



eBook Gratuit

APPRENEZ

Apache JMeter

eBook gratuit non affilié créé à partir des
contributeurs de Stack Overflow.

#jmeter

Table des matières

À propos.....	1
Chapitre 1: Démarrer avec Apache JMeter.....	2
Remarques.....	2
Versions.....	2
Exemples.....	3
Installation ou configuration.....	3
Vue d'ensemble des composants Apache JMeter à haut niveau.....	4
Chapitre 2: Apache JMeter: enregistrement de scénario de test.....	7
Introduction.....	7
Exemples.....	7
Enregistrement de script avec la fonctionnalité de modèle JMeter.....	7
Enregistrement de script avec l'enregistreur proxy JMeter.....	9
Scripts de performance d'enregistrement pour les appareils mobiles.....	12
Enregistrement du trafic HTTPS.....	13
Enregistrement de script avec l'extension Chrome de BlazeMeter.....	16
Enregistrement de script avec BadBoy.....	18
Chapitre 3: Corrélations Apache JMeter.....	20
Introduction.....	20
Exemples.....	20
Corrélation à l'aide de l'extracteur d'expression régulière dans Apache JMeter.....	20
Corrélation à l'aide de XPath Extractor dans JMeter.....	25
Corrélation à l'aide de l'extracteur CSS / JQuery dans JMeter.....	28
Corrélation à l'aide de l'extracteur JSON.....	32
Corrélation automatisée à l'aide du «SmartJMX» de BlazeMeter.....	35
Chapitre 4: Paramétrage d'Apache JMeter.....	39
Introduction.....	39
Exemples.....	39
Paramétrage à l'aide de fichiers externes.....	39
Paramétrage à l'aide de bases de données.....	46
Paramétrage à l'aide du plugin 'Parameterized Controller'.....	56

À propos

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [apache-jmeter](#)

It is an unofficial and free Apache JMeter ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official Apache JMeter.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapitre 1: Démarrer avec Apache JMeter

Remarques

JMeter est un outil de test de charge utilisé pour les [tests de performance](#) . Un testeur de performances peut enregistrer des actions dans un navigateur Web ou créer manuellement un *script* qui peut ensuite être exécuté avec des centaines ou des milliers d'utilisateurs.

JMeter peut être utilisé pour créer des utilisateurs et des scénarios incroyablement dynamiques en utilisant ses différents éléments. Par exemple, [CSV Data Set Config](#) peut être utilisé pour spécifier un ensemble d'utilisateurs à connecter à une application Web. Le [Regular Expression Extractor](#) ou le [CSS / JQuery Extractor](#) peut être utilisé pour enregistrer les identifiants de session à utiliser dans les requêtes futures. Le [JSR223 PreProcessor](#) couplé au langage [Groovy](#) peut être utilisé pour créer des données uniques dynamiques pour chaque utilisateur à envoyer dans le cadre d'un corps `POST` .

Versions

Version	Version Java	Date de sortie
3.2	Java 8+	2017-04-14
3.1	Java 7+	2016-11-20
3.0	Java 7+	2016-05-17
2.13	Java 6+	2015-03-13
2.12	Java 6+	2014-11-10
2.11	Java 6+	2014-01-05
2.10	Java 6+	2013-10-21
2.9	Java 6+	2013-01-28
2.8	Java 5+	2012-10-06
2.7	Java 5+	2012-05-27
2.6	Java 5+	2012-02-01
2.5.1	Java 5+	2011-10-03
2,5	Java 5+	2011-08-17
2.4	Java 5+	2010-07-12

Version	Version Java	Date de sortie
2.3.4	Java 1.4+	2009-06-21

Exemples

Installation ou configuration

1. Téléchargez une archive distribuée à partir de la section Binaries de JMeter depuis la page [Download Apache JMeter](#) .
2. Selon la version que vous avez téléchargée, vérifiez [les exigences minimales de la version Java](#) et installez Java si nécessaire. Assurez-vous que la variable d'environnement `JAVA_HOME` est définie et pointe vers une version correcte.
3. Extrayez l'archive de distribution dans le répertoire de votre choix.
4. Ouvrez l'interface utilisateur de JMeter:
 - **Sous Windows** : accédez au `<jmeter_location>\bin` et exécutez `jmeterw.bat` ou `jmeter.bat`
 - **Sur Linux / Mac** : accédez au `<jmeter_location>/bin` et exécutez `jmeter` ou `"jmeter.sh"`.

Par exemple:

```
cd /Users/me/apache-jmeter/bin
./jmeter
```

Remarque : si la commande ci-dessus échoue avec l'erreur `Permission denied` , définissez les autorisations d'exécution sur le fichier `jmeter` :

```
cd /Users/me/apache-jmeter/bin
chmod u+x ./jmeter
```

Si vous pouvez voir l'interface utilisateur de JMeter, la configuration de base a réussi.

sur le volet de test et en accédant au composant que vous souhaitez ajouter.

2. **Workbench** : est un endroit temporaire pour démarrer le script. Avec tous les composants disponibles dans Plan de test, vous obtenez `HTTP(s) Test Script Recorder` afin d' record les actions du navigateur. Les scripts peuvent être enregistrés dans le Workbench, à condition de cocher la case "Save Workbench", sinon ils ne sont pas.
3. **Threads (Users)** : vous pouvez définir un nombre d'utilisateurs (virtuels) à exécuter, le temps de montée en charge et le nombre de boucles. Vous pouvez également définir sur Plan de test si les groupes de threads doivent être exécutés en séquence ou en parallèle dans le cas de plusieurs groupes de threads. Quelques exemples sont `Thread Group`, `setUp Thread Group`, and `tearDown Thread Group`
4. **Logic Controller** : Permet de définir le flux d'exécution et le regroupement des échantillonneurs. L'un des exemples utiles est `Transaction Controller`, où vous combinez tous les échantillonneurs de la page de connexion (toutes les ressources, y compris les images, les fichiers .css et .js) de sorte que le temps de réponse combiné puisse être récupéré.
5. **Sampler** : Sampler est le cœur du JMeter. Il donne des composants pour simuler des requêtes de différents protocoles tels que HTTP, JDBC, FTP, SMTP etc. Par exemple, HTTP sampler permet de simuler un paquet HTTP (de GET, POST ou de toute méthode supportée). Les protocoles de flux principaux sont pris en charge, pour d'autres, vous pouvez utiliser des plug-ins gratuits ou commerciaux.
6. **Config Element configuration**: Les éléments de configuration peuvent être utilisés pour définir les valeurs par défaut et les variables à utiliser ultérieurement par les échantillonneurs. Notez que ces éléments sont généralement traités au début du périmètre dans lequel ils se trouvent, c'est-à-dire avant tous les échantillonneurs dans la même portée. `CSV Dataset Config` vous permet de fournir des données de test telles que les noms d'utilisateur, les mots de passe du scénario de connexion à `from a file`. `User Defined variables` élément de configuration des `User Defined variables` vous permet de définir des variables pouvant être utilisées dans le plan de test, mais dans lesquelles chaque thread a sa propre copie.
7. **Timer** : Par défaut, un thread JMeter exécute les échantillonneurs en séquence sans pause. Les composants présentés ici fournissent la fonctionnalité permettant d'introduire `User Think Time` sous différentes formes parmi les échantillonneurs. Quelques exemples sont le `Constant Timer`, `Constant Throughput Timer`.
8. **Pre Processors** : vous permettent d'effectuer des opérations / actions avant l'exécution de l'échantillonneur. `JSR223 Pre Processor` avec [Apache Groovy](#) (similaire au style de codage Java) vous permet d'apporter des modifications à l'échantillonneur avant de l'envoyer.
9. **Post Processors** : permettent d'effectuer des opérations / actions après l'exécution de l'échantillonneur. Quelques exemples utiles sont la récupération de valeurs dynamiques telles que l'ID de session, à l'aide du post-processeur `Regular Expression Extractor` pour tout type de texte, `CSS/jQuery Extractor` pour HTML, `JSON Extractor` pour JSON, `XPath Extractor` pour XML.
10. **Assertions** : Comme son nom l'indique, vous pouvez affirmer la réponse des échantillonneurs de différentes manières, comme rechercher du texte, la taille de la réponse et la durée pour recevoir la réponse, etc. Vous pouvez par exemple utiliser l' `Response Assertion` pour rechercher du texte. dans la réponse. Si l'assertion échoue, JMeter marque l'échantillonneur, auquel l'assertion est appliquée, comme échec.
11. **Les auditeurs**: les auditeurs vous permettent de sauvegarder les résultats des tests, voir l'exécution du test, etc., par exemple, en utilisant l' `View Results Tree` , vous pouvez voir la

demande / réponse des échantillonneurs et indiquer s'ils sont marqués PASS (couleur verte) / FAIL JMeter. En utilisant le rapport d'agrégat, vous pouvez enregistrer les résultats du test au format CSV. Il est important de noter que vous utilisez les écouteurs avant le test (pour le débogage du script de test) ou après le test (pour afficher les résultats sous forme de graphique ou de résumé) mais pas pendant l'exécution. nous devons supprimer les écouteurs pendant le test car il consomme beaucoup de ressources système. Donc, nous exécutons le test en mode non-interface graphique et sauvegardons les résultats en utilisant l'option `-l` aux formats `.csv/.jtl`. Publiez le test, vous pouvez charger ces fichiers enregistrés dans l'un des écouteurs du JMeter pour afficher les graphiques / le résumé.

Voici la syntaxe générale (`you add any component on need basis`):

```
Test Plan
  Thread Group
    Config Element
    Logic Controller
      Pre Processor
      Sampler
      Timer
      Post Processor
      Assertion
    Listener
```

Les références:

1. [Plan de test et composants](#)
2. [Ordre d'exécution](#)
3. [Règles de portée](#)

Lire Démarrer avec Apache JMeter en ligne: <https://riptutorial.com/fr/jmeter/topic/1941/demarrer-avec-apache-jmeter>

Chapitre 2: Apache JMeter: enregistrement de scénario de test

Introduction

L'enregistrement des scénarios de test est l'un des moyens les plus pratiques de créer des scripts de test. En effet, les enregistrements de test vous permettent d'imiter des flux de travail utilisateur réalistes, au lieu de créer manuellement un script de test. Les enregistrements capturent toutes les requêtes du navigateur vers l'application Web, puis créent automatiquement un fichier jmx pouvant être exécuté dans des tests de performances. En utilisant la fonctionnalité d'enregistrement / lecture de JMeter ou des outils tiers tels que BlazeMeter et BadBoy, les testeurs peuvent accélérer leur travail 3 fois.

Exemples

Enregistrement de script avec la fonctionnalité de modèle JMeter

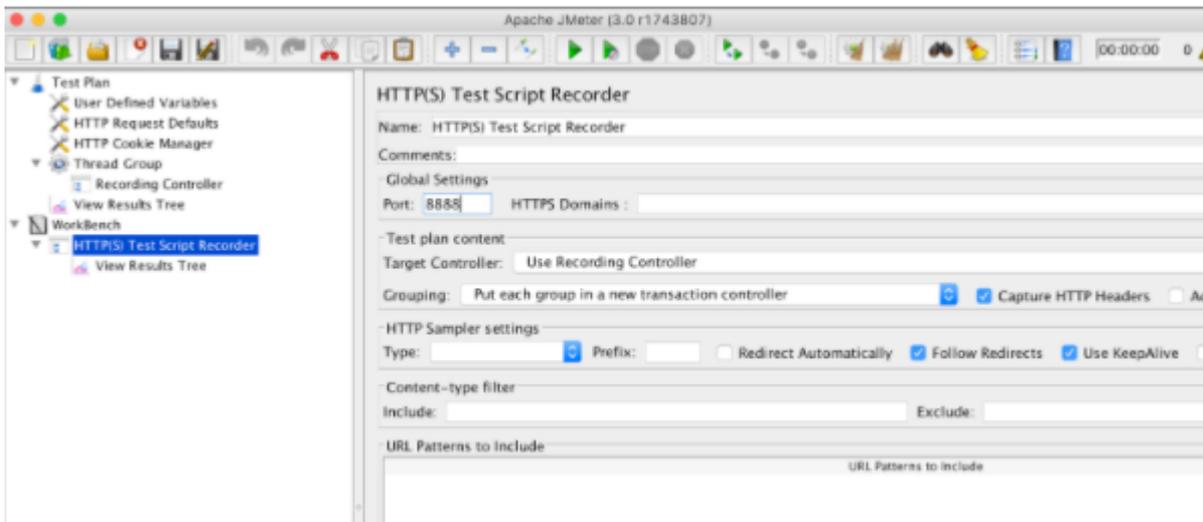
Dans la version 2.10, [JMeter a](#) introduit un mécanisme qui vous permet de gagner du temps lors de la création de scripts: les modèles JMeter. Ces modèles sont des squelettes qui peuvent être réutilisés comme base pour vos nouveaux scripts.

JMeter dispose déjà de plusieurs modèles disponibles avec des descriptions détaillées et vous pouvez également ajouter les vôtres. Les modèles ont toute la configuration et les éléments requis pour enregistrer vos scripts de performance à partir de rien.

Voici comment utiliser la fonctionnalité de modèle JMeter:

Configurez JMeter

1. Ouvrir JMeter
2. Sélectionnez le modèle pour l'enregistrement du script:
Fichier -> Modèles... -> Sélectionner un modèle -> Enregistrement -> Créer JMeter ajoutera les éléments pertinents à l'arbre de test.



Configurez le proxy de votre navigateur

Pour utiliser JMeter Recorder, vous devez configurer votre navigateur pour envoyer toutes les demandes via un proxy. N'importe quel navigateur peut être utilisé pour ces besoins, bien qu'il puisse exister des différences entre les configurations des navigateurs, spécifiques au navigateur et pouvant varier selon le système d'exploitation.

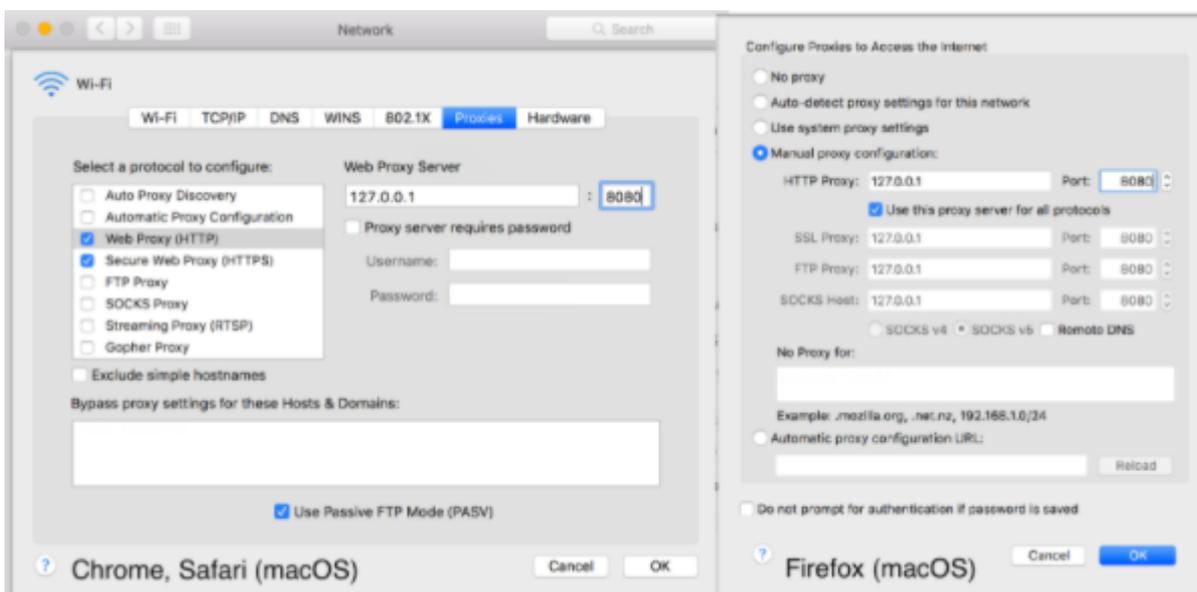
3. Pour configurer votre navigateur:

Chrome : bouton Menu -> Paramètres -> Afficher les paramètres avancés... -> Réseau -> Modifier les paramètres du proxy

Safari : Préférences -> Avancé -> Proxies -> Modifier les paramètres ...

Firefox : bouton Menu -> Préférences -> Avancé -> Réseau -> Connexion -> Paramètres ..

4. Par exemple, vous pouvez utiliser le localhost 127.0.0.1. Remplacez le port par le port dans HTTP (S) Script Recorder.



Si vous rencontrez des problèmes lors de cette étape, vérifiez que vous ne disposez d'aucun plug-

in tiers capable de gérer les paramètres de proxy de votre navigateur. Si vous le faites, comme Hola VPN par exemple, les paramètres du proxy ne seront pas disponibles dans le menu de votre navigateur.

5. Cliquez sur le bouton "Démarrer", situé au bas de la page "HTTP (S) Test Script Recorder", et parcourez le flux de travail de l'application Web à tester. Lorsque vous revenez à JMeter, vous devriez voir toutes les requêtes capturées depuis votre navigateur.

Enregistrement de script avec l'enregistreur proxy JMeter

JMeter vous permet également de configurer manuellement votre espace de travail. C'est plus complexe, mais vous pouvez faire en sorte que les scripts correspondent exactement à vos besoins.

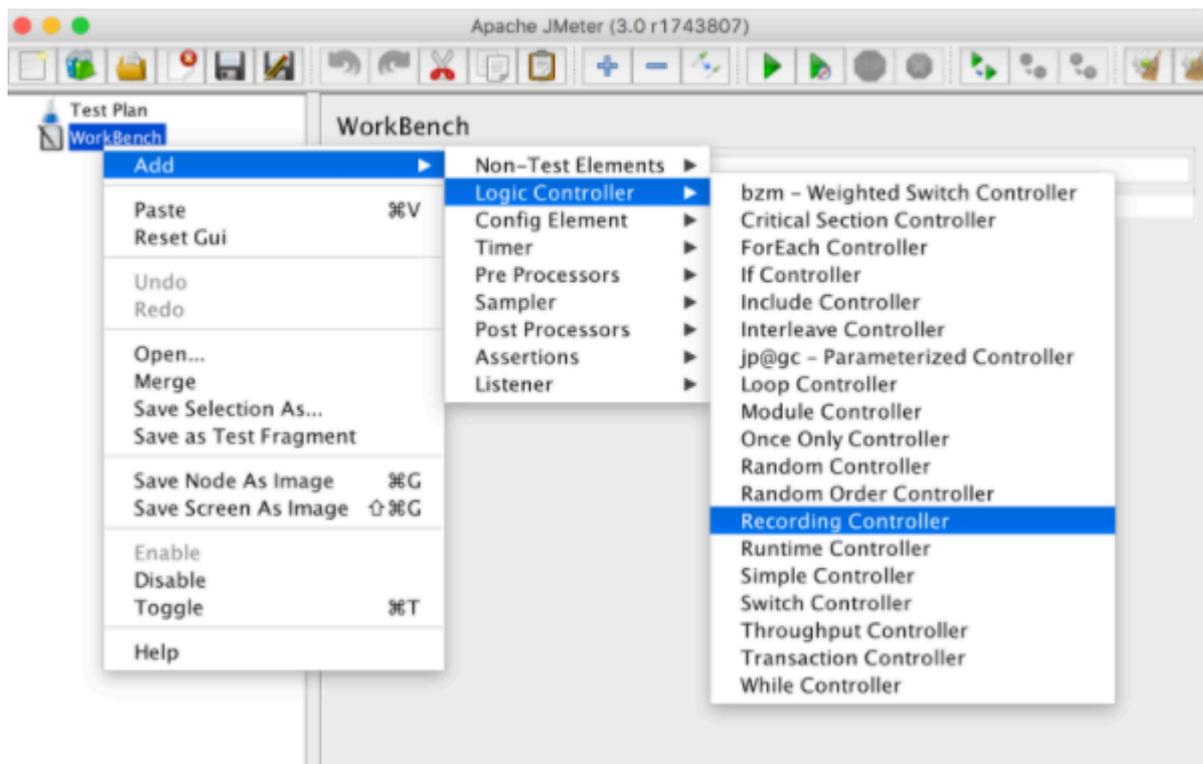
Configurez le proxy de votre navigateur

1. Configurez votre navigateur, comme décrit au chapitre 1.

Configurez JMeter

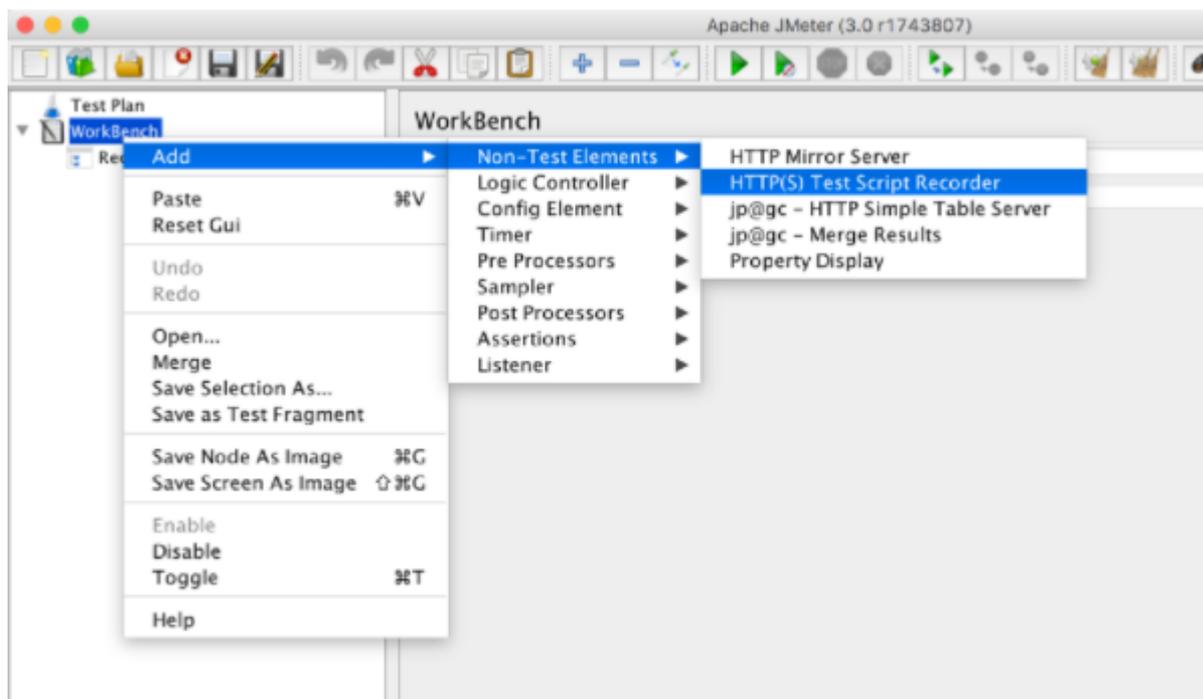
La branche «WorkBench» peut être utilisée comme espace de travail temporaire pour la création de scripts. Gardez à l'esprit que les entrées ajoutées à cette section ne seront pas enregistrées avec le plan de test. Par conséquent, si vous souhaitez réutiliser la même configuration d'enregistrement à l'avenir, vous devrez la copier et la coller dans la section «Plan de test».

