
效能測試實務 與工具Jmeter應用

郭忠義

jykuo@ntut.edu.tw

臺北科技大學資訊工程系

Outline

- Lab. 1 新增執行緒群組錄製測試
- Lab. 2 新增檢視報表
- Lab. 3 設定執行緒屬性
- Lab. 4 設定高斯隨機計時器
- Lab. 5 設定測試資料檔案
- Lab. 6 篩選檔案
- Lab. 7 遠端控制測試
- Lab. 8 Redmine效能測試與伺服器監控
- Lab. 9 自訂變數- Random Variable
- Lab. 10 監控效能PerfMon

JMeter 簡介

- JMeter以Java開發，用於Web應用程式功能和效能測試。
- 相關下載網頁：<http://jmeter.apache.org/>
- 使用者手冊：<http://jmeter.apache.org/usermanual/index.html>

Jmeter 下載



The screenshot shows the Apache JMeter download page. At the top, there's a navigation bar with links like 'About', 'Download', 'Documentation', 'Tutorials (PDF format)', and 'Binaries'. A red box highlights the 'Download' link. Below the navigation, there's a sidebar with links for 'Overview' and 'License'. The main content area has a large 'Download Apache JMeter' heading. A red box highlights the 'Download Releases' link under the 'Download' section. Below this, there's a paragraph about using mirrors for downloads and a 'Other mirrors:' dropdown set to 'http://apache.stu.edu.tw/'. A red box highlights the 'KEYS' link. At the bottom, there's a section for 'Apache JMeter 3.0 (Requires Java 7 or later)' with a 'Binaries' link and a red box highlighting the download links for 'apache-jmeter-3.0.tgz md5 sha pgp' and 'apache-jmeter-3.0.zip md5 sha pub'. A red box also highlights the text '點選此處下載'.

Apache JMeter - Dc x

jmeter.apache.org/download_jmeter.cgi

應用程式 已匯入 Google 翻譯 Yahoo奇摩 Kuo Jong Yih Ho Yahoo奇摩字典 臺灣博碩士論文 Google 學術搜尋 首頁 - Dropbox 中央氣象局全球觀 May 9-13 ApacheCon North America 2016

THE APACHE SOFTWARE FOUNDATION

APACHE JMeter™

ApacheCon Vancouver BC Canada May 9-13 2016

About

- [Overview](#)
- [License](#)

Download

- [Download Releases](#)
- [Release Notes](#)

Documentation

- [Get Started](#)
- [User Manual](#)
- [Best Practices](#)
- [Component Reference](#)
- [Functions Reference](#)
- [Change History](#)
- [Javadocs](#)
- [Building JMeter and Add-Ons](#)
- [JMeter Wiki](#)
- [FAQ \(Wiki\)](#)

Tutorials (PDF format)

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

Download Apache JMeter

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 all the mirrors.

You are currently using <http://ftp.tc.edu.tw/pub/Apache/>. 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: Change

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 md5 checksum from the main site. The **SHA** link downloads the sha checksum from the main site. Please **verify the integrity** of the downloaded file.

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

KEYS

Apache JMeter 3.0 (Requires Java 7 or later)

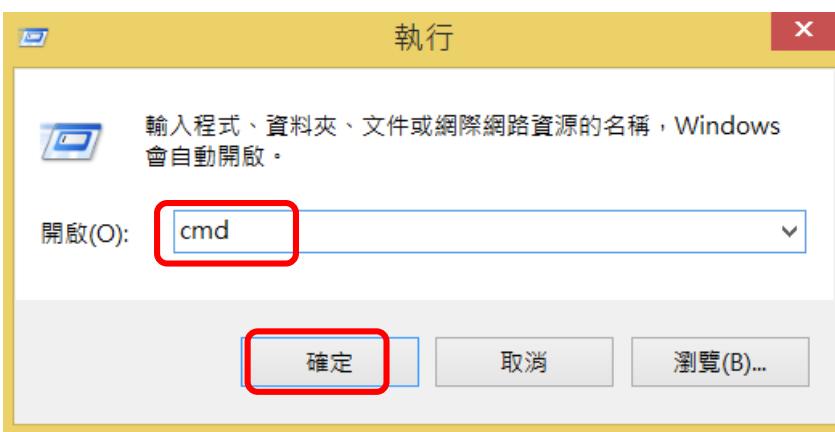
Binaries

點選此處下載

apache-jmeter-3.0.tgz md5 sha pgp
apache-jmeter-3.0.zip md5 sha pub

查詢 IP

- 右鍵 - 執行 - cmd - 確定
- 命令提示字元 - ipconfig

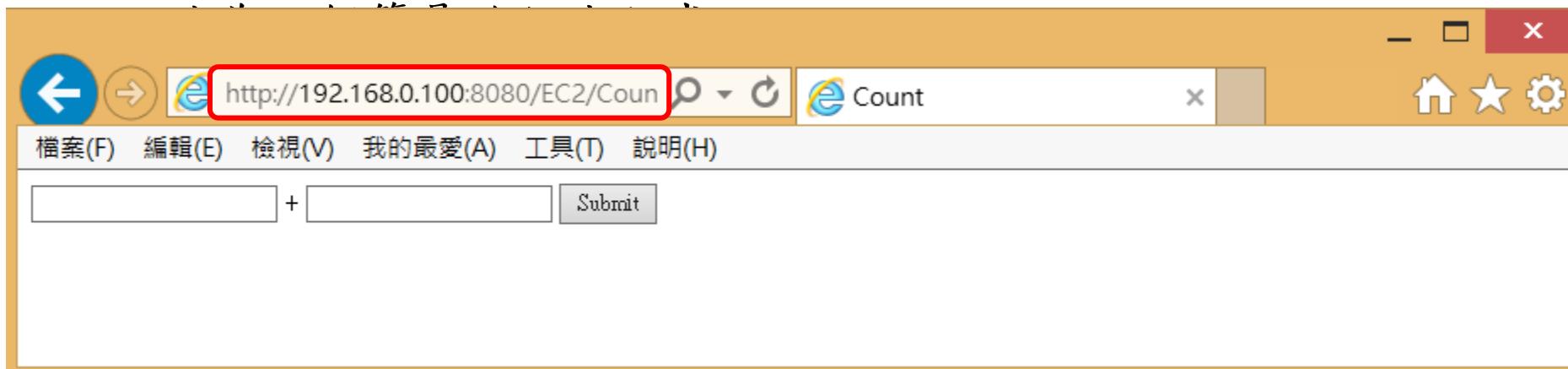


建置環境

- 下載JDK
- 下載ECLIPSE J2EE
- 下載Apache Web Server
- 下載count.jsp
- ECLIPSE J2EE執行count.jsp

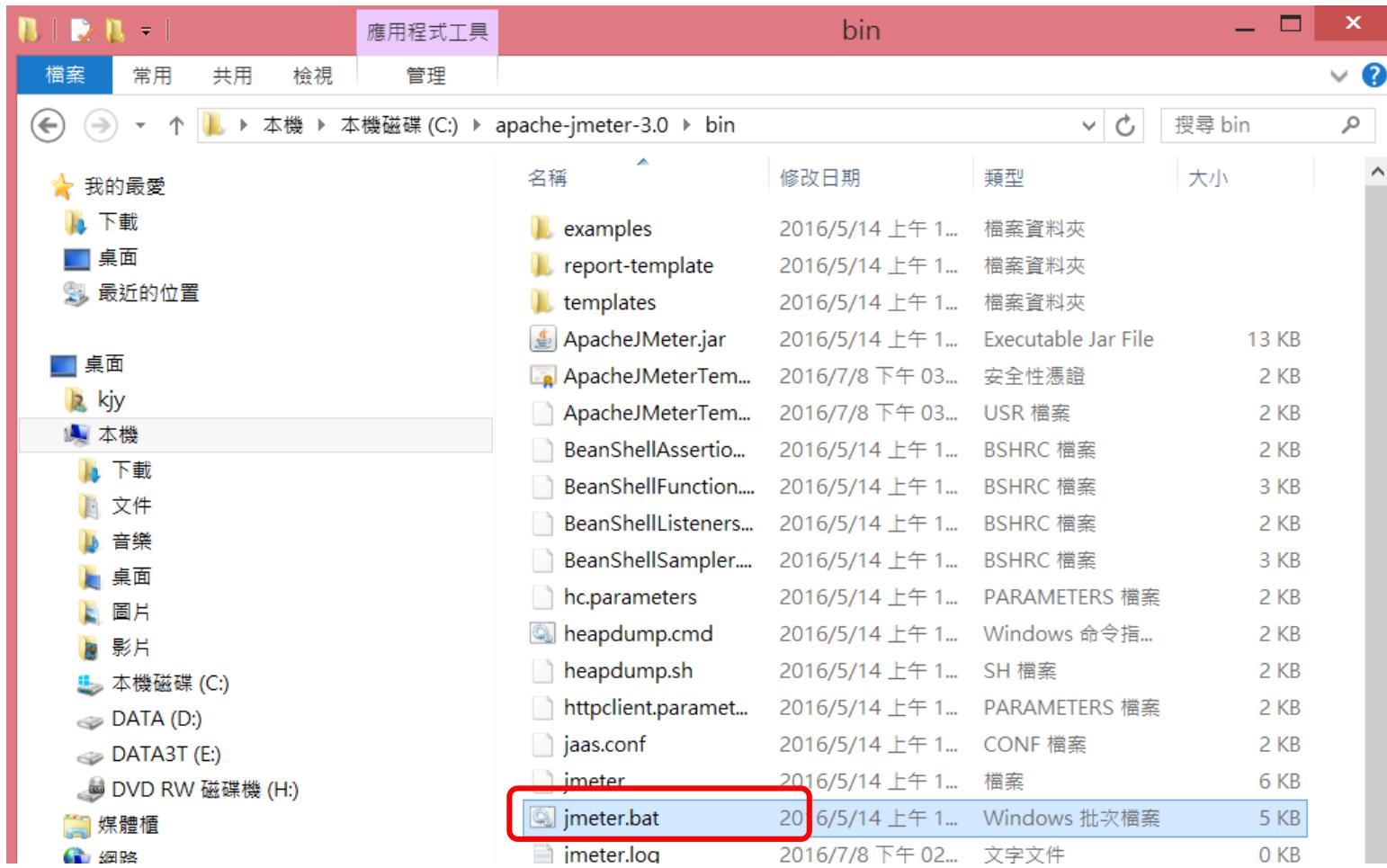
受測網站

- 在虛擬機 Terminal 下指令
 - sudo ls (輸入密碼)
 - wget https://myweb.ntut.edu.tw/~jykuo/train/tomcat.sh
 - source tomcat.sh
- Count



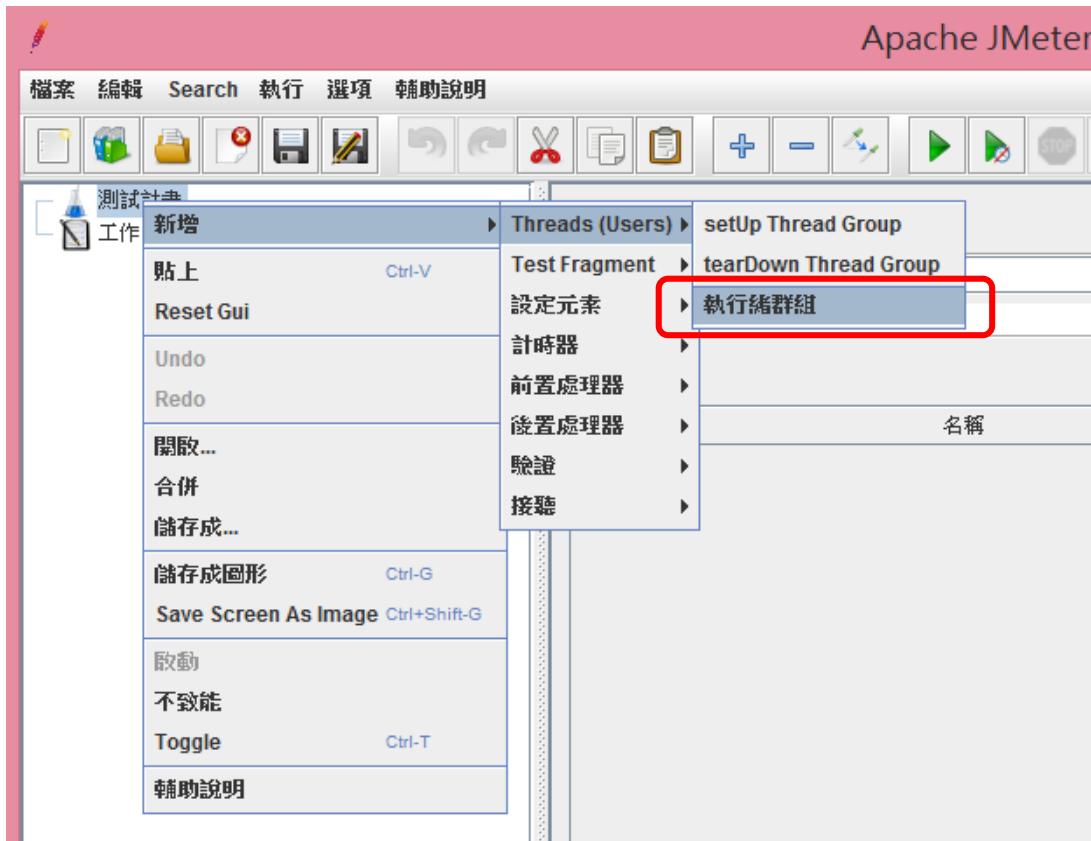
開啟JMeter

- 下載後解壓縮，點選bin資料夾內的jmeter.bat



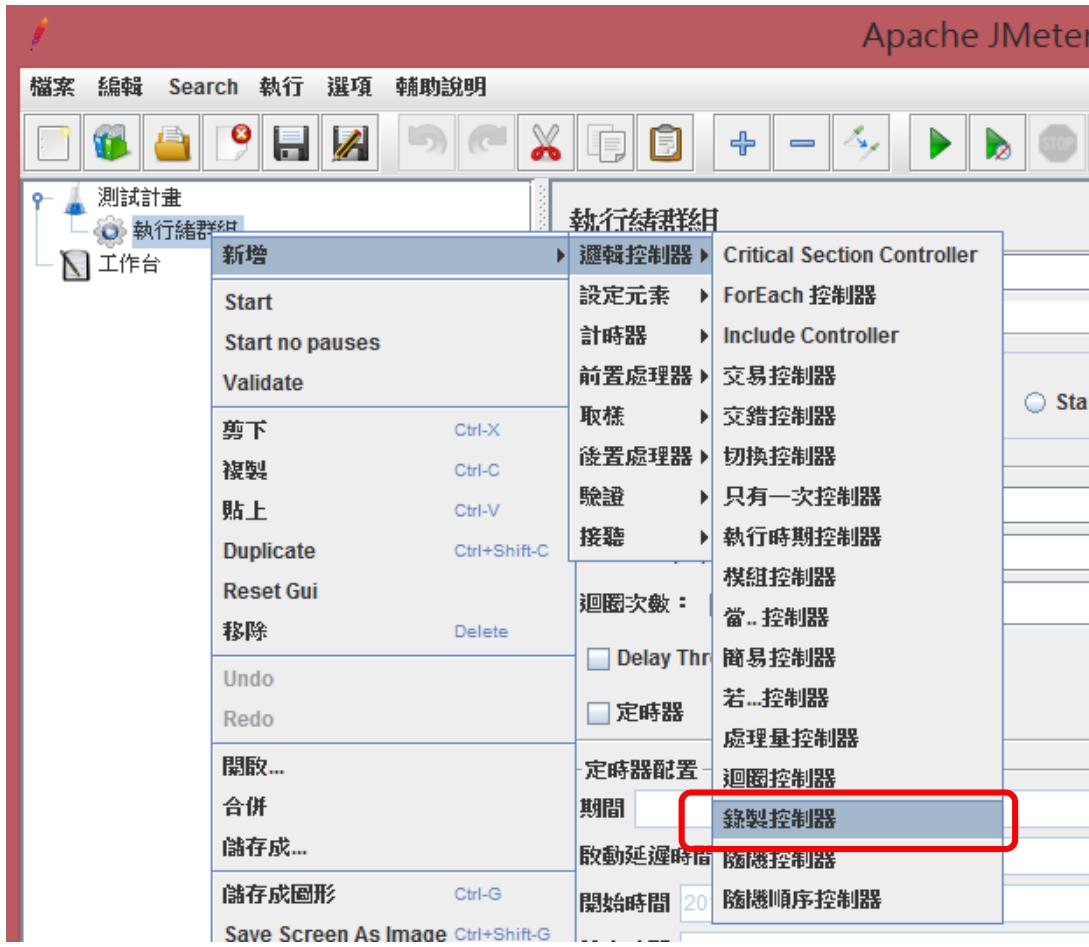
Lab. 1 新增執行緒群組錄製測試

- 新增執行緒群組並開始新的測試工作
 - 測試計畫右鍵新增Threads(Users): 執行緒群組



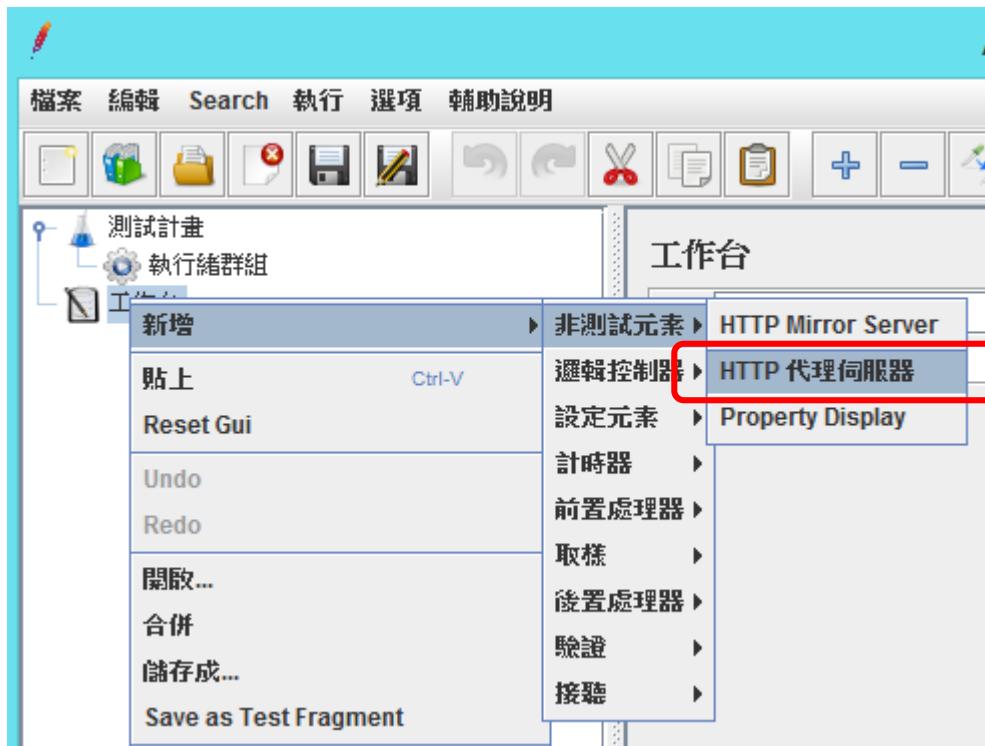
Lab. 1 新增執行緒群組錄製測試

- 新增錄製控制器以錄製測試動作 (測試劇本)



Lab. 1 新增執行緒群組錄製測試

- 工作台右鍵新增非測試元素: HTTP代理伺服器



Lab. 1 新增執行緒群組錄製測試

Apache JMeter (3.0 r1743807)

檔案 編輯 Search 執行 選項 幫助說明

測試計畫 執行緒群組 錄製控制器 工作台

HTTP 代理伺服器

名稱: HTTP 代理伺服器
備註:

Global Settings
端口: 8888 HTTPS Domains: []

TEST plan Content
目標控制器: 使用錄製控制器
群組: 不要將取樣分群組
HTTP Sampler settings
Type: []
Content-type filter
Include: []
要包含的型式:
除外的型式:
新增 [] 刪除 [] Add from Clipboard []

8888(設定時要注意本機上8888埠是否有其他軟體占用)

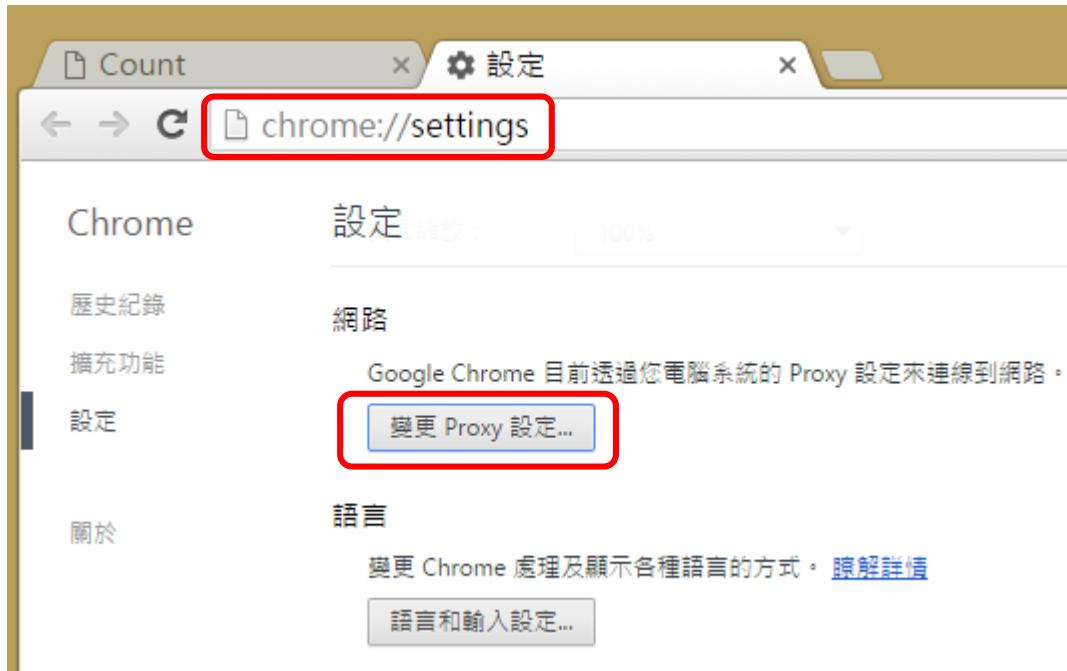
新增 [] 刪除 [] Add from Clipboard [] Add suggested Excludes []

Notify Child Listeners of filtered samplers
 Notify Child Listeners of filtered samplers

