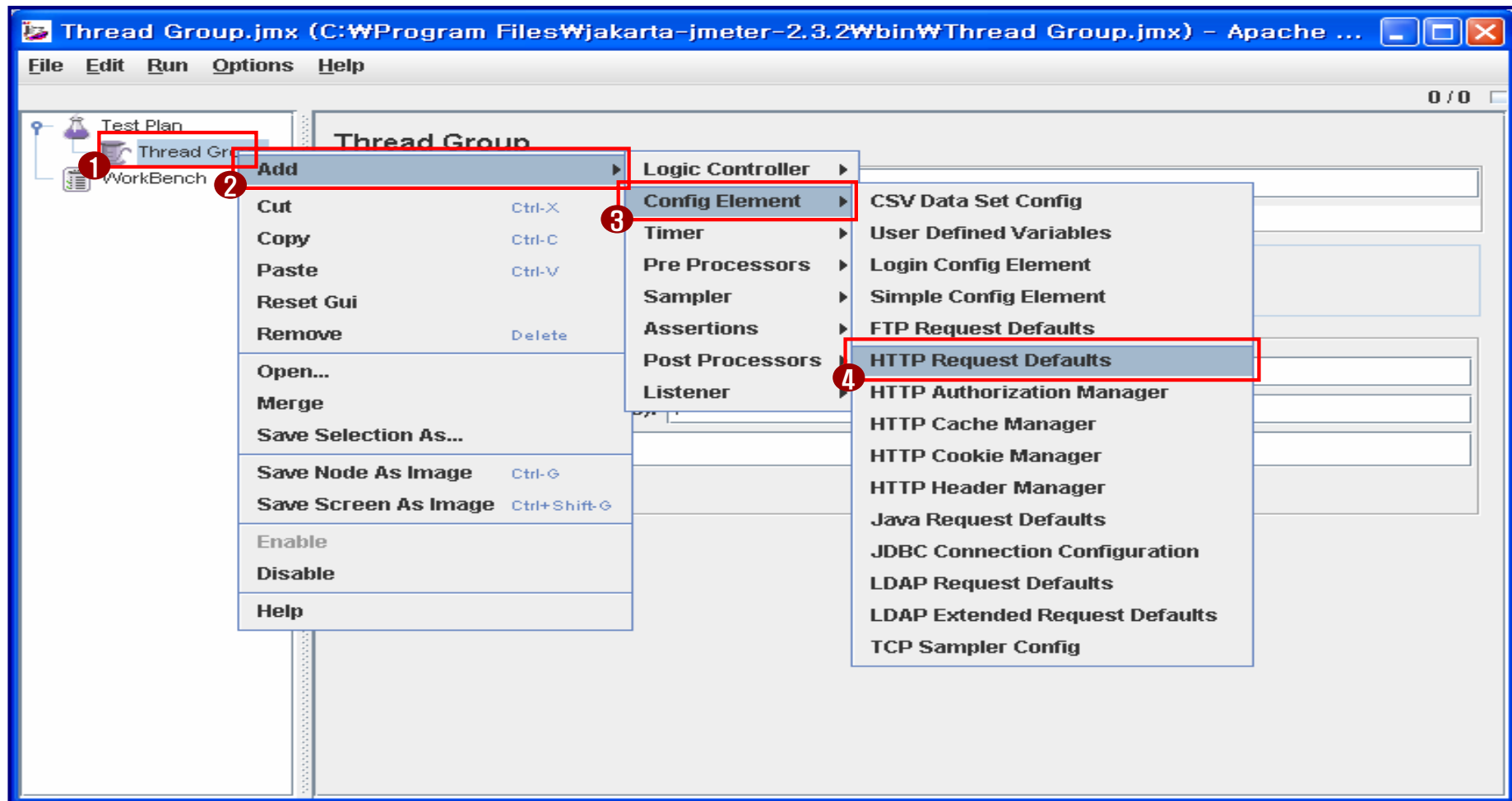
- 오른쪽 항목들에 대해 작성 합니다.
 - Loop Count는 1000으로 설정합니다.



3. 도구 기능 소개

4.4 Http Request Defaults 작성하기 (1/2)

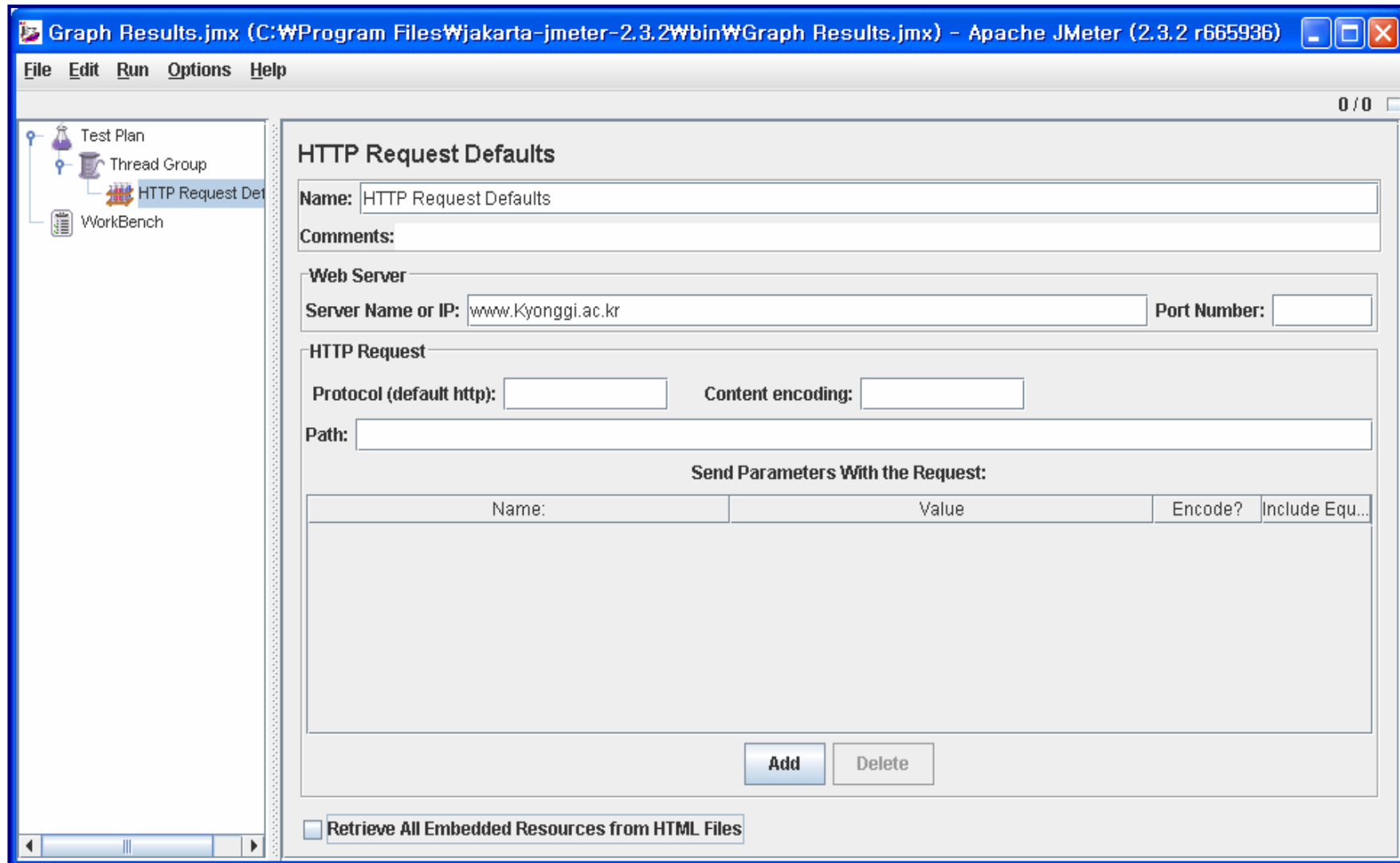
- Http Request Defaults를 추가 합니다.
 - Thread Group 선택 후 마우스 오른쪽 버튼 -> Add -> Config Element -> Http Request Defaults



