

HP Unified Functional Testing

для ОС WindowsSoft

Версия ПО: 11.51

Тестирование API Учебник

Дата выпуска документа: январь 2013 г.

Дата выпуска программного обеспечения: январь 2013 г.



Официальное уведомление

Гарантия

Гарантии на продукты и услуги компании HP формулируются только в заявлениях о прямой гарантии, сопровождающих эти продукты и услуги. В них нет ничего, что может быть истолковано как дополнительная гарантия. Компания HP не несет ответственности за содержащиеся в них технические или редакционные ошибки.

Приводимые в них сведения могут быть изменены без какого-либо уведомления.

Ограничение прав

Конфиденциальное компьютерное ПО. Для обладания, использования или копирования необходима действующая лицензия от компании HP. Согласно FAR 12.211 и 12.212, выдача лицензий на коммерческое компьютерное ПО, документацию на компьютерное ПО и технические данные для коммерческих элементов правительству США производится на условиях стандартной коммерческой лицензии поставщика.

Уведомления об авторских правах

© Hewlett-Packard Development Company, L.P., 2010—2012.

Информация о товарных знаках

Java является зарегистрированным товарным знаком корпорации Oracle или ее дочерних компаний.

Microsoft® и Windows® являются зарегистрированными в США торговыми знаками корпорации Microsoft Corporation.

Oracle® является зарегистрированным в США торговым знаком корпорации Oracle Corporation, Редвуд, штат Калифорния.

UNIX® является зарегистрированным товарным знаком The Open Group.

SlickEdit® является зарегистрированным товарным знаком корпорации SlickEdit Inc.

Обновление документации

Титульная страница этого документа содержит следующие идентификационные данные:

- номер версии ПО, указывающий на версию программы;
- дата выхода документа, которая изменяется при каждом обновлении документа;
- дата выпуска программы, которая указывает дату выпуска данной версии ПО.

Чтобы проверить наличие последних обновлений или убедиться в том, что используется последняя редакция документа, перейдите на веб-сайт:

<http://h20230.www2.hp.com/selfsolve/manuals>

Данный сайт требует регистрации и входа в HP Passport. Чтобы зарегистрировать учетную запись HP Passport, перейдите на веб-сайт:

<http://h20229.www2.hp.com/passport-registration.html>

или щелкните ссылку **New users - please register** на странице входа в HP Passport.

Обновленные или новые редакции можно получать, подписавшись на соответствующую службу поддержки продукта. Для получения дополнительных сведений обратитесь к торговому представителю HP.

Поддержка

Посетите веб-сайт HP Software Support:

<http://www.hp.com/go/hpsoftwaresupport>

На этом сайте можно найти контактную информацию и сведения о продуктах, услугах и технической поддержке, предлагаемых HP Software.

Интерактивная техническая поддержка HP Software предоставляет заказчику возможности самостоятельного поиска решений (Self-solve). Она обеспечивает быстрый и эффективный доступ к интерактивным средствам технической поддержки, которые необходимы для управления бизнесом. Клиенты службы поддержки могут воспользоваться следующими преимуществами сайта:

- поиск интересующих документов базы знаний;
- отправка и контроль описаний конкретных случаев и расширенных запросов для получения технической поддержки;
- загрузка исправлений ПО;
- управление договорами на техническую поддержку;
- поиск контактов в HP для технической поддержки;
- проверка сведений о доступных услугах;
- участие в обсуждении различных вопросов с другими заказчиками ПО;
- исследование определенных проблем и регистрация для обучения программному обеспечению.

В большинстве случаев для получения поддержки требуется регистрация HP Passport, а также договор на услуги технической поддержки. Чтобы зарегистрировать учетную запись HP Passport, перейдите по адресу:

<http://h20229.www2.hp.com/passport-registration.html>

Для получения дополнительных сведений об уровнях доступа см.:

http://h20230.www2.hp.com/new_access_levels.jsp

Содержание

Введение в учебник	7
Уроки в учебнике.....	7
Глава 1: Введение в HP Unified Functional Testing — тестирование API	9
Что такое SOA?.....	9
Почему следует автоматизировать тестирование SOA?	11
Общие сведения о терминологии	11
Что собой представляет пример приложения в этом учебнике?.....	13
Как выполняется запуск приложения?	14
Глава 2: Создание простого теста	15
Как создать новый тест?	16
Как осуществляется работа с панелями Unified Functional Testing Service Test?	17
Как создать шаг теста?.....	18
Как соединить шаги теста?	22
Как сопоставить данные из нескольких источников?	24
Как осуществляется управление шагом с помощью данных?	27
Глава 3: Тест веб-службы	33
Как импортировать веб-службу?	34
Как создать тест веб-службы?	36
Как интегрировать данные в тест?.....	42
Как использовать несколько источников данных и пользовательский код?.....	47
Глава 4: Тестирование служб REST	53
Как создать операцию службы REST?	54
Как запустить тест REST?	60
Как назначить данные используемому методу REST?.....	62
Как проверить полученные данные?.....	67
Как разрешить изменения в используемой службе REST?	69

Введение в учебник

Добро пожаловать в API Testing для тестирования без GUI — руководство для самостоятельного изучения, которое можно распечатать, которое поможет вам создать тесты для веб-служб, служб REST и других приложений без графического интерфейса пользователя.

После изучения этого учебника вы сможете разрабатывать, запускать и отслеживать простые тесты в собственных приложениях. Рекомендуется изучать учебные материалы в приведенном здесь порядке.

Уроки в учебнике

В учебник включены следующие уроки

Урок 1. Введение в HP Unified Functional Testing — тестирование API: Знакомство с Учебником по HP Unified Functional Testing для тестирования API.

Урок 2. Создание простого теста: Введение в основы создания теста путем перетаскивания операций на полотно. Также описывается управление теста с помощью внешних данных.

Урок 3. Тест веб-службы: Описывается создание теста для веб-службы и проверка результатов.

Урок 4. Тестирование служб REST: Описывается моделирование и создание тестов для служб REST.

1

Введение в HP Unified Functional Testing — тестирование API

HP Unified Functional Testing для тестирования API содержит расширяемый механизм для создания и выполнения функциональных тестов систем, у которых отсутствует пользовательский интерфейс. В этом документе описывается начало работы с HP Unified Functional Testing и создание первых тестов. В нем представлено описание основных функций продукта, а также описание способов интеграции их в используемые тесты.

Тест создается путем перетаскивания операций на полотно. Выполняется задание параметров, назначение значений и запуск тестов. Наличие ожидаемых значений можно проверить с помощью встроенного механизма контрольных точек или с помощью пользовательского кода C#.

Что такое SOA?

За последние годы аутсорсинг и комплексные бизнес-потребности потребовали от компаний совместной работы и обмена информацией. Помимо этого, учитывая растущее число слияний и поглощений организаций, компаниям стало все труднее обмениваться данными. Если до слияния у двух компаний имелось и обслуживалось две собственные фирменные компьютерные системы, то после слияния обмен данными мог потребовать значительных временных и финансовых затрат.

Для решения этих проблем несколько поставщиков разработали технологии для обеспечения связи в режиме B2B (Business-to-Business). Примерами технологий связи в режиме B2B являются RMI, COM, CORBA, EDI и веб-службы. Помимо обеспечения возможности для систем объединения по каналам, эти технологии также обрабатывают разрешения, обеспечивающие возможность передачи данных между системами по сети аналогично интрасети.

Веб-службы — это независимые приложения, которые могут запускаться по интрасети на различных платформах. Они используют XML и SOAP в качестве базового языка, что делает их удобными для разработчиков решением. Поскольку веб-службы основаны на наборе стандартизированных правил и спецификаций, они более мобильны по сравнению с другими технологиями.

SOA (Сервисно-ориентированная архитектура) — это архитектурный стиль, обеспечивающий возможность взаимодействия для нескольких ПО. Службой называется единица работы, выполненная поставщиком служб, как указано клиентом. При использовании SOA требуется, чтобы службы взаимодействовали друг с другом, но при этом не зависели друг от друга. Службы являются автономными и слабо связанными элементами. Необходимо, чтобы в них содержались определенные сведения друг о друге, но отсутствовали взаимозависимости.

Системы SOA главным образом основаны на веб-службах. В веб-службах клиент отправляет запрос и поставщик веб-служб отправляет ответ с использованием протокола SOAP. HP Unified Functional Testing для тестирования API позволяет проверить режим работы веб-служб автоматически.

Почему следует автоматизировать тестирование SOA?

Автоматизированное тестирование SOA представляет собой область, в которой используются продукты и процессы, назначением которых является снижение рисков обновления приложений или развертывание новых служб. В целом автоматизированное тестирование сводится к применению производственных рабочих нагрузок к системам перед развертыванием, а также к измерению производительности системы и оценке условий работы конечного пользователя. Правильно спроектированный тест производительности отвечает на следующие вопросы.

- Достаточно ли время отклика службы с точки зрения пользователей?
- Будут ли получены от сервера правильные значения?
- Как служба будет обрабатывать исключения и недопустимые значения?
- Будет ли служба стабильно работать под ожидаемой нагрузкой и в случае ее превышения?

Отвечив на эти вопросы, вы сможете более точно разработать тест. Эффективный процесс автоматизированного тестирования помогает принять более осознанные решения о выпуске и предотвращает проблемы доступности и простоев системы.

Общие сведения о терминологии

При работе с тестированием SOA могут встретиться следующие термины:

HTTP	Протокол передачи гипертекста , протокол связи, который используется для передачи или предоставления данных через Интернет. Пользователи используют HTTP для публикации и извлечения HTML-страниц.
JMS	Служба сообщений Java , межплатформенный Middleware API на основе Java, предназначенный для передачи сообщений между двумя и более клиентами.
REST	Передача репрезентативного состояния . Стиль архитектуры ПО для распределенных систем.

SOAP	Протокол SOAP или протокол сервисно-ориентированной архитектуры , протокол для обмена структурированными и типизированными данными между одноранговыми узлами в распределенной среде с помощью передачи XML через HTTP или JMS. SOAP позволяет сериализовать распределенные компоненты в XML-документы для транспорта и десериализации при достижении назначения. Благодаря этому стимулируется возможность взаимодействия между компонентами, основанными на различных технологиях.
Тест	Шаги, выполняемые пользователем, описаны в тесте . Тест симулирует действия реальных пользователей, использующих приложение.
UDDI	Универсальное обнаружение, описание и интеграция , независимый от платформы онлайн-реестр, в котором указываются компании со всего мира. Часто используется как база данных для общедоступных веб-служб.
Веб-службы	Стандартизированные веб-приложения, использующие стандарты XML, SOAP, WSDL и UDDI по протоколам Интернета. XML используется для добавления тегов к данным, SOAP используется для передачи данных, WSDL описывает службы, а UDDI содержит их список.
WSDL	Язык описания веб-служб , язык на основе XML, разработанный для описания веб-служб. Документ WSDL содержит важные сведения о веб-службе, такие как данные об ее портах и операциях, необходимых для реализации.
WS-I	Web Service Interoperability, стандарт, созданный организацией Web Service Interoperability для стимулирования совместимости между веб-службами.

XML	<p>Расширяемый язык разметки представляет собой язык разметки общего назначения. Он называется расширяемым, поскольку он позволяет определять собственные теги. Он обеспечивает возможность совместного использования структурированных данных в различных информационных системах, особенно через Интернет. XML используется для сериализации данных и кодирования документов.</p>
XSD	<p>Определение XML-схемы файлы содержат набор правил, которые официально определяют иерархическую структуру XML-документа. Чтобы XML-документ был распознан средствами анализа и обработчиками как допустимый, он должен соответствовать этим правилам и ограничениям.</p> <p>WSDL-документы могут ссылаться на внешнюю схему и могут содержать встроенную схему.</p>

Что собой представляет пример приложения в этом учебнике?

Основой этого учебника служит пример приложения, поставляемый в комплекте с этим продуктом — **API Flights**. Доступно и как веб-служба, и как служба REST.

Пример приложения работает с базой данных бронирования авиабилетов. Можно извлекать авиарейсы для определенных назначений, создавать заказы для клиентов, обновлять забронированные авиабилеты и удалять их.

Дополнительные сведения о методах и операциях службы введите help в окно командной строки примера приложения.

