**JMeter效能測試入門篇**

**1. Jmeter簡介**

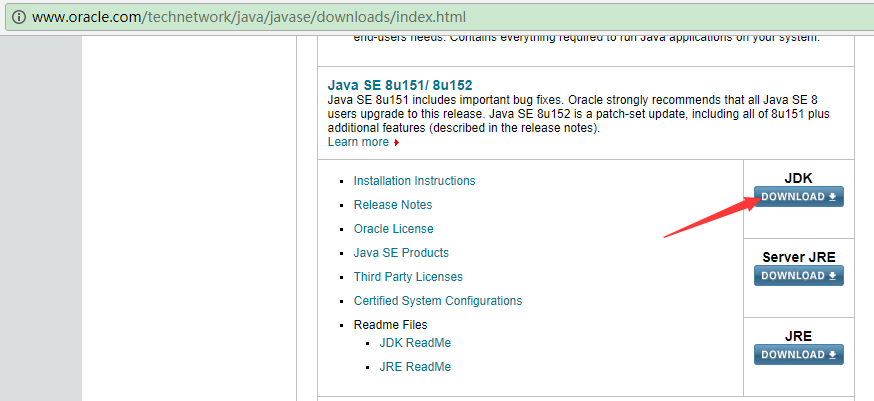
Apache JMeter是一款純java編寫負載功能測試和效能測試開源工具軟體。相比Loadrunner而言，JMeter小巧輕便且免費，逐漸成為了主流的效能測試工具，是每個測試人員都必須要掌握的工具之一。

本文為JMeter效能測試完整入門篇，從Jmeter下載安裝到編寫一個完整效能測試指令碼、最終執行效能測試並分析效能測試結果。

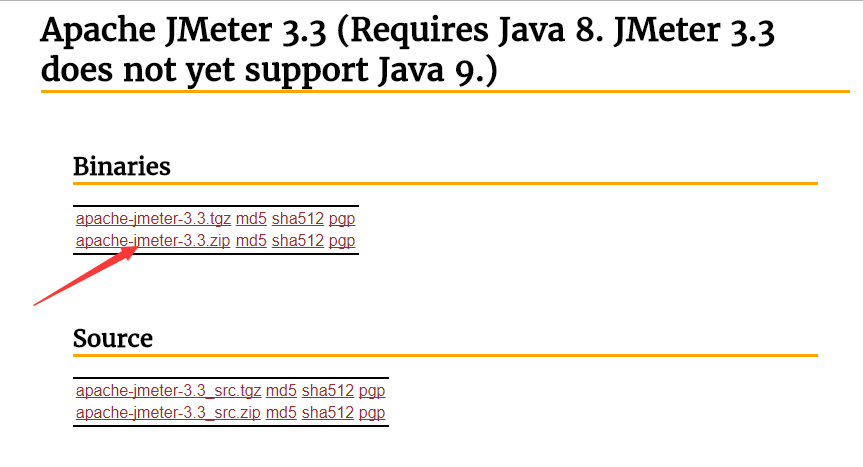
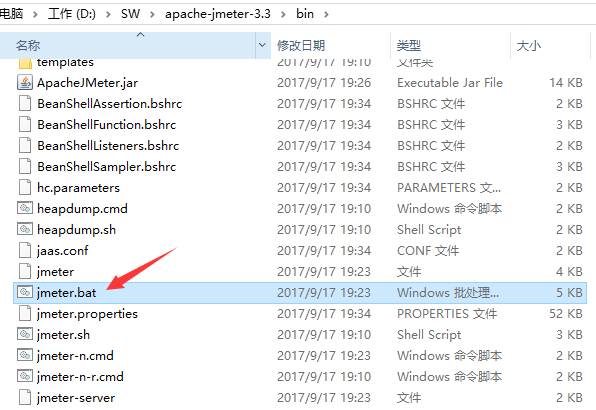
執行環境為Windows 10系統，JDK版本為1.8，JMeter版本為3.3。

**2. Jmeter安裝**

**2.1 JDK安裝**

由於Jmeter是基於java開發，首先需要下載安裝JDK （目前JMeter只支援到Java 8，尚不支援 Java 9）   
1. 官網下載地址：[http://www.oracle.com/technetwork/java/javase/downloads/index.html](http://www.oracle.com/technetwork/java/javase/downloads/index.html" \t "_blank)   
2. 選擇Java SE 8u151/ 8u152，點選JDK下載   
  