3. 도구 기능 소개

4.4 Http Request Defaults 작성하기 (2/2)

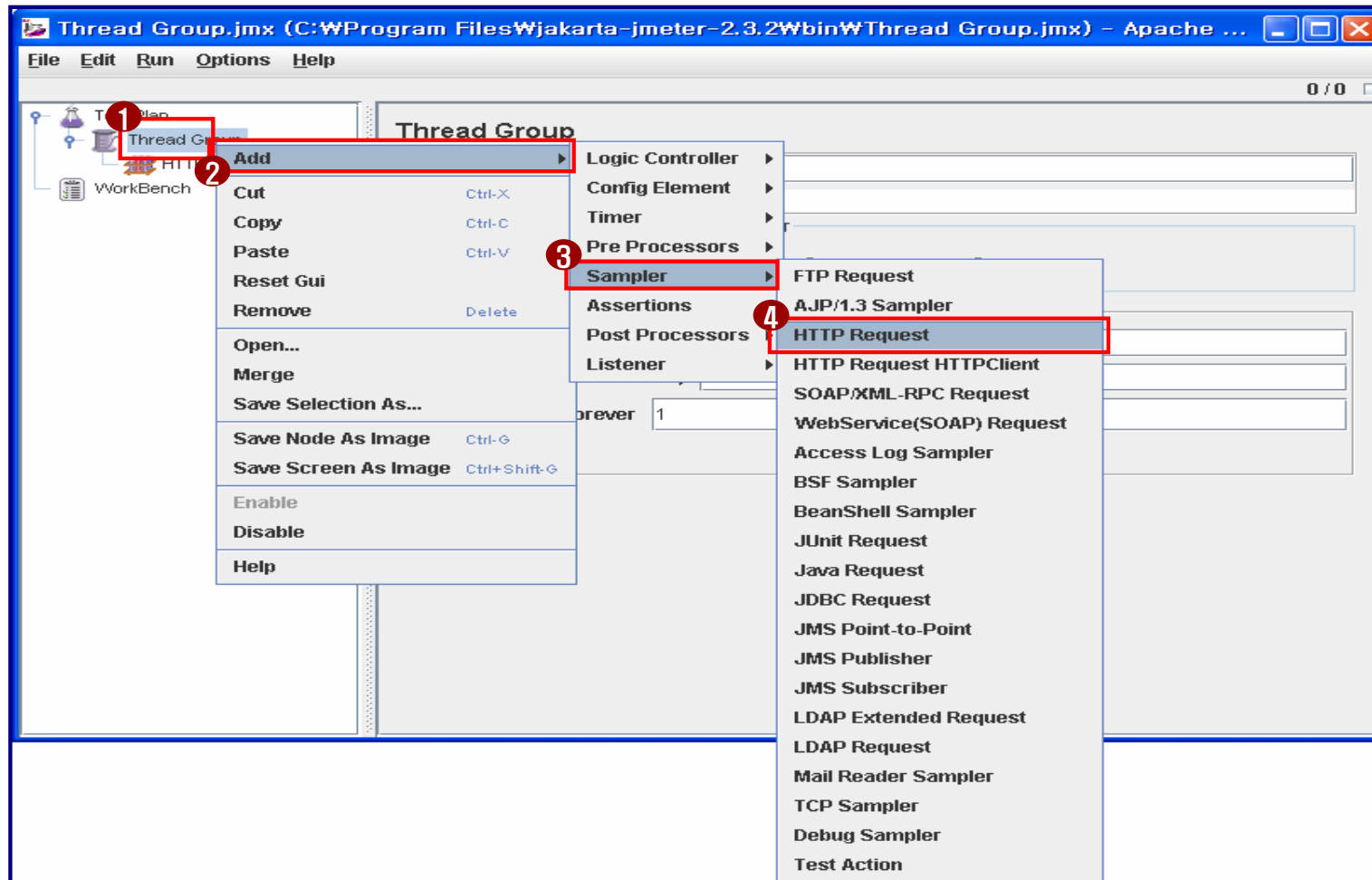
- Http Request Defaults를 작성합니다.
 - 19페이지를 참조하여 다음의 항목들을 작성합니다.



3. 도구 기능 소개

4.5 Http Request 작성하기 (1/2)

- Http Request를 추가 합니다.
 - Thread Group 선택 후 마우스 오른쪽 버튼 -> Add -> Sampler -> HTTP Request 선택



3. 도구 기능 소개

4.5 Http Request 작성하기 (2/2)

- Http Request의 입력 정보는 다음과 같습니다.
 - 22페이지를 참고하여 아래의 항목들을 작성합니다.

HTTP Request

Name:

Comments:

Web Server

Server Name or IP: Port Number:

HTTP Request

Protocol (default http): Method: Content encoding:

Path:

☒ Redirect Automatically ☐ Follow Redirects ☒ Use KeepAlive ☐ Use multipart/form-data for HTTP POST

Send Parameters With the Request:

Name:	Value	Encode?	Include Equ...

Send Files With the Request:

File Path:	Parameter ...	MIME Type:

Optional Tasks

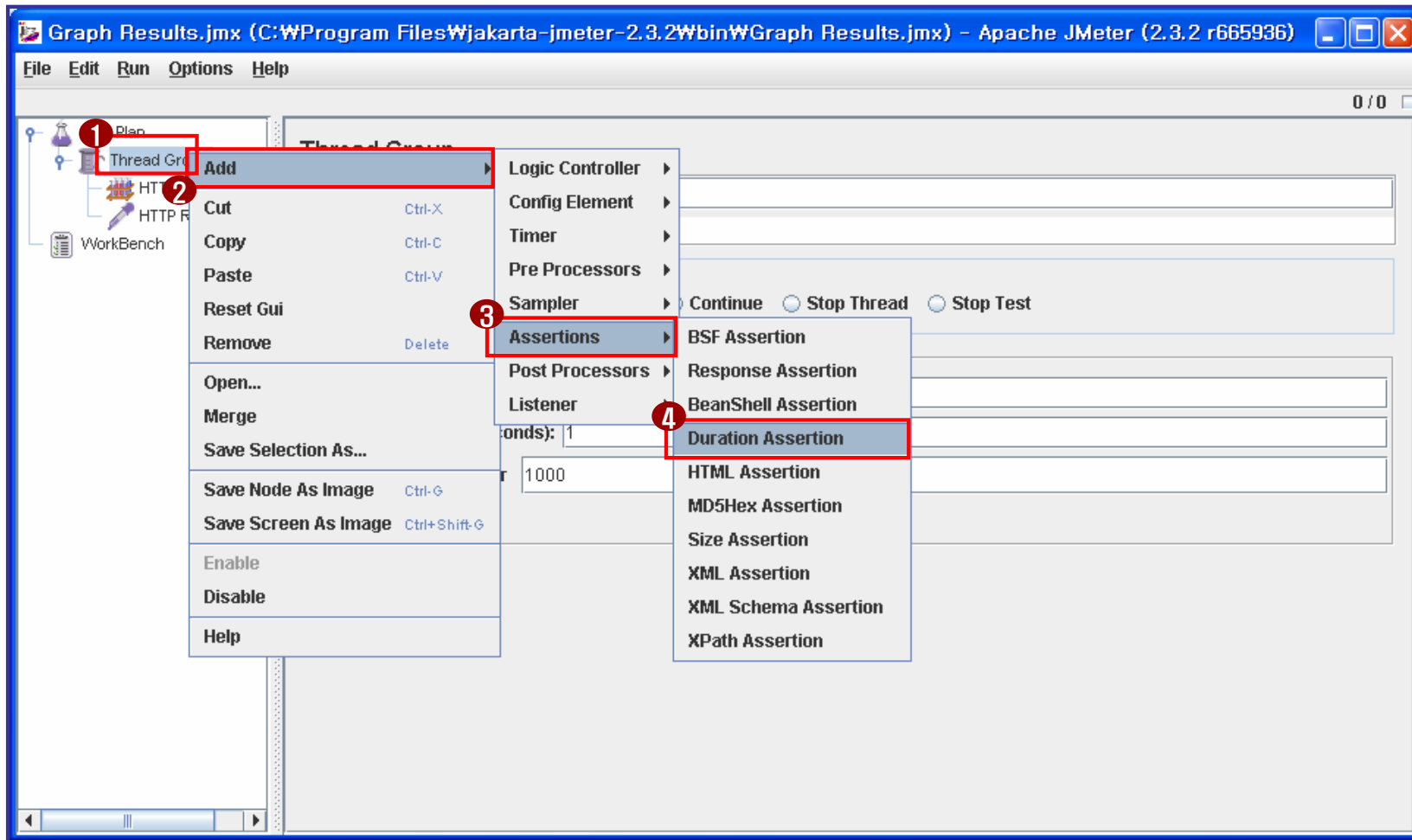
☐ Retrieve All Embedded Resources from HTML Files ☐ Use as Monitor ☐ Save response as MD5 hash?