Примечание: Чтобы включить поддержку функций Excel для источников данных, рекомендуется установить на компьютере Microsoft Excel.

Как выполняется запуск приложения?

В качестве первого шага выполняется открытие приложения по работе с рейсами, которое будет доступно для тестирования.

Запуск службы HP Flights

- 1 Убедитесь в наличии прав администратора. Они необходимы в ОС Windows для запуска службы HP Flights.
- 2 Нажмите **Пуск > (Все) Программы > HP Software > HP Unified Functional Testing > Sample Applications > Flight API**. Откроется окно командной строки, в котором будет указана доступность этого приложения.
- 3 Если в окне отображается сообщение о том, что порт 24240, задаваемый по умолчанию, недоступен, измените файл **<каталог_установки>SampleApplication\HPFlights_Service.exe.config** в текстовом редакторе. В разделе **appSettings** замените ключ порта 24240 на допустимый порт.
- 4 Сверните окно командной строки примера приложения. Не закрывайте окно командной строки, поскольку при этом будет остановлена работа службы.

Дальнейшие шаги

После открытия приложения можно начать создание тестов для приложений, у которых отсутствует пользовательский интерфейс, с помощью приложения API Flights. В следующих уроках описывается процесс создания теста для базовых операций, веб-служб и служб REST.

2

Создание простого теста

В этом уроке описываются шаги создания тестов с помощью выполнения простых действий.

Этот урок состоит из следующих разделов.

- Как создать новый тест?
- Как осуществляется работа с панелями Unified Functional TestingService Test?
- Как создать шаг теста?
- Как осуществляется управление шагом с помощью данных?
- Как соединить шаги теста?
- Как сопоставить данные из нескольких источников?

Как создать новый тест?

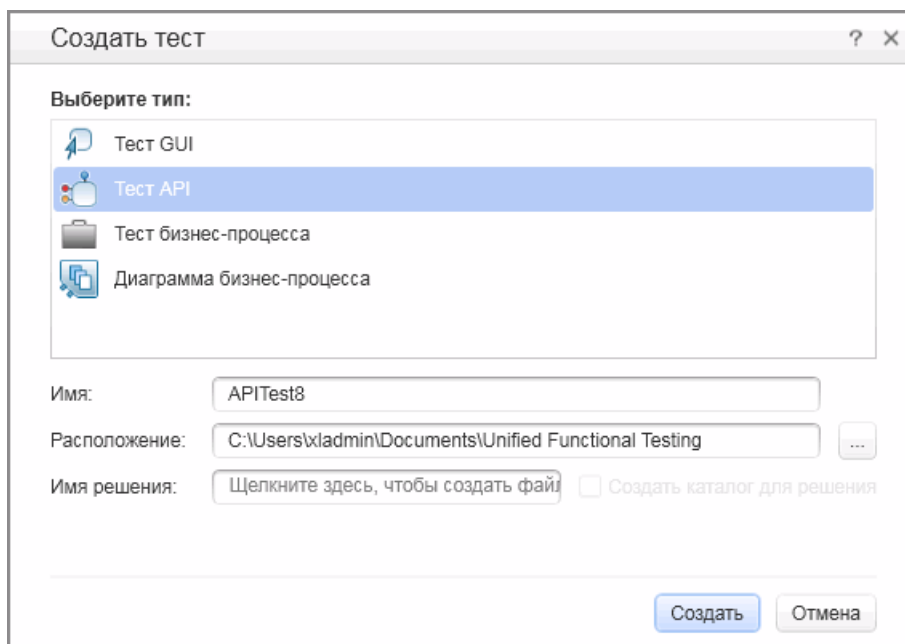
Полотно является центральной консолью, на которой создаются и запускаются тесты.

1 Откройте UFT HP.

Нажмите **Пуск > (Все) программы > HP Software > HP Unified Functional Testing > Unified Functional Testing**. Откроется страница «Пуск».

2 Создайте новый тест.

Нажмите **Файл > Создать > Тест**. Откроется диалоговое окно «Создать тест». Выберите тип **Тест API**. На компьютерах, на которых установлен HP LoadRunner, тип **Нагрузочный тест API** также отображается.



3 Создание нового решения

В поле **Имя** замените имя по умолчанию на **BasicTest** и нажмите **Создать**. Откроется пустой тест, при этом на холсте будут отображаться разделы **Запуск**, **Поток теста** и **Завершение**.

Поток теста — это раздел теста, содержащий действия, функциональность которых необходимо протестировать. Раздел **Запуск** оптимально подходит для определения элементов, которые необходимо инициализировать до теста, например, переменные теста.

Как осуществляется работа с панелями Unified Functional Testing Service Test?

Большинство панелей в интерфейсе Unified Functional Testing являются перемещаемыми закрепляемыми окнами. Чтобы настроить отображение панелей по умолчанию в исходных местоположениях выберите **Просмотр > Сброс макета окон**.

Ниже перечислены основные панели.

- **Область обозревателя решений** (слева) Иерархия дерева всех тестов и действий в текущем решении с соответствующими ссылками, потоками и событиями.
- **Панель инструментов** (слева) Коллекция встроенных и импортированных действий, которые можно добавить как шаги теста. С этой панели можно перетаскивать действия на холст.
- **Холст**. (средняя область) Рабочая область, которая используется для организации шагов тестов.
- **Панель данных**. (нижняя часть) Иерархия дерева источников данных, которые можно использовать с тестом: импортированные файлы в формате Excel и XML, таблицы базы данных или определенная вручную таблица.
- **Панель вывода** (нижняя часть) Информационная область, содержащая сведения о запуске и состоянии теста.

Как создать шаг теста?

Шаги теста создаются путем перетаскивания действий из **панели инструментов** на холст.

В этом разделе создается простой шаг теста, который используется для иллюстрации использования панели инструментов и различных панелей.

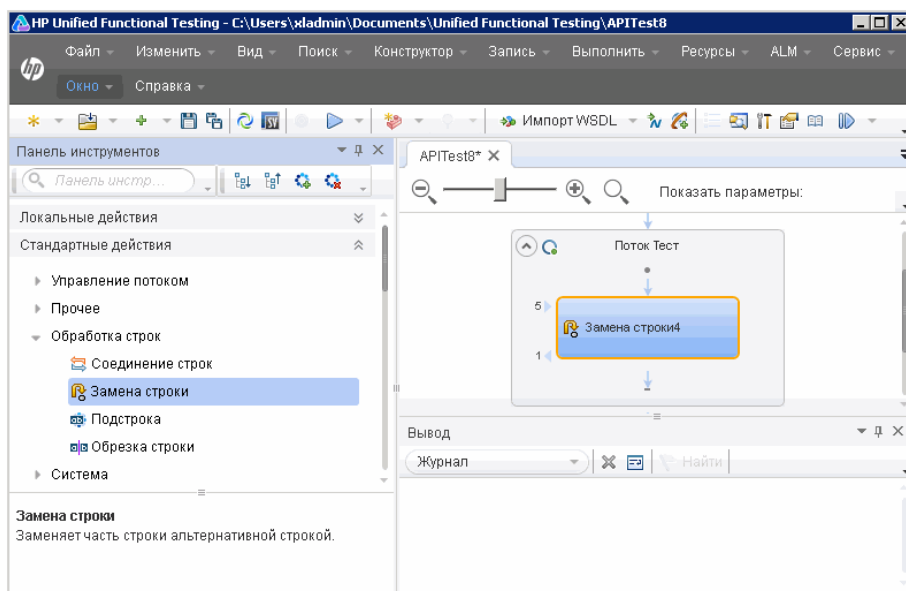
Создание простого шага теста «Замена строки»

1 Найдите действие «Замена строки».

На левой панели перейдите на вкладку **Панель инструментов**, чтобы отобразить **Панель инструментов**. Разверните категорию **Обработка строк** и выберите **Замена строки**.

2 Создание шага

Перетащите действие **Замена строки** на холст и поместите его в **Поток теста**. При выполнении этого действия выполняется поиск текста в определенной строке и этот текст заменяется новым текстом. В качестве альтернативы дважды щелкните действие на панели инструментов, чтобы добавить его на холст.



3 В представлении «Общие» измените отображаемое имя шага.



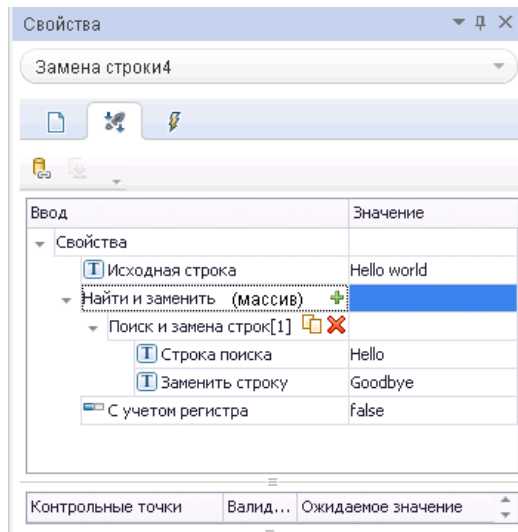
Выберите **Просмотр > Свойства**. На холсте выберите шаг **Замена строки**, а на панели **Свойства** щелкните вкладку **Общие**. В строке **Имя** введите **Изменить текст** и нажмите клавишу ENTER. При этом будет изменено имя шага на холсте.

4 Установка входных свойств



На панели **Свойства** перейдите на вкладку **Ввод/Контрольные точки**. Введите следующие значения:

- **Исходная строка:** Hello world.
- **Строка поиска:** Hello
- **Строка замены:** Goodbye
- **С учетом регистра:** false



5 Запустите тест.



Чтобы открыть диалоговое окно «Выполнить тест», нажмите кнопку **Выполнить** или нажмите кнопку F5 . Нажмите **Параметры**, чтобы развернуть диалоговое окно. Установите флажок **Папка временных результатов**. Нажмите кнопку **Выполнить** и выполните тест.

6 Просмотрите результаты.

Откроется Run Results Viewer.

Выберите **Просмотр > Развернуть все** или нажмите кнопку на панели инструментов «Развернуть все». Откройте узел **Изменить текст**. Просмотрите исходные строки и строки замены и обратите внимание на результирующую строку **Goodbye world**. Фактически, это ожидаемая строка — тест пройден.

После завершения просмотра результатов закройте Run Results Viewer.

The screenshot shows the HP Run Results Viewer interface. The main window title is 'APITest8 \ Result2 - HP Run Results Viewer'. The left sidebar shows a tree view with nodes: 'Обзор Тест APITest8', 'Запуск', 'Поток Тест', 'Обзор', 'Итерация 1', 'Замена строки4', and 'Конец'. The main area displays 'Result Details' for 'Step Name: Замена строки4'. Below this, it says 'Step Done' and shows a table with columns 'Object', 'Details', 'Result', and 'Time'. The table contains one row: 'Замена строки4', 'Done', '21.03.2013 - 16:28'. At the bottom, there are tabs for 'Result Details', 'Screen Recorder', and 'System Monitor'. Below the tabs is a section for 'Captured Data' with a sub-section 'Свойства шага' and a table with columns 'Имя' and 'Значен'.

Object	Details	Result	Time
Замена строки4		Done	21.03.2013 - 16:28

Имя	Значен
Тип	HP.ST.Ext.BasicActivities.ReplaceStringActivity
ID шага	ReplaceStringActivity4
Исходная строка	'Hello world'

7 Задайте контрольную точку.

В предыдущих шагах выходные данные просматривались вручную для проверки соответствия результата ожидаемому значению. Контрольные точки обеспечивают возможность определения успешности выполнения действия без проверки результата вручную. Контрольные точки являются средством проверки теста: успех или ошибка теста определяются по его контрольным точкам.

Перейдите обратно на панель «Свойства» (правая панель) и убедитесь, что отображается вкладка **Ввод/Контрольные точки**. Щелкните в нижней части панели, в разделе **Контрольные точки** и установите флажок в строке **Результаты**, чтобы включить контрольную точку. В столбце **Ожидаемое значение** введите ожидаемую строку **Goodbye world**.

Выполните тест повторно. В Run Results Viewer разверните узлы и отметьте контрольную точку. Это указывает на то, что контрольная точка считается пройденной, поскольку результат соответствует ожидаемому значению.