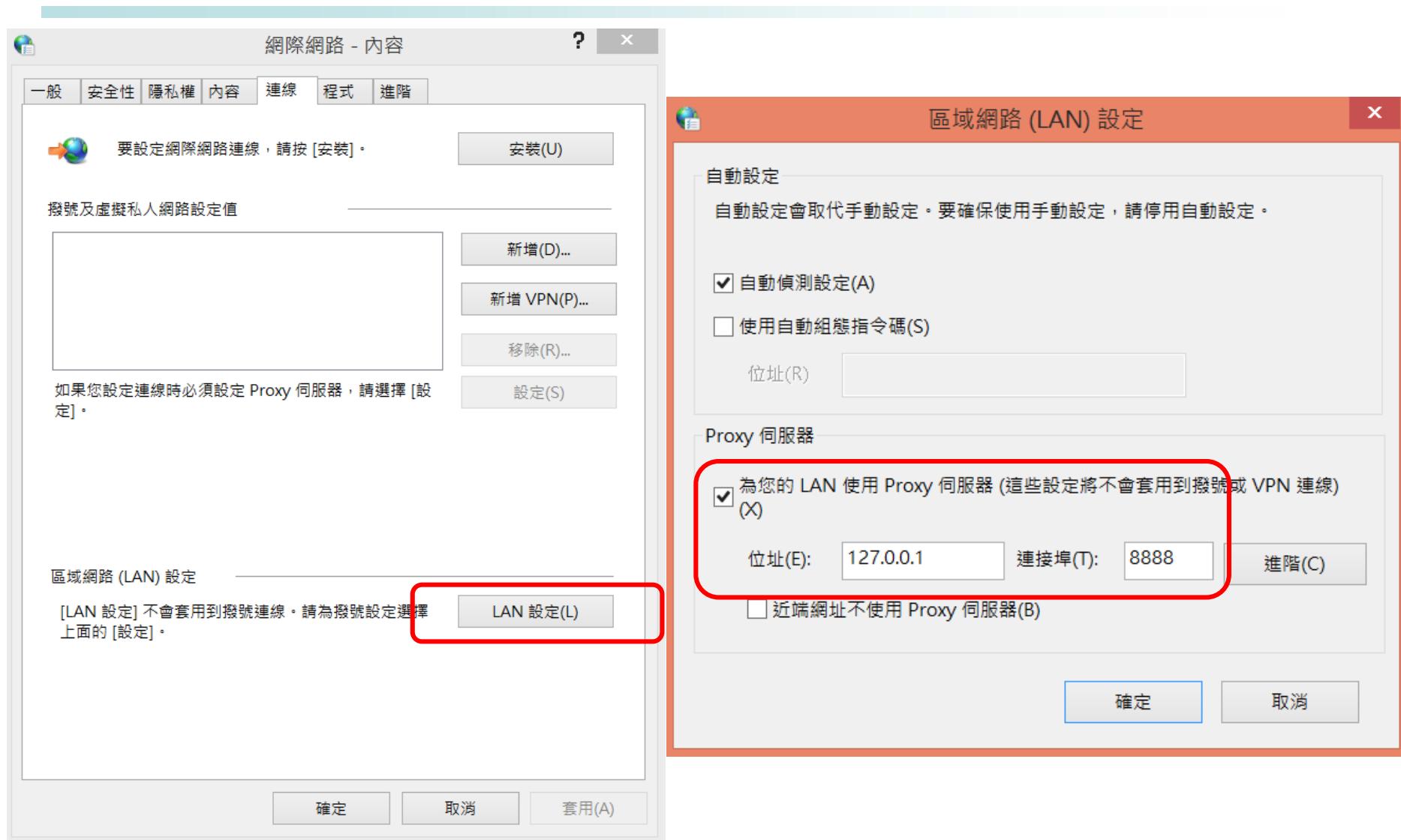
開始 [] 停止 [] 重新啟動 []

Lab. 1 新增執行緒群組錄製測試

- 設定瀏覽器使用Proxy 伺服器
 - Chrome://settings，顯示進階設定，網路，變更Proxy設定

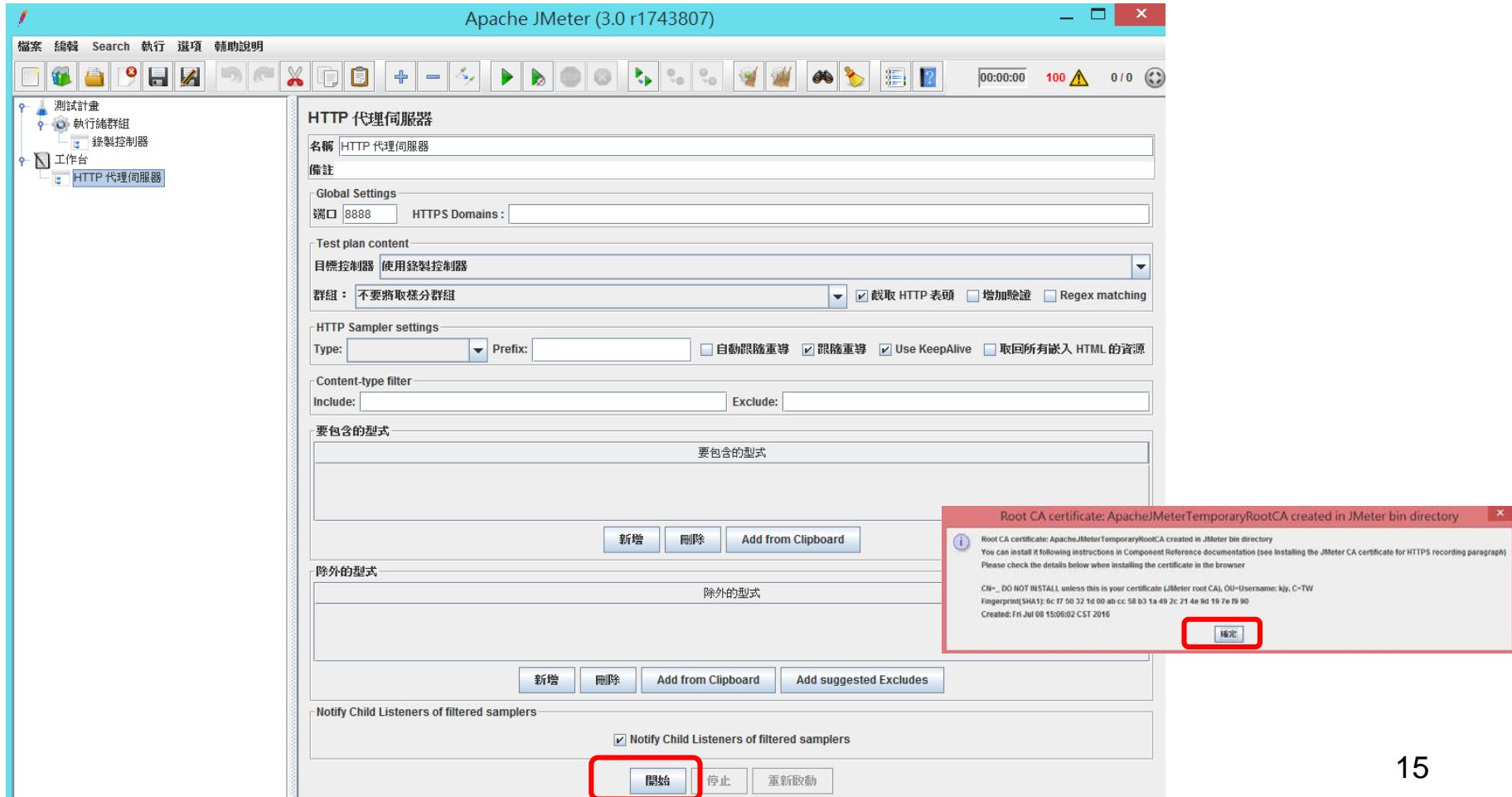


Lab. 1 新增執行緒群組錄製測試



Lab. 1 新增執行緒群組錄製測試

- 按開始，啟動HTTP代理伺服器。按確定。



Lab. 1 新增執行緒群組錄製測試

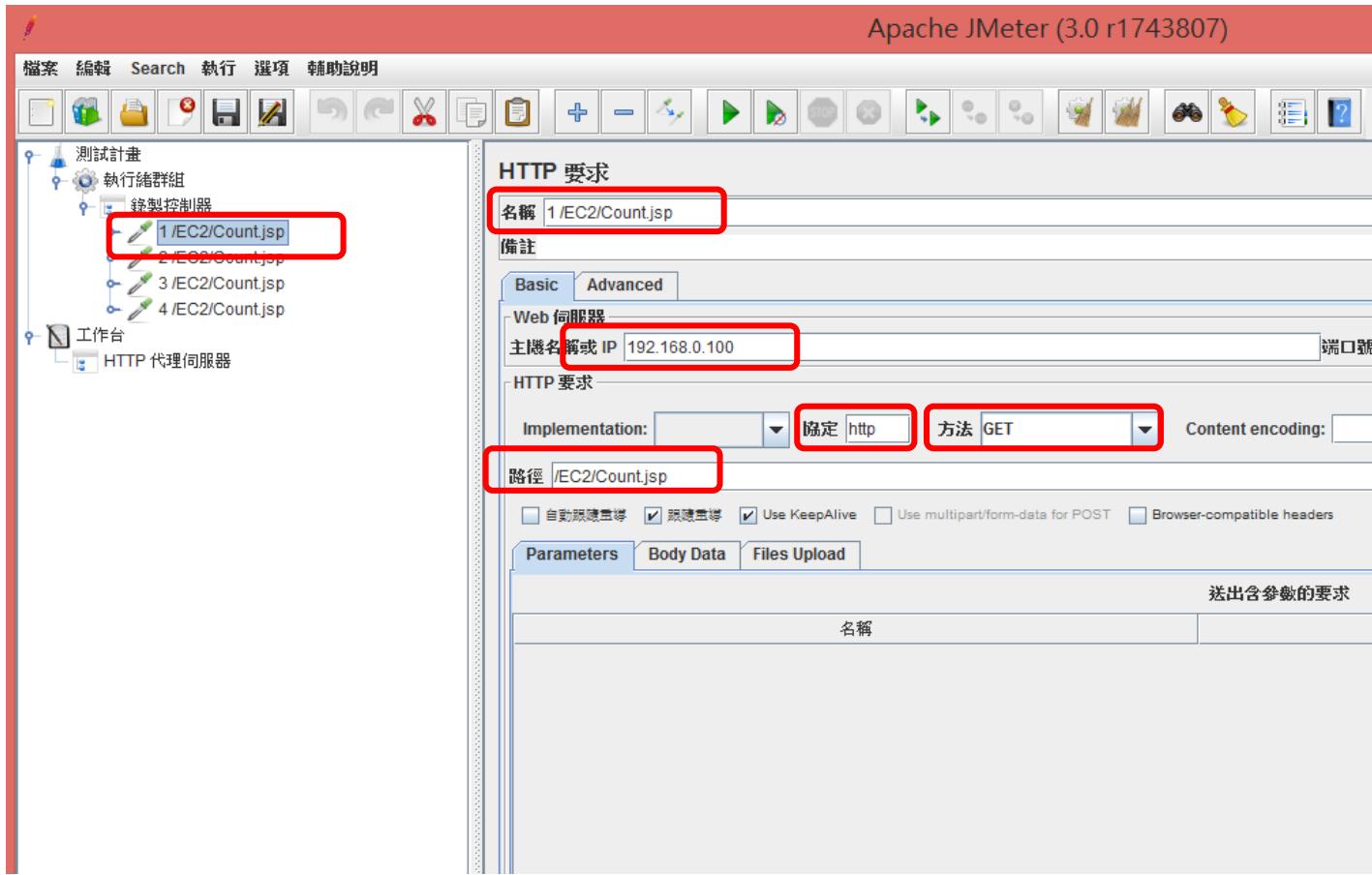
- 開始錄製
 - 使用瀏覽器，於`http://192.168.0.100:8080/EC2/Count.jsp`
 - 進行兩次加法運算

The screenshot shows a web browser window titled "Count". The address bar displays the URL `192.168.0.100:8080/EC2/Count.jsp`. Below the address bar are two input fields: the first contains the number `58`, and the second contains the expression `+ 12`. To the right of these fields is a `Submit` button. All three elements—the first input field, the second input field, and the `Submit` button—are highlighted with red boxes.

The screenshot shows a web browser window titled "Count". The address bar displays the URL `192.168.0.100:8080/EC2/Count.jsp`. The main content area of the browser shows the text `The summation = 70`. At the bottom of the page, there is a link labeled `Try again!`, which is highlighted with a red box.

Lab. 1 新增執行緒群組錄製測試

- 觀察HTTP代理伺服器，自動錄製加法運算過程的Http要求



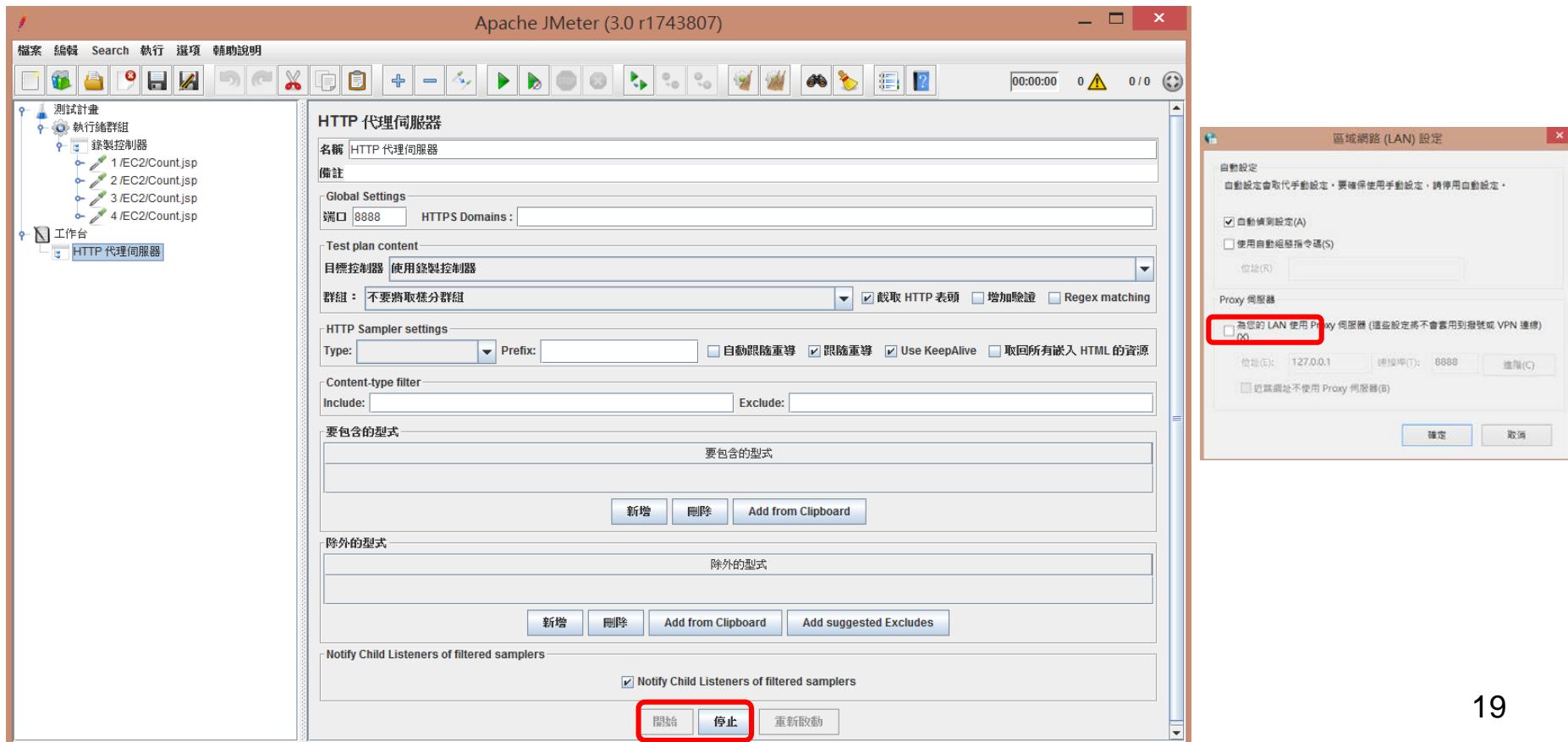
Lab. 1 新增執行緒群組錄製測試

- 觀察HTTP代理伺服器，自動錄製加法運算過程的Http要求
 - 在“送出含參數的要求”內，summandField:12，addendField:13，submit: Submit

The screenshot shows the Apache JMeter interface version 3.0 r1743807. On the left, the Test Plan tree shows a Recording Controller with four sub-samples named 1/EC2/Count.jsp, 2/EC2/Count.jsp, 3/EC2/Count.jsp, and 4/EC2/Count.jsp. The second sample, 2/EC2/Count.jsp, is selected and highlighted with a red box. The main panel displays the 'HTTP 請求' (HTTP Request) configuration. The 'Name' field is set to '2 /EC2/Count.jsp'. The 'Host' field contains '192.168.0.100'. The 'Port' field is set to '8080'. Under the 'HTTP Request' tab, the 'Implementation' dropdown is set to '協定 http', the 'Method' dropdown is set to 'POST', and the 'Content encoding' dropdown is set to 'big5'. The 'Path' field is set to '/EC2/Count.jsp'. Below these settings, there are several checkboxes: '自動跟蹤重複' (Auto Follow Redirects), '跟蹤重複' (Follow Redirects) which is checked, 'Use KeepAlive' which is checked, 'Use multipart/form-data for POST' which is unchecked, and 'Browser-compatible headers' which is unchecked. At the bottom of the request configuration, there are three tabs: 'Parameters', 'Body Data', and 'Files Upload'. The 'Parameters' tab is selected, showing a table for '送出含參數的要求' (Send Parameters). The table has two rows: one for 'summandField' with value '12', and one for 'addendField' with value '13'. A third row for 'submit' with value 'Submit' is present but empty. The 'Body Data' and 'Files Upload' tabs are also visible but empty.

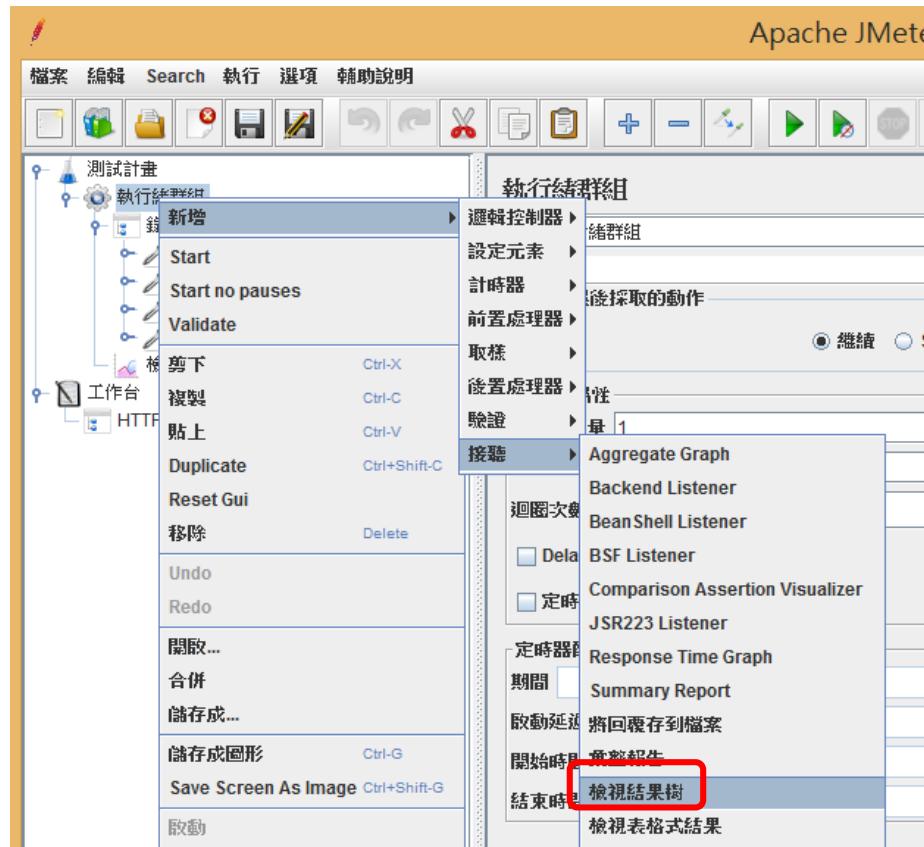
Lab. 1 新增執行緒群組錄製測試

- 按下停止鍵，停止HTTP代理伺服器，取消瀏覽器Proxy伺服器設定



Lab. 1 新增執行緒群組錄製測試

- 新增”檢視結果樹”以觀察結果
 - 在執行緒群組上點選右鍵，選擇新增->接聽->檢視結果樹



Lab. 1 新增執行緒群組錄製測試

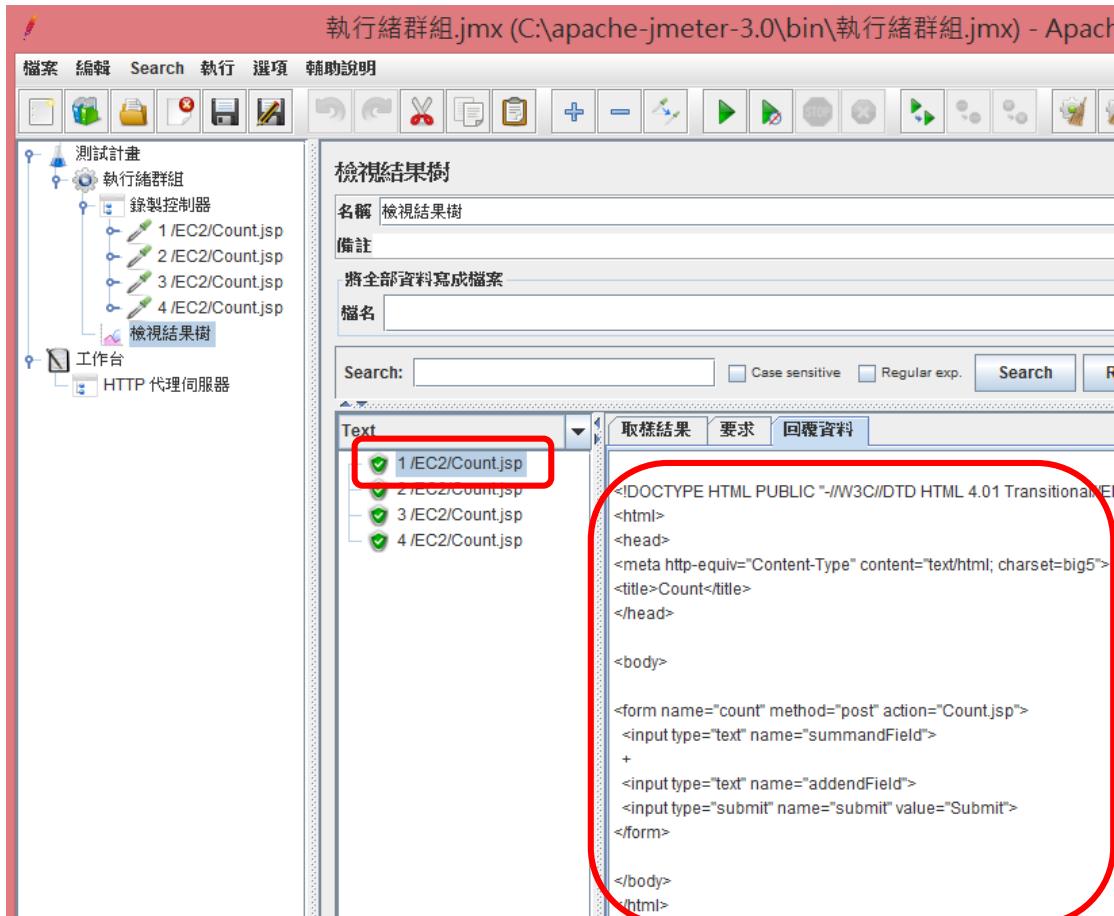
- 點選執行
 - 執行前要求存檔



Screenshot of the JMeter interface showing the "执行" (Execute) menu open. The "开始" (Start) option is highlighted with a red rectangle. The menu also includes options like "Start no pauses", "Remote Shutdown", and "Remote Shutdown All". To the right, the "View Results Tree" window is visible, showing a list of four samples under the path "1 /EC2/Count.jsp", each with a green checkmark indicating success. Sample details are listed below: Latency: 4, Size in bytes: 681, Headers size in bytes: 221, Body size in bytes: 460, Sample Count: 1, Error Count: 0, Data type ("text"|"bin"|""): text, Response code: 200, Response message: OK.

