



HP Unified Functional Testing

软件版本: 12.50
Windows® 操作系统

教程

文档发布日期: 2015 年 7 月
软件发布日期: 2015 年 7 月

法律声明

担保

HP 产品和服务的唯一担保已在此类产品和服务随附的明示担保声明中提出。此处的任何内容均不构成额外担保。HP 不会为此处出现的技术或编辑错误或遗漏承担任何责任。

此处所含信息如有更改，恕不另行通知。

受限权利声明

机密计算机软件。必须拥有 HP 授予的有效许可证，方可拥有、使用或复制本软件。按照 FAR 12.211 和 12.212，并根据供应商的标准商业许可的规定，商业计算机软件、计算机软件文档与商品技术数据授权给美国政府使用。

版权声明

© Copyright 1992 - 2015 Hewlett-Packard Development Company, L.P.

商标声明

Adobe® 和 Acrobat® 是 Adobe Systems Incorporated 的商标。

Google™ 和 Google Maps™ 是 Google Inc 的商标。

Intel® 和 Pentium® 是 Intel Corporation 在美国和其他国家/地区的商标。

Microsoft®、Windows®、Windows® XP 和 Windows Vista® 是 Microsoft Corporation 在美国的注册商标。

Oracle 和 Java 是 Oracle 和/或其附属公司的注册商标。

文档更新

此文档的标题页包含以下标识信息：

- 软件版本号，用于指示软件版本。
- 文档发布日期，该日期将在每次更新文档时更改。
- 软件发布日期，用于指示该版本软件的发布日期。

要检查是否有最新的更新，或者验证是否正在使用最新版本的文档，请访问：

<https://softwaresupport.hp.com>。

需要注册 HP Passport 才能登录此站点。要注册 HP Passport ID，请访问：<https://softwaresupport.hp.com> 并单击“Register”。

支持

请访问 HP 软件联机支持网站：<https://softwaresupport.hp.com>

此网站提供了联系信息，以及有关 HP 软件提供的产品、服务和支持的详细信息。

HP 软件联机支持提供客户自助解决功能。通过该联机支持，可快速高效地访问用于管理业务的各种交互式技术支持工具。作为尊贵的支持客户，您可以通过该支持网站获得下列支持：

- 搜索感兴趣的知识文档
- 提交并跟踪支持案例和改进请求
- 下载软件修补程序
- 管理支持合同
- 查找 HP 支持联系人

- 查看有关可用服务的信息
- 参与其他软件客户的讨论
- 研究和注册软件培训

大多数提供支持的区域都要求您注册为 HP Passport 用户再登录，很多区域还要求用户提供支持合同。要注册 HP Passport ID，请访问 <https://softwaresupport.hp.com>，然后单击“注册”。

要查找有关访问级别的详细信息，请访问：<https://softwaresupport.hp.com/web/softwaresupport/access-levels>。

HP Software Solutions & Integrations 和最佳实践

请访问 <https://h20230.www2.hp.com/sc/solutions/index.jsp> 上的 **HP Software Solutions Now** 以了解 HP 软件目录中的产品如何一起使用、交换信息和解决业务需求。

请访问 <https://hpln.hp.com/group/best-practices-hpsw> 上的 **Cross Portfolio Best Practices Library** 以访问各种最佳实践文档和材料。

目录

欢迎使用 UFT 教程	7
UFT 指南和参考	7
其他联机资源	9
第 1 部分: Unified Functional Testing 简介	10
自动测试的优点	11
UFT 的测试过程	11
UFT 主窗口	13
第 2 部分: 分析应用程序和创建测试	17
了解应用程序	18
浏览航班预订应用程序	18
为所有测试创建解决方案	20
第 3 部分: 创建和运行自动 GUI 测试	23
课程 1: 创建 GUI 测试和操作	24
课程 2: 创建对象存储库	27
UFT 测试对象识别 - 概述	28
练习 2a: 从应用程序添加对象	29
练习 2b: 使用导航并识别进程创建对象存储库	34
课程 3: 向测试添加步骤	38
在 GUI 测试中添加测试步骤 - 概述	39
练习 3a: 在关键字视图中向 Login 操作添加步骤	39
在关键字视图和编辑器中分析 Login 操作	43
练习 3b: 通过录制将步骤添加到 FlightFinder 操作	45
练习 3c: 使用工具箱窗格将步骤添加到 Select Flight 操作	48
练习 3d: 使用步骤生成器将步骤添加到 Book Flight 操作	51
高级练习 3e (可选) - 使用编辑器添加步骤	56
课程 4: 运行和分析 GUI 测试	61
练习 4a: 运行测试	62
练习 4b: 导航运行结果	64
练习 4c: 分析运行结果	65
课程 5: 参数化步骤和对象	66
参数化测试、操作和对象 - 概述	67
练习 5a: 创建参数化测试	67
练习 5b: 定义数据表参数	68
练习 5c: 将参数值添加到数据表	71
练习 5d: 运行参数化测试	73
课程 6: 创建检查点和输出值	78
了解检查点和输出值类型	79
练习 6a: 创建检查点测试	80

练习 6b:检查对象值	81
练习 6c:检查表值	84
练习 6d:检查文本值	89
练习 6e:在对象存储库中管理检查点	95
练习 6f:使用检查点运行和分析测试	96
练习 6g: 创建输出值测试	99
练习 6h:添加输出值步骤	100
课程 7:创建函数和函数库	108
函数和函数库 - 概述	109
练习 7a:创建函数	109
练习 7b:将函数库与测试关联	111
练习 7c:使用函数执行检查	112
课程 8:在测试中使用洞察	117
洞察对象标识 - 概述	118
练习 8a:为洞察对象创建测试	118
练习 8b:将洞察对象添加到对象存储库	119
练习 8c:在测试中使用洞察对象	122
 第 4 部分: 创建和运行自动 API 测试	124
课程 1:创建 API 测试	125
课程 2:创建简单的 API 测试步骤	126
课程 3:使用标准活动创建 API 测试步骤	130
UFT API 测试标准活动 - 概述	131
练习 3a:使用标准活动创建测试	131
课程 4:参数化 API 测试步骤	135
参数化 API 测试步骤 - 概述	136
练习 4a:从数据源参数化测试步骤	136
练习 4b:从以前步骤的输出参数化测试步骤	145
练习 4c:使用自定义表达式参数化具有多个源的测试	148
课程 5:运行 API 测试	154
练习 5a:运行测试	155
练习 5b:导航运行结果	156
练习 5c: 分析运行结果	157
课程 6:创建并运行 Web Service 的 API 测试	158
练习 6a: 创建 Web Service 测试	159
练习 6b: 导入 Web Service	159
练习 6c:构建并参数化 Web Service 测试	161
练习 6d:运行 Web Service 测试	167
课程 7:创建并运行 REST 服务的 API 测试	170
练习 7a:创建 REST 服务测试	171
练习 7b:创建 REST 服务结构	171
练习 7c: 使用 REST 服务方法创建测试	175
练习 7d:运行 REST 服务测试	178
练习 7e:解决 REST 服务冲突	179
课程 8:创建并运行 Web 应用程序服务 (WADL) 的 API 测试	183
练习 8a:为 Web 应用程序服务创建测试	184

练习 8b:导入 Web 应用程序服务模型	184
练习 8c:编辑 Web 应用程序服务方法	186
练习 8d:构建带有 Web 应用程序服务方法的测试	189
练习 8e:运行 Web 应用程序服务测试	193
第 5 部分: 在单个测试中创建和运行 GUI 和 API 测试	195
课程 1:创建测试以同时运行 GUI 和 API 测试	196
课程 2:从 GUI 测试中调用 API 测试	197
课程 3:运行一个调用 API 测试的 GUI 测试	201
接下来要执行的操作	203
向我们发送反馈	205

欢迎使用 UFT 教程

《UFT 教程》是自定进度的指南，它为您提供使用 UFT 测试应用程序的基本知识。本教程将使您了解创建和运行自动 GUI 和 API 测试以及分析运行结果的过程。

完成本教程之后，您可以将学到的技能应用到自己的应用程序测试中。

备注: 要了解更多有关创建和运行 Web 应用程序的 GUI 测试的信息，请参阅《Web 应用程序的 UFT GUI 测试教程》，该文档可从 <UFT 安装文件夹>\help 目录中获取。

教程受众和范围

本教程针对 UFT 的新用户。无需对 UFT、QuickTest 或 Service Test 有所了解。对测试概念和功能测试过程的一般了解可能会有帮助，但不是必需的。UFT 可用于创建 GUI 测试、API 测试、业务流程测试和包含 GUI 与 API 测试或测试调用的复合测试。

备注: 此教程中引用的文件系统路径与 Windows 7 操作系统相关。其他操作系统中的路径可能稍有不同。

将 UFT 用于 BPT

除了测试之外，如果您连接到支持 BPT 的 ALM 服务器，UFT 还允许您创建用于业务流程测试的关键字驱动的业务组件、脚本化业务组件或 API 业务组件。本教程中描述的步骤专用于创建 GUI 和 API 测试，但您也可以将大多数步骤应用于创建关键字驱动的组件、脚本化组件或 API 组件。有关业务组件和 BPT 的更多详细信息，请参见《HP Unified Functional Testing 用户指南》和《HP Business Process Testing 用户指南》。

备注: 除非另有指定，否则在此指南中对 **Application Lifecycle Management** 或 **ALM** 的引用将应用于 ALM 和 Quality Center 的所有当前支持版本。注意，您正使用的 ALM 或 Quality Center 的特定版本可能不支持某些功能和选项。

有关支持的 ALM 或 Quality Center 版本的列表，请参阅《HP Unified Functional Testing 产品可用性矩阵》，您可在 [HP 支持矩阵](#) 页面中找到此文档 (需要 HP Passport)。

有关 ALM 或 Quality Center 版本的详细信息，请参见《HP Application Lifecycle Management 用户指南》或《HP Quality Center User Guide》。

UFT 指南和参考

下表提供了 UFT 指南、联机帮助和参考的列表：

注: 要检查以下任一指南的最近更新，请访问 HP 软件产品手册网站 (<https://softwaresupport.hp.com/group/softwaresupport/search-result?keyword=>)。

入门

参考	描述

参考	描述
新增功能	描述 Unified Functional Testing 最新版本中的最新功能。 您还可以从 Unified Functional Testing 帮助 菜单访问 新增功能 。
产品视频	单击链接或选 帮助 > 产品功能视频 查看演示主要产品功能的短视频。
自述文件	提供有关 Unified Functional Testing 的最新新闻和信息。 有关最新的自述文件, 请访问 HP 软件产品手册网站 (需要 HP Passport), 网址为 https://softwaresupport.hp.com/group/softwaresupport/search-result?keyword= 。
UFT PAM	产品可用性矩阵 (PAM) 提供有关此版本的 UFT 支持的技术和集成的当前信息。
UFT 教程	UFT 教程是自定进度的可打印指南, 用于指导您完成创建复合应用程序的 GUI、API 和完整 UFT 测试的过程。
Web 应用程序的 GUI 测试教程	Web 应用程序的 GUI 测试教程是自定进度的可打印指南, 用于指导您完成创建 Web 应用程序的 GUI 测试的过程并熟悉测试环境。

PDF 指南

指南	描述
UFT 用户指南	《HP Unified Functional Testing 用户指南》描述了如何使用 UFT 测试应用程序。它逐步详细地说明了如何创建、调试和运行测试以及如何报告测试过程中检测到的缺陷。
Run Results Viewer	《HP Run Results Viewer 用户指南》介绍如何使用 Run Results Viewer 解释 和使用 GUI 或 API 测试的结果。
UFT 安装指南	《HP Unified Functional Testing 安装指南》提供有关如何在独立计算机上安装和设置 UFT 的完整逐步说明。
UFT 快速入门	UFT 安装快速入门指南介绍了执行 UFT 基本安装的步骤。
UFT 插件指南	《HP Unified Functional Testing 插件指南》介绍如何设置 UFT 插件支持和标准 Windows 测试支持。插件支持您使用 GUI 测试和业务组件测试任何支持的环境。
新增功能	描述 Unified Functional Testing 最新版本中的最新功能。
安全参考	包含使用 UFT 的安全相关信息以及提供更多信息的参考。
Runtime Engine Guide	包含有关如何使用运行时引擎安装 (不带 UFT IDE) 运行测试的信息。

参考

UFT 联机帮助主页上提供了参考的链接。

参考	描述
对象模型参考	GUI 测试的对象模型参考包括每个 UFT 测试对象的描述、方法和属性列表、语法、示例和标识属性。
VBScript 参考	描述编写 VBScript 脚本时可能会使用的对象、方法、属性、函数和其他元素的 Microsoft Visual Basic Scripting 文档。
自动化对象模型参考	列出了允许您从其他应用程序内控制 UFT 的对象、方法和属性。
对象存储库自动化参	描述允许您从 UFT 外部操作 UFT 共享对象存储库及其内容的对象。

参考	描述
考	
运行结果架构参考	提供有关运行结果 XML 架构结构的详细信息，描述此 XML 报告中使用的元素和属性。
测试对象架构参考	描述创建测试对象配置 XML 内容（供创建 UFT 扩展性项目时使用）可用的元素和属性的参考。
对象存储库架构参考	描述针对对象存储库架构定义的元素和复杂类型。

有关其他联机参考，请参见 PDF 指南的“欢迎使用”部分。

其他联机资源

以下其他联机资源可用：

资源	描述
HP 软件联机支持	HP 软件支持网站 (www.hp.com/go/hpsoftwaresupport)。要访问，请选择帮助 > HP 软件支持 。 备注： 大多数提供支持的区域都要求您注册为 HP Passport 用户再登录，很多区域还要求用户提供支持合同。要注册 HP Passport 用户 ID，请访问： http://h20229.www2.hp.com/passport-registration.html
测试论坛	<ul style="list-style-type: none">GUI 测试: http://h30499.www3.hp.com/t5/Unified-Functional-Testing/bd-p/sws-Fun_TEST_SFAPI 测试: http://h30499.www3.hp.com/t5/Service-Test-Support-and-News/bd-p/sws-Serv_TEST_SFBPT: http://h30499.www3.hp.com/t5/Business-Process-Validation/bd-p/sws-BPT_SF
UFT 产品页	HP Unified Functional Testing 产品页 (http://www8.hp.com/us/en/software-solutions/unified-functional-testing-automated-testing/index.html)，提供有关 UFT 的信息和相关链接。要访问，请选择帮助 > 有用的链接 > 产品页 。
疑难解答 & 知识库	HP 软件支持网站上的疑难解答页面 (http://h20230.www2.hp.com/troubleshooting.jsp)，您可在此搜索 HP 软件自助解决知识库。要访问，请选择帮助 > 知识库 或帮助 > 疑难解答 。
HP 软件社区	HP IT 专家社区站点 (http://h10124.www1.hp.com/campaigns/IT_Experts/pages/home.html)，可在其中与其他 HP 软件用户交流，阅读有关 HP 软件的文章和博客，并下载其他软件产品。
HP 手册站点	HP 软件支持手册网站 (http://support.openview.hp.com/selfsolve/manuals)，可在其中搜索选定 HP 软件产品的最新文档。要访问，请选择帮助 > 有用的链接 > HP 手册站点 。
新增功能	UFT 新增功能帮助，该帮助介绍此版本 UFT 中的新功能和增强。
产品视频	UFT HPLN (HP Live Networks) 页面 (https://hpln.hp.com/page/uft-120-videos)，其中提供了所有产品视频的列表。
HP 软件网站	HP 软件网站 (www.hp.com/go/software)。此站点提供了有关 HP 软件产品的最新信息。包括新的软件发布、研讨会及展览会、客户支持和其他更多信息。

第 1 部分: Unified Functional Testing 简介

在本课程中，您将概括性地了解自动测试和 UFT。您还将开始准备测试。

备注: 建议使用此教程的软副本，因为存在要求您将测试步骤复制并粘贴到 UFT 中的部分。但是，请记住在此教程的选定部分，UFT 将录制您的操作或识别您使用鼠标指向的对象。在该会话期间，您应该避免将焦点切换到 PDF 或帮助窗口。

此部分包含以下内容:

· 自动测试的优点	11
· UFT 的测试过程	11
· UFT 主窗口	13

自动测试的优点

如果曾经手动测试过应用程序或网站，您应该知道手动测试这些应用程序的缺点。

手动测试耗时且冗长，需要大量人力资源投入。最糟糕的是，时间限制通常导致无法在应用程序发布之前彻底手动测试每个功能。这会让您怀疑是否有严重的缺陷未被检测出来。

UFT 对应用程序的 GUI 和非 GUI (服务) 层进行自动测试可以加快测试过程，从而解决手动测试存在的各种问题。您可以创建检查应用程序或网站所有方面的测试，然后在每次应用程序或网站发生更改时运行这些测试。因此，您只需一次性创建测试，即可在随后每次应用程序发生更改时运行这些测试，而无需在每次应用程序更新时更新测试。

当 UFT 运行这些测试时，它会模拟人类用户操作：在应用程序或网站上移动光标、单击应用程序 GUI 中的对象、输入关键输入或运行应用程序的 API 流程。但是，与手动测试不同，UFT 的操作速度比任何人类用户都要快。

自动测试的优点有很多：

自动测试的优点	
快速	自动测试的速度比人类用户执行的手动测试快得多。
可靠	自动测试将在每次运行时执行完全相同的操作，从而消除人为错误。
可重复	可以测试应用程序或网站在重复执行相同操作之后如何反应。
可编程	可以对能够测试各种复杂但重要的场景的复杂测试进行编程，从而找到常规手动测试不易发现的问题或缺陷。
全面	可以构建涵盖应用程序或网站中所有功能的一组测试。
可重用	即使用户界面或内部应用程序 API 发生更改，也可以在不同版本的应用程序或网站上重用自动测试。

UFT 的测试过程

创建 UFT 测试时，需要执行许多步骤：

1. 分析应用程序

计划测试的第一步是分析应用程序以确定测试需求：

应用程序开发环境是什么？	要使用应用程序的用户界面对象，您将需要加载相应的 UFT GUI 测试插件。例如，应用程序可能在 Web、.NET 或 Java 环境中构建。因此，您需要在 UFT 使用 Web、.NET 或 Java 插件才能使用这些环境。
想要测试哪些业务流程和功能？	为此，请思考下用户将在应用程序中执行的活动以及应用程序为执行这些业务流程需要执行的内部操作。 创建 GUI 测试步骤以模拟用户界面中用户的操作。创建 API 测试步骤以执行应用程序在后台运行的流程。
您的应用程序是否使用标准应用程序活动	根据您的应用程序 API 所运行的功能，使用 UFT API 测试附带的预置活动或导入/创建自定义活动。

或自定义设计服务?	
如何将测试分为可测试的小单元?	您应该将要测试的流程和功能分为较小的任务，以便在 GUI 测试中创建 UFT 操作。这些较小和更多的模块化操作使您的测试更易读取和遵循，并且可以帮助长期维护更轻松。

甚至在此阶段，您可以开始创建测试框架并向 GUI 测试添加操作。

2. 准备测试基础结构

基于您的测试需求，您必须确定所需的资源并相应地创建这些资源。

- 对于 GUI 测试，这些资源包括“共享对象存储库”（包含代表应用程序中对象的测试对象）、“函数库”（包含要在测试中使用的自定义函数）、“恢复场景”（指示 UFT 在应用程序出现问题时如何响应），“环境变量文件”（包含常见环境变量的定义）或“外部数据表”（用于参数化测试步骤）。
- 对于 API 测试，这些资源包括“WSDL”或“WADL”文件（描述应用程序服务的方法），“REST 服务”（创建以用作原型或应用程序的 REST 流程），“外部数据源”、“虚拟化项目”（与服务调用结合使用），“.NET 程序集”（由测试步骤引用）或“Java 类”（在测试步骤中使用）。这些资源必须在 UFT 中导入或创建。

还需要配置 UFT 设置，以便 UFT 将执行您可能需要的任何其他任务，例如每次运行测试时均显示结果报告、启用或禁用测试运行的调试，以及类似任务。

3. 构建测试并将步骤添加到每个测试

测试基础结构和资源准备就绪后，可以开始构建测试：

- 对于 GUI 测试，可以创建一个或多个空测试，然后向其添加操作以创建测试框架。然后将对象存储库与相关操作关联，以便可以通过关键字驱动方法或创建脚本来插入步骤。
- 对于 API 测试，您可以创建一个或多个空测试、通过将活动拖到测试画布中将测试步骤添加到这些测试并且定义这些步骤的输入、输出和检查点属性。还可以对在一个操作（可作为单独的测试步骤运行）中运行多次的步骤进行分组。
- 也可以将所有测试添加到单个解决方案。这样的解决方案使您能够同时存储、管理和编辑任何相关测试，而无需在打开一个测试前关闭其他测试。
- 您可能还要配置测试首选项和设置（对于 GUI 测试）或特定于测试的属性（对于 API 测试）。

4. 增强测试

可以通过多种方式增强测试的功能：

对于 GUI 测试...	<ul style="list-style-type: none"> 插入“检查点”作为测试步骤，以检查应用程序是否正常运行。例如，这些检查点可以检查页面的特定值、单个测试对象或文本字符串。 通过使用多组数据参数化测试步骤值，可以检查应用程序如何使用不同的值来执行相同的操作。可以通过将测试步骤的固定值替换为“参数”实现此目的。 可以使用 VBscript 将“编程和条件语句或循环语句”以及其他编程逻辑添加到测试。
对于 API 测试...	<ul style="list-style-type: none"> 通过选择“检查点属性”并输入步骤属性的预期值，可以验证测试步骤和测试步骤的各个属性。 通过使用多组数据参数化测试步骤属性，可以检查应用程序如何使用不同的值来执行相同的流程。可以通过将固定值替换为“参数”实现此目的。 通过“自定义代码活动”、“事件处理程序”或使用 UFT 活动向导创建的“自定义活动”向您的测试步骤添加功能。

5. 调试、运行和分析测试

您可以使用 UFT 的调试功能对测试进行调试，以确保它顺利运行，不会中断。测试正常运行后，运行该测试以检查应用程序的行为。运行时，UFT 将在应用程序的用户界面上执行每个步骤（运行 GUI 测试时）或运行应用程序的 API 流程（运行 API 测试时）。

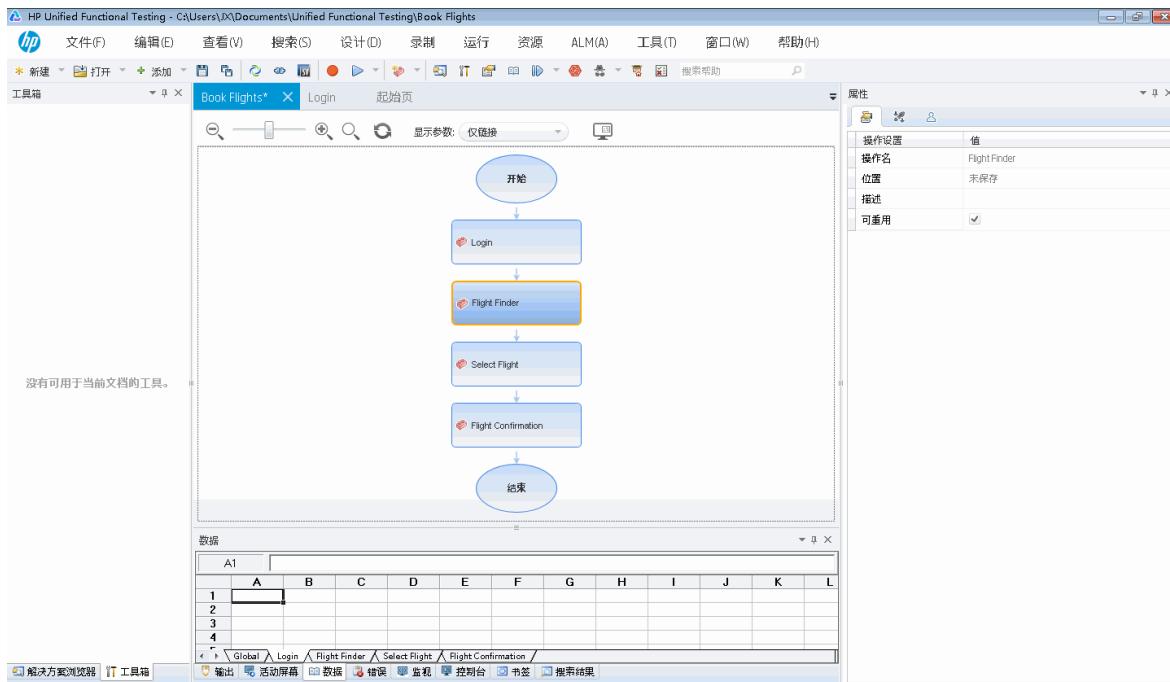
6. 报告缺陷

如果具有访问 ALM 服务器的权限，则可以向 ALM 项目报告您发现的缺陷。如果无此权限，可以手动将缺陷报告到您自己的缺陷数据库。

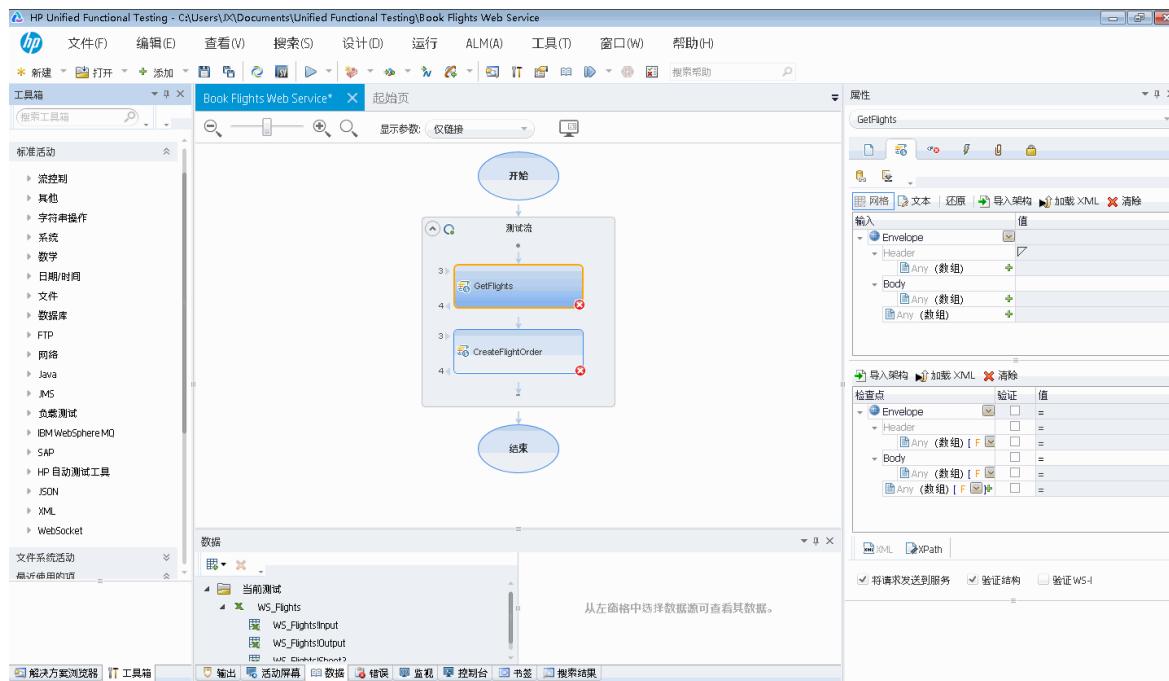
UFT 主窗口

开始使用 UFT 和创建测试之前，您应该先熟悉 UFT 主窗口。

以下图像显示了创建 GUI 测试后的 UFT 窗口，其中测试流显示在画布中，且会显示工具栏、解决方案浏览器、“数据”窗格和“属性”窗格。



以下图像显示了创建 API 测试后的 UFT 窗口，其中测试流显示在画布中，且会显示工具栏、“工具箱”窗格、“数据”窗格和“属性”窗格。



主窗口显示多个元素:

测试文档

UFT 在文档窗格 (在 UFT 窗口中央) 中显示打开的文档。可以使用位于工具栏下面的文档选项卡导航到打开的文档，并使它们处于活动状态。

文档窗格可显示以下类型的文件:

测试	GUI 和 API 测试显示在画布中对应的选项卡上。GUI 测试显示测试流，以及测试中包含的每个操作的单独项。 API 测试显示测试中包含的不同步骤的测试流。 BPT 测试以网格形式显示，从而列出测试中包含的各个组件、组或流。
GUI 测试操作	每个 GUI 测试包含单独的操作或对其他操作的调用。您可以在以下视图之一中查看每个操作： <ul style="list-style-type: none">关键字视图: 在基于图标的模块化表中显示每个步骤 (和对象层次结构)。编辑器: 以 VBScript 行的形式显示每个步骤，将其显示在文本/代码编辑器中。在基于对象的步骤中，VBScript 行定义对象层次结构。
业务组件	业务组件可用于为应用程序中的每个业务流程设计单个模块化测试“单元”。然后，可以将这些组件添加到业务流程测试中，作为应用程序场景一起运行。 可以显示以下类型的业务组件之一： <ul style="list-style-type: none">关键字 GUI 组件: 这些组件仅显示在“关键字视图”中。脚本化 GUI 组件: 这些组件可以显示在关键字视图或编辑器中。(默认情况下，它们在编辑器中打开)API 组件: 这些组件与测试画布一起显示 (与 API 测试的方式相同)。
函数库	函数库可用于创建、编辑和修改要在测试中使用的函数。这些函数可以写入单个函数库，然后通过将该函数库与每个测试关联，可以将其用于多个测试。 函数库显示在编辑器中。

应用程序区域	每个 GUI 业务组件还包含应用程序区域。应用程序区域充当组件的对象存储库、函数库和配置设置的容器。每个应用程序区域还可以与多个业务组件关联。 应用程序区域用户界面将显示为一系列侧栏选项卡。
用户代码文件	在 API 测试中，可以添加特殊事件处理程序代码或自定义代码文件。此代码可用于补充和扩展 API 测试步骤的预置功能。事件处理程序代码包含在 TestUserCode.cs 文件（已包含在默认 API 测试中）中。 这些文件显示在编辑器中。
起始页	此页面欢迎您使用 UFT 并提供最近文件、新功能描述、产品论坛的链接和其他支持链接。可使用快捷方式按钮创建文档或打开现有文档。
内部浏览器页面	还可以查看论坛的 Internet 页面及其他与产品相关的资料，例如从起始页或“帮助”菜单进行访问的资料。

工具栏和菜单

除了文档窗格以外，UFT 窗口还包含以下元素：

- **标题栏。** 显示当前测试或解决方案的路径。
- **菜单栏。**
- **UFT 工具栏。**

窗格

UFT 窗口包含多个用于协助创建和设计测试文档的窗格。

后续课程将详细描述以下某些窗格和工具栏选项。有关其他窗格和工具栏选项的详细信息，请参阅《HP Unified Functional Testing 用户指南》。

名称	工具栏按钮	描述	默认位置
解决方案浏览器		显示当前已打开或包含在解决方案中的所有测试、组件和应用程序区域，以及与当前测试和组件关联的资源。使用解决方案浏览器，可以管理这些资源。	UFT 窗口左侧的选项卡。
工具箱		<ul style="list-style-type: none">对于 GUI 测试和组件：显示所有可供测试使用的关键字，并使您能够将对象从“工具箱”窗格拖放到测试，或将函数从“工具箱”窗格调用到测试。对于 API 测试和组件：显示可在测试中使用的所有活动，并且可用于将这些活动拖放到画布上。	UFT 窗口左侧的选项卡。
文档窗格	N/A	显示所有打开的文档。每个文档都有一个在单击后可使其处于活动状态的选项卡。	位于 UFT 窗口中央的无标签窗格。每个文档选项卡都标有文档名称。 要显示：打开测试文档。
属性		对于 GUI 测试和组件：显示当前选定的测试、操作、组件或应用程序区域的所有属性 对于 API 测试和组件：显示选定测试步骤/测试流或选定数据源（在“数据”窗格中）的所有属性。	UFT 窗口右侧的窗格。 要显示： <ul style="list-style-type: none">选择“查看”>“属性”在工具栏中单击属性按钮。双击画布中的 API 测试