После завершения просмотра результатов закройте Run Results Viewer.

Как соединить шаги теста?

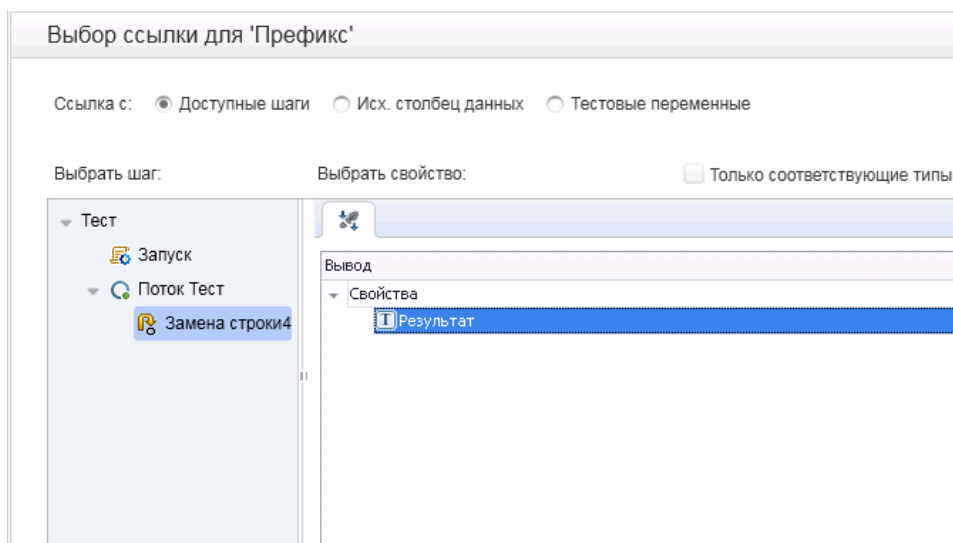
В этом разделе используются выходные данные одного шага в качестве входных данных для другого шага.

1 Добавьте шаг «Добавить соединение строк».

На панели инструментов выберите **Соединение строк** в категории **Обработка строк**. Перетащите действие на холст и поместите его ниже шага Изменить текст в потоке текста. Это действие объединяет две строки.

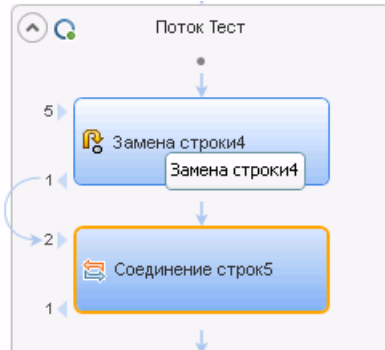
2 Задайте префикс.

На холсте выберите шаг **Соединение строк**. На панели **Свойства** перейдите на вкладку **Ввод/Контрольные точки**. В верхнем разделе **Ввод** переместите мышь в ячейку **Значение** строки **Префикс**. Нажмите кнопку **Ссылка на источник данных**. Откроется диалоговое окно «Выбор источника связи».



3 Свяжите шаги.

В диалоговом окне «Выбор исходной ссылки» выберите параметр **Доступные шаги**. Выберите узел **Поток теста** > **ChangeText**. На правой панели дважды щелкните узел **Результаты**. Теперь на холсте указывается, что данные перемещаются из **Изменить текст** в **ConcatenateString**.



4 Настройте суффикс.

На панели **Свойства** введите текст `Welcome to the Basic Test.` в поле **Значение** свойства **Суффикс**.

Ввод	Значение
Свойства	
Префикс	{Step.OutputProperties.R
Суффикс	Welcome to the Basic Test

5 Запустите тест.



Чтобы запустить тест, нажмите кнопку **Выполнить**, либо клавишу F5.

6 Просмотрите отчет.

Разверните дерево результатов выполнения и выберите узел **ConcatenateStringsActivity**. В отчете будут показаны результаты соединенных строк: **Goodbye World.Welcome to the Basic Test.**

После завершения просмотра результатов закройте Run Results Viewer.

Как сопоставить данные из нескольких источников?

Диалоговое окно «Выбор исходной ссылки» можно создать ссылку на один или несколько из следующих источников данных для предоставления входных значений: **Доступные шаги**, **Столбец источника данных** и **Переменные теста**. В разделе выше использовался источник **Доступные шаги** для одного значения, а для другого значения данные были введены вручную.

Для использования нескольких источников данных в качестве значения свойства можно создать пользовательское выражение. В этом разделе диалоговое окно «Выбор исходной ссылки» используется для создания выражения для свойства **Суффикс**, которое использует вводимые вручную и автоматические значения из параметра **Доступные шаги**.

1 Задайте префикс.



На холсте выберите шаг **ConcatenateString**. Откройте представление **Ввод/Контрольные точки** на панели «Свойства». Щелкните ячейку **Значение** строки **Префикс** и щелкните символ **X**, чтобы очистить содержимое. Введите новый префикс Hello world.



2 Откройте диалоговое окно «Выбор исходной ссылки».



Щелкните ячейку **Значение** строки **Суффикс** и щелкните символ **X**, чтобы очистить содержимое. Нажмите кнопку **Создать ссылку на источник данных**. Откроется диалоговое окно «Выбор источника связи».



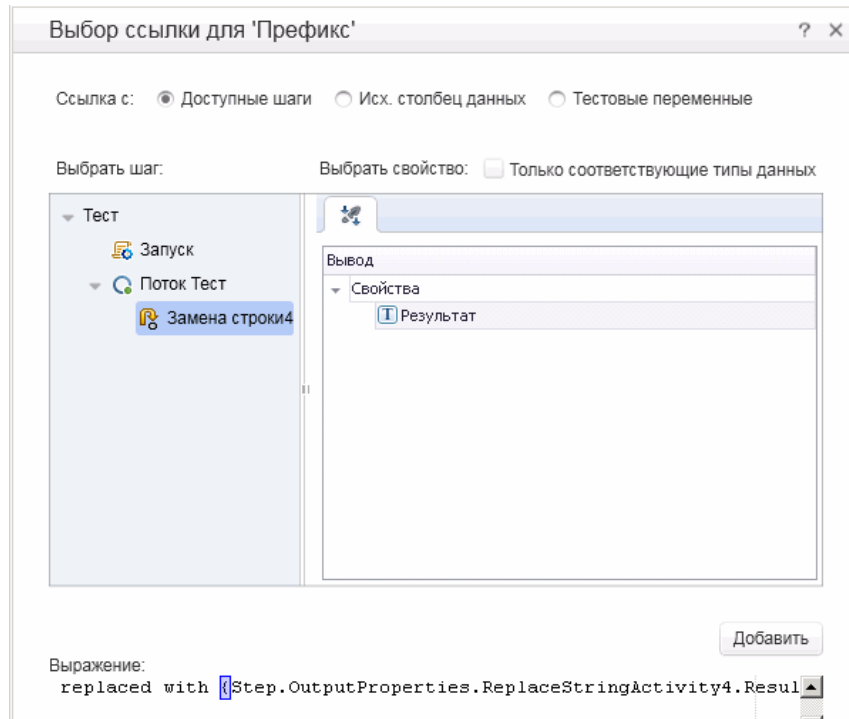
3 Измените суффикс.

В диалоговом окне «Выбор источника связи» нажмите **Пользовательское выражение**, чтобы развернуть его. В поле **Выражение** введите следующее: " было заменено с " (добавление пробела до и после фразы для повышения удобства чтения).



4 Добавление дополнительного источника

Установите флажок **Доступные шаги** и выберите узел **Изменить текст** на левой панели. Выберите узел **Результат** на правой панели и нажмите кнопку **Добавить**. В поле **Выражение** будут показаны оба источника.



5 Запуск теста и просмотр отчета



Нажмите кнопку **Выполнить**, чтобы выполнить тест. Разверните результаты и выберите узел **ConcatenateString**. В отчете будут показаны результаты соединенных строк.

The screenshot shows a window titled "Captured Data" with a scrollable table. The table has two columns: "Имя" (Name) and "Значение" (Value). The rows contain the following data:

Имя	Значение
Тип	HP.ST.Ext.BasicActivities.ConcatenateStringsActivity
ID шага	ConcatenateStringsActivity5
Сообщение	Успешно сцепленные строки
Префикс	' was replaced with Goodbye world'
Суффикс	'Welcome to the Basic Test'
Результат	' was replaced with Goodbye worldWelcome to the Basic Test'
Имя	'Соединение строк5'
Комментарий	"

6 Закройте окно «Просмотр результатов работы».

Как осуществляется управление шагом с помощью данных?

Управление с помощью данных — это назначение данных шагам теста из источника данных, например из файла Excel или XML-файла или из локальной таблицы. Задача управления с помощью данных заключается в запуске одинаковых бизнес-процессов с различными значениями. Благодаря этому обеспечивается возможность проверки используемого приложения в различных сценариях путем изменения только таблиц данных.

Управление шагами теста с помощью данных

1 Управление входными аргументами с помощью данных.



Выберите на холсте шаг **Изменить текст**. Откройте представление **Ввод/Контрольные точки** на панели «Свойства» и нажмите кнопку **Управление данными**. Откроется диалоговое окно «Управление данными».

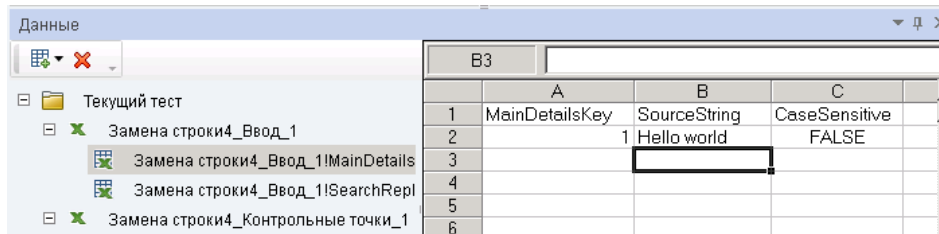
2 Укажите поставщика данных.

В диалоговом окне «Управление данными» выполните следующие действия.

- a Задайте тип **поставщика данных** в Excel.
- b Включите управление данными для **Входных и контрольных точек**.
- c Снимите флажок **Настроить 'Поток теста' как цикл ForEach с помощью нового источника данных**, который обеспечивает повторение потока теста в соответствии с числом строк данных. Число итераций будет задано вручную на более поздних шагах.
- d Нажмите кнопку **ОК**, чтобы закрыть диалоговое окно «Управление данными».
- e Примите всплывающее сообщение. Механизм управления данными заменяет значения констант новыми выражениями, {DataSource.Change Text_Input!MainDetails.SourceString}.

3 Просмотр панели «Данные»

Убедитесь, что панель «Данные» отображается. Если панель не отображается, выберите **Вид > Данные**. Разверните узел **Change Text_Input** и выберите узел **Change Text_Input!MainDetails**. На панели данных отображается таблица данных со столбцом для каждого входного свойства и одна строка значений, соответствующая свойству **Hello World**. и **FALSE** (или пустой флажок для установок без Excel), введенных ранее.



4 Добавление новых данных

Добавьте две дополнительные строки к таблице **Change Text_Input!MainDetails**. Убедитесь, что текст скопирован точно, включая имеющиеся знаки препинания.

MainDetailsKey	SourceString	CaseSensitive
1	Hello world.	FALSE
2	Я люблю брокколи.	TRUE
3	Версия продукта: 11.	FALSE

5 Добавьте новый поисковый запрос и замените данные.

Щелкните узел **Text_Input!SearchReplaceString** и добавьте две дополнительные строки к таблице. Убедитесь, что текст скопирован точно, включая имеющиеся знаки препинания.

MainDetailsKey	Ключ	Значение	CaseSensitive
1	Hello	Goodbye	FALSE
2	брокколи	мороженое	TRUE
3	11	12	FALSE

6 Добавьте значения контрольных точек.

Разверните узел **Изменить Text_Checkpoints** и выберите узел **Изменить Text_Checkpoints!MainDetails**. Добавьте значения к этому столбцу как показано ниже.