Embedded URLs must match:

3. 도구 기능 소개

4.6 Duration Assertion 작성하기 (1/2)

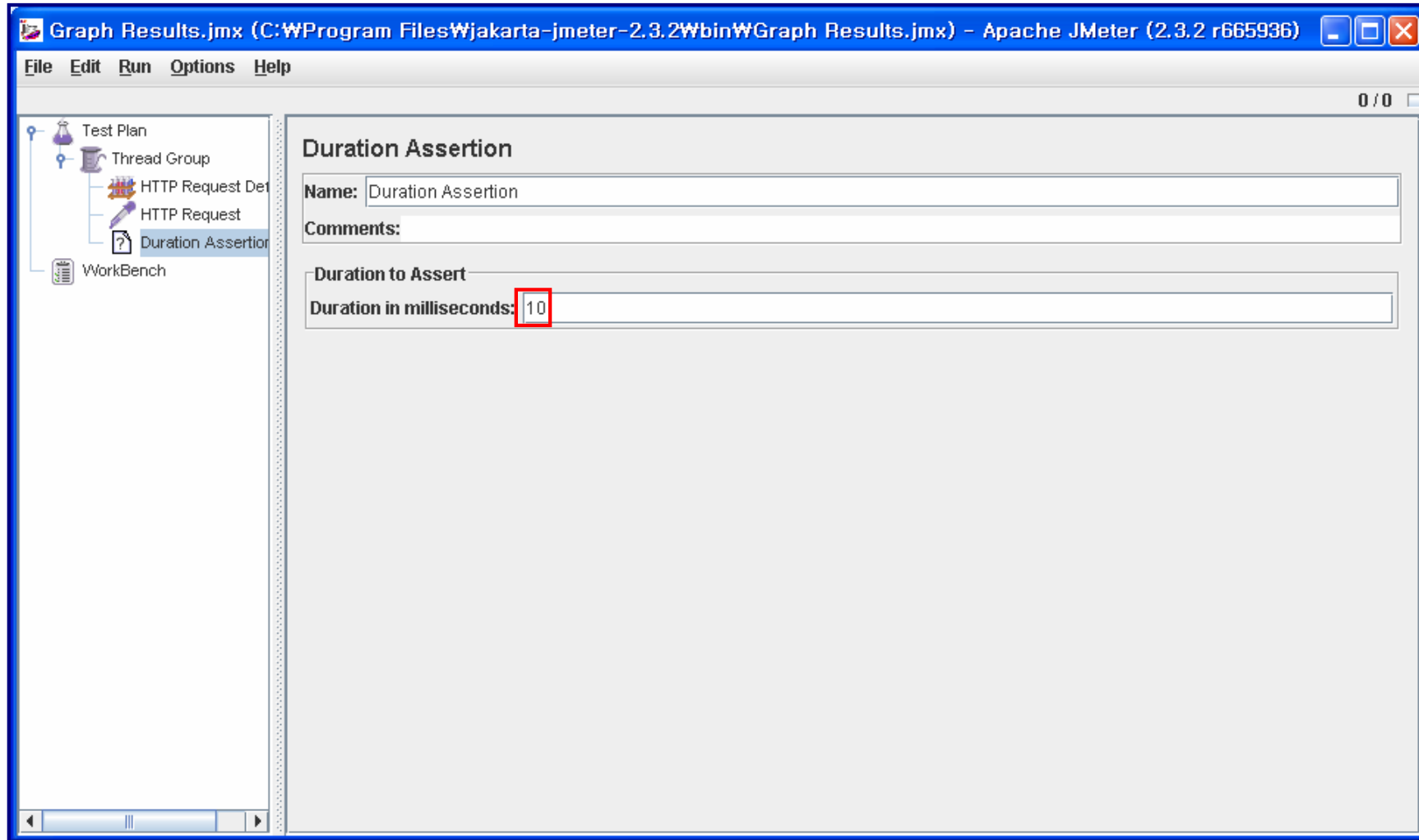
- Duration Assertion 테스트 수행을 위한 Duration Assertion을 추가해야 합니다.
 - Thread Group선택 후 마우스 오른쪽 버튼 -> Add -> Assertions -> Duration Assertion 선택