			<p>步骤。</p> <ul style="list-style-type: none"> 右键单击画布中的 API 测试步骤并选择“属性”。
数据		帮助您参数化测试。	UFT 窗口底部的选项卡。
输出	N/A	显示运行会话期间的信息。	UFT 窗口底部的选项卡。 要显示它, 请选择 查看 > 输出 。
错误	N/A	显示测试或组件存在的问题的列表: 缺少来自测试的引用 (如缺少对象存储库或 GUI 测试的恢复场景, 或缺少对外部文件的引用或缺少 API 测试的属性值)。	UFT 窗口底部的选项卡。 要显示它, 请选择 查看 > 错误 。
活动屏幕	N/A	提供在录制会话期间执行某个步骤时显示的应用程序的快照。 此窗格不用于 API 测试。	UFT 窗口底部的选项卡。 要显示它, 请选择“ 查看 > “ 活动屏幕 ”。
调试窗格		<p>帮助您调试测试。</p> <p>有多个调试窗格:</p> <ul style="list-style-type: none"> 断点 调用堆栈 局部变量 控制台 监视 线程 (仅限 API 测试) 加载的模块 (仅限 API 测试) 	UFT 窗口底部的选项卡。
任务	N/A	显示为当前测试定义的任务并可用于管理这些任务。此窗格还显示测试的操作、函数库或用户代码文件的 TODO 注释步骤。	UFT 窗口底部的选项卡。 要显示它, 请选择 查看 > 任务 。
搜索结果	N/A	显示您在“查找”对话框中或使用其他“搜索”菜单项定义的搜索条件所有出现的地方。	UFT 窗口底部的选项卡。 要显示: <ul style="list-style-type: none"> 选择“查看 > “搜索结果”。 执行搜索。
书签	N/A	显示测试文档中包含的书签的列表和位置。	UFT 窗口底部的选项卡。 要显示它, 请选择 查看 > 书签 。
运行步骤结果	N/A	显示单个 API 测试步骤的测试运行的运行结果。 此窗格不用于 GUI 测试。	UFT 窗口底部的选项卡。 要显示: <ul style="list-style-type: none"> 选择“查看 > “运行结果”。 通过右键单击 API 测试步骤并选择“运行步骤”来运行步骤。

第 2 部分: 分析应用程序和创建测试

[Unified Functional Testing 简介 \(第 10 页\)](#) 概述了自动测试和 UFT。

在本课程中，您将分析应用程序以确定需要测试的项。

此部分包含以下内容：

· 了解应用程序	18
· 浏览航班预订应用程序	18
· 为所有测试创建解决方案	20

了解应用程序

在开始创建应用程序测试之前，您需要确定应用程序中要测试的确切内容。要这样做，您需要根据其应用程序进程（应用程序为完成特定任务而执行的不同活动）对应用程序进行分析。

在此教程中，您将测试一个航班预订应用程序。此应用程序模拟航班信息和预订服务。

此应用程序包含两个单独的组件：

- **预订航班层。**此层是应用程序的用户界面，提供查找和预订航班的界面。
- **航班 API 层。**此应用程序提供航班预订应用程序的服务 (API) 端。有关服务的方法和操作的详细信息，请在航班 API 窗口中单击“Open Help Page”按钮。

使用预订航班层，您将创建应用程序的 GUI 测试。使用航班 API 层，您将创建应用程序的 API 测试。

在您计划测试航班预订应用程序时，请考虑以下事项：

需要测试哪些业务流程？	这是一个基本问题。请考虑以下问题： <ul style="list-style-type: none">• 您的应用程序应执行什么流程？• 基于决定的流程，用户执行什么操作才能实现这些流程？
该应用程序是如何组织的？	<ul style="list-style-type: none">• 针对每个用户活动，应用程序是否有独立的部分/页面/模块？• 用户可以执行哪些活动？• 应用程序中的这些部分/页面/模块在哪里？• 这些用户活动的预期结果是什么？• 哪些场景后流程支持这些用户活动？
在应用程序的每个部分/页面/模块中，哪些用户界面元素需要测试？	<ul style="list-style-type: none">• 在每个区域中，哪些用户界面对象需要测试？• 测试需要模拟哪些用户操作？
哪些活动可能会在多个场景中使用？	哪些特定用户操作或应用程序流程要重复执行？例如，登录应用程序可能是用户重复执行的操作，连接到用户凭据数据库是应用程序重复执行的操作。
为了测试目的，需要支持哪些开发环境？	UFT 提供各种插件以支持大量测试环境。为了加载适合您应用程序的插件，您必须考虑应用程序开发过程中使用的技术，以确保 UFT 支持您的技术。此外，还必须在打开 UFT 和创建测试时加载这些插件。这可确保创建和运行测试时 UFT 能正确识别应用程序中的对象。

浏览航班预订应用程序

在开始创建测试之前，您需要浏览示例应用程序并查看该应用程序的用户操作以及哪些流程支持该应用程序流。

浏览和使用应用程序时，考虑[上一课](#)中提出的问题。使用这些问题指导您思考如何从应用程序创建一个或多个测试。

1. 打开航班预订应用程序。

同时打开预订航班 (GUI) 层和航班 API (服务) 层:

- 可从“开始”>“所有程序”>“**HP Software**”>“**HP Unified Functional Testing**”>“Sample Applications”>“Flight **GUI**”访问预订航班层。
- 可从“开始”>“所有程序”>“**HP Software**”>“**HP Unified Functional Testing**”>“Sample Applications”>“Flight **API**”访问航班 API 层。

有关在 Windows 8 和 Windows Server 2012 中访问 UFT 和 UFT 工具和文件的详细信息, 请参阅[在 Windows 8.X 或更高版本操作系统中访问 UFT \(第 204 页\)](#)。

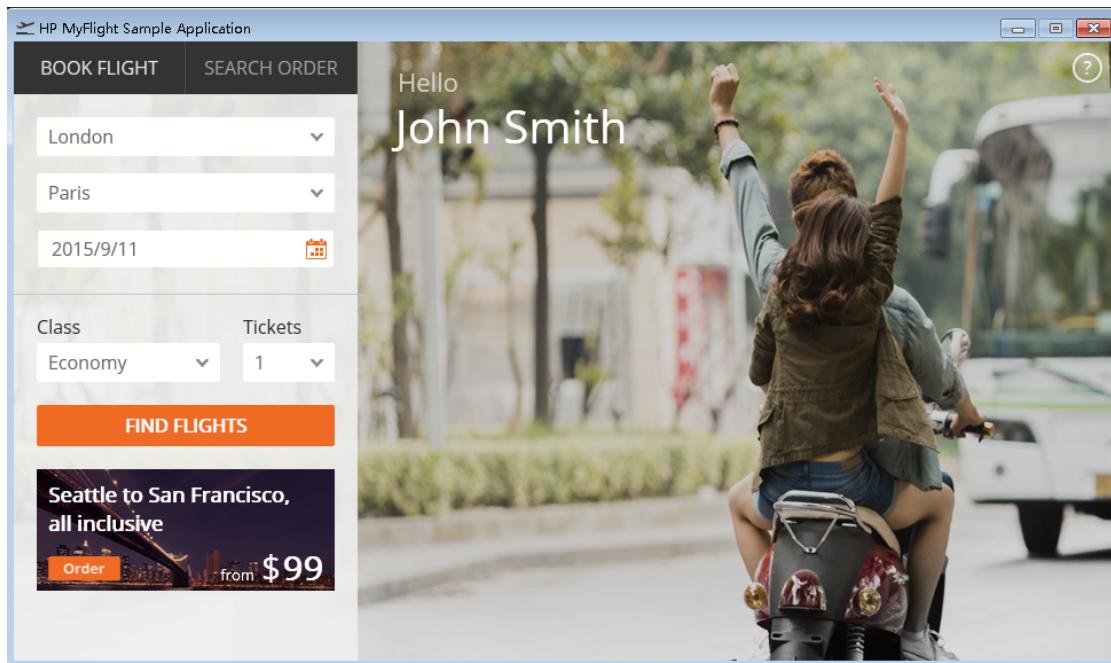
2. 登录 Book Flights 应用程序。

在 Book Flights 应用程序的起始页中, 输入 John 作为用户名, 输入 hp 作为密码。

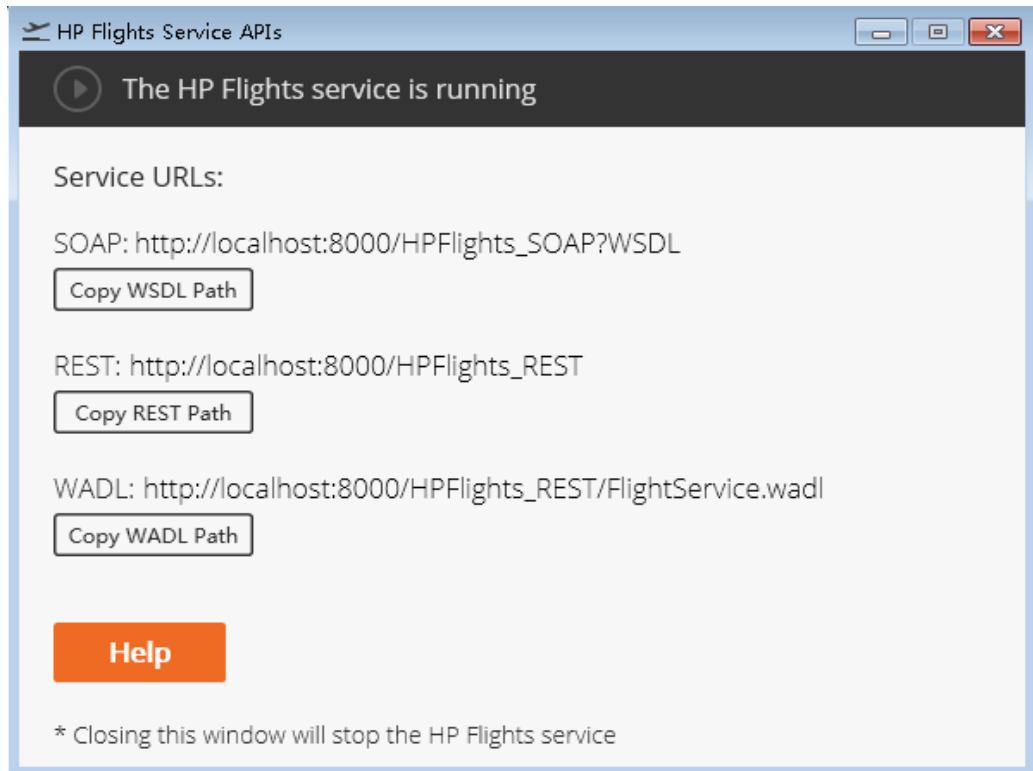
备注: 航班 API 层不需要登录信息。

成功登录后, 应用程序层显示应如下所示:

预订航班 (GUI 层)



航班 API 层



3. 浏览应用程序层。

对于预订航班层...	在每个页面上输入请求的信息或选项，完成预订流程。 浏览应用程序时，考虑要测试的用户操作以及设置测试需要创建的对象。
对于航班 API 层...	单击“Help”按钮可查看该应用程序中所包含方法的描述。 在浏览应用程序中包含的方法列表中，注意为每种方法提供的属性详细信息。稍后设计测试，为这些方法提供属性值时可以使用此数据。

4. 退出应用程序浏览会话。

- 对于预订航班层：“Order Details”窗口中显示航班预订订单详细信息后，请单击“New Search”使应用程序返回到开始处。
- 对于航班 API 层：最小化航班 API 窗口。请勿关闭此窗口，因为此操作将停止服务。

现在即可使用这些应用程序创建应用程序每一层的测试。继续[为所有测试创建解决方案 \(第 20 页\)](#)，创建学习此教程过程中将使用的测试的解决方案。

为所有测试创建解决方案

在 UFT 中，您可以创建一个“解决方案”用作测试的容器。创建解决方案后，您可以向此解决方案添加任何测试、业务组件或应用程序区域。

使用解决方案，您可以方便地将测试分组到一起。例如，您可以创建包含特定应用程序所有测试的解决方案，或创建仅包含应用程序 GUI 或 API 测试部分的解决方案。

您可以将所有类型的 UFT 文档添加到解决方案，可以将测试添加到多个解决方案。

在本练习中，您将新建一个解决方案，用于保存将在本教程学习期间创建的测试。

1. 启动 UFT。

执行以下某项操作：

如果 UFT 当前未打开	<ol style="list-style-type: none">打开 UFT。<ul style="list-style-type: none">◦ 在桌面或“开始”屏幕上（在 Windows 8.x 或 Windows Server 2012 中）双击 UFT 图标。◦ 在 Windows 7 或 Windows Server 2008 R2 中，选择“开始”>“所有程序”>“HP Software”>“HP Unified Functional Testing”。在插件管理器中，确认已选中“only”和“WPF”插件。清除所有其他插件。单击“确定”关闭插件管理器并打开 UFT。 <p>UFT 加载所选插件时，将显示 UFT 初始屏幕。</p>
如果 UFT 当前已打开	<ol style="list-style-type: none">请选择“帮助”>“关于 HP Unified Functional Testing”以检查是否加载插件。已加载插件及其名称旁边的复选框将显示在“关于”对话框中。如果未加载 WPF 插件，则必须退出并重新启动 UFT。再次打开插件管理器时，请选择 WPF。 <p>如果启动 UFT 时插件管理器未打开，请在 UFT 打开后，选择“工具”>“选项”。然后，在“启动选项”窗格（“工具”>“选项”>“常规”选项卡>“启动选项”节点）中，选择“启动时显示插件管理器”选项。</p>

2. 浏览起始页。

在 UFT 中，起始页提供最近文件、新功能信息的链接，以及有帮助的支持和社区论坛的链接。在页面的右上角，您可以定义用于在使用 UFT 时显示和关闭起始页的选项。

备注: 如果起始页未显示，您可以通过选择“查看”>“起始页”来显示。

3. 新建解决方案。

- 在工具栏上，单击“新建”向下箭头  并选择“新建解决方案”。
- 输入解决方案的详细信息：
 - **名称:** Flight Reservation Application
 - **位置:** 默认情况下，所有解决方案和测试均保存在 C:\%HOMEPATH%\My Documents\Unified Functional Testing 中。在此教程中，您无需修改此路径。
- 单击创建。

在解决方案浏览器窗格中，现在您可以看到此解决方案的名称显示在窗格顶部。您添加到此解决方案的测试将会作为此解决方案的子节点显示出来。

备注: 如果解决方案浏览器隐藏，则单击“解决方案浏览器”按钮  或选择“查看”>“解决方案浏览器”以使其显示。

现在，您可以开始创建 GUI 测试（如[创建和运行自动 GUI 测试（第 23 页）](#)中所述），或 API 测试（如[创建和运行自动 API 测试（第 124 页）](#)中所述）。

第 3 部分: 创建和运行自动 GUI 测试

完成应用程序的分析和测试目标的计划之后，您将创建应用程序的测试。这项工作的主要部分是创建应用程序用户界面 (GUI) 的测试。这样可以确保应用程序中的控件和对象按设计运行。

创建 GUI 测试涉及多个单独过程:

- 为应用程序中的对象创建包含测试对象的对象存储库
- 创建补充函数以用于测试内部的函数库
- 向测试添加代表用户在应用程序中所执行操作的步骤
- 创建检查点以验证应用程序中的特定对象
- 参数化测试对象值，查看应用程序对不同的输入值如何反应
- 运行测试并分析测试运行结果

下面的课程将详细介绍和教授这些过程。

此部分包含以下内容:

· 课程 1: 创建 GUI 测试和操作	24
· 课程 2: 创建对象存储库	27
· 课程 3: 向测试添加步骤	38
· 课程 4: 运行和分析 GUI 测试	61
· 课程 5: 参数化步骤和对象	66
· 课程 6: 创建检查点和输出值	78
· 课程 7: 创建函数和函数库	108
· 课程 8: 在测试中使用洞察	117

课程 1: 创建 GUI 测试和操作

创建测试应用程序 GUI 的步骤之前，必须先创建测试并创建提供测试结构的操作。

每个 UFT GUI 测试均由对操作的调用组成。操作是将测试划分成多个逻辑部分的单元（在测试内）。通过将测试分成多个操作，可以设计更模块化、易于理解和高效的测试。

您的测试可包含许多不同类型的操作：

内部和外部操作	<ul style="list-style-type: none">“内部操作”是存储在本地测试中的操作（也称为源测试）。“外部操作”是对存储在其他测试中的操作的参考调用。 <p>备注: 由测试调用的外部操作在解决方案浏览器中的测试节点下显示为独立节点。</p> <p>例如，如果您要在多个测试中使用某个操作，您将该操作存储为一个测试中的内部操作，并从其他测试插入对该操作的调用。在其他调用该操作的测试中，该操作可用作外部操作。</p>
可重用操作	<p>当插入对新操作的调用时，它默认为“可重用”，使您能够从任何测试调用该操作。</p> <p>使用可重用操作时，您仅需更新与原始测试一同存储的现有操作。当您修改原始操作时，所有包含对该操作的调用的测试都将更新。这些可重用操作在调用可重用操作的测试中是只读的。</p> <p>在下列情况下，可重用操作非常有用：</p> <ul style="list-style-type: none">您可能需要在多个测试中包括某个过程，例如登录到应用程序。您可能需要在相同测试中多次插入某个过程，例如填充用户在尝试访问应用程序安全部分时打开的用户凭据对话框。 <p>如果要防止将某操作用于其他测试，则可以将该操作设为“不可重用”。</p> <p>提示: 如果您计划在很多测试中使用相同或相似的操作，则可以考虑创建存储库测试，以存储可重用操作。然后，可以将调用从其他测试插入到这些现有操作中。</p>
复制操作	<p>如果要修改操作步骤，还可以在测试中插入操作的副本。复制该操作时，它将成为复制的目标测试的“内部”操作。这些副本未链接到源测试，因此源操作中的任何变更都不会在副本中更新。</p>

通过以下方式将操作添加到测试：

- 调用新操作。** 在测试中插入新的空操作。操作是包含该操作的测试的内部操作。
- 调用操作副本。** 在测试中插入现有操作的副本。
- 调用现有操作。** 将调用插入现有操作中（当前测试的外部操作）。

1. 创建新测试并将其添加到解决方案。

- 在 UFT 工具栏上，单击“添加”按钮向下箭头  并选择“新建测试”。
- 在“将新测试添加到解决方案”对话框中，选择“GUI 测试”。
- 为测试输入以下详细信息：
 - 名称:** Book Flights
 - 位置:** 默认情况下，UFT 将文档保存在 C:\%HOMEPATH%\My Documents\Unified Functional Testing 中。对于本课程，您无需修改此路径。

d. 单击添加。

一个空测试将在画布中打开，带有一个测试流选项卡（名为“Book Flights”）并在文档窗格中打开一个空操作选项卡（名为“Action 1”）。

此测试在解决方案浏览器窗格中也显示为 Flight Reservation Application 解决方案节点的子节点。

2. 重命名 Action 1，以使它拥有逻辑名称。

a. 在画布中，右键单击“Action1”框，然后选择“操作属性”。

b. 在“操作属性”对话框的“常规”选项卡中，将“Action1”重命名为“登录”，并单击“确定”。

c. 在打开的警告中，单击“是”。

现在操作块名称在画布中应显示为“登录”。

3. 新建名为 Flight Finder 的操作。

a. 在画布的任意位置右键单击，然后选择“调用新操作”。

将打开“插入对新操作的调用”对话框：



b. 在“名称”对话框中，输入“Flight Finder”作为操作名称。

c. 确保已选中“可重用操作”和“位于测试结尾”选项。

d. 单击“确定”。

在画布中将添加另一个名为“Flight Finder”的操作块。

4. 向测试添加其他操作。

- a. 在 UFT 工具栏上，单击“插入对新操作的调用”按钮 。
- b. 在“插入对新操作的调用”对话框的“名称”框中，输入“Select Flight”。
- c. 确保已选中“可重用操作”和“位于测试结尾”选项。
- d. 单击“确定”将操作添加到测试。

在画布中将添加另一个名为“Select Flight”的操作块。

5. 创建最终操作。

使用上面使用的方法之一，将另一个名为“Flight Confirmation”的操作添加到测试。

现在，您的测试已包含测试应用程序所需的所有操作。

6. 保存测试。

在 UFT 工具栏上，单击“保存”按钮 。

您可能注意到在单击“保存”之前，在文档窗格的“Book Flights”选项卡上和解决方案浏览器中的“Book Flights”节点上均显示有一个星号(*)。显示这些星号是用来指示文档含有尚未保存的内容。保存测试时，操作选项卡中的所有变更也会保存。

7. 浏览画布中的测试流。

单击画布中的“Book Flights”选项卡可查看整个测试流。

- 您可以使用鼠标或画布上方的缩放栏放大或缩小测试流。
- 当测试流放大到最大且超出画布范围时，您也可以使用画布左下角的微型图浏览测试流。



如果微型图未显示，则单击画布上方的“微型图”按钮  以使其显示。如果测试具有多个操作或嵌套操作，则此微型图将十分有用。

现在，您已具有测试结构，随时可以为测试创建对象存储库。请继续课程 2: 创建对象存储库 (第 27 页)。

课程 2:创建对象存储库

GUI 测试的基础是用于测试应用程序用户界面的测试对象集合。这些测试对象由 UFT 识别，然后存储在与测试关联的对象存储库中。

现在，您已创建测试及其测试结构（通过创建操作），您需要创建用于测试的测试对象。本课程介绍在测试中使用的测试对象、运行时对象以及对象存储库的基本概念。

此课程包含以下内容：

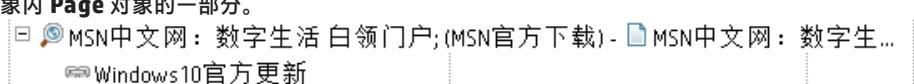
- [UFT 测试对象识别 - 概述](#) 28
- [练习 2a:从应用程序添加对象](#) 29
- [练习 2b:使用导航并识别进程创建对象存储库](#) 34

UFT 测试对象识别 - 概述

在创建和运行 GUI 测试时, UFT 使用“测试对象”识别应用程序中的对象, 然后根据应用程序的对象创建测试步骤。这些测试对象以 UFT 的“测试对象模型”为基础。

测试对象模型是 UFT 用于表示应用程序中对象的对象类型或类的大型集合。每个测试对象类都具有 UFT 可用于了解对象的标识属性列表、这些可唯一标识该类对象的属性的子集, 以及 UFT 可在对象上执行的一组相关操作。

设计并运行测试时, 此处有两种不同的对象类型:

测试对象	测试对象作为应用程序中 UFT 创建的实际对象的表示进行存储。UFT 通过识别应用程序中对象的属性和值的选择集来创建测试对象。然后, UFT 将存储对象上的有助于在运行会话期间识别并检查对象的信息, 并在运行会话期间使用数据识别应用程序对象。 每个测试对象都是较大测试对象层次结构的一部分。例如, Link 对象可以是 (Web) Browser 对象内 Page 对象的一部分。  顶层对象 (例如 Browser 对象) 称为容器对象, 因为它们可以包含较低级别的对象 (例如 Page 对象或 Frame 对象)。
运行时对象	运行时对象是 UFT 在运行会话期间执行操作 (方法) 的应用程序中的实际对象。UFT 识别运行时对象的属性并将其转换到测试对象中。

UFT 识别应用程序中的对象时, 它将相应的测试对象添加到“对象存储库”中。此对象存储库将用作测试对象的仓库。当 UFT 运行测试时, 它将在测试的对象存储库中查找测试步骤中包含的对象。

将对象添加到对象存储库时, UFT 将:

- 识别代表应用程序中已识别对象的 UFT 测试对象类, 并创建相应的测试对象。
- 在应用程序中读取对象属性的当前值, 并将标识属性和值的列表与测试对象一起存储。
- 为测试对象选择唯一名称。

此处有两种不同类型的对象存储库:

共享对象存储库	共享对象存储库是在单个测试中独立存在的对象存储库。共享对象存储库中的测试对象可用于多个测试/操作。此类型对象存储库的这一特点使其成为存储和维护测试对象的首选存储库类型, 因为使用该共享对象存储库, 您对测试对象进行的任何更新均会应用到所有测试。
本地对象存储库	本地对象存储库包含在特定操作上下文中使用的测试对象。这种类型的对象存储库不能用于其他操作。默认情况下, 所有操作均有一个本地对象存储库。

创建对象存储库时, 建议仅包括测试所需要的测试对象。这使对象存储库相对较小, 并有助于简化维护和对象选择。另外, 确保您提供了逻辑名称, 以便其他人可以在创建或修改测试时轻松地选择正确的对象。

对象存储库还可能包括检查点和输出对象。检查点对象类型将在[课程 5:参数化步骤和对象 \(第 66 页\)](#)中进行说明。

练习 2a: 从应用程序添加对象

在[UFT 测试对象识别 - 概述 \(第 28 页\)](#)中，您学习了有关 UFT 测试对象模式以及 UFT 如何识别和存储对象的内容。

在本练习中，您将使用 UFT 的对象识别功能识别对象并创建对象存储库。

1. 启动 UFT 打开 Book Flights 测试。

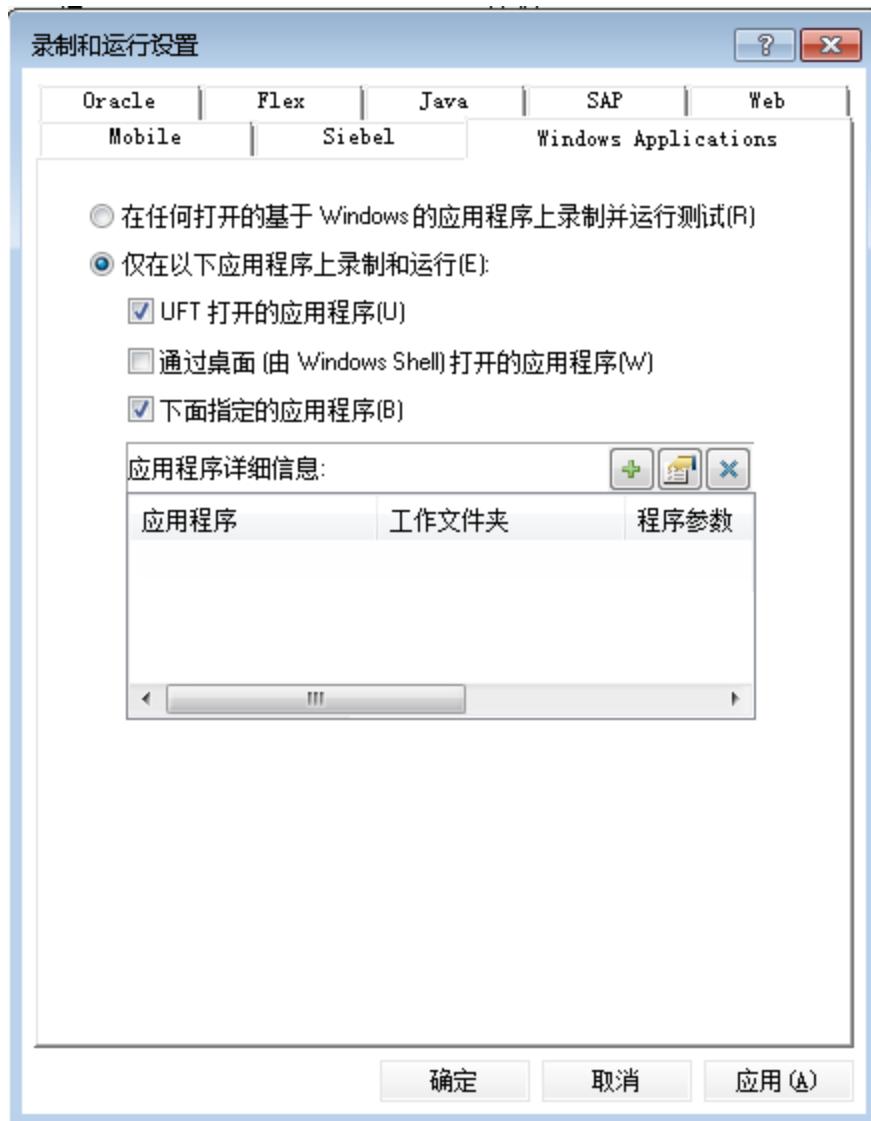
- a. 如果 UFT 未打开，请将 UFT 打开，如[为所有测试创建解决方案 \(第 20 页\)](#)中所述。
- b. 在起始页上的“最近使用的测试/组件”区域中，单击“Book Flights”。

将打开 Book Flights 测试，其中显示您在[课程 1: 创建 GUI 测试和操作 \(第 24 页\)](#)中创建的 Book Flights 测试 (及其操作)。

2. 为 UFT 设置识别设置。

- a. 选择“录制”>“录制和运行设置”。将打开“录制和运行设置”对话框。
- b. 在“录制和运行设置”对话框中，选择“Windows 应用程序”选项卡。
- c. 在“Windows 应用程序”选项卡中，选择“仅在以下应用程序上录制和运行:”选项。

- d. 在“仅在以下应用程序上录制和运行:”选项下,选择“UFT 打开的应用程序”和“下面指定的应用程序”选项的复选框:



- e. 在“应用程序详细信息”区域中,单击“添加”按钮 .
- f. 在“应用程序详细信息”对话框中,输入到应用程序的路径,以及应用程序的工作文件夹:
- 应用程序: <UFT 安装文件夹>\samples\Flights Application\FlightsGUI.exe
 - 工作文件夹: <UFT 安装文件夹>\samples\Flights Application
- g. 选择“启动应用程序”选项,并单击“确定”。
- h. 在主“录制和运行设置”对话框中,单击“确定”。稍后,当您在应用程序上录制步骤或为应用程序运行测试时,UFT 将可以使用该应用程序。

3. 打开航班预订应用程序。

如果尚未打开，则打开航班预订应用程序。

备注: 可以在打开 UFT 之前或之后打开航班预订应用程序。

4. 查看对航班预订应用程序中某些对象可用的属性和操作。

- a. 在工具栏上，单击“对象侦测器”按钮 。将打开“对象侦测器”对话框：



- b. 将“对象侦测器”对话框拖到应用程序一侧。这使您可以更清楚地查看应用程序上要侦测的对象。
c. 验证是否按下了“侦测时使对象侦测器保持在最上”切换按钮 。
d. 单击指向手按钮 .