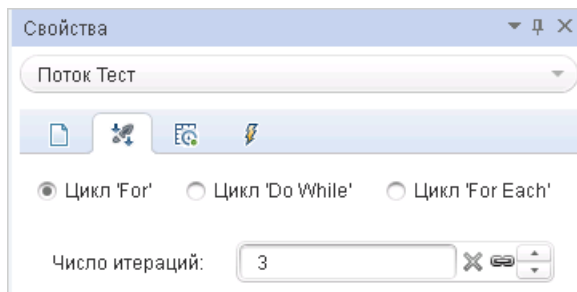
Примечание: В третьей строке мы намеренно добавили восклицательный знак ! для возникновения ошибки.

Результат
Прощай мир.
Я люблю мороженое.
Версия продукта: 12.

7 Настройка числа итераций

Число итераций — это число раз, которое должен быть повторен шаг. Это число будет задано равным 3 в соответствии с числом строк данных в данной таблице.

Перейдите обратно к холсту и щелкните внутри рамки «Ход теста», но не внутри шага теста. Откройте представление **Ввод/Контрольные точки** в панели «Свойства». Нажмите **Цикл For** и задайте **Число итераций** равным **3**.



8 Запуск теста и просмотр отчета



Чтобы скомпилировать и запустить тест, нажмите кнопку **Выполнить** либо клавишу F5. Тест будет выполнен три раза с использованием трех строк данных в таблице.

После выполнения теста откроется средство просмотра результатов (Run Results Viewer). Разверните узел **Ход теста** и перейдите к строке с красным символом **X**, который указывает на контрольную точку с ошибкой. В работе контрольной точки произошла ошибка, поскольку ожидаемый результат содержал знак восклицания, который отсутствовал в исходной строке.

9 Исправьте ошибку и повторно выполните тест.

На панели «Данные» исправьте данные узла **Change Text_Checkpoints!MainDetails**. В третьей строке столбца **Результаты** для контрольной точки замените восклицательный знак точкой.

Повторно запустите сценарий и убедитесь, что отчет не содержит ошибок.

Дальнейшие шаги

Теперь, после того, как было освоено создание простых шагов тестов, можно создавать шаги с помощью веб-служб. В следующих уроках будет описан процесс импорта WSDL и создания текстов веб-служб.

3

Тест веб-службы

Тестирование API UFT позволяет создавать тесты для веб-служб на основе WSDL.

Этот урок состоит из следующих разделов.

- Как импортировать веб-службу?
- Как создать тест веб-службы?
- Как интегрировать данные в тест?
- Как использовать несколько источников данных и пользовательский код?

Как импортировать веб-службу?

WSDL-файл определяет операции в веб-службе. Для использования WSDL-файла, его необходимо импортировать к тест. В этом разделе описывается импорт WSDL-файла примера приложения.

1 Запустите приложение приложения для бронирования авиабилетов.

Убедитесь, что служба приложения для бронирования авиабилетов доступна, как описано в разделе «Как выполняется запуск приложения?» на стр. 14.

2 Создание нового решения

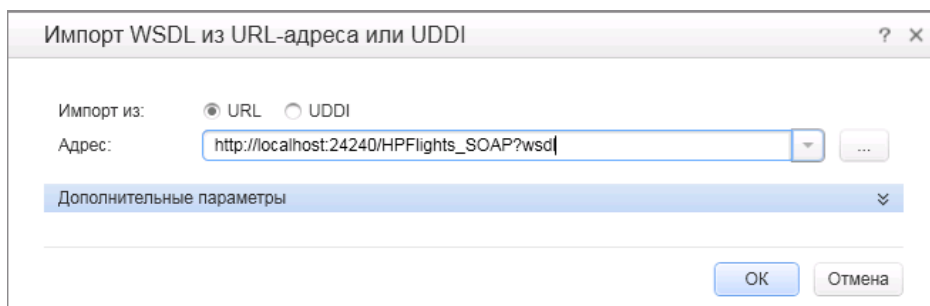
Выберите **Файл > Создать > Тест** и укажите имя **WebServiceTest** для теста API. Нажмите кнопку **Создать**.

3 Откройте диалоговое окно «Импорт службы».

Выберите **Импорт WSDL > Импорт WSDL из URL-адреса или UDDI** на панели инструментов.

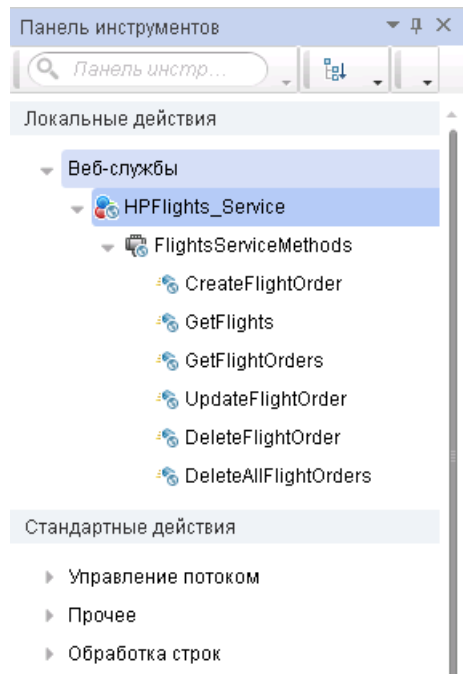
4 Укажите источник импорта.

Выберите параметр **URL-адрес** и укажите расположение **http://localhost:24240/HPFlights_SOAP?wsdl**. Нажмите кнопку **OK**.



5 Просмотрите операции службы.

При операции импорта будет создана новая ветвь операций веб-служб на **панели инструментов** в категории **Веб-службы**. Разверните узел для просмотра операций.



Как создать тест веб-службы?

В этом разделе создается новый заказ на бронирование авиарейса с помощью веб-службы **HPFlights**.

Для создания заказа на бронирование авиарейса сначала необходимо получить сведения о доступных авиарейсах. Сначала выполняется шаг **GetFlights**, в котором извлекаются все авиарейсы до нужного назначения. В следующем шаге теста первый возвращенный номер авиарейса используется как входные данные для шага **CreateFlightOrder**.

1 Создание шага **GetFlights**

Разверните узел **Веб-службы > HPFlights_Service** и перетащите действие **GetFlights** в поток теста.

2 Назначьте значения для **DepartureCity** и **ArrivalCity**.



Откройте представление **Ввод/Контрольные точки** и разверните узел **Тело > GetFlights**. Чтобы выбрать город, щелкните стрелку в строке, чтобы открыть раскрывающийся список. Выберите **Денвер** в качестве **DepartureCity** и **Лос-Анджелес** в качестве **ArrivalCity**.

Свойства ▼ 🔍 ✕

GetFlights ▼

📄

📄 Загрузить XML

Ввод	Значение
Envelope	
Header	
Any (массив)	
Body	
GetFlights	
DepartureCity	Denver
ArrivalCity	Los Angeles

Загрузить XML

Контрольные точки	Валид...	Ожидаемое значение
Envelope	<input type="checkbox"/>	=
Header	<input type="checkbox"/>	=
Any (мас [F	<input type="checkbox"/>	=
Body	<input type="checkbox"/>	=
GetFlightsResponse	<input type="checkbox"/>	=
GetFlightsResult	<input type="checkbox"/>	=

XML XPath ▼

Отправить запрос в службу Проверить структуру

3 Создание шага CreateFlightOrder.

Перетащите действие **CreateFlightOrder** с панели инструментов в поток теста, ниже шага **GetFlights**.

4 Задание значений для шага CreateFlightOrder



В представлении **Ввод/Контрольные точки** разверните узел **Тело > CreateFlightOrder > FlightOrder** и задайте значения для создания авиарейса.

- **Класс** — выберите класс, например, Бизнес в раскрывающемся списке.
- **CustomerName** — любое значение
- **DepartureDate** — используйте раскрывающийся список, чтобы открыть календарь и выбрать дату в будущем (не менее двух дней от сегодняшней даты).
- **FlightNumber** — пока оставьте пустым. Оно будет задано в следующих шагах.
- **NumberOfTickets** — используйте полосу прокрутки для задания любого значения.

5 Свяжите выходные данные GetFlights с шагом CreateFlightOrder.



a Щелкните значок **Создать ссылку на источник данных** в правом углу строки **FlightNumber**. Откроется диалоговое окно «Выбор источника связи».

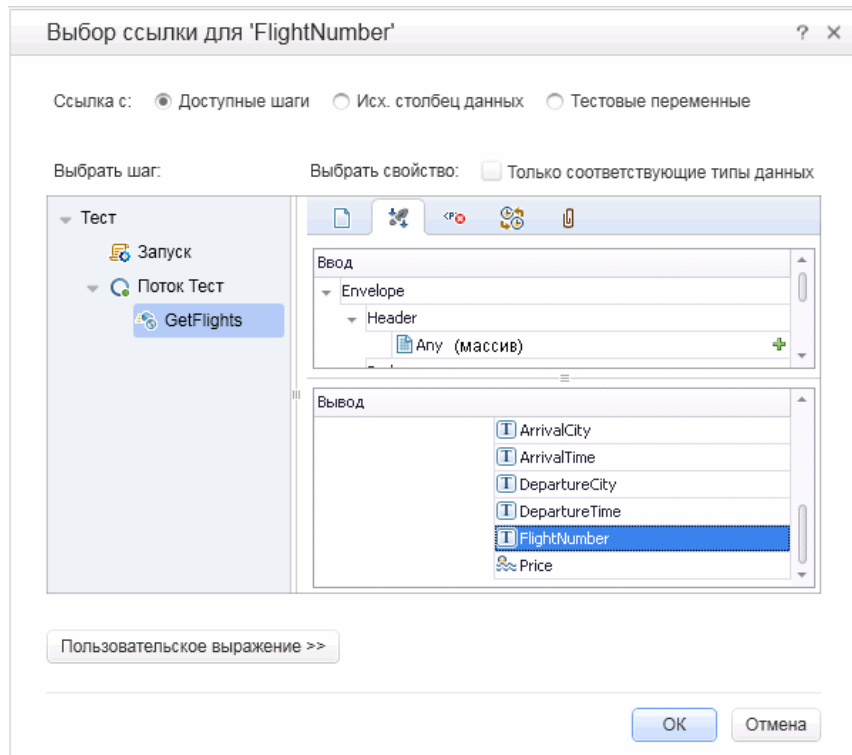
b Выберите **Доступные шаги** и выберите узел **GetFlights**.

c На правой панели нажмите кнопку **Ввод/Контрольные точки**. В разделе **Выходные данные** разверните все узлы ниже узла **Тело**.

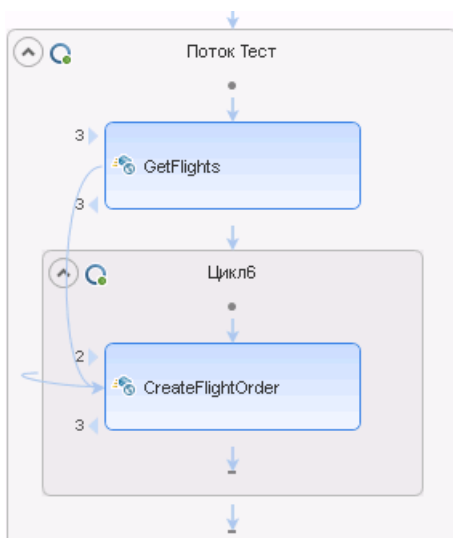


Нажмите кнопку **Добавить** в строку узла **Flight (массив)** для создания массива **Flight[1]**.

- d Разверните массив **Flight[1]**, выберите элемент **FlightNumber** и нажмите **OK**. В приложении отображается запрос на необходимость заключения шага назначения в цикл. Нажмите **Да**.



Холст указывает на соединение между двумя шагами.



6 Сброс числа итераций

Число итераций — это число раз, которое должен быть повторен шаг. Перейдите обратно к холсту и щелкните внутри рамки «Поток теста», но не внутри шага теста. Откройте представление **Ввод** панели «Свойства». Нажмите **Цикл For** и задайте **Число итераций** равным **1**.



7 Запустите тест.

Нажмите кнопку **Выполнить**. Проверьте журнал на вкладке **Вывод**. Автоматически откроется Run Results Viewer.