2. Ajouter "Contrôleur d'enregistrement" à "WorkBench": Cliquez avec le bouton droit sur "WorkBench" -> "Ajouter" -> "Logic Controller" -> "Recording Controller"



3. Ajoutez «HTTP (S) Test Script Recorder» au même «WorkBench»: faites un clic droit sur

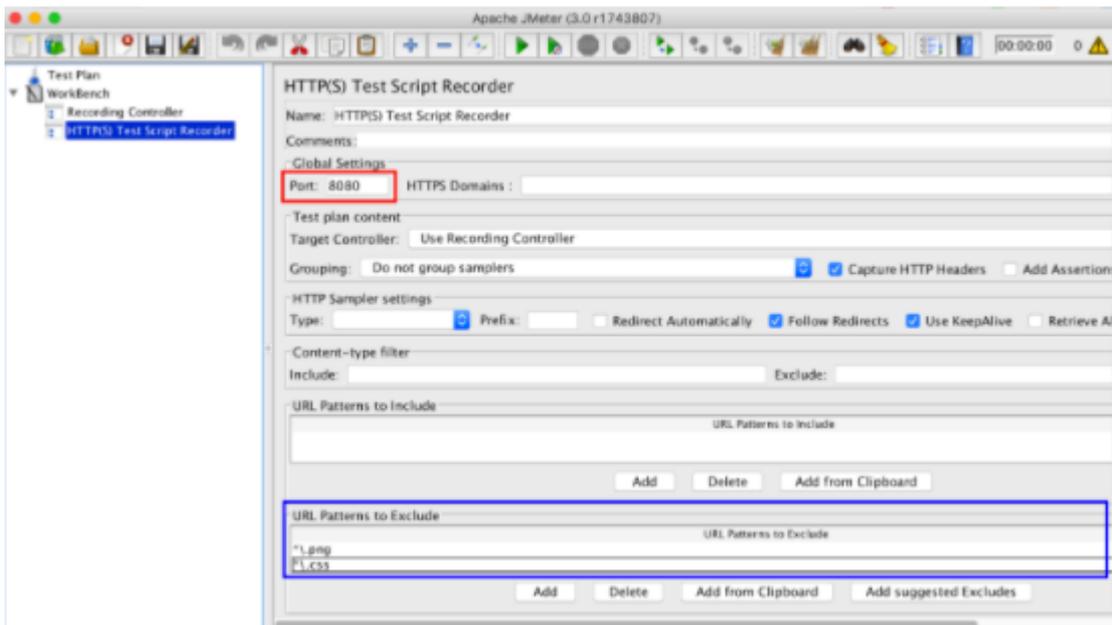
«WorkBench» -> «Ajouter» -> «Eléments non testés» -> «HTTP (S) Test Script Recorder»



4. Sur la page de configuration «HTTP (S) Test Script Recorder» dans «Global Settings: Port», vous devez placer le même port que celui spécifié dans la configuration du proxy de votre navigateur, par exemple 8080.
5. Si vous souhaitez exclure des requêtes vers des ressources spécifiques, vous pouvez utiliser la section «Modèles d'URL à exclure». Cela peut être utile si vous souhaitez inclure uniquement les types de contenu que vous souhaitez demander (par exemple *.html, *.php, etc.) ou exclure les types de contenu que vous ne souhaitez pas demander (par exemple, *.jpg, *.png, *.js, etc.).

Quand l'utiliserions-nous? Par exemple, lorsque vous enregistrez un script qui appelle une application tierce ou lorsque vous testez un script côté serveur, vous ne souhaitez peut-être pas télécharger d'actifs, car ils risquent d'encombrer vos tests et de consommer de la bande passante un certain chemin.

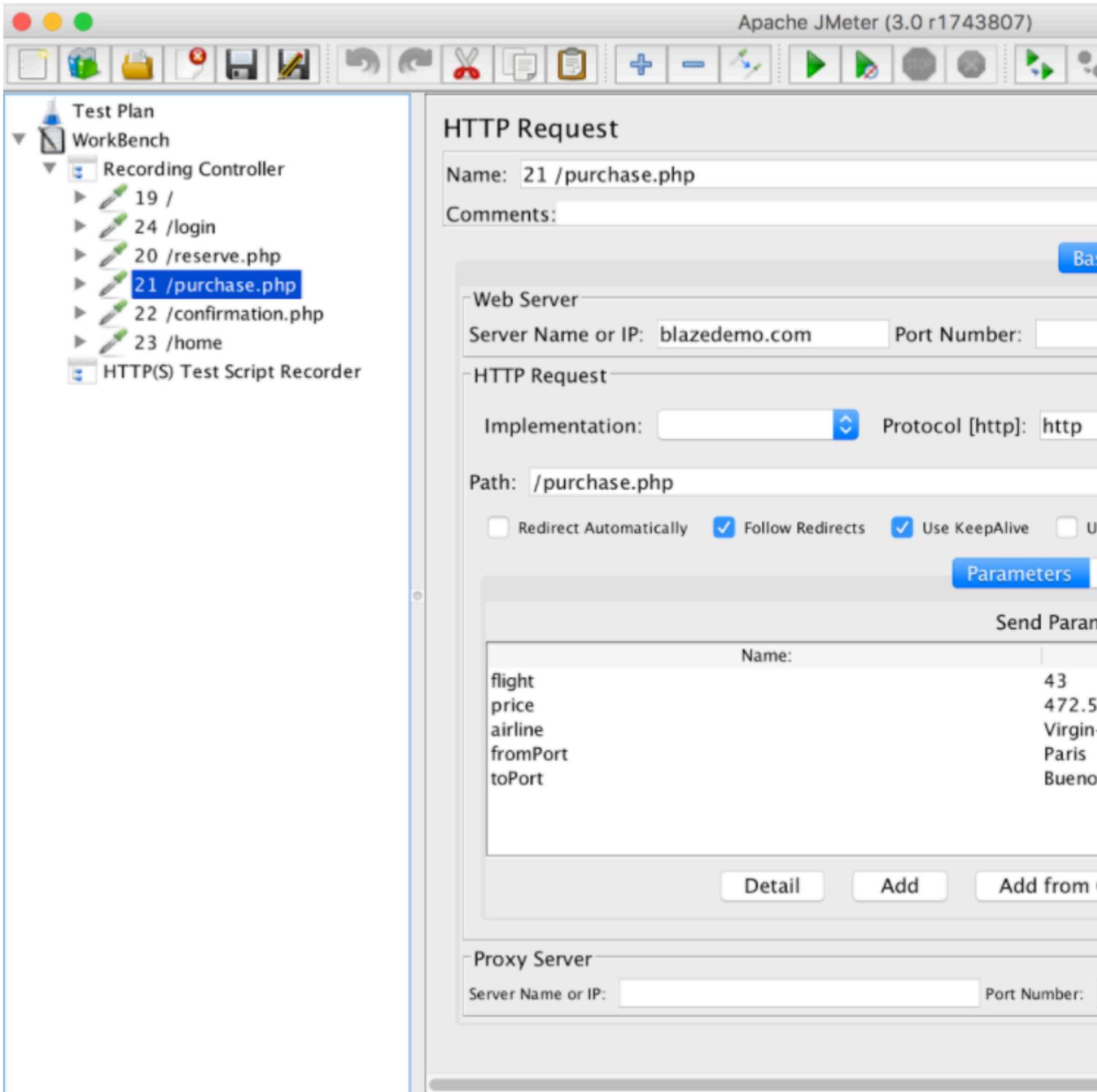
Les modèles d'exclusion les plus courants sont: *.png* “,”. *.jpg* “,”. *.gif* “,”. *.css* ”,”. *.js* ”. Vous pouvez également combiner différents motifs ensemble. Ce modèle combiné devrait éliminer toutes les requêtes redondantes susceptibles de vous distraire des requêtes importantes: «. (bmp | css | js | gif | ico | jpe? g | png | swf | woff) ”



6. D'un autre côté, vous pouvez vouloir imiter le comportement complet du navigateur et inclure le chargement de toutes les ressources. Dans ce cas, il n'est pas nécessaire d'exclure les modèles d'URL. Gardez à l'esprit que le navigateur télécharge toutes les ressources incorporées à partir de la page demandée et qu'il dispose d'un mécanisme de mise en cache susceptible d'affecter les performances du résultat.

Dans ce cas, il est recommandé de télécharger toutes les ressources incorporées dans le script: Cliquez avec le bouton droit sur «Plan de test» -> «Ajouter» -> «Elément de configuration» -> «HTTP Request Defaults» -> «Advanced» -> select Case à cocher «Récupérer toutes les ressources incorporées».

7. Pour que JMeter se comporte davantage comme un véritable navigateur, il est recommandé d'ajouter le «gestionnaire de cache HTTP», qui vous permet de simuler la fonctionnalité de mise en cache du navigateur dans vos tests de performances. Faites un clic droit sur "Plan de test" -> "Ajouter" -> "Elément de configuration" -> "Gestionnaire de cache HTTP".
8. Cliquez maintenant sur le bouton "Démarrer", situé au bas de la page "HTTP (S) Test Script Recorder", et parcourez le flux de travail de l'application Web à tester. Lorsque vous revenez à JMeter, vous devez voir toutes les requêtes capturées à partir de votre navigateur sous le «Recording Controller».



Scripts de performance d'enregistrement pour les appareils mobiles

JMeter peut également être utilisé pour enregistrer des tests de performances mobiles. L'enregistrement de scripts mobiles est très similaire à l'enregistrement de scripts d'applications Web.

Configurez JMeter

1. Configurez «Modèles JMeter» comme spécifié au chapitre 1.

Configurez votre téléphone portable

Une fois la configuration de JMeter préparée, y compris l'élément JMeter «HTTP (S) Test Script Recording» démarré sur un port spécifié, vous pouvez configurer votre téléphone mobile pour envoyer une demande à l'application Web que vous testez via le proxy JMeter.

2. IOS :

- Réglage -> Wi-Fi
- Cliquez sur le réseau connecté
- Accédez à la section de configuration «HTTP PROXY».
- Cliquez sur l'onglet "Manuel"
- Définir l'adresse IP de l'ordinateur sur lequel l'application JMeter est exécutée sous «Serveur»
- Définissez le port spécifié sur «HTTP (S) Test Script Recording» sous «Port»

Android:

- Réglage -> Wi-Fi
- Cliquez longuement sur le réseau connecté et cliquez sur l'option "Modifier le réseau"
- Cliquez sur la case à cocher «Options avancées»
- Définissez l'option "Proxy" sur "Manuel"
- Définissez le «Nom d'hôte du proxy» comme adresse IP de votre ordinateur et «Port proxy» comme spécifié dans la configuration «Enregistrement de script de test HTTP (S)» sous «Port».
- Cliquez sur "Enregistrer"

3. Vous pouvez maintenant lancer l'application sur votre appareil mobile. Les demandes seront enregistrées sur JMeter.

Enregistrement du trafic HTTPS

Si votre application Web utilise le cryptage SSL, vous devez capturer le trafic HTTPS au lieu de HTTP. Pour enregistrer le trafic HTTPS avec JMeter, vous devez configurer les certificats SSL.

Configurez votre proxy SSL

1. Assurez-vous que le proxy SSL est configuré de la même manière que le proxy HTTP est configuré:

Configure Proxies to Access the Internet

- No proxy
- Auto-detect proxy settings for this network
- Use system proxy settings
- Manual proxy configuration:

HTTP Proxy: Port:

Use this proxy server for all protocols

SSL Proxy: Port:

FTP Proxy: Port:

SOCKS Host: Port:

Configurez JMeter

2. Lancez l'enregistrement du script en utilisant la fonctionnalité «JMeter Recording Template», comme expliqué dans l'exemple «Enregistrement de script avec la fonctionnalité de modèle JMeter».
3. Après avoir ouvert l'application Web, vous verrez un message concernant une connexion non sécurisée. Pour continuer, il vous suffit d'accepter le certificat factice JMeter:
 - Cliquez sur 'Avancé'
 - Cliquez sur 'Ajouter une exception ...'
 - Désélectionnez "Stocker cette exception de manière permanente"
 - Cliquez sur 'Confirmer l'exception de sécurité'



Your connection is not secure

The owner of **example.com** has configured their website improperly. To protect your information from being stolen, Firefox has not connected to this website.

[Learn more...](#)

Go Back

Advanced

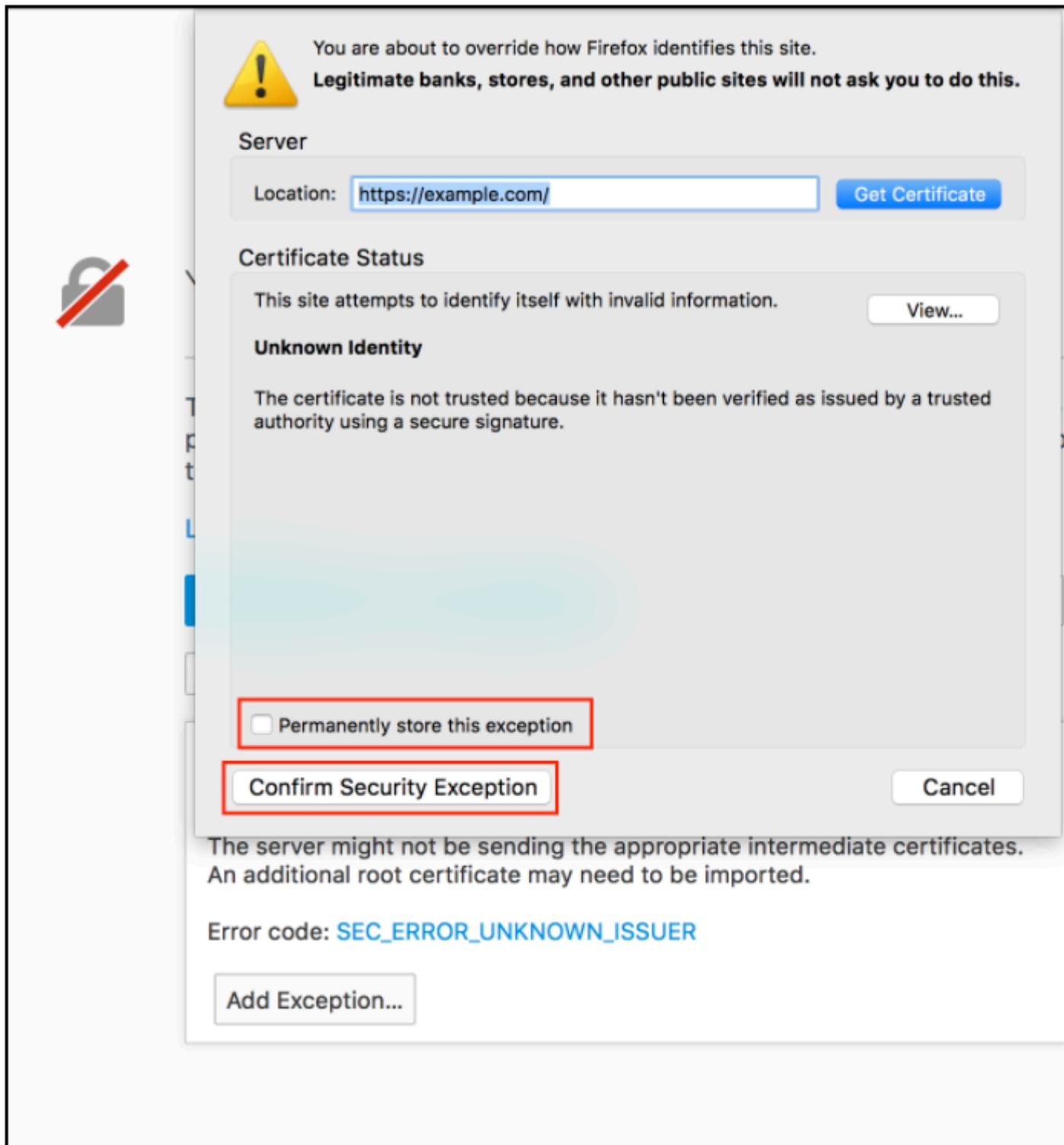
Report errors like this to help Mozilla identify misconfigured sites

example.com uses an invalid security certificate.

The certificate is not trusted because the issuer certificate is unknown. The server might not be sending the appropriate intermediate certificates. An additional root certificate may need to be imported.

Error code: [SEC_ERROR_UNKNOWN_ISSUER](#)

Add Exception...



4. Si vous voyez le «Ce site fournit une identification valide et vérifiée. Il n'y a pas besoin d'ajouter une exception. »Message d'avertissement, vous devez effacer l'historique du navigateur pour votre application, y compris les cookies, le cache, les données du site Web hors connexion. Ensuite, procédez à nouveau avec les mêmes étapes.

Cette approche fonctionne également pour l'enregistrement de scripts mobiles, car le certificat JMeter doit être installé uniquement sur l'hôte utilisé pour exécuter JMeter.

Enregistrement de script avec l'extension Chrome de BlazeMeter

Jusqu'à présent, nous avons couvert les méthodes de base pour enregistrer des scénarios de test. Mais l'une des manières les plus rapides et les plus simples d'enregistrer vos scripts de performance, également gratuits, consiste à utiliser l'extension [Chrome de BlazeMeter Recorder](#) . Ces enregistrements peuvent être exécutés dans JMeter ou dans BlazeMeter.

La raison pour laquelle l'extension est si utile, c'est qu'elle vous permet d'enregistrer des scripts de performance depuis votre navigateur sans avoir à configurer votre proxy.

Record FollowMe Sign Up | Login

ENTER THE NAME OF THE TEST

● ■ || ✎ .jmx ↻

Concurrency

50 Light load

Load Origin EU WEST (IRELAND)

User Agent DEFAULT

Include Filter Pattern * ?

http://*/*, https://*/*

▶ Advanced Options Help

Pour créer un nouveau script de performance:

1. Ouvrez l'enregistreur depuis votre Chrome
2. Entrez un nom de test dans le champ supérieur
3. Commencez l'enregistrement en cliquant sur le bouton d'enregistrement, sous la forme d'un cercle, et exécutez les actions Web que vous souhaitez enregistrer. Toutes vos demandes seront capturées. L'extension Blazemeter Chrome prend également en charge l'enregistrement du trafic HTTPS.
4. Une fois l'enregistrement terminé, cliquez sur le bouton Stop en forme de carré. Vous pouvez également mettre votre enregistrement en pause, puis le reprendre, ainsi que le modifier, au format .jmx ou JSON ou dans le cloud.
5. Exportez votre enregistrement - pour exécuter le test dans JMeter, exportez au format .jmx en cliquant sur le bouton .jmx. Pour lancer le test dans BlazeMeter, cliquez sur "play".