按下指向手时，UFT 将隐藏，而“对象侦测器”对话框将显示在航班预订应用程序之上。

提示: 如果需要在航班预订应用程序、UFT 或任何其他打开的窗口之间来回切换，则按

CTRL 将指向手更改回常规 Windows 指针。在需要使用指针时一直按住 **CTRL** 按钮，然后在准备使用指向手时释放按钮。

- e. 将鼠标光标悬停在页面上的各个对象上方，查看“对象侦测器”对话框中发生的情况。

备注: 如果 UFT 无法在正确位置识别您的对象，请检查您是否正在以 100% 视图查看应用程序或页面，而没有放大或缩小当前视图。

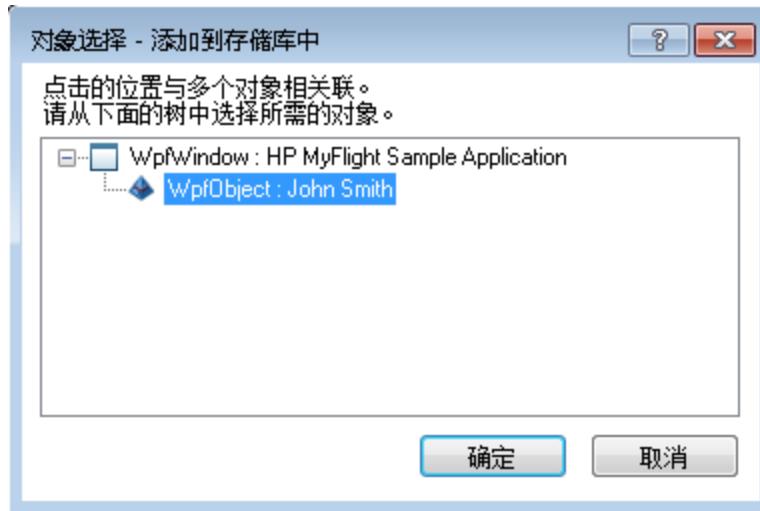
例如，如果您以 90% 或 120% 查看页面，则您会被要求单击或选择实际对象左侧或右侧的某个区域，以便识别该对象。

- f. 单击“用户名”编辑框。将显示该对象的测试对象层次结构。在层次结构框中，注意对象的名称是 WpfEdit，这是其对象类的类型。
- g. 在应用程序的“登录”窗口中，单击“用户名”编辑框内部。此操作将激活应用程序中的对象。
- h. 在 UFT 的“对象侦测器”对话框中，再次单击指向手按钮 。在“对象层次”对话框中，可以看到对象侦测器显示为 **agentName**。
- i. 关闭“对象侦测器”对话框。

5. 将测试所需的对象添加到对象存储库中。

在此步骤中，您将使用“侦测器”进一步处理步骤并指示 UFT 仅识别测试所需的对象，并将它们添加到对象存储库。

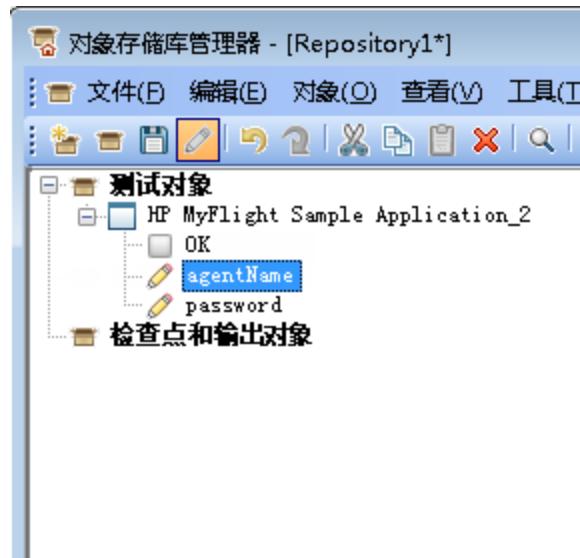
- a. 在应用程序的“登录”窗口中，单击“用户名”编辑框内部。
- b. 在 UFT 中，选择“资源”>“对象存储库管理器”。将打开“对象存储库管理器”窗口。
- c. 在“对象存储库管理器”窗口中，选择“对象”>“添加对象”。UFT 和对象存储库管理器将隐藏。
- d. 在“登录”页面中，再次单击“用户名”编辑框。将打开“对象选择 - 添加到存储库中”对话框：



- e. 在“对象选择”对话框中，选择“agentName”对象，并单击“确定”。“agentName”对象将与其父对象“Login”窗口对象一起添加到对象存储库中。

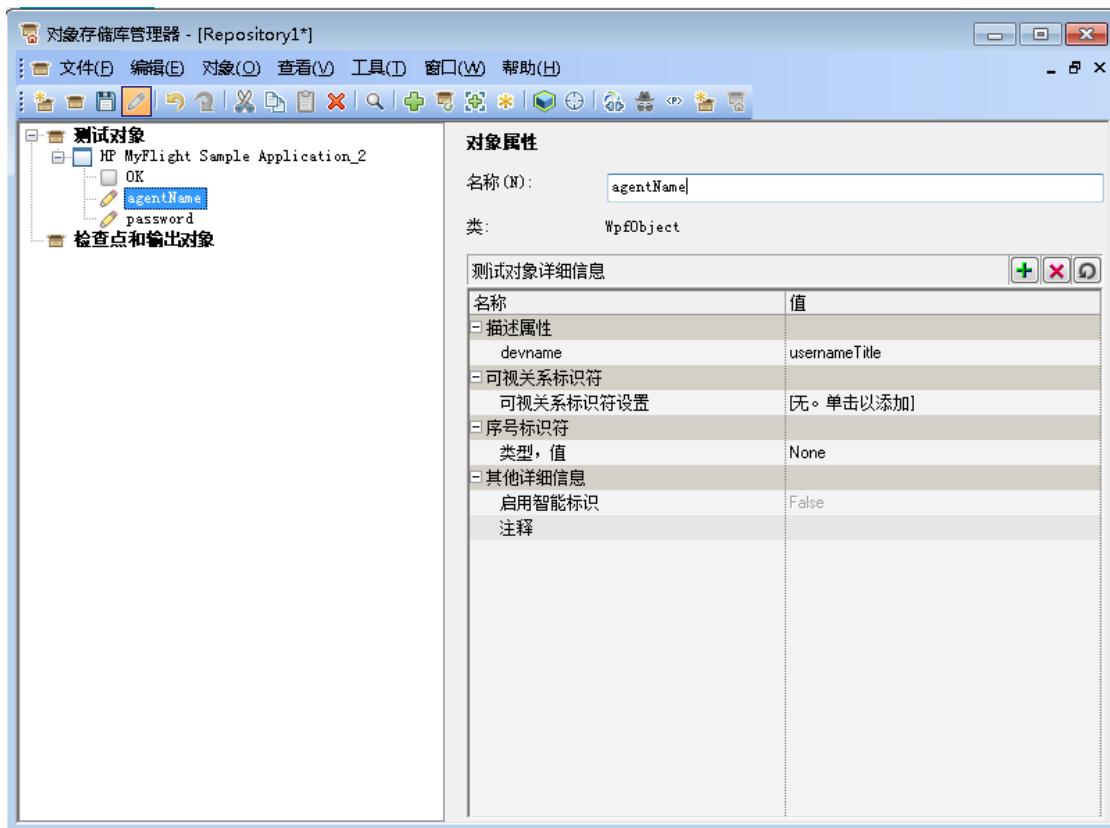
- f. 重复执行上述过程为“密码”编辑框和“确定”按钮添加对象。

将所有对象添加到对象存储库后，存储库应类似于：



6. 查看 UFT 对其中一个对象的识别情况。

在“测试对象”树中，选择“agentName”对象，并注意在对象存储库右窗格中显示的对象属性。这些是 UFT 用来在运行会话期间标识对象的描述属性：



7. 保存对象存储库。

- a. 在“对象存储库管理器”窗口中，单击“保存”
- b. 浏览到保存解决方案和测试的文件夹 C:\%HOMEPATH%\My Documents\Unified Functional Testing。在该文件夹中，新建名为 Tutorial_Object Repositories 的文件夹，并将其打开。
- c. 在 Tutorial_Object Repositories 文件夹中，使用名称 Login 保存对象存储库并单击“确定”。

备注: 此步骤仅保存对象存储库。在保存时，该对象存储库并未与测试相连，即使在 UFT 主窗口打开了测试也如此。

8. 将对象存储库与 Login 操作关联。

将对象存储库与操作关联使您可在关联的操作中的任何步骤中使用来自该存储库的任何对象。

备注: 同一对象存储库可与多个测试和操作关联。

- a. 打开主 UFT 窗口。
- b. 在主 UFT 窗口中，打开解决方案浏览器。
- c. 在解决方案浏览器中，在“Book Flights”节点下的“Login”节点。
- d. 右键单击 Login 操作并选择**将存储库与操作关联**。将打开“打开共享对象存储库”对话框。
- e. 在“打开共享对象存储库”对话框中，导航到在之前步骤中保存对象存储库的文件夹，并选择 Login.tsr 文件。单击“打开”。
- f. 在打开的对话框中，单击是。

对象存储库现在与 Login 操作关联，并作为该操作的子元素显示在解决方案浏览器中。

9. 保存测试。

选择“文件”>“保存”保存测试。

现在，您已学会了使用 UFT 识别应用程序对象，并创建了包含这些对象的对象存储库，并将对象存储库与测试相关联。所以，您随时可以创建应用程序其他区域的其他对象存储库。请继续[练习 2b: 使用导航并识别进程创建对象存储库 \(第 34 页\)](#) 了解更多信息。

练习 2b: 使用导航并识别进程创建对象存储库

在[练习 2a: 从应用程序添加对象 \(第 29 页\)](#) 中，您学习了在应用程序中如何使用 UFT 添加对象，如何为航班预订应用程序的“登录”页面创建共享对象存储库，以及将对象存储库与测试相关联。

在本课程中，您将为该站点中剩余的每个页面创建共享对象存储库。您将会使用“导航并识别”进程，使用该进程，您可以立刻识别页面或应用程序部分中的所有对象。

提示: 建议您始终为应用程序或网站中的每个部分/页面创建单独的共享对象存储库。这使您能够在添加或修改测试步骤或执行维护任务时更容易地找到正确的对象。

在本练习中，您将指示 UFT 识别每个页面上的所有对象。此时，您无需将对象存储库与特定操作关联。

1. 登录到航班预订应用程序的 Flight Finder 页面。

- a. 如果尚未打开，则打开航班预订应用程序。
- b. 在“登录”页面中，输入登录凭据：
 - 用户名: john
 - 密码: hp
- c. 单击 **OK**。将打开 Book Flight 页面。

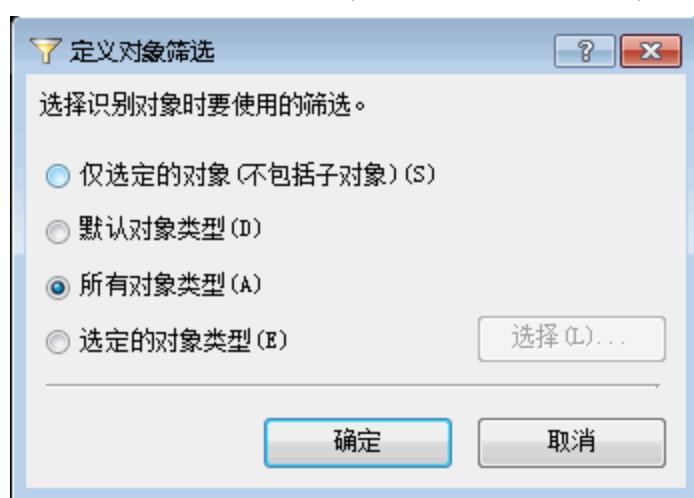
2. 新建共享对象存储库。

- a. 如果“对象存储库管理器”已关闭，则通过在 UFT 中选择“资源”>“对象存储库管理器”打开它。
- b. 在“对象存储库管理器”窗口中，单击“新建”。将打开空白的共享对象存储库。

3. 通过设置“定义对象筛选”，启动“导航并识别”进程。

除了如在之前的练习中那样单独添加对象，还可以使用“导航并识别”机制在某进程的应用程序中识别并添加所有对象。

- a. 在“对象存储库管理器”窗口中，选择“对象”>“导航并识别”。UFT 和对象存储库管理器将隐藏。
- b. 在“导航并识别”工具栏中，单击“定义对象筛选”按钮。将打开“定义对象筛选”对话框。
- c. 在“定义对象筛选”对话框中，选择“所有对象类型”，并单击“确定”。



4. 从 Flight Finder 页面识别所有对象。

在此步骤中，您将指示 UFT 识别 Book Flight 页面中与您的筛选匹配的所有对象，并将它们添加到共享对象存储库。

- a. 在航班预订应用程序的 Book Flight 页面中，单击应用程序的标题栏，使 UFT 要识别的页面处于活动状态。

- b. 在“导航并识别”工具栏中，单击**识别**。当 UFT 开始将页面上的对象表示添加到新对象存储库时，应用程序将闪烁，并显示“添加对象”消息框。

备注: 添加这些对象需要几秒钟时间。请不要在“导航并识别”进程运行时与应用程序进行交互。

- c. 关闭“导航并识别”工具栏。UFT 和“对象存储库管理器”窗口将再次可见。

5. 保存共享对象存储库。

- a. 在“对象存储库管理器”窗口中，单击“保存”。将打开“保存共享对象存储库”对话框。
- b. 浏览到之前练习创建的 **Tutorial_Object Repositories** 文件夹。
- c. 将此对象存储库命名为 Flight Finder 并单击“确定”。

6. 为剩余应用程序页面创建对象存储库。

- a. 使用上一步中所述的过程，为以下每个页面新建共享对象存储库：
 - **Select Flight**
 - **Flight Details**

重要事项: 识别 Flight Details 页面中的对象之前，请确保在“Passenger Name”框中输入字符串。此操作可以激活“Order”按钮，并使得 UFT 可以正确识别该按钮。在其他练习中，您会用到此按钮。

- b. 将对象存储库分别命名为 Select Flight 和 Flight Confirmation。

7. 将 Flight Finder 对象存储库与 Flight Finder 操作关联。

- a. 切换到 UFT 窗口。如果解决方案浏览器尚未打开，则通过单击 UFT 窗口左下角的“解决方案浏览器”选项卡将其打开。
- b. 在解决方案浏览器的 Book Flights 节点中，右键单击“Flight Finder”节点并选择“将存储库与操作关联”。将打开“打开共享对象存储库”对话框。
- c. 浏览到 **Tutorial_ObjectRepositories** 文件夹。
- d. 在 Tutorial_ObjectRepositories 文件夹中，选择“Flight Finder.tsr”文件并单击“打开”。
- e. 在打开的对话框中，单击是。

对象存储库现在已与 **Flight Finder** 操作关联，并作为该操作的子项显示在解决方案浏览器中。

8. 将剩余的对象存储库与相关操作关联。

将对象存储库与操作关联，如下所示：

操作	对象存储库
Select Flight	Select Flight.tsr
Flight Confirmation	Flight Confirmation.tsr

随后将步骤添加到每个操作时，所有需要的测试对象都将变为可用。

9. 保存测试。

单击“保存”。

现在，您已创建了对象存储库并已将其与测试相关联，您可以使用这些对象创建测试步骤了。请继续课程 3:向测试添加步骤 (第 38 页) 学习创建测试步骤。

课程 3:向测试添加步骤

在“[课程 1:创建 GUI 测试和操作](#)”中，您创建了测试和操作以为航班预订应用程序的测试提供结构。在“[课程 2:创建对象存储库](#)”中，您为应用程序创建了带有测试对象的对象存储库。

在本课程中，您将学习创建应用程序测试的最终必需步骤。您将学习如何将测试步骤添加到 GUI 测试，使您可以在用户界面中准确运行用户操作测试。

此课程包含以下内容：

· 在 GUI 测试中添加测试步骤 - 概述	39
· 练习 3a:在关键字视图中向 Login 操作添加步骤	39
· 练习 3b: 通过录制将步骤添加到 FlightFinder 操作	45
· 练习 3c:使用工具箱窗格将步骤添加到 Select Flight 操作	48
· 练习 3d:使用步骤生成器将步骤添加到 Book Flight 操作	51
· 高级练习 3e (可选) - 使用编辑器添加步骤	56

在 GUI 测试中添加测试步骤 - 概述

要在 GUI 测试中创建测试步骤，必须使用测试步骤中的对象并指示 UFT 要对测试对象执行什么操作。这样，UFT 就可以通过将测试对象方法（操作）转换为要对应用程序中的运行时对象执行的操作，来回放要在应用程序上执行的操作。

为协助实现这一目的，UFT 提供了多种添加测试步骤的方式：

关键字视图	使用关键字视图，您可以在步骤网格中选择测试对象，并为这些测试对象添加必要的操作（方法）。关键字视图将根据需要自动为对象层次结构排序。 选择相应的对象和方法后，您的测试步骤将显示在网格中，该网格中显示了对象名称、对象方法、添加的所有参数和步骤的文档摘要。
编辑器	在编辑器中，您可以输入对象（需要时可包括对象必要的层次结构）以及对象方法和参数。如果您在编写应用程序代码方面很有经验，则使用代码可以成为一种更简单的创建测试步骤的方式。
录制	UFT 还具有内置的录制机制，可自动识别应用程序中的测试对象和您对应用程序的对象执行的操作。在应用程序中进行录制时，UFT 会将您的操作转换为测试步骤，这些步骤显示对象名称和要对对象执行的操作（方法）。这样您就可以执行用户将执行的操作，然后使用 UFT 自动创建测试，无需手动在 UFT 中进行编辑。
工具箱窗格	在文档窗格中选择 GUI 操作选项卡时，UFT 将自动在“工具箱”窗格中显示该操作的关联对象和函数。然后，您可以将这些测试对象（或函数）拖动到关键字视图或编辑器中，UFT 将自动使用对象创建步骤。（但从“工具箱”窗格中拖动对象后，仍需要提供该对象的方法。）
步骤生成器	使用“步骤生成器”对话框，您可以在单个对话框中选择和提供测试步骤的所有详细信息。通过步骤生成器，您可以选择当前与选定操作关联的任何测试对象、该操作的方法以及必要的参数。选择此信息后，UFT 将在选定位置插入一个包含所有详细信息的步骤。

在接下来的练习中，您将使用其中每个方法创建测试步骤。

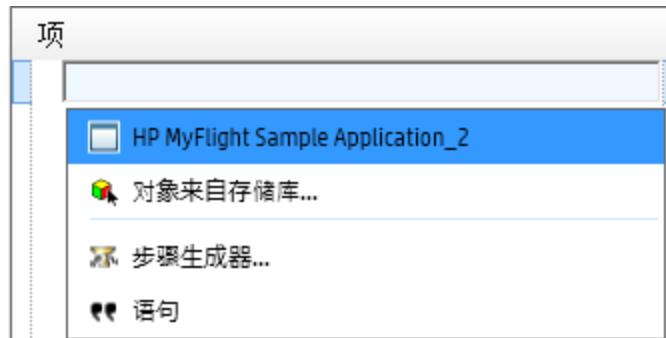
练习 3a: 在关键字视图中向 Login 操作添加步骤

在此练习中，您将使用关键字视图在 Login 操作中插入步骤。

1. 启动 UFT 打开 Book Flights 测试。
 - a. 如果 UFT 当前未打开，请按[为所有测试创建解决方案（第 20 页）](#)中所述将其打开。确保已加载 WPF 插件。
 - b. 在起始页上的“最近使用的解决方案”区域中，单击“Flight Reservation Application”。
将打开 Flight Reservation Application 解决方案，其中显示您在[课程 1: 创建 GUI 测试和操作（第 24 页）](#)中创建的 Book Flights 测试。
2. 打开 Login 操作并显示“关键字视图”。
 - a. 如果测试流画布当前未打开，请单击文档窗格中的“Book Flights”选项卡显示该画布。
 - b. 在画布中，双击 Login 操作。
在文档窗格中，Login 操作将在单独的选项卡中打开。
 - c. 如果显示编辑器，则在工具栏中单击“关键字视图”按钮  以显示“关键字视图”。

3. 添加第一个步骤以登录航班预订应用程序。

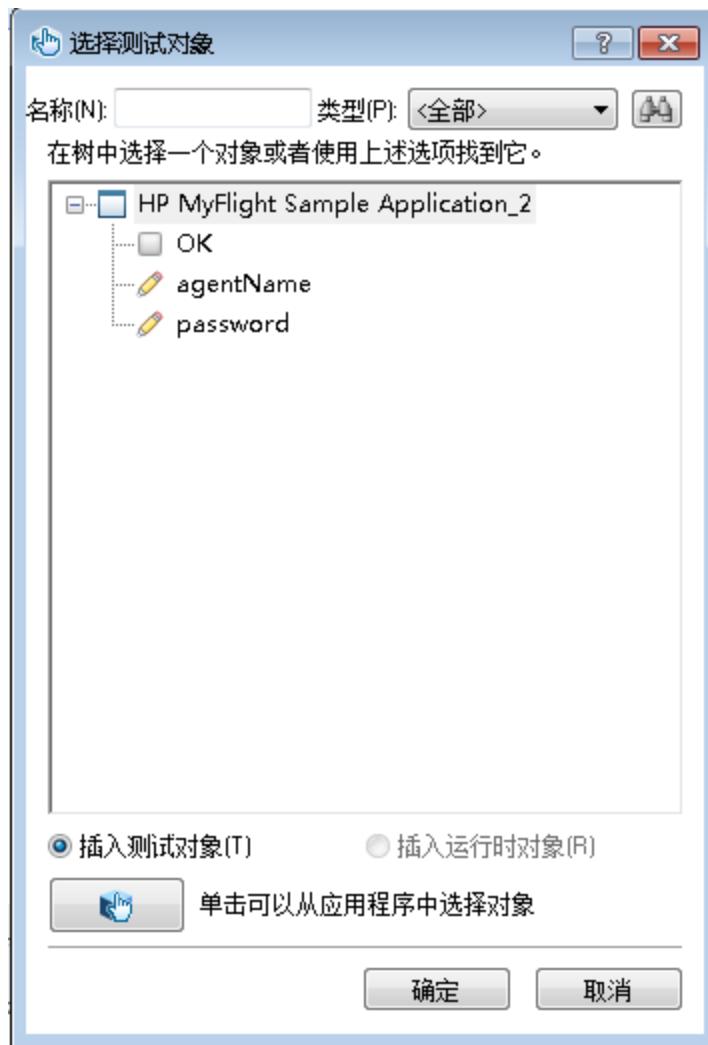
- a. 在“关键字视图”的“项”列中，单击“新建步骤”按钮。“项”列的网格中出现一个空的编辑字段。
- b. 在“项”列中单击该空编辑字段。将打开“项”下拉列表，显示相关对象存储库中的顶层父对象，并指示您选择一个项。



在此步骤中，父对象为“Login”窗口。现在请“勿”选择此对象，因为您并不想在主窗口上执行操作。您仅需在执行操作的对象上插入步骤。

- c. 选择对象来自存储库以打开“选择测试对象”对话框。

d. 在“选择测试对象”对话框中，展开测试对象树:



e. 在测试对象树中，选择“agentName”并单击“确定”。

将关闭“选择测试对象”对话框，并将一个步骤添加到操作。

请注意，三行将添加到“关键字视图”。尽管 UFT 不会在父测试对象上执行操作，但是会为每个父测试对象添加一行。这些行是执行步骤的对象的路径的一部分

运行会话期间，UFT 使用父对象来标识需要执行操作的实际对象。

在此步骤中，由三个新行的最后一个表示：

- 选定的 **agentName** WpfEdit 对象添加到“项”单元格。
- 默认方法 **Set** 添加到“操作”单元格。
- 文本添加到文档单元格中，指示此步骤将清除编辑框中的文本。这是因为此步骤在“值”单元格中缺少一个必需值，需要使用用户名进行更新。

项	操作	值	文档
HP MyFlight Sample Application			
agentName	Set		清除 "agentName" edit box 中的文本。

- f. 单击“值”单元格并输入 john。输入字符串，然后按“Enter”。

插入此值即完成该步骤。当您单击“关键字视图”中的其他区域时，此步骤的文档将在“文档”单元格中更新：

项	操作	值	文档
HP MyFlight Sample Application_2			
agentName	Set	"John"	输入 "John" (在 "agentName" edit box 中)。

提示: 引号会自动添加到您在“值”列中输入的值两旁，指示这是“字符串”值。如果方法支持“索引”值，且您输入了“索引”值，则不会添加引号。

- g. 选择“查看”>“编辑器”以显示编辑器，其中会显示 VBScript 中步骤的语法：

```
WpfWindow("HP MyFlight Sample Application").WpfEdit("agentName").Set "john"
```

注意，此步骤正在名称为 agentName 的 **WpfEdit** (编辑框) 测试对象上执行，并且：

- 测试对象前会显示 WpfEdit (编辑框) 测试对象的测试对象层次结构。在此步骤中，层次结构包含 WpfWindow 对象。
- 测试对象后会紧接着显示要在对象上执行的方法。在此步骤中，方法为 **Set**。
- 要在“agentName”编辑框中输入的文本将立即在“Set”方法后显示。文本显示在引号内，指示此值为字符串。在此步骤中，要输入的文本为 john。
- 句号 (句点) 分隔了步骤的各个部分。

- h. 单击“关键字视图”按钮  返回到“关键字视图”。

4. 添加下一个步骤。

- a. 在“项”列中“agentName”行的正下方，单击“新建步骤”按钮。将添加另一个包含编辑字段的行。
- b. 单击空白的编辑字段。将打开“项”列表，列出上一步骤中测试对象的同级对象。
- c. 从“项”列表选择 **password**。此时，仅会添加一个新行，因为对象将与上一步骤共享相同的父对象。

在此步骤中：

- **password** WpfEdit 测试对象添加到“项”单元格。
- 默认方法 **Set** 添加到操作单元格。需要更改此方法，因为需要对密码进行编码。
- 文本添加到文档单元格中，指示此步骤将清除编辑框中的文本。这是因为此步骤在值单元格中还缺少一个必需值，需要使用 password 进行更新。
- d. 在操作单元格中单击，以显示向下箭头，然后单击向下箭头以显示所选测试对象的可用方法列表。

e. 在方法列表中, 选择 “SetSecure”。此方法支持使用加密文本。

5. 使用 HP 密码编码器应用程序生成编码的密码。

- a. 选择 “开始” > “所有程序” > “HP Software” > “HP” > “**HP Unified Functional Testing**” > “Tools” > “Password Encoder” 或 <UFT 安装文件夹>\bin\CryptonApp.exe。
- b. 在密码编码器的“密码”框中, 输入 hp。
- c. 单击**生成**。密码编码器将对密码进行加密, 并在**编码字符串**框中显示该密码。
- d. 单击**复制**。
- e. 在 UFT 中, 在 Login 操作的密码步骤的“值”单元格中粘贴编码值, 然后按 **ENTER** 键。
此步骤的“文档”列中将显示更新的文档。
- f. 关闭“密码编码器”对话框。
如果在此时运行操作, UFT 会自动打开航班预订应用程序, 并插入您在“用户名”和“密码”框中指定的值。

6. 在 Login 操作中插入最后一个步骤。

- a. 在最后一个步骤下面的“项”列中单击“新建步骤”, 以插入下一个步骤。将打开空白的编辑字段。
 - b. 单击该编辑字段。将打开“项”列表, 列出上一步骤中测试对象的同级对象。
 - c. 从“项”列表中选择“确定”。
- 此步骤指示 UFT 在输入应用程序的用户名和密码后单击“确定”。

7. 保存测试。

选择“文件”>“保存”。

要了解更多有关关键字视图和编辑器的信息, 请继续[在关键字视图和编辑器中分析 Login 操作 \(第 43 页\)](#)。

要继续向测试添加步骤, 请转到[练习 3b: 通过录制将步骤添加到 FlightFinder 操作 \(第 45 页\)](#)。

在关键字视图和编辑器中分析 Login 操作

您已经创建了一些测试步骤, 让我们来看一下这些步骤在关键字视图和编辑器中如何显示:

关键字视图

添加步骤后, 关键字视图应与此类似:

项	操作	值	文档
HP MyFlight Sample Application			
agentName	Set	"John"	输入 "John" (在 "agentName" edit box 中)。
password	SetSecure	"55f29215900ab4b14c77"	在 "password" edit box 中输入加密密码。
OK	Click		单击 "OK" button。
+ 新建步骤			

可以看到, 测试中的这些步骤排列在基于图标的网格中, 该网格显示了测试步骤顺序和对象层次结构。关键字视图中的每一行代表一条不同的信息:

行				描述
	HP MyFlight Sample Application			“HP MyFlight Sample Application”窗口对象是此操作中包含的测试对象的父对象。 以下步骤中包含的所有测试对象显示为此对象的子节点。
	agentName Set "John"			agentName 是 UFT 对其执行操作的编辑框的名称。 Set 方法是对 agentName 对象执行的操作。 UFT 在编辑框中输入 john 作为文本。
	password SetSecure			password 是 UFT 对其执行操作的编辑框的名称。 SetSecure 方法是对 passwordWatermakr 对象执行的操作。 UFT 将在 password 编辑框中输入数字字符串作为文本。
	OK Click			OK 是在 agentName 和 password 编辑框中输入信息后 UFT 单击的按钮的名称。 Click 方法是对该按钮执行的操作。

关键字视图中的每个步骤都有许多不同的元素:

关键字视图元素	描述
项	步骤的项 (测试对象、实用程序对象、函数调用或语句)。此项显示在分层的、基于图标的树中。
操作	要对项执行的操作, 如 Click 、 Set 或 Select 。
值	选定操作的参数值 (如需要)。例如, 要在编辑框中输入的文本, 或单击图像时要使用的鼠标按钮。
文档	自动提供的关于步骤操作的语句, 使用易于了解的句子进行描述。例如 Click the "OK" button。
分配	向变量或从变量分配一个值, 以便可以稍后在测试中使用该值。默认情况下此列不可见。
注释	要添加的与步骤有关的任何文本信息。例如, 您可以添加注释 Return to page used in first step of test。 默认情况下此列不可见。

提示: 通过右键单击 “关键字视图” 中的列标题并从列表选择列名称, 可选择隐藏或显示各列。

有关关键字视图的更多详细信息, 请参阅《HP Unified Functional Testing 用户指南》中的关键字视图部分。

编辑器

添加步骤后, 编辑器应与此类似:

```

Main
1 WpfWindow("Login").WpfEdit("agentName").Set "john"
2 WpfWindow("Login").WpfEdit("password").SetSecure "53b3cd55b30d8acf6af1"
3 WpfWindow("Login").WpfButton("OK").Click
4

```

与关键字视图不同, 在编辑器中, 每个步骤由一个脚本行表示, 其格式为:

<对象层次结构>. <方法> <方法参数>

因此, 通过每个步骤 (和脚本中的每一行), 可以看出许多信息:

测试对象层次结构	对于编辑器中的每个步骤, 您可以获得“完整的”对象层次结构, 包括: <ul style="list-style-type: none"> 测试对象类型 每个对象的对象名称 (如 UFT 所标识) 在上图示例的第一行中, 可以同时看到这两个元素: <ul style="list-style-type: none"> WpfWindow 是测试对象类型 "Login" 是对象名称 (如 UFT 所标识)
对象方法	除了对象层次结构, 您还可以看到方法 (要对对象执行的操作)。每个方法均显示为粗体文本。 在上图示例的第一行中, 对象执行 Set 方法。
方法参数	对于许多方法, 必须提供必需或可选的参数。在编辑器中, 这些参数在方法名称后列出。 在上图示例的第一行中, Set 方法将输入参数 John。

有关编辑器的详细信息, 请参阅 UFT 中有关编辑代码和文本文档的部分。

练习 3b: 通过录制将步骤添加到 FlightFinder 操作

在“[练习 3a:在关键字视图中向 Login 操作添加步骤](#)”中, 您通过使用关键字视图创建步骤来向要在航班预订应用程序的 Login 页面上运行的测试添加了步骤。

在此练习中, 您将录制为应用程序中 Flight Finder 页面创建的 Flight Finder 操作的步骤。此操作将使用 Flight Finder 共享对象存储库中包含的测试对象。

提示: 开始录制会话之前, 您可以将应用程序窗口和此教程窗口并排放置在屏幕上, 以便在录制过程中阅读教程。

1. 启动 UFT 打开 Book Flights 测试。

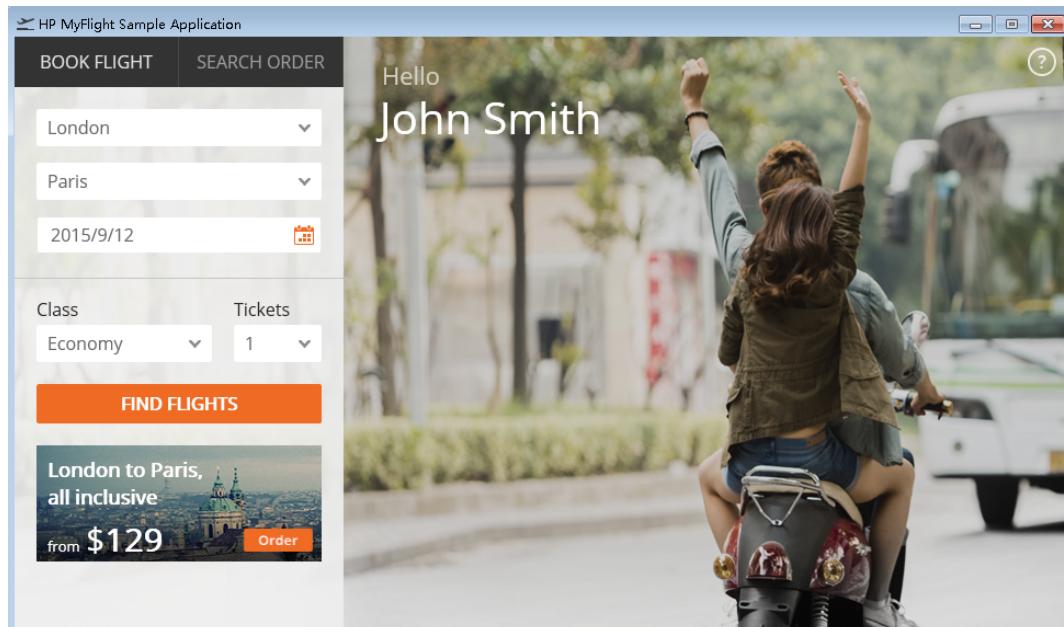
- a. 如果当前未打开 UFT, 请将 UFT 打开, 如[为所有测试创建解决方案 \(第 20 页\)](#) 中所述。确保已加载 WPF 插件。
- b. 选择“文件”>“打开”>“解决方案”。将打开“打开解决方案”对话框。
- c. 在“打开解决方案”对话框中, 导航到位于 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 中的 **Flight Reservation Application.ftsln** 文件, 然后单击“打开”。
将打开 Flight Reservation Application 解决方案, 其中显示您在[课程 1: 创建 GUI 测试和操作 \(第 24 页\)](#) 中创建的 Book Flights 测试。

2. 打开 Flight Finder 操作。

- a. 如果测试流画布当前未打开, 请单击文档窗格中的“Book Flights”选项卡显示该画布。
- b. 在画布中双击“Flight Finder”操作。
在文档窗格中, Flight Finder 操作将在单独的选项卡中打开。