3. 安裝下載的JDK   
4. 配置系統環境變數

**2.2 JMeter安裝**

1. 官網下載地址：[http://jmeter.apache.org/download\_jmeter.cgi](https://jmeter.apache.org/download_jmeter.cgi" \t "_blank)
2. 下載最新JMeter 3.3版本：apache-jmeter-3.3.zip   
   
3. 下載完成後解壓zip包
4. 啟動JMeter   
   雙擊JMeter解壓路徑（apache-jmeter-3.3\bin）bin下面的jmeter.bat即可   
   

**3. 測試例項**

我們選取最常見的百度搜索介面：

**3.1 介面地址**

[http://www.baidu.com/s?ie=utf-8&wd=jmeter](https://www.baidu.com/s?ie=utf-8&wd=jmeter)效能測試

**3.2 請求引數**

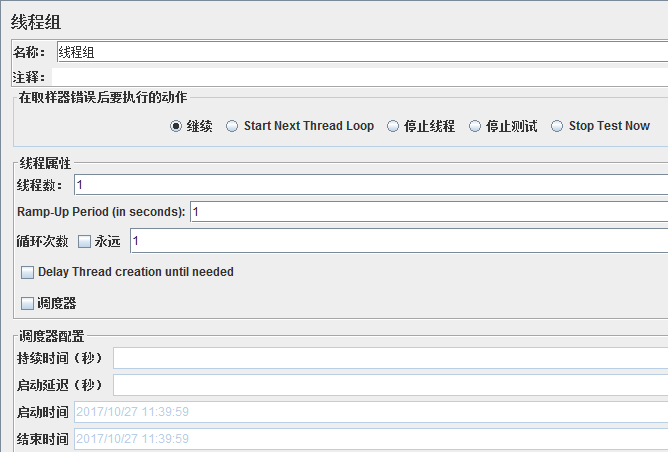
ie：編碼方式，預設為utf-8   
wd: 搜尋詞

**3.3 返回結果**

搜尋結果，我們可以通過校驗結果中是否含有搜尋詞wd來判斷本次請求成功或失敗。

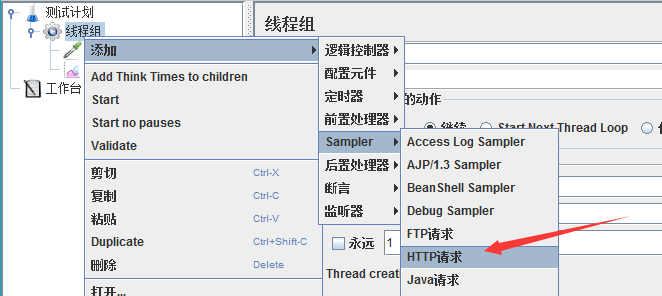
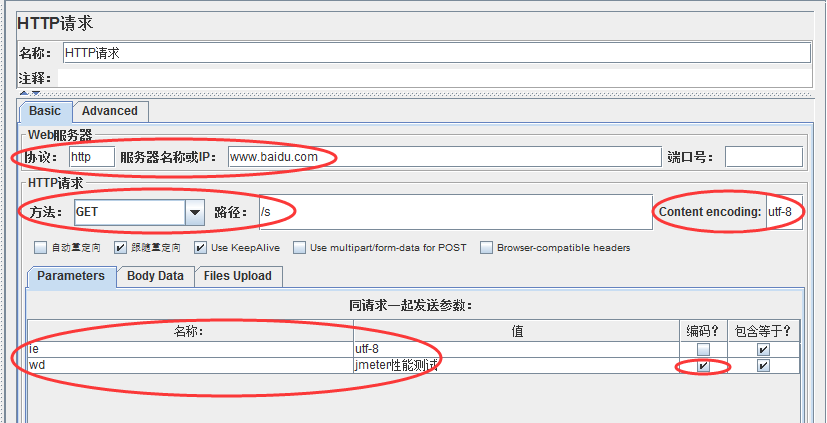
**4. JMeter指令碼編寫**