8 Проверка результатов

На левой панели щелкните родительский узел и выберите **Развернуть все** в контекстном меню. Щелкните узел **CreateFlightOrder**. На панели **Записанные данные** с помощью прокрутки перейдите к разделу снимка HTTP вызова веб-службы и просмотрите панель «Ответ». Обратите внимание на выходные данные запроса: **OrderNumber** и **TotalPrice**. Скопируйте значение **TotalPrice** в буфер для использования в следующем шаге.

Captured Data	
Запрос	Ответ
<p>Заголовок HTTP</p> <pre> HP.SO&Q.SampleApp/IHPFlights_Service/CreateFlightOrder Content-Type: text/xml; charset=utf-8 Host: localhost:24240 Port: 411 Content-Length: 335 Date: Thu, 21 Mar 2013 17:17:23 GMT Server: Microsoft-HTTPAPI/2.0 </pre>	<p>Заголовок HTTP</p> <pre> Connection: close Content-Length: 335 Content-Type: text/xml; charset=utf-8 Date: Thu, 21 Mar 2013 17:17:23 GMT Server: Microsoft-HTTPAPI/2.0 </pre>
<p>SOAP</p> <pre> <?xml xmlns="http://schemas.xmlsoap.org/soap/envelope/"> <Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/"> <Body> <CreateFlightOrderResponse xmlns="HP.SO&Q.SampleApp"> <CreateFlightOrderResult xmlns="HP.SO&Q.SampleApp"> <Class>Business</Class> </pre>	<p>SOAP</p> <pre> <?xml xmlns="http://schemas.xmlsoap.org/soap/envelope/"> <Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/"> <Body> <CreateFlightOrderResponse xmlns="HP.SO&Q.SampleApp"> <CreateFlightOrderResult xmlns="HP.SO&Q.SampleApp"> <OrderNumber>17</OrderNumber> </pre>

Совет. Щелкните ссылку **Запрос** или **Ответ**, чтобы открыть SOAP в другом браузере.

После завершения просмотра результатов закройте Run Results Viewer.

9 Задайте контрольную точку.

Выберите шаг **CreateFlightOrder** на полотне. Откройте представление **Ввод/Контрольные точки** панели «Свойства», щелкните сетку «Контрольные точки» и разверните узел **CreateFlightOrderResponse**. Вставьте содержимое буфера из предыдущего шага в поле **TotalPrice**. Установите флажок в строке **TotalPrice**, чтобы включить ее как контрольную точку.

10 Выполните тест и просмотрите результаты контрольных точек.

Повторно выполните тест и разверните дерево результатов. Выберите узел **Контрольные точки** для **CreateFlightOrder**. В отчете отображается контрольная точка и указываются ожидаемые и фактические значения. Если ожидаемое значение не возвращено сервером, в отчете указывается ошибка.

Имя	Результат	Свойство	Фактический результат	Стиль оценки	Ожидаемые значения	Сведения
"Контрольная точка 1"	✓	""	""	Структурная проверка	""	""
"Контрольная точка 2"	✓	"Envelope[1]/Body [1]/CreateFlightOrderResponse [1]/CreateFlightOrderResult [1]/TotalPrice[1]"	"232"	=	"232"	""

После завершения просмотра результатов закройте Run Results Viewer.

Как интегрировать данные в тест?

В этом разделе описывается интеграция данных из существующего источника и управление теста с помощью данных. При управлении теста с помощью данных, на панели «Данные» автоматически создается таблица данных с редактируемыми значениями.

1 Импорт образца данных

На панели «Данные» в нижней части окна UFT выберите **Создать > Excel**. Откроется диалоговое окно «Добавить новый источник данных Excel».

- a Перейдите к Excel-файлу примера приложения **SampleAppData.xlsx** в папке *<каталог_установки>/SampleApplication*. По умолчанию задается папка **C:\Program Files\HP\Unified Functional Testing\SampleApplication**.
- b Включите параметр **Файл Excel содержит строку заголовка**, поскольку пример файл содержит строку заголовка.
- c Введите **WS_Flights** в качестве **имени источника данных**.

- d Выберите **Источник данных**, на который указывает ссылка в качестве режима импорта. При этом создается ссылка на Excel-файл в его исходном местоположении и поэтому при изменении данных, источник данных будет актуальным.
- e Нажмите кнопку **ОК**.

Создать источник данных Excel

Имя источника данных: * WS_Flights

Путь к файлу Excel: * C:\Program Files (x86)\HP\Unified Functiona ...

Файл Excel содержит строку заголовка

Расположение файла:

Создать копию Excel-файла

Установить связь с Excel-файлом в его исходном расположении

Разрешить другим средствам перезаписывать данные

ОК Отмена

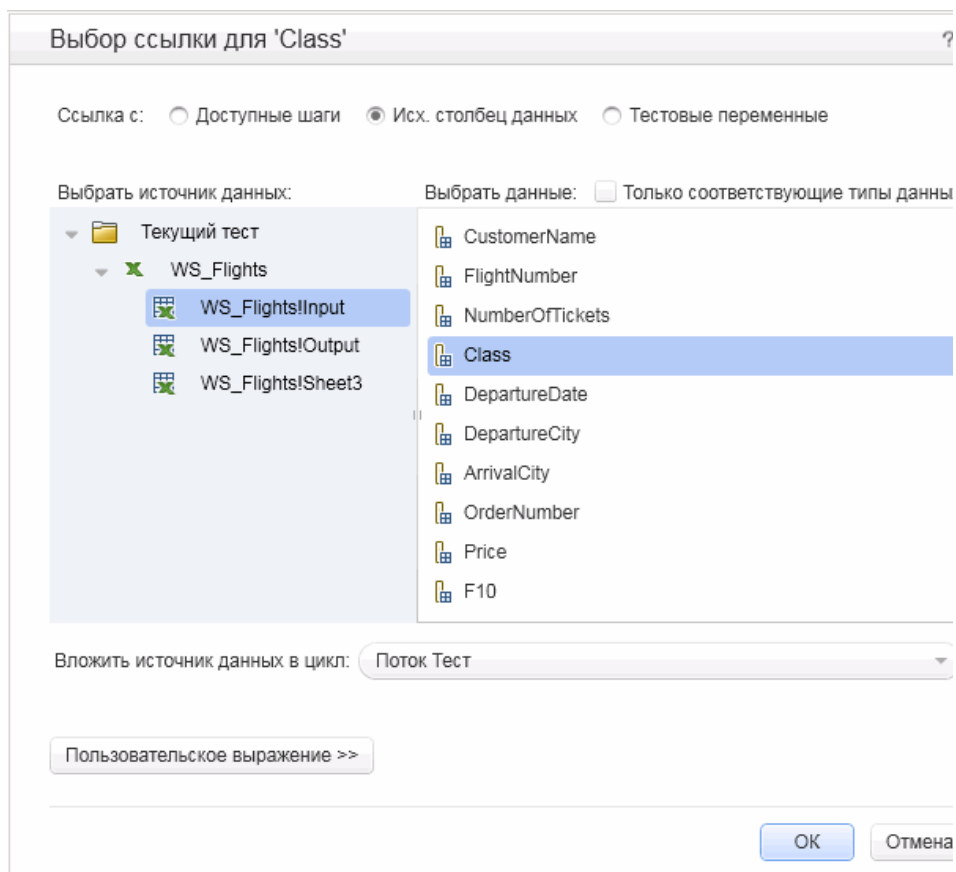
2 Откройте диалоговое окно «Выбор ссылки на данные».

Выберите шаг на холсте **CreateFlightOrder** и откройте представление **Ввод/Контрольные точки**. В разделе **Входные данные** разверните все узлы и выберите строку **Класс**. Щелкните значок **Ссылка на источник данных**. Откроется диалоговое окно «Выбор источника связи».



3 Извлечение значения из источника данных

Выберите параметр **Столбец источника данных**.



4 Используйте образец данных Excel.

Выберите узел **WS_Flights!Input** и на правой панели выберите **Класс**. Нажмите кнопку **OK**. После этого тест будет ссылаться на этот столбец в образце данных при выполнении теста.

Повторите эту процедуру для других входных параметров: **CustomerName**, **DepartureDate**, **FlightNumber** и **NumberofTickets**.

5 Отключение контрольных точек

В представлении **Ввод/Контрольные точки** щелкните сетку «Контрольные точки». Снимите флажок для свойства **TotalPrice**, чтобы исключить его как контрольную точку.

6 Установка настроек перехода

Настройки перехода позволяют указать способ использования данных в текущем источнике данных. Можно указать строку для начала, число строк для перехода и направление для перемещения для следующего набора значений. Также можно настроить действия, выполняемые при достижении конца таблицы данных: циклический переход или продолжение использования последней строки.

- a На холсте щелкните **Поток теста**, но не в шаге.
- b На панели «Свойства» нажмите кнопку представления **Источники данных**.
- c Выберите узел **WS_Flights!Input** и щелкните **Изменить**, чтобы открыть диалоговое окно «Переход по данным».



- d Укажите сведения о переходе по данным: **Начать с:** первая строка, **перейти:** перейти на 3 строки вперед, **закончить в:** последней строке и **при достижении последней строки:** Циклический переход.

Переход по данным

Начало

Начать с: Первая строка

Строка: 1

Переместить

Переместить на: 3 строк(у) Вперед

Конец

Закончить на: Последняя строка

Строка: 7

По достижении последней строки

Действие: Циклический переход

ОК Отмена

- e Нажмите кнопку **ОК**.

7 Выполнение теста и проверка результатов



Нажмите кнопку **Выполнить** и проанализируйте результаты в окне вывода. Автоматически откроется Run Results Viewer. Разверните дерево результатов и выберите шаг **CreateFlightOrder**. С помощью прокрутки перейдите вниз на вкладку **Записанные данные** и обратите внимание на данные из Excel-файла в запросе SOAP (левая панель) и на результаты в ответе SOAP (правая панель).

SOAP	SOAP
<pre> http://schemas.xmlsoap.org/soap/envelope/ ... <CreateFlightOrderResponse xmlns="HP.SOAQ.SampleApp"> <Class> <CustomerName>John Freeman</CustomerName> <DepartureDate>2015-10-10T00:00:00</DepartureDate> <FlightNumber>1042</FlightNumber> <NumberOfTickets>1</NumberOfTickets> </Class> </CreateFlightOrderResponse> </pre>	<pre> <s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"> <s:Body> <CreateFlightOrderResponse xmlns="HP.SOAQ.SampleApp"> <CreateFlightOrderResult xmlns:i="http://schemas.xmlsoap.org/soap/encoding/"> <OrderNumber>22</OrderNumber> <TotalPrice>125</TotalPrice> </CreateFlightOrderResult> </CreateFlightOrderResponse> </s:Body> </s:Envelope> </pre>

После завершения просмотра результатов закройте Run Results Viewer.

Как использовать несколько источников данных и пользовательский код?

В этом разделе описывается определение данных с помощью нескольких источников данных и отправка данных в отчет с помощью шага пользовательского кода.

1 Создайте новый тест.

Создайте новый тест с именем **WebServicesCustom** и импортируйте WSDL служб **HP Flights**, как описано в «Как импортировать веб-службу?» на стр. 34.

2 Создание шагов тестов

Перетащите действия на холст в следующем порядке. Из папки **Веб-службы**: **GetFlights** и **CreateFlightOrder**. Из папки **Прочее** перетащите в **Пользовательский код**.

3 Добавление источника данных

На панели «Данные» выберите **Создать > Excel**. В диалоговом окне «Добавить новый источник данных Excel» выполните следующие действия.

- a Перейдите к Excel-файлу примера приложения. По умолчанию задается папка **C:\Program Files\HP\Unified Functional Testing\SampleApplication**.
- b Выберите **файл Excel, содержащий флажок строки заголовка**.
- c Введите **WS_Flights** в качестве **имени источника данных**.
- d Выберите режим **Источник данных, на который указывает ссылка**.

4 Назначьте значения для GetFlights.

Выберите шаг **GetFlights** на холсте и откройте представление **Ввод/Контрольные точки** на панели «Свойства». В разделе **Входные данные** выберите **DepartureCity= Денвер** и **ArrivalCity=Лос-Анджелес**.