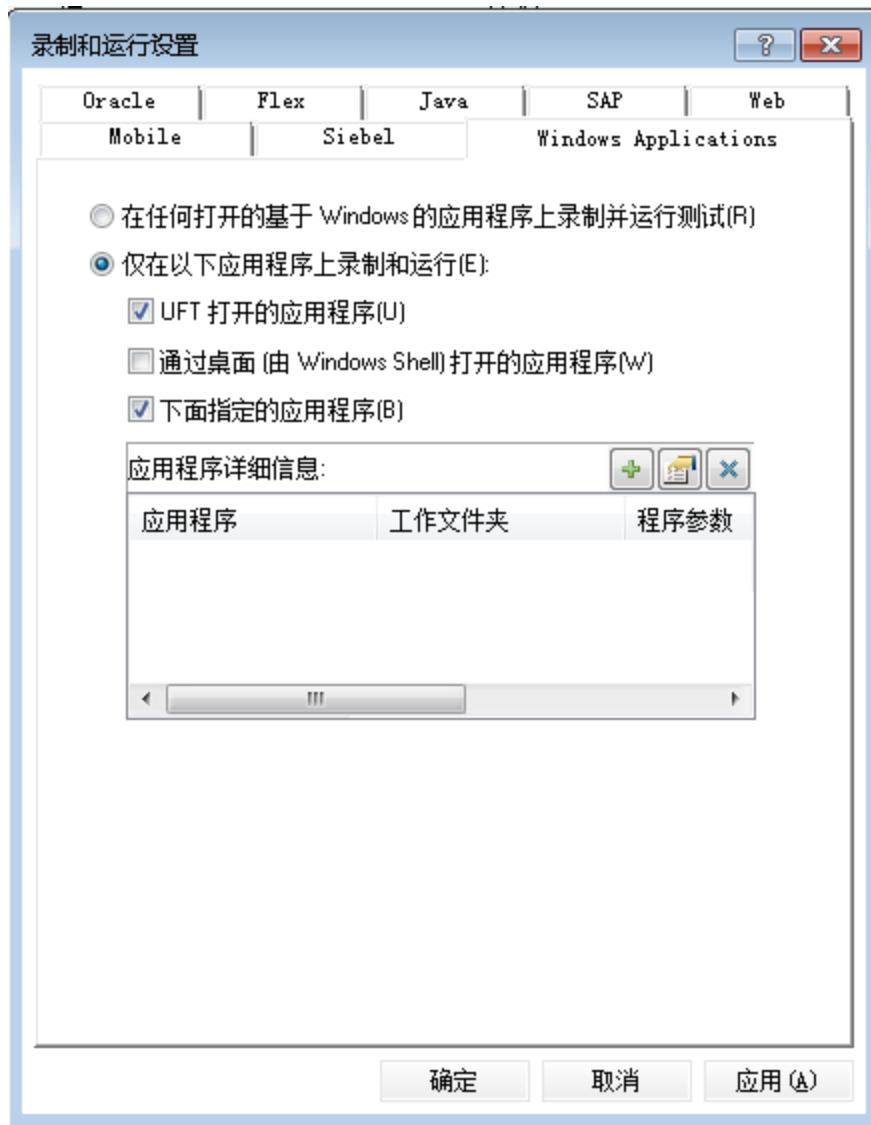
3. 打开航班预订应用程序

- a. 如果尚未打开航班预订应用程序, 请按[浏览航班预订应用程序 \(第 18 页\)](#) 中所述打开该应用程序。
将打开航班预订应用程序的“Login”页面。
- b. 登录应用程序。使用 john 作为用户名, hp 作为密码。
将打开应用程序的 Flight Finder 页面。



- c. 在 UFT 中, 选择“录制”>“录制和运行设置”。将打开“录制和运行设置”对话框。

- d. 在“录制和运行设置”对话框的“Windows 应用程序”选项卡中，选择“在任何打开的基于 Windows 的应用程序上录制并运行测试”。这使 UFT 能够在打开的应用程序上运行。



- e. 在 UFT 中，单击“录制”按钮 。将在屏幕顶部附近显示录制工具栏并且 UFT 开始录制。

UFT 窗口将消失，改为显示应用程序的 Flight Finder 页面。

- f. 在 Flight Finder 页面中，通过选择不同的项更改以下选择：

- **From:** Los Angeles
- **To:** Sydney
- **Date:** 明天的日期

重要事项: 在编辑框中输入日期。使用 **M/D/YYYY** 格式。

- **Class:** Business

- **Tickets:** 2

备注: 如果您在月末或年末执行此教程, 请在录制时选择不同的月份或年份。UFT 仅在您更改应用程序时才录制操作, 因此能够确保步骤的录制。如果接受默认值(或重新选择默认值), 那么 UFT 不会录制操作

- g. 完成选择后, 单击“FIND FLIGHTS”继续。将打开 Select Flight 页面。
- h. 在“录制”工具栏上, 单击“停止”按钮可停止录制。

现在您已预订了从旧金山到悉尼的虚拟机票。UFT 将从您单击 UFT 中的“录制”按钮时起录制您在应用程序中的操作, 直到您单击录制工具栏上的“停止”为止。

4. 保存测试。

单击“保存”。

请勿关闭测试, 因为您将继续向其他操作添加步骤。继续[练习 3c: 使用工具箱窗格将步骤添加到 Select Flight 操作 \(第 48 页\)](#), 将其他步骤添加到其他操作。

练习 3c: 使用工具箱窗格将步骤添加到 Select Flight 操作

在[练习 3b: 通过录制将步骤添加到 FlightFinder 操作 \(第 45 页\)](#)中, 通过使用 UFT 的录制功能向 Flight Finder 操作添加步骤。这完全以您在航班预订应用程序的用户界面中执行的操作为基础创建步骤。

在此练习中, 您将相关测试对象从“工具箱”窗格拖动到操作中。

备注: 列出的“工具箱”窗格项取决于文档窗格中处于活动状态的操作。如果“测试流”选项卡或“函数库”选项卡处于活动状态, 或者未打开任何测试, 则“工具箱”窗格将为空。

1. 启动 UFT 打开 Book Flights 测试。

- a. 如果当前未打开 UFT, 请将 UFT 打开, 如[为所有测试创建解决方案 \(第 20 页\)](#)中所述。
- b. 单击“打开”按钮向下箭头, 并选择“打开解决方案”。将打开“打开解决方案”对话框
- c. 在“打开解决方案”对话框中, 导航到位于 C:\%HOMEPATH%\My Documents\Unified Functional Testing 中的 Flight Reservation Application.ftsln 文件, 然后单击“打开”。将打开 Flight Reservation Application 解决方案, 其中显示您在[课程 1: 创建 GUI 测试和操作 \(第 24 页\)](#)中创建的 Book Flights 测试。

2. 打开 Select Flight 操作。

在“解决方案浏览器”窗格中, 双击“Select Flight”操作。
在文档窗格中, Select Flight 操作将在单独的选项卡中打开。

3. 显示“工具箱”窗格。

在 UFT 工具栏中, 单击“工具箱”按钮 。“工具箱”窗格将在 UFT 窗口的右侧打开。

备注: “工具箱”窗格显示操作的测试对象和函数, 与该操作是否在关键字视图或编辑器中显示无关。

4. 从网格中选择要预订的航班。

- a. 在“工具箱”窗格中, 展开“测试对象”树。
- b. 在“项”列中, 单击“新建步骤”按钮。将打开空白的编辑字段。
- c. 在“工具箱”窗格中, 找到“flightsDataGrid”对象并将其拖动到关键字视图或编辑器中的编辑字段(取决于打开的视图)。

备注: 您可能注意到此对象存储库的对象数要远超过 Login 存储库。这是因为创建此对象存储库时没有删除无关对象。

所选表测试对象与其默认方法 **SelectCell** 一起被添加到步骤中。

- 在“关键字视图”中, “文档”单元格为空, 因为尚未提供此方法的必需参数。请记住, 此步骤显示在“关键字视图”中的三行上, 因为父测试对象是步骤的一部分。
- 在编辑器中, 该步骤显示如下:

```
WpfWindow("HP MyFlight Sample Application").WpfTable  
("flightsDataGrid").SelectCell
```

- d. 如果关键字视图未打开, 请选择“查看”>“关键字视图”打开它。

- e. 在 flightsDataGrid 对象“值”列的中间, 单击“配置值”图标 。

备注: 单击此图标时, 将看到显示 row, Column 的工具提示。这说明您正在设置行参数的值。配置方法参数的值时, 您将始终看到一条工具提示, 通知您正在设置的参数。

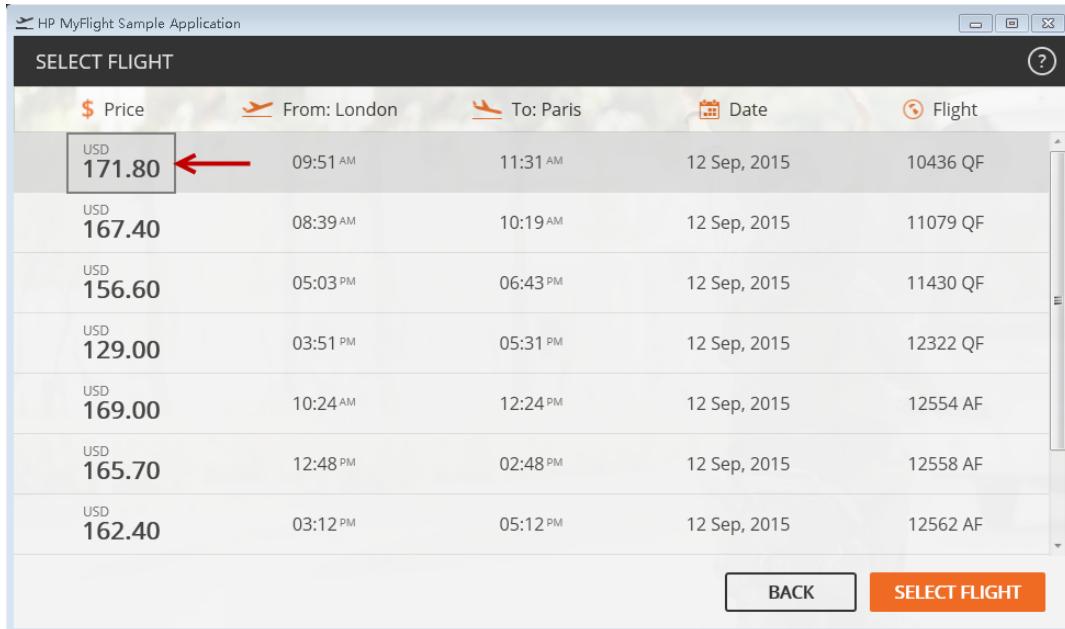
- f. 输入 0 作为行参数的值

- g. 在“值”列的右侧, 单击“配置值”图标 。

备注: 单击此图标后, 将看到显示 row, Column 的工具提示。这说明您正在设置“列”参数的值。

- h. 输入 0 作为列参数的值。

通过输入此参数，指示 UFT 选择第一行中的航班，如下例所示:



添加方法参数后，步骤将在关键字视图和编辑器中得到更新:

- 现在“关键字视图”中“值”列显示的步骤值为“0”, “0”。
- 编辑器在 **SelectCell** 方法后面添加“0”, “0”。现在编辑器中的语句如下所示:

```
WpfWindow("HP MyFlight Sample Application").WpfTable
("flightsDataGrid").SelectCell "0", "0"
```

- 现在“文档”列中包含说明该步骤操作的语句。

5. 添加步骤以单击 Select Flight 按钮。

从航班列表选择一个单元格后，还必须单击 Select Flight 按钮才能继续航班预订流程。

- a. 在“工具箱”窗格中，找到“SELECT FLIGHT”对象。
- b. 在“项”列中，单击“新建步骤”按钮。将打开空白的编辑字段。
- c. 从“工具箱”窗格中，将 SELECT FLIGHT 对象拖动到“关键字视图”的步骤网格中位于“flightsDataGrid”对象下方的编辑字段中。

已将一个包含 SELECT FLIGHT 对象的步骤添加到测试:

项	操作	值	文档
HP MyFlight Sample Application_2			
flightsDataGrid	SelectCell	0,0	选择行 0、列 0(在 "flightsDataGrid" table 中) 中的单元格。
SELECT FLIGHT	Click		单击 "SELECT FLIGHT" button。

在这种情况下，SELECT FLIGHT 对象的默认方法 **Click** 是要对该测试使用的方法。

在编辑器中，现在操作步骤如下所示:

```
WpfWindow("HP MyFlight Sample Application").WpfTable  
("flightsDataGrid").SelectCell "0", "0"  
WpfWindow("HP MyFlight Sample Application").WpfObject("SELECT FLIGHT").Click
```

6. 保存测试。

选择“文件”>“保存”。

请勿关闭测试，因为您仍需向其他操作添加步骤。继续[练习 3d:使用步骤生成器将步骤添加到 Book Flight 操作 \(第 51 页\)](#)学习通过使用步骤生成器添加步骤。

练习 3d:使用步骤生成器将步骤添加到 Book Flight 操作

在[练习 3c:使用工具箱窗格将步骤添加到 Select Flight 操作 \(第 48 页\)](#)中，您已使用“工具箱”窗格中显示的对象创建测试步骤。

在本课程中，您将使用另外一种方式 - 步骤生成器 - 来创建测试步骤。步骤生成器可用于在一个对话框中定义整个步骤，而不是在“关键字视图”的各列中插入步骤的不同部分。

1. 打开 Flight Confirmation 操作。

在解决方案浏览器中，在“Book Flights”节点下双击“Flight Confirmation”操作。

在文档窗格中，Flight Confirmation 操作将在单独的选项卡中打开。

2. 使用步骤生成器添加步骤。

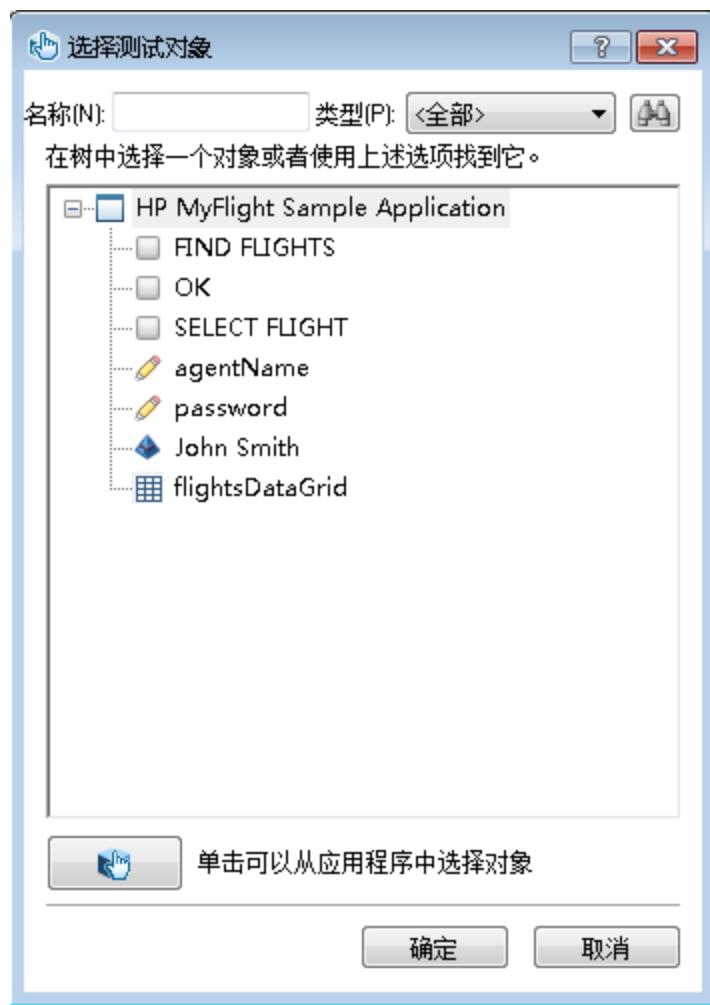
- 如果编辑器尚未显示，请通过选择“查看”>“编辑器”显示编辑器。

- b. 在编辑器的第一行中, 右键单击并选择“插入步骤”>“步骤生成器”。将打开“步骤生成器”对话框。



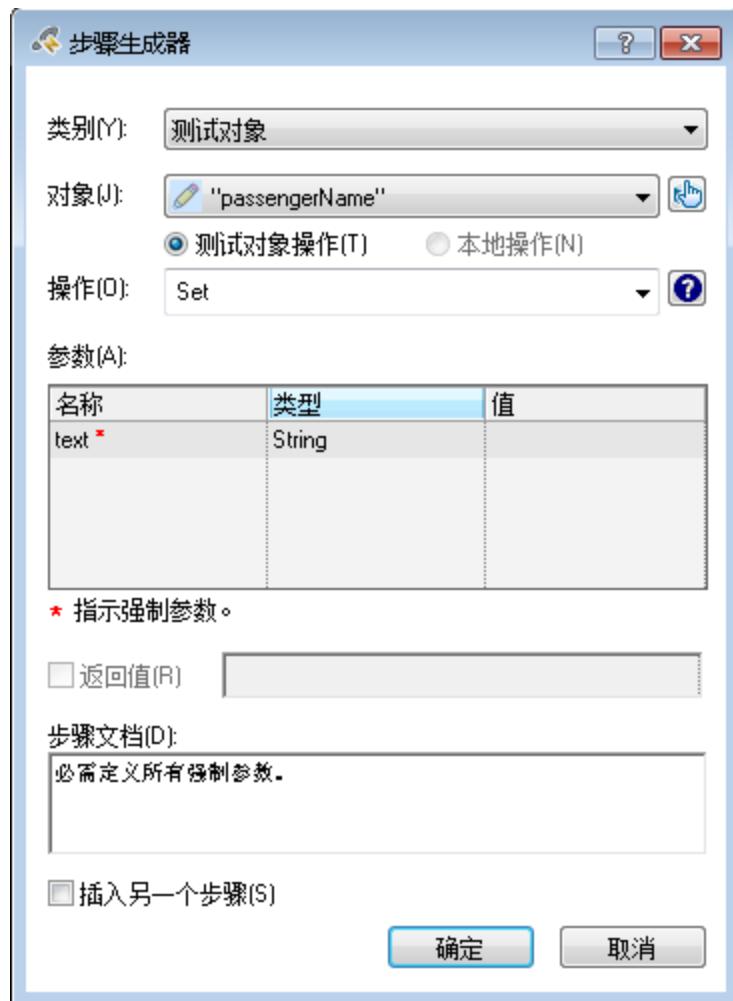
- c. 在“对象”下拉列表中, 选择“"Book Flights"”。
- d. 单击“选择对象”按钮 。将打开“选择测试对象”对话框。

- e. 在“选择测试对象”对话框中，展开“Book Flights”节点：



- f. 选择“passengerName”编辑框 ，然后单击“确定”。

步骤生成器将显示“passengerName”对象的默认选项:



g. 使用关键字视图中所用的相同方式在步骤生成器中定义参数和值:

- 在“操作”下拉列表中，将“操作”设置为“Set”。请注意，在“操作”下拉列表中选择测试对象时，将显示默认操作。您可以根据需要为此对象选择其他操作。但在此练习中，**默认操作是正确的操作。**
 - 在“参数”表的“值”列中单击（与在关键字视图中一样），然后输入您选择的名称。请注意，如果参数是强制的，则会在参数名称旁显示一个红色星号。
 - “步骤文档”显示此步骤的说明信息，其内容与“关键字视图”的“文档”单元格中的内容相同。
 - 选中“插入另一个步骤”复选框以在添加此步骤后再次打开“步骤生成器”对话框。
- h. 单击“确定”。步骤生成器保持打开状态，但已在后台将包含您所输入详细信息的步骤添加到编辑器中。

3. **添加一个步骤，等待应用程序窗口中的进度栏加载。**

在 Flight Details 窗口中，您将注意到对象存储库中有一个名为 **progBar** 的进度栏对象。为使应用程序在测试时正确运行，必须添加一个步骤以确保进度栏在单击按钮完成订单前加载。

- 在步骤生成器的“测试对象”下拉列表中，选择“progBar”对象。步骤生成器将使用 **progBar** 对象的默认属性，包括其默认方法、“值”更新对话框字段。
- 在“操作”下拉菜单中，选择“WaitProperty”。此方法指示 UFT 在测试运行期间等待，直到特定属性达到指定状态。
- 在“参数”表中，输入以下信息：

项	值
PropertyName	值
PropertyValue	100

- 选中“插入另一个步骤”复选框，然后单击“确定”。

将在后台在“关键字视图”中插入另一个步骤，“步骤生成器”框保持打开状态。

4. 使用步骤生成器向测试添加步骤以完成订单。

定义订单的详细信息后，必须提供在 Flight Details 窗口中单击“ORDER”和“NEW SEARCH”按钮以完成订单的步骤。您将再次使用步骤生成器插入此语句。

备注: 单击“ORDER”按钮的步骤必须插入在上一个步骤中创建进度栏对象的步骤之前。

- 使用先前步骤中的过程，输入以下详细信息：

	ORDER 按钮步骤	NEW SEARCH 按钮步骤
对象	ORDER (Wpf 按钮)	NEW SEARCH (Wpf 按钮)
操作	单击	单击
参数	留空	留空
“插入另一个步骤”复选框	选择	Clear

- 单击“确定”。将关闭步骤生成器，并将步骤添加到“关键字视图”中。

您已将两个步骤都添加到了测试，关键字视图应如下所示：

项	操作	值	文档
passengerName	Set	"John Smith"	输入 "John Smith" (在 "passengerName" edit box.)
ORDER	Click		单击 "ORDER" button。
progBar	WaitProperty	"value","100"	等到 "progBar" progress bar 的 "value" 属性值...
NEW SEARCH	Click		单击 "NEW SEARCH" button。
+ 新建步骤			

在编辑器中，这些步骤显示如下：

```

WpfWindow("HP MyFlight Sample Application").WpfEdit("passengerName").Set "John
Smith"
WpfWindow("HP MyFlight Sample Application").WpfButton("ORDER").Click
WpfWindow("HP MyFlight Sample Application").WpfProgressBar
("progBar").WaitProperty "value", "100"

```

```
WpfWindow("HP MyFlight Sample Application").WpfButton("NEW SEARCH").Click
```

5. 保存测试。

单击“保存”。

现在您完成了第一个测试的创建，可以运行该测试了。继续[课程 4:运行和分析 GUI 测试 \(第 61 页\)](#)了解更多有关运行测试的信息。

如果您想学习高级课程，请继续[高级练习 3e \(可选\) - 使用编辑器添加步骤 \(第 56 页\)](#)学习如何在编辑器中添加步骤。

高级练习 3e (可选) - 使用编辑器添加步骤

除了通过关键字视图、“工具箱”窗格或步骤生成器向测试添加步骤以外，您还可以将步骤直接添加到编辑器中。

但通过编辑器添加步骤时，您必须对应用程序和测试对象有更深的了解。在关键字视图、“工具箱”窗格和步骤生成器中，测试对象的所有信息均由 UFT 在对话框中提供。在编辑器中，您必须了解许多信息：

- 完整的测试对象层次结构
- 测试对象的名称（如对象存储库中所记录）
- 测试对象的类型，例如 WpfWindow、WpfButton 等。
- 要使用的方法

您使用以上这些信息在编辑器中创建行。在本课程中，您将了解在哪里查找这些信息以及如何将这些信息输入编辑器来创建测试步骤。

您将在编辑器中为已经具有测试步骤的操作创建语句。

1. 启动 UFT 打开 Book Flights 测试。

- a. 如果当前未打开 UFT，请将 UFT 打开，如[为所有测试创建解决方案 \(第 20 页\)](#)中所述。确保已加载 WPF 插件。
- b. 选择“文件”>“打开”>“解决方案”。将打开“打开解决方案”对话框。
- c. 在“打开解决方案”对话框中，导航到位于 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 中的 **Flight Reservation Application.ftsln** 文件，然后单击“打开”。将打开 Flight Reservation Application 解决方案，其中显示您在[课程 1:创建 GUI 测试和操作 \(第 24 页\)](#)中创建的 Book Flights 测试。

2. 打开 Flight Confirmation 操作。

在“解决方案浏览器”窗格中，双击“Flight Confirmation”操作。

“Flight Confirmation”操作将在文档窗格的单独选项卡中打开。请勿删除操作中现有的测试步骤。

3. 了解父对象的对象详细信息。

- 如果编辑器尚未打开, 请通过选择“查看”>“编辑器”打开。

请注意, 现有步骤如下所示:

```
WpfWindow("HP MyFlight Sample Application").WpfEdit("passengerName").Set  
"John Smith"  
WpfWindow("HP MyFlight Sample Application").WpfButton("ORDER").Click
```

有关这些语句中每个元素的描述, 请参阅[在关键字视图和编辑器中分析 Login 操作 \(第 43 页\)](#)中的“编辑器”部分。

- 在解决方案浏览器中, 在“Book Flights”节点下展开“Flight Confirmation”操作节点。
- 双击“Flight Confirmation.tsr”对象存储库。将打开对象存储库管理器, 其中包含航班预订应用程序的 Flight Details 页面的对象。
- 在对象存储库管理器中, 选择“HP MyFlight Sample Application”对象(顶层节点)。这是 Flight Details 页面中所有对象的父对象。

对象的详细信息和属性在“对象属性”区域中显示(窗口右侧):



- 记录“HP MyFlight Sample Application”对象的以下详细信息:

- **名称:** HP MyFlight Sample Application
- **类:** WpfWindow

在编辑器中创建语句时需要此信息。此信息是使用此页面上测试对象的步骤的所有语句的第一部分。

4. 了解子对象的对象详细信息。

[在练习 3d: 使用步骤生成器将步骤添加到 Book Flight 操作 \(第 51 页\)](#) (该练习是此练习的基础)

中, 您创建了两个步骤:一个步骤输入航班订单的名称, 第二个步骤单击“ORDER”按钮。为在编辑器中创建这些步骤, 还必须了解这些步骤中所包含对象的对象详细信息。

- 在对象存储库管理器中, 选择“passengerName”对象。测试对象详细信息显示在对象存储库管理器中的“对象属性”部分中:



- 记录对象的以下属性:
 - **名称:** passengerName
 - **类:** WpfEdit
- 执行与 **ORDER** 对象相同的步骤。

5. 为编辑器中的步骤创建语句。

查看此操作步骤所涉及对象的对象属性后, 您应了解以下信息:

对象	名称	Class
HP MyFlight Sample Application 窗口 (父对象)	HP MyFlight Sample Application	WpfWindow
Passenger Name (编辑框)	passengerName	WpfEdit
ORDER 按钮	ORDER	WpfButton

必须使用对象详细信息创建包含对象层次结构和要对对象执行的方法(操作)的语句。(每个对象均有多种可用于测试对象的受支持方法。有关所有可用对象及其方法的完整详细信息, 请在完成教程练习后参阅《HP UFT Object Model Reference for GUI Testing》。)

- a. 在第一个新行中, 按以下格式输入第一个步骤 (输入订单的客户姓名) 的父对象层次结构:

```
<对象类>("<对象名称>").
```

对于此步骤, 您应输入以下内容:

```
WpfWindow("HP MyFlight Sample Application").
```

- b. 使用相同的格式输入第一个步骤的子对象 (**passengerName**)。

对于此步骤, 您应输入以下内容:

```
WpfEdit("passengerName").
```

- c. 在 **WpfEdit("passengerName")** 对象后面, 为 **passengerName** 对象输入 **Set** 方法。

您的步骤应如下所示:

```
WpfWindow("HP MyFlight Sample Application").WpfEdit("passengerName").Set
```

- d. 在 **Set** 方法后面, 输入 "John Smith" 作为要针对 **passengerName** 对象输入 (Set) 的字符串。

您的步骤应如下所示:

```
WpfWindow("HP MyFlight Sample Application").WpfEdit("passengerName").Set  
"John Smith"
```

- e. 使用与上述相同的过程添加第二个步骤 (单击 “ORDER” 按钮) 的对象层次结构。

添加第二个步骤的对象层次结构后, 该语句应如下所示:

```
WpfWindow("HP MyFlight Sample Application").WpfButton("ORDER").
```

- f. 在 **WpfButton("ORDER")** 对象后面添加 **Click** 方法。现在您的语句应如下所示:

```
WpfWindow("HP MyFlight Sample Application").WpfButton("ORDER").Click
```

备注: **Click** 方法不需要任何参数, 因此不需要在方法名称后添加其他信息。

完成两个语句后, 编辑器应显示以下信息:

```
WpfWindow("HP MyFlight Sample Application").WpfEdit("passengerName").Set "John  
Smith"  
WpfWindow("HP MyFlight Sample Application").WpfButton("ORDER").Click
```

6. **删除多余的行。**

执行前面的步骤后，您向操作添加了另外两行（总共四个语句）。从操作删除最后两个语句，确保测试成功运行。

7. **保存测试。**

选择文件 > 保存。

课程 4:运行和分析 GUI 测试

在[课程 3:向测试添加步骤 \(第 38 页\)](#) 中, 您创建了多个操作并在每个操作中创建了多个测试步骤来测试航班预订应用程序。现在测试创建已经完成, 您可以运行该测试, 查看航班应用程序的表现如何。

在本课程中, 您将学习如何运行测试以及查看运行结果。

此课程包含以下内容:

- [练习 4a:运行测试](#) 62
- [练习 4b:导航运行结果](#) 64
- [练习 4c:分析运行结果](#) 65

练习 4a:运行测试

在[课程 3:向测试添加步骤 \(第 38 页\)](#) 中，您创建了运行航班预订应用程序以预订航班的基本测试。

在本练习中，您将学习如何运行您刚完成的测试。

1. 启动 UFT 打开 Book Flights 测试。

- a. 如果当前未打开 UFT，请将 UFT 打开，如[为所有测试创建解决方案 \(第 20 页\)](#) 中所述。确保已加载 WPF 插件。
- b. 选择“文件”>“打开”>“解决方案”。将打开“打开解决方案”对话框。
- c. 在“打开解决方案”对话框中，导航到位于 **C:\%HOMEPATH%\My Documents\Unified Functional Testing 的 Flight Reservation Application.ftsln** 文件，然后单击“打开”。
将在解决方案浏览器中打开 Flight Reservation Application 解决方案，并且 Book Flights 测试将在文档窗格的单独选项卡中打开。

2. 将 UFT 配置为在运行结果中保存所有图像。

执行测试运行时，UFT 允许您选择是否将所有图像保存到测试结果。

- a. 选择“工具”>“选项”>“GUI 测试”选项卡>“Screen capture”节点。将打开“屏幕捕获”选项窗格。
- b. 在“屏幕捕获”选项窗格中，选中“当发生以下情况，将捕获的静态图像保存到结果”复选框，然后从下拉菜单中选择“总是”。
- c. 单击确定关闭“选项”对话框。

3. 配置测试的录制和运行设置。

在某些情况下，您需要 UFT 在测试运行开始时为您打开应用程序。在这些情况下，您可以通过设置“录制和运行设置”实现此目的。

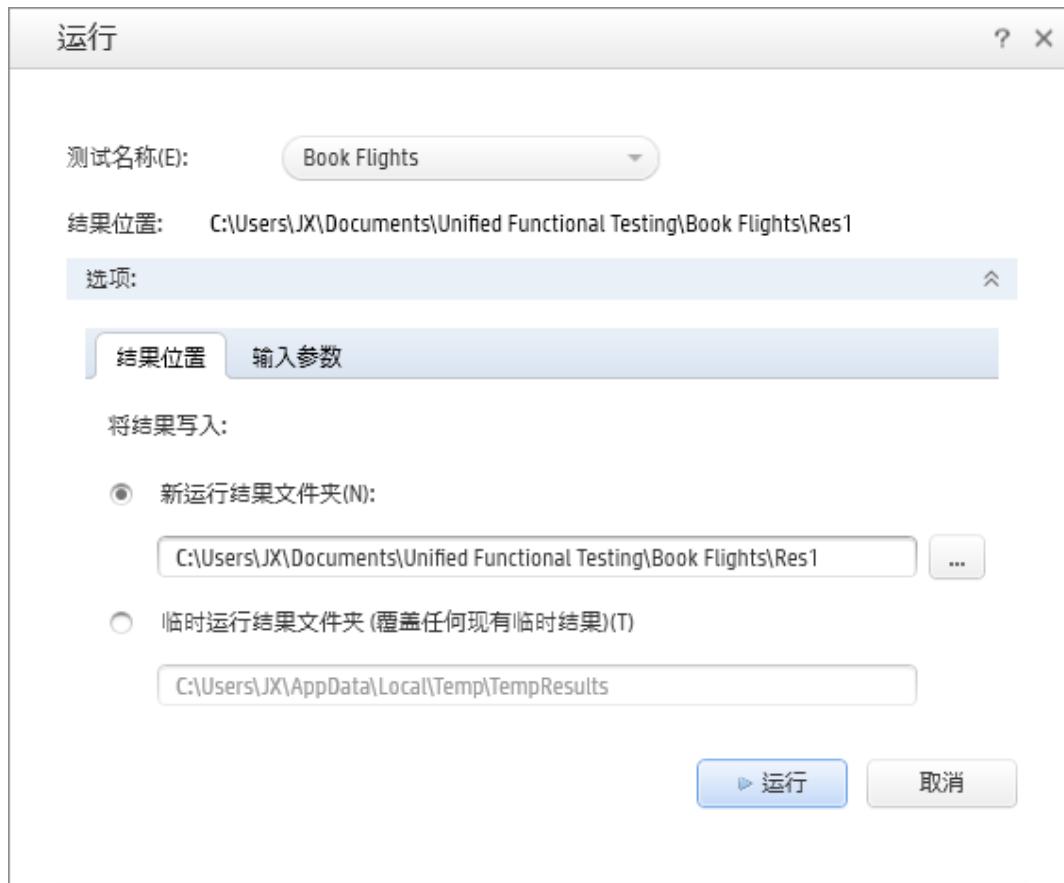
- a. 选择“录制”>“录制和运行设置”。将打开“录制和运行设置”对话框。
- b. 在“录制和运行设置”对话框的“Windows 应用程序”选项卡中，选择“仅在以下应用程序上录制和运行:”选项。
- c. 在“仅在以下应用程序上录制和运行:”选项下，选择“以下指定的应用程序”选项。
- d. 在“应用程序详细信息”区域中，单击“添加”按钮 。将打开“应用程序详细信息”对话框。
- e. 在“应用程序详细信息”对话框中，输入应用程序详细信息:

应用程序	C:\%HOMEPATH%\Unified Functional Testing\samples\Flights Application\FlightsGUI.exe
工作文件夹	C:\%HOMEPATH%\Unified Functional Testing\samples\Flights Application\

- f. 单击“确定”关闭“应用程序详细信息”对话框。
- g. 在“录制和运行设置”对话框中，单击“应用”，然后单击“确定”以启用设置并关闭对话框。

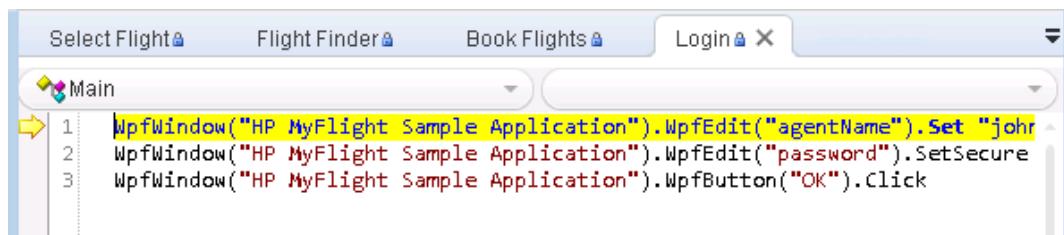
4. 开始运行测试。

- 在解决方案浏览器中，选择“Book Flights”测试节点。
- 在工具栏中，单击“运行”按钮 。将打开“运行测试”对话框。
- 在“运行测试”对话框中，单击“选项”栏展开运行测试选项区域，确保已选中“新建运行结果文件夹”选项。接受默认文件夹名称：



- 单击“运行”以关闭“运行”对话框并开始测试运行。

当 UFT 打开应用程序并开始运行测试时，请仔细观看。在应用程序中，您可以看到 UFT 执行您插入的每个步骤；关键字视图或编辑器中左空白处的黄色箭头和突出显示的行指示 UFT 正在运行的步骤：



如果出现任何错误，则转到错误消息中指示的测试位置，并验证是否按照相关练习的说明配置了步骤。

完成测试运行后，在文档窗格中，运行结果将在单独的选项卡中打开。请继续[练习 4b:导航运行结果 \(第 64 页\)](#)学习更多有关运行结果的内容。

练习 4b:导航运行结果

在[练习 4a:运行测试 \(第 62 页\)](#)中，您运行了创建的 Book Flights 测试。完成测试运行后，运行结果将自动显示此测试运行的结果。

备注: 默认情况下，运行结果显示在基于 HTML 的报告中。还可以选择将运行结果显示在“选项”对话框“运行会话”窗格的 Run Results Viewer 中（“工具”>“选项”>“常规”选项卡>“运行会话”节点）。本教程的此课程以基于 HTML 的报告为基础。

运行结果打开后，UFT 将显示以下信息：

The screenshot shows the HP UFT Run Results Viewer interface. At the top, there's a header bar with the title 'UFT 报告: Book Flights - Res1' and other navigation links like 'Login' and '起始页'. Below the header, the main area displays a summary for the 'Book Flights - Res1' run. It includes a green checkmark icon, the run name, and counts of errors (0) and warnings (0). Below this summary, there are two tabs: '错误列表' (Error List) and '测试流' (Test Flow). The '错误列表' tab is currently selected and shows a message '无错误/警告' (No errors/warnings). On the right side of the main area, there's a '详细信息' (Detailed Information) panel which is currently empty. At the bottom of the interface, there's a footer with language selection ('简体中文 | English'), the HP logo, and a 'UFT 报告' link. A status bar at the very bottom shows the number of errors (0), warnings (0), messages (0), and a search icon.