**4.1 新增執行緒組**

右鍵點選“測試計劃” -> “新增” -> “Threads(Users)” -> “執行緒組”   
   
這裡可以配置執行緒組名稱，執行緒數，準備時長（Ramp-Up Period(in seconds)）迴圈次數，排程器等引數：   
   
執行緒組引數詳解：   
1. 執行緒數：虛擬使用者數。一個虛擬使用者佔用一個程序或執行緒。設定多少虛擬使用者數在這裡也就是設定多少個執行緒數。   
2. Ramp-Up Period(in seconds)準備時長：設定的虛擬使用者數需要多長時間全部啟動。如果執行緒數為10，準備時長為2，那麼需要2秒鐘啟動10個執行緒，也就是每秒鐘啟動5個執行緒。   
3. 迴圈次數：每個執行緒傳送請求的次數。如果執行緒數為10，迴圈次數為100，那麼每個執行緒傳送100次請求。總請求數為10\*100=1000 。如果勾選了“永遠”，那麼所有執行緒會一直髮送請求，一到選擇停止執行指令碼。   
4. Delay Thread creation until needed：直到需要時延遲執行緒的建立。   
5. 排程器：設定執行緒組啟動的開始時間和結束時間(配置排程器時，需要勾選迴圈次數為永遠)   
持續時間（秒）：測試持續時間，會覆蓋結束時間   
啟動延遲（秒）：測試延遲啟動時間，會覆蓋啟動時間   
啟動時間：測試啟動時間，啟動延遲會覆蓋它。當啟動時間已過，手動只需測試時當前時間也會覆蓋它。   
結束時間：測試結束時間，持續時間會覆蓋它。

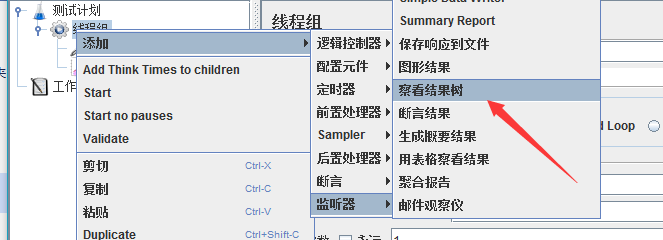
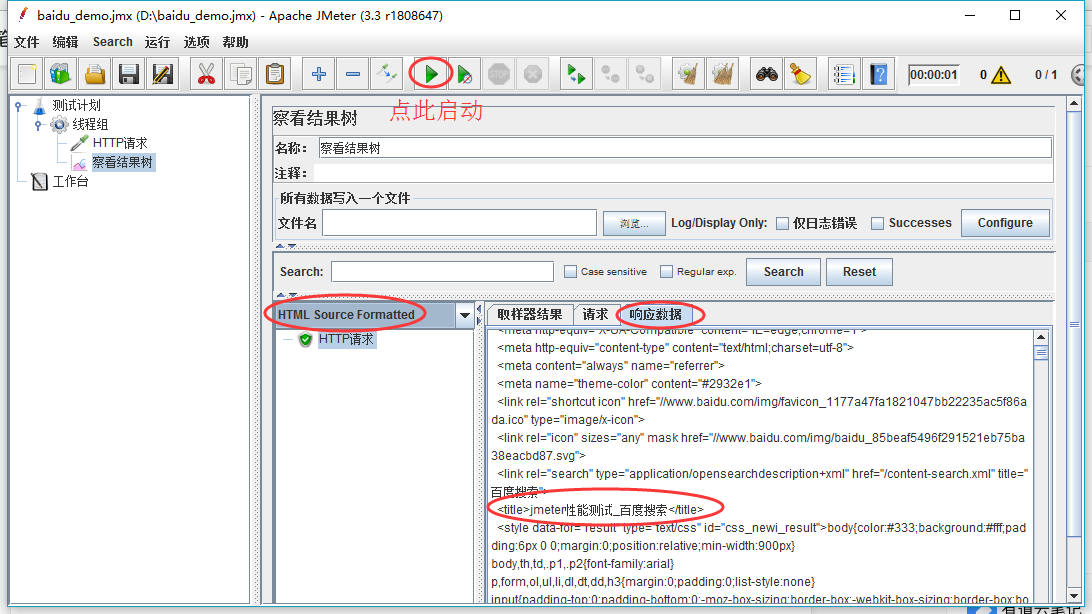
因為介面除錯需要，我們暫時均使用預設設定，待後面真正執行效能測試時再回來配置。