5 Назначение значений для CreateFlightOrder

Выберите действие на холсте **CreateFlightOrder** и откройте представление **Ввод/Контрольные точки**. Разверните узел **тело > FlightOrder** и задайте входные свойства следующим образом:



- **Класс.** Эконом
- **CustomerName.** Щелкните кнопку Создать ссылку на источник данных в правом углу строки **CustomerName**. Откроется диалоговое окно «Выбор источника связи». Выберите **Столбец источника данных** и разверните дерево, чтобы отобразить узел **WS_Flights!Input**. На правой панели выберите параметр **CustomerName**. Нажмите кнопку **OK**.
- **DepartureDate.** Дата в формате ГГГГ-ММ-ДДТЧЧ:ММ:СС Пример: 2015-02-18T00:00:00. Чтобы открыть календарь используйте раскрывающийся список. Дата должна быть в будущем (спустя не менее двух дней после текущего).
- **NumberofTickets.** 3



► **FlightNumber.** Ссылка из предыдущего шага



a Нажмите кнопку **Создать ссылку на источник данных** в правом углу строки **FlightNumber**.

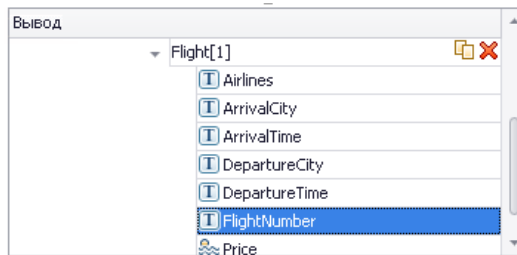
b В диалоговом окне «Создать ссылку на источник данных», выберите **Доступные шаги**, разверните ветвь **Поток теста** и щелкните **GetFlights**.



c На правой панели выберите кнопку **Ввод/Контрольные точки**.



d В разделе **Выходные данные** нажмите кнопку **Добавить** в строке узла **Flight (массив)** для создания массива **Flight[1]**. Разверните массив, выберите **FlightNumber** и нажмите кнопку **ОК**.



6 Создание свойства для шага пользовательского кода

Выберите действие **Пользовательский код** на холсте и откройте представление **Ввод/Контрольные точки** на панели «Свойства». Разверните кнопку на панели инструментов **Добавить свойство** и выберите **Добавить входное свойство**. Создайте новое свойство типа **String** с именем **FlightInfo**.

7 Определите значения для шага пользовательского кода.

В этом шаге выполняется определение значения с использованием нескольких значений. В этом примере задается значение, которое является сочетанием **CustomerName**, строки константы и **OrderNumber**.



a Нажмите кнопку **Создать ссылку на источник данных** в правом углу строки **FlightInfo**. Откроется диалоговое окно «Выбор источника связи».

b Нажмите **Пользовательское выражение**, чтобы открыть область «Выражение».

c Выберите параметр **Столбец источника данных**. В узле дерева **WS_Flights!Input** выберите **CustomerName**. Нажмите **Добавить**.

- d В области **Выражение** введите `_OrderNumber_` (со знаками подчеркивания) после существующих выражений.
- e Выберите **Доступные шаги** и разверните ветвь **Поток теста**. Выберите узел **CreateFlightOrde**. На правой панели выберите кнопку **Ввод/Контрольные точки**. На нижней панели разверните узел **Вывод Тело**, разверните дерево, выберите элемент **OrderNumber** и щелкните **Добавить**.



У входного свойства `CustomCode` с именем **FlightInfo** имеется следующее значение:

```
{DataSource.WS_Flights!Input.CustomerName}_OrderNumber_{Step.OutputProperties.StServiceCallActivity(x).Body.CreateFlightOrderResponse.CreateFlightOrderResult.OrderNumber}
```

Нажмите кнопку **ОК**, чтобы закрыть диалоговое окно.

8 Создание события

В этом шаге будет создаваться обработчик событий для использования прикладного программного интерфейса (API). К этому разделу можно добавить код C#. Определение событий позволяет адаптировать тест к нестандартным требованиям и выполнить действия, которые не встроены в UFT. В этом примере добавляется код, передающий пользовательскую строку в отчет.



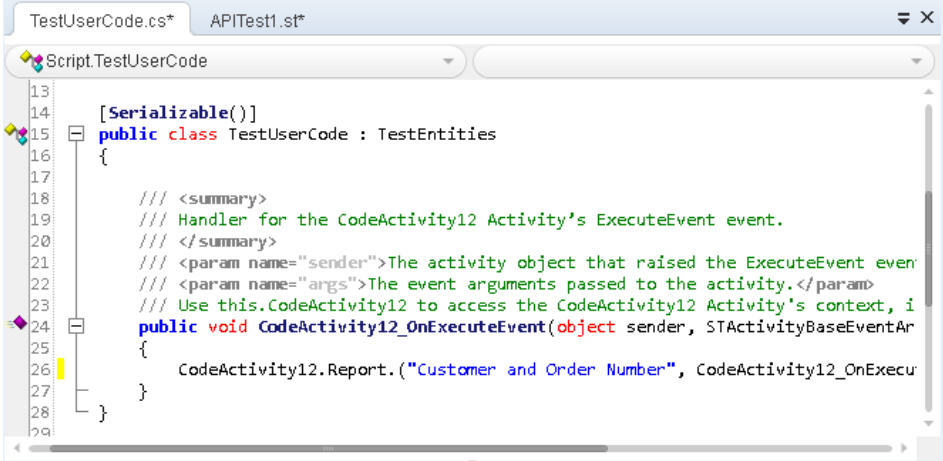
Выберите шаг **Пользовательский код** на холсте. На панели «Свойства» нажмите кнопку **События**. В строке **ExecuteEvent** щелкните раскрывающийся список и выберите **Создать обработчик по умолчанию**. UFT создает событие с именем **CodeActivity(x)_OnExecuteEvent** и открывает новую вкладку **TestUserCode.cs**.

9 Отредактируйте раздел "Todo", используя индекс, назначенный операции.

Замените текст, к которому добавлены комментарии, в разделе **Todo** на следующий:

```
CodeActivity(x).Report("Customer and Order  
Number",CodeActivity(x).Input.FlightInfo);
```

В следующем примере событию назначен индекс 12, поэтому строка будет выглядеть следующим образом: `CodeActivity12.Report("Customer and Order Number",CodeActivity12.Input.FlightInfo);`



```

13
14 [Serializable()]
15 public class TestUserCode : TestEntities
16 {
17
18     /// <summary>
19     /// Handler for the CodeActivity12 Activity's ExecuteEvent event.
20     /// </summary>
21     /// <param name="sender">The activity object that raised the ExecuteEvent even
22     /// <param name="args">The event arguments passed to the activity.</param>
23     /// Use this.CodeActivity12 to access the CodeActivity12 Activity's context, i
24     public void CodeActivity12_OnExecuteEvent(object sender, STActivityBaseEventAr
25     {
26         CodeActivity12.Report("Customer and Order Number", CodeActivity12_OnExecu
27     }
28 }
29

```

1 Выполните тест и проверьте результаты.

Перейдите к результатам в шаге **Пользовательский код**. Обратите внимание на новую запись на панели **Записанные данные**: Клиент и номер заказа.

Совет. Также можно использовать действие **Сообщение отчета** в папке **Прочее** для отправки текста и значений свойств в отчет.

Дальнейшие шаги

Теперь, после изучения создания теста для веб-службы, можно соотнести эти данные с другими типа служб и компонентов приложений. В следующем уроке будет пошагово описываться процесс создания теста для службы REST.

4

Тестирование служб REST

С помощью UFT можно моделировать и создавать тесты для служб REST.

Этот урок состоит из следующих разделов.

- Как создать операцию службы REST?
- Как запустить тест REST?
- Как назначить данные используемому методу REST?
- Как проверить полученные данные?
- Как разрешить изменения в используемой службе REST?

Как создать операцию службы REST?

В этом разделе описывается моделирование действия службы REST с помощью примера приложения. При работе со службами REST для них рекомендуется создать метод прототипа, который можно использовать повторно. После создания этого метода его можно повторно использовать на различных шагах теста или в отдельных тестах.

1 Запустите приложение приложения для бронирования авиабилетов.

Убедитесь, что служба приложения для бронирования авиабилетов запущена, как описано в разделе «Как выполняется запуск приложения?» на стр. 14.

2 Получите документ моделирования службы REST.

В окне командной строки примера приложения введите **h** и нажмите ENTER. Откроется браузер с данными моделирования для службы REST. Этот файл **index.htm**, находится в папке **<installation_folder>\SampleApplication\Help** folder.

3 Сохранить текст запроса в файл.

Скопируйте текст запроса для **FlightOrders > ReserveOrder (POST)** в буфер. Скопируйте только XML-код.

```
<FlightOrderDetails xmlns="HP.SOAQ.SampleApp">
  <Класс>Бизнес</Class>
  <CustomerName>John Doe</CustomerName>
  <DepartureDate>2012-12-12</DepartureDate>
  <FlightNumber>1304</FlightNumber>
  <NumberOfTickets>21</NumberOfTickets>
</FlightOrderDetails>
```

Создайте новый файл в текстовом редакторе и вставьте содержимое буфера. Сохраните файл как **body.xml** в любое расположение.

4 Получите URL-адрес конечной точки службы REST.

Перейдите обратно в окно браузера и скопируйте URL-адрес для операции **FlightOrders > ReserveOrder (POST)**, http://localhost:24240/HPFlights_REST/FlightOrders/, в буфер.

5 Создайте новый тест.

- a** Выберите **Файл > Создать > Тест** и выберите типы **Тест API**.
- b** Укажите имя **RESTServiceTest** для нового теста.
- c** Нажмите кнопку **Создать**.

6 Добавьте службу REST.



- a** Нажмите кнопку **Добавить службу REST**. Откроется диалоговое окно «Добавить службу REST».
- b** На левой панели щелкните узел **Создать службу** и переименуйте его в **SampleRESTService**.

7 Добавление ресурса

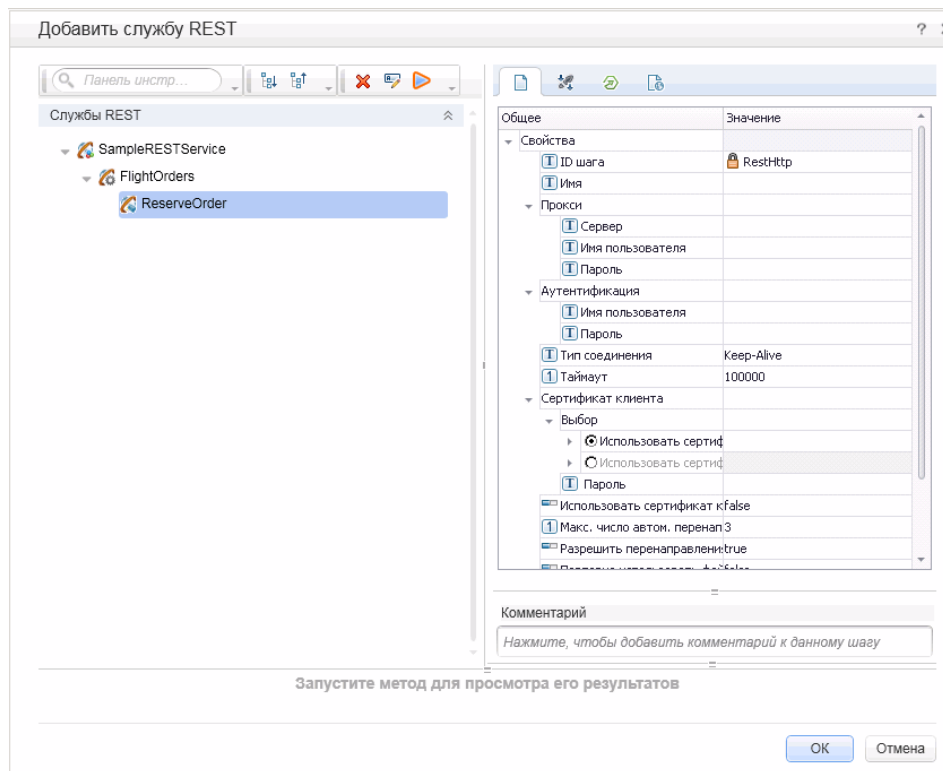


Нажмите кнопку панели инструментов **Добавить ресурс** и переименуйте ресурс в **FlightOrders**.