3. 도구 기능 소개

4.6 Duration Assertion 작성하기 (2/2)

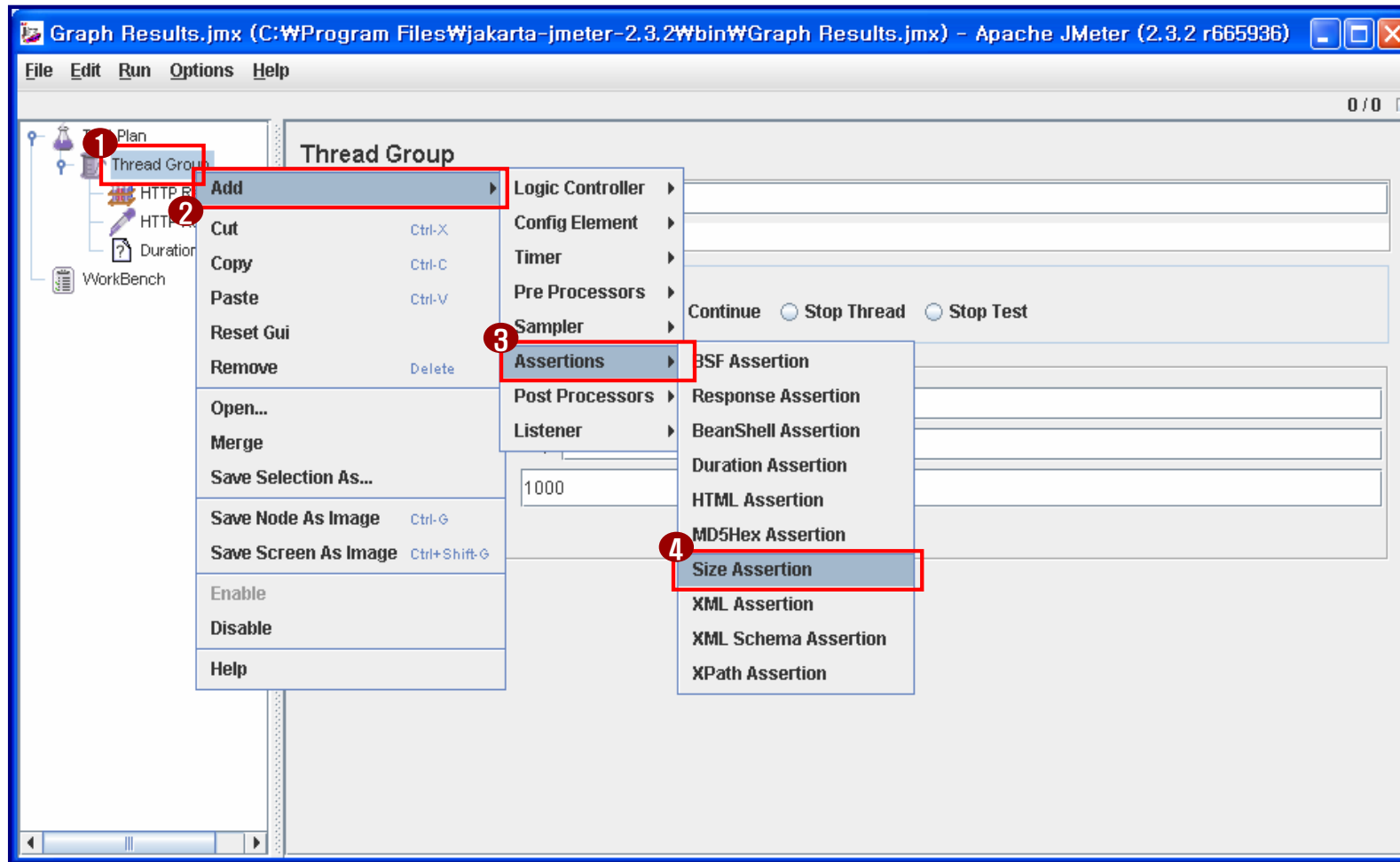
- Duration Assertion 테스트를 위해 원하는 응답 시간을 10Milliseconds로 입력합니다.



3. 도구 기능 소개

4.7 Size Assertion 작성하기 (1/2)

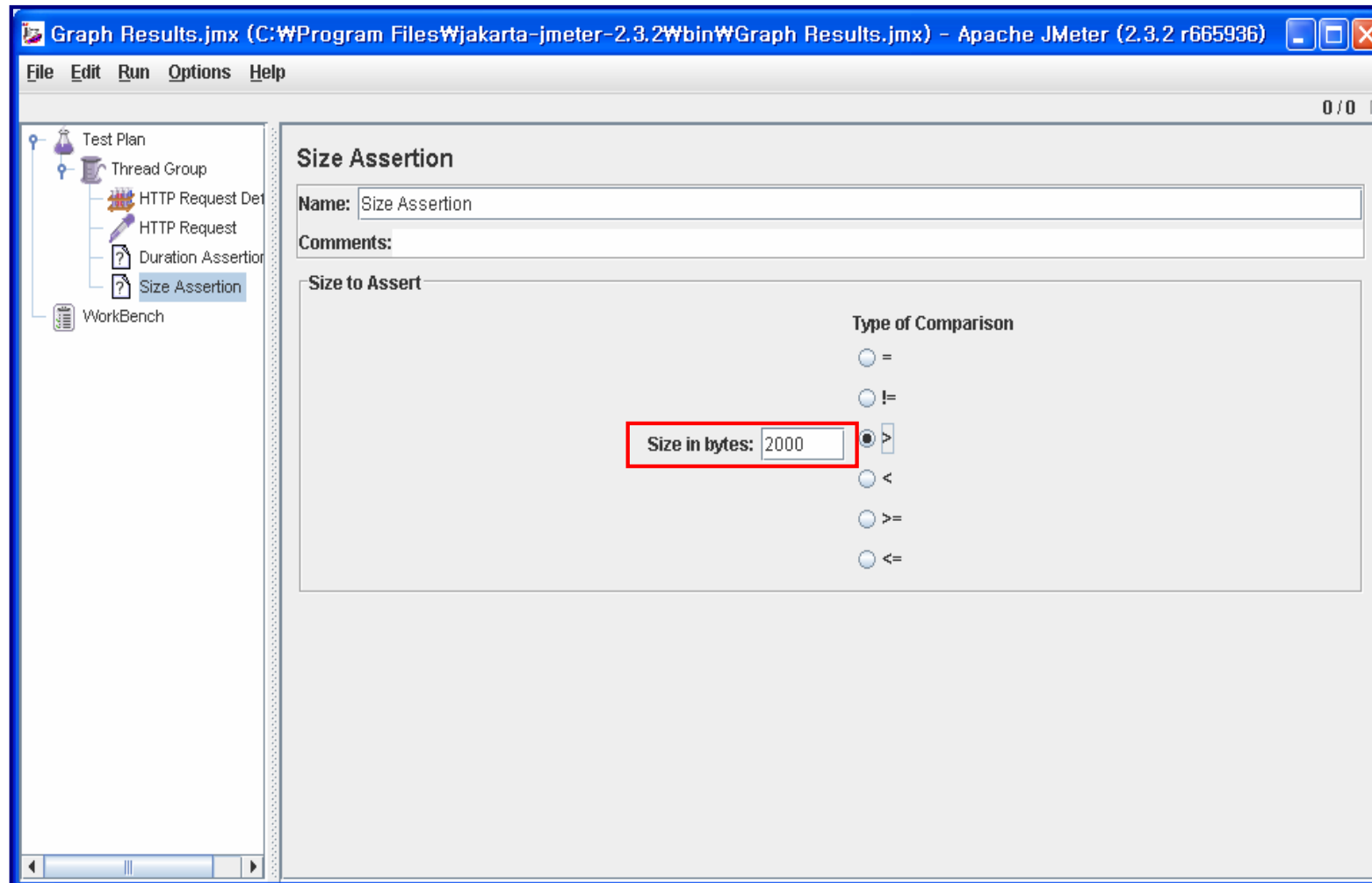
- Size Assertion 테스트 수행을 위한 Size Assertion을 추가합니다.
 - Thread Group 선택 후 마우스 오른쪽 버튼 -> Add -> Assertions -> Size Assertion 선택