**4.2 新增HTTP請求**

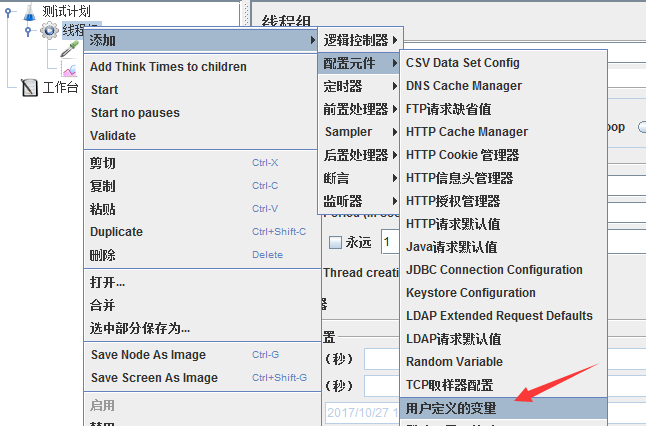
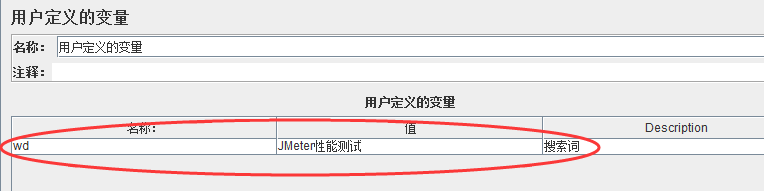
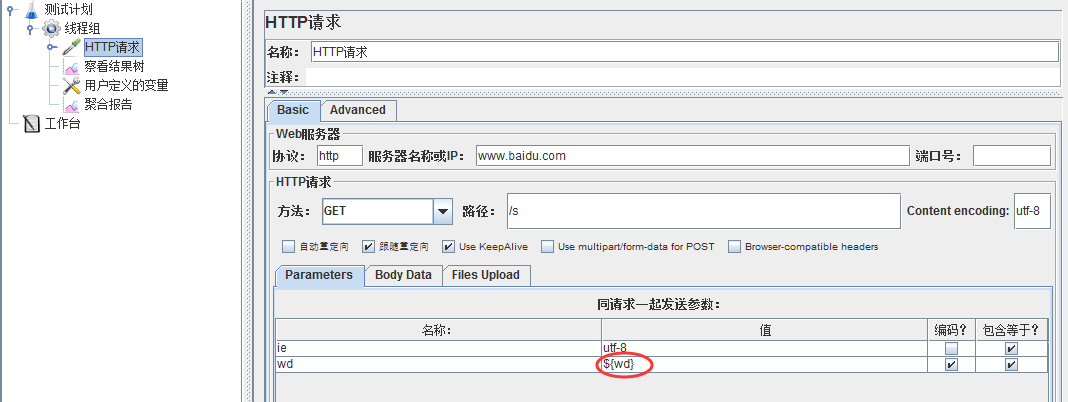
右鍵點選“執行緒組” -> “新增” -> “Sampler” -> “HTTP請求”   
   
對於我們的介面[http://www.baidu.com/s?ie=utf-8&wd=jmeter](https://www.baidu.com/s?ie=utf-8&wd=jmeter" \t "_blank)效能測試，可以參考下圖填寫：   
   
Http請求主要引數詳解：

1. Web伺服器   
   協議：向目標伺服器傳送HTTP請求協議，可以是HTTP或HTTPS，預設為HTTP   
   伺服器名稱或IP ：HTTP請求傳送的目標伺服器名稱或IP   
   埠號：目標伺服器的埠號，預設值為80   
   2.Http請求   
   方法：傳送HTTP請求的方法，可用方法包括GET、POST、HEAD、PUT、OPTIONS、TRACE、DELETE等。   
   路徑：目標URL路徑（URL中去掉伺服器地址、埠及引數後剩餘部分）   
   Content encoding ：編碼方式，預設為ISO-8859-1編碼，這裡配置為utf-8
2. 同請求一起傳送引數   
   在請求中傳送的URL引數，使用者可以將URL中所有引數設定在本表中，表中每行為一個引數（對應URL中的 name=value），注意引數傳入中文時需要勾選“編碼”