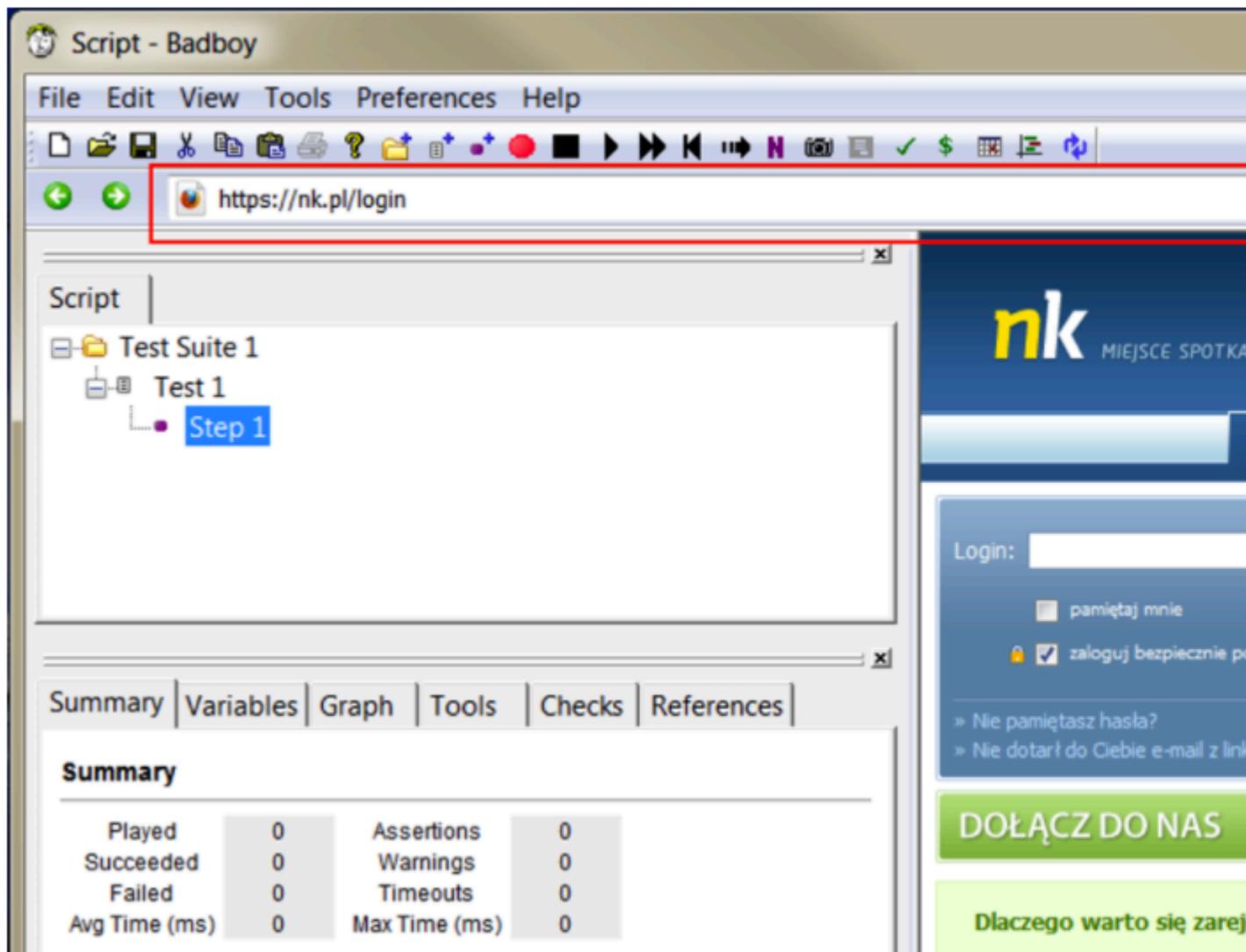
Pour plus d'informations, voir [ici](#) .

Enregistrement de script avec BadBoy

BadBoy est un autre outil d'enregistrement utile. Cependant, cela ne fonctionne que pour Windows.

Pour créer un nouveau script de performance:

1. Installez BadBoy [ici](#)
2. Entrez l'URL sous test dans la barre d'adresse



3. Appuyez sur le bouton d'enregistrement en forme de cercle rouge et effectuez les actions que vous souhaitez capturer.
4. Exportez votre script vers JMeter - Fichier -> Exporter vers JMeter

Pour plus d'informations, voir [ici](#) .

L'utilisation d'un enregistreur de scripts de performance est un excellent moyen d'éviter les tâches de routine tout en obtenant les meilleurs scripts de test. Après l'enregistrement, configurez le test en fonction du nombre d'utilisateurs virtuels que vous souhaitez tester, ainsi que des configurations de test supplémentaires, exécutez votre test et analysez les résultats pour identifier les erreurs et les goulots d'étranglement et caractériser les tendances de votre système.

Lire Apache JMeter: enregistrement de scénario de test en ligne:

<https://riptutorial.com/fr/jmeter/topic/8798/apache-jmeter--enregistrement-de-scenario-de-test>

Chapitre 3: Corrélations Apache JMeter

Introduction

Dans les tests de performances de JMeter, les corrélations permettent de récupérer des données dynamiques à partir de la réponse du serveur et de les publier dans les requêtes suivantes. Cette fonctionnalité est essentielle pour de nombreux aspects des tests, tels que les applications protégées basées sur des jetons.

Exemples

Corrélation à l'aide de l'extracteur d'expression régulière dans Apache JMeter

Si vous devez extraire des informations d'une réponse textuelle, le plus simple est d'utiliser des expressions régulières. Le modèle de correspondance est très similaire à celui utilisé dans Perl. Supposons que nous voulons tester un flux de travail d'achat de billets d'avion. La première étape consiste à soumettre l'opération d'achat. L'étape suivante consiste à s'assurer que nous sommes en mesure de vérifier tous les détails en utilisant l'ID d'achat, qui doit être retourné pour la première demande. Imaginons que la première requête renvoie une page HTML avec ce type d'identifiant que nous devons extraire:

```
<div class="container">
  <div class="container hero-unit">
    <h1>Thank you for you purchse today!</h1>
    <table class="table">
      <tr>
        <td>Id</td>
        <td>Your purchase id is 1484697832391</td>
      </tr>
      <tr>
        <td>Status</td>
        <td>Pending</td>
      </tr>
      <tr>
        <td>Amount</td>
        <td>120 USD</td>
      </tr>
    </table>
  </div>
</div>
```

Ce type de situation est le meilleur candidat pour utiliser l'extracteur JMeter Regular Expression. Regular Expression est une chaîne de texte spéciale permettant de décrire un modèle de recherche. De nombreuses ressources en ligne aident à écrire et à tester les expressions régulières. L'un d'eux est <https://regex101.com/> .

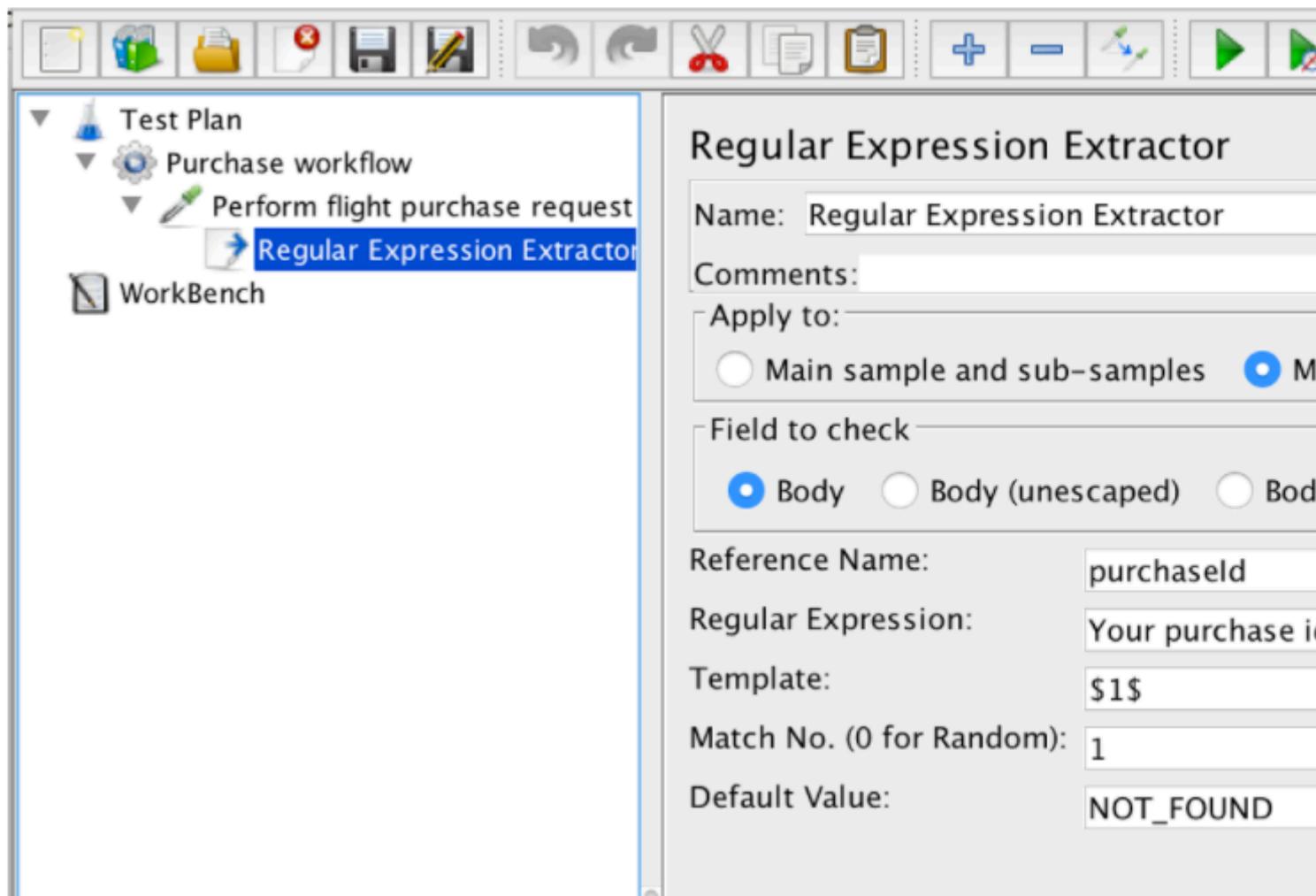
REGULAR EXPRESSION

:/ Your purchase id is (\d*).

TEST STRING

```
.....|
<tbody>
  <tr>
    <td>Id</td>
    <td>Your purchase id is 1484697832391. T
  </tr>
  <tr>
    <td>Status</td>
    <td>PendingCapture</td>
  </tr>
.....
```

Pour utiliser ce composant, ouvrez le menu JMeter et: *Ajouter -> Post-processeurs -> Extracteur d'expression régulière*



Le Regular Expression Extractor contient les champs suivants:

- Nom de référence - le nom de la variable pouvant être utilisée après l'extraction
- Expression régulière - une séquence de symboles et de caractères exprimant une chaîne (motif) qui sera recherchée dans le texte
- Template - contient des références aux groupes. Comme une regex peut avoir plusieurs groupes, elle permet de spécifier la valeur de groupe à extraire en spécifiant le numéro de groupe: \$ 1 \$ ou \$ 2 \$ ou \$ 1 \$\$ 2 \$ (extrayez les deux groupes)
- N ° de correspondance - spécifie quelle correspondance sera utilisée (la valeur 0 correspond à des valeurs aléatoires / tout nombre positif N signifie que la valeur N / match doit être utilisée avec le contrôleur ForEach)
- Par défaut - la valeur par défaut qui sera stockée dans la variable en cas de non-correspondance est stockée dans la variable.

La case à cocher «Appliquer à» traite des exemples de demandes de ressources incorporées. Ce paramètre définit si une expression régulière sera appliquée aux exemples de résultats principaux ou à toutes les demandes, y compris les ressources incorporées. Il y a plusieurs options pour ce paramètre:

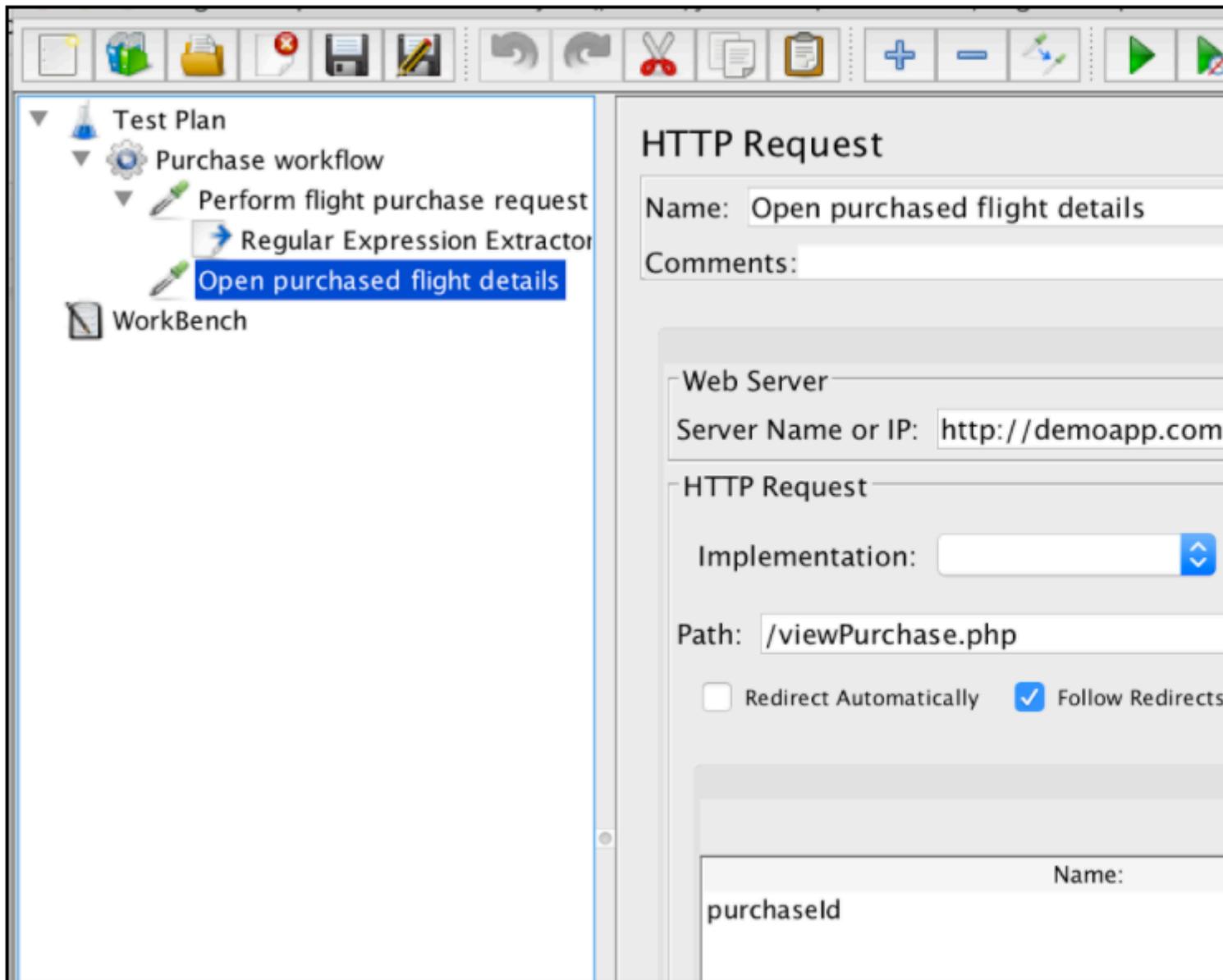
- Échantillon principal et sous-échantillons
- Échantillon principal seulement
- Sous-échantillons uniquement

- JMeter Variable - l'assertion est appliquée au contenu de la variable nommée, qui peut être remplie par une autre requête

La case à cocher «Champ à vérifier» permet de choisir le champ auquel l'expression régulière doit être appliquée. Presque tous les paramètres sont auto-descriptifs:

- Corps - le corps de la réponse, par exemple le contenu d'une page Web (à l'exclusion des en-têtes)
- Body (non échappé) - le corps de la réponse, avec tous les codes d'échappement HTML remplacés. Notez que les échappements HTML sont traités sans tenir compte du contexte, de sorte que certaines substitutions incorrectes peuvent être effectuées (* cette option a une forte incidence sur les performances)
- Body - Body as a Document - le texte extrait de différents types de documents via Apache Tika (* peut également avoir un impact sur les performances)
- Corps - En-têtes de demande - peut ne pas être présent pour les exemples non HTTP
- Corps - En-têtes de réponse - peut ne pas être présent pour les exemples non HTTP
- Corps - URL
- Code de réponse - par exemple 200
- Corps - Message de réponse - par exemple OK

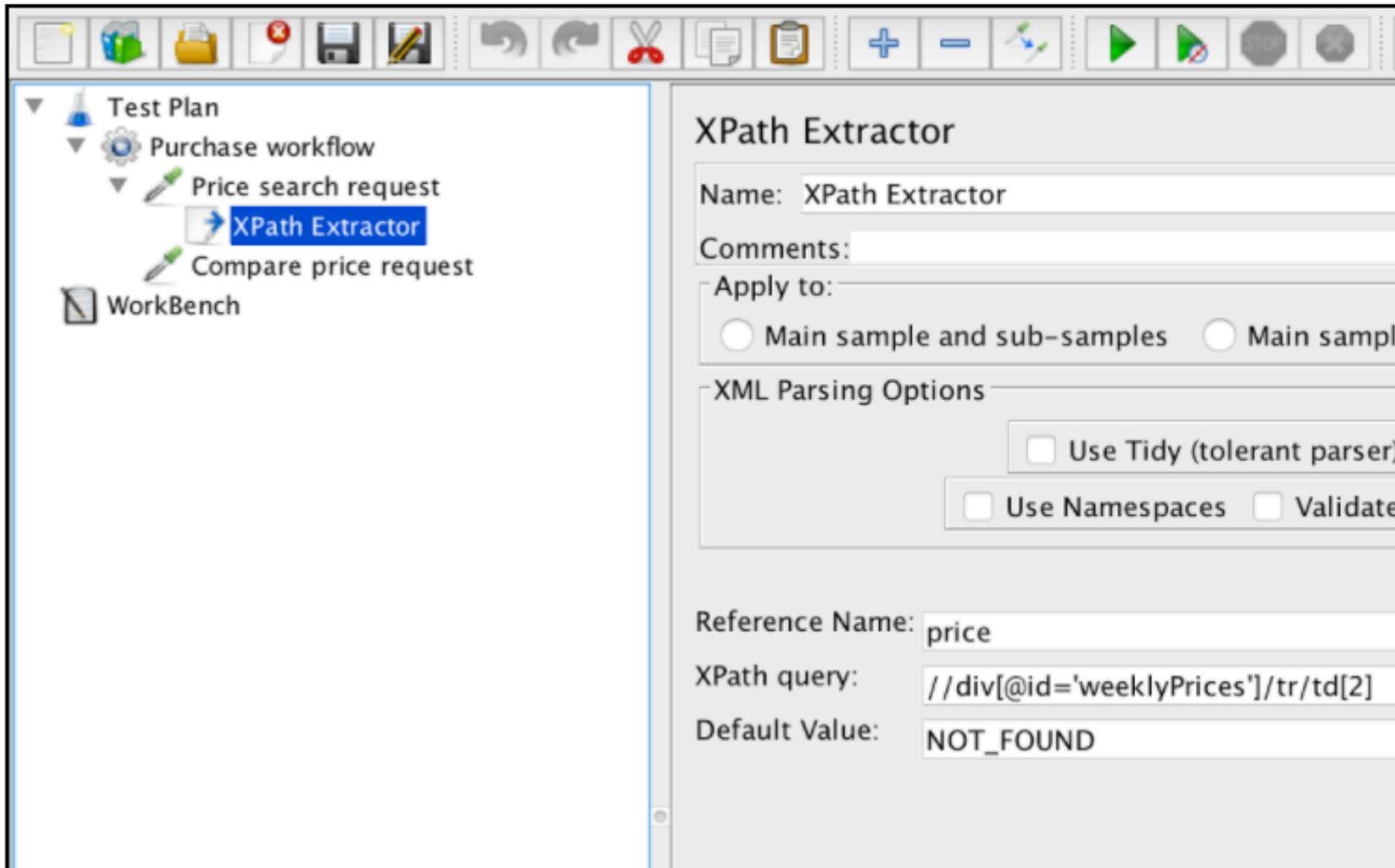
Une fois l'expression extraite, elle peut être utilisée dans les requêtes suivantes à l'aide de la variable \$ {purchaseld}.



Cette table contient toutes les contractions prises en charge par les expressions régulières JMeter:

chars	action
abc...	Match that character (metacharacter)
\\.*...	Match that metacharacter following
" "	De-meta any chars inside quotes
\t,\n,\r,\f	tab, newline, return, form feed
.	Match any character
[]	Character class
[^]	Inverse Character class
[-]	Character ranges
\w	Match a "word" character (alphanumeric)
\W	Match a non-word character
\s	Match a whitespace character
\S	Match a non-whitespace character
\d	Match a digit character
\D	Match a non-digit character
anchors	action
^	Match the beginning of the line
\$	Match the end of the line
\b	Match a word boundary
\B	Match a non-(word boundary)

Ajouter -> Post-processeurs -> XPath Extractor



XPath Extractor contient plusieurs éléments de configuration communs qui sont mentionnés dans la section «Corrélation à l'aide de l'extracteur d'expression régulière». Cela inclut Nom, Appliquer à, Nom de référence, Correspondance n ° (depuis JMeter 3.2) et Valeur par défaut.

Il existe de nombreuses ressources Web avec des aide-mémoire et des éditeurs en ligne pour créer et tester votre xpath créé (comme [celui-ci](#)). Mais sur la base des exemples ci-dessous, nous pouvons trouver le moyen de créer les localisateurs de xpath les plus courants.