Lab. 1 新增執行緒群組錄製測試

- 執行結束後，點選檢視結果樹觀看測試結果 – 回覆資料



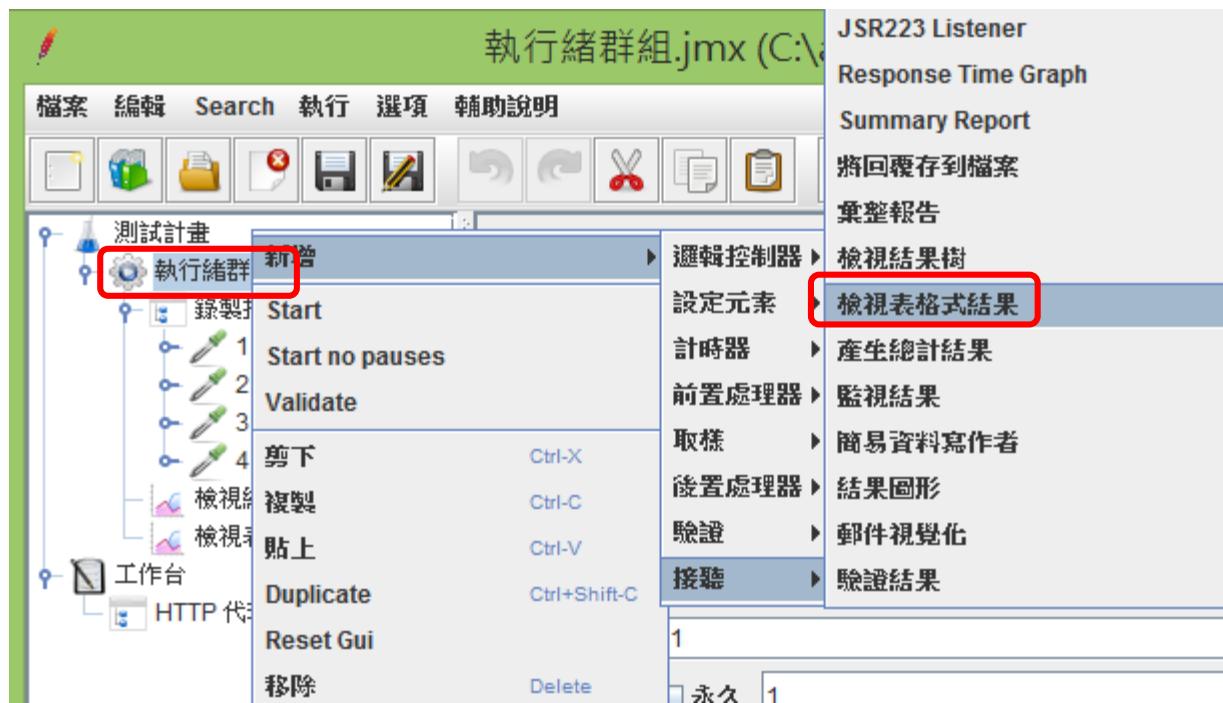
Lab. 1 新增執行緒群組錄製測試

- 執行結束後，點選檢視結果樹觀看測試結果 – 回覆資料

The screenshot shows the Apache JMeter interface with the title bar "執行緒群組.jmx (C:\apache-jmeter-3.0\bin\執行緒群組.jmx) - Apache JMeter". The menu bar includes 檔案, 編輯, Search, 執行, 選項, and 輔助說明. The toolbar contains various icons for file operations and monitoring. The left sidebar shows a tree structure with "測試計畫", "執行緒群組", "錄製控制器", "1 /EC2/Count.jsp", "2 /EC2/Count.jsp", "3 /EC2/Count.jsp", "4 /EC2/Count.jsp", and "檢視結果樹". The "檢視結果樹" node is selected. The main panel displays the "檢視結果樹" (View Results Tree) listener settings: 名稱 (Name) is set to "檢視結果樹", 备註 (Notes) is "將全部資料寫成檔案", and 檔名 (File Name) is empty. Below these are search filters for Search, Case sensitive, and Regular exp., with a Search button. The results table has columns for Text, 取樣結果 (Sampling Result), 要求 (Request), and 回覆資料 (Response). The "Text" column lists four requests: "1 /EC2/Count.jsp", "2 /EC2/Count.jsp", "3 /EC2/Count.jsp", and "4 /EC2/Count.jsp". The "取樣結果" column shows green checkmarks for all. The "回覆資料" column displays the response content for each request. The second request's response is highlighted with a red box around the status code and body: <!DOCTYPE HTML PUBLIC "-//I/W3C//DTD HTML 4.01 Transitional//I</head><body>The summation = 70<P>Try again!</body></html>

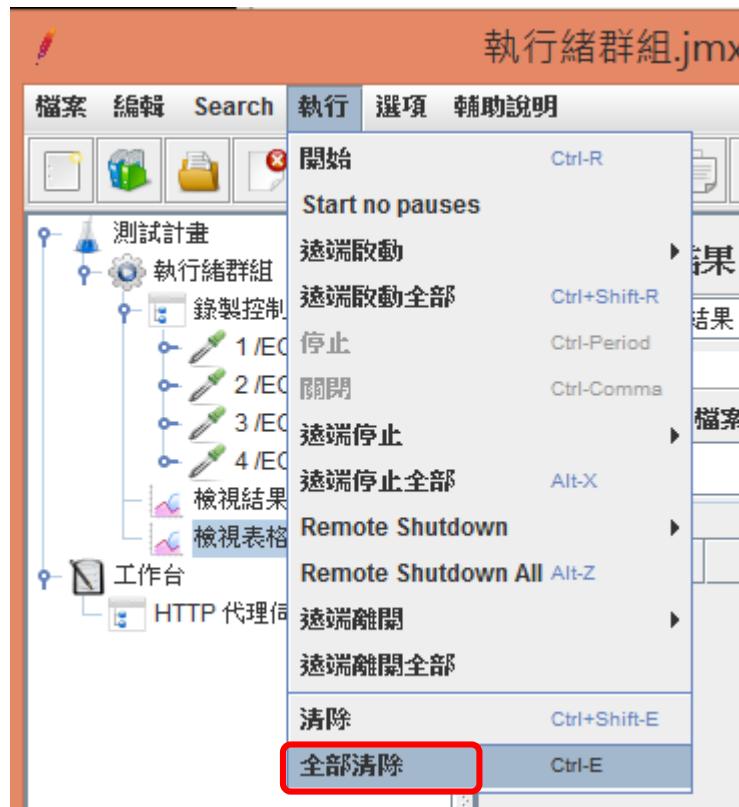
Lab. 2 新增檢視報表

- 執行緒群組 - 新增 - 檢視表格式結果



Lab. 2 新增檢視報表

- 執行 - 全部清除 - 清除之前執行結果
- 執行 - 開始



Lab. 2 新增檢視報表

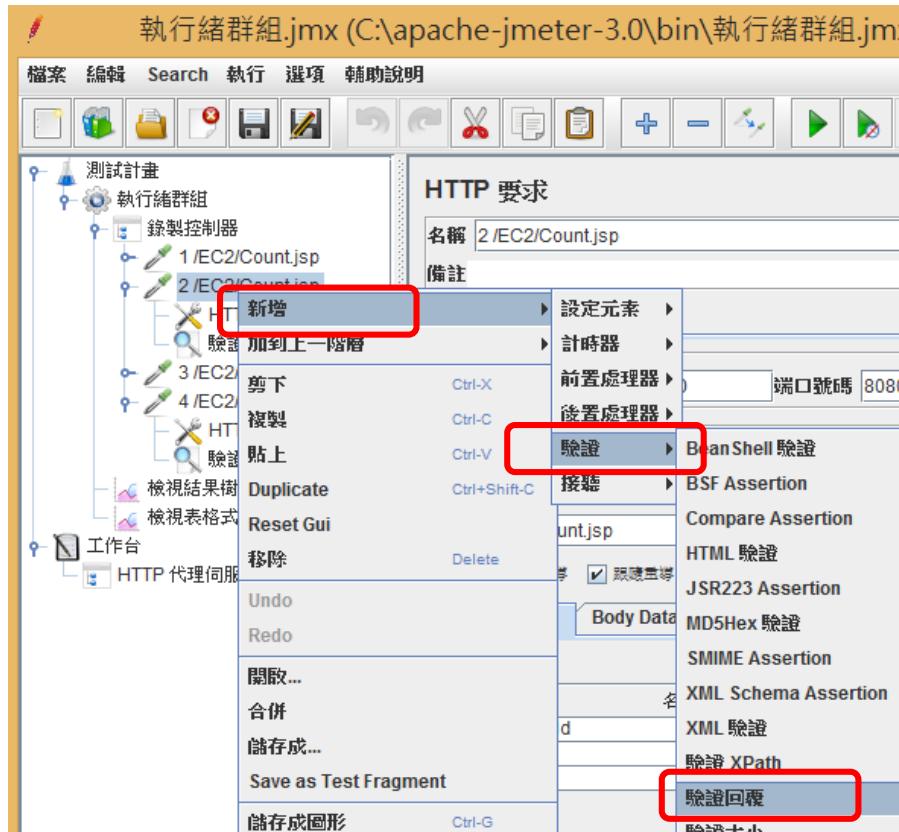
- 檢視表格式結果執行 – 開始

The screenshot shows the Apache JMeter interface with the title bar "執行緒群組.jmx (C:\apache-jmeter-3.0\bin\執行緒群組.jmx) - Apache JMeter (3.0 r1743807)". The menu bar includes 檔案, 編輯, Search, 執行, 選項, and 輔助說明. The toolbar contains various icons for file operations and monitoring. The left sidebar shows the test plan structure, with a red box highlighting the "檢視表格式結果" (View Results Tree) item under the "執行緒群組" (Thread Group) node. The main panel displays the "檢視表格式結果" (View Results Tree) configuration window. It has fields for "名稱" (Name: 檢視表格式結果), "備註" (Notes), and "將全部資料寫成檔案" (Save all data to file). Below these are buttons for "瀏覽..." (Browse...), "Log/Display Only:" (checkboxes for "只記錄錯誤" (Log errors only) and "Successes"), and "設定" (Settings). The bottom half of the screen shows a table of results with four rows. The columns are: 取樣編號 #, Start Time, Thread Name, Label, 取樣時間(微秒), Status, 位元組, Latency, and Connect Time(ms). The "Latency" and "Connect Time(ms)" columns are highlighted with red boxes. The data in the table is as follows:

取樣編號 #	Start Time	Thread Name	Label	取樣時間(微秒)	Status	位元組	Latency	Connect Time(ms)
1	20:11:41.785	執行緒群組 1-1	1 /EC2/Count.jsp	5	✓	681	5	1
2	20:11:41.791	執行緒群組 1-1	2 /EC2/Count.jsp	2	✓	525	2	0
3	20:11:41.794	執行緒群組 1-1	3 /EC2/Count.jsp	1	✓	681	1	0
4	20:11:41.795	執行緒群組 1-1	4 /EC2/Count.jsp	2	✓	527	2	0

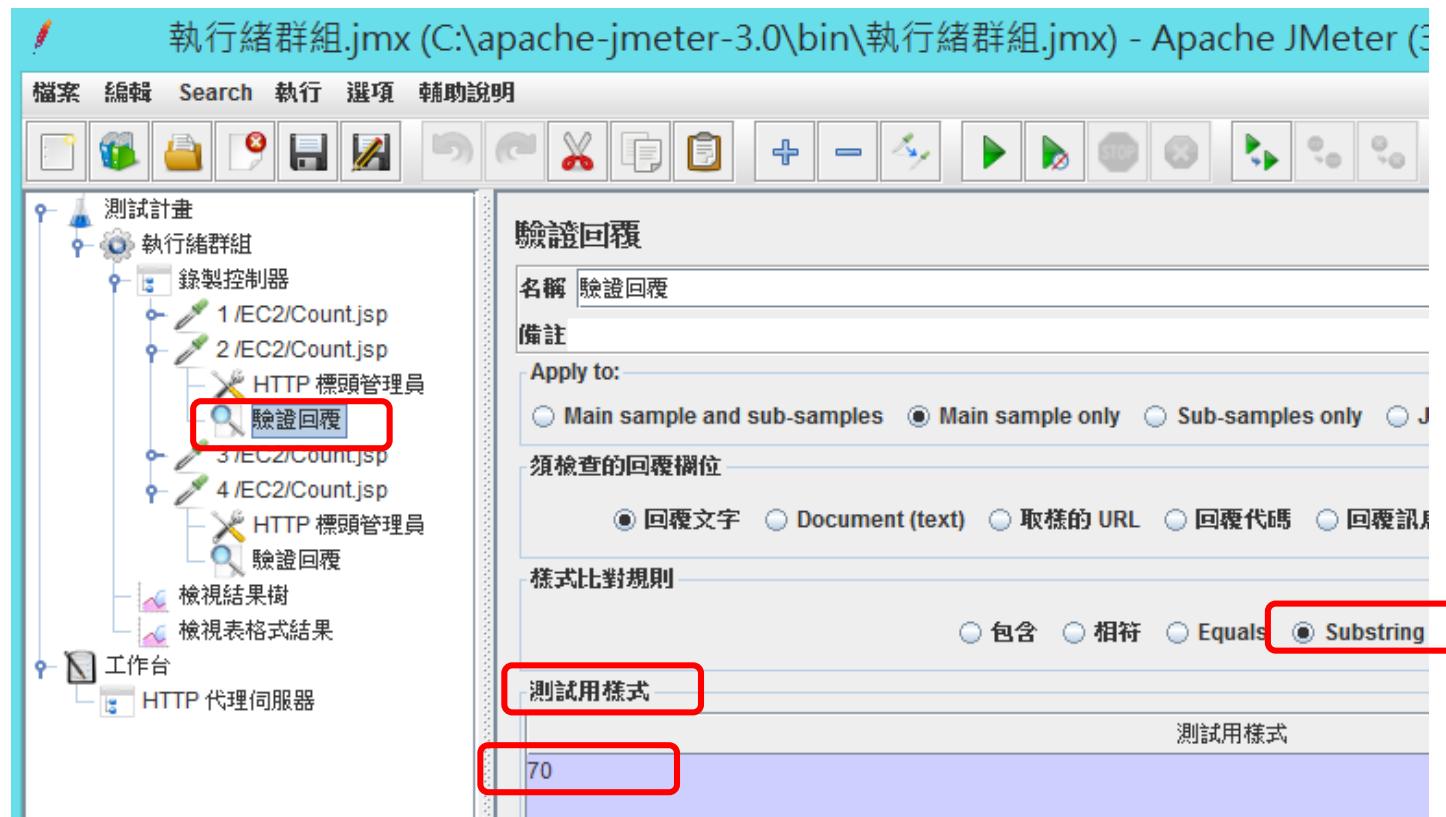
Lab. 2 新增檢視報表

- 在"2 /Count.jsp"，新增「驗證」：「驗證回覆」



Lab. 2 新增檢視報表

- 「驗證回覆」，在「測試用樣式」處，新增，"70"。



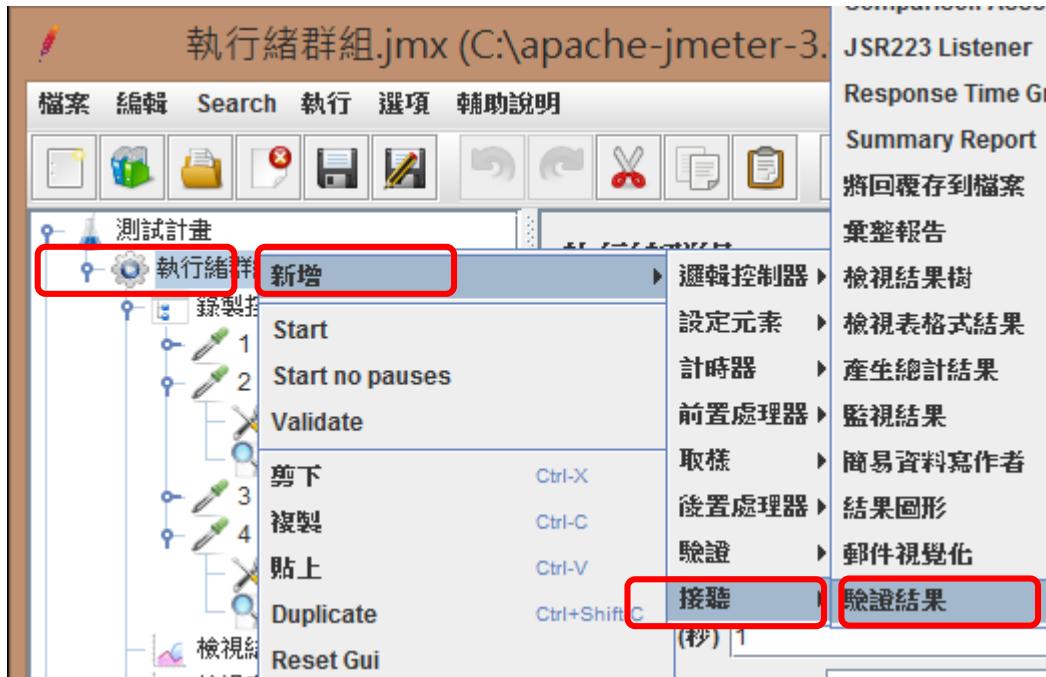
Lab. 2 新增檢視報表

- 在"4 /Count.jsp"，新增「驗證」：「驗證回覆」，並在「測試用樣式處」新增“1011”。



Lab. 2 新增檢視報表

- 執行緒群組 - 新增 - 接聽 - 驗證結果



Lab. 2 新增檢視報表

- 執行 – 全部清除，執行 – 開始

The screenshot shows the Apache JMeter interface with the title bar "執行緒群組.jmx (C:\apache-jmeter-3.0\bin\執行緒群組.jmx) - Apache JMeter (3.0 r1743807)". The menu bar includes 檔案, 編輯, Search, 執行, 選項, 輔助說明. The toolbar contains various icons for file operations and monitoring. The left sidebar shows a hierarchical tree of test elements under "執行緒群組": 錄製控制器 (with 1 /EC2/Count.jsp, 2 /EC2/Count.jsp), HTTP 標頭管理員 (驗證回複), 錄製控制器 (with 3 /EC2/Count.jsp, 4 /EC2/Count.jsp), HTTP 標頭管理員 (驗證回複). The "檢視結果樹" and "檢視表格式結果" nodes are also listed. The "驗證結果" node is highlighted with a red box. The main panel displays the "驗證結果" configuration with "名稱" set to "驗證結果", "備註" empty, and "將全部資料寫成檔案" checked. The "檔名" field is empty, and the "Log/Display Only" section has three checkboxes: "只記錄錯誤" (checked), "Successes" (unchecked), and "設定" (unchecked). Below this, the "驗證:" section lists four items: 1 /EC2/Count.jsp, 2 /EC2/Count.jsp, 3 /EC2/Count.jsp, and 4 /EC2/Count.jsp. The fourth item is highlighted with a red box, and its details show an assertion failure: "驗證回複 : Test failed: text expected to contain /1010/".