默认情况下，运行结果显示以下信息：

测试流	结果在树中的图形表示，按照在测试运行期间访问的操作和应用程序页面进行组织。运行期间执行的步骤在树中用图标表示，并且可以展开（箭头）以查看每个步骤。可以指示 UFT 在每次运行中使用不同数据集多次运行测试或操作。每次运行称为一个迭代，每个迭代都有编号。（您运行的测试只有一个迭代。）
错误列表	所有错误和警告的列表，以列表形式显示。
步骤摘要信息	高级别的结果概述报告，包括测试的常规信息、通过或失败的步骤以及每个测试步骤的详细信息等。
外部资源的链接	测试所使用的或测试运行中的外部资源的链接，包括： <ul style="list-style-type: none">• 数据表• 运行时视频• 应用程序日志

在此练习中，测试已运行成功，因为 UFT 能根据您添加的步骤来导航到航班预订应用程序。如果出现错误且测试未能成功运行，则错误将在运行结果中显示。在这种情况下，请返回并确保完全按照此教程中所述配置了步骤。

现在，您已了解了运行结果所显示的内容。请继续[练习 4c:分析运行结果 \(第 65 页\)](#) 学习有关运行结果的详细信息。

练习 4c:分析运行结果

在此练习中，您将检查在运行[练习 4a:运行测试 \(第 62 页\)](#) 中的测试时，UFT 执行的步骤。您可以查看每个步骤的应用程序窗口的快照。

1. 查看特定步骤的运行结果。

在“测试流”中的结果树下，找到“Flight Finder”节点，查看在航班预订应用程序的 Flight Finder 页面上执行的所有步骤。

在结果树中，选择“fromCity.Select”步骤：

The screenshot shows the UFT Results Tree interface. At the top, there's a header bar with a green checkmark icon, the title 'GUITest1 - Res8', and status indicators for errors (0) and warnings (0). Below the header, the main area has tabs for '错误列表' (Errors) and '测试流' (Test Flow), with '测试流' currently selected. A search bar is also present. The results tree displays a hierarchical structure of steps: 'Flight Finder' > 'agentName.Set' > 'password.SetSecure' > 'OK.Click' > 'fromCity.Select' (which is highlighted with a blue selection bar) > 'toCity.Select'. To the right of the tree, a '步骤详细信息' (Step Details) panel is open, showing information for the selected 'fromCity.Select' step, including its name, description ('Los Angeles'), execution time ('2015-09-18 11:35:43'), and details about the test object ('WpfComboBox: "fromCity"') and storage ('Local').

现在，运行结果显示以下信息：

- 测试流，且突出显示步骤
- 测试步骤摘要，显示已突出显示步骤的详细信息

2. 关闭运行结果。

在文档窗格中，关闭包含运行结果的选项卡。

现在您已经设置并运行了第一个测试，您可以继续学习增强测试的其他方式。选择以下课程之一以了解更多：

- [课程 5:参数化步骤和对象 \(第 66 页\)](#)
- [课程 6: 创建检查点和输出值 \(第 78 页\)](#)
- [课程 7:创建函数和函数库 \(第 108 页\)](#)
- [课程 8:在测试中使用洞察 \(第 117 页\)](#)

课程 5:参数化步骤和对象

在[课程 3:向测试添加步骤 \(第 38 页\)](#) 中, 您创建了测试步骤来检查在航班预订应用程序上执行的一系列步骤是否顺利运行。在[课程 4:运行和分析 GUI 测试 \(第 61 页\)](#) 中, 您运行了测试, 但该测试只使用单个数据集。但是, 当测试您的应用程序时, 您可能会要查看使用多个数据集执行的相同操作。

例如, 您可能要使用十个不同的数据集在您的应用程序上运行一个测试。您可以创建十个单独的测试, 每个均带有自己的数据集, 或者您可以将十个参数集添加到单个测试。如果添加参数, 则测试将运行十次, 每次使用一个不同的数据集。

在本课程中, 您会将参数添加到测试并使用多个数据集运行测试。

此课程包含以下内容:

· 参数化测试、操作和对象 - 概述	67
· 练习 5a:创建参数化测试	67
· 练习 5b:定义数据表参数	68
· 练习 5c:将参数值添加到数据表	71
· 练习 5d:运行参数化测试	73

参数化测试、操作和对象 - 概述

使用数据来参数化测试、操作或测试对象值时，可从多个位置提供数据源：

- **数据表:** 一个包含参数名称和值的 Excel 电子表格
- **环境变量:** 测试中具有固定值的变量集
- **随机数字:** 测试运行时由 UFT 生成的随机数字

其中最常用的是数据表参数。数据表是一个 Excel 电子表格，显示在 UFT 窗口底部的“数据”窗格中。

备注: 如果未显示“数据”窗格，请选择“查看”>“数据”或在工具栏中单击“数据”按钮 。

数据表中有两种不同类型的表：

全局数据表	全局数据表包含测试中所有操作使用或可用的数据参数和数据。将参数插入全局表后，测试中的所有操作以及操作中的所有步骤均可使用该参数。 测试将运行与全局数据表中的行数相同的迭代次数。因此，例如，如果有五行数据，则测试将运行五次迭代。
操作表	对于测试中的每个操作，UFT 将为该操作添加另一个表（名称与该操作相同）。数据参数和数据仅对该操作中的步骤可用。 如果使用一个操作表中的多行数据，UFT 将运行操作的次数与数据表中的行数相同（在一个测试迭代中）。

在学习此课程的过程中，您将仅使用数据表参数。有关其他类型的参数的详细信息，请参阅《HP Unified Functional Testing 用户指南》中有关参数化的部分

练习 5a: 创建参数化测试

在[练习 3b: 通过录制将步骤添加到 FlightFinder 操作 \(第 45 页\)](#) 中，您预订了从旧金山到悉尼的航班。在这些步骤中，Angeles 和 Sydney 值是常量值。这意味着 UFT 每次运行均使用 **Los Angeles** 和 **Sydney** 作为出发和到达城市。

在此练习中，您将新建测试，把出发和到达城市定义为参数，以便可以为每次测试运行使用不同的出发和到达城市。

1. **启动 UFT 打开 Book Flights 测试。**
 - a. 如[为所有测试创建解决方案 \(第 20 页\)](#) 中所述打开 UFT。确保已加载 WPF 插件。
 - b. 在起始页上的“最近使用的解决方案”区域，单击“Flight Reservation Application”。将打开 Flight Reservation Application 解决方案，其中包含您在[课程 1: 创建 GUI 测试和操作 \(第 24 页\)](#) 中创建的 **Book Flights** 测试。
2. **将 Book Flights 另存为 Book Flights Parameter。**
 - a. 在解决方案浏览器中，选择“Book Flights”节点。

- b. 选择文件 > 另存为。在“将测试另存为”对话框中，浏览到 C:\%HOMEPATH%\My Documents\Unified Functional Testing，并将测试另存为 Book Flights Parameter。
在解决方案浏览器中，Book Flights 测试将替换为新的 Book Flights Parameter 测试。Book Flights 测试仍单独保存在文件系统中。

3. 将 Book Flights 测试添加回解决方案。

如果所有测试均引用自相同的解决方案，则可以将它们同时打开。这使您能够在想要比较或编辑这两个测试时来回进行切换。一次仅能运行一个测试。

- a. 选择“文件”>“添加”>“现有测试”。
- b. 在“添加现有测试”对话框中，浏览到 C:\%HOMEPATH%\My Documents\Unified Functional Testing，并选择“Book Flights”测试。
- c. 单击“添加”将其添加到解决方案。

解决方案浏览器中将再次显示 Book Flights 测试。请注意，上面显示的是您刚才创建的 Book Flights Parameter 测试，因为测试按字母顺序列出。

创建参数化测试后，请继续[练习 5b:定义数据表参数 \(第 68 页\)](#)学习创建测试中的数据表参数。

练习 5b: 定义数据表参数

在本课程中，您将把出发和到达城市定义为参数，以便可以为每次测试运行使用不同的出发城市。

1. 启动 UFT 并在必要时打开 Book Flights Parameter 测试。

- a. 如[为所有测试创建解决方案 \(第 20 页\)](#)中所述打开 UFT。确保已加载 WPF 插件。
- b. 选择“文件”>“打开”>“解决方案”。将打开“打开解决方案”对话框
- c. 导航到位于 C:\%HOMEPATH%\My Documents\Unified Functional Testing 中的 Flight Reservation Application.ftsln 文件，并单击“打开”。
将打开“Flight Reservation Application”解决方案，其中包含您在[练习 5a: 创建参数化测试 \(第 67 页\)](#)中创建的 Book Flights Parameter 测试。
- d. 在解决方案浏览器中，选择“Book Flights Parameter”节点。

2. 确保“数据”窗格可见。

如果您在 UFT 窗口底部未看到“数据”窗格，请选择“查看”>“数据”。

3. 打开 Flight Finder 操作。

- a. 在画布中双击“Flight Finder”操作。在文档窗格中，Flight Finder 操作将在单独的选项卡中显示。
- b. 如需要，请选择“查看”>“关键字视图”。

4. 选择要参数化的文本。

在“关键字视图”中的“fromCity”行中，单击“值”单元格，然后单击参数化按钮 。

将显示参数列表:

测试/操作参数 (0)			DataTable (0)	Environment (21)	Random Number (0)
名称	值				
+添加新参数					

5. 设置参数化属性。

- 在参数列表中选择“数据表”选项卡。这使您能够将常量值 (**London**) 替换为参数。
请注意，未显示任何参数，因为您尚未创建数据表参数。
- 在参数列表的“数据表”选项卡中，单击“添加新参数”按钮。将打开“值配置选项”对话框：



- 在“值配置选项”对话框中，选择“参数”单选按钮。

- d. 确认从“参数”下拉菜单中选择了 **DataTable** 选项。这意味着参数的值将取自 UFT 的“数据”窗格。启用“名称”框，并显示“p_Item”：



- e. 删除“p_Item”参数，然后输入“fromCity”。

- f. 单击确定关闭对话框。

UFT 会将 **fromCity** 参数作为一个新列添加到“数据”窗格，并在该列的第一行中插入 Los Angeles (上一个常量值)。

Los Angeles 将是 UFT 在应用程序的测试运行期间使用的几个出发城市中的第一个。

项	操作	值	文档
HP MyFlight Sample Application			
fromCity	Select	DataTable("fromCity", dtGlobal...)	从 "fromCity" list 中选择 <the 'fromCity' Data Table column 的值>
toCity	Select	"Frankfurt"	从 "toCity" list 中选择 "Frankfurt" 项。
日	SetDate	"17-Sep-2015"	将 "日" calendar 中的日期或日期范围设置为 "17-Sep-2015"。
Class	Select	"Business"	从 "Class" list 中选择 "Business" 项。
numOfTickets	Select	"2"	从 "numOfTickets" list 中选择 "2" 项。
FIND FLIGHTS	Click		单击 "FIND FLIGHTS" button。
+新建步骤			

请留意“关键字视图”中步骤外观的变更。之前，该步骤显示为 fromCity Select Los Angeles。现在，单击“值”单元格时，将显示以下信息，这表明该值使用名称为 fromCity 的“数据”窗格参数进行了参数化：

```
DataTable("fromCity", dtGlobalS...)
```

6. 为 toCity 步骤添加数据表参数。

使用上一个步骤中描述的过程，为名为 toCity 的 toCity 对象添加数据表参数。

完成后，您的测试应如下所示:

项	操作	值	文档
HP MyFlight Sample Application			
fromCity	Select	DataTable("fromCity", dtGlobalSh... 从 "fromCity" list 中选择 <the 'fromCity' Data Table column 的值>	
toCity	Select	DataTable("toCity", dtGlobalSh... 从 "toCity" list 中选择 <the 'toCity' Data Table column 的值> 项...	
日	SetDate	"17-Sep-2015"	将 "日" calendar 中的日期或日期范围设置为 "17-Sep-2015"。
Class	Select	"Business"	从 "Class" list 中选择 "Business" 项。
numOfTickets	Select	"2"	从 "numOfTickets" list 中选择 "2" 项。
FIND FLIGHTS	Click		单击 "FIND FLIGHTS" button。
+新建步骤			

7. 保存测试。

选择“文件”>“保存”。

继续[练习 5c:将参数值添加到数据表 \(第 71 页\)](#) 学习如何使用测试运行所用的值填充数据表。

练习 5c:将参数值添加到数据表

在[练习 5b:定义数据表参数 \(第 68 页\)](#) 中您已了解，UFT 在“数据”窗格中显示参数值。在此练习中，您将把 (fromCity 对象的) 另一个出发城市添加到“数据”窗格，以便 UFT 可以使用此数据测试应用程序。

1. 启动 UFT 并在必要时打开 Book Flights Parameter 测试。

- 如[为所有测试创建解决方案 \(第 20 页\)](#) 中所述打开 UFT。确保已加载 WPF 插件。
 - 单击“打开”向下箭头 ，并选择“打开解决方案”。将打开“打开解决方案”对话框。
 - 导航到位于 C:\%HOMEPATH%\My Documents\Unified Functional Testing 中的 Flight Reservation Application.ftsln 文件，并单击“打开”。
- 将打开“Flight Reservation Application”解决方案，其中包含您在[练习 5a:创建参数化测试 \(第 67 页\)](#) 中创建的 Book Flights Parameter 测试。
- 在解决方案浏览器中，选择“Book Flights Parameter”节点。

2. 打开 Flight Finder 操作。

在解决方案浏览器中，双击“Flight Finder”操作。

“Flight Finder”操作将在文档窗格的单独选项卡中打开。

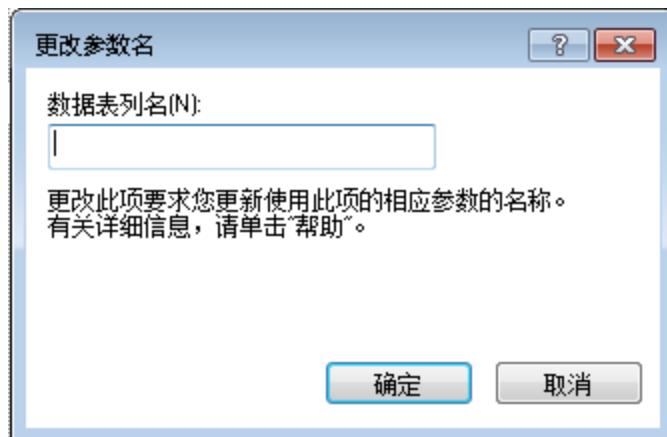
3. 在 fromCity 列中输入另一个城市。

在“数据”窗格中，为 fromCity 参数输入以下值:

行	值
2	Denver
3	Frankfurt
4	London

4. 为 toCity 对象创建数据表参数和值。

- a. 在“数据”窗格中，双击 B 列的标题行。将打开“更改参数名”对话框。



- b. 在“更改参数名”对话框中，输入 toCity 作为参数名并单击“确定”。

列标题(先前为 B) 将更新为新的参数名:

	fromCity	toCity	C
1	Los Angeles		
2			
3			
4			
5			

< > \ Global \ Login \ Flight Finder /

- c. 为 toCity 参数输入如下值:

行	值
1	Sydney
2	Los Angeles
3	London
4	Frankfurt

添加第二个参数及其值后，“数据”窗格应如下所示:

C1			
	fromCity	toCity	C
1	Los Angeles	Sydney	
2	Denver	Los Angeles	
3	Frankfurt	London	
4	London	Frankfurt	
5			

5. 参数化 toCity 步骤。

- a. 在 Flight Finder 操作的 toCity 行中，单击“值”单元格，然后单击参数化按钮 。将

- 打开“值配置选项”对话框。
- b. 在“值配置选项”对话框中，选择“参数”单选按钮。
 - c. 在“参数”类型下拉菜单中，选择“数据表”。
 - d. 在“数据表”区域的“位置”中，选择“全局表”。“名称”下拉菜单会更改为反映全局数据表的参数。
 - e. 在“名称”框中，选择 toCity 参数并单击“确定”。

在“关键字视图”中，将更新 toCity 对象的“值”单元格以显示参数化：



6. 保存测试。

在工具栏上，单击“保存”。

您已添加参数和值，并将测试步骤与这些值链接，您已准备好运行参数化测试。继续[练习 5d:运行参数化测试 \(第 73 页\)](#)

练习 5d:运行参数化测试

在[练习 5b:定义数据表参数 \(第 68 页\)](#) 和[练习 5c:将参数值添加到数据表 \(第 71 页\)](#) 中，您为 Flight Finder 操作中的“toCity”和“fromCity”对象创建了数据表参数。这样可将常量对象值替换为测试的数据表中不断变化的值。

但如果要在此次运行测试，则测试将仅使用全局数据表第一行中的数据运行一次。因为参数化的目的就是观察应用程序使用不同数据集时的运行状况，因此您需要指示 UFT 多次运行测试。

在本课程中，您将对 UFT 和测试进行配置，确保整个测试将使用测试的数据表中的数据运行多次。

1. 启动 UFT 并在必要时打开 Book Flights Parameter 测试。

- a. 如[为所有测试创建解决方案 \(第 20 页\)](#) 中所述打开 UFT。确保已加载 WPF 插件。
 - b. 单击“打开”向下箭头，并选择“打开解决方案”。将打开“打开解决方案”对话框。
 - c. 导航到位于 C:\%HOMEPATH%\My Documents\Unified Functional Testing 中的 Flight Reservation Application.ftsln 文件，并单击“打开”。
- 将打开“Flight Reservation Application”解决方案，其中包含您在[练习 5a:创建参数化测试 \(第 67 页\)](#) 中创建的 Book Flights Parameter 测试。
- d. 在解决方案浏览器中，单击“Book Flights”测试的节点。测试流画布将在文档窗格的单独选项卡中打开。

2. 更改“录制和运行设置”，以便航班预订应用程序不会自动打开。

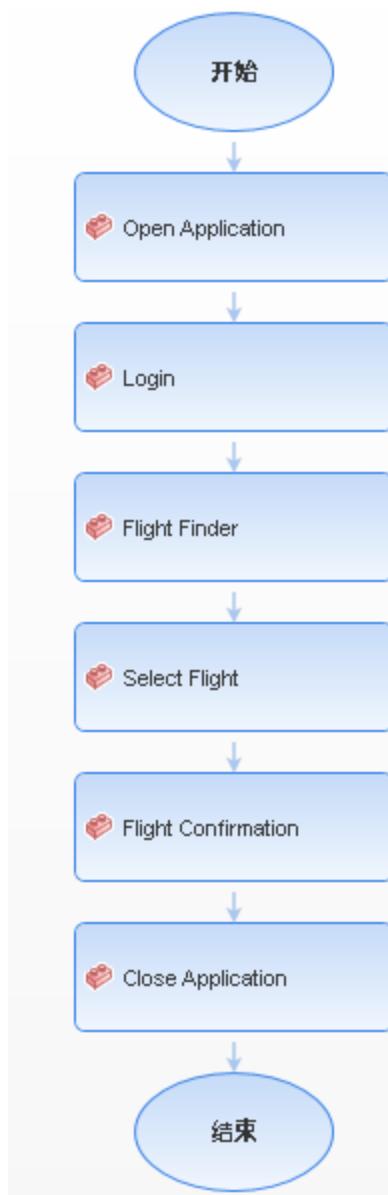
在 Book Flights 测试（创建此测试所依据的测试）中，您配置了运行和录制设置，测试运行开始时将自动打开航班预订应用程序。在此测试运行中，您希望将 UFT 打开应用程序作为测试步骤的一部分。

- a. 选择录制 > 录制和运行设置。将打开“录制和运行设置”对话框。
 - b. 在“录制和运行设置”对话框的“Windows 应用程序”选项卡中，选择“在任何打开的基于 Windows 的应用程序上录制并运行测试”选项，然后单击“应用”。
 - c. 单击确定关闭对话框。
3. **为应用程序的打开和关闭添加其他操作。**
- 运行参数化测试时，UFT 将多次运行整个测试，具体次数取决于数据表中的行数。但为了实现多次运行，必须添加步骤以打开和关闭应用程序，使 UFT 运行四个应用程序页面中每个页面的操作（“Login”、“Flight Finder”、“Select Flight”和“Flight Details” / “Confirmation”）。
- a. 在文档窗格中，选择“Book Flights”选项卡（包含测试流画布）。
 - b. 在工具栏上，单击“插入对新操作的调用”按钮 。将打开“插入对新操作的调用”对话框。
 - c. 在“插入对新操作的调用”对话框中，将新操作命名为 Open Application。所有其他设置和选项保留默认设置。
一个名为“Open Application”的新操作块将被添加到测试流末尾。
 - d. 在“Book Flights”选项卡（包含测试流画布）中，右键单击“Open Application”操作并选择“上移”。将“Open Application”操作块移动到“Flight Confirmation”操作上方。
 - e. 右键单击并选择“上移”，直到“Open Application”块成为测试中的第一个操作。

备注: 您还可以根据需要在测试流中拖放操作块。

- f. 在工具栏中，再次单击“插入对新操作的调用”按钮 。
- g. 在“插入对新操作的调用”对话框中，将新操作命名为 Close Application。所有其他设置和选项保留默认设置。

插入这两个新操作后，您的测试应如下所示:



4. 向打开和关闭应用程序添加语句。

您在创建 Book Flights 测试时，指示 UFT 使用测试的运行设置自动打开应用程序。在此测试中，您需要将应用程序的打开和关闭作为单独的步骤添加。为此，您将使用 **SystemUtil** 语句。

- 在解决方案浏览器中，双击“Open Application”操作节点。在文档窗格中，Open Application 操作将在单独的选项卡中打开。
- 选择“查看” > “编辑器” 打开编辑器。
- 将以下行粘贴到编辑器中：

```
SystemUtil.Run "C:\Program Files (x86)\HP\Unified Functional
```

Testing\samples\Flights Application\FlightsGUI.exe"

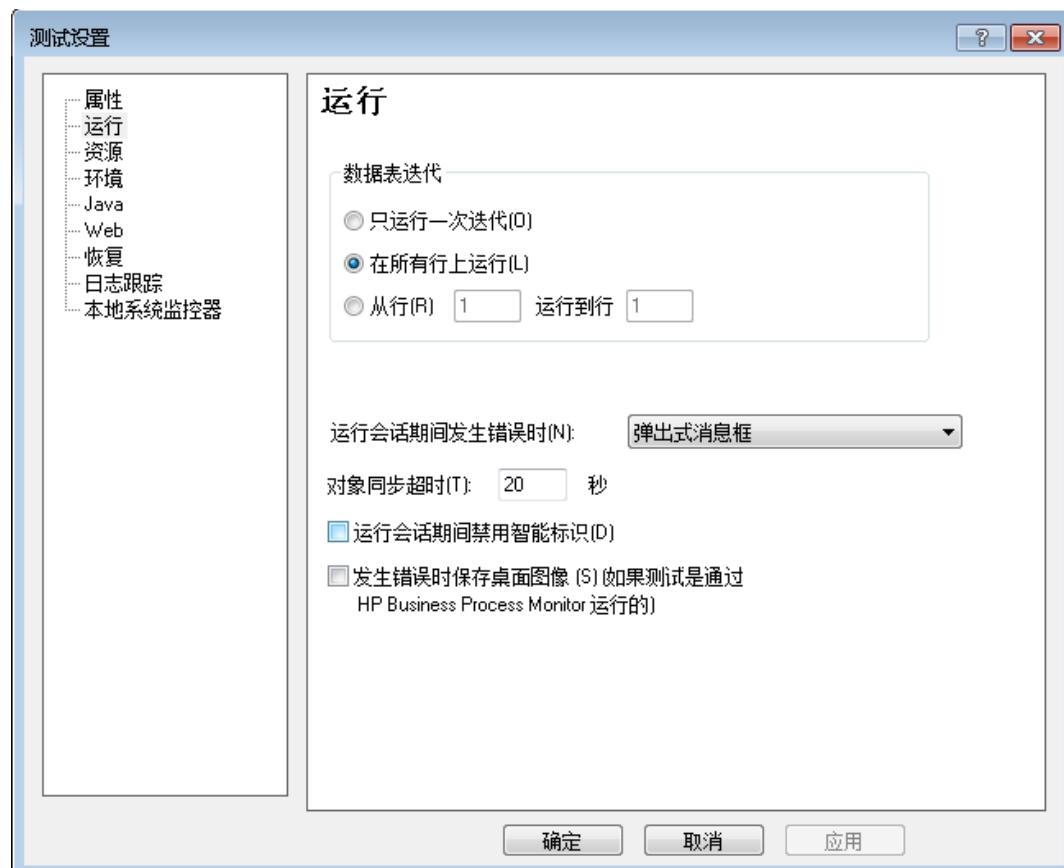
备注: 如果要从教程的 PDF 副本粘贴此行, 请确保对粘贴的文本进行编辑, 使此方法全部在一行上。

- d. 在解决方案浏览器中, 双击“Close Application”操作节点。在文档窗格中, Close Application 操作将在单独的选项卡中打开。
- e. 将以下行粘贴到编辑器中:

```
SystemUtil.CloseDescendentProcesses
```

5. 指示 UFT 对数据表中的每一行运行一个迭代。

- a. 选择“文件”>“设置”。将打开“设置”对话框。
- b. 在“设置”对话框中, 选择“运行”节点。
- c. 在“数据表迭代”部分中, 选择“在所有行上运行”选项。这能确保 UFT 对全局数据表中的每一行运行测试迭代。



现在, 当您运行测试时, UFT 将运行测试的多次迭代, 对应于全局数据表中的四行。

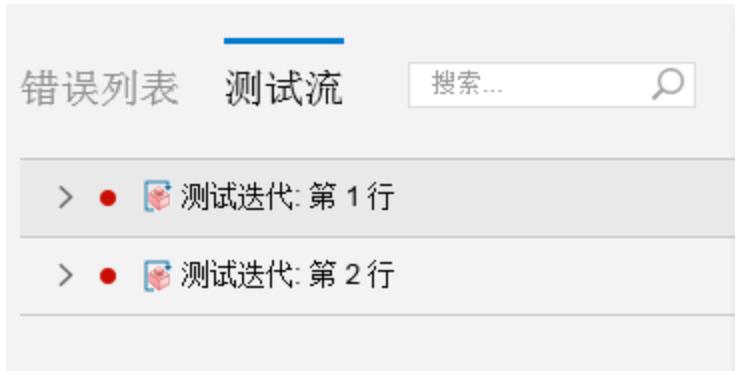
6. 运行 Book Flights Parameter 测试。

- a. 单击“运行”按钮 。将打开“运行”对话框。
- b. 在“运行”对话框的“结果位置”选项卡中，选择“新建运行结果文件夹”并接受默认文件夹名称。
- c. 单击“确定”。测试运行完成后，将打开运行结果。

7. 分析运行结果。

在 Run Results Viewer 中，右键单击结果树中顶部的节点并选择“全部展开”。

请注意，结果针对每个测试迭代显示五个不同的节点。这与全局数据表中的多行对应：



如果在 Flight Finder Summary 节点下搜索直至看到“fromCity.Select”或“toCity.Select”步骤，您将注意到步骤详细信息已修改，以与数据表中的值匹配。

8. 关闭运行结果。

在文档窗格中，关闭包含运行结果的选项卡。

现在您已经了解了如何使用数据参数化测试，请继续[课程 6: 创建检查点和输出值 \(第 78 页\)](#) 学习如何使用检查点和输出值增强测试。

课程 6: 创建检查点和输出值

在[课程 4:运行和分析 GUI 测试 \(第 61 页\)](#) 中, 您运行了在之前的课程中创建的测试, 以检查在航班预订应用程序上执行的一系列步骤是否顺利且正确地运行。

创建基本测试步骤后, 其中可进行的一项增强是为测试和测试步骤添加检查点和输出值。检查点可验证测试正在运行时应用程序中是否显示了预期的信息。输出值将某个值导出到测试中的其他位置以用作参数。

在本课程中, 您将插入检查点, 并使用函数来检查航班预订应用程序检查点中一些对象的有效性。

此课程包含以下内容:

· 了解检查点和输出值类型	79
· 练习 6a: 创建检查点测试	80
· 练习 6b: 检查对象值	81
· 练习 6c: 检查表值	84
· 练习 6d: 检查文本值	89
· 练习 6e: 在对象存储库中管理检查点	95
· 练习 6f: 使用检查点运行和分析测试	96
· 练习 6g: 创建输出值测试	99
· 练习 6h: 添加输出值步骤	100

了解检查点和输出值类型

在 UFT 中，您可以插入“检查点”查看应用程序是否运行正常。这些检查点将在整个测试流中作为独立测试步骤运行。使用“输出值”获取由特定步骤或对象生成的值，并将此值传递到其他步骤。

检查点

您可以使用检查点验证各种不同的应用程序对象：

对象类型	检查点描述	用法示例
标准	检查对象的属性值。	检查是否选择了单选按钮。
图像	检查图像的属性值。 通过选择“标准检查点”选项并再选择图像对象来检查图像。	检查图像源文件是否正确。
表	检查表中的信息。 通过选择“标准检查点”选项并再选择表对象来检查表。	检查表单元格中的值是否正确。
页面	检查网页的特征。 通过选择“标准检查点”选项并再选择浏览器中的网页来检查页面。	检查网页加载需要的时间或网页是否包含断开的链接。
文本	检查文本字符串是否显示在应用程序的适当位置。	检查预期文本字符串是否显示在测试对象中的预期位置。
文本区域	检查文本字符串是否显示在基于 Windows 的应用程序中的定义区域中。	检查对话框区域是否包含在应用程序的另一个部分中输入的文本。
位图	检查以位图方式捕获之后的应用程序的区域。	检查网页（或其任何部分）是否按预期显示。
数据库	检查应用程序或网站访问的数据库内容。	检查数据库查询中的值是否正确。
可访问性	标识要检查是否符合第 508 条的网站区域。	检查网页上的图像是否包括 W3C Web 内容可访问性规则所要求的 ALT 属性。
文件内容	检查在运行会话期间生成或访问过的文档中的文本。	检查动态生成的 PDF 中的标题是否正确显示区域性公司总部联系信息。
XML	检查 XML 文档的数据内容。	检查元素的内容，以确保其标记、属性和值未更改。 备注： XML 文件检查点用于检查指定 XML 文件；XML 应用程序检查点用于检查网页中的 XML 文档。

输出值

您可以使用各种不同类型的输出值：

对象类型	描述	示例

标准	从应用程序大部分对象中获取值并存储值。	获取编辑字段的字符串输出。
文件内容	获取选定文件或部分选定文件的输出。	获取 HTML 页面的输出。
表	获取表对象单元格或选定单元格的输出。	获取表对象中第 1 行第 1 列单元格的输出。
文本/文本区域	获取应用程序中某对象或某区域的文本输出。	获取错误消息的文本输出。
数据库	获取数据库单元格或选定数据库单元格的输出。	获取应用程序中某对象访问的数据库的输出。
XML	获取 XML 文档中包含的元素的输出。	获取定义产品价格的 XML 中 <价格> 属性的输出。

在关键字视图或编辑器中编辑步骤或进行录制时，您可以添加大部分检查点和输出值。下面的练习解释了如何创建某些上述检查点。

UFT 创建检查点或输出值时，它将依据检查点或输出值内的信息（例如，检查的值）分配名称。即使您修改所依据的信息，检查点或输出值名称也将保持不变。在查找关键字视图中显示的检查点或输出值时，请记住这点。但是，请注意，UFT 可能会缩短关键字视图中显示的名称。

有关检查点和输出值的其他详细信息，请参阅《HP Unified Functional Testing 用户指南》。

请继续[练习 6a: 创建检查点测试 \(第 80 页\)](#) 学习创建使用检查点的测试。

如果要添加输出值，请继续[练习 6g: 创建输出值测试 \(第 99 页\)](#)。

练习 6a: 创建检查点测试

在本练习中，您会将在[课程 4: 运行和分析 GUI 测试 \(第 61 页\)](#) 中运行的 Book Flights 测试另存为新测试，以创建检查点。

备注: 检查点无需在独立测试中进行管理。现在仅需要为此教程创建一个新测试。在日常的工作流中，您可以将检查点添加到任何测试。

1. 启动 UFT 打开 Book Flights 测试。

- 如果 UFT 当前未打开，请按[为所有测试创建解决方案 \(第 20 页\)](#) 中所述将其打开。确保已加载 WPF 插件。
- 单击“打开”按钮向下箭头 ，并选择“打开解决方案”。将打开“打开解决方案”对话框。
- 导航到位于 C:\%HOMEPATH%\My Documents\Unified Functional Testing 中的 Flight Reservation Application.ftsln 文件，并单击“打开”。

将打开 Flight Reservation Application 解决方案，其中包含您在[课程 1: 创建 GUI 测试和操作 \(第 24 页\)](#) 中创建的 Book Flights 测试。

2. 将测试另存为 Book Flights Checkpoint。

- 在解决方案浏览器中，右键单击“Book Flights”测试节点，并选择“另存为”。

- b. 在“将测试另存为”对话框中，浏览到 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 目录，并将测试另存为 Book Flight Checkpoint。
在解决方案浏览器中，将 Book Flights 测试替换为“Book Flights Checkpoint”测试。Book Flights 测试仍单独保存在文件系统中。

3. 将 Book Flights 测试添加回解决方案。

如果 Book Flights 和 Book Flights Checkpoint 测试包含在同一个解决方案中，则可以同时打开。这使您能够在想要比较或编辑测试时来回进行切换。

备注: 一次仅能运行一个测试。

- a. 选择“文件”>“添加”>“现有测试”。
- b. 导航到位于 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 中的 Book Flights 测试，并单击“添加”。
再次将 Book Flights 测试添加到解决方案浏览器中。请注意这些测试在解决方案浏览器中是按字母顺序排列的。
解决方案会自动保存。

请继续[练习 6b:检查对象值 \(第 81 页\)](#) 学习如何插入检查点。

练习 6b:检查对象值

在本练习中，您会将标准检查点添加到在[练习 6a:创建检查点测试 \(第 80 页\)](#) 中创建的测试。此检查点会验证“Flight Details”窗口中为“Passenger Name”字段输入的值。

备注: 航班预订应用程序必须在“Flight Details”页面打开，然后您才能插入检查点。

1. 启动 UFT 并打开“Book Flights Checkpoint”测试。

- a. 如需要，请如[为所有测试创建解决方案 \(第 20 页\)](#) 中所述打开 UFT。确保已加载 WPF 插件。
- b. 在起始页上的“最近使用的解决方案”区域，单击“Flight Reservation Application”。
将打开“Flight Reservation Application”解决方案，其中包含您在[练习 6a:创建检查点测试 \(第 80 页\)](#) 中创建的 Book Flights Parameter 测试。
- c. 在解决方案浏览器中，双击“Book Flights Checkpoint”节点。
在文档窗格中，Book Flights Checkpoint 测试将在单独的选项卡中打开。

2. 显示要添加检查点的操作。

为了添加检查“Passenger Name”编辑框属性值的检查点，在测试自动输入乘客姓名后，必须将该姓名添加到测试的相应操作中。

在画布中，双击“Flight Confirmation”操作，以将其打开。

3. 打开航班预订应用程序的 Flight Details 页面。

- a. 打开航班预订应用程序，如[浏览航班预订应用程序 \(第 18 页\)](#) 中所述。

- b. 输入登录信息:
 - **用户名:** john
 - **密码:** hp
- c. 单击 **OK**。将打开 Flight Finder 页面。
- d. 输入航班搜索详细信息:
 - **Departure City:** Los Angeles
 - **Arrival City:** Sydney
 - **Date:** 明天的日期
 - **Class:** Business
 - **Tickets:** 2
- e. 单击“Find Flights”按钮。将打开 Select Flight 页面。
- f. 在“Select Flight”页面中，选择第一行，并单击“Select Flight”。将打开“Flight Details”页面。

4. 创建标准检查点。

- a. 如果显示编辑器，则单击“关键字视图”按钮  以显示关键字视图。
- b. 在“关键字视图”中，通过单击网格的右空白处选择“passengerName”行。

备注: 不要单击“项”列 - 此操作将仅选择对象。添加检查点需要选择整个步骤。

- c. 选择“设计”>“检查点”>“标准检查点”。将打开“检查点属性”对话框：



此对话框将显示“passengerName”对象属性：

- “名称”是在应用程序中定义的对象名称。在此例中，名称为“passengerName”。
- “类”是对象的类型。在此例中，类型为“WpfEdit”，意思是此对象类型是编辑框。
- **类型列中的 ABC 图标**指示属性值是常量。

如果插入检查点，UFT 建议对每个对象类使用默认属性检查：

属性	值	说明
enabled	True	将检查对象当前是否已启用。
isreadonly	False	将检查是否可向编辑框输入信息。当前，对象设置为允许输入文本字符串。
text	未指定默认值	将检查在对象中输入的文本。当前没有值。您需要输入值应该与在此操作第一步中为“passengerName”编辑框指定的值相同。

- d. 在“检查点属性”对话框的**名称**框中，输入 CheckName 作为新检查点的名称。
- e. 在对象属性区域中向下滚动，并选择包含属性名为 **text** 的行。此行将变为黄色，表示您已选中此行。
- f. 在该“text”属性行中，单击“值”列。

- g. 在对象属性网格下方的“配置值”区域中，单击“常量”单选按钮。
- h. 在“常量”值编辑框中，输入 John Smith。(这是您在此操作第一步的“Passenger Name”框中输入的值名称。)请注意，对象属性网格也会更新为此值。
- i. 在“检查点属性”对话框底部的插入语句区域中，选择“当前步骤之后”。这将在**passengerName Set** 步骤之后插入检查点。
- j. 接受其余的默认设置，并单击确定。

UFT 将在测试中选定步骤的下方添加一个标准检查点步骤:

项	操作	值	文档
HP MyFlight Sample Application_2			
passengerName	Set	"John Smith"	输入 "John Smith" (在 "passengerName" edit box 中)。
passengerName	Check	CheckPoint("passengerName")	检查 "passengerName" edit box 中的文本与预期文本是否匹配。
ORDER	Click		单击 "ORDER" button。
Order 107 completed	Click	34,4	单击 "Order 107 completed" object。
NEW SEARCH	Click		单击 "NEW SEARCH" button。
+ 新建步骤			

5. 保存测试。

在工具栏上，单击“保存” 。

使用此过程，您可以插入许多不同类型的检查点。请继续[练习 6c:检查表值 \(第 84 页\)](#) 学习如何在应用程序中检查表对象。

练习 6c:检查表值

在[练习 6b:检查对象值 \(第 81 页\)](#) 中，您为应用程序中的对象添加了检查点。在本练习中，您会将表检查点添加到测试。表检查点将检查“Select Flights”页面上航班网格中的值。

1. 启动 UFT 并打开“Book Flights Checkpoint”测试。

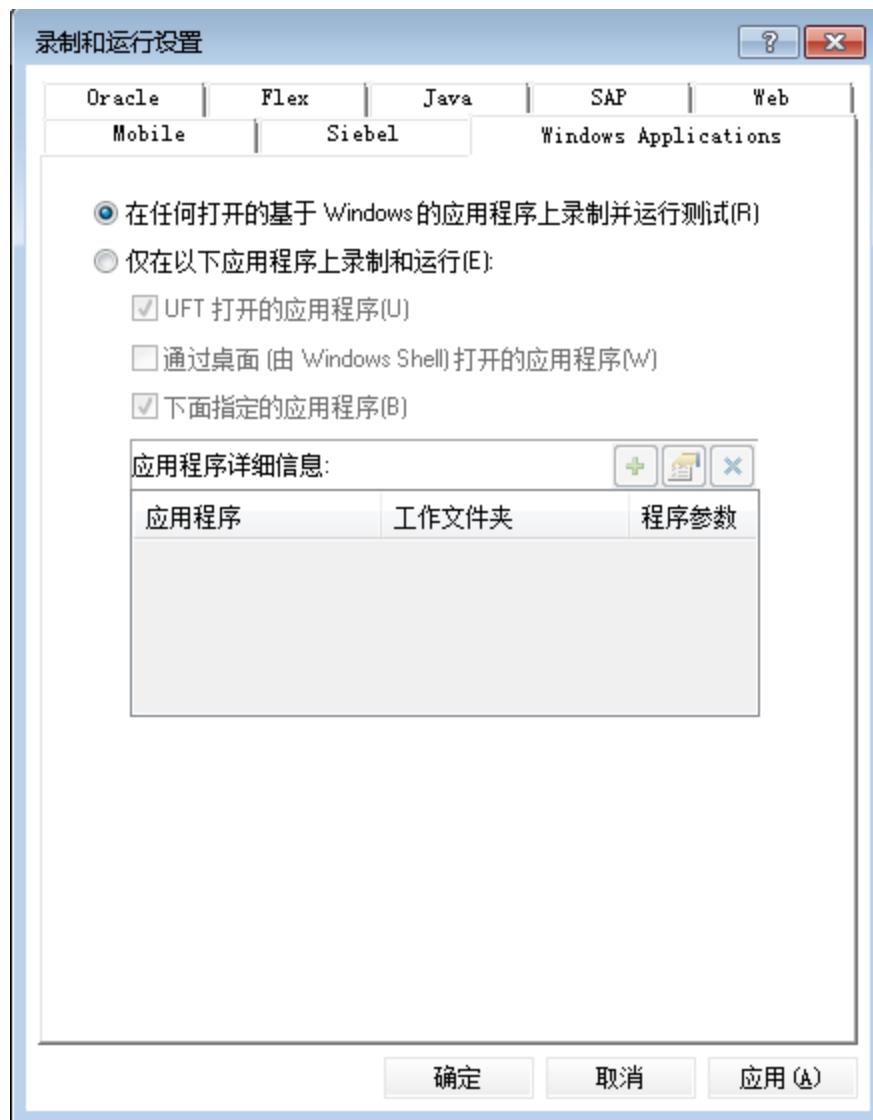
- a. 如需要，请如[为所有测试创建解决方案 \(第 20 页\)](#) 中所述打开 UFT。确保已加载 WPF 插件。
- b. 单击“打开”按钮向下箭头 ，然后选择“打开解决方案”。将打开“打开解决方案”对话框。
- c. 在“打开解决方案”对话框中，导航到位于 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 的 **Flight Reservation Application.ftsln** 文件，然后单击“打开”。将打开“Flight Reservation Application”解决方案，其中包含您在[练习 6a:创建检查点测试 \(第 80 页\)](#) 中创建的“Book Flights Parameter”测试。
- d. 在解决方案浏览器中，双击“Book Flights Checkpoint”节点。

2. 查找要添加表检查点的步骤。

- a. 如果 Select Flight 操作尚未打开，则在解决方案浏览器中，双击“Select Flight”操作节点。该操作将在文档窗格的单独选项卡中显示。
- b. 如果显示编辑器，则选择“查看”>“关键字视图”以显示关键字视图。
- c. 在“关键字视图”中，选择“flightsDataGrid”步骤（即选择预订航班的步骤）。

3. 打开航班预订应用程序的 Select Flight 页面。

- a. 打开航班预订应用程序，如[浏览航班预订应用程序 \(第 18 页\)](#) 中所述。
 - b. 输入登录信息:
 - **用户名:** john
 - **密码:** hp
 - c. 单击 **OK**。将打开 Flight Finder 页面。
 - d. 输入航班搜索详细信息:
 - **Departure City:** Los Angeles
 - **Arrival City:** Sydney
 - **Date:** 明天的日期
 - **Class:** Business
 - **Tickets:** 2
 - e. 单击“Find Flights”按钮。将打开 Select Flight 页面。
4. **将 UFT 配置为在打开的应用程序页面上进行录制。**
- a. 在 UFT 中，选择“录制”>“录制和运行设置”。将打开“录制和运行设置”对话框：

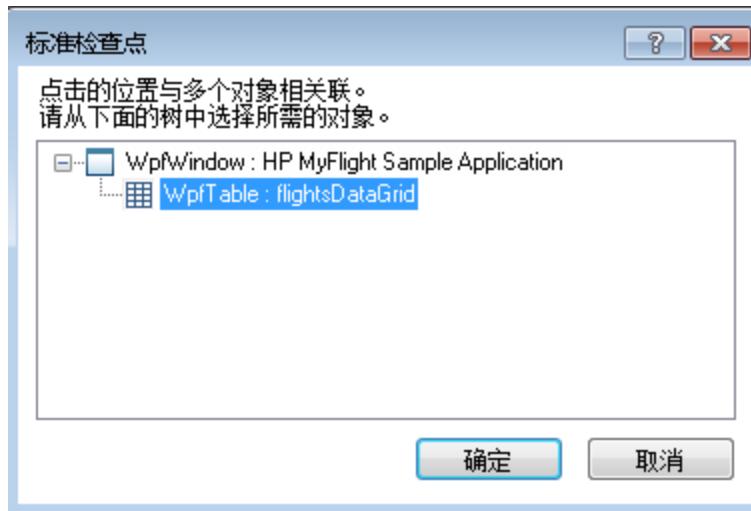


- b. 在“Windows 应用程序”选项卡中，选择“在任何打开的基于 Windows 的应用程序上录制并运行测试”选项。
- c. 单击确定关闭对话框。

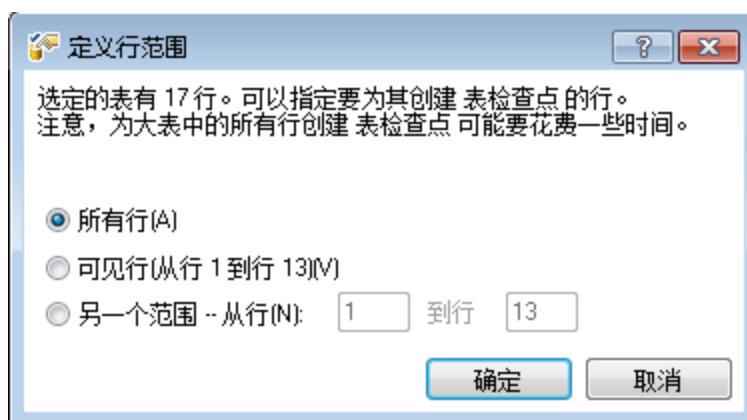
5. 创建表检查点。

- a. 在工具栏上，单击“录制”按钮 。UFT 开始录制会话，且 UFT 窗口将隐藏。
- b. 在“录制”工具栏上单击“插入检查点或输出值”下拉箭头 ，并选择“标准检查点”。鼠标指针将变成指向手。

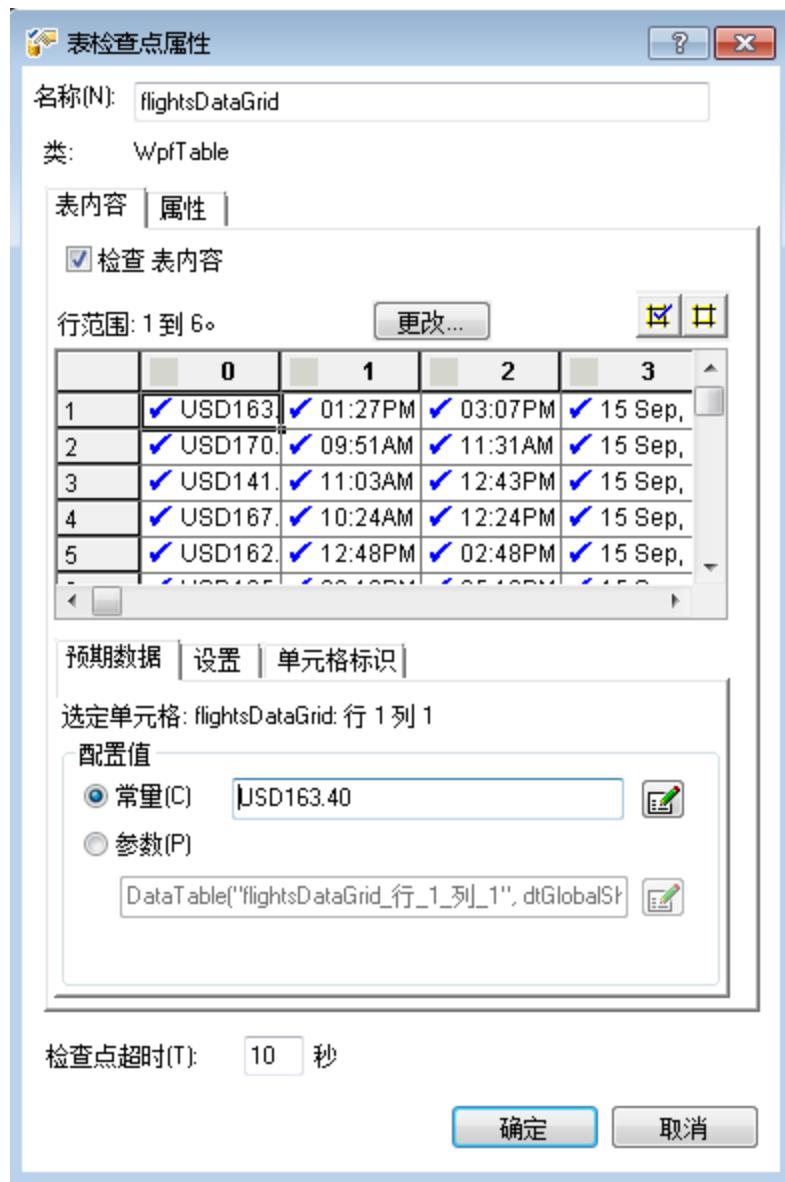
- c. 在航班预订应用程序中，单击航班表。将打开“标准检查点”对象选择对话框：



- d. 在“标准检查点”选择对话框中，选择“WpfTable:flightsDataGrid”对象并单击“确定”。将打开“定义行范围”对话框：



- e. 在“定义行范围”对话框中，选择“所有行”单选按钮，并单击“确定”。将打开“表检查点属性”对话框：



注意，默认情况下，复选标记显示在所有单元格中。可以双击单元格进行选择或清除对单元格的选择，或双击行或列标题进行选择或清除对行或列中所有单元格的选择。

- f. 在“表检查点属性”对话框的“名称”框中，输入 CheckCost 作为新检查点名称。
g. 在网格中，双击每个列标题，清除复选标记。

备注: 您需要向右滚动网格才能查看所有表对象的列。

- h. 在网格中，双击第 1 行，第 0 列选择此单元格。(UFT 仅检查包含复选标记的那些单元格。)

	0	1	2	3
1	✓ USD163	✓ 01:27PM	✓ 03:07PM	✓ 15 Sep,
2	✓ USD170	✓ 09:51AM	✓ 11:31AM	✓ 15 Sep,
3	✓ USD141	✓ 11:03AM	✓ 12:43PM	✓ 15 Sep,
4	✓ USD167	✓ 10:24AM	✓ 12:24PM	✓ 15 Sep,
5	✓ USD162	✓ 12:48PM	✓ 02:48PM	✓ 15 Sep,

备注: 表中显示的是日期相关的数据。如果您在某天创建此检查点，而在另一天运行此测试，则需要使用上述步骤更新此检查点以确保检查点通过。

- i. 滚动显示行和列，以确保仅选中第 1 行第 0 列中的单元格。如果选中任何其他单元格，则在上面双击以删除检查。
- j. 接受其余的默认设置，并单击确定。

6. 停止录制会话。

在“录制”工具栏上，单击“停止” 。

在定义表对象的检查点属性后，UFT 将表检查点步骤添加到测试。它将作为 flightsDataGrid 对象步骤下的新步骤显示在关键字视图中：

项	操作	值	文档
HP MyFlight Sample Application	Check	CheckPoint("HP MyFlight Sample Application")	检查 "HP MyFlight Sample Application" window
flightsDataGrid	SelectCell	0,1	选择行 0、列 1(在 "flightsDataGrid" table 中)
flightsDataGrid	Check	CheckPoint("flightsDataGrid_2")	Check whether the content of specified cells in the
SELECT FLIGHT	Click		单击 "SELECT FLIGHT" button。
+ 新建步骤			

7. 保存测试。

在工具栏上，单击“保存” 。

现在，您已为表对象添加了检查点，请继续[练习 6d:检查文本值 \(第 89 页\)](#) 学习添加检查点。

练习 6d:检查文本值

在之前的练习中，您将检查点添加到了常规测试对象和表对象中。在此对象中，您将向测试添加文本检查点，检查显示在对象内部订单进程结尾处的文本。

1. 启动 UFT 并打开“Book Flights Checkpoint”测试。

- a. 如需要，请如[为所有测试创建解决方案 \(第 20 页\)](#) 中所述打开 UFT。确保已加载 WPF 插件。
- b. 在起始页上的“最近使用的解决方案”区域中，单击“Flight Reservation Application”解决方案。

将打开“Flight Reservation Application”解决方案，其中包含您在[练习 6a:创建检查点测试 \(第 80 页\)](#) 中创建的“Book Flights Parameter”测试。

- c. 在解决方案浏览器中，双击“Book Flights Checkpoint”节点。
在文档窗格中，Book Flights Checkpoint 测试将在单独的选项卡中打开。

2. 查找要添加文本检查点的步骤。

- a. 在解决方案浏览器中，双击“Flight Confirmation”操作节点。“Flight Confirmation”操作将在文档窗格的单独选项卡中打开。
- b. 如果编辑器打开，则单击“关键字视图”按钮  以显示关键字视图。
- c. 在“关键字视图”中，突出显示“progBar”步骤（如果完全展开所有步骤，则在倒数第二行）。

3. 打开航班预订应用程序的 Flight Details 页面。

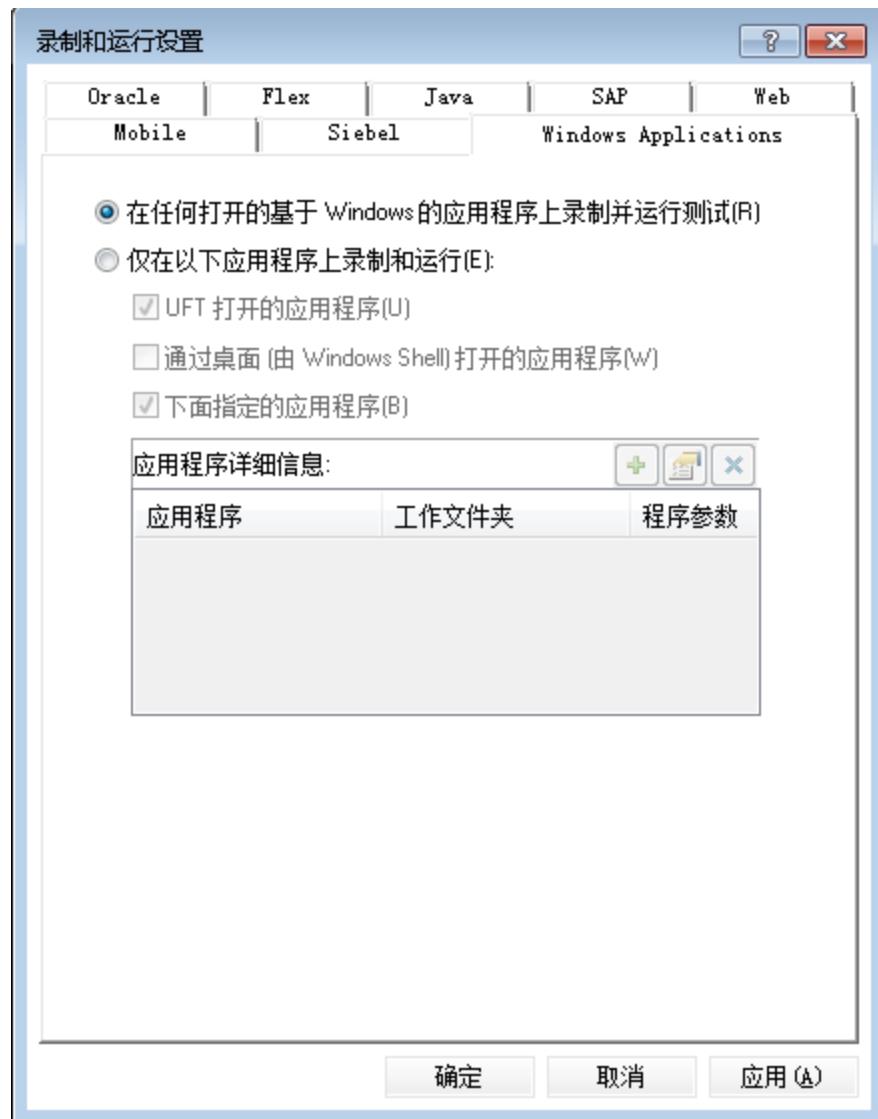
- a. 打开航班预订应用程序，如[浏览航班预订应用程序（第 18 页）](#)中所述。
- b. 输入登录信息：
 - **用户名:** john
 - **密码:** hp
- c. 单击**OK**。将打开 Flight Finder 页面。
- d. 输入航班搜索详细信息：
 - **Departure City:** Los Angeles
 - **Arrival City:** Sydney
 - **Date:** 明天的日期
 - **Class:** Business
 - **Tickets:** 2
- e. 单击“Find Flights”按钮。将打开 Select Flight 页面。
- f. 在“Select Flight”页面中，选择第一行，并单击“Select Flight”。将打开“Flight Details”页面。
- g. 在“Flight Details”页面的“Passenger Name”框中，输入 John Smith 并单击“ORDER”。

窗口中部将显示对话框，提示订单已完成。不再对应用程序执行其他操作。

4. 将 UFT 配置为在打开的应用程序页面上进行录制。

- a. 在 UFT 中，选择“录制”>“录制和运行设置”。将打开“录制和运行设置”对话框。

- b. 在“Windows 应用程序”选项卡中，确认已选中“在任何打开的基于 Windows 的应用程序上录制并运行测试”：

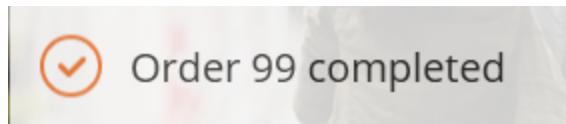


- c. 单击确定关闭对话框。

5. 创建文本检查点。

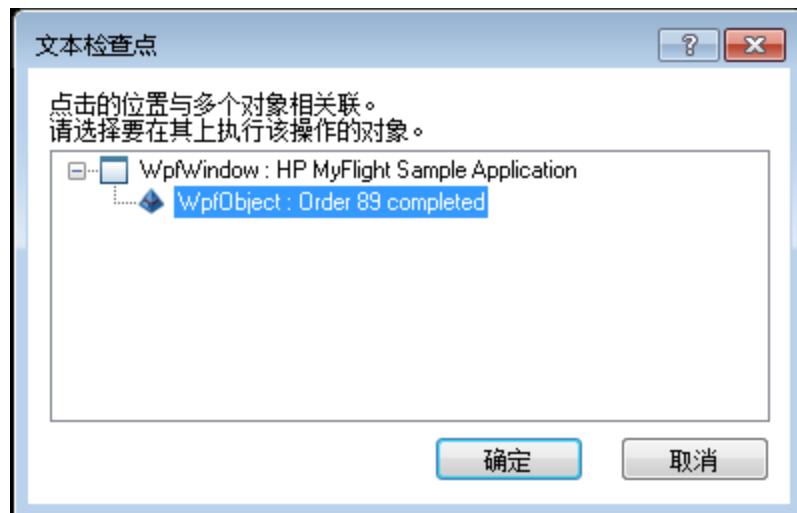
- 在工具栏上，单击“录制”按钮 。UFT 窗口将隐藏，“录制”工具栏将在窗口顶部打开。
- 在“录制”工具栏上单击“插入检查点或输出值”按钮 ，并选择“文本检查点”。鼠标指针将变成指向手。

- c. 在航班预订应用程序中的“Flight Details”窗口中，单击“Flight Details”窗口中部的“Order # Completed”图形：



备注: 订单号可能会基于您之前是否已运行航班预订应用程序而有所不同。

将打开“文本检查点”对象选择对话框：

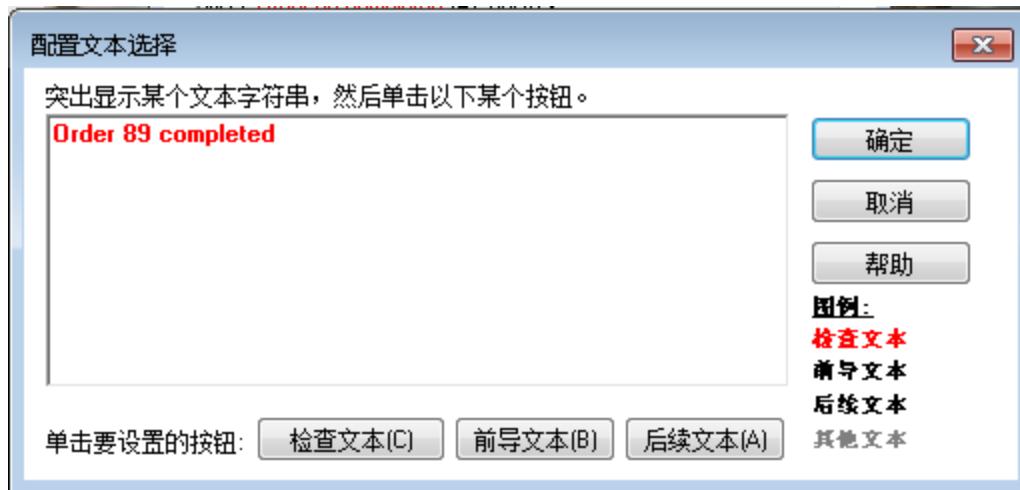


- d. 在“文本检查点”对象选择对话框中，选择“WpfObject: Order # Completed”对象，并单击“确定”。将打开“文本检查点属性”对话框：