3. 도구 기능 소개

4.7 Size Assertion 작성하기 (2/2)

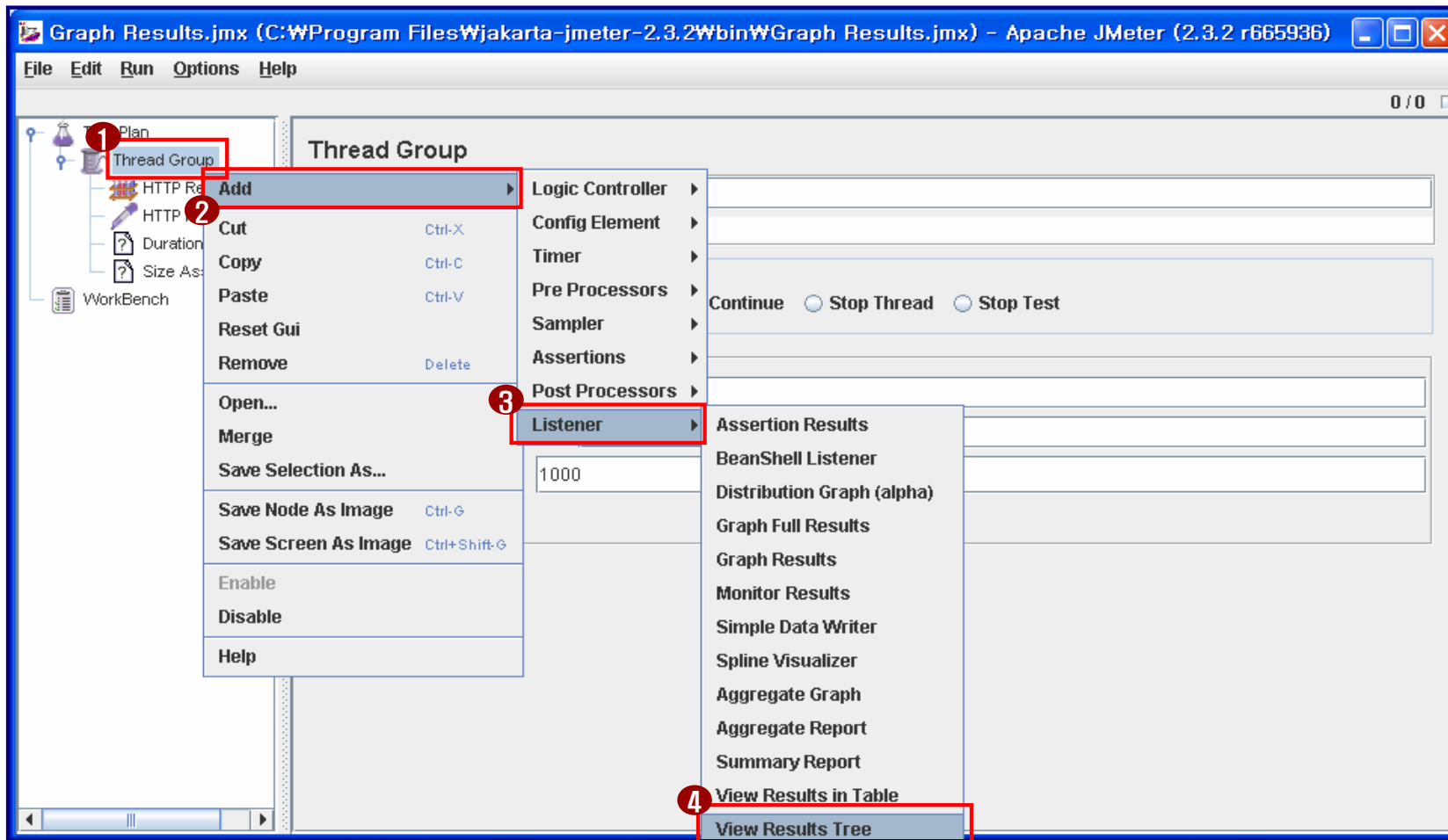
- Size Assertion 테스트를 위해 크기는 2,000Byte로 설정하였으며 비교 타입은 >으로 설정하였습니다.



3. 도구 기능 소개

4.8 Listener 추가하기 (1/3)

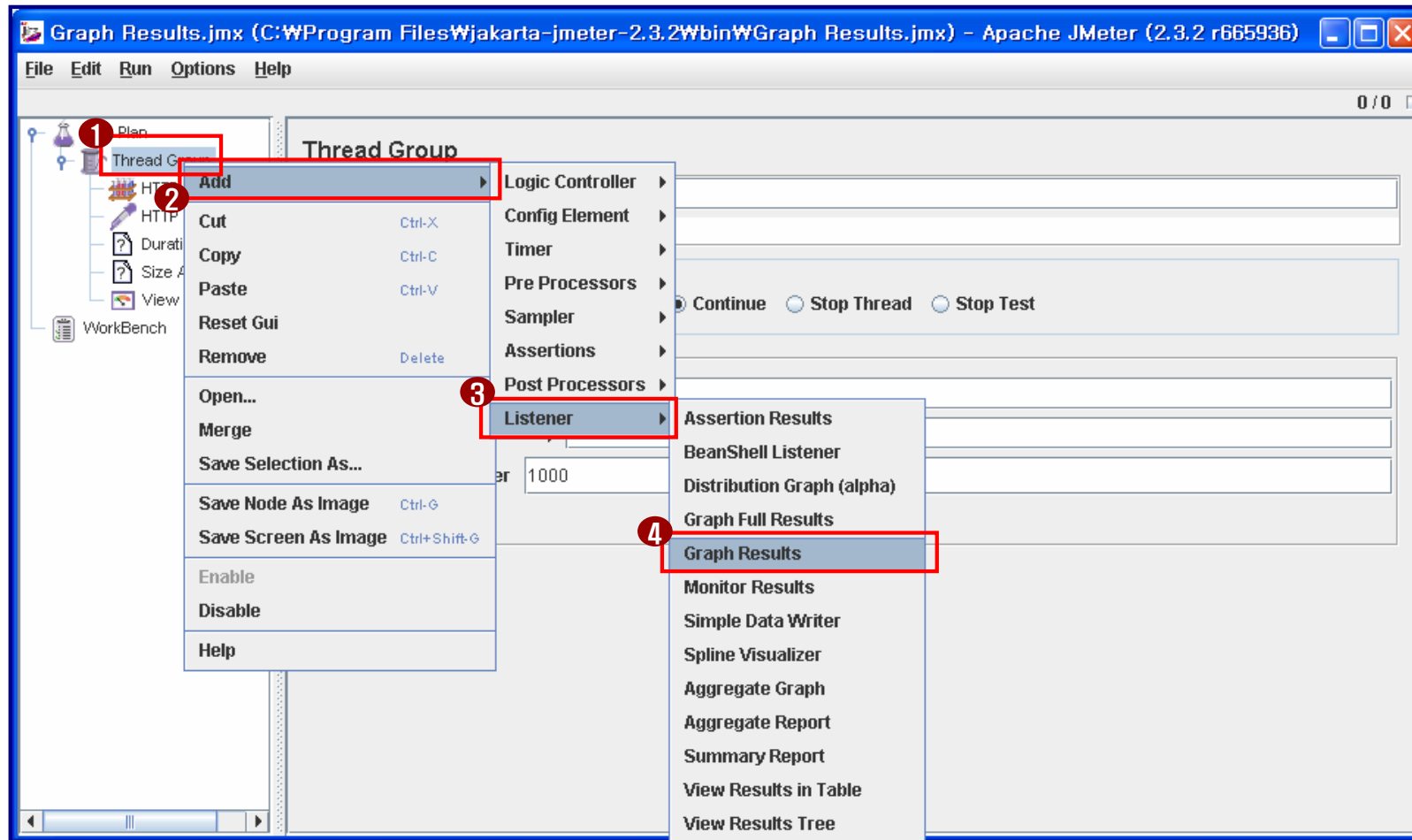
- 결과를 보기 위해 3개의 Listener를 추가합니다.
 - View Results Tree Listener를 추가합니다
 - Thread Group 선택 후 마우스 오른쪽 버튼 -> Add -> Listener -> View Results Tree 선택