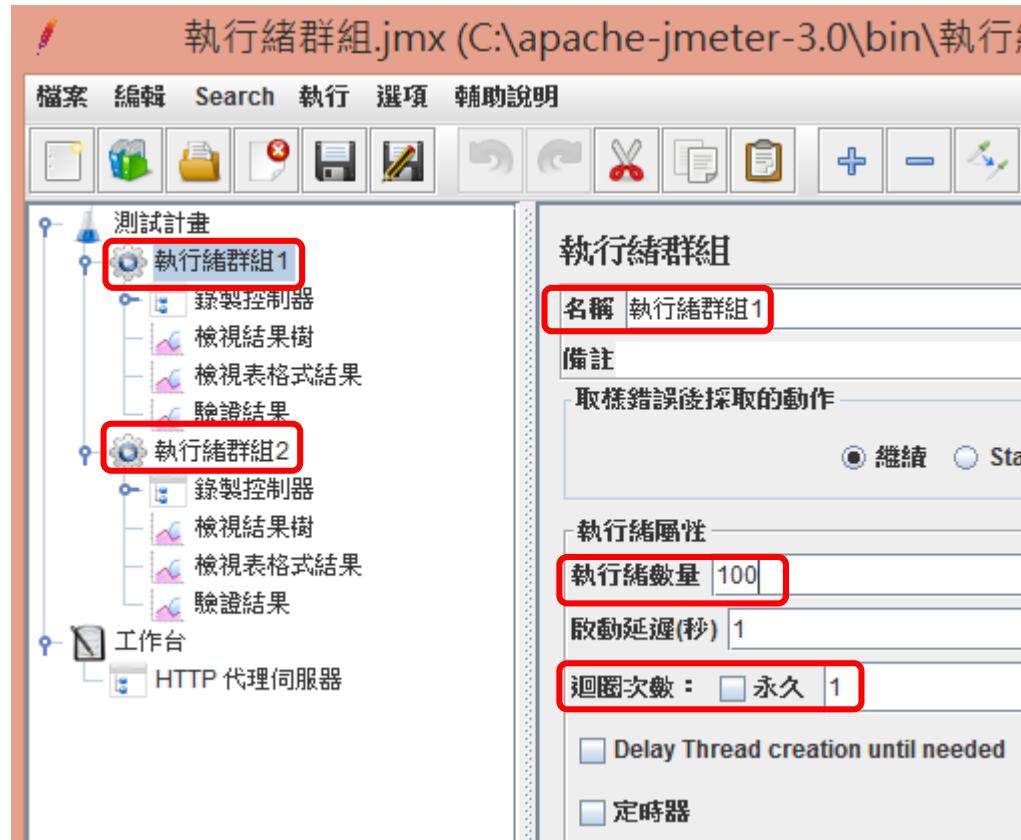
Lab. 3 設定執行緒屬性

- 製作兩個執行緒群組
 - 複製執行緒群組：執行緒群組 - 複製
 - 貼上執行緒群組：測試計畫 - 貼上
 - 修改執行緒群組名稱：執行緒群組1，執行緒群組2



Lab. 3 設定執行緒屬性

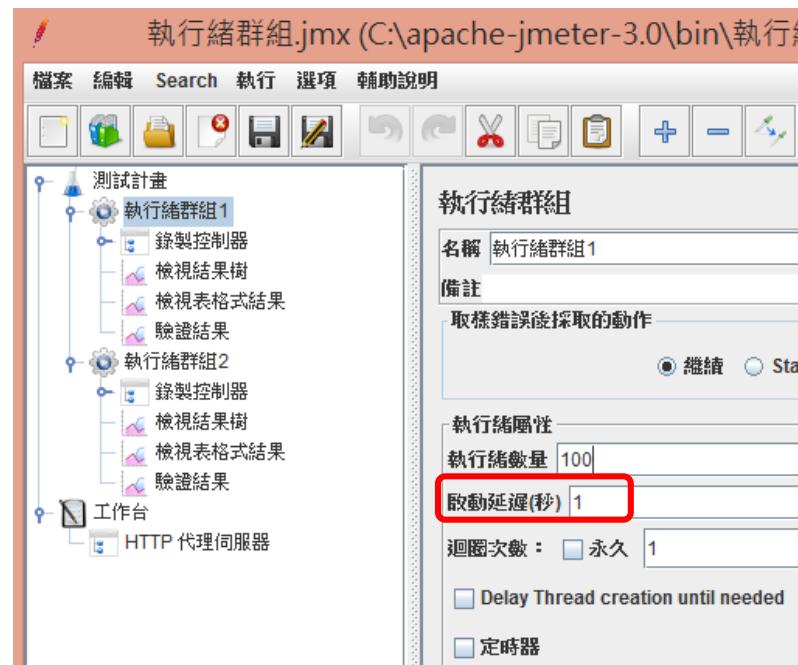
- 將執行緒群組1，執行緒數量設為100，迴圈次數設為1
- 將執行緒群組2，執行緒數量設為1，迴圈次數設為100



Lab. 3 設定執行緒屬性

- 啟動延遲時間為0，立即建立所有執行緒
- 啟動延遲時間為 T 秒，全部執行緒數設置成 N，
 - 每隔 T/N 秒建立一個執行緒。
- 計算合理啟動延遲時間
 - 假設執行緒數為 100，
 - 估計點擊率(Rate)為每秒 10 次
 - 結果 $100/10 = 10$ 秒。
 - Rate 與 aggregate graph 處理量比對

錯誤率	處理量	每秒仟位元組
0.00%	2.2/sec	1.4
0.00%	2.2/sec	1.1
0.00%	2.2/sec	1.4
0.00%	2.1/sec	1.1
0.00%	2.1/sec	1.4



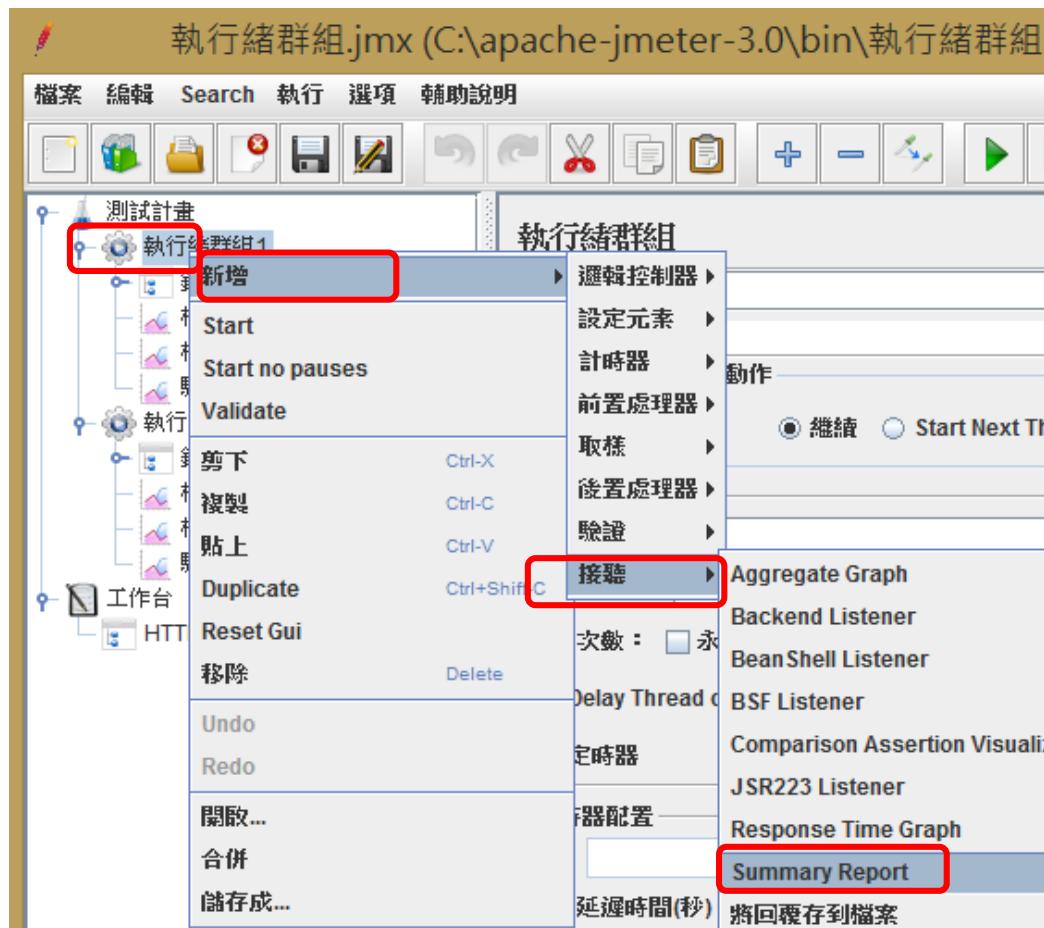
Lab. 3 設定執行緒屬性

- 勾選定時器
 - 期間，執行幾秒
 - 啟動延遲時間
 - 0秒，立即建立所有執行緒
 - 啟動延遲時間為 T 秒，全部執行緒數設置成 N，
 - 每隔 T/N 秒建立一個執行緒。
 - 開始時間
 - 結束時間



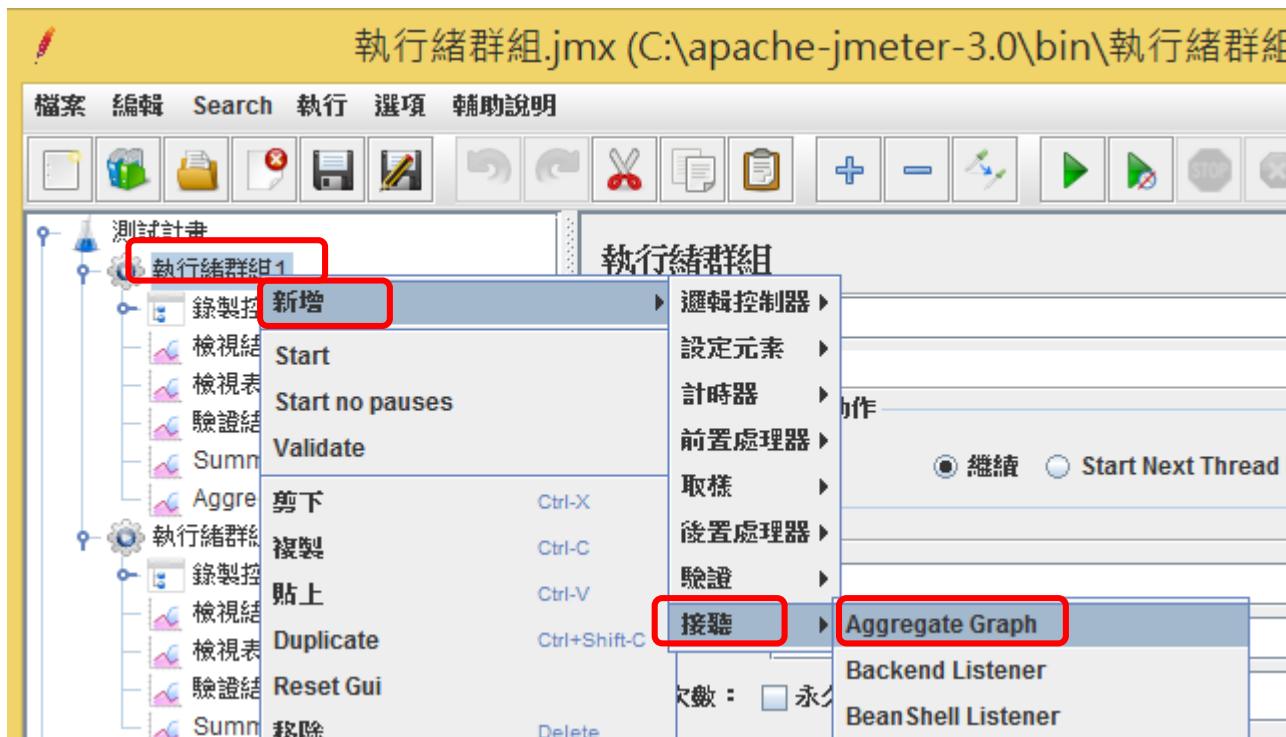
Lab. 3 設定執行緒屬性

- 分別在兩個執行緒內建立，新增 – 接聽 - Summary Report



Lab. 3 設定執行緒屬性

- 分別在兩個執行緒內建立，新增 – 接聽 - Aggregate Graph



Lab. 3 設定執行緒屬性

- 執行 – 開始 – Summary Report
 - 平均值Average：一個請求Request的平均回應時間。
 - 中間值Median：中位數，50%使用者的回應時間。
 - 90% Line：90%使用者的回應時間。
 - Min：最小回應時間。
 - Max：最大回應時間。
 - Error%：錯誤率，測試中出現錯誤的請求數量/請求總數。
 - Throughput：處理量，每秒完成的請求數（Request per Second）。
 - 每秒仟位元KB/Sec：每秒從伺服器端接收到的資料量。
 - Avg. Bytes – 平均回應位元大小。

Lab. 3 設定執行緒屬性

- web應用環境，回應時間是從請求到回覆HTML結果耗費的時間
 - 回應時間應包括流覽器重繪HTML頁面時間
 - 負載測試工具不會考慮流覽器的重繪時間。
 - 實際效能測試，可在正常回應時間加一個固定值，如0.5秒。
- 回應時間標準
 - 使用者不會注意到少於0.1秒的延遲
 - 少於1秒的延遲不會中斷使用者的正常思考
 - 延遲時間少於10秒，使用者會繼續等待回應
 - 延遲時間超過10秒，使用者將會放棄並開始其他操作。
 - 延遲時間超過8秒，90%使用者將會放棄並開始其他操作。
 - 考慮絕對回應時間，而非平均回應時間。

Lab. 3 設定執行緒屬性

- 執行 – 開始 – Summary Report
 - 執行緒數量設為 100，迴圈次數設為 1

Label	取樣數	平均值	最小值	最大值	Std. Dev.	錯誤率	處理量	每秒仟位元組	Avg. Bytes
1 /EC2/Count.jsp	100	1	1	3	0.45	0.00%	99.6/sec	66.24	681.0
2 /EC2/Count.jsp	100	0	0	2	0.51	0.00%	99.8/sec	51.17	525.0
3 /EC2/Count.jsp	100	0	0	1	0.49	0.00%	99.8/sec	66.37	681.0
4 /EC2/Count.jsp	100	0	0	1	0.44	100.00%	99.9/sec	51.41	527.0
總計	400	0	0	3	0.53	25.00%	397.6/sec	234.34	603.5

- 執行緒數量設為 1，迴圈次數設為 100

Label	取樣數	平均值	最小值	最大值	Std. Dev.	錯誤率	處理量	每秒仟位元組	Avg. Bytes
1 /EC2/Count.jsp	100	0	0	1	0.49	0.00%	335.6/sec	223.17	681.0
2 /EC2/Count.jsp	100	0	0	1	0.44	0.00%	336.7/sec	172.62	525.0
3 /EC2/Count.jsp	100	0	0	1	0.49	0.00%	335.6/sec	223.17	681.0
4 /EC2/Count.jsp	100	0	0	2	0.46	100.00%	336.7/sec	173.53	527.0
總計	400	0	0	2	0.47	25.00%	1333.3/sec	786.05	603.5

Lab. 3 設定執行緒屬性

- Aggregate Graph

- 執行緒數量設為100，迴圈次數設為1

Label	取樣數	平均值	中間值	90% Line	95% Line	99% Line	最小值	最大值	錯誤率	處理量	每秒仟位元組
1 /EC2/Count...	100	1	1	2	2	2	1	3	0.00%	99.6/sec	66.2
2 /EC2/Count....	100	0	1	1	1	2	0	2	0.00%	99.8/sec	51.2
3 /EC2/Count....	100	0	1	1	1	1	0	1	0.00%	99.8/sec	66.4
4 /EC2/Count....	100	0	1	1	1	1	0	1	100.00%	99.9/sec	51.4
總計	400	0	1	2	1	2	0	3	25.00%	397.6/sec	234.3

- 執行緒數量設為1，迴圈次數設為100

Label	取樣數	平均值	中間值	90% Line	95% Line	99% Line	最小值	最大值	錯誤率	處理量	每秒仟位元組
1 /EC2/Count...	100	0	1	1	1	1	0	1	0.00%	335.6/sec	223.2
2 /EC2/Count....	100	0	1	1	1	1	0	1	0.00%	336.7/sec	172.6
3 /EC2/Count....	100	0	1	1	1	1	0	1	0.00%	335.6/sec	223.2
4 /EC2/Count....	100	0	1	1	1	1	0	2	100.00%	336.7/sec	173.5
總計	400	0	1	1	1	1	0	2	25.00%	1333.3/sec	786.1

Lab. 3 設定執行緒屬性

- 檢視表格式結果

- 執行緒數量設為100，迴圈次數設為1，，第一個執行緒啟動延遲3

取樣編號 #	Start Time	Thread Name	Label	取樣時間(微秒)	Status	位元組	Latency	Connect Time(ms)
1	01:27:16.974	執行緒群組 1 1-1	1 /EC2/Count.jsp	3	✓	681	3	1
2	01:27:16.977	執行緒群組 1 1-1	2 /EC2/Count.jsp	1	✓	525	1	0
3	01:27:16.978	執行緒群組 1 1-1	3 /EC2/Count.jsp	1	✓	681	1	0
4	01:27:16.979	執行緒群組 1 1-1	4 /EC2/Count.jsp	1	✗	527	1	0
5	01:27:16.985	執行緒群組 1 1-2	1 /EC2/Count.jsp	1	✓	681	1	0
6	01:27:16.986	執行緒群組 1 1-2	2 /EC2/Count.jsp	1	✓	525	1	0
7	01:27:16.987	執行緒群組 1 1-2	3 /EC2/Count.jsp	1	✓	681	1	0
8	01:27:16.988	執行緒群組 1 1-2	4 /EC2/Count.jsp	1	✗	527	1	0

- 執行緒數量設為1，迴圈次數設為100

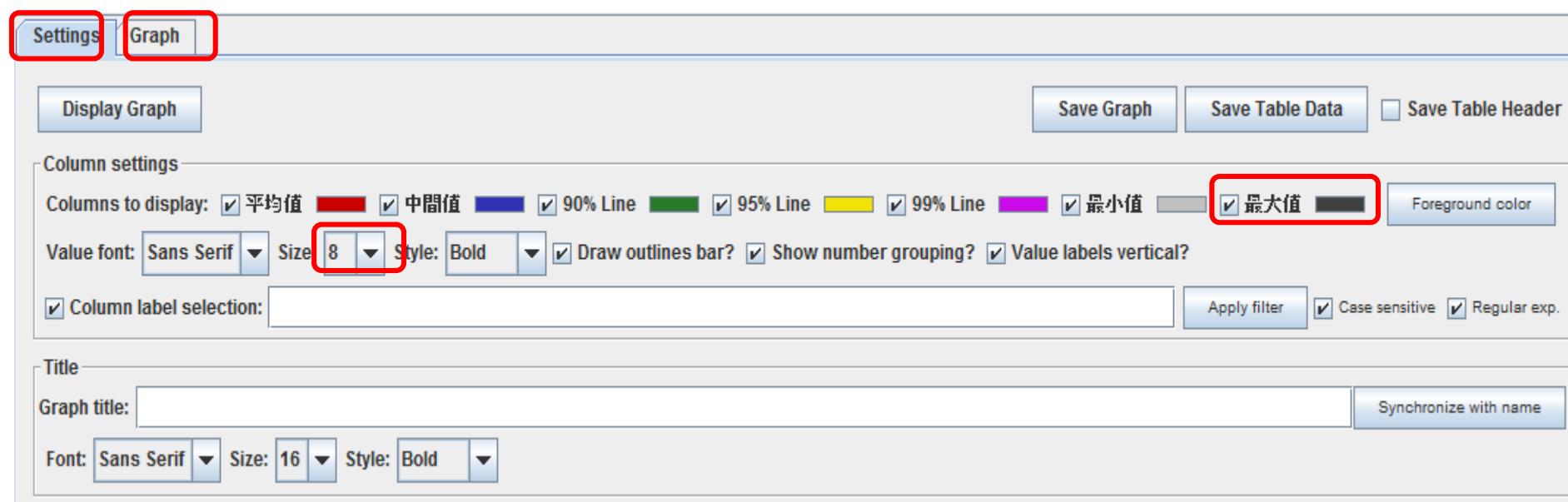
取樣編號 #	Start Time	Thread Name	Label	取樣時間(微秒)	Status	位元組	Latency	Connect Time(ms)
1	01:27:16.987	執行緒群組 2 2-1	1 /EC2/Count.jsp	1	✓	681	1	1
2	01:27:16.989	執行緒群組 2 2-1	2 /EC2/Count.jsp	0	✓	525	0	0
3	01:27:16.989	執行緒群組 2 2-1	3 /EC2/Count.jsp	1	✓	681	1	0
4	01:27:16.990	執行緒群組 2 2-1	4 /EC2/Count.jsp	1	✗	527	1	0
5	01:27:16.991	執行緒群組 2 2-1	1 /EC2/Count.jsp	1	✓	681	1	0
6	01:27:16.992	執行緒群組 2 2-1	2 /EC2/Count.jsp	1	✓	525	1	0
7	01:27:16.993	執行緒群組 2 2-1	3 /EC2/Count.jsp	0	✓	681	0	0
8	01:27:16.993	執行緒群組 2 2-1	4 /EC2/Count.jsp	1	✗	527	1	0

Lab. 3 設定執行緒屬性

- 執行 - 全部清除
- 重新設定 - 執行緒屬性
 - 執行緒群組1 - 執行緒數量設為1000，迴圈次數設為1
 - 執行緒群組2 - 執行緒數量設為1，迴圈次數設為1000
 - 檢視接聽報表有何差異
 - Summary Report
 - 檢視表格式結果
 - Aggregate Graph

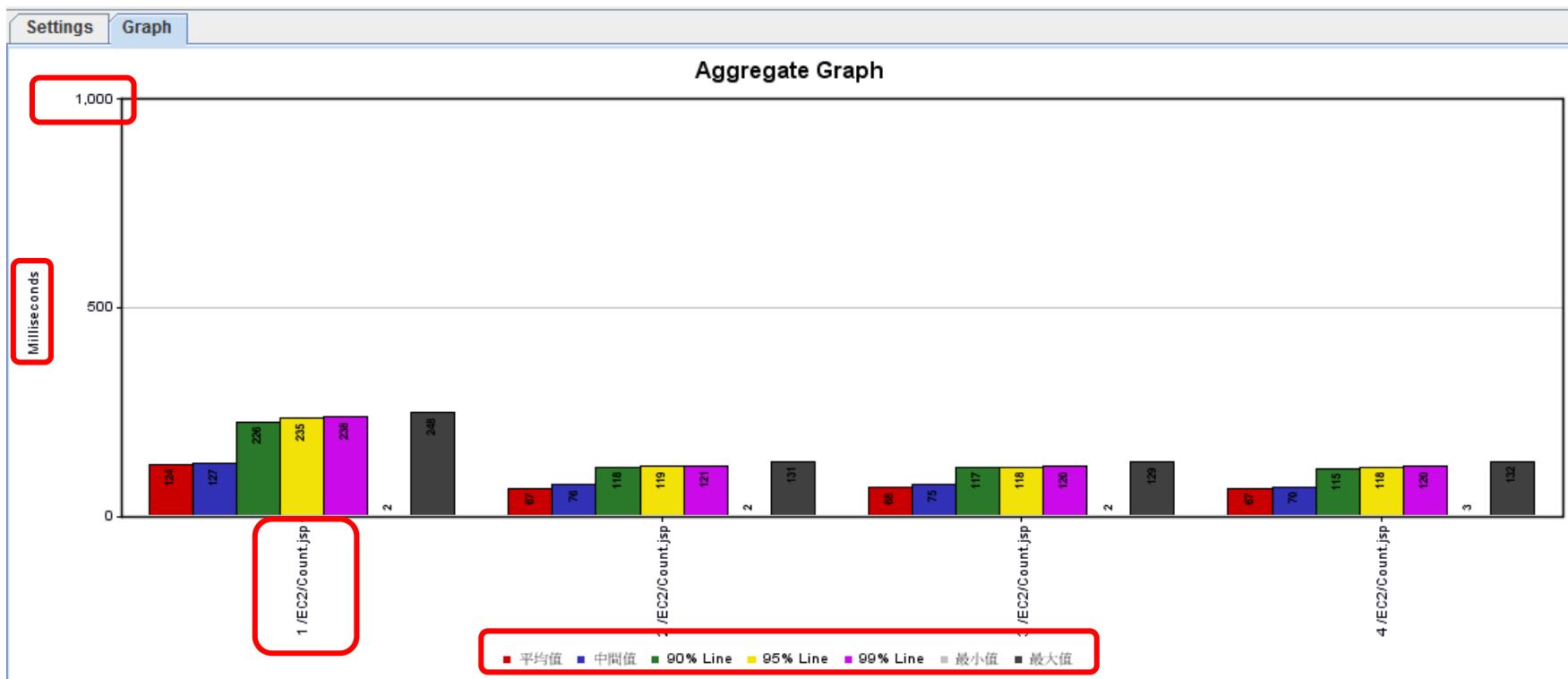
Lab. 3 設定執行緒屬性

- Aggregate Graph , Setting , Graph
 - 執行緒數量設為1000，迴圈次數設為1



Lab. 3 設定執行緒屬性

- Aggregate Graph , Setting , Graph
 - 執行緒數量設為1000，迴圈次數設為1

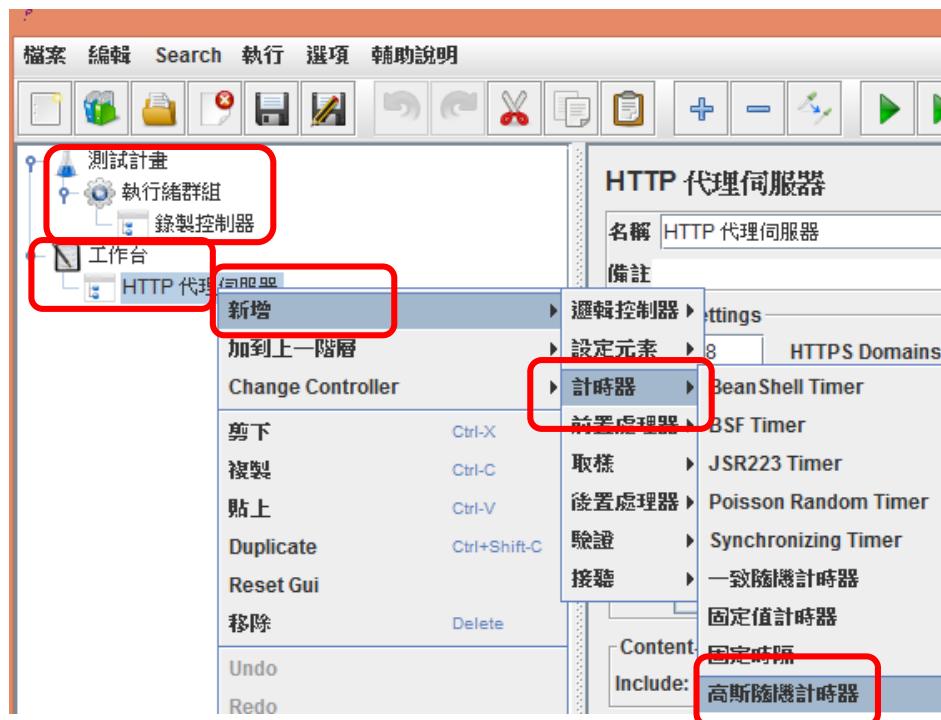


Lab. 4 設定高斯隨機計時器

- 思考時間 (think time)
 - 在兩次成功的請求之間的暫停時間。
 - 使用者需時間閱讀文字內容，或者填表，或者找正確連結等。
 - 未考慮思考時間會導致測試結果的失真。
- 在代理伺服器增加一個高斯隨機計時器
 - 計時器會使相應的取樣器延遲。延時的規則是，在上一個請求被回應並延時指定時間後，下一個被計時器影響的取樣請求才會發送出去。
 - 必須**手動刪除第一個取樣器中自動產生的計時器**，因第一個取樣器不需要定時器。