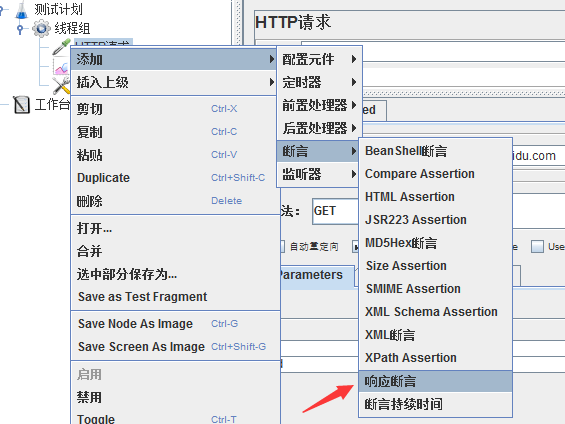
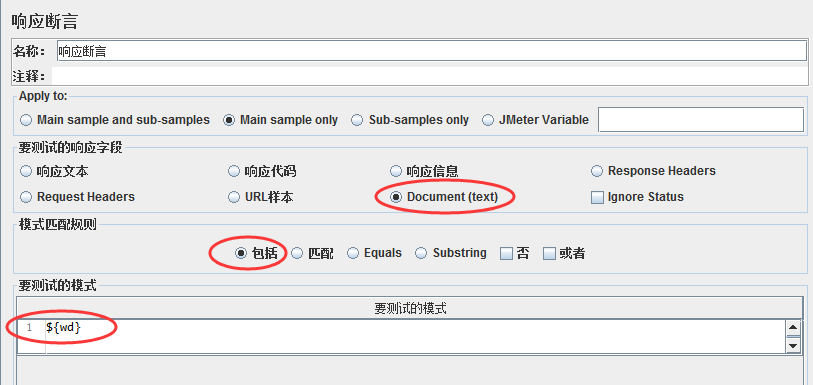
**4.3 新增察看結果樹**

右鍵點選“執行緒組” -> “新增” -> “監聽器” -> “察看結果樹”   
   
這時，我們執行Http請求，修改響應資料格式為“HTML Source Formatted”，可以看到本次搜尋返回結果頁面標題為”jmeter效能測試\_百度搜索“。   


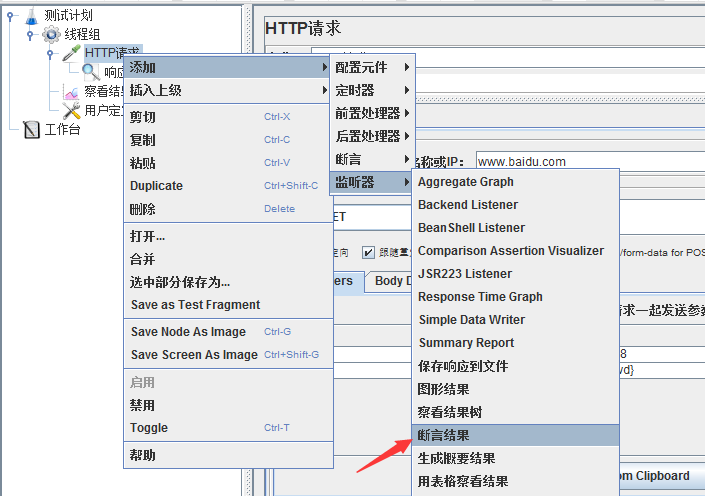
**4.4 新增使用者自定義變數**

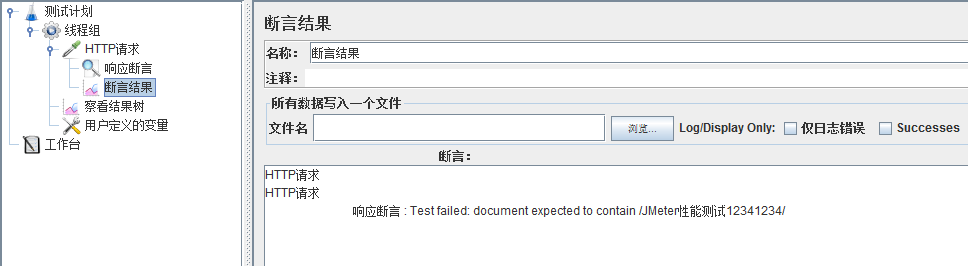
我們可以新增使用者自定義變數用以Http請求引數化，右鍵點選“執行緒組” -> “新增” -> “配置元件” -> “使用者定義的變數”：   
   