3. 도구 기능 소개

4.8 Listener 추가하기 [2/3]

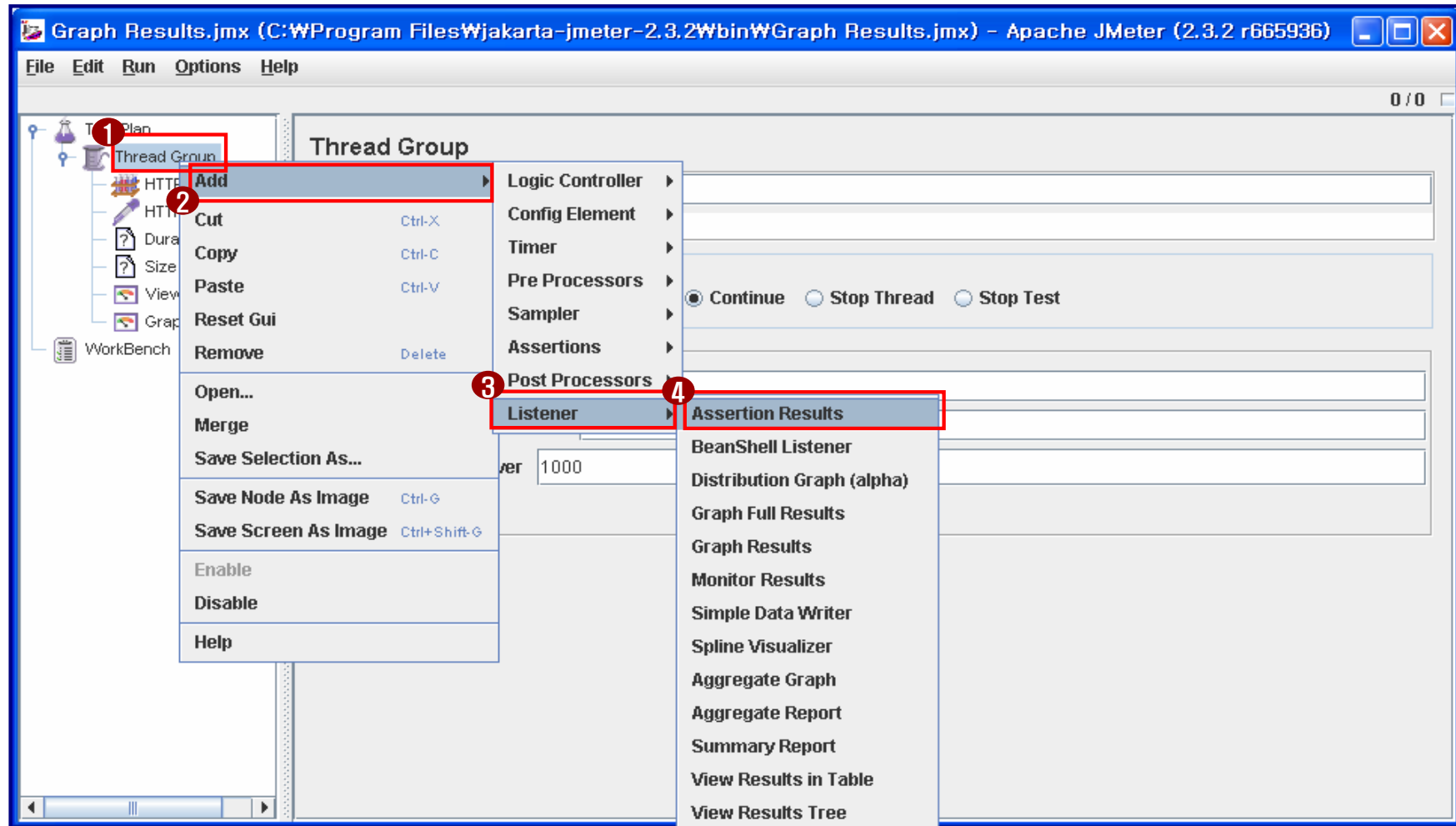
- 아래 화면은 Graph Results Listener를 추가하는 장면입니다.
- Thread Group 선택 후 마우스 오른쪽 버튼 -> Add -> Listener -> Graph Results Tree 선택



3. 도구 기능 소개

4.8 Listener 추가하기 (3/3)

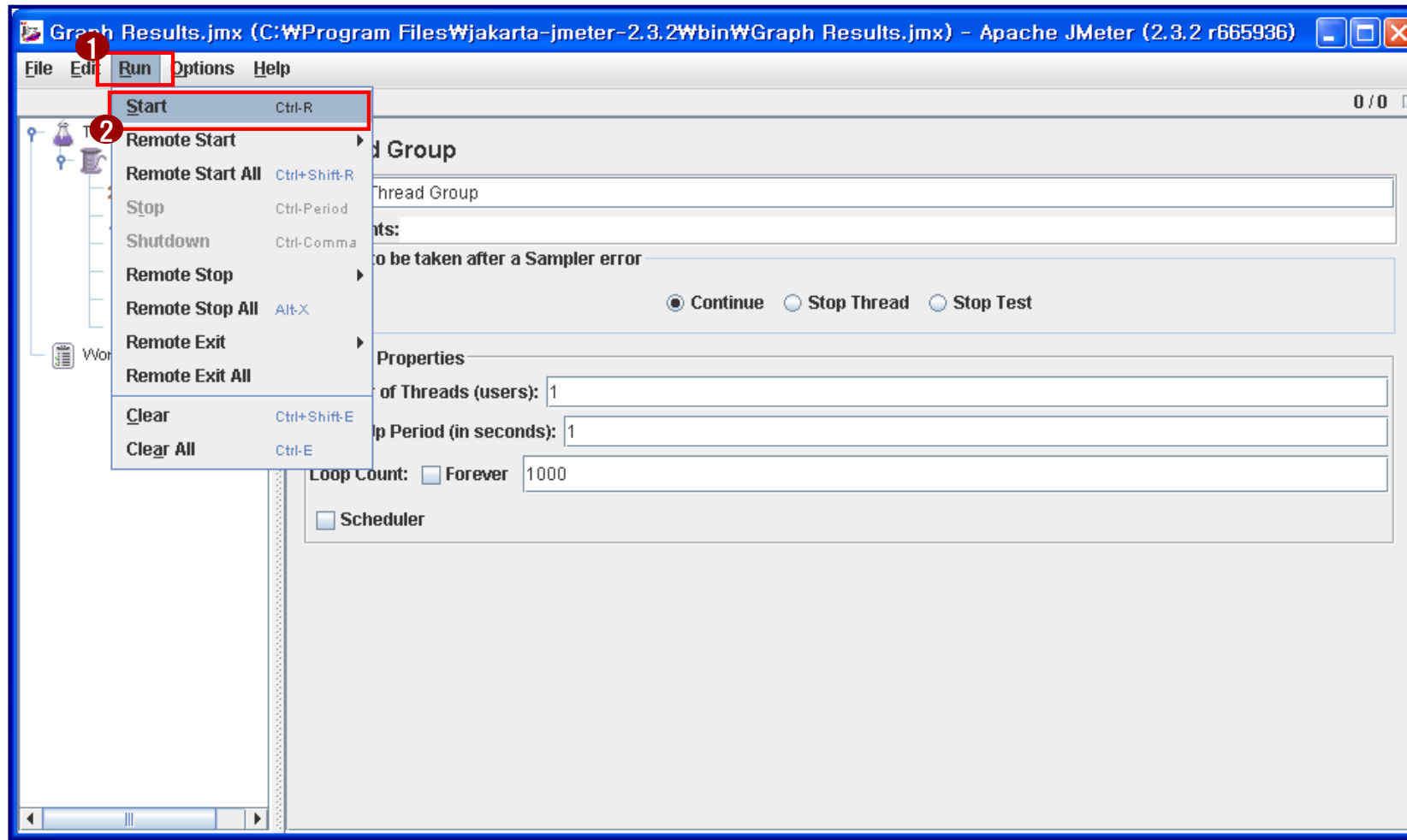
- 아래 화면은 Assertion Results Listener를 추가하는 장면입니다.
- Thread Group선택 후 마우스 오른쪽 버튼 -> Add -> Listener -> Assertion Results Tree 선택