<code>//hr[@class='edge' and position()=1]</code>	every first
<code>//table[count(tr)=1 and count(tr/td)=2]</code>	all tables
<code>//div/form/parent::*</code>	all divs t
<code>./div/b</code>	a relative
<code>/html/body/div/*[preceding-sibling::h4]</code>	give me
<code>//tr/td[font[@class="head" and text()="TRACK"]]</code>	all td tha "TRACK"
<code>./table/tr[last()]</code>	the last r
<code>//rdf:Seq/rdf:li/em:id</code>	using na
<code>//a/@href</code>	hrefs of a
<code>//*[count(*)=3]</code>	all nodes
<code>//var //acronym</code>	all vars a

Si vous voulez analyser HTML dans XHTML, nous devons cocher l'option «Use Tidy». Après avoir choisi le statut «Utiliser Tidy», il existe également des options supplémentaires:

Si 'Use Tidy' est coché:

- Quiet - définit le drapeau Tidy Quiet
- Signaler les erreurs - si une erreur de rangement se produit, définissez l'assertion en conséquence
- Show Warnings (Afficher les avertissements): définit l'option d'avertissement Show Show (Tidy show)

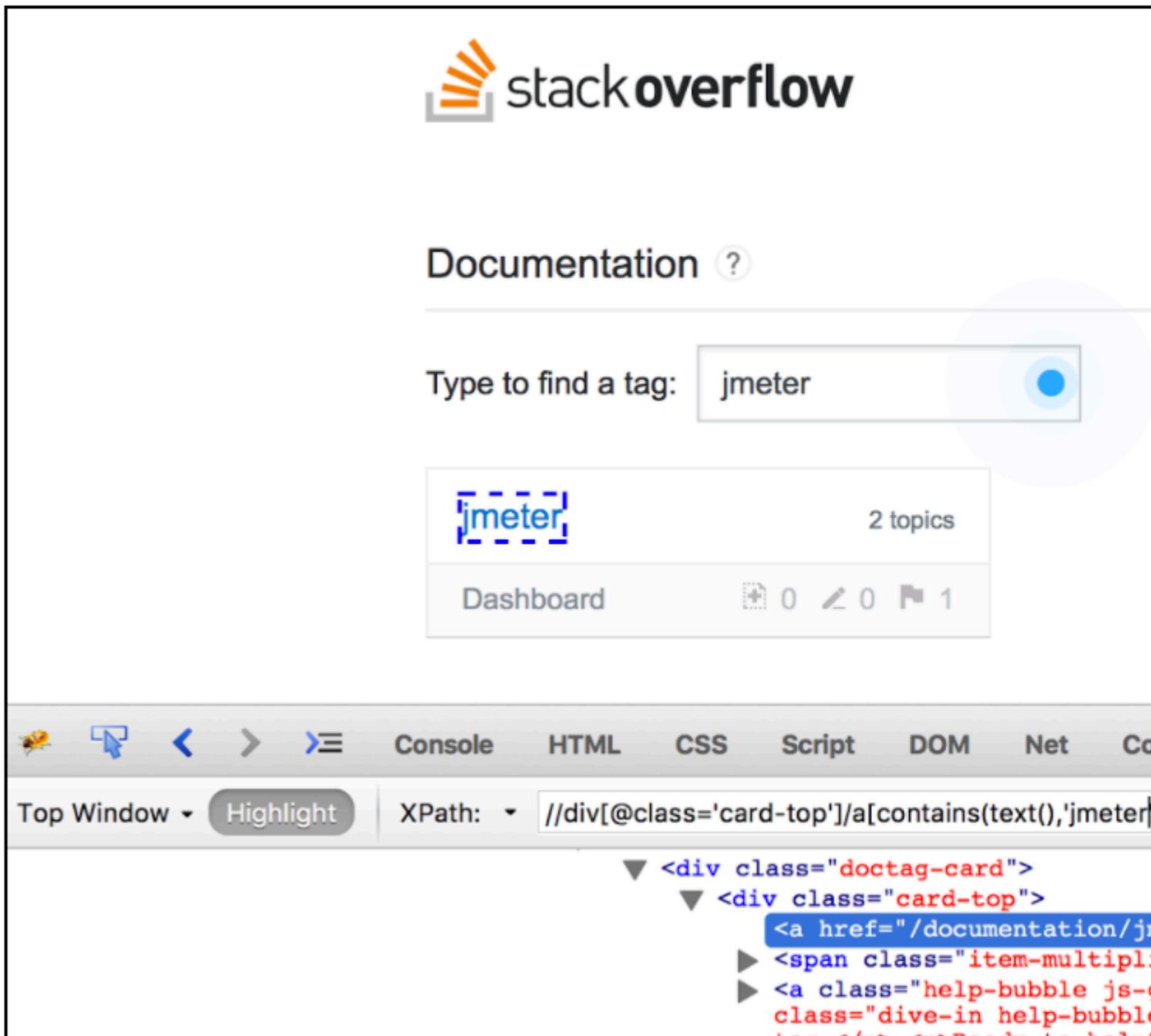
Si 'Use Tidy' est décoché:

- Utiliser les espaces de noms - si coché, l'analyseur XML utilisera la résolution d'espace de noms

- Valider XML - vérifie le document par rapport à son schéma spécifié
- Ignorer les espaces - ignore les espaces blancs
- Récupérer des DTD externes - si cette option est sélectionnée, les DTD externes sont récupérées

"Renvoyer l'intégralité du fragment XPath au lieu du contenu du texte" est auto-descriptif et doit être utilisé si vous souhaitez non seulement renvoyer la valeur xpath, mais également la valeur dans son localisateur xpath. Cela peut être utile pour les besoins de débogage.

Il convient également de mentionner qu'il existe une liste de plug-ins de navigateur très pratiques pour tester les localisateurs XPath. Pour Firefox, vous pouvez utiliser le plug-in « [Firebug](#) », tandis que pour Chrome, « [XPath Helper](#) » est l'outil le plus pratique.



Corrélation à l'aide de l'extracteur CSS / JQuery dans JMeter

L'extracteur CSS / JQuery permet d'extraire des valeurs d'une réponse du serveur en utilisant une syntaxe de sélecteur CSS / JQuery, qui aurait pu être difficile à écrire en utilisant une expression régulière. En tant que post-processeur, cet élément doit être exécuté pour extraire les nœuds, les valeurs de texte ou d'attribut requis d'un échantillonneur de requête et pour stocker le résultat dans la variable donnée. Ce composant est très similaire à XPath Extractor. Le choix entre CSS, JQuery ou XPath dépend généralement des préférences de l'utilisateur, mais il convient de mentionner que XPath ou JQuery peuvent traverser et traverser le DOM, tandis que CSS ne peut pas parcourir le DOM. Supposons que nous souhaitons extraire tous les sujets de la documentation Stack Overflow liée à Java. Vous pouvez utiliser le plug-in [Firebug](#) pour tester vos sélecteurs CSS / JQuery dans Firefox ou le [testeur de sélecteur CSS](#) dans Chrome.

Documentation ?

Type to find a tag:

Java Language

153 topics

Dashboard

 16  0  4

JavaScript

101 topics

Dashboard

 0  0 

java-ee

6 topics

java.util.scanner

1 topic



Top Window ▾

Highlight

Console

HTML

CSS

Script

DOM

Net

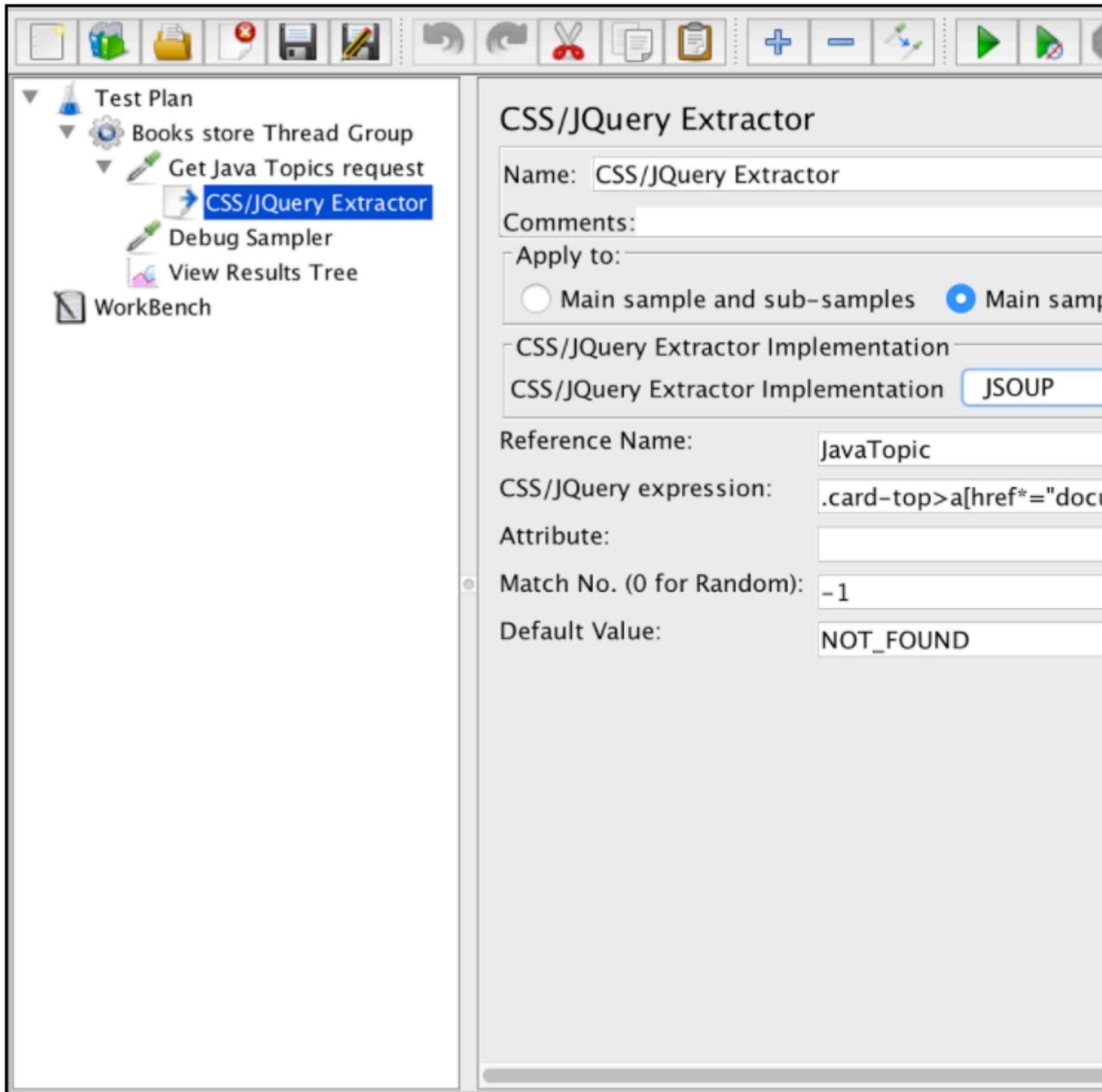
C

CSS: ▾

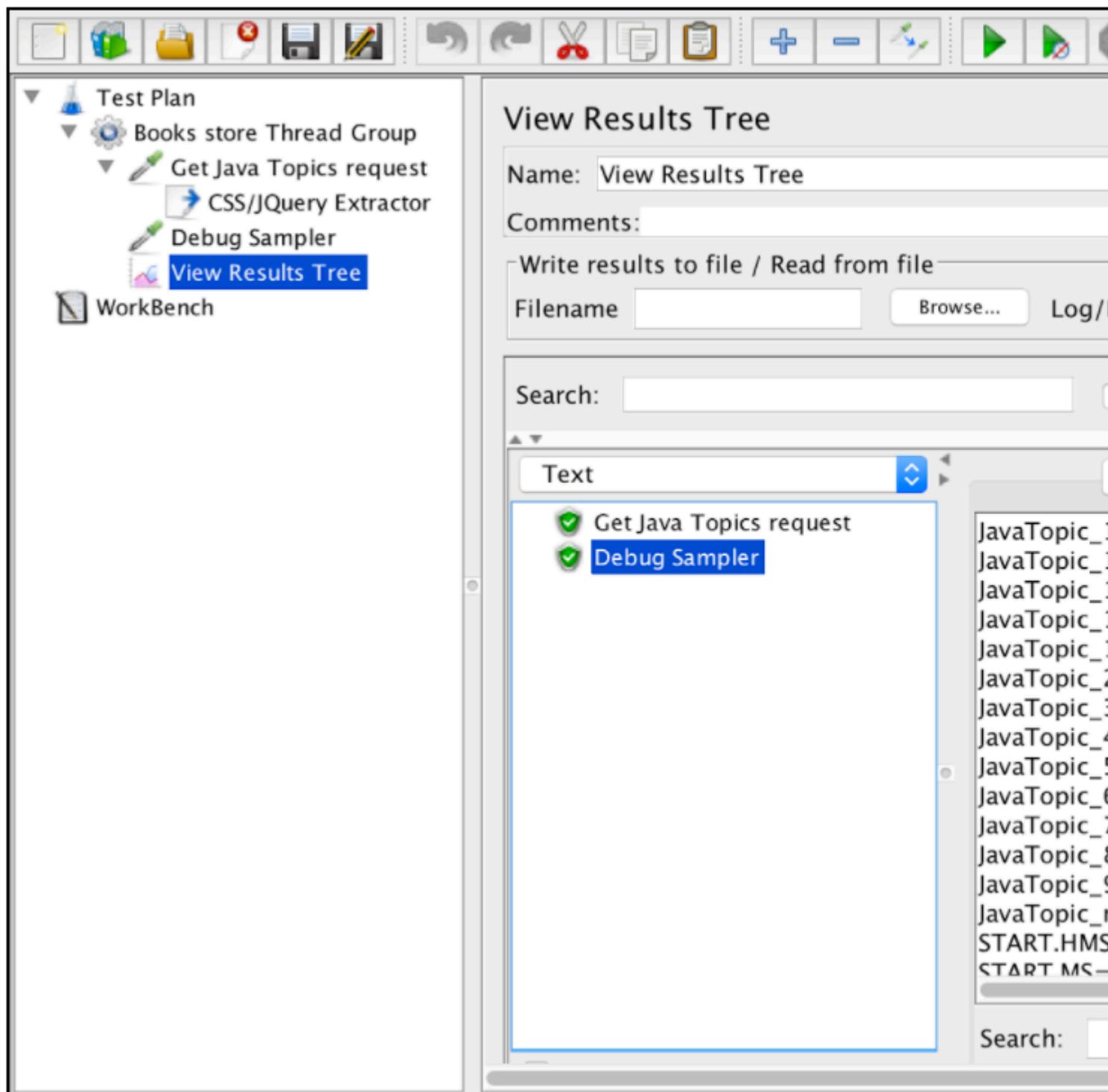
.card-top>a[href*="documentation"]

```
<div class="doctag-cards-container">
  <table class="doctag-cards">
    <tbody>
      <tr>
        <td>
          <div class="doctag-card">
            <div class="card-top">
              <a href="/documentation/j...>
                <span class="item-multipl...>
                <a class="help-bubble js-...>
                  track="helpbubble.shown({
                    type="4" data-bubble-posi...
                  class="dive-in help-bubble
```

Pour utiliser ce composant, ouvrez le menu JMeter et: *Ajouter -> Post-processeurs -> Extracteur CSS / JQuery*



Presque tous les champs de cet extracteur sont similaires aux champs d'extraction de l'expression régulière, vous pouvez donc obtenir leur description à partir de cet exemple. Une différence est cependant le champ «Implémentation CSS / JQuery Extractor». Depuis JMeter 2.9, vous pouvez utiliser l'extracteur CSS / JQuery basé sur deux implémentations différentes: l'implémentation [jsoup](#) (description détaillée de sa syntaxe [ici](#)) ou le [Lagarto JODD](#) (la syntaxe détaillée peut être trouvée [ici](#)). Les deux implémentations sont presque identiques et ne présentent que de petites différences de syntaxe. Le choix entre eux est basé sur les préférences de l'utilisateur.

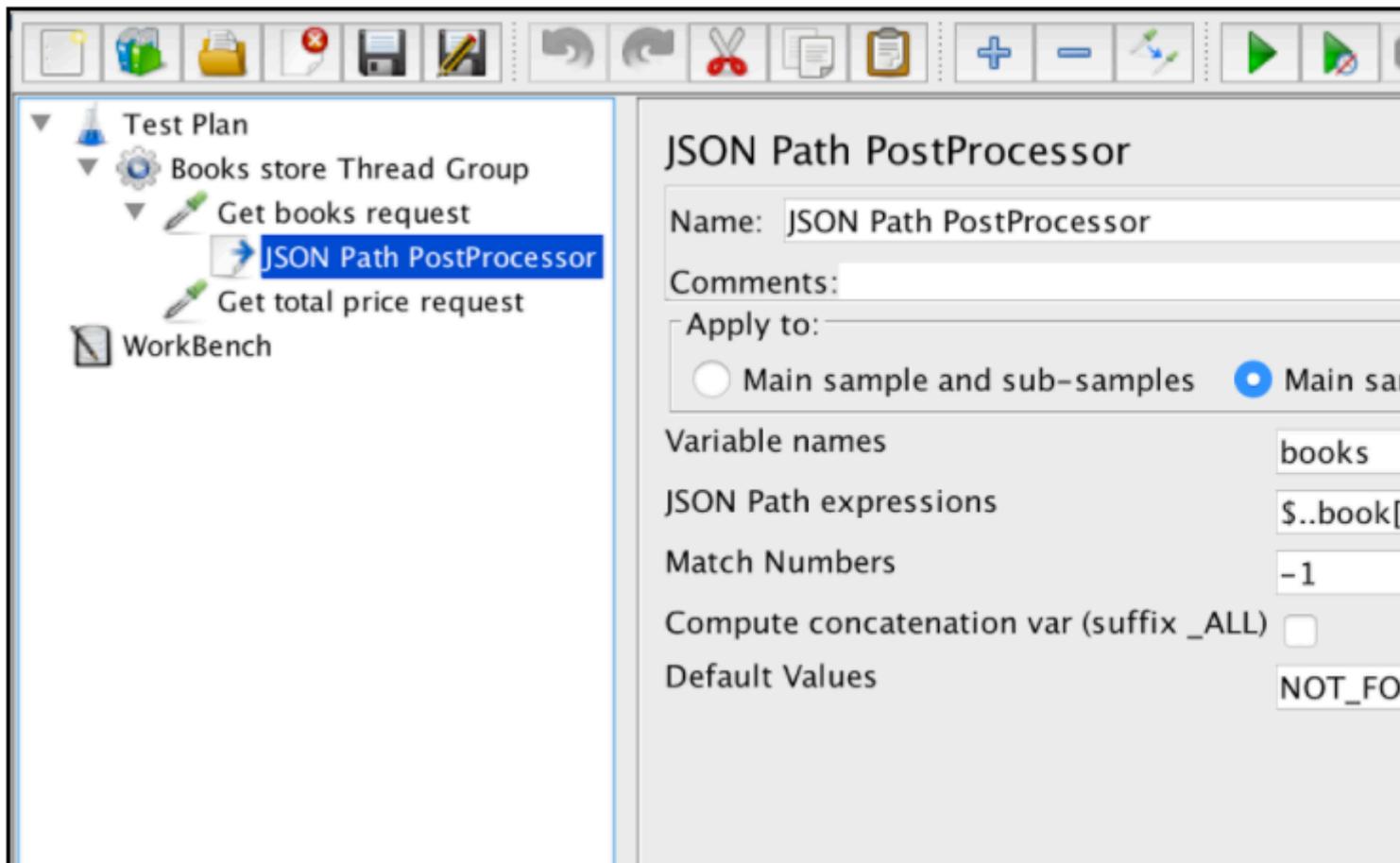


En fonction de la configuration mentionnée ci-dessus, nous pouvons extraire tous les sujets de la page demandée et vérifier les résultats extraits à l'aide de l'échantillonneur de débogage et de l'écouteur de vue des résultats.

Corrélation à l'aide de l'extracteur JSON

JSON est un format de données couramment utilisé dans les applications Web. JMeter JSON Extractor permet d'utiliser des expressions JSON Path pour extraire des valeurs de réponses basées sur JSON dans JMeter. Ce post-processeur doit être placé en tant qu'enfant de l'échantillonneur HTTP ou de tout autre échantillonneur ayant des réponses.

Pour utiliser ce composant, ouvrez le menu JMeter et: *Ajouter -> Post-processeurs -> Extracteur JSON.*



Le JSON Extractor est très similaire au Regular Expression Extractor. Presque tous les domaines principaux sont mentionnés dans cet exemple. Il n'y a qu'un seul paramètre JSON Extractor spécifique: 'Compute concatenation var'. Si plusieurs résultats sont trouvés, cet extracteur les concaténera en utilisant le séparateur ',' et en les stockant dans une variable nommée `_ALL`.