8 Добавьте метод к службе REST.



Нажмите кнопку **Добавить метод** и переименуйте метод к **ReserveOrder**.



9 Настройте свойства HTTP метода.



Откройте представление **Ввод/контрольные точки HTTP** на правой панели.

a Вставьте содержимое буфера в строку **URL-адрес**.

b Задайте для **Метода HTTP** значение **POST**.



c Нажмите кнопку **Добавить** в строке **Заголовки запроса (массив)**, чтобы добавить элемент массива.

d Разверните массив **Заголовки запроса**. С помощью сведений о методе со страницы справки задайте имя и значение заголовка следующим образом:

- **Имя** строка — Content-Type
- **Значение** строка — text/xml

Ввод	Значение
Свойства	
URL	http://localhost:24240/HPFlights_RES
Метод HTTP	POST
Версия HTTP	1.1
RequestHeaders (массив)	
RequestHeaders[1]	
Имя	Content-Type
Значение	text/xml

10 Создание входных свойств



a Откройте представление **Ввод/Контрольные точки** на правой панели.



b Выберите **Добавить > Добавить входное свойство**.

c Добавьте свойство типа **String** с именем **Class**.

d Добавьте другое свойство типа **String** с именем **Customer_Name**.

e Добавьте другое свойство типа **Date/Time** с именем **Departure_Date**.

f Добавьте свойство типа **Int** с именем **Flight_Number**.

g Добавьте другое свойство типа **Int** с именем **Number_of_Tickets**.

11 Создание выходных свойств



a Выберите **Добавить > Добавить выходное свойство**.

b Добавьте свойство типа **Int** с именем **Total_Price**.

c Добавьте другое свойство типа **Int** с именем **Order_Number**.

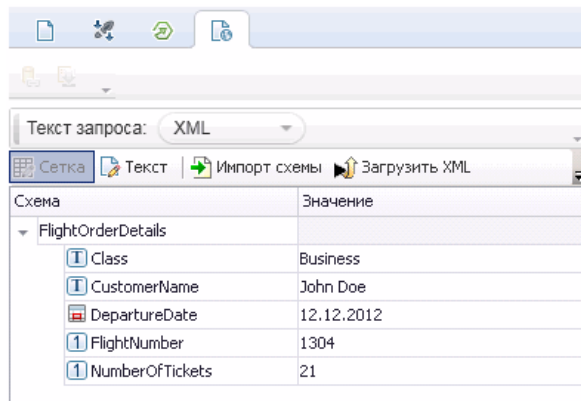
12 Импорт тела запроса



a Откройте представление **HTTP** на правой панели.

b Выберите **XML** в качестве типа **Тела**.

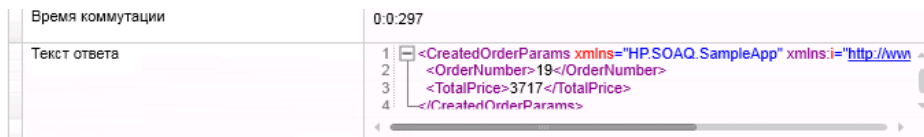
- с Нажмите кнопку **Загрузить XML** и загрузите файл **body.xml**, сохраненный ранее.



13 Тестирование метода



Нажмите кнопку панели инструментов **Выполнить метод**, чтобы проверить допустимость метода. С помощью прокрутки просмотрите результаты и убедитесь, что тело ответа содержит номер заказа и цену.

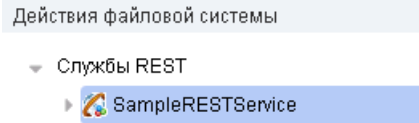


14 Добавление метода на «Панель инструментов»

Нажмите кнопку **ОК** в диалоговом окне «Добавить службу REST». Конструктор служб REST добавляет службу REST с ее ресурсами и методами на **Панель инструментов** в категорию **Локальные действия**.

15 Предоставьте общий доступ к действию REST, чтобы сделать действие доступным для всех тестов.

На **Панели инструментов** выберите родительский узел службы REST, **SampleRESTService**, и выберите **Переместить в > Действия файловой системы** в контекстном меню. Действие службы REST перемещается в раздел **Действия файловой системы** на **Панели инструментов**.



Создание действия прототипа для службы REST с входными параметрами и данными HTTP завершено. Теперь действие можно перетащить в тесты и запускать практически без вмешательства пользователя.

Как запустить тест REST?

В этом разделе описан запуск метода службы REST, созданный в предыдущем разделе. Данные будут включаться в тест с помощью файла образца данных, включенного в продукт.

1 Создание шага теста

Дважды щелкните **Действия файловой системы > Службы REST > SampleRESTService > FlightOrders > метод ReserveOrder**, чтобы добавить его к **Потоку теста** на холсте.

2 Разверните внутренний HTTP-запрос оболочки метода REST.

Щелкните стрелку на левой стороне рамки метода **ReserveOrder** для просмотра внутренней рамки **HTTP-запрос**. Щелкните в рамке **HTTP-запроса** и откройте представление **HTTP** на панели «Свойства».



Значение свойств — это значения, импортированные ранее из файла **body.xml**. Эти значения были использованы при только что выполненном запуске теста.

The screenshot shows a test runner interface. On the left, a test flow diagram titled "Поток Тест" (Test Flow) contains a step named "ReserveOrder4". Inside this step, an "HTTP-запрос" (HTTP Request) step is highlighted with a yellow border. On the right, the "Свойства" (Properties) panel is open, showing a table of properties for the selected HTTP request.

Схема	Значение
FlightOrderDetails	
Class	Business
CustomerName	John Doe
DepartureDate	12.12.2012
FlightNumber	1304
NumberOfTickets	21

3 Запустите тест.



Выберите **Выполнить > Выполнить**, чтобы запустить тест.

4 Просмотрите отчет.



На левой панели Run Result Viewer нажмите кнопку **Развернуть все** или выберите **Развернуть все** в контекстном меню и выберите узел **HTTP-запрос**.

На панели **Записанные данные** щелкните ссылку **ResponseBody**, чтобы открыть ответ в отдельном окне браузера.

Captured Data	
RequestHeader_Content-Type	text/xml
RequestHeader_Host	localhost:24240
RequestHeader_Content-Length	314
RequestHeader_Expect	100-continue
RequestHeader_Connection	Keep-Alive
ResponseHeader_Content-Length	2418
ResponseHeader_Content-Type	text/html
ResponseHeader_Date	Thu, 21 Mar 2013 18:35:57 GMT
ResponseHeader_Server	Microsoft-HTTPAPI/2.0
ResponseBody	<?xml version="1.0" ... DIV> </BODY></HTML>

Убедитесь, что **Тело ответа** содержит значения для элементов **OrderNumber** и **TotalPrice**. Это соответствует описанию операции на странице справки службы REST, на которой указано следующее: Создается новый заказ на бронирование авиарейса и возвращаются OrderNumber и TotalPrice.

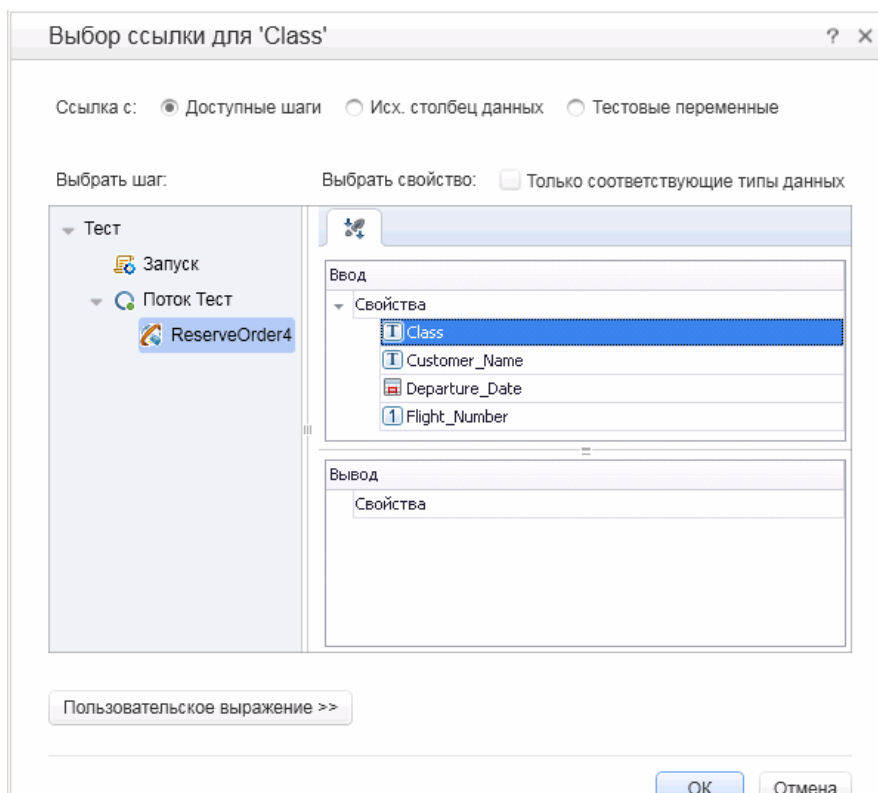
```
<CreatedOrderParams xmlns="HP.SOAQ.SampleApp"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <OrderNumber>18</OrderNumber>
  <TotalPrice>3717</TotalPrice>
</CreatedOrderParams>
```

Как назначить данные используемому методу REST?

Для назначения данных этому методу будет создана ссылка внутренних свойств HTTP-запроса на внешние свойства оболочки REST. Благодаря этому обеспечивается возможность управления свойствами сообщения запроса с более высокого уровня, т.е. уровня метода REST.

Назначение данных шагу метода REST

- 1 Выберите внутреннюю рамку **HTTP-запроса** и откройте представление **HTTP** на панели «Свойства».
- 2 Нажмите кнопку **Создать ссылку на источник данных** в правом углу строки **Class**. Откроется диалоговое окно «Выбор источника связи».
- 3 Выберите **Доступные шаги** в качестве источника данных, выберите узел **ReserveOrder** на левой панели и выберите свойство **Class** на правой панели. Нажмите кнопку **OK**.



- 4 Повторите описанный выше шаг для других внутренних свойств: **CustomerName**, **DepartureDate**, **FlightNumber** и **NumberOfTickets**. В полученном HTTP-представлении будут отображены новые ссылки.

Схема	Значение
FlightOrderDetails	
Class	{Step.InputProperties.RESTActivity5.Clas
CustomerName	{Step.InputProperties.RESTActivity5.Cus
DepartureDate	{Step.InputProperties.RESTActivity5.Dep
FlightNumber	{Step.InputProperties.RESTActivity5.Flig
NumberOfTickets	{Step.InputProperties.RESTActivity5.Nu

5 Импорт образца данных

Будет импортирован источник данных, значения которого будут использоваться для свойств метода REST.



- a На панели **Данные**, расположенной в нижней части окна, разверните кнопку «Источник данных» и выберите **Excel**. Откроется диалоговое окно «Создать источник данных Excel».



- b Нажмите кнопку браузера, расположенную рядом с полем **Путь к файлу Excel** и перейдите к Excel-файлу **SampAppData** примера приложения в папке *<каталога установки >\SampleApplication*.

- c Установите флажок **Файл Excel содержит строку заголовка**, поскольку пример содержит строку заголовка.

- d В поле **Имя источника данных** введите REST_Flights.

- e Установите флажок **Создать копию Excel-файла**. При этом сохраняется копия файла данных с тестом.

- f Нажмите кнопку **ОК**. Данные импортируются на панель «Данные».