新增一個引數wd，存放搜尋詞：   
   
並在Http請求中使用該引數，格式為：${wd}   


**4.5 新增斷言**

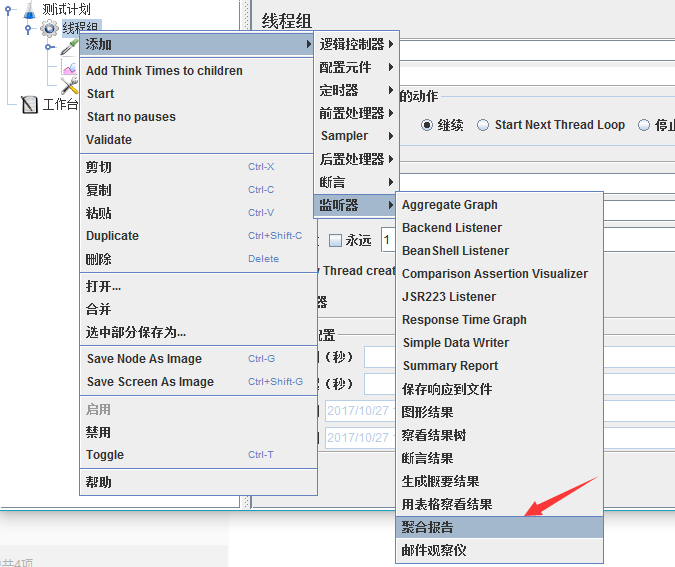
右鍵點選“HTTP請求” -> “新增”-> “斷言” -> “響應斷言”   
   
我們校驗返回的文字中是否包含搜尋詞，新增引數${wd}到要測試的模式中：   


**4.6 新增斷言結果**

右鍵點選“HTTP請求” -> “新增”-> “監聽器” -> “斷言結果”   