Supposons cette réponse du serveur avec JSON:

```
{
  "store": {
    "book": [
      { "category": "reference",
        "author": "Nigel Rees",
        "title": "Sayings of the Century",
        "price": 8.95
      },
      { "category": "fiction",
        "author": "Evelyn Waugh",
        "title": "Sword of Honour",
        "price": 12.99
      },
      { "category": "fiction",
        "author": "Herman Melville",
        "title": "Moby Dick",
        "isbn": "0-553-21311-3",
        "price": 8.99
      }
    ]
  }
}
```

```
    },
    { "category": "fiction",
      "author": "J. R. R. Tolkien",
      "title": "The Lord of the Rings",
      "isbn": "0-395-19395-8",
      "price": 22.99
    }
  ],
  "bicycle": {
    "color": "red",
    "price": 19.95
  }
}
```

Le tableau ci-dessous fournit un excellent exemple de différentes manières d'extraire des données d'un JSON spécifié:

<code>\$.store.book[*].author</code>	the author
<code>\$.author</code>	all authors
<code>\$.store.*</code>	all things red bicycle
<code>\$.store..price</code>	the price
<code>\$.book[2]</code>	the third
<code>\$.book[(@.length-1)]</code> <code>\$.book[-1:]</code>	the last
<code>\$.book[0,1]</code> <code>\$.book[:2]</code>	the first
<code>\$.book[?(@.isbn)]</code>	filter all
<code>\$.book[?(@.price<10)]</code>	filter all
<code>\$.*</code>	all Elements JSON structure

Grâce à ce [lien](#), vous pouvez trouver une description plus détaillée du format du chemin JSON, avec des exemples connexes.

Corrélation automatisée à l'aide du «SmartJMX» de BlazeMeter

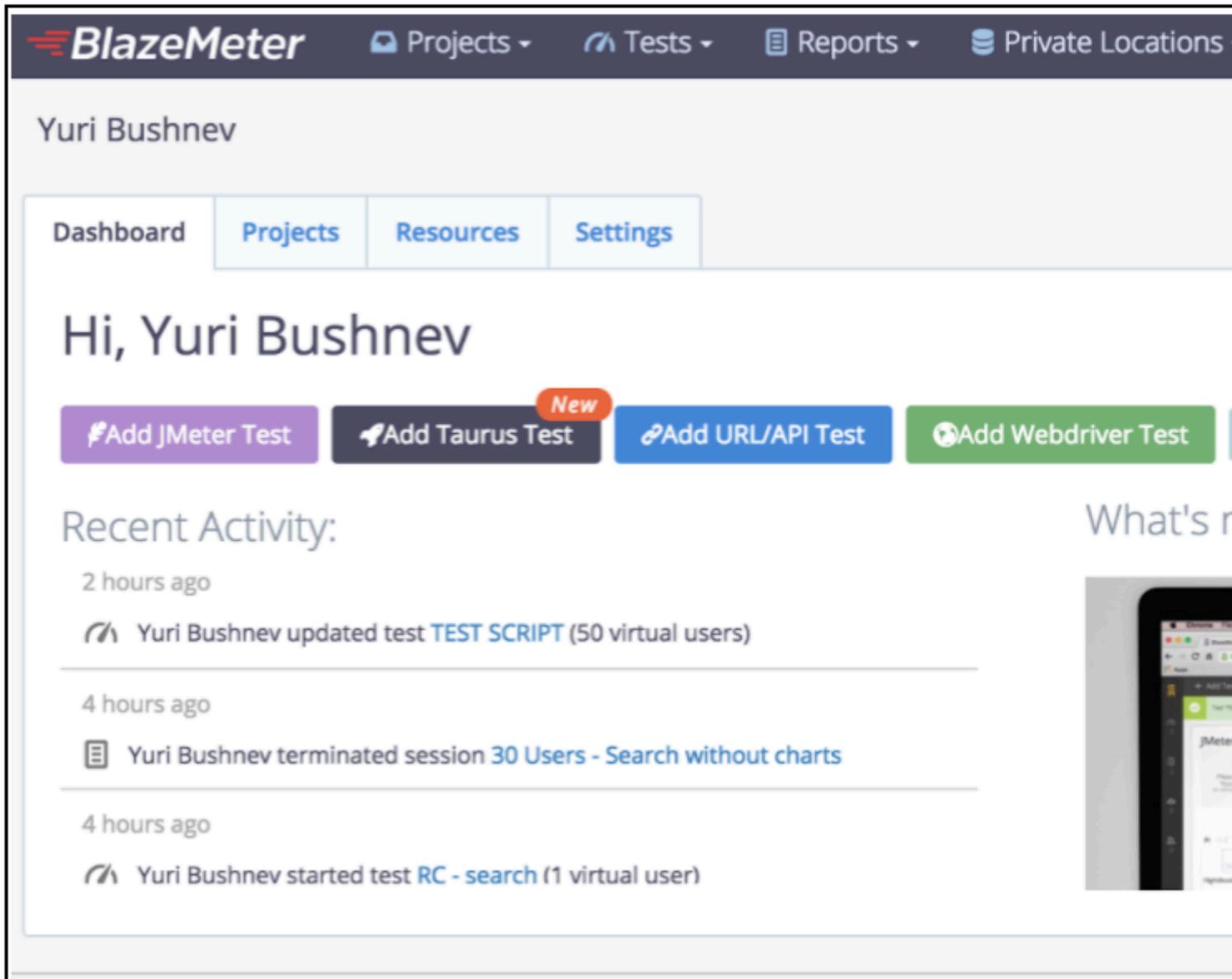
Lorsque vous écrivez manuellement vos scripts de performance, vous devez vous occuper de la corrélation. Mais il existe une autre option pour créer vos scripts - l'enregistrement de scripts d'automatisation. D'une part, l'approche manuelle aide vos scripts d'écriture structurés et vous pouvez ajouter tous les extracteurs requis en même temps. En revanche, cette approche prend

beaucoup de temps.

L'enregistrement de scripts d'automatisation est très simple et vous permet de faire le même travail, mais beaucoup plus rapidement. Mais si vous utilisez des méthodes d'enregistrement communes, les scripts seront peu structurés et nécessiteront généralement l'ajout de paramètres supplémentaires. La fonctionnalité «Smart JMX» de l'enregistreur BlazeMeter combine les avantages des deux méthodes. Il peut être trouvé à ce lien: [

<https://a.blazemeter.com/app/recorder/index.html>][1]

Après l'enregistrement, allez à la section «Enregistreur».



Pour démarrer l'enregistrement de script, vous devez d'abord configurer le proxy de votre navigateur ([couvert ici](#)), mais cette fois, vous devriez obtenir un hôte proxy et un port fournis par l'enregistreur BlazeMeter.

The image shows a screenshot of the Recorder application interface. On the left, a dark sidebar contains the Recorder logo and a 'Proxy Setup' section. The proxy address 'prox1.blazemeter.com:63536' is highlighted with a red box. Below it, there is a field for the certificate URL 'http://bz' and a warning box stating 'Proxy will be active for 24 hours. Don't forget to terminate the proxy when you've finished recording!'. At the bottom of the sidebar is a red button labeled 'TERMINATE PROXY'. On the right, a light gray header has a search bar. Below it, the 'Recently Captured Requests' section is visible, with a red arrow pointing to a placeholder '[proxy host]:[port]'.

Une fois le navigateur configuré, vous pouvez poursuivre l'enregistrement du script en appuyant sur le bouton rouge en bas. Vous pouvez maintenant accéder à l'application testée et effectuer des flux de travail utilisateur pour l'enregistrement.

The image shows the Recorder application interface. On the left, the 'Proxy Setup' section displays the current proxy as 'prox1.blazemeter.com:63536' and the target URL as 'http://bz'. A warning box states: 'Proxy will be active for 24 hours. Don't forget to terminate the proxy when you've finished recording!'. A red 'TERMINATE PROXY' button is at the bottom. On the right, the 'Recently Captured Requests' section shows a list of requests with their timestamps, methods, and URLs.

Timestamp	Method	URL
[20:45:39]	POST	http://blazedemo.com/confirmation.php
[20:45:36]	POST	http://blazedemo.com/purchase.php
[20:45:31]	GET	http://blazedemo.com/favicon.ico
[20:45:29]	GET	http://blazedemo.com/assets/bootstrap.min.js
[20:45:29]	GET	http://blazedemo.com/assets/bootstrap-table.js
[20:45:29]	GET	http://ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js
[20:45:29]	GET	http://blazedemo.com/assets/bootstrap-table.js
[20:45:29]	GET	http://blazedemo.com/assets/bootstrap.min.js
[20:45:29]	GET	http://blazedemo.com/reserve.php

Une fois le script enregistré, vous pouvez exporter les résultats dans un fichier JMX «SMART». Un fichier jmx exporté contient une liste d'options qui vous permettent de configurer votre script et de le paramétrer, sans efforts supplémentaires. L'une de ces améliorations est que le JMX «SMART» recherche automatiquement les candidats à la corrélation, le remplace par l'extracteur approprié et fournit un moyen facile de paramétrer davantage.

Lire [Corrélations Apache JMeter en ligne](https://riptutorial.com/fr/jmeter/topic/8978/correlations-apache-jmeter): <https://riptutorial.com/fr/jmeter/topic/8978/correlations-apache-jmeter>

Chapitre 4: Paramétrage d'Apache JMeter

Introduction

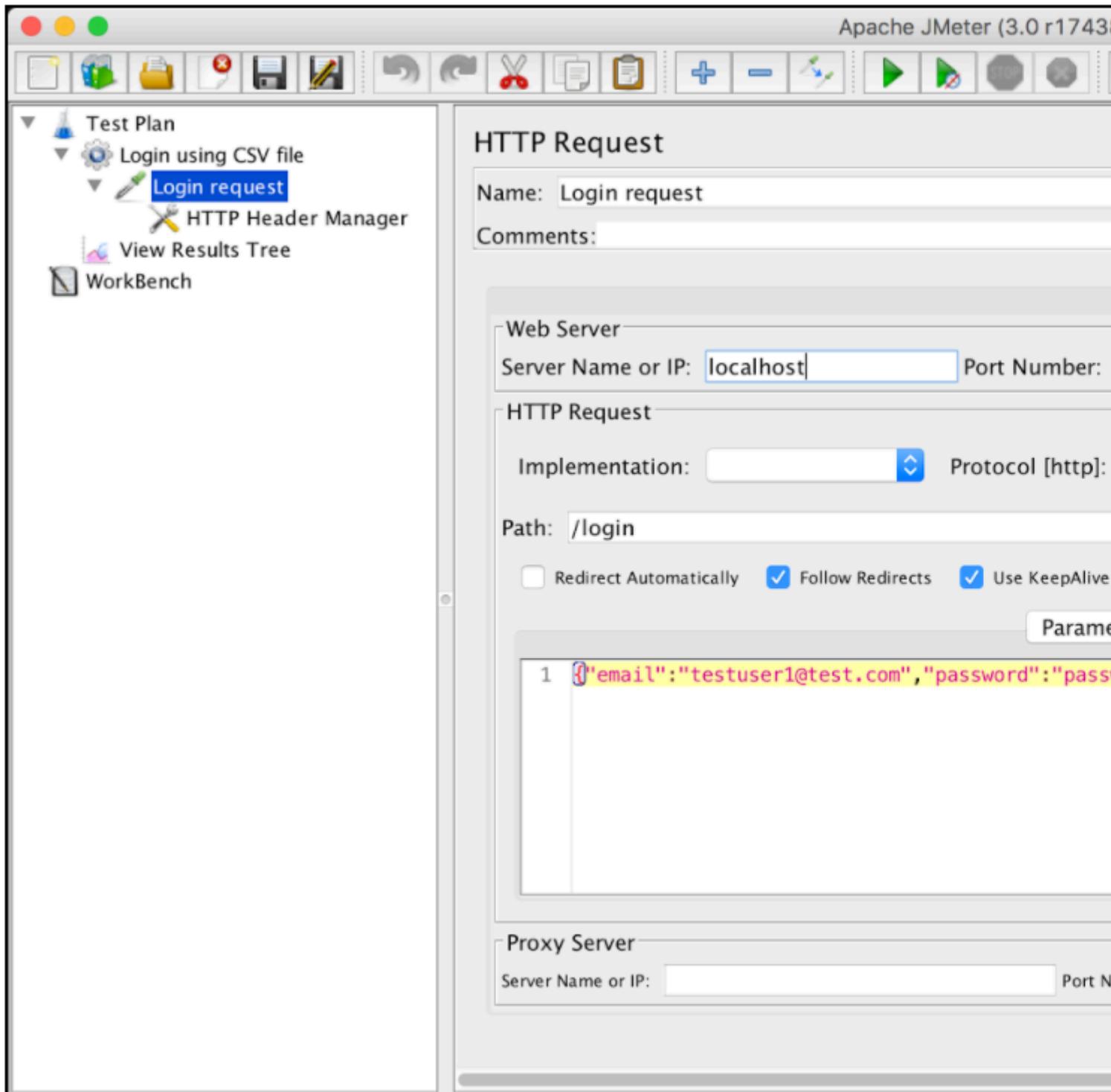
Le paramétrage est la création de différents jeux de données pour différents utilisateurs dans le même script de test. Par exemple, exécuter plusieurs utilisateurs avec des informations d'identification différentes dans le même script. Cela en fait l'un des principaux aspects de la création de tests de performance.

Exemples

Paramétrage à l'aide de fichiers externes

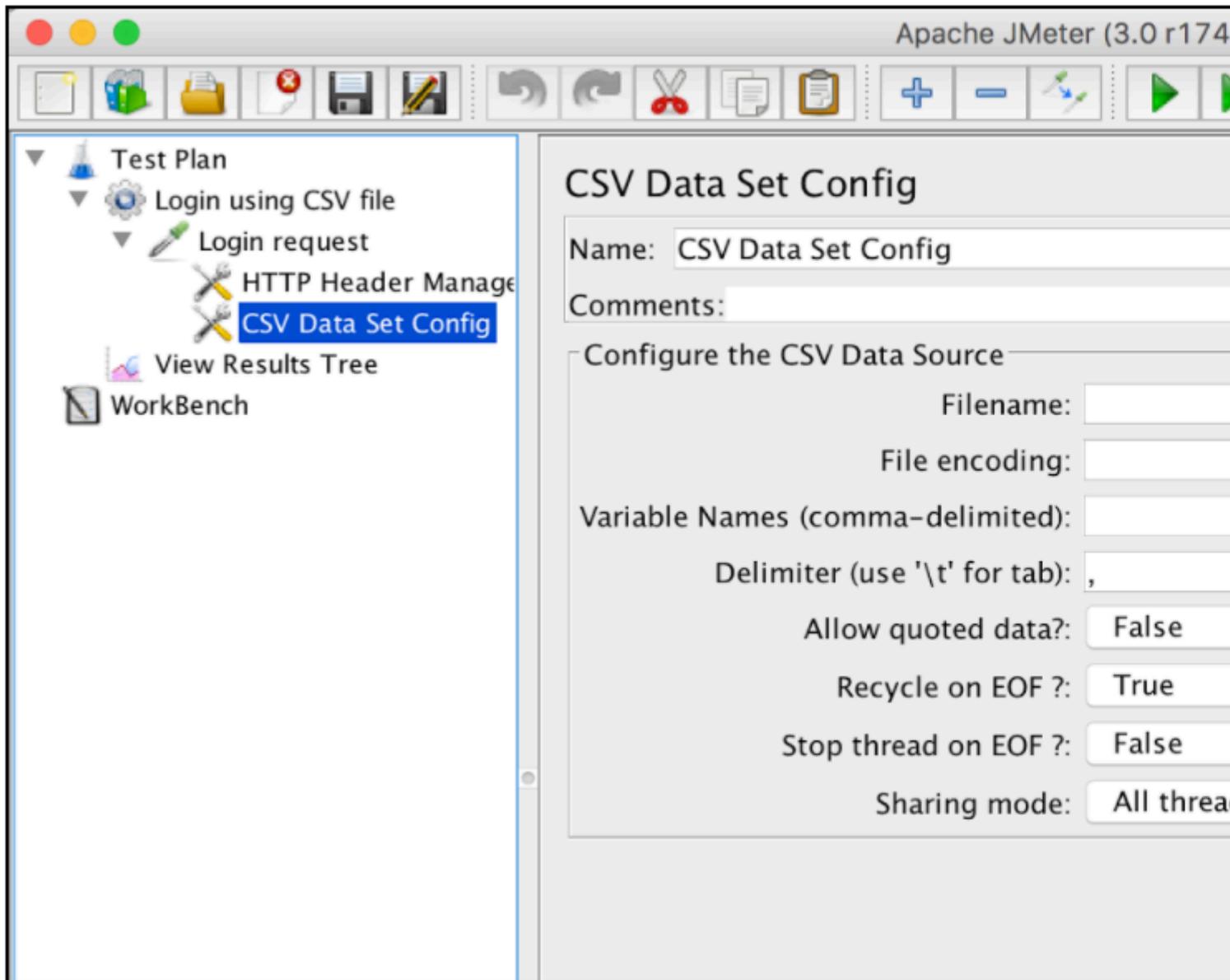
Une des façons les plus courantes de paramétrer vos scripts de performance consiste à utiliser un fichier CSV. Le meilleur exemple d'utilisation des fichiers d'entrée CSV est un processus de connexion. Si vous souhaitez tester votre application sur différents utilisateurs, vous devez fournir une liste des informations d'identification de l'utilisateur.

Supposons que nous ayons une demande de connexion qui fonctionne pour un utilisateur spécifique:



Nous pouvons facilement paramétrer cette requête en utilisant un fichier CSV externe et en exécutant le script sur différents utilisateurs. Pour ajouter une configuration de paramétrage CSV:

Faites un clic droit sur la demande de connexion -> Ajouter -> Élément de configuration -> Config.



Une brève explication des paramètres 'CSV Data Set Config':

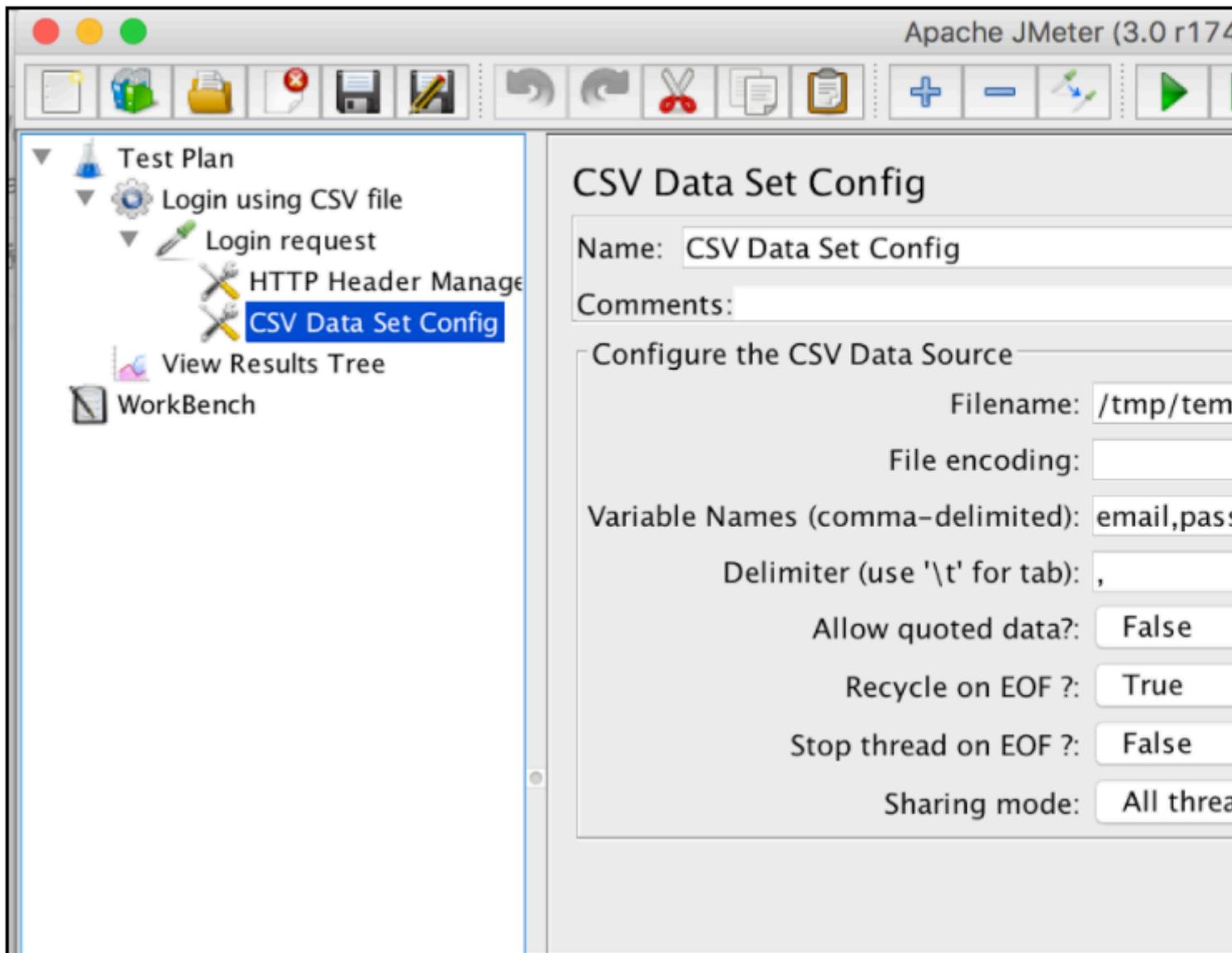
- Name - nom de l'élément tel qu'il sera utilisé dans l'arborescence JMeter
- Nom de fichier - nom du fichier d'entrée. Les noms de fichiers relatifs sont résolus en fonction du chemin du plan de test actif. Les noms de fichiers absolus sont également supportés
- File Encoding - encodage du fichier d'entrée, si ce n'est pas la plateforme par défaut
- Noms de variables - liste des noms de variables séparés qui seront utilisés comme conteneur pour les valeurs analysées. Si vide, la première ligne du fichier sera interprétée comme la liste des noms de variables
- Delimiter - délimiteur qui sera utilisé pour séparer les valeurs analysées du fichier d'entrée
- Autoriser les données cotées? - true si vous souhaitez ignorer les guillemets et permettre à de tels éléments de contenir un délimiteur.
- Recycler sur EOF? - true au cas où le plan de test du fichier devrait parcourir le fichier plus d'une fois. Il demandera à JMeter de déplacer le curseur au début du fichier
- Arrêter le fil sur EOF? - false en cas d'itération de boucle sur le fichier CDC et true si vous souhaitez arrêter le thread après avoir lu l'intégralité du fichier