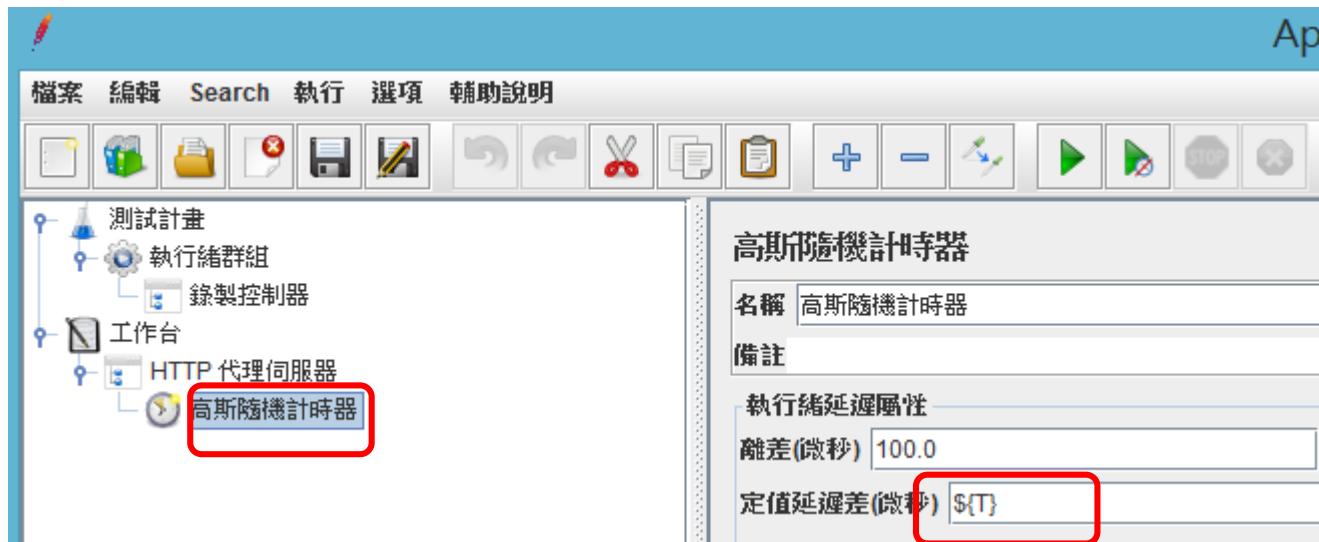
Lab. 4 設定高斯隨機計時器

- 測試計畫 - 新增 - 執行緒群組 - 新增 - 錄製控制器
- 工作台 - 新增 - 非測試元素 - HTTP代理伺服器
- 新增 - 計時器 - 高斯隨機控制器



Lab. 4 設定高斯隨機計時器

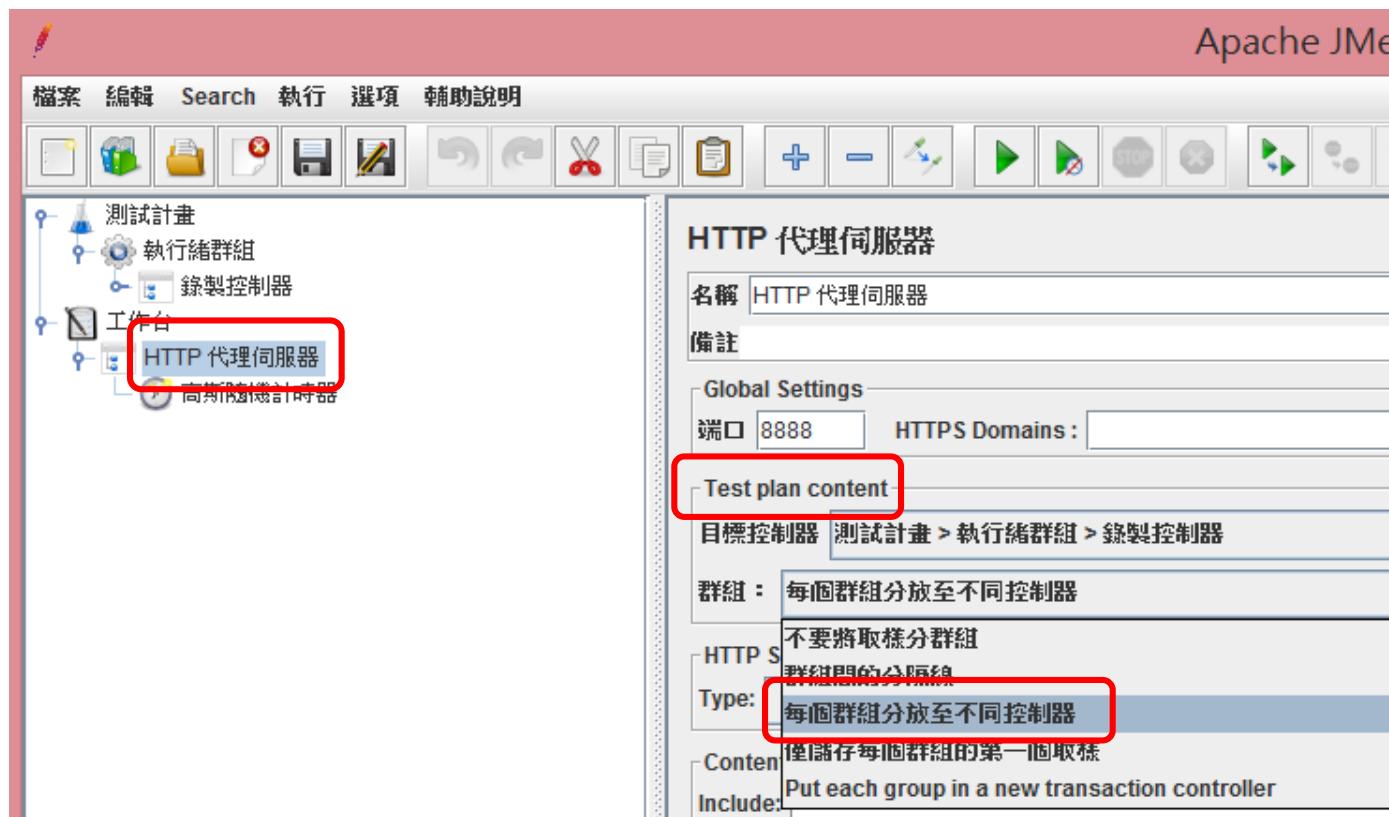
- 修改 – 定值延遲差 \${T}
 - 記錄真實操作時間延遲



Lab. 4 設定高斯隨機計時器

□ HTTP代理伺服器 - 修改 - Test plan content

- 目標控制器 – 測試計畫 > 執行緒群組 > 錄製控制器
- 群組 – 每個群組分放至不同控制器

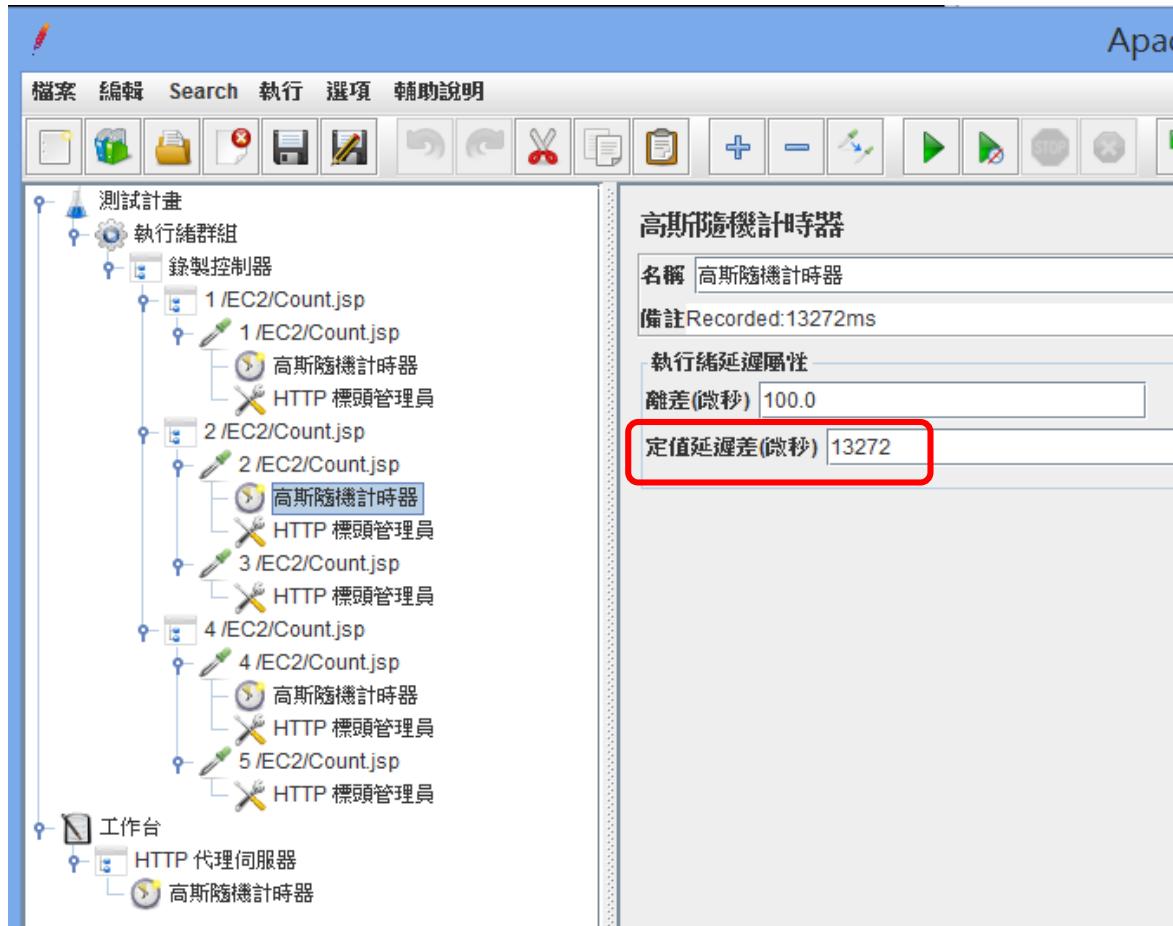


Lab. 4 設定高斯隨機計時器

- 瀏覽器-設定-變更Proxy – LAN 設定- Proxy伺服器 – 127.0.0.1
- HTTP代理伺服 - 啟動
- 使用瀏覽器瀏覽(<http://192.168.0.100:8080/Count.jsp>)
- HTTP 代理伺服器 - 停止
- 取消 – 瀏覽器 Proxy
- 錄製完成，檢查各個http請求，是否有空白頁，或無關頁。
 - 若有，刪除。否則可能導致錯誤。
 - 手動刪除第一個取樣器中自動產生的計時器。

Lab. 4 設定高斯隨機計時器

- 錄製控制器真實記錄使用者動作延遲時間



Lab. 4 設定高斯隨機計時器

- 執行緒群組 - 新增 - 接聽 - Aggregate Graph
- 執行緒群組 - 新增 - 接聽 - 檢視表格式結果
- 執行緒群組，執行測試

Label	取樣數	平均值	中間值	90% Line	95% Line	99% Line	最小值	最大值	錯誤率	處理量	每秒仟位元組
1 /EC2/Count....	1	11	11	11	11	11	11	11	0.00%	90.9/sec	60.5
2 /EC2/Count....	1	5	5	5	5	5	5	5	0.00%	200.0/sec	102.5
3 /EC2/Count....	1	3	3	3	3	3	3	3	0.00%	333.3/sec	221.7
4 /EC2/Count....	1	13	13	13	13	13	13	13	0.00%	76.9/sec	39.6
5 /EC2/Count....	1	6	6	6	6	6	6	6	0.00%	166.7/sec	110.8

取樣編號 #	Start Time	Thread Name	Label	取樣時間(微秒)	Status	位元組	Latency	Connect Time(ms)
1	06:51:17.002	執行緒群組 1-1	1 /EC2/Count.jsp	11	✓	681	11	2
2	06:51:30.114	執行緒群組 1-1	2 /EC2/Count.jsp	5	✓	525	5	1
3	06:51:30.119	執行緒群組 1-1	3 /EC2/Count.jsp	3	✓	681	3	0
4	06:51:37.772	執行緒群組 1-1	4 /EC2/Count.jsp	13	✓	527	12	2
5	06:51:37.785	執行緒群組 1-1	5 /EC2/Count.jsp	6	✓	681	6	0

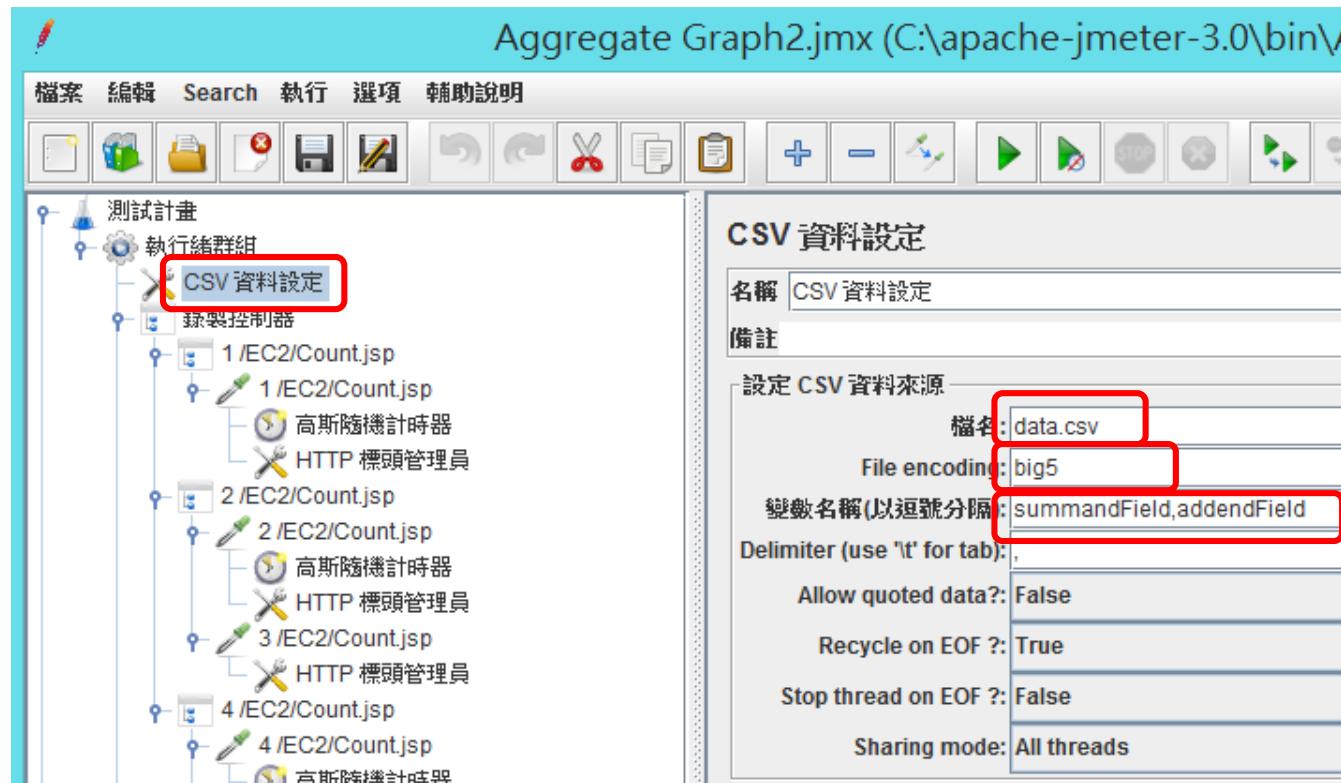
Lab. 5 設定測試資料檔案

- 執行緒群組 - 新增 - 設定元素 - CSV 資料設定



Lab. 5 設定測試資料檔案

- CSV 資料設定
 - 檔名: data.csv 放在 Jmeter_HOME/bin/ 下面
 - 變數名稱: 自訂兩個變數名稱 summandField,addendField



CSV 資料設定

名稱: CSV 資料設定

備註

設定 CSV 資料來源

檔名: (highlighted with a red box)

File encoding: (highlighted with a red box)

變數名稱 (以逗號分隔): (highlighted with a red box)

Delimiter (use 't' for tab):

Allow quoted data?: False

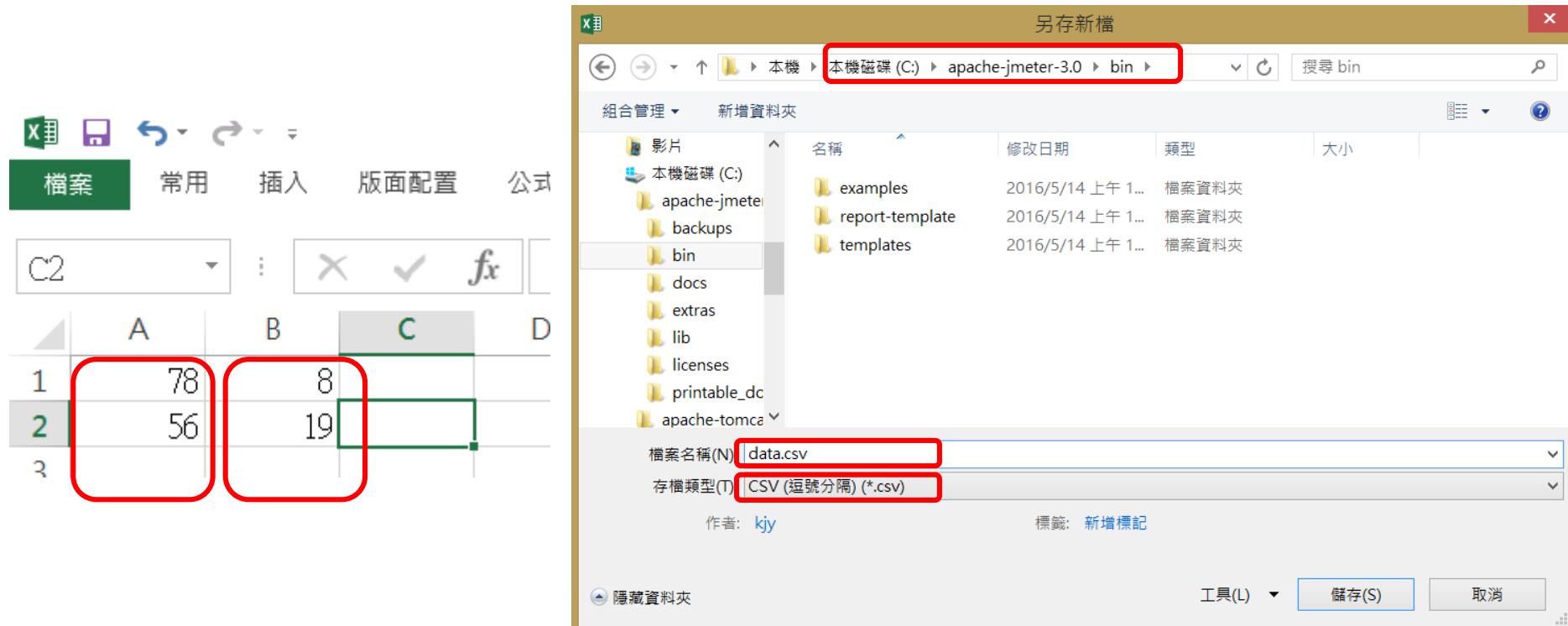
Recycle on EOF ?: True

Stop thread on EOF ?: False

Sharing mode: All threads

Lab. 5 設定測試資料檔案

- 儲存成data.CSV
 - 第一、二欄分別是summandField,addendField



Lab. 5 設定測試資料檔案

- 2 /Count.jsp、4 /Count.jsp
 - summandField 值填入 \${summandField}，注意不可有空白鍵
 - addendField的值填入 \${addendField}

Aggregate Graph2.jmx (C:\apache-jmeter-3.0\bin\Aggregate Graph2.jmx) - Apache JMeter (3.0)