3. 도구 기능 소개

4.9 테스트 실행하기

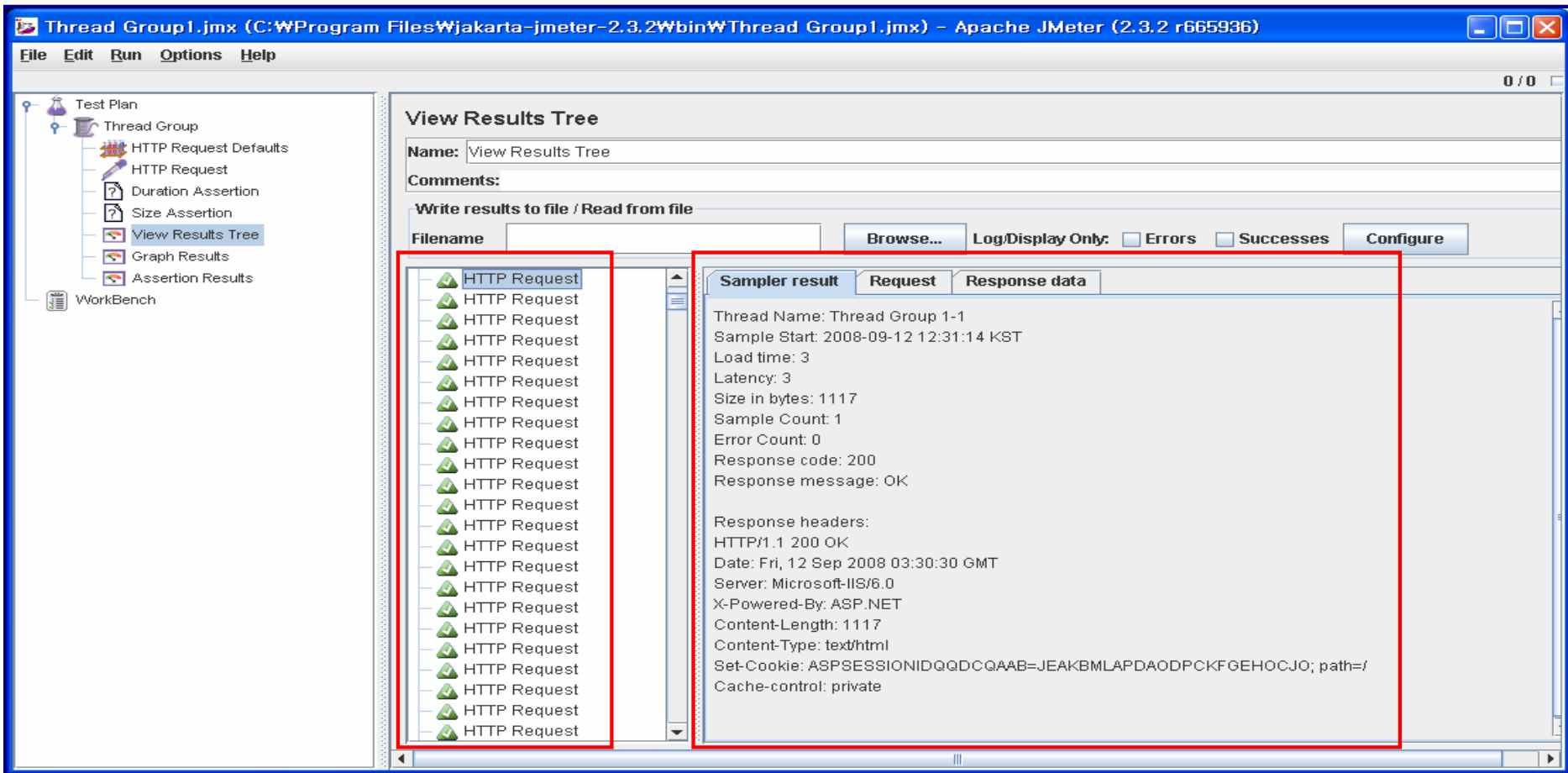
- 마지막으로 테스트를 실행 합니다.
 - 메뉴 바의 Run -> Start 클릭



3. 도구 기능 소개

4.10 테스트 결과 보기 (1/4)

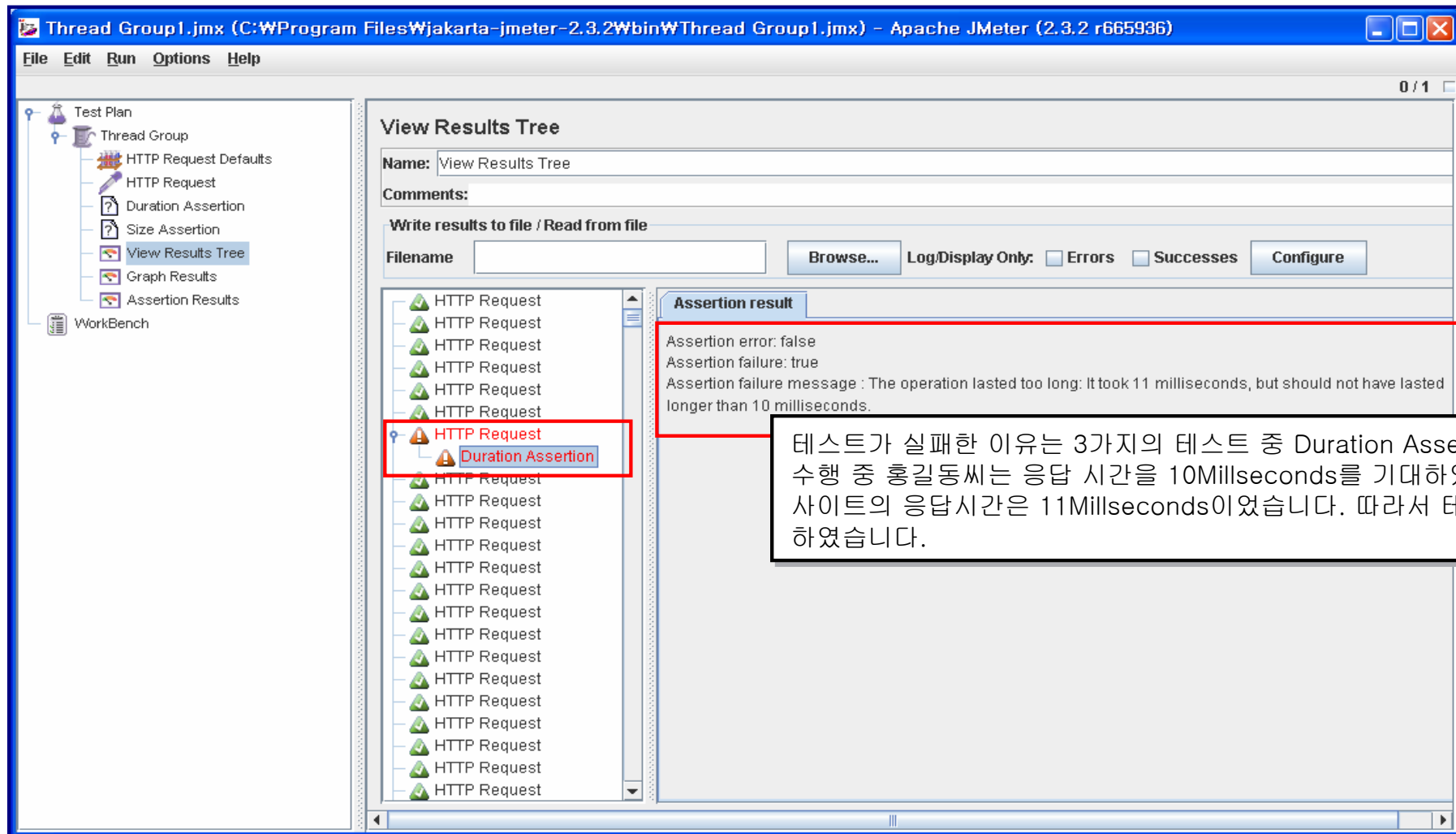
- View Results Tree를 클릭하면 다음과 같은 화면이 출력됩니다.
 - 테스트를 수행한 결과, 3가지의 테스트가 모두 성공했다는 것을 아래의 화면을 통해 알 수 있습니다.