- Mode de partage:
 - Tous les threads - le fichier est partagé entre tous les utilisateurs virtuels (par défaut)
 - Groupe de threads actuel - le fichier sera ouvert une fois pour chaque groupe de threads
 - Fil actuel - chaque fichier sera ouvert séparément pour chacun des threads
 - Identifiant - tous les threads partageant le même identifiant partagent également le même fichier

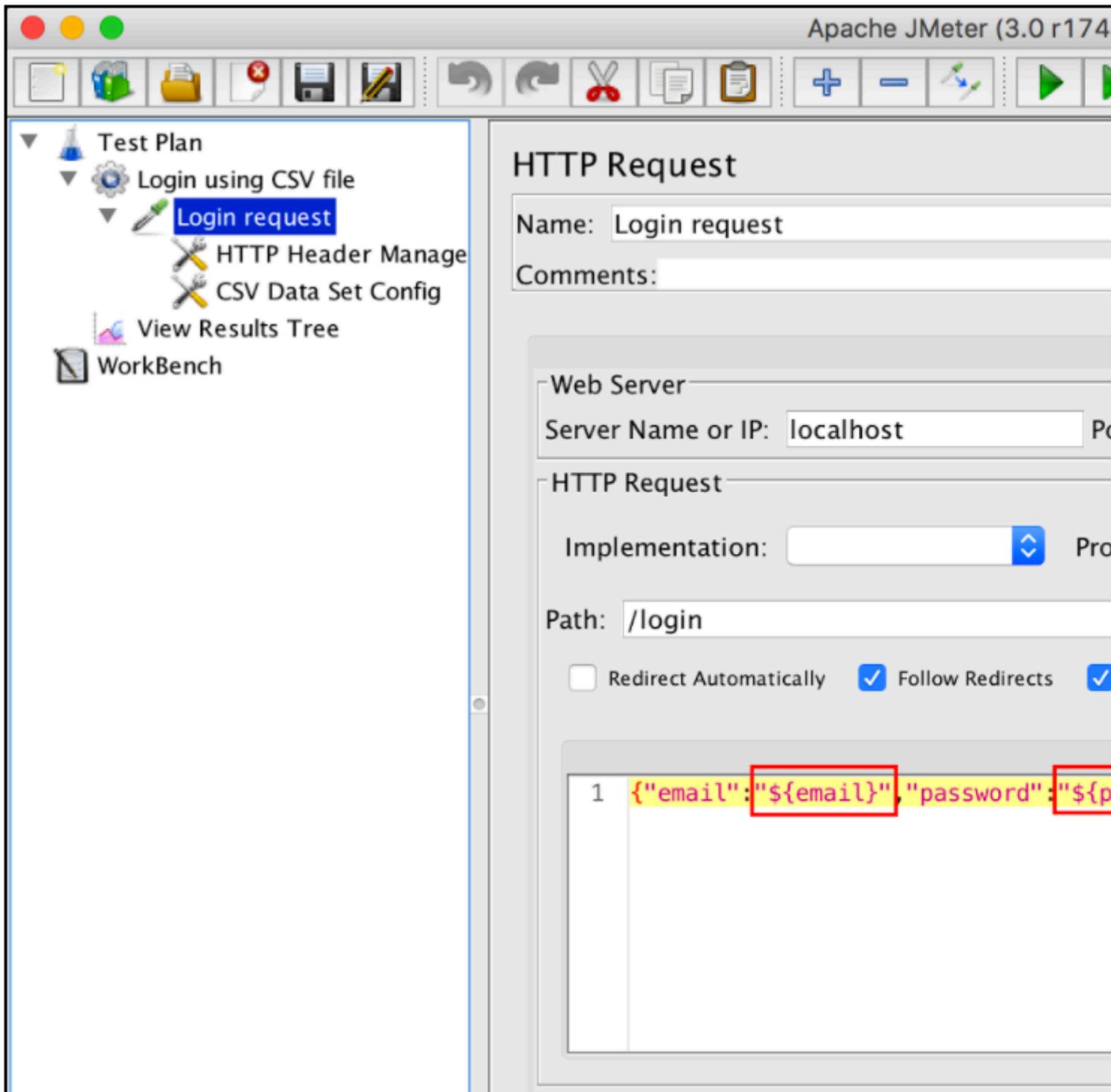
Créons un fichier csv contenant différents utilisateurs avec des noms et des mots de passe:

```
[→ tempTestFiles cat /tmp/tempTestFiles/TestU
testuser1@test.com,password1
testuser2@gmail.com,password2
testuser3@gmail.com,password3
testuser4@gmail.com,password4
testuser5@gmail.com,password5
testuser6@gmail.com,password6
testuser7@gmail.com,password7
testuser8@gmail.com,password8
testuser9@gmail.com,password9
testuser10@gmail.com,password10
```

Nous pouvons maintenant utiliser ce fichier avec la configuration CSV Data Set. Dans notre cas, il suffit d'ajouter les valeurs de configuration «Filename» et «Variables Names»:

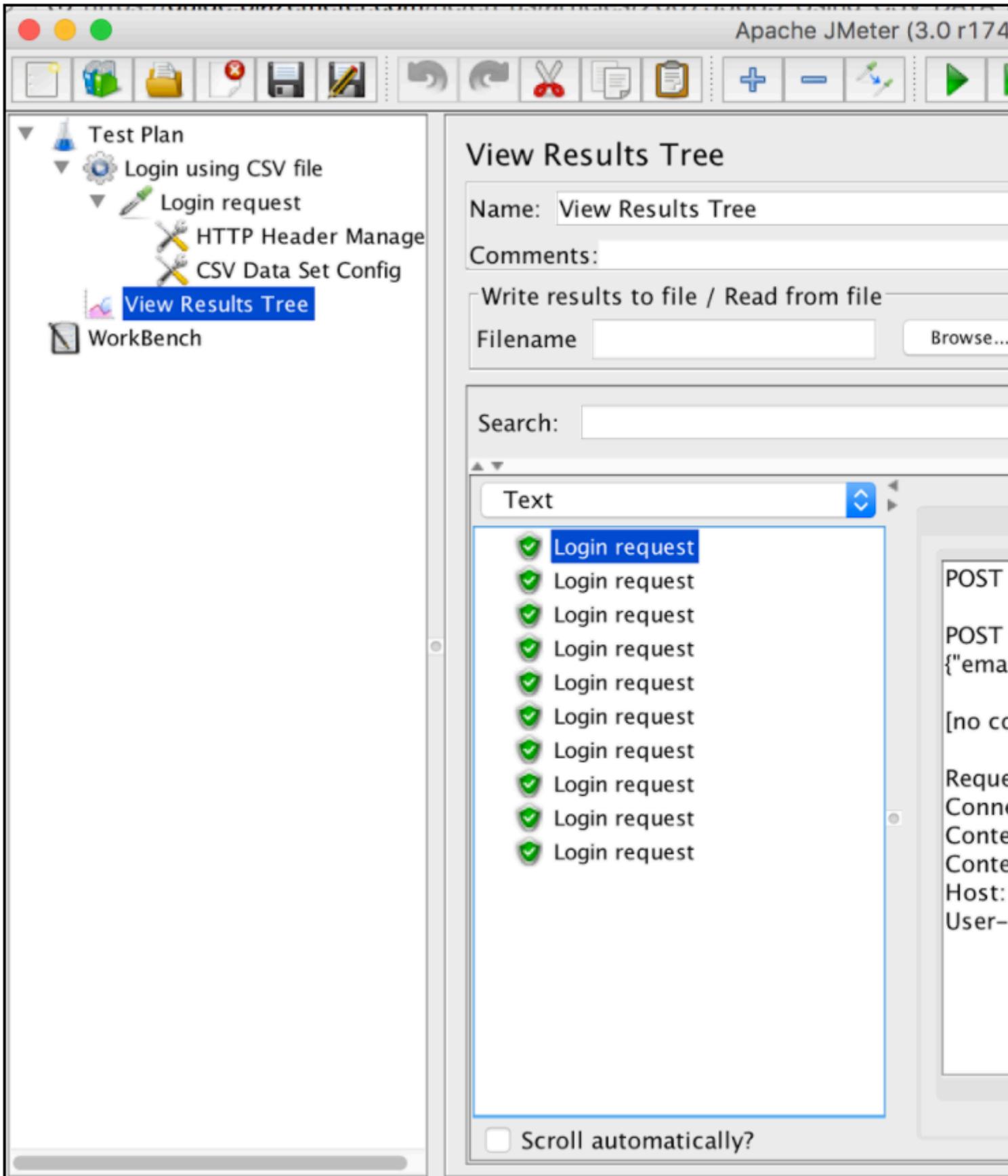


La dernière étape consiste à paramétrer la demande de connexion avec des variables CSV. Cela peut être fait en substituant les valeurs initiales aux variables appropriées du champ de configuration «Noms de variables» de la configuration de fichier CSV, comme ceci:

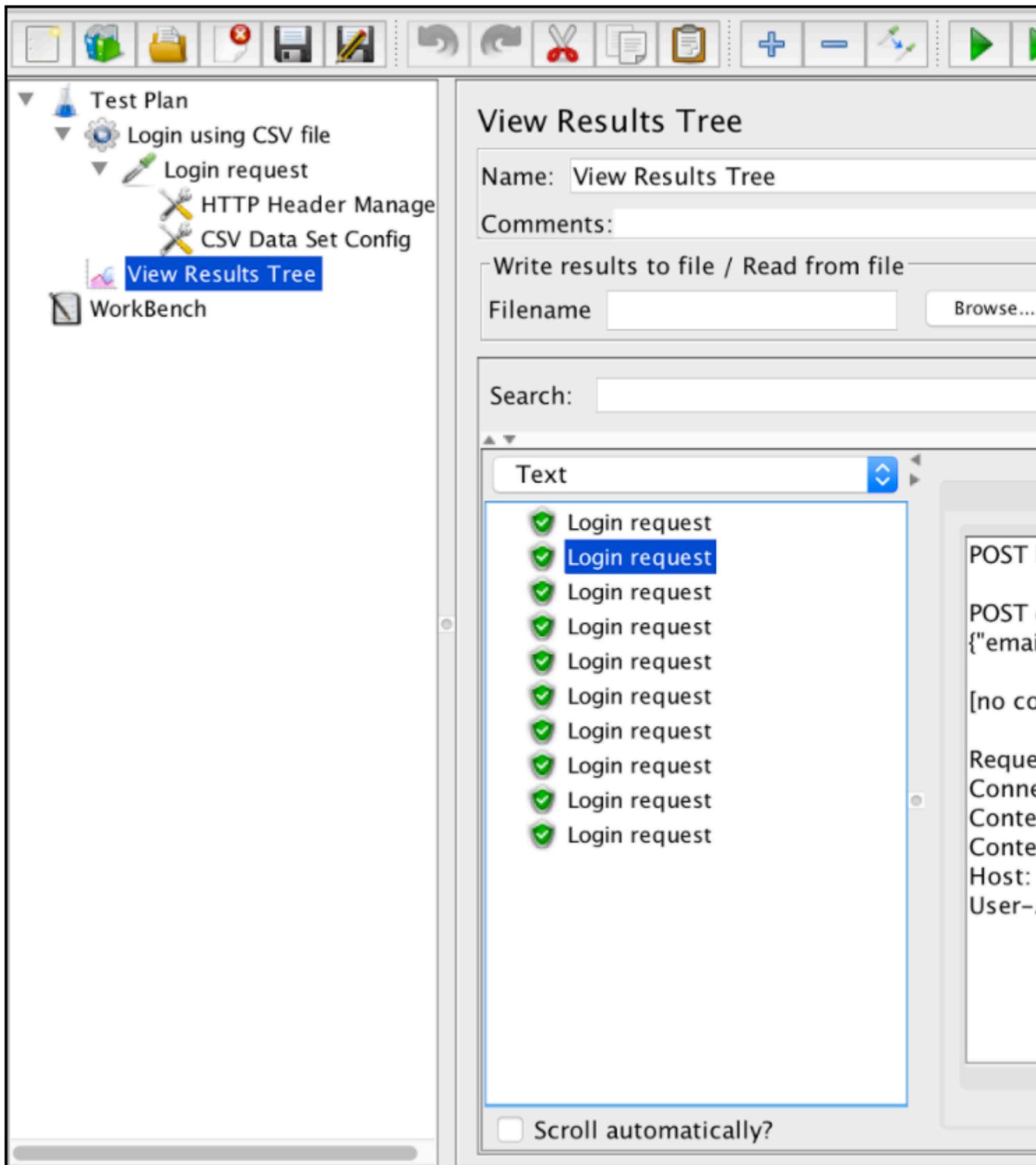


Si nous exécutons notre script de test maintenant, JMeter substituera ces variables aux valeurs du fichier 'TestUsers.csv'. Chaque utilisateur virtuel JMeter recevra des informations d'identification à partir de la ligne de fichier csv suivante.

La demande de connexion par le premier utilisateur:



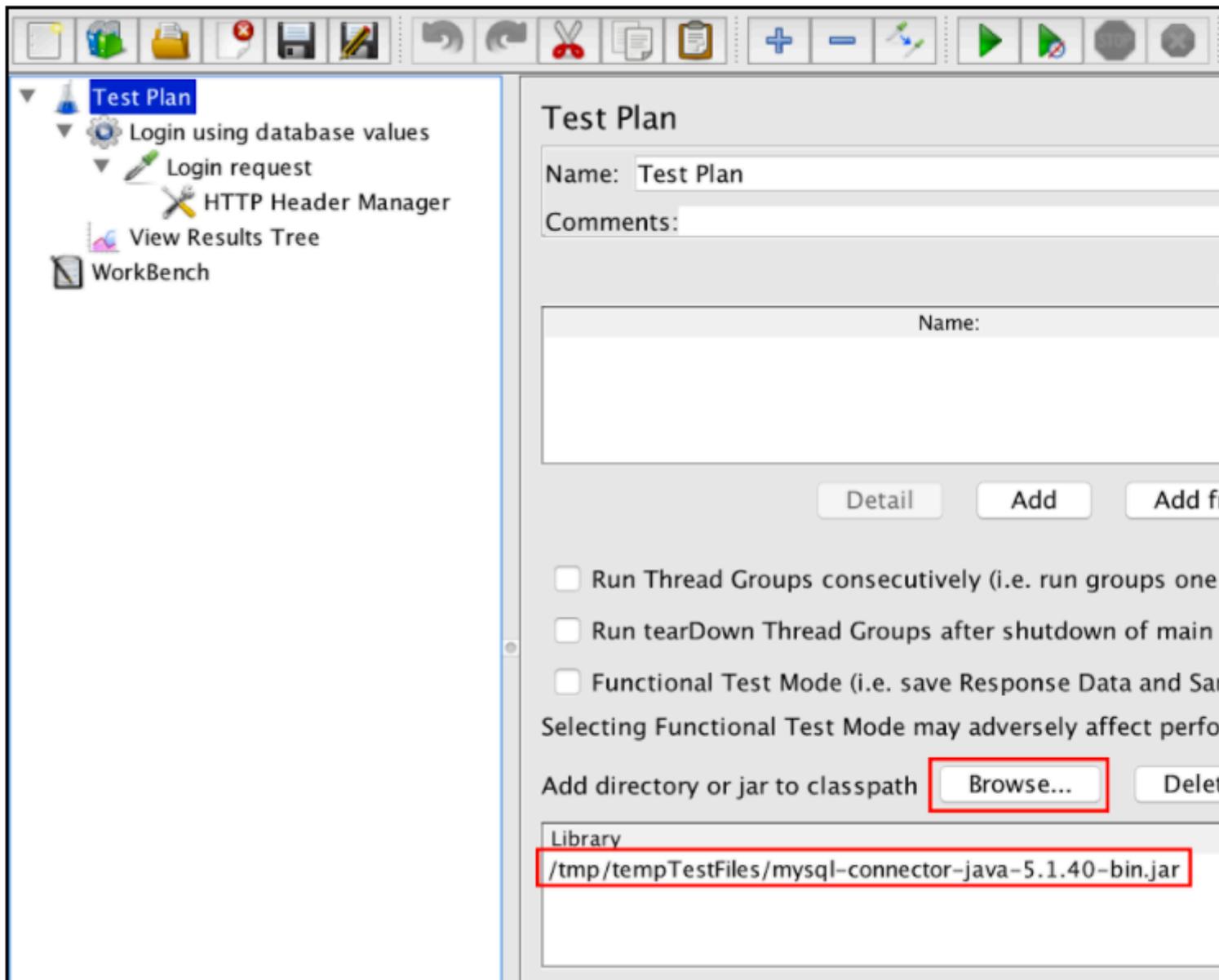
La demande de connexion par le deuxième utilisateur:



Paramétrage à l'aide de bases de données

Une autre façon de paramétrer vos scripts de performances consiste à utiliser les données de base de données via JDBC. JDBC est une interface de programmation d'application qui définit comment un client peut accéder à une base de données.

Tout d'abord, téléchargez le pilote JDBC dans votre base de données (reportez-vous au fournisseur de la base de données). Par exemple, le pilote mysql peut être trouvé ici. Ensuite, vous pouvez l'ajouter en ajoutant le fichier .jar au plan de test en utilisant le formulaire ci-dessous:



Mais il est préférable d'ajouter le fichier Jar dans le dossier lib et de redémarrer JMeter.

Après cela, configurez la connexion à la base de données en utilisant l'élément «JDBC Connection Configuration». Comme ceci: *Faites un clic droit sur Thread Group -> Add -> Config Element -> Configuration de la connexion JDBC*

JDBC Connection Configuration

Name:

Comments:

Variable Name Bound to Pool

Variable Name:

Connection Pool Configuration

Max Number of Connections:

Max Wait (ms):

Time Between Eviction Runs (ms):

Auto Commit:

Transaction Isolation:

Connection Validation by Pool

Test While Idle:

Soft Min Evictable Idle Time(ms):

Validation Query:

Database Connection Configuration

Database URL:

JDBC Driver class:

Username:

Password:

Paramètres de configuration de la connexion JDBC:

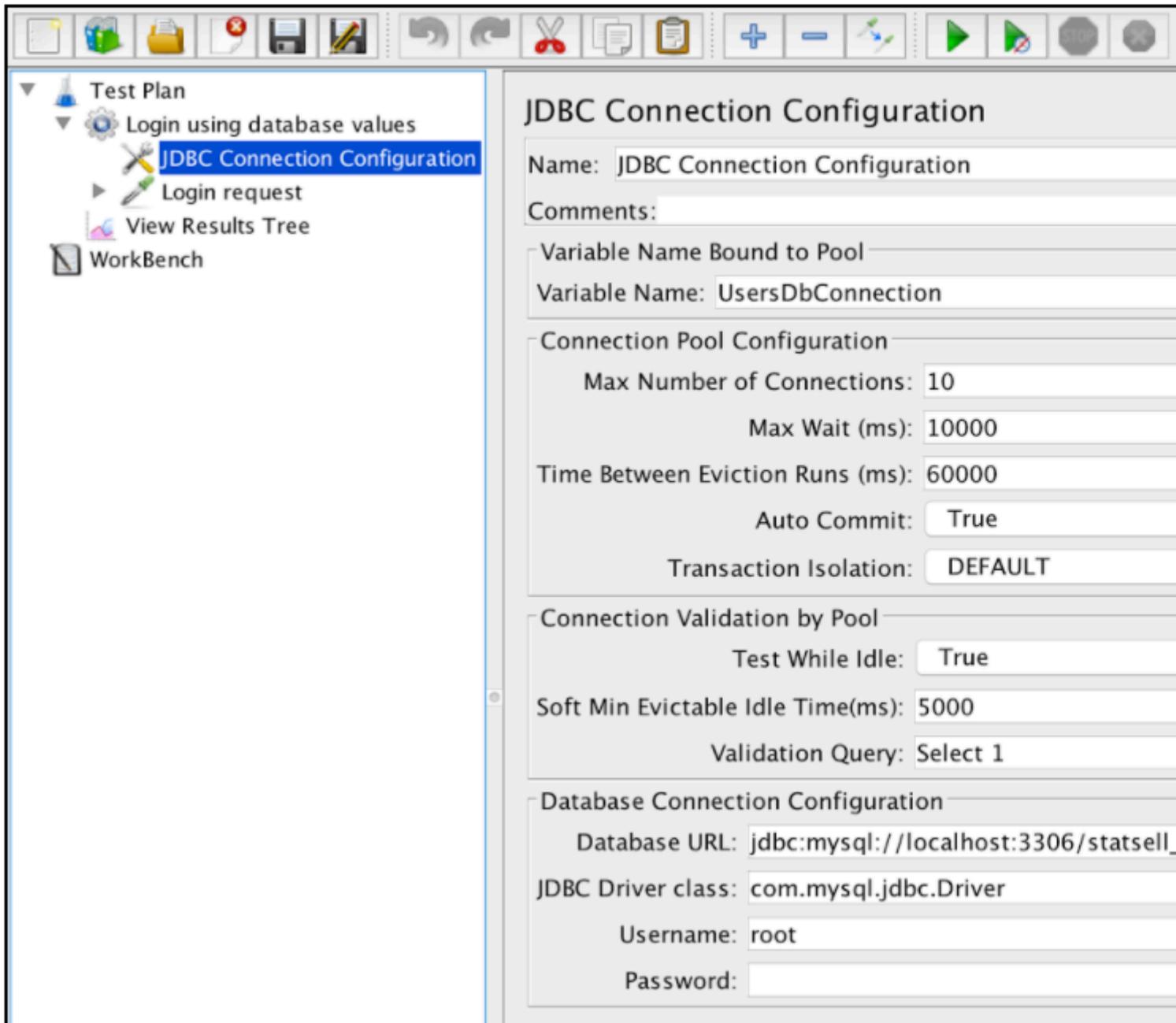
- Nom - nom de la configuration de connexion qui sera affichée dans l'arborescence du groupe de threads
- Nom de la variable - nom qui sera utilisé comme identifiant unique pour la connexion à la base de données (plusieurs connexions peuvent être utilisées et chacune sera associée à un nom différent)