檔案 編輯 Search 執行 選項 輔助說明

測試計畫

執行緒群組

CSV 資料設定

錄製控制器

1 /EC2/Count.jsp

1 /EC2/Count.jsp

高斯隨機計時器

HTTP 標頭管理員

2 /EC2/Count.jsp

2 /EC2/Count.jsp

高斯隨機計時器

HTTP 標頭管理員

3 /EC2/Count.jsp

3 /EC2/Count.jsp

HTTP 標頭管理員

4 /EC2/Count.jsp

4 /EC2/Count.jsp

4 /EC2/Count.jsp

5 /EC2/Count.jsp

HTTP 標頭管理員

Aggregate Graph

檢視表格式結果

工作台

HTTP 代理伺服器

HTTP 要求

名稱: 2 /EC2/Count.jsp

備註

Basic Advanced

Web 伺服器

主機名稱或 IP: 192.168.0.100 端口號碼: 8080

HTTP 要求

Implementation: [dropdown] 協定: http 方法: POST Content encoding: big5

路徑: /EC2/Count.jsp

自動跟蹤重導: 跟蹤重導: Use KeepAlive: Use multipart/form-data for POST: Browser-compatible headers:

Parameters Body Data Files Upload

送出含參數的請求

名稱	值
summandField	\${summandField}
addendField	\${addendField}
submit	Submit

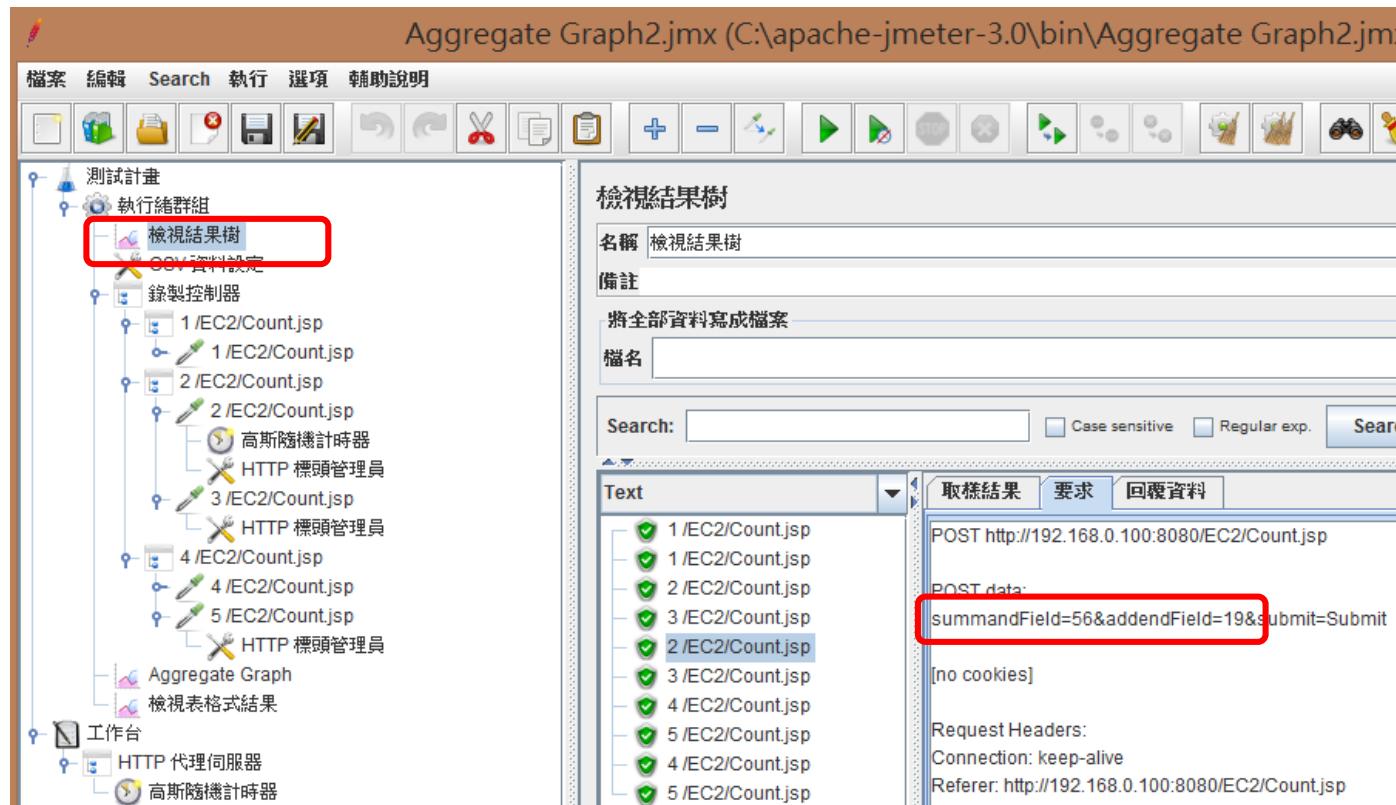
Lab. 5 設定測試資料檔案

- 執行緒群組 - 執行緒屬性 - 執行緒數量 - 2
- 執行緒群組 - 新增 - 接聽 - 檢視結果樹
- 執行 - 開始



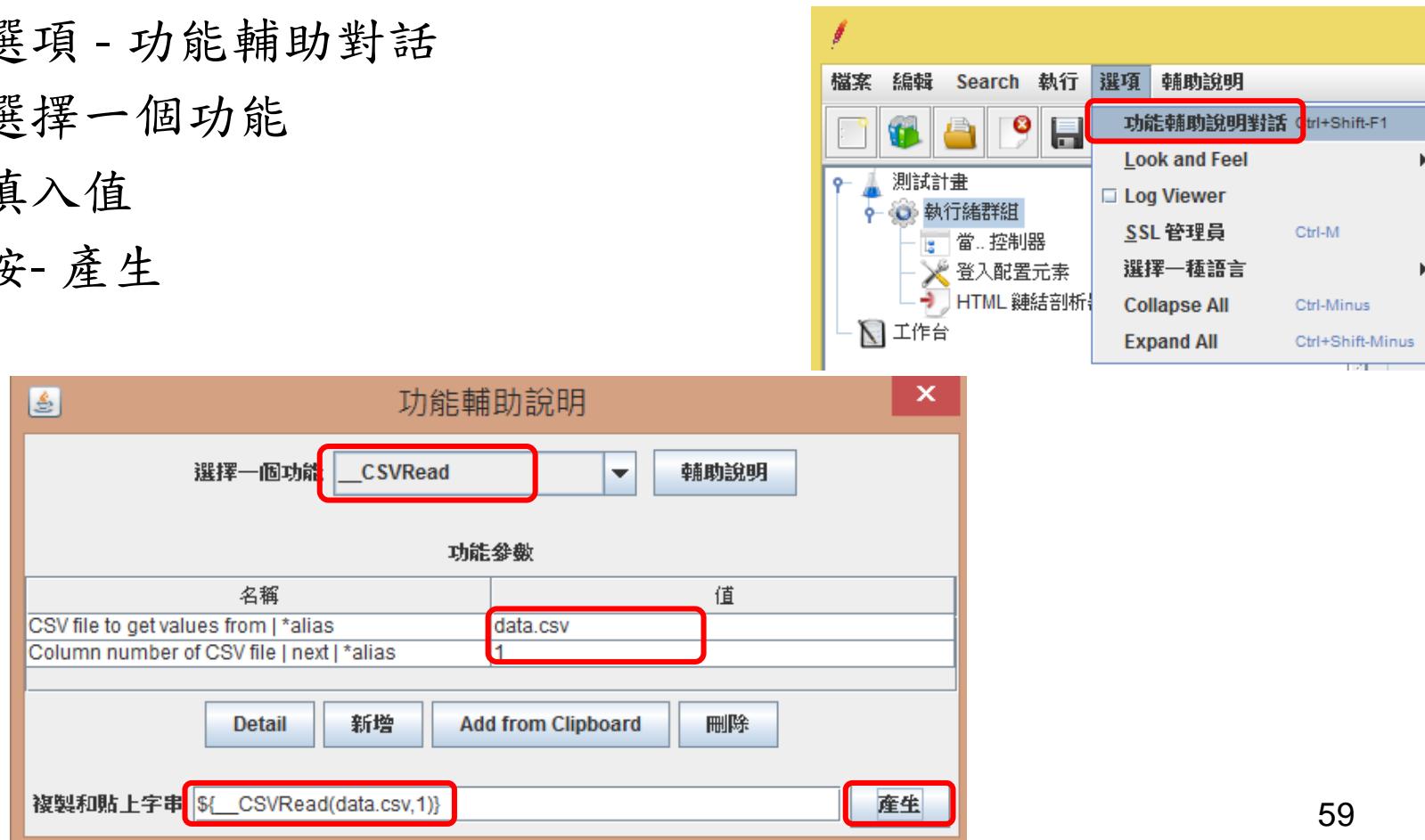
Lab. 5 設定測試資料檔案

- 觀察 - 檢視結果樹-要求-2 /Count.jsp 、4 /Count.jsp
 - summandField=78&addendField=8 ,
summandField=56&addendField=19



Lab. 5 設定測試資料檔案

- 使用功能(function)讀取檔案測試資料
 - 選項 - 功能輔助對話
 - 選擇一個功能
 - 填入值
 - 按- 產生



Lab. 5 設定測試資料檔案

- 2 /Count.jsp、4 /Count.jsp
 - summandField 值填入 \${__CSVRead(data.csv,0)}，兩個底線
 - addendField值\${__CSVRead(data.csv,1)}\${__CSVRead(data.csv,next)}

Aggregate Graph2.jmx (C:\apache-jmeter-3.0\bin\Aggregate Graph2.jmx) - Apache JMeter (3.0)

檔案 編輯 Search 執行 選項 輔助說明

測試計畫

- 執行緒群組
 - CSV 資料設定
 - 錄製控制器
 - 1 /EC2/Count.jsp
 - 1 /EC2/Count.jsp
 - 高斯隨機計時器
 - HTTP 標頭管理員
 - 2 /EC2/Count.jsp
 - 2 /EC2/Count.jsp
 - 高斯隨機計時器
 - HTTP 標頭管理員
 - 3 /EC2/Count.jsp
 - HTTP 標頭管理員
 - 4 /EC2/Count.jsp
 - 4 /EC2/Count.jsp
 - 5 /EC2/Count.jsp
 - HTTP 標頭管理員
 - Aggregate Graph
 - 檢視表格式結果

工作台

HTTP 代理伺服器

HTTP 要求

名稱: 2 /EC2/Count.jsp

備註:

Basic Advanced

Web 伺服器

主機名稱或 IP: 192.168.0.100 端口號碼: 8080

HTTP 要求

Implementation: 協定: http 方法: POST Content encoding: big5

路徑: /EC2/Count.jsp

自動跟蹤重導: 跟蹤重導: Use KeepAlive: Use multipart/form-data for POST: Browser-compatible headers:

Parameters Body Data Files Upload

送出含參數的請求

名稱	值
summandField	\${summandField}
addendField	\${addendField}
submit	Submit

60

Lab. 5 設定測試資料檔案

- 觀察 - 檢視結果樹-要求-2 /Count.jsp 、4 /Count.jsp

Aggregate Graph2.jmx (C:\apache-jmeter-3.0\bin\Aggregate Graph2.jmx)

檔案 編輯 Search 執行 選項 幫助說明

測試計畫
執行緒群組
檢視結果樹
CSV 資料設定
錄製控制器
1 /EC2/Count.jsp
2 /EC2/Count.jsp
3 /EC2/Count.jsp
HTTP 標頭管理員
4 /EC2/Count.jsp
5 /EC2/Count.jsp
HTTP 標頭管理員
Aggregate Graph
檢視表格式結果
工作台
HTTP 代理伺服器
高斯隨機計時器

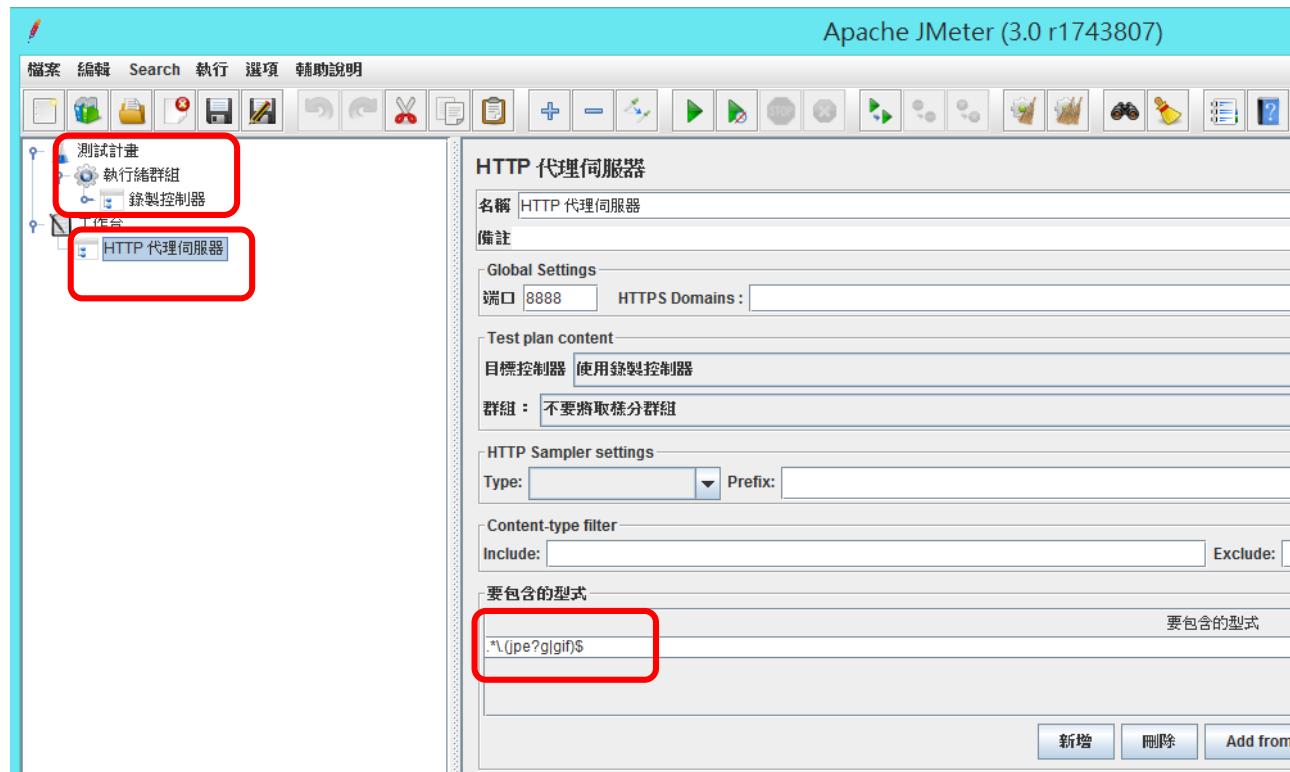
檢視結果樹
名稱 檢視結果樹
備註
將全部資料寫成檔案
檔名
Search: Case sensitive Regular exp. Search

Text
1 /EC2/Count.jsp
2 /EC2/Count.jsp
3 /EC2/Count.jsp
4 /EC2/Count.jsp
5 /EC2/Count.jsp

取樣結果 要求 回覆資料
POST http://192.168.0.100:8080/EC2/Count.jsp
POST data:
summandField=78&addendField=8&submit=Submit
[no cookies]
Request Headers:
Connection: keep-alive

Lab. 6 篩選檔案

- 測試計畫 - 新增 - 執行緒群組 - 新增 - 錄製控制器
- 工作台 - 新增 - 非測試元素 - HTTP代理伺服器
 - HTTP代理伺服器要包含的型式欄位填入「.*\.(jpe?g|gif)\$」



Lab. 6 篩選檔案

- 瀏覽器-設定-變更Proxy – LAN 設定- Proxy伺服器 – 127.0.0.1
- HTTP代理伺服 - 啟動
- 使用瀏覽器瀏覽網頁(如www.ntu.edu.tw)
 - HTTP 代理伺服器將會錄製網頁所操作的Http要求內容，要包含的形式(*.jpg)，到錄製控制器
- 停止 - HTTP 代理伺服器
- 取消 – 瀏覽器 Proxy
- 執行緒群組，執行測試
- 在HTTP代理伺服器中要包含的型式欄位填入
 - ".*\.(jpe?g|gif)\$"

Lab. 6 篩選檔案

- 篩選出.jpg 和.gif 的檔案

Apache JMeter (3.0 r1743807)

檔案 編輯 Search 執行 選項 輔助說明

The screenshot shows the Apache JMeter interface. On the left, under the '錄製控制器' (Recording Controller) node, there is a list of files. A red box highlights the first file in the list: '123 /images/banner/inner_banner3.jpg'. To the right, the 'HTTP 要求' (HTTP Request) configuration panel is open. It displays the following details:

- 名稱:** 123 /images/banner/inner_banner3.jpg
- 備註:** (empty)
- Basic** tab is selected.
- Web 伺服器:**
 - 主機名稱或 IP:** www.ntu.edu.tw
- HTTP 要求:**
 - Implementation:** (dropdown menu)
 - 協定:** http
 - 方法:** GET
 - Content encoding:** (dropdown menu)
- 路徑:** /images/banner/inner_banner3.jpg
- 選項:**
 - 自動跟蹤重導
 - 跟蹤重導
 - Use KeepAlive
 - Use multipart/form-data for POST
 - Browser-compatible headers
- Parameters** tab is selected.

正規表示式

- name="sesskey"\stype="hidden"\svalue="([^\"]+)"
- sesskey=([^\"]+)"
- 正規表示式文法

表示式	意義
()	找()中匹配的值，回傳。
[]	可選
+	串接，出現一次或更
\s	空白
.	配任何字元
?	停止在第一個匹配成功時
^	後面的字元會被忽略

正規表示式

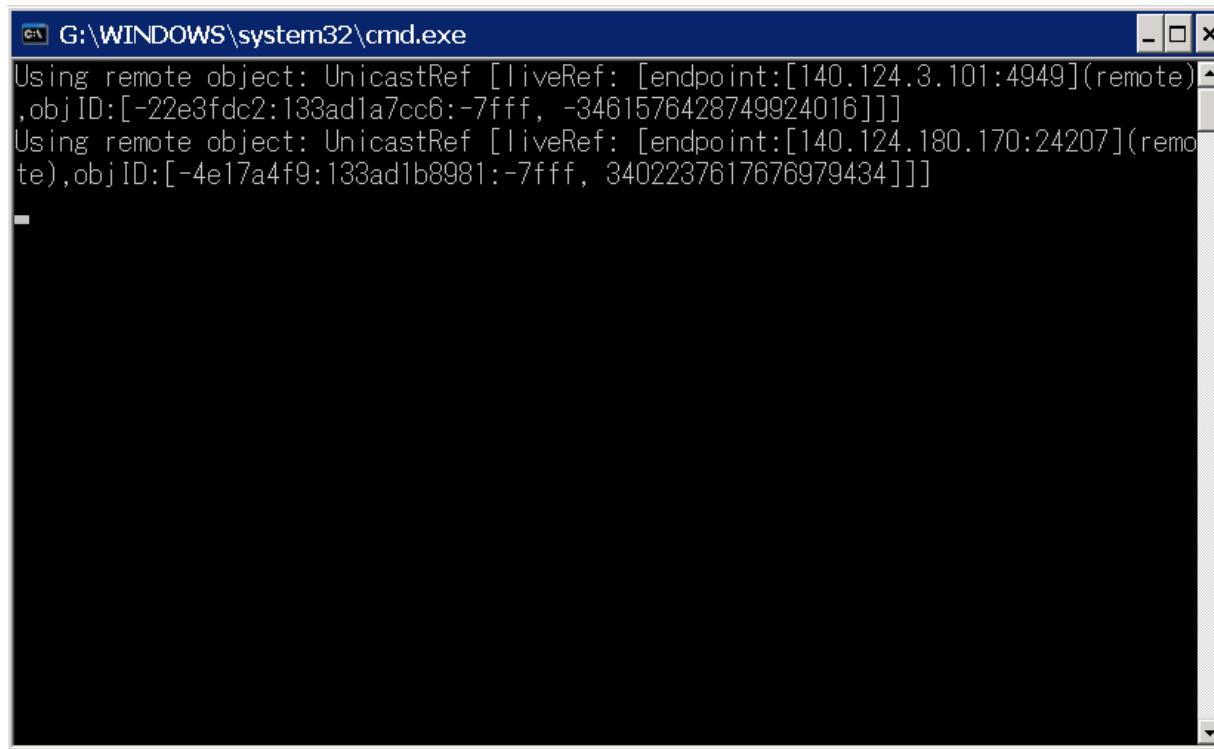
- 若.html內涵以下字串，要回傳readme.txt
name="file" value="readme.txt"
- 表示式
 - **name="file" value="(.)?">**

Lab. 7 遠端控制測試

- 目的
 - 設定各個伺服器電腦(不同客戶端使用相同版本Java, JMeter)，則測試使用的資料檔案，不需要送到不同的客戶端

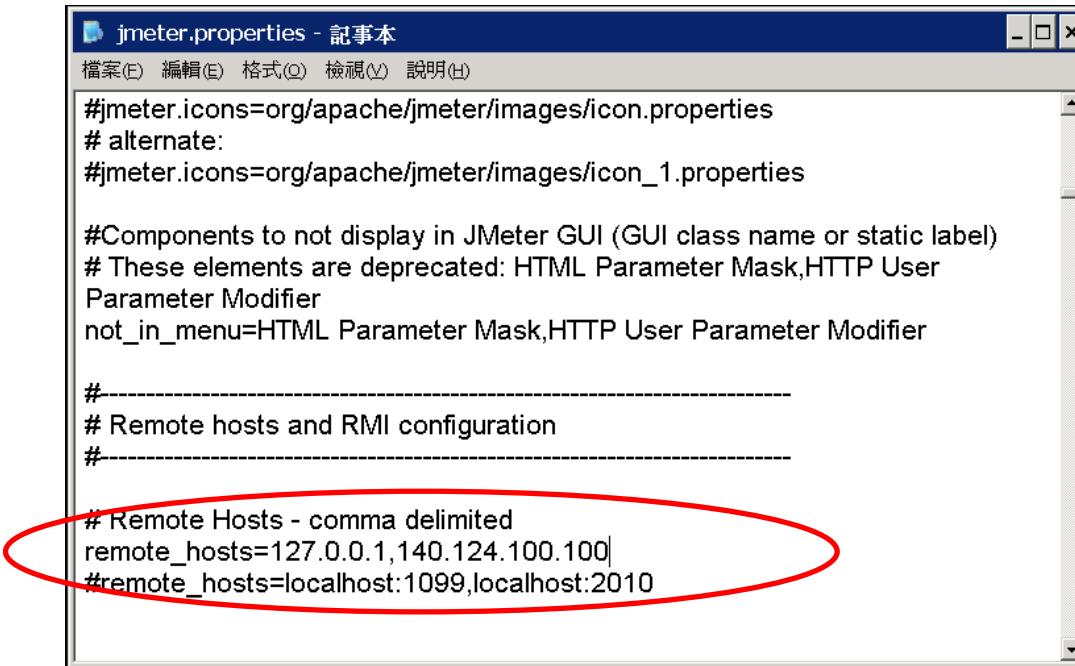
Lab. 7 遠端控制測試

- 啟動受控端伺服器
 - 執行JMETER_HOME/bin/jmeter-server.bat啟動所有JMeter受控端伺服器電腦(遠端控制使用Java RMI溝通)。