6 Создание ссылки свойств метода REST на источник данных

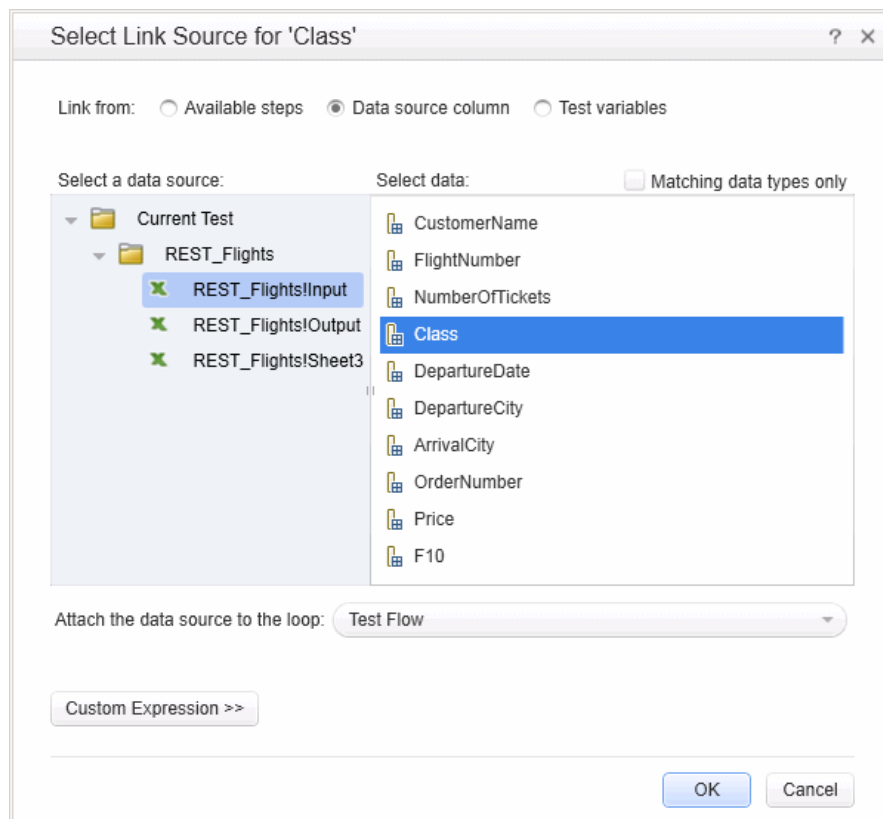


- a Выберите внешнюю рамку **ReserveOrder** и откройте представление **Ввод/Контрольные точки** на правой панели.



- b Нажмите кнопку **Создать ссылку на источник данных** в правом углу строки **Class**. Откроется диалоговое окно «Выбор источника связи».

- c Выберите **Источник столбца данных** в качестве источника данных, выберите узел **ReserveOrder** на левой панели и выберите свойство **Class** на правой панели. Нажмите кнопку **OK**.



- d Повторите описанный выше шаг для других внутренних свойств: **Customer_Name**, **Departure_Date** и **Flight_Number**.
- e Укажите значение 2 для **Number_of_Tickets**.

f В полученном представлении свойств будут отображены новые ссылки.

Ввод	Значение
Свойства	
Class	{DataSource.SampleAppDataInput.Class}
Customer_Name	{DataSource.SampleAppDataInput.CustomerNam}
Departure_Date	{DataSource.SampleAppDataInput.DepartureDate}
Flight_Number	{DataSource.SampleAppDataInput.FlightNumber}
Number_of_Tickets	2

7 Настройка итераций

Настройка итерация позволяет определить, как данный метод REST использует несколько наборов данных из одного источника данных.



a Выберите рамку **Поток теста** на холсте и откройте представление **Ввод/Контрольные точки** на правой панели.

b Настройте **Цикл For** с 3 итерациями.

Цикл 'For' Цикл 'Do While' Цикл 'For Each'

Число итераций: 3

8 Запустите тест.



Выберите **Выполнить > Выполнить**, чтобы запустить тест.

9 Убедитесь, что в запросе используются данные таблицы.



Запустите тест. На левой панели Run Result Viewer нажмите кнопку **Развернуть все** и выберите узлы **HTTP-запрос**. Щелкните ссылку **Тело запроса** на панели «Записанные данные». Обратите внимание, что в тесте использовались данные из панели «Данные» для назначенных свойств: Class, CustomerName, DepartureDate и FlightNumber.

```
- <FlightOrderDetails xmlns="HP.SOAQ.SampleApp">  
<Класс>Бизнес</Class>  
<CustomerName>John Freeman</CustomerName>  
<DepartureDate>2012-12-12</DepartureDate>  
<FlightNumber>1304</FlightNumber>  
<NumberOfTickets>21</NumberOfTickets>  
</FlightOrderDetails>
```

10 Сохраните данные ответа.

Данные ответа в этом запуске будут сохранены для использования в будущих шагах.

- a На левой панели Run Result Viewer выберите узел **HTTP-запрос**. Щелкните ссылку **Тело ответа** на панели «Записанные данные». Откроется окно браузера с XML-ответом.
- b Сохраните все содержимое окна в файле Response.xml. Закройте Run Results Viewer.

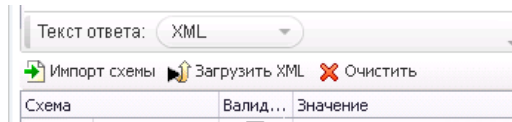
Как проверить полученные данные?

Чтобы убедиться, что получены правильные выходные данные данного метода REST, будут заданы контрольные точки.

1 Вставка контрольной точки



- a Щелкните фрейм **Запрос HTTP**. Откройте представление **HTTP** на панели «Свойства».
- b Нижняя часть панели содержит выходные свойства или выходную схему. Эти элементы будут использовать как контрольные точки для проверки ответа сервера. На нижней панели выберите **XML** в раскрывающемся списке **Тело**.
- c Нажмите кнопку **Загрузить XML** и загрузите файл **Response.xml**, сохраненный ранее. Установите флажок **Проверка** в строках **OrderNumber** и **TotalPrice**.
- d Задайте значение **OrderNumber** большим (>) 10, а значение **TotalPrice** меньшим (<) 255.



2 Запустите тест.

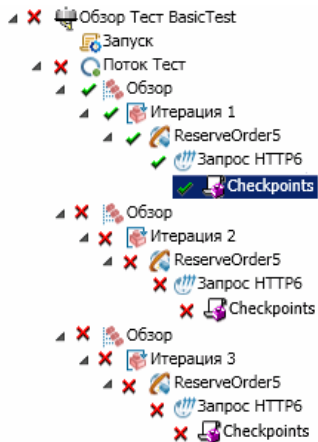


Выберите **Выполнить > Выполнить**, чтобы запустить тест.

3 Убедитесь, что контрольная точка пройдена.



Запустите тест. На левой панели Run Result Viewer нажмите кнопку **Развернуть все**. Обратите, что не все контрольные точки являются пройденными.



4 Определите причину ошибок контрольных точек.

Выберите одну из контрольных точек с ошибками. На правой панели просмотрите панель «Записанные данные» и отметьте «Фактические результаты» и «Ожидаемые значения». В следующем примере **OrderNumber** был допустимым, но **TotalPrice** был недопустимым, поскольку его значение превышало 255.

Имя	Результат	Свойство	Фактический результат	Стиль оценки	Ожидаемые значения	Сведения:
"Checkpoint 2"	✓	"CreatedOrderParams [1]/OrderNumber[1]"	"32"	>	"10"	""
"Checkpoint 3"	✗	"CreatedOrderParams [1]/TotalPrice[1]"	"328"	<	"255"	""

5 Сохраните тест.

Закройте Run Results Viewer и сохраните тест.

Как разрешить изменения в используемой службе REST?

Первоначально был создан прототип метода службы REST **ReserveOrder** с определенными свойствами, такими как URL-адрес и имена свойств. При изменении этих свойств после создания теста, шаг теста более не будет соответствовать прототипу. Мастер разрешения конфликтов обнаруживает изменения свойств метода и помогает разрешить их.

- 1 На панели инструментов выберите узел **ReserveOrder** и выберите **Изменить службу** из контекстного меню. Откроется диалоговое окно «Изменить службу REST».



- 2 Выберите метод **ReserveOrder** и откройте представление **Пользовательский ввод/контрольные точки** на панели **Свойства**.



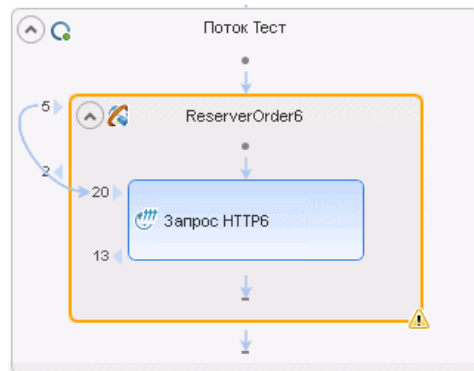
- 3 Выберите свойство **Flight_Number** и нажмите кнопку на панели инструментов **Изменить свойство**. В диалоговом окне «Изменить свойство» переименуйте свойство в **Flight_Number_1**. Нажмите кнопку **OK**.



- 4 Выберите свойство **Number_of_Tickets** и нажмите кнопку на панели инструментов **Удалить свойство**. Подтвердите предупреждение и нажмите кнопку **OK**.

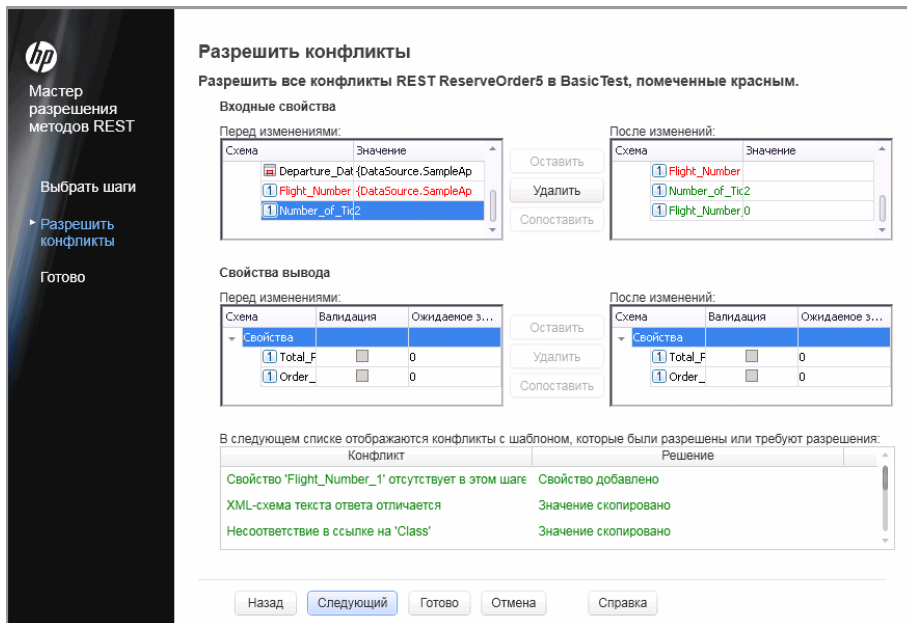
- 5 Нажмите кнопку **OK** в диалоговом окне «Изменить службу REST».

- 6 Просмотрите холст. Обратите внимание на значок уведомления в нижнем правом углу рамки метода REST.



- 7 Щелкните стрелку раскрывающегося списка рядом со значком уведомления и выберите текстовое сообщение: Этот шаг должен быть разрешен. Разрешить шаг. Откроется мастер разрешения методов REST.

- 8 На первом экране мастера отображаются шаги с ошибками. Если было задействовано несколько шагов, то можно выбрать шаги для разрешения и шаги для пропуска. Нажмите кнопку **Далее**.
- 9 На экране разрешения конфликтов выберите свойство **Number_of_Tickets** на правой панели **После изменений**. Нажмите **Сохранить**. При этом в существующем шаге будет сохранено свойство, даже если оно было удалено из прототипа метода.



- 10 На панели **После изменений** выберите старое свойство **Flight_Number** и нажмите **Удалить**. Свойство **Flight_Number** является устаревшим. Вместо него в методе будет содержаться новое автоматически обнаруживаемое свойство **Flight_Number_1**.

С помощью прокрутки перейдите к нижнему разделу экрана мастера, чтобы просмотреть журнал всех конфликтов и их разрешений.

- 11 Нажмите кнопку **Далее**. Нажмите кнопку **Готово**, чтобы закрыть мастер и вернуться к тесту.

Дальнейшие шаги

Теперь, после изучения создания тестов для стандартных действий, веб-служб и служб REST, вы можете создавать собственные тесты для своих приложений без графического интерфейса.