- Nombre maximal de connexions: nombre maximal de connexions autorisées dans le pool de connexions. En cas de 0, chaque thread aura son propre pool avec une seule connexion.
- Max Wait (ms) - Le pool génère une erreur si le délai spécifié est dépassé lors de la connexion à la base de données
- Time Between Eviction Runs (ms) - nombre de millisecondes à mettre en pause entre les exécutions du thread expulsant les connexions inutilisées du pool de bases de données
- Auto Commit - yes pour activer la validation automatique pour les connexions db associées
- Test While Idle - Vérifiez les connexions inactives avant qu'une demande effective ne soit détectée. Plus de détails: [BasicDataSource.html # getTestWhileIdle](#)
- Soft Min Evictable Idle Time (ms) - période de temps pendant laquelle une connexion spécifiée peut être inactive dans le pool de bases de données avant de pouvoir être expulsée. Plus de détails: [BasicDataSource.html # getSoftMinEvictableIdleTimeMillis](#)
- Validation Query - requête de vérification de santé qui sera utilisée pour vérifier si la base de données répond toujours
- Database URL - Chaîne de connexion JDBC pour la base de données. Voir ici pour des exemples
- Classe de pilote JDBC: nom approprié de la classe de pilote (spécifique à chaque base de données). Par exemple, «com.mysql.jdbc.Driver» pour la base de données MySQL
- Nom d'utilisateur - nom d'utilisateur de la base de données
- Mot de passe - mot de passe de la base de données (sera stocké en clair dans le plan de test)

Dans notre cas, nous devons configurer les champs obligatoires uniquement:

- Nom de variable lié au pool.
- URL de la base de données
- Classe de pilote JDBC
- Nom d'utilisateur
- Mot de passe

Les autres champs de l'écran peuvent être laissés comme paramètres par défaut:



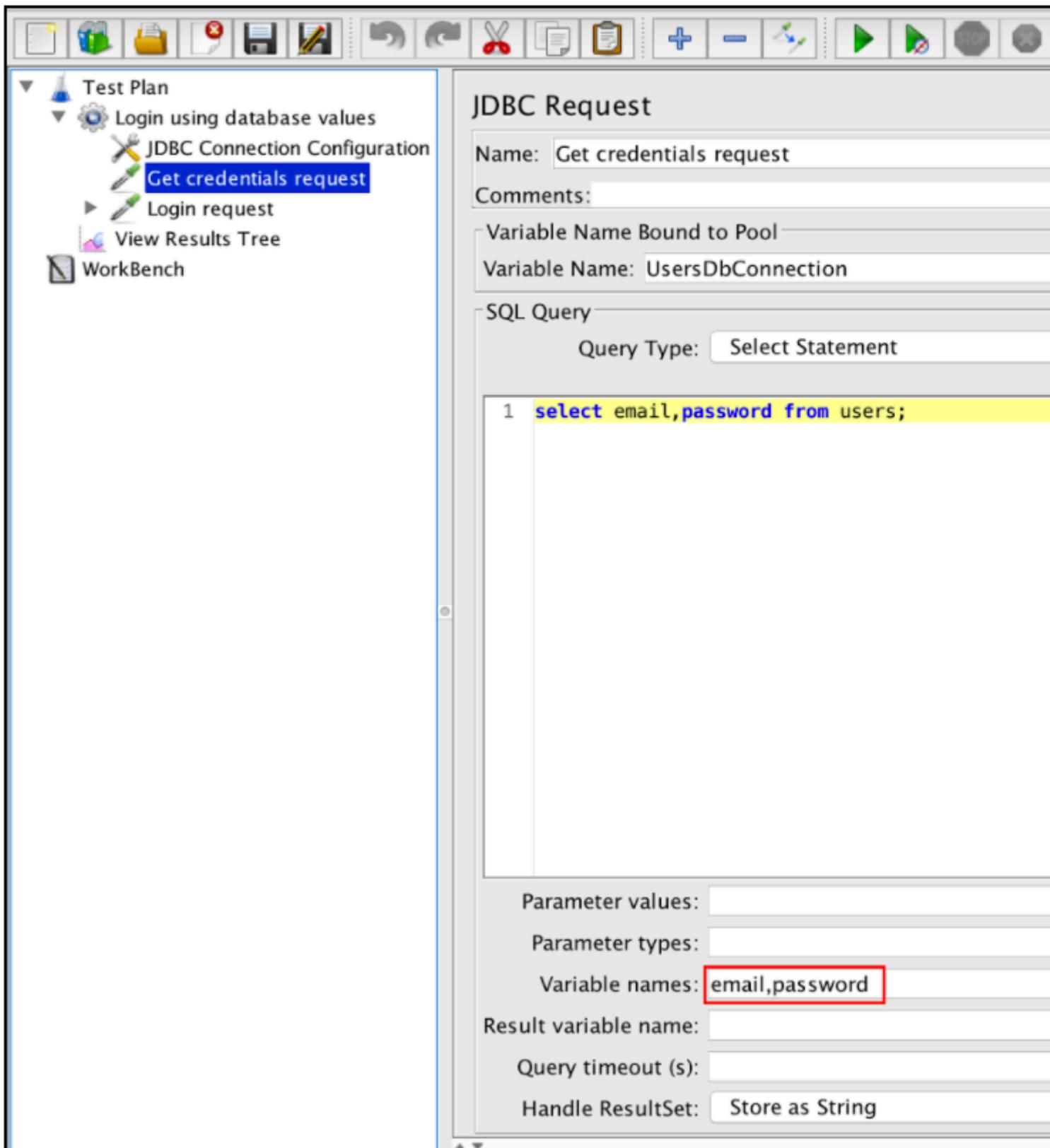
Supposons que nous stockions les informations d'identification de l'utilisateur de test dans la base de données:

The screenshot shows a database query tool interface. At the top, a text area contains the SQL query: `select email,password fro`. Below the query area is a toolbar with a gear icon, a dropdown menu labeled "Query Favorites", and another dropdown menu labeled "Query". The main area displays the results of the query in a table format with two columns: "email" and "password". The results consist of ten rows of test user data.

email	password
testuser1@test.com	password1
testuser2@gmail.com	password2
testuser3@gmail.com	password3
testuser4@gmail.com	password4
testuser5@gmail.com	password5
testuser6@gmail.com	password6
testuser7@gmail.com	password7
testuser8@gmail.com	password8
testuser9@gmail.com	password9
testuser10@gmail.com	password10

Maintenant, lorsque la connexion à la base de données est configurée, nous pouvons ajouter la demande JDBC elle-même et utiliser sa requête pour obtenir toutes les informations d'identification de la base de données: *Cliquez avec le bouton droit sur Thread Group -> Add -> Sample -> JDBC Request*

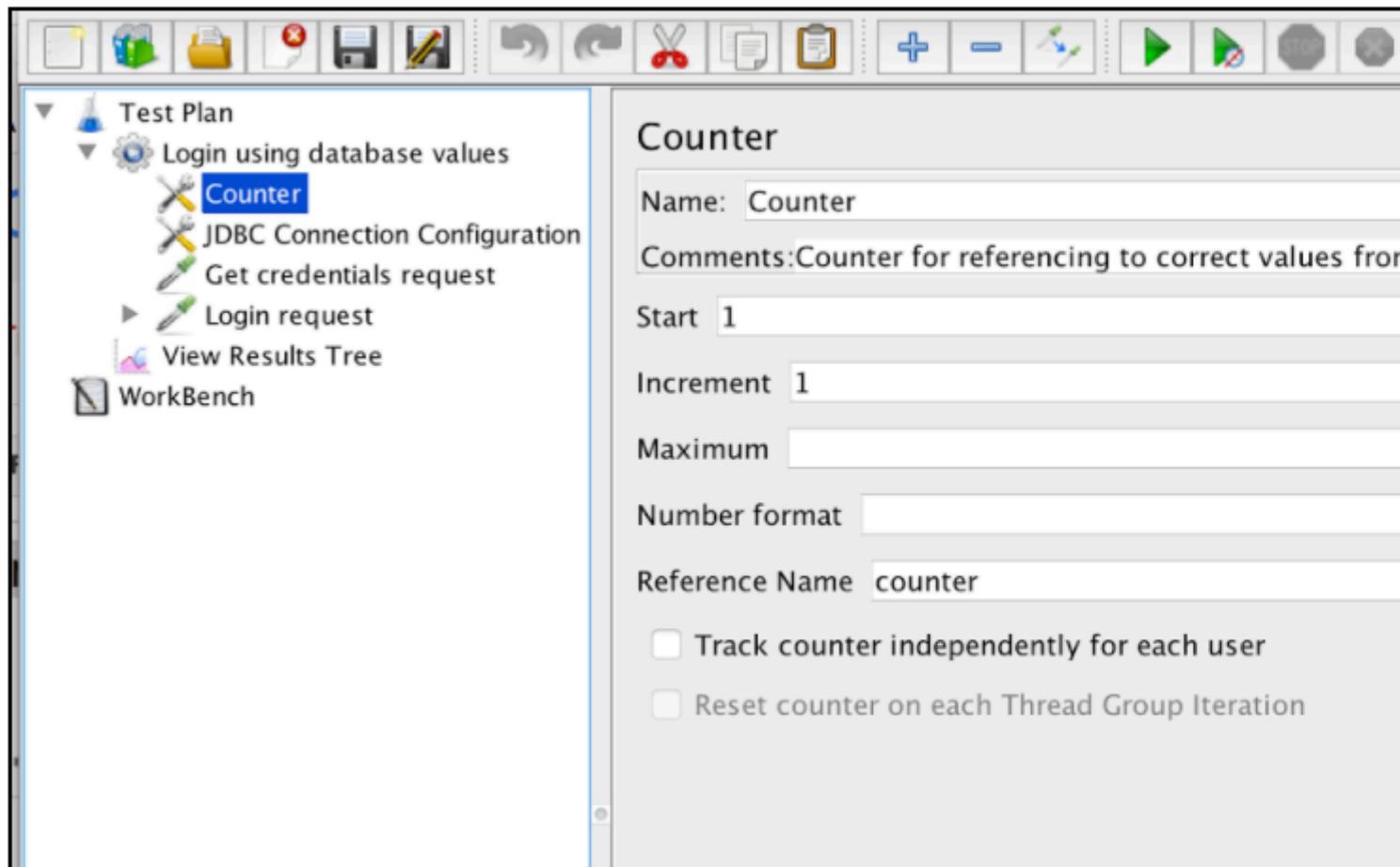
En utilisant la requête "Select Statement" et les "Noms de variables", nous pouvons analyser la réponse aux variables personnalisées.



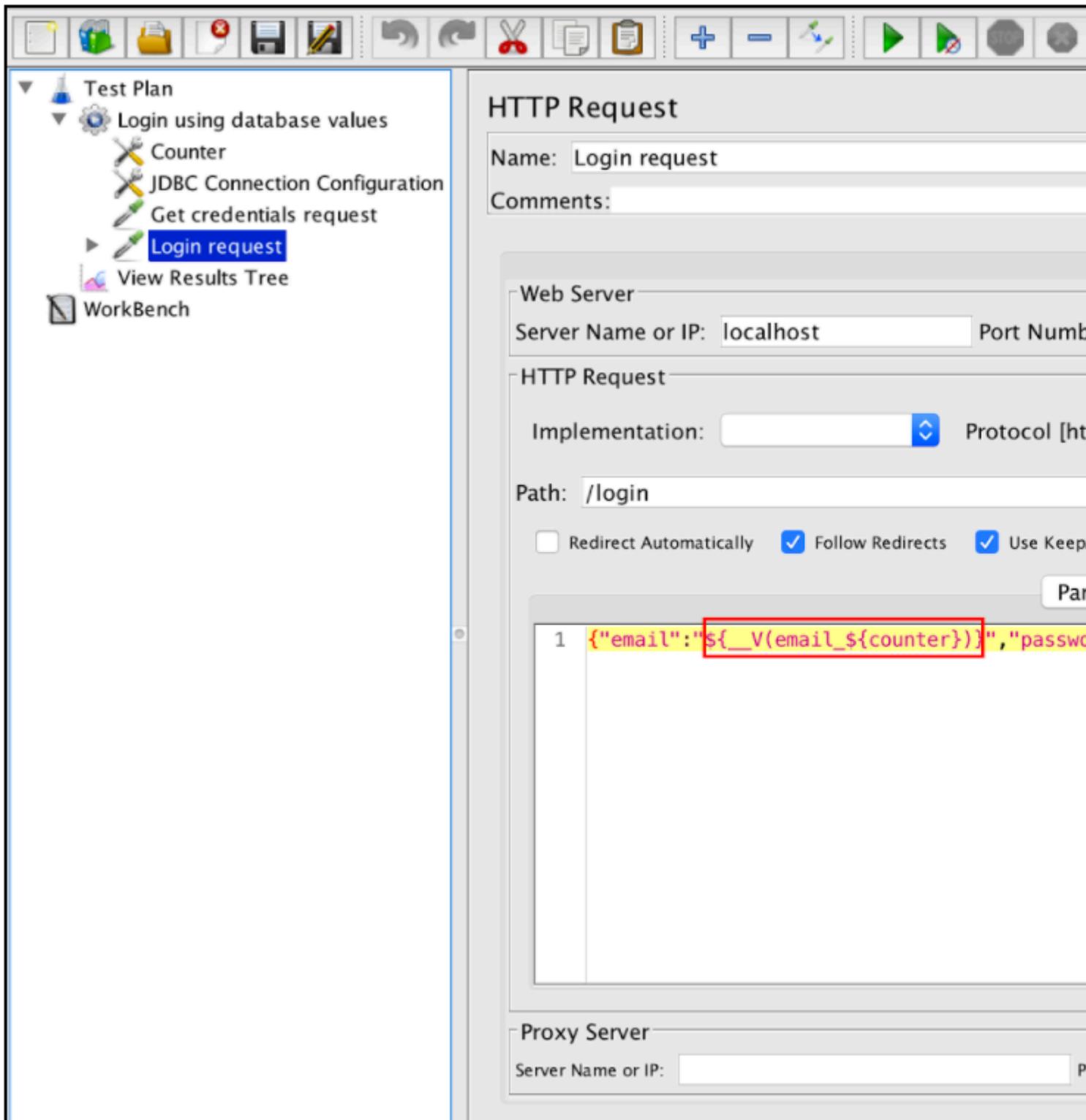
Nous aurons maintenant des variables JMeter qui pourront être utilisées dans les requêtes suivantes. Les variables spécifiées seront créées avec un suffixe incrémentiel (email_1, email_2, email_3... ..).

Pour utiliser ces variables dans la demande de connexion, nous devons ajouter un compteur qui sera utilisé pour accéder aux bonnes valeurs de la réponse à la requête JDBC. Pour ajouter l'élément 'Counter' dans JMeter: *Faites un clic droit sur Thread Group -> Add -> Config Element ->*

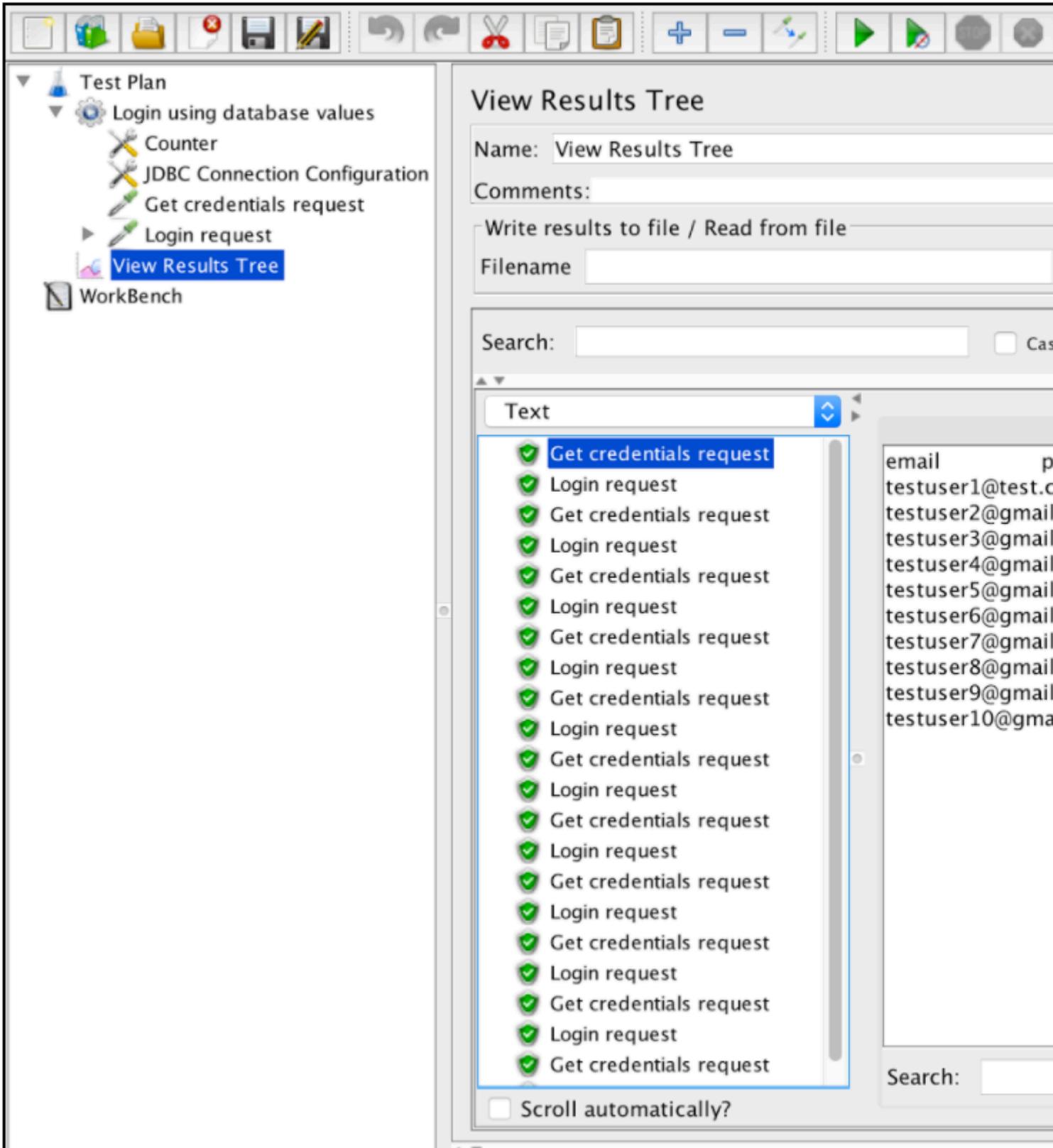
Counter

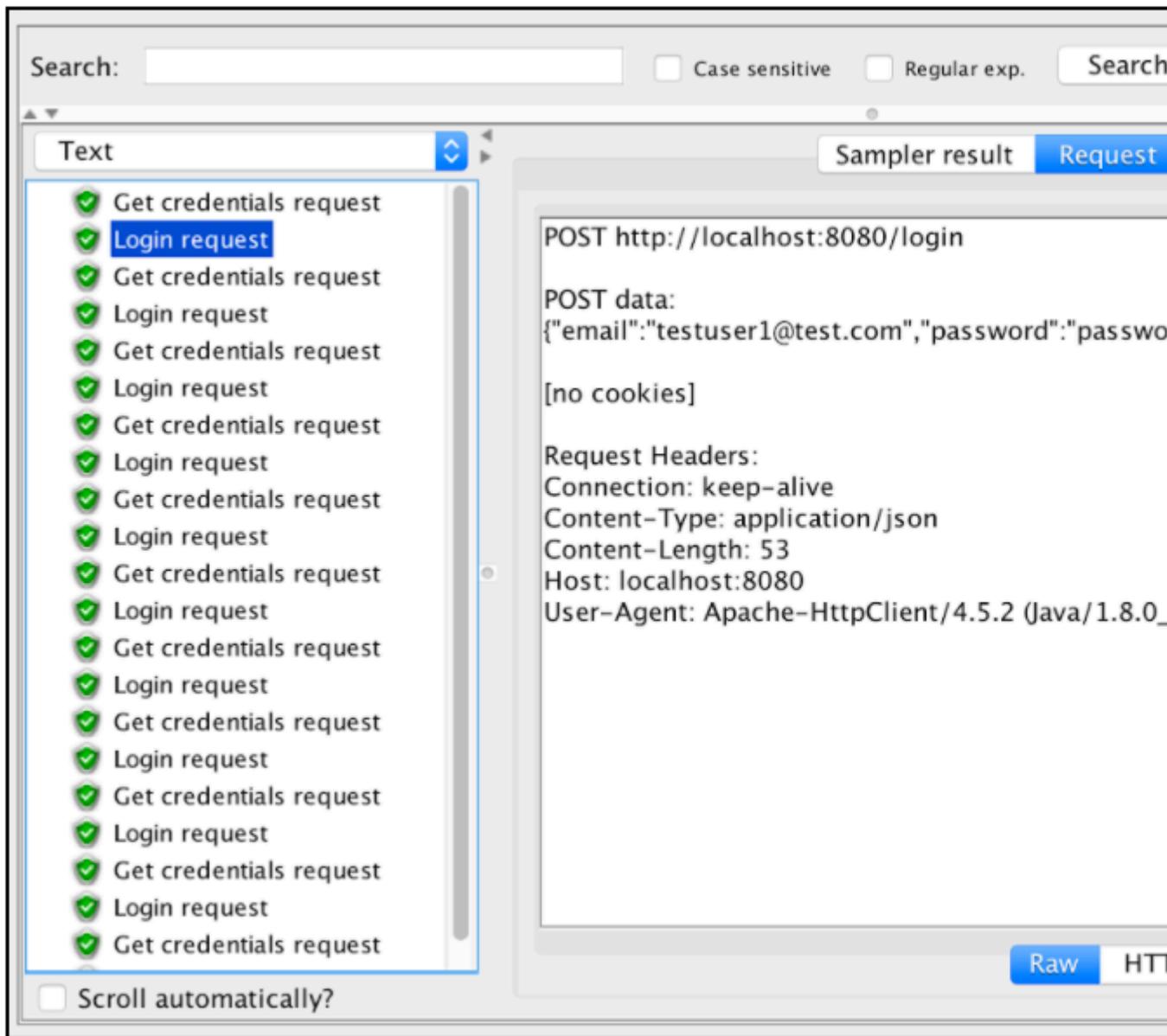


Après cela, nous pouvons mettre à jour la demande de connexion en utilisant la fonction `__V`. Cela renvoie le résultat de l'évaluation d'une expression de nom de variable et peut être utilisé pour évaluer les références de variables imbriquées:



La configuration spécifiée est suffisante pour utiliser les valeurs de base de données pour exécuter le script sur différents utilisateurs:



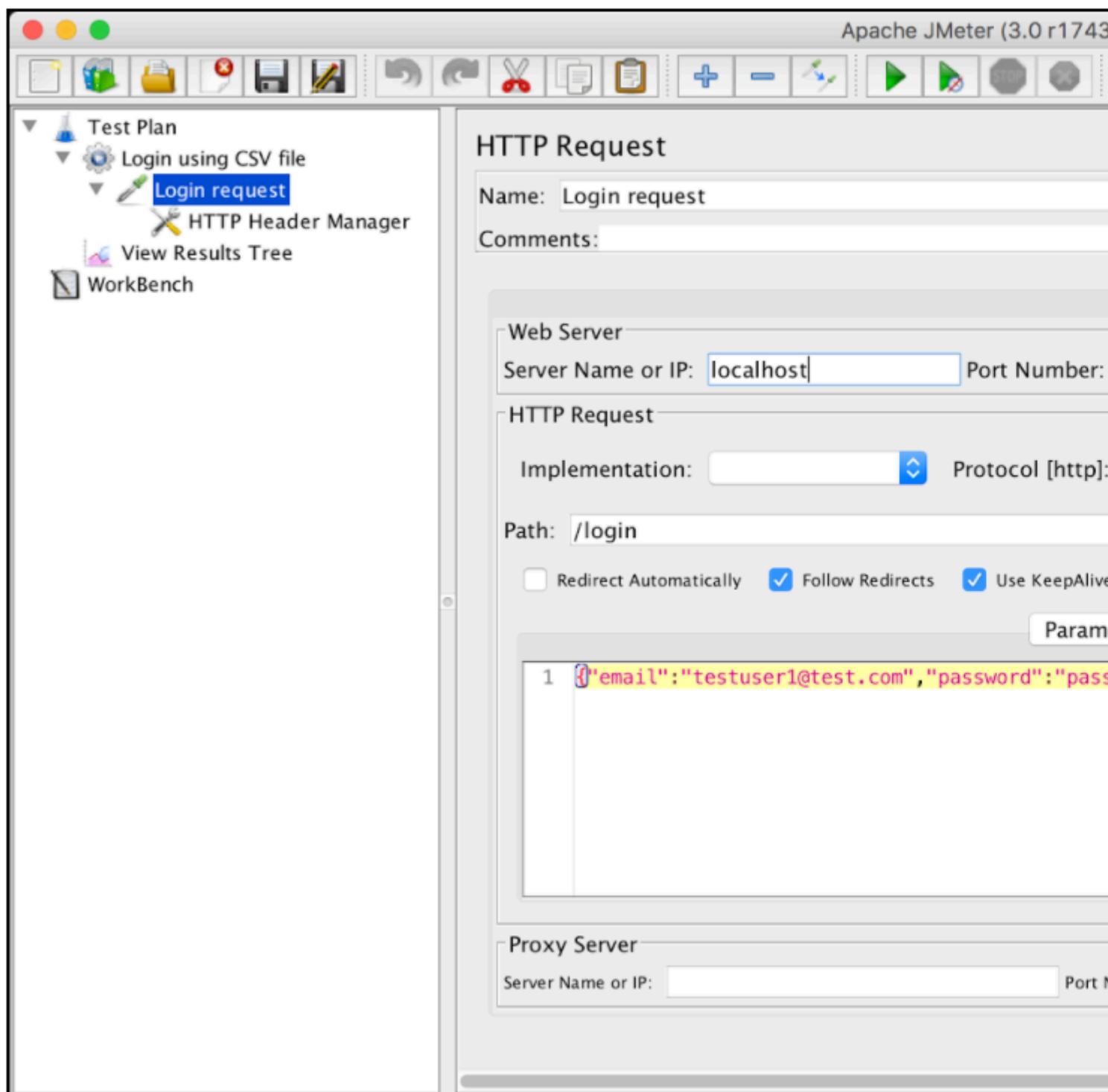


Paramétrage à l'aide du plugin 'Parameterized Controller'

Si vous devez exécuter une séquence répétée de la même action avec des paramètres différents, utilisez le plug-in tiers "Parameterized Controller" du projet [JMeter-Plugins](#).

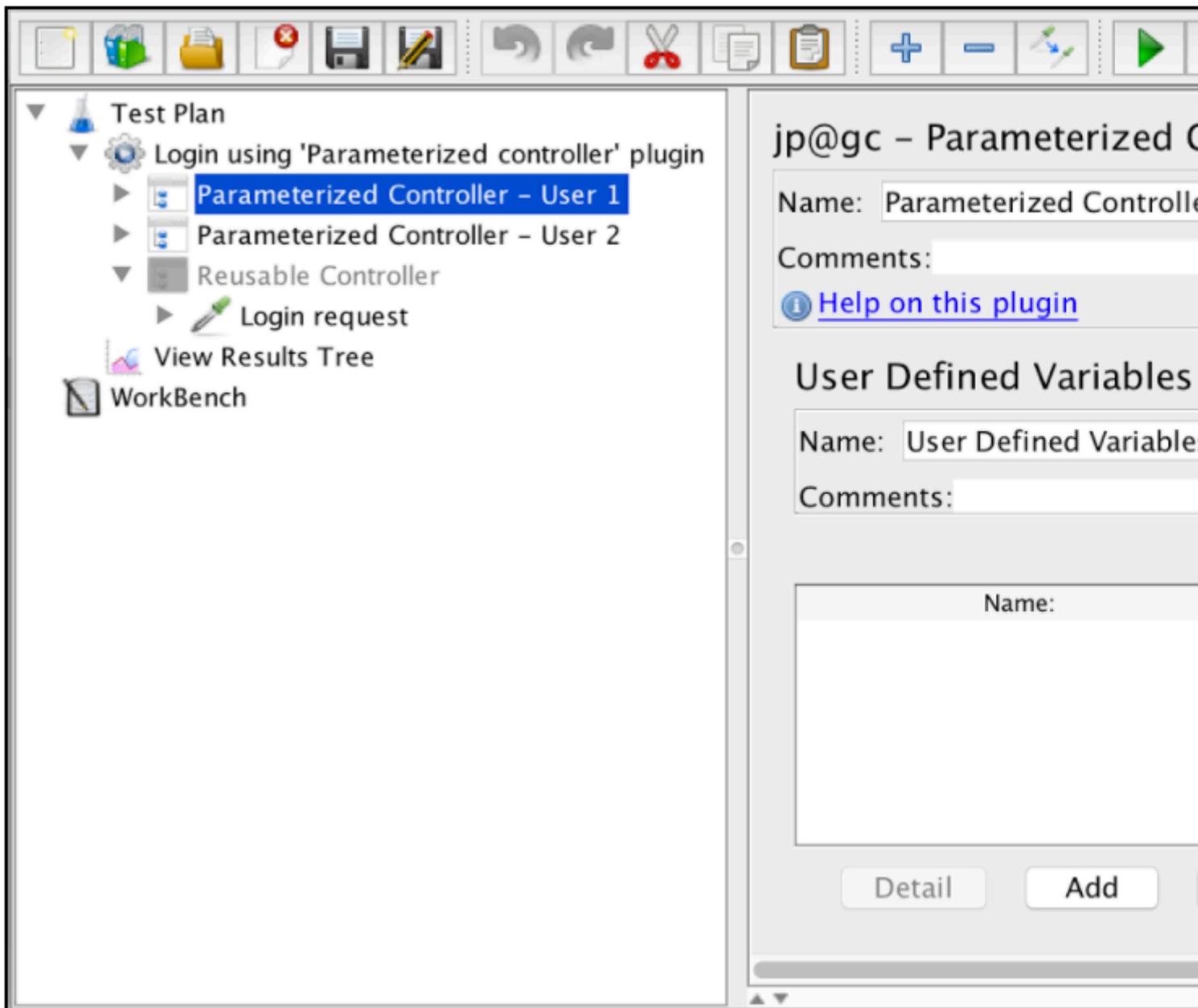
Vous devez d'abord installer ce plugin en suivant la procédure d'installation.

Supposons que nous souhaitons paramétrer le workflow de connexion:

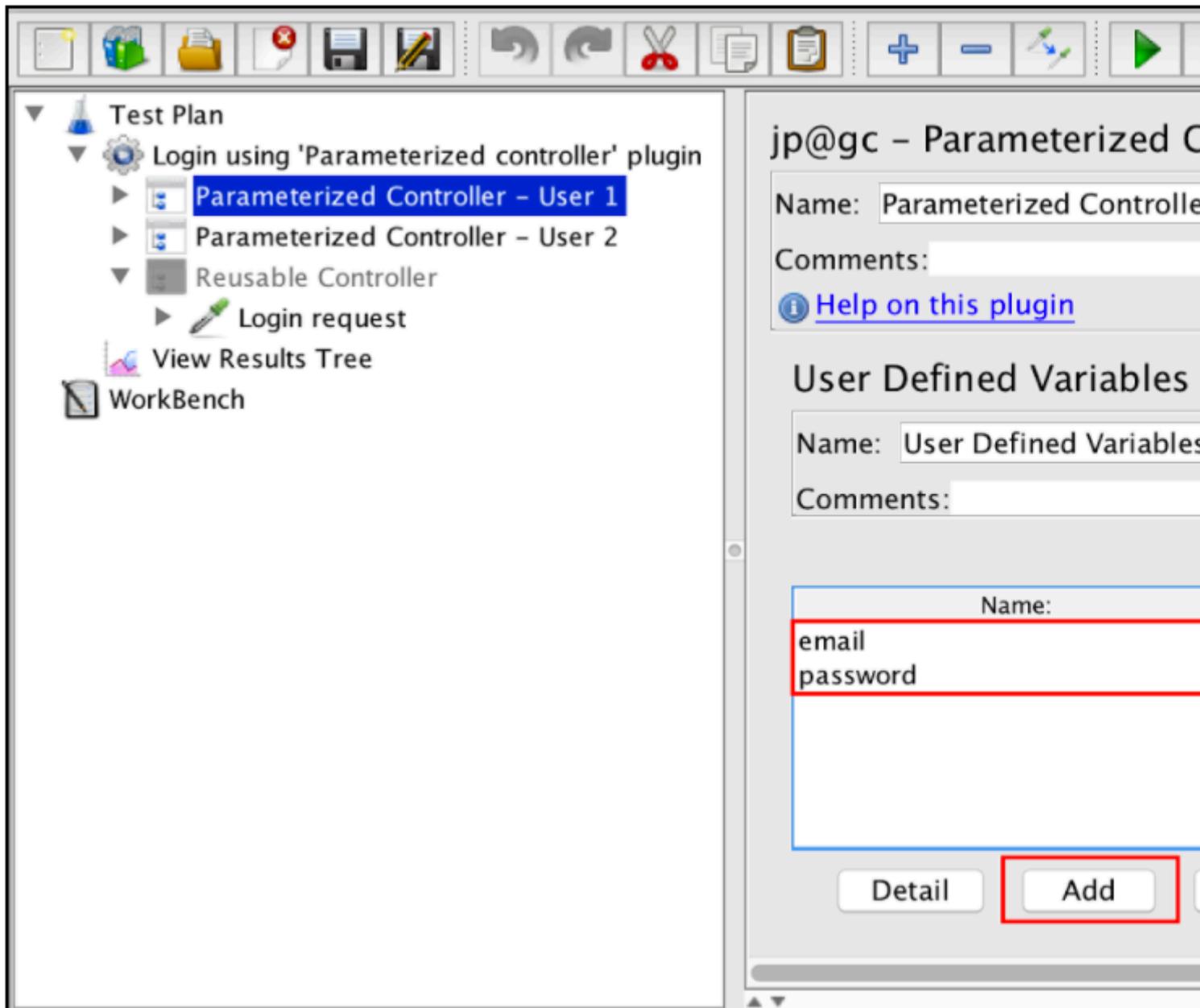


Tout d'abord, vous devez installer le plug-in "Parameterized Controller" car il n'est pas inclus dans le core JMeter. Les étapes d'installation de ce processus peuvent être trouvées ici.

Déplaçons la "Demande de connexion" dans un contrôleur distinct et désactivez-la (cliquez avec le bouton droit de la souris et sélectionnez "Désactiver"). C'est le moyen le plus préférable d'avoir un conteneur de modules dans votre plan de test et d'éviter d'utiliser Workbench en tant que tel. Une fois l'installation terminée, vous pouvez ajouter deux contrôleurs "Contrôleurs paramétrés" avec des informations d'identification utilisateur différentes: *Cliquez avec le bouton droit sur Groupe de threads -> Ajouter -> Contrôleur logique -> Contrôleur paramétré*

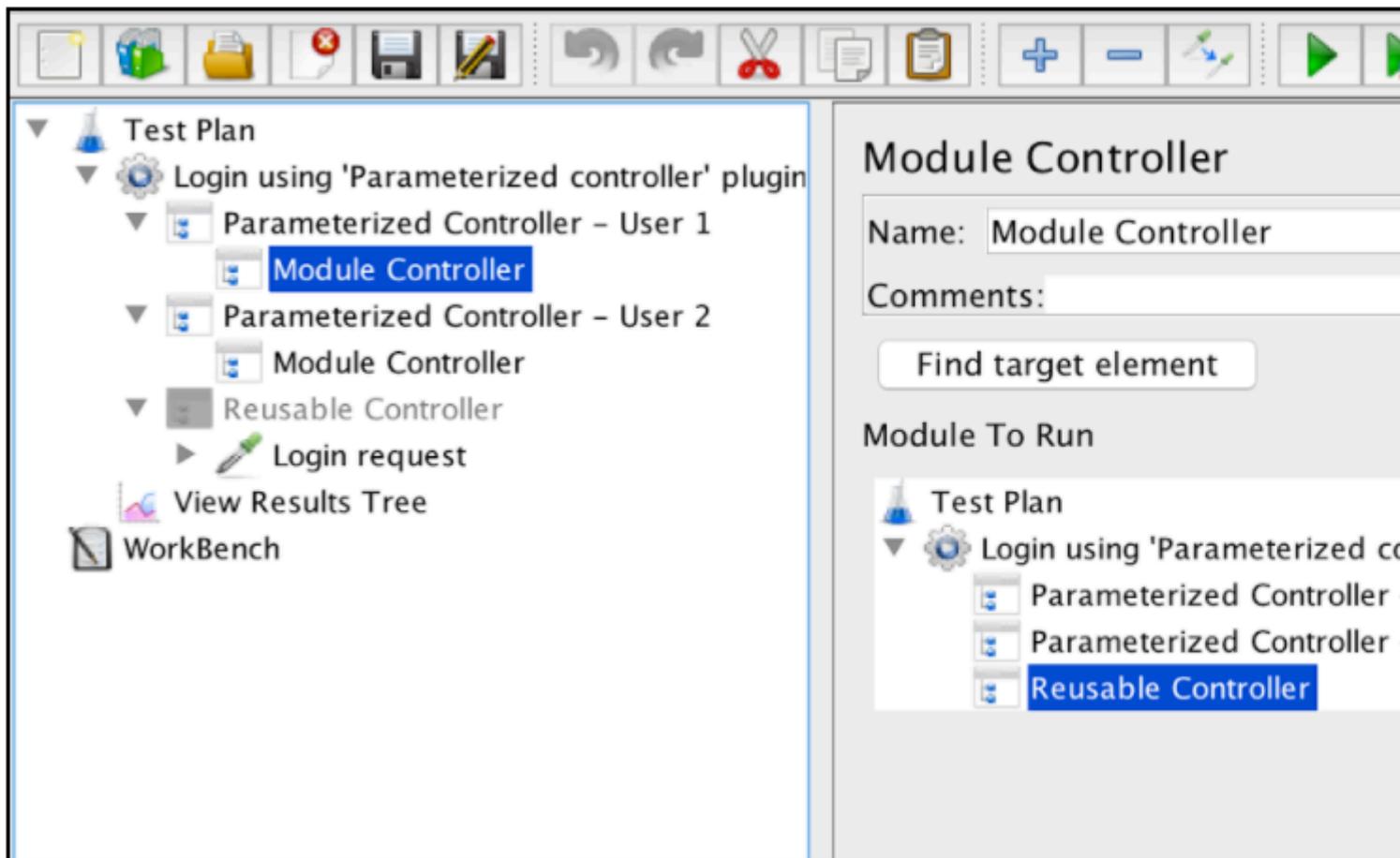


Les contrôleurs paramétrés contiennent la section "Variables définies par l'utilisateur" dans laquelle vous pouvez spécifier vos paramètres. Placez les informations d'identification du premier utilisateur dans le premier contrôleur paramétré et les deuxièmes informations d'identification de l'utilisateur dans le second contrôleur paramétré.



À l'intérieur des deux contrôleurs paramétrés, ajoutez des références au «contrôleur réutilisable» pour appeler la «demande de connexion» avec différents paramètres. Cela peut être fait de cette façon:

Faites un clic droit sur 'Contrôleur paramétré' -> 'Ajouter' -> 'Contrôleur logique' -> 'Contrôleur de module'



Lors de l'exécution de votre script, vous verrez que la «demande de connexion» a déclenché séparément chacun des contrôleurs paramétrés. Cela peut être très utile si vous devez exécuter votre script sur différentes combinaisons de paramètres d'entrée.

The screenshot displays the Apache JMeter interface. On the left, the Test Plan tree is expanded to show a 'Login using 'Parameterized controller' plugin' test plan. This plan contains three 'Parameterized Controller' elements (User 1 and User 2) and one 'Reusable Controller'. Each controller has a 'Module Controller' child. Below the controllers is a 'Login request' element. The 'View Results Tree' element is highlighted in blue. At the bottom of the tree is the 'WorkBench' element.

On the right, the 'View Results Tree' panel is visible. It has a 'Name' field containing 'View Results Tree' and a 'Comments' field. There is a checkbox for 'Write results to file / Read from file' and a 'Filename' field. Below this is a 'Search' field. The main area of the panel shows a list of results under the heading 'Text'. Two 'Login request' entries are listed, both with a green checkmark icon. The second 'Login request' entry is highlighted in blue.

Lire Paramétrage d'Apache JMeter en ligne:

<https://riptutorial.com/fr/jmeter/topic/9602/parametrage-d-apache-jmeter>

Crédits

S. No	Chapitres	Contributeurs
1	Démarrer avec Apache JMeter	Aliaksandr Belik , Chulbul Pandey , Community , Kiril S. , M Navneet Krishna , Milamber , Naveen , NaveenKumar , Namachivayam , RowlandB , UBIK LOAD PACK , Venkatesh Achanta
2	Apache JMeter: enregistrement de scénario de test	UBIK LOAD PACK , Yuri Bushnev
3	Corrélations Apache JMeter	UBIK LOAD PACK , Yuri Bushnev
4	Paramétrage d'Apache JMeter	UBIK LOAD PACK , Yuri Bushnev