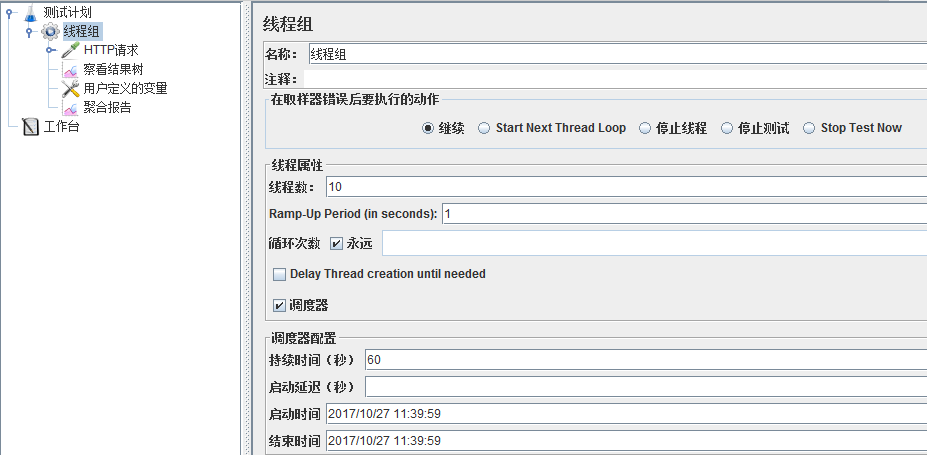
這時，我們再執行一次就可以看到斷言結果成功或失敗了   


**4.7 新增聚合報告**

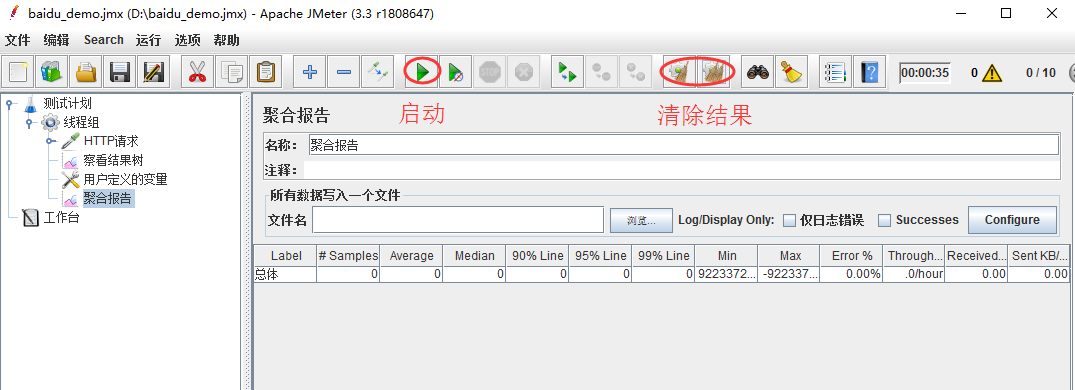
右鍵點選“執行緒組” -> “新增” -> “監聽器” -> “聚合報告”，用以存放效能測試報告   
   
這樣，我們就完成了一個完整Http介面的JMeter效能測試指令碼編寫。

**5. 執行效能測試**

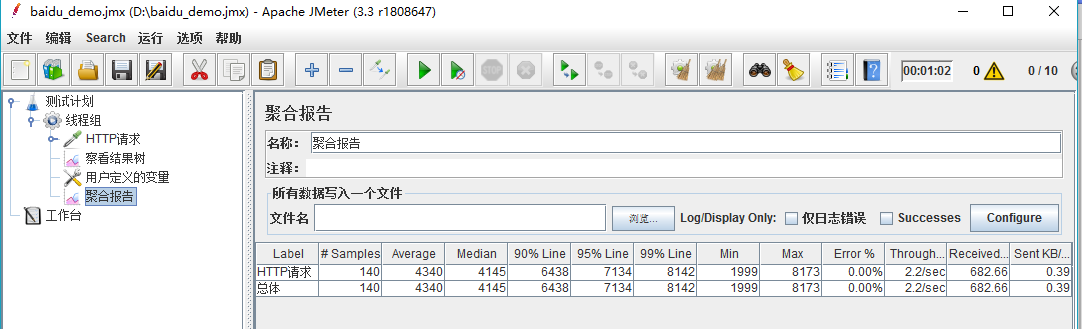
**5.1 配置執行緒組**

點選執行緒組，配置本次效能測試相關引數：執行緒數，迴圈次數，持續時間等，這裡我們配置併發使用者數為10，持續時間為60s   


**5.2 執行測試**

點選綠色小箭頭按鈕即可啟動測試，測試之前需要點選小掃把按鈕清除之前的除錯結果。   


**6. 分析測試報告**

待效能測試執行完成後，開啟聚合報告可以看到：   


聚合報告引數詳解：   
1. Label：每個 JMeter 的 element（例如 HTTP Request）都有一個 Name 屬性，這裡顯示的就是 Name 屬性的值   
2. #Samples：請求數——表示這次測試中一共發出了多少個請求，如果模擬10個使用者，每個使用者迭代10次，那麼這裡顯示100   
3. Average：平均響應時間——預設情況下是單個 Request 的平均響應時間，當使用了 Transaction Controller 時，以Transaction 為單位顯示平均響應時間   
4. Median：中位數，也就是 50％ 使用者的響應時間   
5. 90% Line：90％ 使用者的響應時間   
6. Min：最小響應時間   
7. Max：最大響應時間   
8. Error%：錯誤率——錯誤請求數/請求總數   
9. Throughput：吞吐量——預設情況下表示每秒完成的請求數（Request per Second），當使用了 Transaction Controller 時，也可以表示類似 LoadRunner 的 Transaction per Second 數   
10. KB/Sec：每秒從伺服器端接收到的資料量，相當於LoadRunner中的Throughput/Sec

一般而言，效能測試中我們需要重點關注的資料有： #Samples 請求數，Average 平均響應時間，Min 最小響應時間，Max 最大響應時間，Error% 錯誤率及Throughput 吞吐量。