3. 도구 기능 소개

4.10 테스트 결과 보기 (2/4)

- 만약 실패한 테스트가 존재한다면 아래의 화면과 같이 빨간색 글씨로 표시가 되며 그 부분을 클릭하면 테스트에 대한 상세 정보를 볼 수 있습니다.



테스트가 실패한 이유는 3가지의 테스트 중 Duration Assertion 테스트 수행 중 홍길동씨는 응답 시간을 10Milliseconds를 기대하였지만 실제 웹 사이트의 응답시간은 11Milliseconds이었습니다. 따라서 테스트는 실패하였습니다.

3. 도구 기능 소개

4.10 테스트 결과 보기 (3/4)

- Graph Results를 클릭하면 부하 테스트에 대한 결과를 볼 수 있습니다.

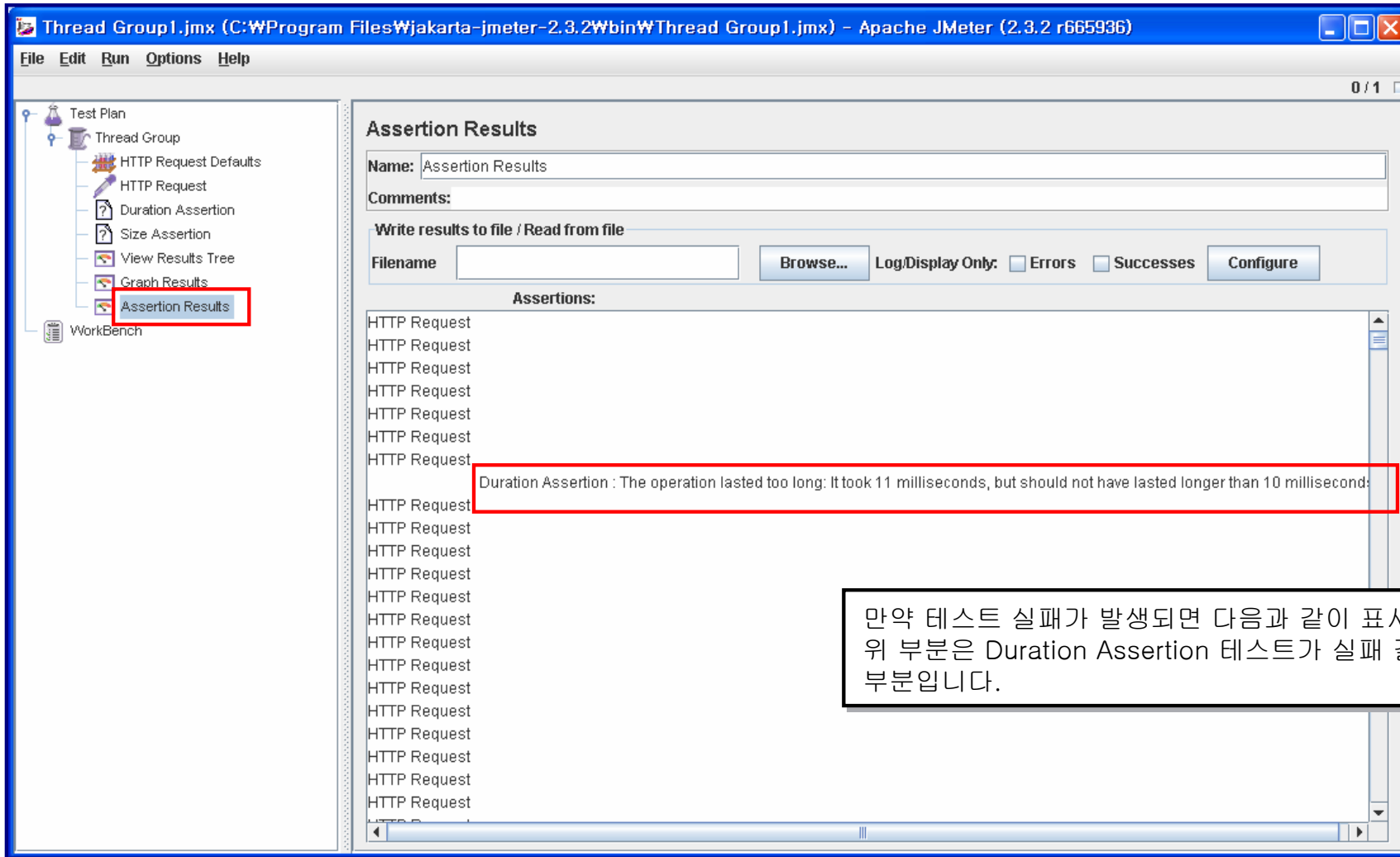


Deviation은 편차를 나타내는 것으로 편차값이 작을 수록 일정한 응답시간을 보여주는 것을 의미합니다. 따라서 Deviation값이 갑자기 크게 증가된다면 응답시간이 느리다는 것을 의미하며 시스템을 점검해 볼 필요가 있습니다.

3. 도구 기능 소개

4.10 테스트 결과 보기 (4/4)

- 마지막으로 Assertion Results부분을 클릭하면 다음과 같은 결과를 볼 수 있습니다.



만약 테스트 실패가 발생되면 다음과 같이 표시가 됩니다.
위 부분은 Duration Assertion 테스트가 실패 결과를 보여주는 부분입니다.

5. FAQ

5. FAQ

질문1) ASF (Apache Software Foundation) 란 무슨 단체인가요?



답1) ASF는 Apache Software Foundation의 약자로서 Apache HTTP Server를 포함한 Apache 프로젝트를 지원하는 비영리 단체이며 전 세계에 있는 소프트웨어 개발자들의 커뮤니티입니다. ASF에서 만들어지는 소프트웨어는 Apache License이며 그러므로 ASF에서 만들어지는 소프트웨어는 Free Software 또는 Open Source Software입니다. 현재 많은 프로젝트가 진행되고 있으며 프로그래머에 있어서 중요한 단체 중에 하나입니다.

질문2) Apache License란 무엇인가요?



답2) Apache License는 Apache Software Foundation(ASF)에 의해서 만들어지는 소프트웨어에 붙는 License입니다. ASF에서 만들어지는 소프트웨어는 모두 이 License가 적용됩니다. Apache License는 Free Software License이며 이는 비용을 지불하지 않고서 사용자가 사용할 수 있다는 것을 말합니다.

6. 도구 평가

6. 도구 평가

JMeter는 테스트를 수행하여 그 결과를 확인할 수 있도록 도와주는 테스트 어플리케이션으로 특히 개발한 Web Application 성능을 테스트하는데 유용한 도구입니다.

JMeter는 순수 100% 자바로 작성된 프로그램으로 JDK 1.3이상 버전이 설치된 운영체제라면 어디서든지 사용할 수 있습니다.

7. 용어집

7. 용어집

- 본 매뉴얼에서 사용하고 있는 용어가 정리되어 있습니다.

용어	설명
테스트 플랜 (Test Plan)	JMeter가 수행할 일련의 실행 코드 집합을 말합니다.
Thread Group	테스트 플랜이 수행할 하나의 Group을 말합니다.
Duration Assertion	지정한 시간 내에 응답결과를 리턴 하는 지의 여부를 검증하는 것을 말합니다.