Lab. 7 遠端控制測試

- 主控端加入受控端伺服器IP
 - 開啟JMETER_HOME/bin/jmeter.properties檔案，開啟remote_hosts，加入要遠端受控的JMeter 伺服器IP，多台IP使用逗號隔開，例如127.0.0.1,140.124.100.100。
 - IP請查詢電腦環境自行設定



```
jmeter.properties - 記事本
檔案(E) 編輯(E) 格式(O) 檢視(V) 說明(H)

#jmeter.icons=org/apache/jmeter/images/icon.properties
# alternate:
#jmeter.icons=org/apache/jmeter/images/icon_1.properties

#Components to not display in JMeter GUI (GUI class name or static label)
# These elements are deprecated: HTML Parameter Mask,HTTP User
Parameter Modifier
not_in_menu=HTML Parameter Mask,HTTP User Parameter Modifier

#
#-----#
# Remote hosts and RMI configuration
#-----#

# Remote Hosts - comma delimited
remote_hosts=127.0.0.1,140.124.100.100
#remote_hosts=localhost:1099,localhost:2010
```

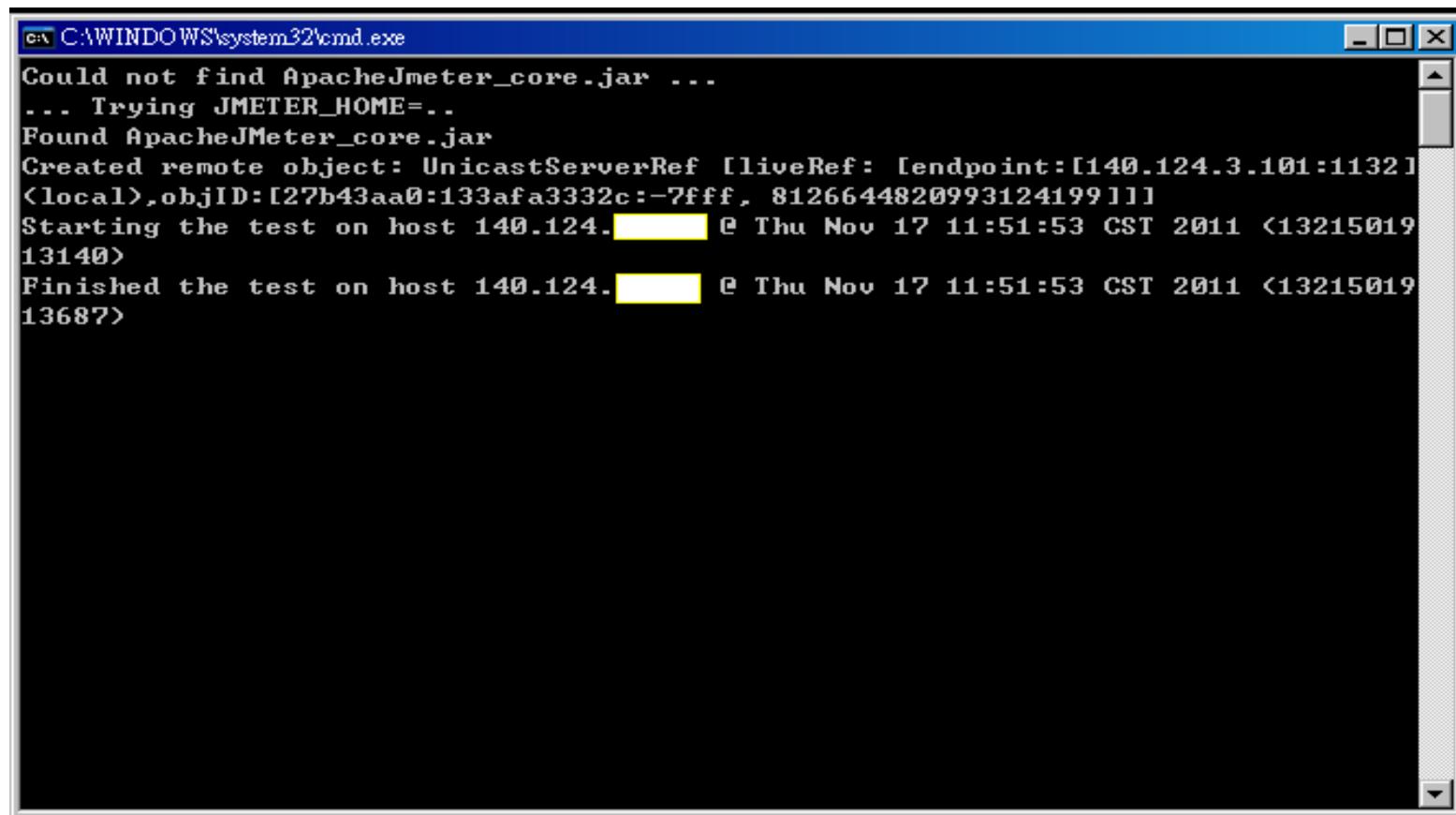
Lab. 7 遠端控制測試

- 啟動受控端電腦
 - 從主控端JMeter 介面啟動遠端電腦



Lab. 7 遠端控制測試

- 受控端電腦執行測試
 - 受控端會出現執行執行開始與結束訊息



```
c:\> C:\WINDOWS\system32\cmd.exe
Could not find ApacheJmeter_core.jar ...
... Trying JMETER_HOME=..
Found ApacheJMeter_core.jar
Created remote object: UnicastServerRef [liveRef: [endpoint:[140.124.3.101:11321
<local>,objID:[27b43aa0:133afa3332c:-7fff, 8126644820993124199]]]
Starting the test on host 140.124. [REDACTED] @ Thu Nov 17 11:51:53 CST 2011 <13215019
13140>
Finished the test on host 140.124. [REDACTED] @ Thu Nov 17 11:51:53 CST 2011 <13215019
13687>
```

Exercise 遠端遙控多台電腦壓測

- 10 分鐘
 - 錄製壓力測試劇本
 - 兩人一組，輪流當主控端與受控端電腦
 - 執行測試，觀察測試報表
 - 三人一組，輪流當主控端與受控端電腦
 - 執行測試，觀察測試報表

Lab. 8 Redmine效能測試與伺服器監控

- 錄製Redmine，admin/admin登入，選擇專案清單、網站管理
- 設定執行緒數量為10、100、1000、10000
- 觀察Aggregate Graph、檢視結果樹的數據
 - redmine/login出現紅色錯誤100%，為csrf-token錯誤，不影響登入
- 觀察Glances伺服器效能數據
- 觀察Conky伺服器效能數據

Lab. 8 Redmine效能測試與伺服器監控

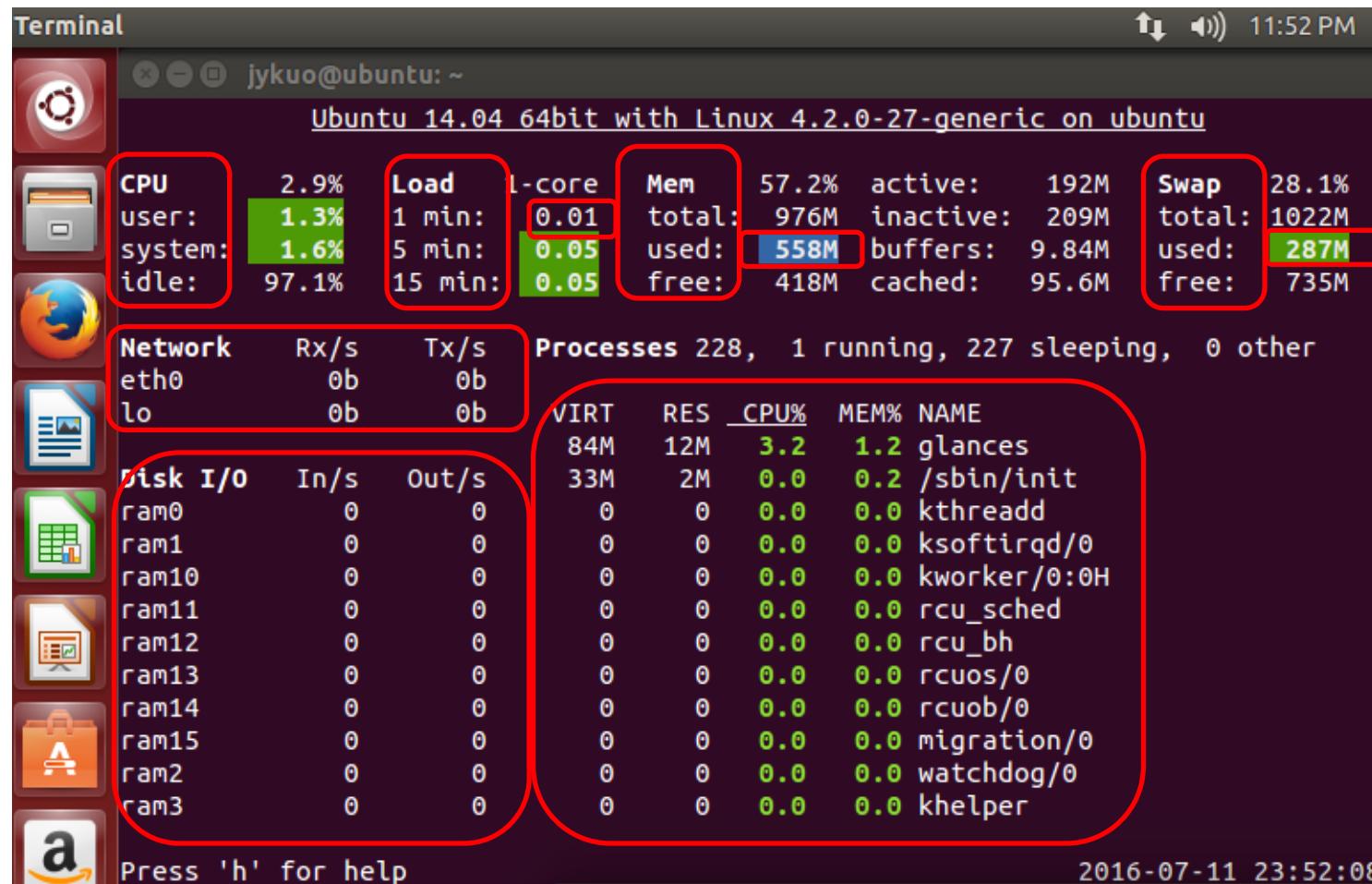
- 安裝Ubuntu 14.04 伺服器監效能控程式
 - Glances
 - Sudo apt-get install Glances
 - glances
 - Conky
 - sudo apt-get install conky-all
 - conky&

Lab. 8 Redmine效能測試與伺服器監控

- Glances
 - CPU 資訊（使用者的相關應用，系統核心程式和空閒程式）
 - 總記憶體資訊，包括實體記憶體，交換空間和空閒記憶體等等
 - 前 1 分鐘、5 分鐘和 15 分鐘平均的 CPU 負載
 - 網路連結的下行和上行速度
 - 處理器總數，及其活動狀態
 - 硬碟 I/O 相關（讀寫）速度詳情
 - 當前掛載設備的HD使用情況
 - 高 CPU 和記憶體使用的process，和相關應用的位置
 - 底部顯示當前日期和時間
 - 將消耗最高系統資源的process用紅色標出

Lab. 8 Redmine效能測試與伺服器監控

- Glances



Lab. 8 Redmine效能測試與伺服器監控

- Glances 燈號
 - 綠色：OK
 - 藍色：CAREFUL(50)
 - 紫色：WARNING(70)
 - 紅色：CRITICAL(90)
 - 設定檔/etc/glances/glances.conf
- Process 記憶體與CPU使用
 - VIRT 表示process目前真正在存取使用的記憶體虛擬大小
 - RES 表示process目前真正使用的實體記憶體大小

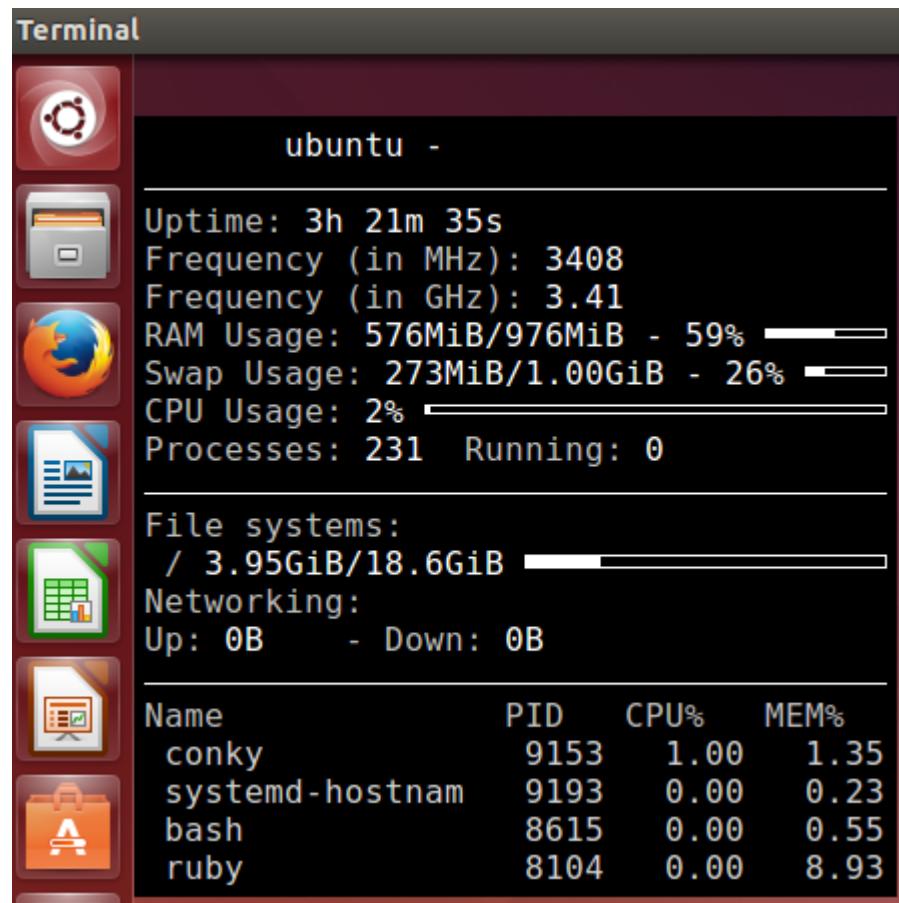
Lab. 8 Redmine效能測試與伺服器監控

- Glances

- 「a」：自動排序模式，這會讓行程列表自動排序。
- 「b」：切換網路 I/O 的單位，bit/s 或 Byte/s。
- 「c」：依據 CPU 使用率排序行程。
- 「d」：顯示/隱藏硬碟 I/O 狀態。
- 「f」：顯示/隱藏檔案系統狀態。
- 「h」：顯示/隱藏輔助 (help) 訊息狀態。
- 「i」：依據 I/O 排序行程。
- 「l」：顯示/隱藏記錄檔 (log) 訊息。
- 「m」：依據記憶體使用率排序行程。
- 「n」：顯示/隱藏網路狀態。
- 「p」：依據名稱排序行程。
- 「s」：顯示/隱藏 sensors 狀態。
- 「w」：刪除已完成的警告訊息。
- 「x」：刪除已完成的警告與錯誤訊息。
- 「1」：在全部 CPU 與單一核心 (core) 之間做切換。
- 「q」：離開。

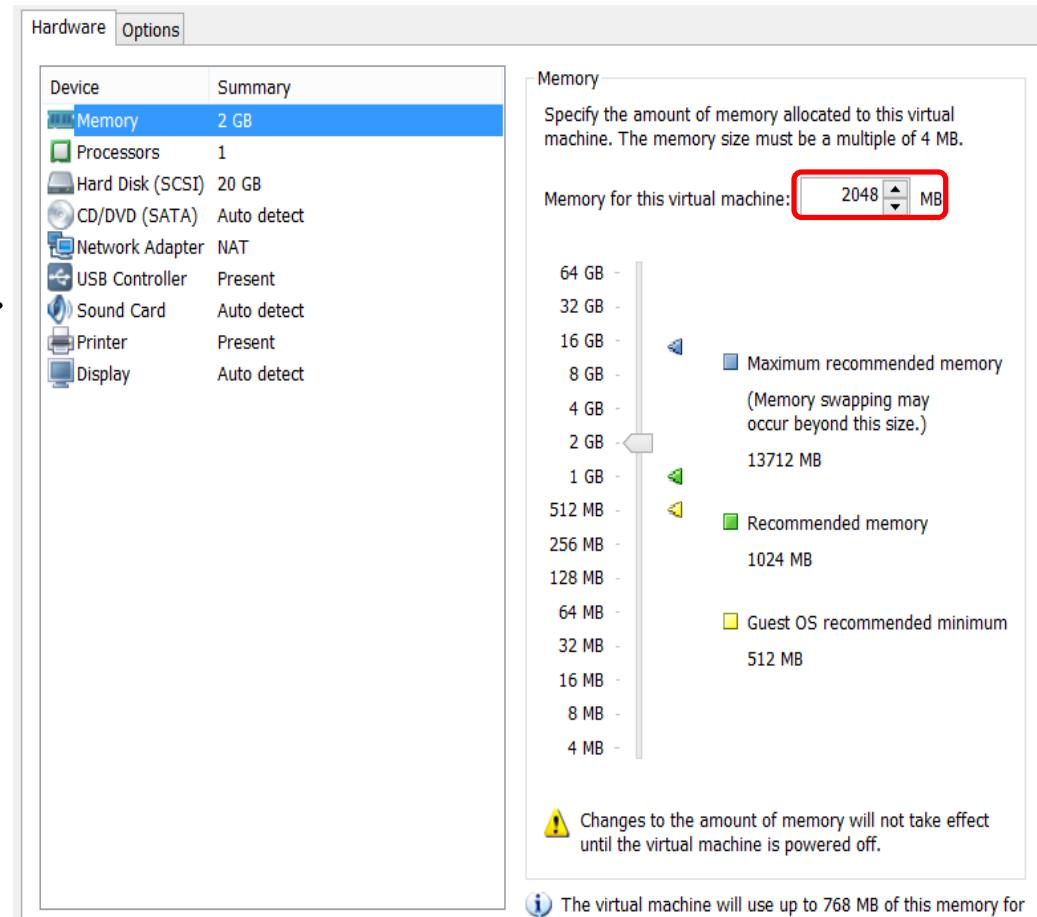
Lab. 8 Redmine效能測試與伺服器監控

- conky



Lab. 8 Redmine效能測試與伺服器監控

- 重設虛擬機器配置
 - Player
 - Manage
 - Virtual Machine Settings..
 - 重設記憶體大小2G



Lab. 8 Redmine效能測試與伺服器監控

- 重新啟動VM Ubuntu，執行 Glances，記憶體變成1.94G

Terminal

jykuo@ubuntu: ~

Ubuntu 14.04 64bit with Linux 4.2.0-27-generic on ubuntu

CPU	5.6%	Load	1-core	Mem	36.5%	active:	643M	Swap	0.0%
user:	2.0%	1 min:	0.06	total:	1.94G	inactive:	277M	total:	1022M
system:	3.6%	5 min:	0.07	used:	724M	buffers:	50.7M	used:	0
idle:	94.4%	15 min:	0.05	free:	1.23G	cached:	400M	free:	1022M

Network	Rx/s	Tx/s	Processes	234,	1 running,	233 sleeping,	0 other
eth0	0b	0b	VIRT	84M	RES	CPU%	MEM% NAME
lo	0b	0b		19M	3.3	0.9	glances

Disk I/O	In/s	Out/s	365M	10M	1.3	0.5	pulseaudio
ram0	0	0	276M	46M	0.3	2.3	Xorg
ram1	0	0	1.1G	102M	0.3	5.1	compiz
ram10	0	0	33M	4M	0.0	0.2	/sbin/init
ram11	0	0	0	0	0.0	0.0	kthreadd
ram12	0	0	0	0	0.0	0.0	ksoftirqd/0
ram13	0	0	0	0	0.0	0.0	kworker/0:0H
ram14	0	0	0	0	0.0	0.0	kworker/u256:0
ram15	0	0	0	0	0.0	0.0	rcu_sched
ram2	0	0	0	0	0.0	0.0	rcu_bh
ram3	0	0	0	0	0.0	0.0	rcuos/0

Press 'h' for help

2016-07-12 01:03:02

81

Lab. 8 Redmine效能測試與伺服器監控

- 重新設定執行緒數量為10、100、1000、10000
- 觀察Aggregate Graph、檢視結果樹的數據
 - redmine/login出現紅色錯誤100%，為csrf-token錯誤，不影響登入
- 觀察Glances伺服器效能數據，是否有改善
- 觀察Aggregate Graph的效能數據
- 重新設置VM的CPU、RAM的大小配置，重新啟動程式，觀察Glances伺服器效能數據，是否有改善

Exercise 對 Redmine 實施壓力測試流程

- 修改壓力測試計畫書
- 修改測試劇本
- 執行遠端控制多台電腦壓力測試
- 監控虛擬機器伺服器效能
- 分析結果
- 調整虛擬機器
- 執行遠端控制多台電腦壓力測試
- 監控虛擬機器伺服器效能
- 分析結果

Lab. 9 自訂變數-固定值

- 使用http代理伺服器，錄製http://ip:8080/Count.jsp
- 設定元素 – 使用者自訂變數
 - 固定變數的值 - num01 / 65

使用者自訂變數

名稱	使用者自訂變數
備註	

使用者自訂變數

名稱	值	Description
num01	65	

Detail 新增 Add from Clipboard 刪除 Up Down



Lab. 9 自訂變數-固定值

- 變數使用的格式: \${num01}

HTTP 要求

名稱: 3 /Count.jsp
備註:

Basic Advanced

Web 伺服器
主機名稱或 IP: 192.168.211.128 端口號碼: 8080 Timeouts (milliseconds)
Connect: Response:

HTTP 要求
Implementation: 協定: http 方法: POST Content encoding: big5

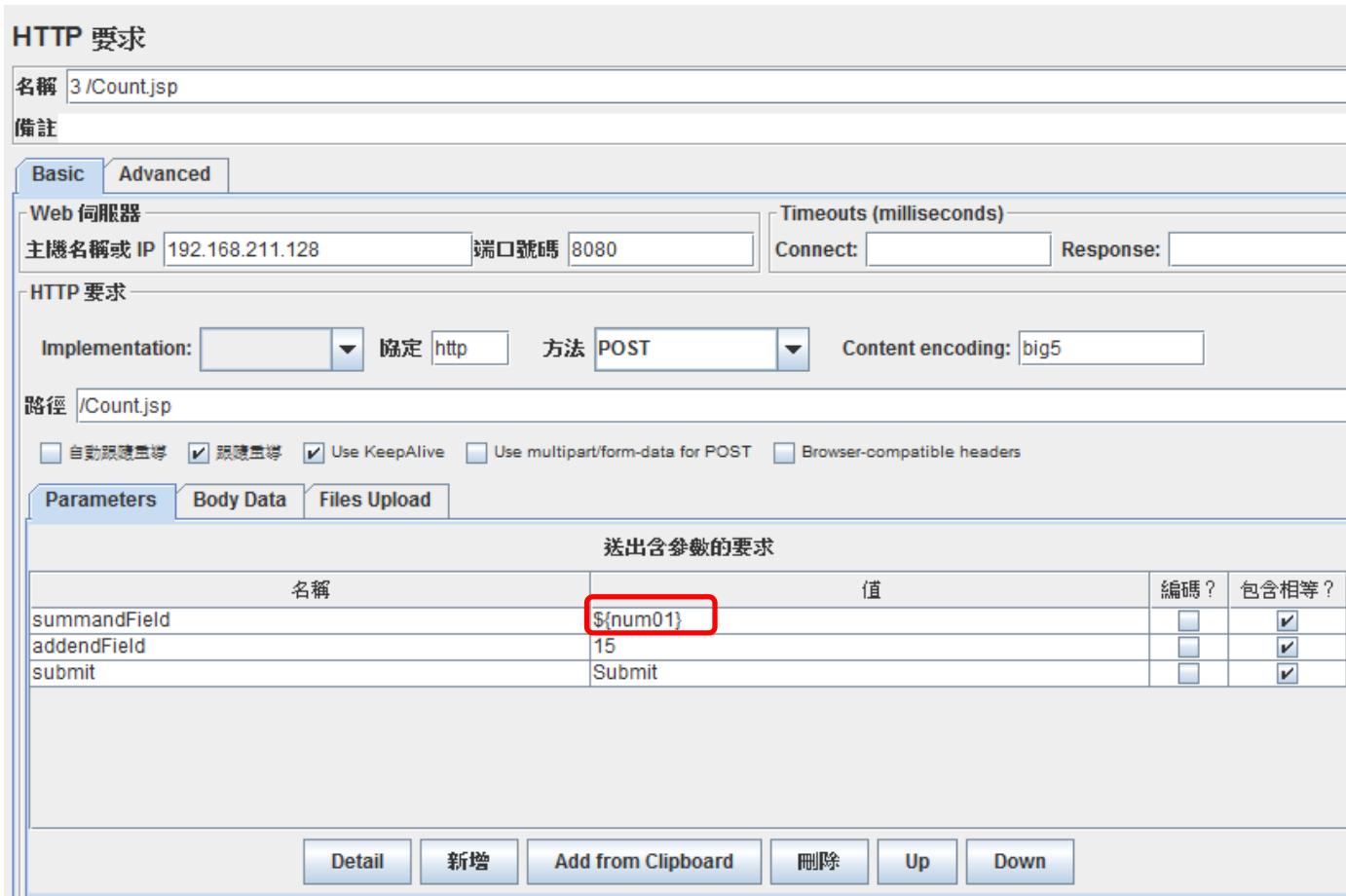
路徑: /Count.jsp
自動跟蹤重導: 跟蹤重導: Use KeepAlive: Use multipart/form-data for POST: Browser-compatible headers:

Parameters Body Data Files Upload

送出含參數的要求

名稱	值	編碼?	包含相等?
summandField	<input type="text" value="\${num01}"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
addendField	<input type="text" value="15"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
submit	<input type="text" value="Submit"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Detail 新增 Add from Clipboard 刪除 Up Down



Lab. 9 自訂變數- Random Variable

- 設定元素 – Random Variable
 - num02 / 1 / 101

Random Variable

名稱	Random Variable
備註	
Output variable	
Variable Name:	num02
Output Format:	
Configure the Random generator	
Minimum Value:	1
Maximum Value:	101
Seed for Random function:	
Options	
Per Thread(User) ?:	<input type="checkbox"/> False

Lab. 9 自訂變數- Random Variable

- 變數使用的格式: \${num02}

Parameters Body Data Files Upload

送出含參數的要求

名稱	值	編碼 ?	包含相等 ?
summandField	\${num01}	<input type="checkbox"/>	<input checked="" type="checkbox"/>
addendField	\${num02}	<input type="checkbox"/>	<input checked="" type="checkbox"/>
submit	Submit	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Detail 新增 Add from Clipboard 刪除 Up Down

Lab. 9 自訂變數- Random Variable

- 新增 – 接聽 – 檢視結果樹
 - addField / num02 會改變(0~101)

The screenshot shows a network traffic analysis interface. On the left, there is a list of recorded requests, mostly '1 /Count.jsp' with one entry '3 /Count.jsp' highlighted. The main pane displays a captured POST request to 'http://192.168.211.128:8080/Count.jsp'. The 'Raw' tab is selected at the bottom. The request details are as follows:

```
POST http://192.168.211.128:8080/Count.jsp

POST data:
summandField=65 addendField=638 submit=Submit

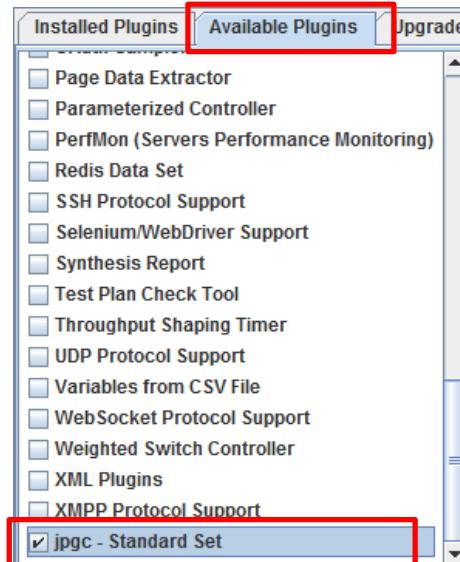
[no cookies]

Request Headers:
Connection: keep-alive
Referer: http://192.168.211.128:8080/Count.jsp
Accept-Language: zh-TW,zh;q=0.8,en-US;q=0.6,en;q=0.4
Origin: http://192.168.211.128:8080
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Upgrade-Insecure-Requests: 1
Content-Type: application/x-www-form-urlencoded
Cache-Control: max-age=0
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/51.0.2704.103 Safari/537.36
Content-Length: 44
```

Two parameters in the POST data, 'summandField=65' and 'addendField=638', are highlighted with red boxes.

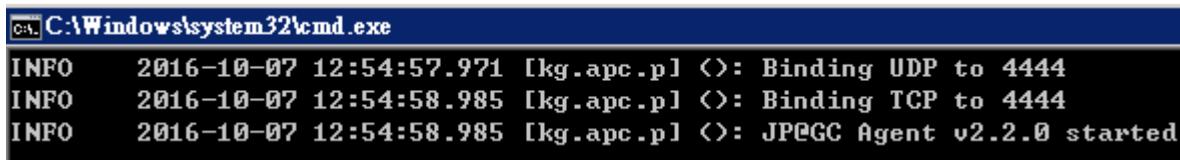
Lab. 10 監控效能PerfMon

- 開啟 <https://jmeter-plugins.org/>
 - 點選 Install，下載 plugins-manager.jar
 - 複製到 jmeter/lib/ext，重新啟動 jmeter
- Plugins Manager
 - 點選 Available Plugins
 - 點選 jpgc – Standard Set
 - 點選 Apply Changes and Restart Jmeter
 - 重新啟動 JMeter



Lab. 10 監控效能PerfMon

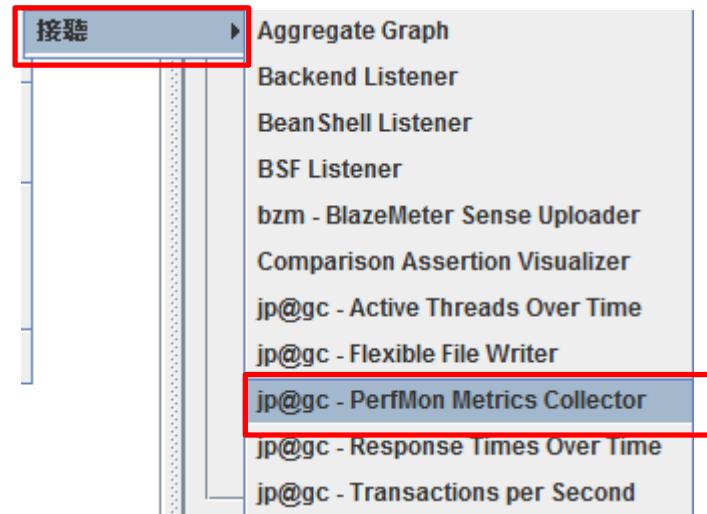
- 啟動 Server Agent
 - 點選 Documentation，下載 Server Agent，
 - ServerAgent-2.2.1.zip
 - 解壓縮，放到受監控的伺服器端，須JRE1.4以上
 - Windows 環境，執行startAgent.bat
 - Unix/Linux 環境，執行 startAgent.sh



```
C:\Windows\system32\cmd.exe
INFO 2016-10-07 12:54:57.971 [kg.apc.p] <>: Binding UDP to 4444
INFO 2016-10-07 12:54:58.985 [kg.apc.p] <>: Binding TCP to 4444
INFO 2016-10-07 12:54:58.985 [kg.apc.p] <>: JPEGC Agent v2.2.0 started
```

Lab. 10 監控效能PerfMon

- JMeter 執行續群組 – 新增 – 接聽 –
 - jp@gc – PerfMon Metrics Collector
- 設定監控參數
 - Host/IP
 - Port 預設 4444
 - Metric to Collect



Servers to Monitor (ServerAgent must be started, see help)		
Host / IP	Port	Metric to collect
140.124.3.101	4444	CPU
140.124.3.101	4444	Memory
140.124.3.101	4444	Network I/O

Lab. 10 監控效能PerfMon

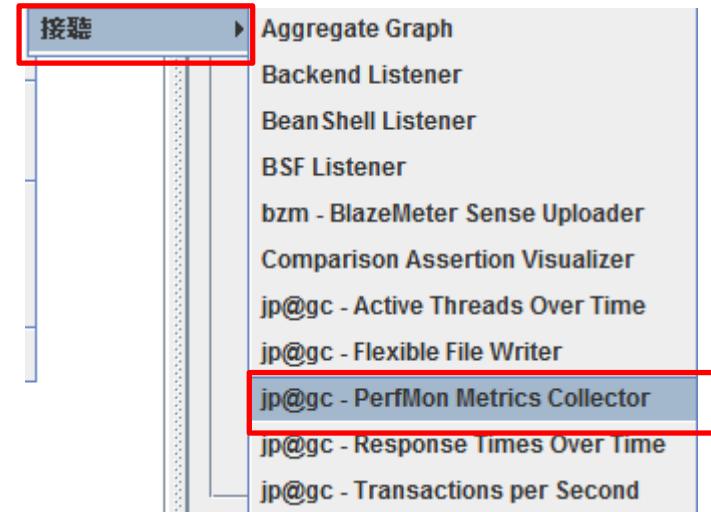
- JMeter 執行續群組 – 新增 – 接聽 –
 - jp@gc – PerfMon Metrics Collector
- 執行續屬性，設定一段時間

執行續屬性

執行續數量 100

啟動延遲(秒) 10

迴圈次數： 永久

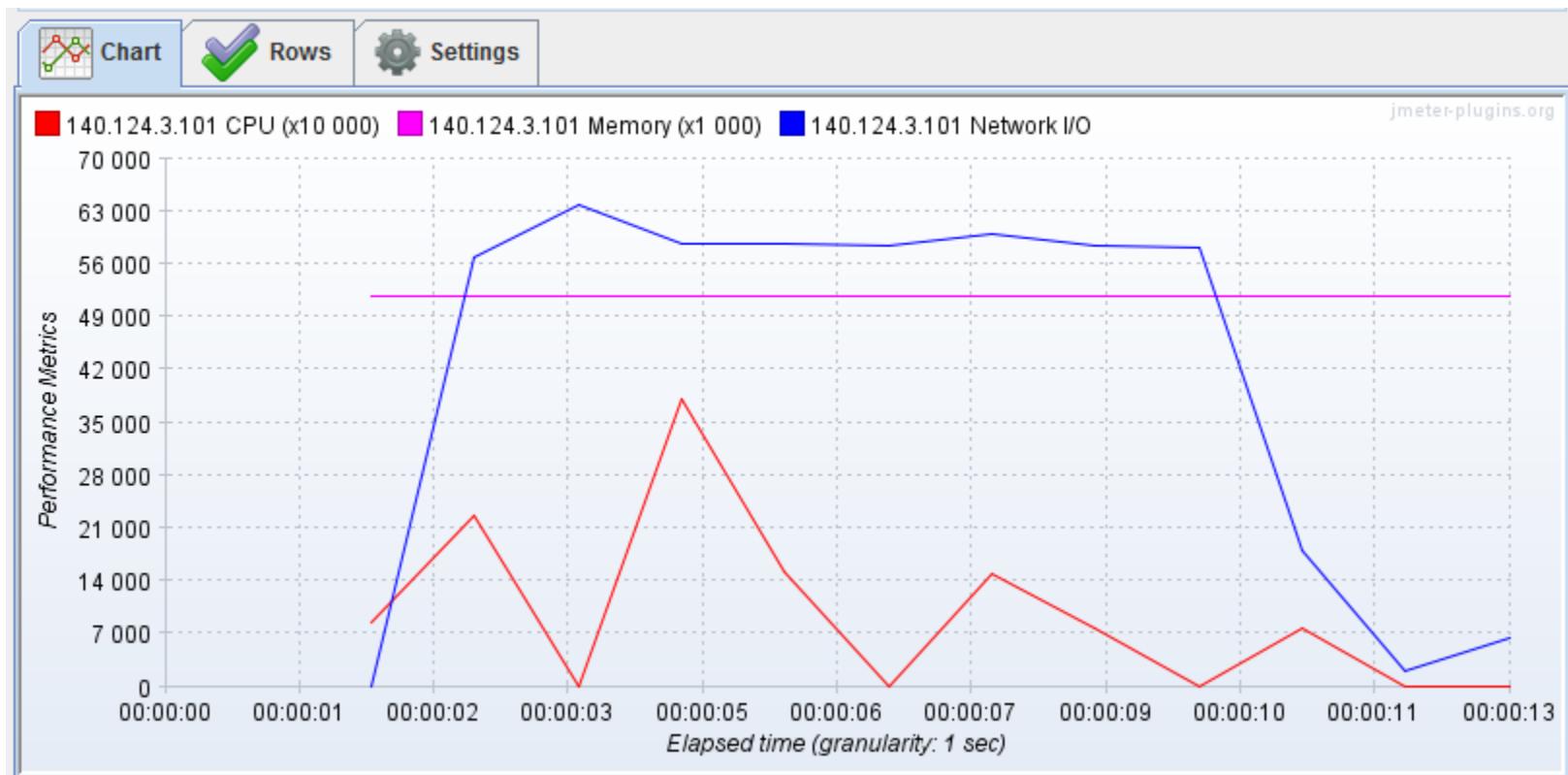


- 點選 Add Row，設定監控參數
 - Host/IP
 - Port 預設 4444
 - Metric to Collect

Servers to Monitor (ServerAgent must be started, see help)		
Host / IP	Port	Metric to collect
140.124.3.101	4444	CPU
140.124.3.101	4444	Memory
140.124.3.101	4444	Network I/O

Lab. 10 監控效能PerfMon

- JMeter 執行



Lab. 11 JDBC Connection MySQL

- 開啟<https://dev.mysql.com/downloads/connector/>
- 點選Connector/J，下載Compressed TAR Archive
- 解壓縮將mysql-connector-java-5.1.42-bin.jar複製到JMeter/lib，重新啟動Jmeter
- 新增JDBC連線設定
 - 執行緒群組 - 新增 - 設定元素 - JDBC連線設定



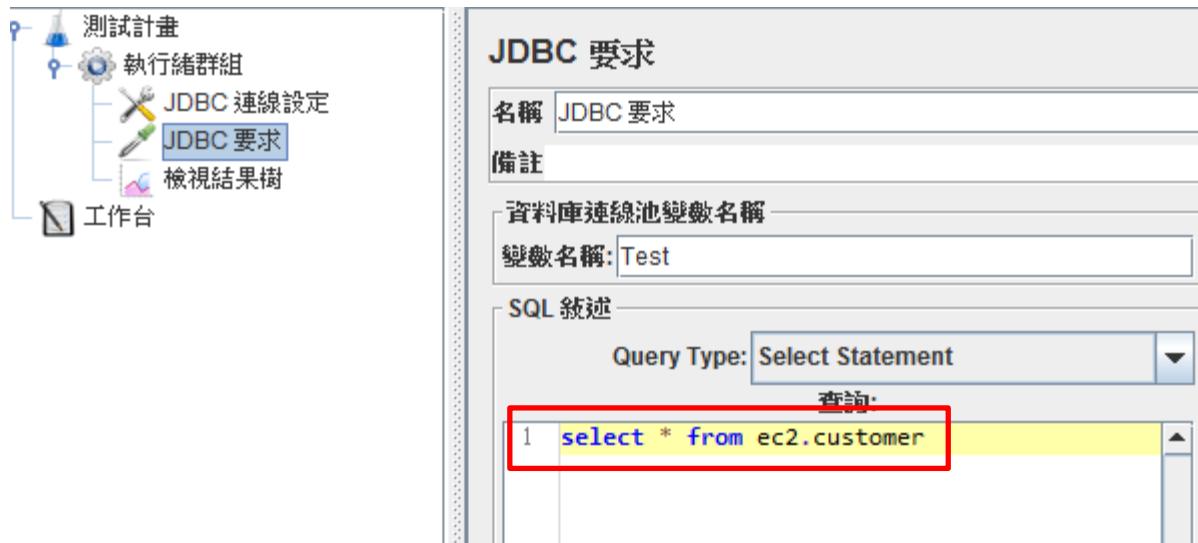
Lab. 11 JDBC Connection MySQL

- 設定JDBC連線設定
 - 設定資料庫連線池變數名稱:Test
 - 設定資料庫連線設定:資料庫URL
 - jdbc:mysql://192.168.244.129:3306/ec2?useUnicode=true&characterEncoding=utf8
 - JDBC驅動程式:
 - com.mysql.jdbc.Driver
 - 使用者: root
 - 密碼: root
- sudo ufw allow in 3306
- mysql -u root -p
- mysql > GRANT ALL PRIVILEGES ON *.* TO root@'%' IDENTIFIED BY 'root' WITH GRANT OPTION;
- mysql > FLUSH PRIVILEGES;



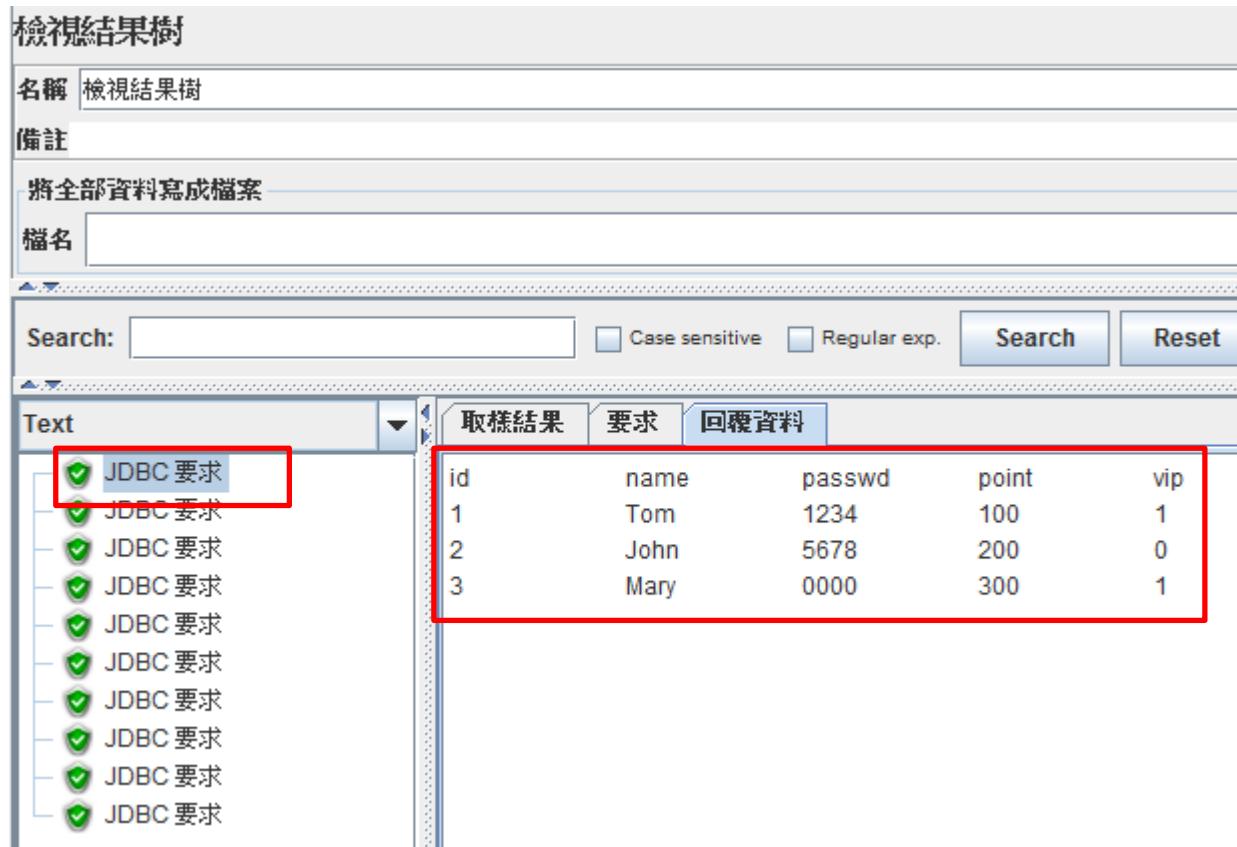
Lab. 11 JDBC Connection MySQL

- 設定JDBC要求
 - JDBC要求資料庫連線池變數名稱須與JDBC連線設定資料庫連線池變數名稱相同: Test
- SQL敘述為測試SQL語句
 - select * from ec2.customer



Lab. 11 JDBC Connection MySQL

- 新增”檢視結果樹”以觀察結果
 - 執行緒群組點右鍵，選擇新增->接聽-> 檢視結果樹
 - 執行 



Exercise 實施 Redmine 效能測試流程

- 撰寫效能測試計畫書
- 錄製測試劇本
- 執行效能測試
- 監控虛擬機器伺服器效能
- 分析結果