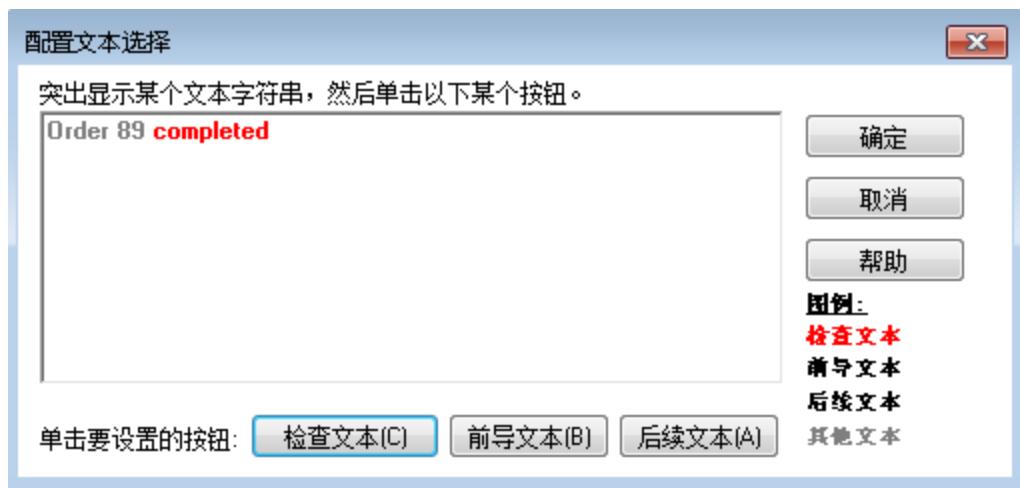


- e. 在“文本检查点属性”对话框的“名称”框中，输入 CheckOrderCompletedText 作为新检查点的名称。

- f. 在“检查点摘要”区域下，单击“配置”按钮。将打开“配置文本选择”对话框：

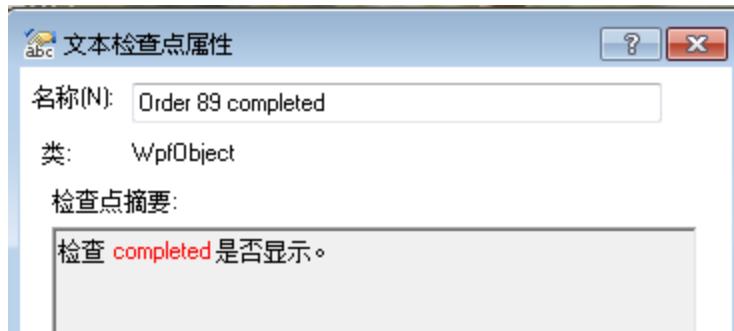


- g. 在“配置文本选择”对话框中，突出显示字符串“已完成”，并单击“检查文本”。“Order #”文本字符串将从红色变为灰色：



- h. 单击“确定”关闭“配置文本选择”对话框。

在“文本检查点属性”对话框中，“检查点摘要”区域将进行更新以反映您的选择：



- i. 接受其余的默认设置，并单击“确定”。

6. 停止录制会话。

在“录制”工具栏上，单击“停止” 停止录制。

UFT 将带有文本检查点的步骤添加到包含 **progBar** 对象的步骤下方。它在关键字视图中显示为对 **Order # Completed** 对象执行的检查点操作:

 progBar	WaitProperty	"value","100"	等到 "progBar" progress bar 的 "value" 属性值在对...
◆ Order 87 completed	Check	CheckPoint("Order 90 completed")	检查 "Order 87 completed" object 中的文本与预期...
□ NEW SEARCH	Click		单击 "NEW SEARCH" button。
+ 新建步骤			

在编辑器中，语句如下所示:

```
WpfWindow("Book Flights").WpfObject("Order 89 completed").Check CheckPoint
("CheckOrderCompletedText")
```

7. 保存测试。

单击“保存”。

现在，您已添加了几个不同类型的检查点。请继续[练习 6e:在对象存储库中管理检查点 \(第 95 页\)](#) 学习有关检查点管理的详细信息。

练习 6e:在对象存储库中管理检查点

在之前的练习中，您在操作中添加了许多不同类型的检查点。除了使用和查看特定操作上下文中的检查点外，您还可以查看对象存储库中的检查点，并修改其属性。

通过在对象存储库中对这些检查点进行修改，您可以在测试中的多个位置上使用同一检查点。例如，如果要验证组织徽标是否在应用程序的每个页面上显示，则可以创建一个检查点，并将其插入到测试中的不同操作或位置中。

在此练习中，您将不会重用检查点。

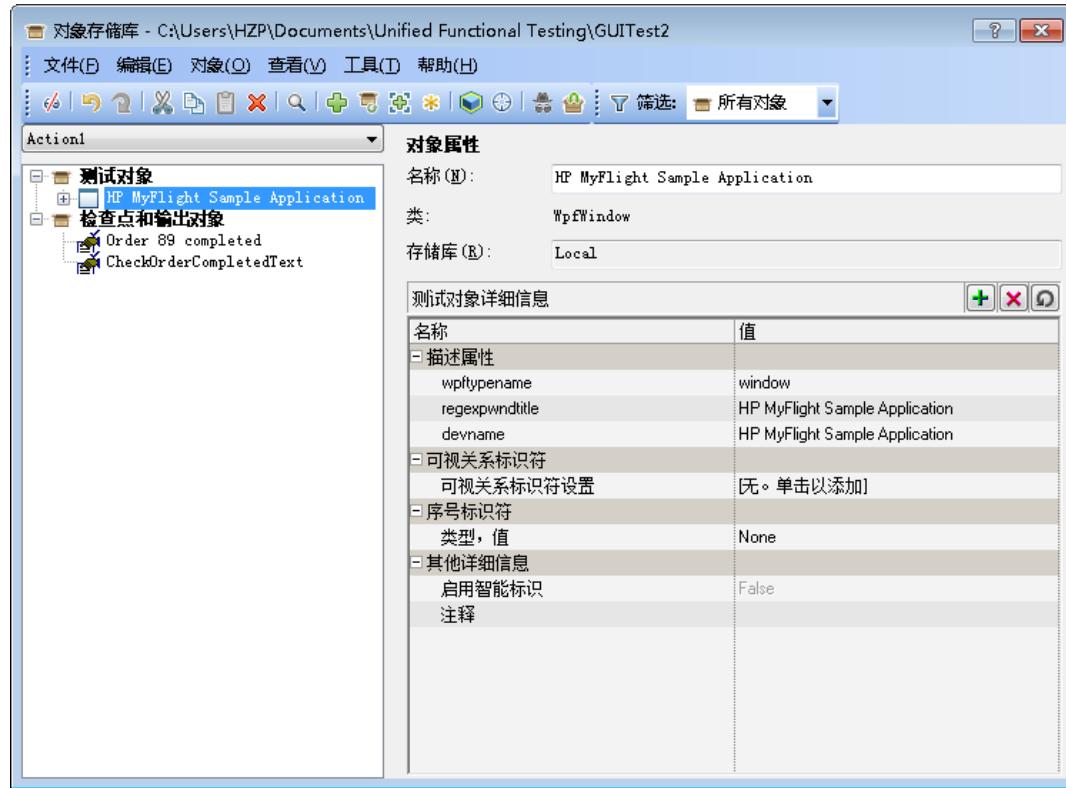
1. 启动 UFT 并打开“Book Flights Checkpoint”测试。

- 如需要，请如[为所有测试创建解决方案 \(第 20 页\)](#) 中所述打开 UFT。确保已加载 WPF 插件。
 - 选择“文件”>“打开”>“解决方案”。将打开“打开解决方案”对话框。
 - 在“打开解决方案”对话框中，导航到位于 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 的 **Flight Reservation Application** 解决方案，然后单击“打开”。
- 将打开“Flight Reservation Application”解决方案，其中包含您在[练习 6a:创建检查点测试 \(第 80 页\)](#) 中创建的“Book Flights Parameter”测试。

2. 将打开“对象存储库”窗口。

- 在解决方案浏览器中，双击“Flight Confirmation”操作。“Flight Confirmation”操作将在文档窗格的单独选项卡中打开。
- 在工具栏上，单击“对象存储库”按钮。“对象存储库”窗口将打开，并显示所有测试对象的树以及当前操作中的所有检查点和输出对象。

树包含所有本地对象和与该操作关联的任何共享对象存储库中的所有对象:



3. 选择一个操作查看其检查点。

- 在对象树上面的“操作”下拉菜单中，选择一个操作以显示其测试对象、检查点对象和输出值对象。
- 完成后，关闭“对象存储库”窗口。

备注: 在此练习中，您无需修改任何对象或检查点属性。

您随时可以运行带有检查点的测试。请继续[练习 6f:使用检查点运行和分析测试 \(第 96 页\)](#)学习有关运行包含检查点的测试并查看其结果的内容。

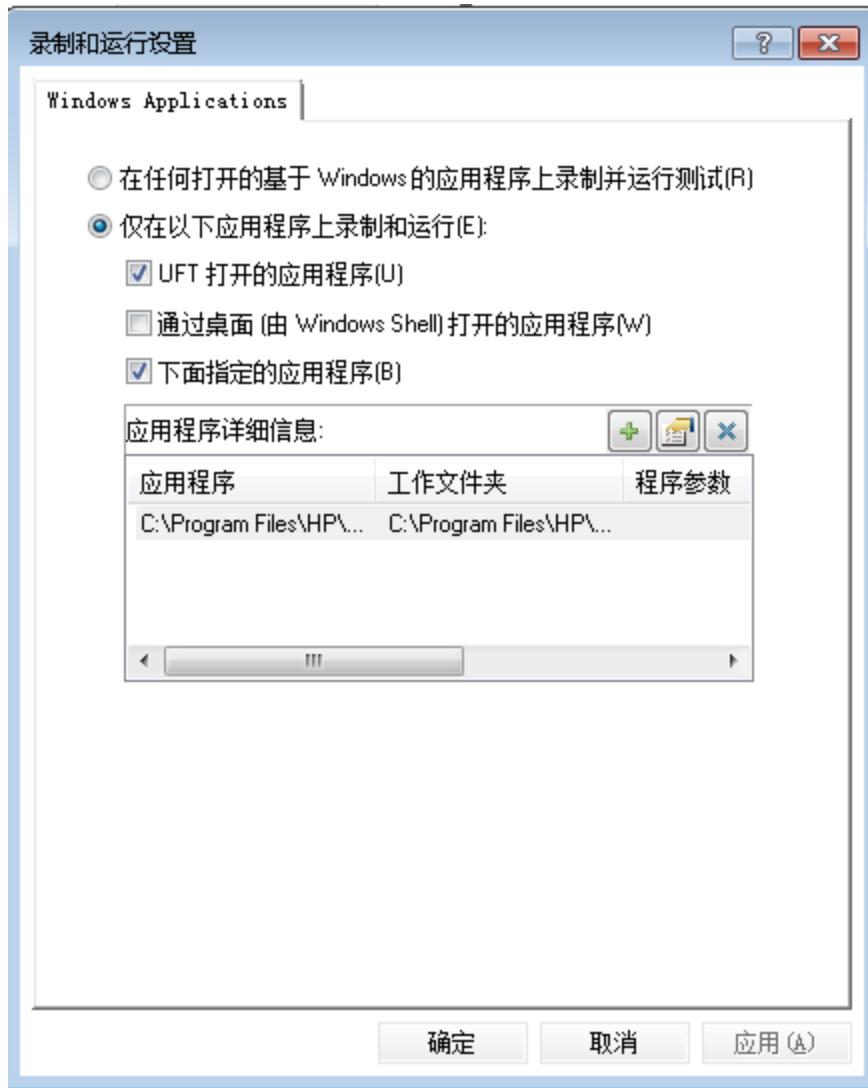
练习 6f: 使用检查点运行和分析测试

现在，您已创建了使用检查点的测试，您应运行该测试以查看检查点的执行情况。在本练习中，您将运行测试，并分析检查点结果。

1. 配置 UFT 打开航班预订应用程序。

- 在 UFT 中，选择“录制”>“录制和运行设置”。将打开“录制和运行设置”对话框。

- b. 在“Windows 应用程序”选项卡中，选择“仅在以下应用程序上录制和运行：”选项：



备注: 应仍旧保存应用程序详细信息，如您在[课程 4:运行和分析 GUI 测试 \(第 61 页\)](#) 中所设置的。

- c. 单击确定关闭对话框。
2. **开始运行测试。**
- 在工具栏中，单击“运行”按钮 。将打开“运行”对话框。
 - 在“结果位置”选项卡中，确保已选中“新建运行结果文件夹”。接受默认结果文件夹名。
 - 单击“确定”。
- UFT 打开航班预订应用程序，并执行这些步骤。在测试运行末尾，将打开运行结果。
3. **查看运行结果。**

如果显示运行结果，则运行结果应为“通过”，表明所有检查点均已通过。如果一个或多个检查

点失败，则测试运行将列为“失败”状态。

4. 查看标准检查点的结果。

- a. 在测试流中，查找“Flight Confirmation”节点。
- b. 在“Flight Confirmation”节点的“passengerName.Set”节点下，选择“Standard Checkpoint: "CheckName"”节点。
摘要将显示标准检查点的详细信息，包括所检查的属性及其值。检查点已通过，因为对象属性的实际值与预期值匹配：

S

5. 查看表检查点的结果。

- a. 在结果树中，展开“Select Flight”节点。
- b. 在“Action:Select Flight”节点下，查找“Check Cost”节点。
- c. 展开“Check Cost”节点，并选择“Standard Checkpoint:CheckCost”节点。

摘要将显示表和检查点的详细信息：

检查点详细信息

数据库检查点

"CheckCost"

描述

验证类型：字符串内容。设置：完全匹配 - 开；忽略空格 - 开；区分大小写 - 关。结果：已检查 24 个单元格；成功：24；失败：0

执行时间

2015-09-18 11:11:34

在某些情况下，步骤摘要也会显示有关表检查点的详细信息，其中包括从表对象捕获的数据。

6. 查看文本检查点的结果。

- a. 在测试流中，查找“Flight Confirmation”节点。
- b. 在“Flight Confirmation”节点的 **progBar.Wait** 步骤下，选择“TExt Checkpoint:CheckOrderCompleted”节点。

步骤摘要将显示检查点的详细信息。检查点已通过，因为实际文本与预期文本匹配：

The screenshot shows the HP UFT interface. On the left, the '测试流' (Test Flow) pane displays a sequence of steps under a 'Flight Confirmation' scenario. The steps include: 'passengerName.Set' (pass), '标准检查点: "passeng...' (pass), 'ORDER.Click' (pass), 'progBar.WaitProperty' (pass), and '文本检查点: "Order 95 c...' (pass). On the right, the '检查点详细信息' (Checkpoint Details) pane is open, showing a '文本检查点' ('Order 95 completed') that has passed. It includes a screenshot of the application window showing the text 'Order 95 completed', a description of the check point, and execution details (2015-09-17 16:28:27). Below this, the '测试对象' (Test Object) section shows 'WpfObject: "Order 95 completed"'. At the bottom right, there are language selection buttons for '简体中文' (Simplified Chinese) and 'English', and the 'UFT 报告' (UFT Report) button.

7. 关闭运行结果。

在文档窗格中，关闭包含运行结果的选项卡。

练习 6g: 创建输出值测试

在本练习中，您将创建要添加输出值步骤的测试。此测试基于您在[课程 1: 创建 GUI 测试和操作 \(第 24 页\)](#)中创建的 Book Flights 测试。

1. 启动 UFT 打开 Book Flights 测试。

- 如需要，请如[为所有测试创建解决方案 \(第 20 页\)](#)中所述打开 UFT。确保已加载 WPF 插件。
 - 单击“打开”按钮向下箭头 ，并选择“打开解决方案”。将打开“打开解决方案”对话框。
 - 导航到位于 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 中的 **Flight Reservation Application.ftsln** 文件，并单击“打开”。
- 将打开 Flight Reservation Application 解决方案，其中包含您在[课程 1: 创建 GUI 测试和操作 \(第 24 页\)](#)中创建的 Book Flights 测试。

2. 将测试另存为 Book Flights Output Value。

- 在解决方案浏览器中，选择“Book Flights”测试节点，然后选择“文件”>“另存为”。
 - 在“将测试另存为”对话框中，浏览到 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 目录，并将测试另存为 Book Flights Output Value。
- 在解决方案浏览器中，将 Book Flights 测试替换为新的 Book Flights Output Value 测试。原始 Book Flights 测试仍保存在文件系统中。

3. 将 Book Flights 测试添加回解决方案。

如果 **Book Flights** 和 **Book Flights Output Value** 测试包含在同一个解决方案中，则可以同时打开。这使您能够在想要比较或编辑这些测试时来回进行切换。

备注: 一次仅能运行一个测试。

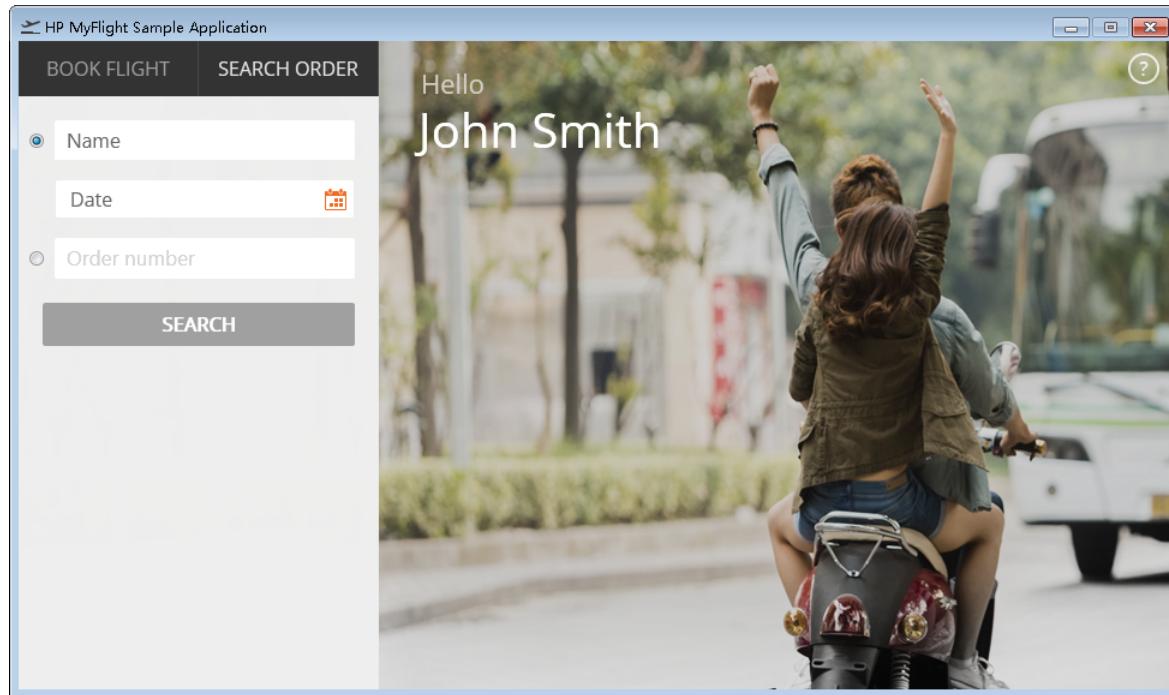
- a. 选择“文件”>“添加”>“现有测试”。
 - b. 在“添加现有测试”对话框中，导航到存储在 C:\HOMEPATH%\My Documents\Unified Functional Testing 中的 Book Flights 测试，然后单击“添加”。
- Book Flights 测试在解决方案浏览器中再次作为独立节点显示。

现在，您已创建了要添加输出值的测试，请继续[练习 6h:添加输出值步骤 \(第 100 页\)](#)学习向测试添加输出值步骤。

练习 6h:添加输出值步骤

如果创建原始 Book Flights 测试，则您会为主应用程序流程创建一项操作，即，您登录该应用程序，输入出发和抵达详细信息，选择航班，并为特定客户预订航班。

在航班预订应用程序中，有一个应用程序附加区域，通过该区域您可以搜索所有之前创建的航班订单：



在本练习中，您将创建获取“Flight Details”页面中步骤输出值的输出值步骤（在测试的“Flight Confirmation”操作中），并将此输出值用作“搜索”页面中对象的参数。

1. 在搜索页面上为测试步骤创建操作。

- a. 在解决方案浏览器中，单击“Book Flights”测试节点。测试流画布将在文档窗格的单独选项卡中打开。
- b. 在工具栏上，单击“插入对新操作的调用”按钮 。将打开“插入对新操作的调用”对话框。
- c. 在“插入对新操作的调用”对话框中，输入 Flight Order Search 作为新操作的名称。

- d. 保留其他设置为默认设置，并单击“确定”。

称为“Flight Order Search”的新操作块将添加到画布中测试流的结尾处，“Flight Order Search”操作在文档窗格的单独选项卡中打开。

2. 为其他应用程序页面创建对象存储库。

当创建 Book Flights 测试时，您将仅为主应用程序页面创建对象存储库。为了为搜索页面创建测试步骤，您需要为搜索页面创建附加对象存储库。

- a. 打开航班预订应用程序，如[浏览航班预订应用程序（第 18 页）](#)中所述。
- b. 输入应用程序的登录凭据：
 - 用户名：john
 - 密码：hp
- c. 单击“确定”登录。将打开“Flight Finder”页面。
- d. 在“Flight Finder”页面中，在窗口左上角，单击“Search Order”按钮。将打开“Search Details”页面。
- e. 在 UFT 中，选择“资源”>“对象存储库管理器”。将打开“对象存储库管理器”窗口。
- f. 在“对象存储库管理器”中，使用“导航并识别”进程，如[练习 2b: 使用导航并识别进程创建对象存储库（第 34 页）](#)中所述。
- g. 学习此页面的所有对象后，请单击“文件”>“保存”。
- h. 导航到 **C:\%HOMEPATH%\My Documents\Unified Functional Testing\Tutorial_Object Repositories** 文件夹，保存此对象存储库且命名为“Search”。
- i. 在航班预订应用程序中，在“Search Details”页面的“Name”框中输入 John Smith，然后单击“Search”。John Smith 所预订的航班列表将在单独的页面中显示。
- j. 在 UFT 中，再次打开“对象存储库管理器”窗口。
- k. 使用“Select Order”页面的“导航并识别”进程识别此页面的所有对象。
- l. 识别“Select Order”页面的所有对象后，请单击“文件”>“保存”。
- m. 在“保存对象存储库”对话框中，再次导航到 **C:\%HOMEPATH%\My Documents\Unified Functional Testing\Tutorial_Object Repositories** 文件夹，保存该对象存储库且命名为 Search_Results。
- n. 关闭“对象存储库管理器”。

3. 将新对象存储库与 Flight Order Search 操作相关联。

- a. 在 UFT 主窗口的解决方案浏览器中，在“Book Flights Output Value”测试节点下方，右键单击 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 操作节点并选择“将存储库与操作关联”。
 - b. 在“打开共享对象存储库”窗口中，导航至 **C:\%HOMEPATH%\My Documents\Unified Functional Testing\Tutorial_Object Repositories** 文件夹并选择“Search.tsr”文件。
 - c. 单击“打开”关联对象存储库。
 - d. 重复执行此过程以关联“Search Results.tsr”对象存储库（也存储在 **C:\%HOMEPATH%\My Documents\Unified Functional Testing\Tutorial_Object Repositories** 文件夹中）。
- Search.tsr** 和 **Search Results.tsr** 对象存储库文件均作为“Flight Order Search”操作子节点显示出来。

4. 添加步骤以使用 Search 和 Flight Orders 页面。

- a. 在解决方案浏览器中，双击“Flight Order Search”操作节点。“Flight Order Search”操作在文档窗格的单独选项卡中显示。
- b. 如果“关键字视图”显示出来，请选择“查看”>“编辑器”以显示编辑器。
- c. 将以下行粘贴到编辑器中：

```
WpfWindow("HP MyFlight Sample Application").WpfTabStrip("WpfTabStrip").Select  
"SEARCH ORDER"  
WpfWindow("HP MyFlight Sample Application").WpfEdit("byName").Set "John  
Smith"  
WpfWindow("HP MyFlight Sample Application").WpfButton("SEARCH").Click  
WpfWindow("HP MyFlight Sample Application").WpfTable  
("ordersDataGrid").SelectCell 1, 1  
WpfWindow("HP MyFlight Sample Application").WpfButton("SELECT ORDER").Click
```

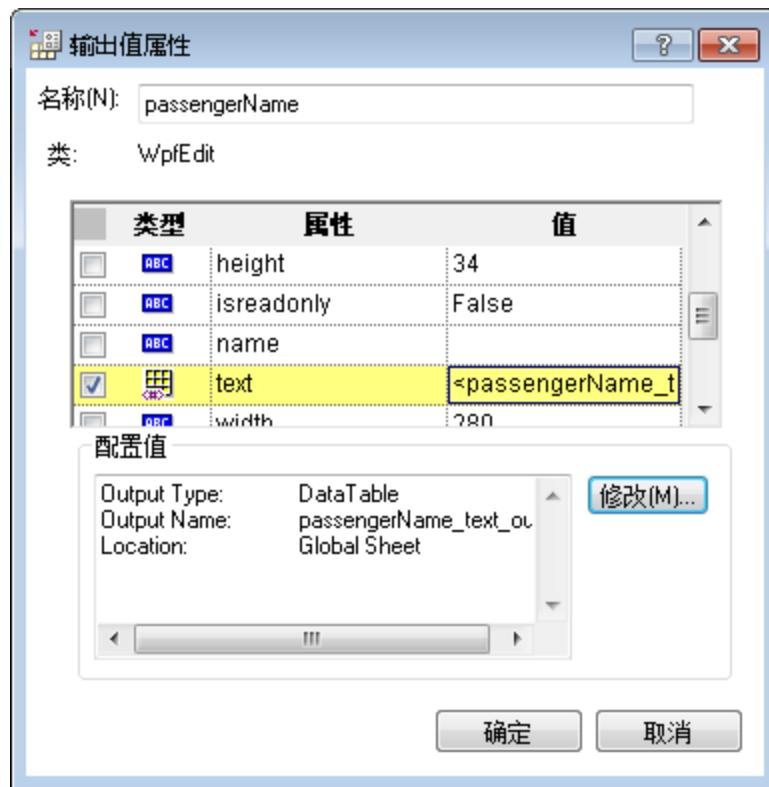
5. 打开航班预订应用程序的 Flight Details 页面。

- a. 在航班预订应用程序的“Search Results”页面，单击“BACK”按钮。将显示“Search Details”页面。
- b. 在“Search Details”页面的左上角，单击“Book Flight”按钮。将打开 Flight Finder 页面。
- c. 在“Flight Finder”页面中，输入航班详细信息：
 - **Departure City:** Los Angeles
 - **Arrival City:** Sydney
 - **Date:** 明天的日期
 - **Class:** Business
 - **Tickets:** 2
- d. 单击“Find Flights”按钮。将打开 Select Flight 页面。
- e. 在“Select Flight”页面中，选择第一行，并单击“Select Flight”。将打开“Flight Details”页面。

6. 向 Flight Confirmation 操作添加输出值步骤。

- a. 在 UFT 中，在解决方案浏览器的“Book Flights”测试节点下方，双击“Flight Confirmation”操作节点。“Flight Confirmation”操作将在文档窗格的单独选项卡中打开。
- b. 如果显示编辑器，则选择“查看”>“关键字视图”以显示“关键字视图”。

- c. 在“关键字视图”中，右键单击“byName”步骤，并选择“插入输出值”。将打开“输出值属性”对话框：



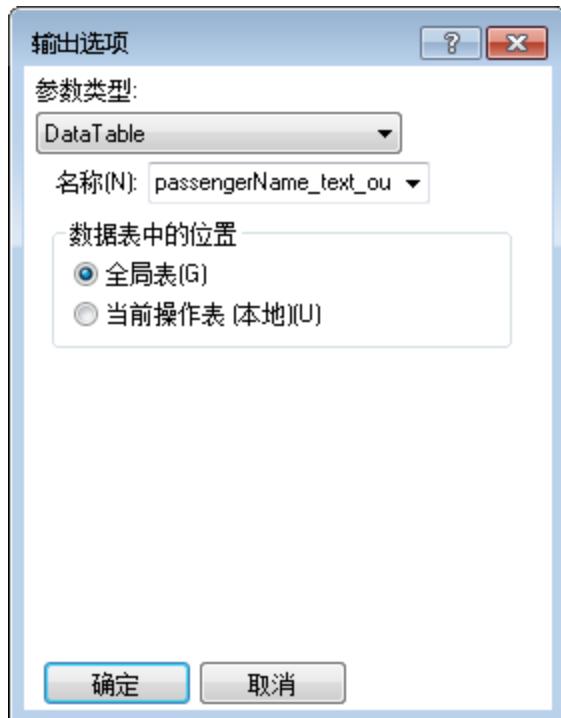
对话框中将显示要向其插入输出值的属性：

- **Name** 是在应用程序中定义的对象名称，在此例中为 **byName**。
 - **Class** 是对象的类型，在此例中为 **WpfEdit**，指示对象为编辑框。
 - “**类型**”列中的“ABC”图标指示属性值是常量。
 - 网格中列出了您可以选择输出的对象属性，包括您可以作为输出发送的对象的“**属性**”名称和“**值**”。
- d. 在“名称”框中，输入 OutputPassengerName 作为新输出值的名称。

- e. 在对象属性网格中，滚动属性并选择“text”属性行。请注意，此时此属性并没有值。



- f. 在属性网格下方，单击“修改”按钮。将打开“输出选项”对话框。(使用此对话框您可以确定存储此测试步骤输出的位置。)



创建输出值时，您可以选择存储输出值的位置：

- DataTable 参数
- 测试/操作参数 (如果已创建)
- 环境变量
- 组件参数 (如果已创建)

在此教程中，我们将输出值保存在数据表中。

- g. 在“参数”类型下拉菜单中，选择“数据表”。UFT 将更新对话框的字段，并建议参数的默认名称。
- h. 在“名称”字段中，输入 passengerName_text。
- i. 在“数据表中的位置”区域中，确保已选中“全局表”。这会确保将此输出保存在全局数据表中，以便测试中的其他操作可以访问该值。
- j. 单击“确定”关闭对话框。UFT 将更新“输出值属性”对话框中的值以反映 DataTable 参数存储选项：

类型	属性	值
ABC	isreadonly	False
ABC	name	
<input checked="" type="checkbox"/>	text	<passengerName_t>
ABC	width	280
ABC	v	99

- k. 在“插入语句”区域中，选择“当前步骤之后”选项，然后单击“确定”。

UFT 立即将“输出”步骤插入到 **passengerName.Set** 步骤后面:

项	操作	值	文档
HP MyFlight Sample Application			
passengerName	Set	"aa"	输入 "aa" (在 "passengerName" edit box 中)。
passengerName	Output	CheckPoint("passengerName")	存储 "passengerName" edit box 选定属性的当前值。
ORDER	Click		单击 "ORDER" button。
progBar	WaitProperty	"value", "100"	等到 "progBar" progress bar 的 "value" 属性值等于 100。
NEW SEARCH	Click		单击 "NEW SEARCH" button。
+ 新建步骤			

在编辑器中，步骤如下所示:

```

WpfWindow("Book Flights").WpfEdit("passengerName").Set "John Smith"
WpfWindow("Book Flights").WpfEdit("passengerName").Output CheckPoint
("OutputPassengerName")
WpfWindow("Book Flights").WpfButton("ORDER").Click
WpfWindow("Book Flights").WpfProgressBar("progBar").WaitProperty "value", "100"
WpfWindow("Book Flights").WpfButton("NEW SEARCH").Click

```

数据表中的全局表也会进行相应更新:

A1	
passengerName_text_out	
1	
2	
3	
4	
5	
6	
7	
8	

Global / Action1 /

7. 使用存储的输出值参数化 Search 操作。

- 在解决方案浏览器中，双击“Flight Order Search”操作节点。“Flight Order Search”操作将在文档窗格的单独选项卡中打开。
- 在“Flight Order Search”选项卡中，选择“byName”行。
- 在“byName”行的“值”列中，单击“配置值”按钮 。将打开“值配置选项”对话框。
- 在“值配置选项”对话框中，选择“参数”单选按钮。
- 在“参数”下拉列表中，选择“DataTable”。对话框将相应地更新其他字段。
- 在“数据表中的位置”区域中，选择“全局表”选项。
- 在“名称”下拉列表中，选择“passengerName_text”参数并单击“确定”。

UFT 更新 **byName** 行以反映该值现在是由数据表参数提供的:

项	操作	值	文档
HP MyFlight Sample Application_2			
WpfTabStrip	Select	"SEARCH ORDER"	选择 "SEARCH ORDER" 选项卡(在 "WpfTabStrip" t...
byName	Set	DataTable("passengerName_text...")	输入 <the 'passengerName_text_out' Data Table c...
SEARCH	Click		单击 "SEARCH" button。
ordersDataGrid	SelectCell	1,2	选择行 1、列 2(在 "ordersDataGrid" table 中) ...
SELECT ORDER	Click		单击 "SELECT ORDER" button。
+ 新建步骤			

在编辑器中，语句如下所示:

```
WpfWindow("HP MyFlight Sample Application").WpfEdit("byName").Set DataTable
("passengerName_text", dtGlobalSheet)
```

8. 保存测试。

单击“保存” 。

9. 运行测试并查看运行结果。

- 选择“录制”>“录制和运行设置”。将打开“录制和运行设置”对话框。
- 在“记录和运行设置”对话框的“Windows 应用程序”选项卡中，选择“仅在以下应用程序上录制和运行:”选项。应用程序详细信息应保存自之前的测试运行。
- 单击“确定”保存设置并关闭对话框。
- 在工具栏中，单击“运行”按钮 。
- 在“结果位置”选项卡中，确保已选中“新运行结果文件夹”选项。接受默认结果文件夹名。
- 单击“确定”。测试运行完成后，运行结果将打开。
- 在测试流中，查找“Flight Confirmation”操作节点。
- 在“Flight Confirmation Summary”节点下，选择“Standard Output: "OutputPassengerName"”节点。运行结果将显示测试步骤的摘要。
摘要将显示输出值的详细信息:
- 在测试流中，查找“Flight Order Search”节点。
- 在“Flight Order Search”节点下方，选择“byName.Set”节点。运行结果将显示测试步骤的摘要。
摘要显示了此步骤的结果，包括 Set 操作使用的值。此值应为输出值。

10. 关闭运行结果。

当完成对运行结果的查看时，请关闭显示运行结果的选项卡。

课程 7:创建函数和函数库

UFT 提供可满足很多测试需求的内置函数和方法。但是，有时您可能需要针对特定测试对象类执行非内置的特定任务。在这些情况下，您可以为此任务创建用户定义的函数。然后，将此函数保存在已与测试关联的函数库文件中，再在需要执行此任务时，将该函数调用作为步骤插入。

在[课程 2:创建对象存储库 \(第 27 页\)](#) 中，您创建了共享对象存储库并将其与测试中的操作关联。在本课程中，您将使用类似的过程：创建函数和函数库，然后将函数库与测试相关联。通过将此函数库与测试相关联，您可以在测试中调用任何函数。

此课程包含以下内容：

· 函数和函数库 - 概述	109
· 练习 7a:创建函数	109
· 练习 7b:将函数库与测试关联	111
· 练习 7c:使用函数执行检查	112

函数和函数库 - 概述

在 UFT 中，您可以创建函数执行 UFT 标准类和方法不支持的特定任务。“函数”是一组已编码的步骤，用于执行无任何合适的内置方法可供使用的特定任务。您可能想要测试包括此类任务，甚至将此任务重复几次。因此，您需要能够轻松地访问函数。

创建函数后，您可以将其存储在“函数库”中。这些函数库将用作用户定义函数的存储库。每个函数库均可分配到一个测试（或多个测试），这样，测试便可以调用函数作为测试步骤。

在本课程中，您将创建一个函数，检查由航班预订应用程序生成的页面上的日期格式，然后将函数调用添加到测试。

练习 7a: 创建函数

在本练习中，您将创建将从其他测试调用的函数。此函数将检查日期是否以正确的格式显示。此函数还检查日期是否有效，例如，月份是否未超过 12，或者日期是否未超过 31。

1. 启动 UFT 打开 Book Flights 测试。

- a. 如果当前未打开 UFT，请将 UFT 打开，如[为所有测试创建解决方案（第 20 页）](#)中所述。确保已加载 WPF 插件。
- b. 选择“文件”>“打开”>“解决方案”。将打开“打开解决方案”对话框。
- c. 在“打开解决方案”对话框中，导航到位于 **C:\%HOMEPATH%\My Documents\Unified Functional Testing 的 Flight Reservation Application.ftsln** 文件，然后单击“打开”。此解决方案将显示在解决方案浏览器中，其中包含 Book Flights 测试。

2. 新建函数库。

- a. 在工具栏上，单击“新建”按钮向下箭头  并选择“函数库”。将打开“新建函数库”对话框。
- b. 在“新建函数库”对话框中，如下所示填充函数库详细信息：

查找范围：	i. 浏览到 C:\%HOMEPATH%\My Documents\Unified Functional Testing 文件夹。 ii. 在此文件夹中，新建文件夹并命名为 Tutorial_Function Libraries 。 iii. 打开此文件夹。
文件名：	CheckDate Function

- c. 单击“创建”。UFT 将在文档窗格的单独选项卡中打开空函数库。

3. 创建函数。

在 CheckDate 函数库中，粘贴以下代码：

```
'The following function checks whether a date string (dateStr)
```

```
'has the characters representing DD-<MMM 字符串>-YYYY

Function check_data_validity( dateStr )
    Dim firstDashPos, secondDashPos
    Dim mmPart, ddPart, yyyyPart
    firstDashPos = inStr( dateStr , "-" )
    secondDashPos = inStrRev( dateStr, "-" )
    If ( (firstDashPos <> 2 and firstDashPos <> 3) or (secondDashPos <> 6 and
secondDashPos <> 7)) Then
        reporter.ReportEvent micFail,"Format check", "Date string is"&" missing at least
one dash ( - )."
        check_data_validity = False
        Exit function
    End If

    if firstDashPos = 2 Then
        ddPart = mid( dateStr, 1, 1)
    else
        ddPart = mid( dateStr, 1,2 )
    End If
    mmPart = mid( dateStr, firstDashPos+1, 3 )
    yyyyPart = mid( dateStr, secondDashPos +1 , 4 )

    If inStr(mmPart, "Jan") and inStr(mmPart, "Feb") and inStr(mmPart, "Mar") and
inStr(mmPart, "Apr") and inStr(mmPart, "May") and inStr(mmPart, "Jun") and inStr
(mmPart, "Jul") and inStr(mmPart, "Aug") and inStr(mmPart, "Sep") and inStr(mmPart,
"Oct") and inStr(mmPart, "Nov") and inStr(mmPart, "Dec") Then
        reporter.ReportEvent micFail, "Format Check", "The month"&" value is invalid.It
is not a valid month string."
        check_data_validity = False
        Exit function
    End If

    If ddPart > 31 Then
        reporter.ReportEvent micFail, "Format Check", "The date"& " value is invalid.It
exceeds 31."
        check_data_validity = False
        Exit function
    End If

    If yyyyPart < 2013 Then
        reporter.ReportEvent micFail, "Format Check", "The year"& " value is invalid.
(Prior to this year.)"
        check_data_validity = False
        Exit function
```

```
End If  
  
check_data_validity = True  
  
End Function
```

4. 保存函数库。

单击“保存”。

5. 关闭函数库。

选择“文件”>“关闭”。函数库选项卡将关闭，但测试仍保持打开状态。

现在，您已创建了函数，您需要将其与测试相关联，以便在测试步骤中使用这些函数。请继续[练习 7b: 将函数库与测试关联 \(第 111 页\)](#) 学习如何将函数库与测试相关联。

练习 7b: 将函数库与测试关联

在[练习 7a: 创建函数 \(第 109 页\)](#) 中，您创建了用于在测试对象上运行日期检查的函数和函数库。但是，在测试中使用此函数之前，必须先将该函数库与测试关联。

在本练习中，您将学习如何将函数库与测试相关联。

1. 启动 UFT 打开 Book Flights 测试。

- 如果当前未打开 UFT，请将 UFT 打开，如[为所有测试创建解决方案 \(第 20 页\)](#) 中所述。确保已加载 WPF 插件。
- 选择“文件”>“打开”>“解决方案”。将打开“打开解决方案”对话框。
- 在“打开解决方案”对话框中，导航到位于 C:\%HOMEPATH%\My Documents\Unified Functional Testing 的 Flight Reservation Application.ftsln 文件，然后单击“打开”。此解决方案将显示在解决方案浏览器中，其中包含 Book Flights 测试。

2. 将 CheckDate Function.qfl 与 Book Flights 测试关联

- 如果解决方案浏览器尚未打开，则通过单击工具栏上的“解决方案浏览器”按钮 将其打开。
- 在解决方案浏览器中，右键单击“Book Flights”测试节点，并选择“添加”>“关联函数库”。将打开“打开函数库”对话框。
- 在“打开函数库”对话框中，导航到位于 C:\%HOMEPATH%\Unified Functional Testing\Tutorial_Function Libraries 的 CheckDate Function.qfl 文件，然后单击“打开”。
- 在“自动相对路径转换”对话框中，单击“是”。(此操作会将到函数库的路径转换为相对路径。)

此时，函数库已与 Book Flights 测试相关联，并显示在 Book Flights 测试的 **Function Libraries** 文件夹中的解决方案浏览器中。

备注: 只要文件夹层次结构保持不变，在将包含测试和其他文件的文件夹从一个位置移动到另一个位置时，使用相对路径就可以保持路径的有效性。

3. 保存测试。

在文档窗格中，选择“Book Flights”画布选项卡，然后单击“保存”。

现在，函数库已与测试相关联，因此，您可以在测试步骤中使用其函数了。请继续[练习 7c: 使用函数执行检查 \(第 112 页\)](#) 学习在测试步骤中使用函数。

练习 7c: 使用函数执行检查

在[课程 6: 创建检查点和输出值 \(第 78 页\)](#) 中，您创建了许多检查点，用于检查航班预订应用程序中的对象。

在本练习中，不再使用 UFT 用户界面创建检查点，而是使用在[练习 7a: 创建函数 \(第 109 页\)](#) 中创建的函数来检查“Flight Finder”操作中日历对象的日期格式是否正确。

1. 启动 UFT 打开 Book Flights 测试。

- a. 如果当前未打开 UFT，请将 UFT 打开，如[为所有测试创建解决方案 \(第 20 页\)](#) 中所述。确保已加载 WPF 插件。
- b. 选择“文件”>“打开”>“解决方案”。将打开“打开解决方案”对话框。
- c. 在“打开解决方案”对话框中，导航到位于 C:\%HOMEPATH%\My Documents\Unified Functional Testing 的 Flight Reservation Application.ftsln 文件，然后单击“打开”。此解决方案将显示在解决方案浏览器中，其中包含 Book Flights 测试。

2. 将测试另存为 Book Flights Function。

- a. 在解决方案浏览器中，右键单击“Book Flights”节点，并选择“另存为”。将打开“将测试另存为”对话框。
- b. 在“将测试另存为”对话框中，浏览至 C:\%HOMEPATH%\My Documents\Unified Functional Testing 文件夹。
- c. 输入 Book Flights Function 作为测试名称。

在解决方案浏览器中，将“Book Flights”测试替换为“Book Flights Function”测试。Book Flights 测试仍单独保存在文件系统中。

3. 将 Book Flights 测试添加回解决方案。

如果 Book Flights 和 Book Flights 函数包含在相同的解决方案中，则可以同时打开。这使您能够在想要比较或编辑这些测试时来回进行切换。

备注: 一次仅能运行一个测试。

- a. 在工具栏上，单击“添加”按钮向下箭头  并选择“添加”“现有测试”。将打开“将测试添加到解决方案中”对话框。

- b. 在“将测试添加到解决方案中”对话框中，导航到 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 文件夹。

- c. 在文件夹中，选择“Book Flights”测试，并单击“打开”。

Book Flights 测试节点将再次添加到解决方案浏览器中。请注意，这些测试是按字母顺序排列的。

4. 在航班预订应用程序中显示 Flight Finder 页面。

- a. 打开航班预订应用程序，如[浏览航班预订应用程序 \(第 18 页\)](#) 中所述。

- b. 在登录页面，输入登录用户凭据：

◦ 用户名: john

◦ 密码: hp

- c. 单击“确定”登录。将打开“Flight Finder”页面。

5. 借测检查点对象的对象属性。

在本练习中，您将使用函数检查日历对象的日期格式是否正确。为此，您必须识别要设置检查点的对象上的属性。

- a. 在 UFT 窗口中，单击工具栏上的“对象侦测器”按钮 。将打开“对象侦测器”对话框。

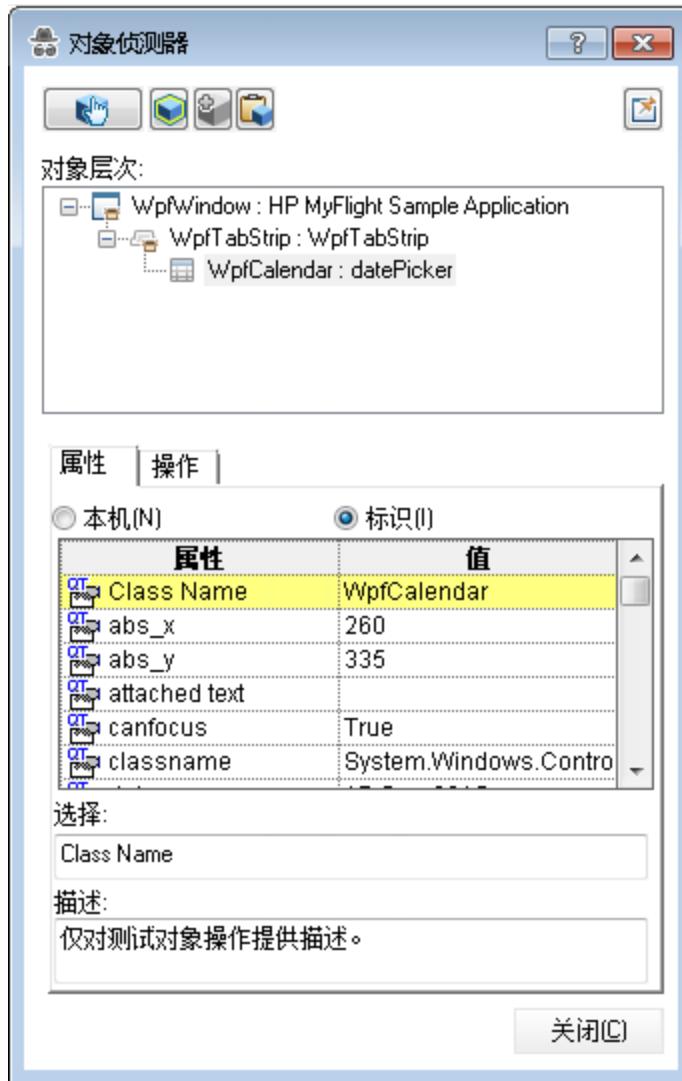
- b. 将“对象侦测器”对话框移到窗口边缘，以便可以在“对象侦测器”对话框仍然打开且可见的情况下侦测应用程序。

- c. 在“对象侦测器”对话框中，单击指向手按钮 。按下指向手时，UFT 将隐藏，而“对象侦测器”对话框将显示在航班预订应用程序之上。

- d. 在航班预订应用程序中，单击日期输入字段对象：



“对象侦测器”对话框将显示对象属性:



- e. 在属性网格中 (对话框底部), 滚动属性直到找到带有日期值的属性 (格式为 **DD-MMM-YYYY**)。请注意此属性的名称, 因为稍后您将在测试中插入的函数调用步骤中用到它。
- f. 单击“关闭”关闭“对象侦测器”并返回到测试。

6. 打开函数调用的操作。

在解决方案浏览器中, 展开“Book Flights Function”测试节点, 并双击“Flight Finder”操作。

“Flight Finder”操作将在文档窗格的单独选项卡中打开。

7. 在编辑器中添加函数调用。

在此步骤中, 您将添加在练习 7a: 创建函数 (第 109 页) 中添加到函数库的 check_data_validity 函数。

- a. 如果“关键字视图”显示出来，请选择“查看”>“编辑器”以显示编辑器。
- b. 在编辑器中，将光标放置在“WpfCalendar.SetDate”步骤后，并按下“ENTER”。
- c. 向编辑器添加步骤以检索 **datePicker** 对象的日期属性：

```
departureDate = WpfWindow("HP MyFlight Sample"&_ "Application").WpfCalendar  
("datePicker").GetROProperty("date")
```

- 此步骤将检索日期属性的值，以便 UFT 在检查该属性时，在该对象上运行检查点函数。
- d. 向编辑器添加其他步骤以调用检查点函数：

```
If check_data_validity(departureDate) Then  
    reporter.ReportEvent micPass, "Date is valid" , departureDate  
End If
```

添加这些步骤后，操作应如下所示（尽管使用的日期不同）：

```
WpfWindow("HP MyFlight Sample Application").WpfComboBox("fromCity").Select  
"Los Angeles"  
WpfWindow("HP MyFlight Sample Application").WpfComboBox("toCity").Select  
Sydney  
WpfWindow("HP MyFlight Sample Application").WpfCalendar("datePicker").SetDate  
"17-Jul-2014"  
departureDate = WpfWindow("HP MyFlight Sample Application").WpfCalendar  
("datePicker").GetROProperty("date")  
If check_data_validity(departureDate) Then  
    reporter.ReportEvent micPass, "Date is valid" , departureDate  
End If  
WpfWindow("HP MyFlight Sample Application").WpfComboBox("Class").Select  
"Business"  
WpfWindow("HP MyFlight Sample Application").WpfComboBox  
("numOfTickets").Select "2"  
WpfWindow("HP MyFlight Sample Application").WpfButton("FIND FLIGHTS").Click
```

- e. 粘贴这些步骤后，确保以下步骤 - WpfWindow("HP MyFlight Sample Application").WpfComboBox("Class").Select "Business" - 在粘贴的步骤后仍以其自身行开头。如果不是，则将光标放置在此步骤前面，按下“ENTER”。
在应用程序中，您希望 UFT 在编辑下一个字段之前运行此函数。
- f. 在操作中，确保删除 **departureDate** 代码行中的换行符，以便行以 **departureDate** 开始，以 ("date") 结束。

8. 在“关键字视图”中查看这些步骤。

选择“查看”>“关键字视图”切换到关键字视图。

请注意函数调用步骤作为 **GetRProperty** 步骤下的单独步骤显示:

项	操作	值	分配	注释	文档
HP MyFlight Sample Application					
fromCity	Select	"Los Angeles"			从 "fromCity" list 中选择 "Los Angeles" ...
toCity	Select	Sydney			从 "toCity" list 中选择 Sydney 项。
datePicker	SetDate	"17-Jul-2014"			将 "datePicker" calendar 中的日期或日...
datePicker	GetRProperty	"date"	存储在 ...		检索 "datePicker" calendar 的 "date" 属...
函数调用					
Reporter	check_data_validity	departureDate			
Class	Select	reporter.ReportEvent micPass, "			从 "Class" list 中选择 "Business" 项。
numOfTickets	Select	"Business"			从 "numOfTickets" list 中选择 "2" 项。
FIND FLIGHTS	Click	"2"			单击 "FIND FLIGHTS" button。
+ 新建步骤					

9. 保存测试。

选择文件 > 保存。

10. 运行测试。

- 在工具栏中，单击“运行”按钮 。将打开“运行”对话框。

备注: 运行测试之前，请确保航班预订应用程序已关闭。

- 在“运行”对话框中，在“结果位置”选项卡中，确保选中“新建运行结果文件夹”选项。接受默认结果文件夹名。
- 单击“运行”。UFT 将按顺序运行步骤。

测试运行完成后，运行结果将打开。

11. 分析运行结果。

- 在运行结果的测试流中，在“Flight Finder”节点下，选择“Date is valid node”。

请注意，步骤名称旁边将显示绿色的复选标记。这表示检查点已通过您添加的每个函数。

课程 8:在测试中使用洞察

有时，在创建测试和测试对象时，普通对象标识无法帮助标识应用程序中要测试的对象。标准对象标识依赖于对象的属性，如在应用程序或浏览器窗口中所处的位置、在窗口中出现的时间或许多其他属性，但属性对标识应用程序中要测试的对象并无帮助。

对于常规对象标识不能正常工作或无法满足您的需要的情况，UFT 还提供了基于图像的对象识别机制，名为“洞察”。这种机制使您能够通过捕获图像的快照并在测试运行期间使用捕获的图像来标识对象。

在本课程中，您将学习如何使用洞察来标识对象并在测试中使用这些对象。

此课程包含以下内容：

· 洞察对象标识 - 概述	118
· 练习 8a:为洞察对象创建测试	118
· 练习 8b:将洞察对象添加到对象存储库	119
· 练习 8c:在测试中使用洞察对象	122

洞察对象标识 - 概述

在 UFT 中，您可以使用“洞察”，这是一种基于图像的标识功能，用于识别应用程序中的对象。洞察根据对象的外观而非使用对象属性（是应用程序/对象设计的一部分）来标识对象。当借助 UFT 不支持的技术或使用在远程计算机上运行的应用程序来设计应用程序时，洞察对象标识尤为有用。

使用洞察对象标识时，UFT 会存储对象的图像，作为其创建的洞察测试对象的一部分。然后在运行测试时，UFT 将这些图像用作主要对象属性来标识运行时应用程序中的对象。

您可以在对象存储库中或在录制时创建洞察对象。在本课程中，您将仅使用在对象存储库中添加洞察对象。

练习 8a:为洞察对象创建测试

在此练习中，您将创建一个测试以使用洞察对象来添加测试步骤。此测试将与包含 **Book Flights** 测试的解决方案分开保存。

1. 启动 UFT。

- a. 如果 UFT 尚未打开，请按[为所有测试创建解决方案（第 20 页）](#)中所述打开。确保已打开 WPF 插件。

- b. 在工具栏中，单击“新建”按钮 。将打开“新建测试”对话框。
- c. 在“新建测试”对话框中，选择“GUI 测试”。
- d. 将测试命名为 Insight。接受默认位置。

备注: 请勿为此测试输入解决方案名称。此测试与本教程其他课程中使用的解决方案分开保存。

- e. 单击“创建”创建测试。

该测试将在文档窗格的单独选项卡中打开。

2. 将对象存储库与操作关联。

在此测试中，将所有测试步骤保存在一个操作中。但您需要多个对象存储库，以便具有相应的测试对象。

在此测试中，您将登录航班预订应用程序，然后单击宣传图像中的链接以预订航班。因此，需要包含该应用程序“Login”页面和“Flight Finder”页面的对象存储库。

- a. 在解决方案浏览器中，右键单击“Action1”节点并选择“将存储库与操作关联”。将打开“打开共享对象存储库”对话框。
- b. 在“打开共享对象存储库”对话框中，导航到您在[课程 2:创建对象存储库（第 27 页）](#)中创建的 **Login.tsr** 对象存储库文件（位于 **C:\%HOMEPATH%\Unified Functional Testing\Tutorial_Object Repositories** 中），然后单击“打开”。
Login.tsr 对象存储库将显示为 **Action1** 节点的子节点。
- c. 重复上述过程，将 **Flight Finder.tsr** 对象存储库文件添加到 **Action1**。

3. 保存测试。

单击“保存”。

您已创建了要使用洞察对象的测试，请继续[练习 8b:将洞察对象添加到对象存储库（第 119 页）](#) 将洞察对象添加到对象存储库，以便用于测试步骤。

练习 8b:将洞察对象添加到对象存储库

在[练习 8a:为洞察对象创建测试（第 118 页）](#) 中，您创建了洞察对象将使用的测试的结构。但必须先将这些对象添加到对象存储库，才能将这些对象添加到测试。在本课程中，您将学习如何使用对象存储库功能将洞察对象包含在测试内。

1. 启动 UFT。

- a. 如果 UFT 尚未打开，请按[为所有测试创建解决方案（第 20 页）](#) 中所述打开。确保已打开 WPF 插件。
- b. 在工具栏中，单击“打开”下拉箭头 并选择“打开测试”。将打开“打开测试”对话框。
- c. 在“新建测试”对话框中，导航到 **Insight** 测试（保存在 **C:%HOMEPATH%\Unified Functional Testing** 中），然后单击“打开”。

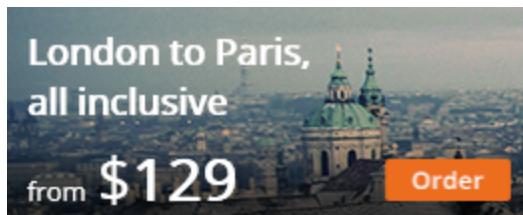
该测试将在文档窗格的单独选项卡中打开。

2. 新建对象存储库。

- a. 选择“资源”>“对象存储库管理器”。将打开“对象存储库管理器”对话框。
- b. 在“对象存储库管理器”窗口中，选择“文件”>“保存”。将打开“保存共享对象存储库”对话框。
- c. 在“保存共享对象存储库”对话框中，导航到 **Tutorial_Object Repositories** 文件夹（位于 **C:%HOMEPATH%\Unified Functional Testing** 中）。
- d. 将对象存储库命名为“Insight”，然后单击“保存”。

3. 打开航班预订应用程序的 Flight Finder 页面。

在航班预订应用程序中，您将注意到 Flight Finder 页面上有一个不断变化的对象，显示特殊航班的广告：



因为此图像会定期在应用程序中更改，所以它是您可以使用洞察对象识别测试的对象类型，因为传统的对象识别依赖于在屏幕上相对于其他对象的位置、序号标识符等属性。另外，位于图像内部的“Order”按钮无法作为单独的对象识别。例如，如果您使用对象侦测器来标识“Order”按钮，则 UFT 无法识别。

- a. 打开航班预订应用程序，如[浏览航班预订应用程序 \(第 18 页\)](#) 中所述。
- b. 在 Login 窗口中，输入用户凭据：
 - 用户名: john
 - 密码: hp
- c. 单击“OK”登录应用程序。将打开 Flight Finder 页面。

4. 向对象存储库添加洞察对象。

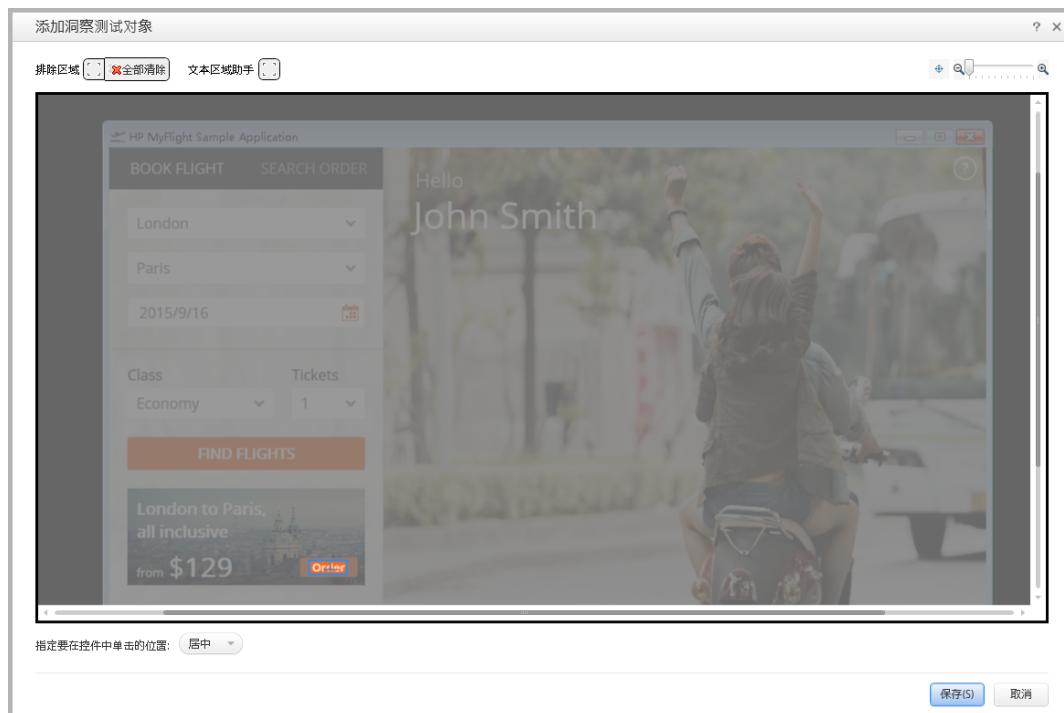
- a. 在“对象存储库管理器”的工具栏中，单击“添加洞察对象”按钮 。将打开“选择识别模式”对话框：



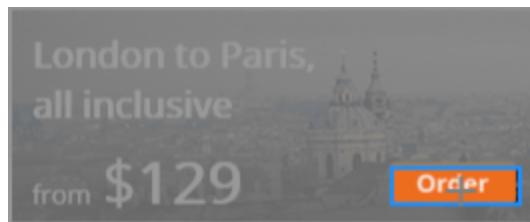
- b. 在“选择识别模式”对话框中，单击“自动”按钮。将在视图中隐藏 UFT 并显示应用程序。

备注: 选择“自动”模式，UFT 可自动选择区域/对象作为洞察对象。如果选择“手动”，则可以指定要用作洞察对象的区域/对象。

- c. 在航班预订应用程序中，单击航班宣传图像中的“Order”按钮。将打开“添加洞察测试对象”对话框：

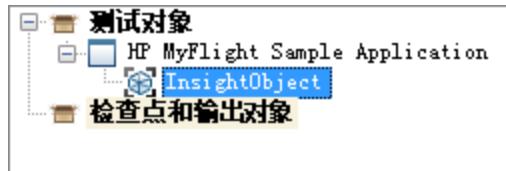


- d. 在图像编辑器(对话框中央)中，拖动对象选择框，以便其包含整个 Order 按钮：



- e. 在“指定要在控件中单击的位置”选项中，选择“居中”。
f. 单击“保存”，将对象添加到对象存储库。

已将洞察对象添加为对象存储库中的顶层对象：



- g. 在“对象存储库管理器”的“对象属性”窗格中(右侧)，将对象“InsightObject”重命名为“Promotion Order”。
h. 在工具栏中，单击“保存” 更改对象名称并保存对象存储库。

现在您已经创建了包含洞察图像的对象存储库，您可以在测试中使用这些对象。请继续[练习 8c:在测试中使用洞察对象\(第 122 页\)](#)学习在测试中用使用洞察对象并运行测试。

练习 8c:在测试中使用洞察对象

在[练习 8b:将洞察对象添加到对象存储库 \(第 119 页\)](#) 中，您已为应用程序中的对象（图像）创建了洞察对象。在此练习中，您将在测试中使用这些对象并查看使用洞察对象标识时测试如何运行。

1. 启动 UFT。

- a. 如果 UFT 尚未打开，请按[为所有测试创建解决方案 \(第 20 页\)](#) 中所述打开。确保已打开 WPF 插件。
- b. 在工具栏中，单击“打开”下拉箭头 并选择“打开测试”。将打开“打开测试”对话框。
- c. 在“新建测试”对话框中，导航到 **Insight** 测试（保存在 **C:\%HOMEPATH%\Unified Functional Testing** 中），然后单击“打开”。
该测试将在文档窗格的单独选项卡中打开。

2. 将洞察对象存储库与测试关联。

- a. 在解决方案浏览器中，右键单击“Action1”节点并选择“将存储库与操作关联”。将打开“打开共享对象存储库”对话框。
- b. 在“打开共享对象存储库”对话框中，导航到保存在 **C:\%HOMEPATH%\Unified Functional Testing\Tutorial_Object Repositories** 中的 **Insight.tsr** 对象存储库文件，然后单击“打开”。
现在 **Insight.tsr** 文件在解决方案浏览器中显示为 **Action1** 节点的子节点，其对象可在测试中使用。
- c. 单击“保存” 保存更改。

3. 向测试添加登录步骤。

向测试添加步骤以登录航班预订应用程序，如[练习 3a:在关键字视图中向 Login 操作添加步骤 \(第 39 页\)](#) 中所述。

4. 向测试添加洞察对象。

- a. 在“关键字视图”中“确定”按钮步骤的下方，单击“项”列并从下拉列表中选择“对象来自存储库”。将打开“选择测试对象”对话框。
- b. 在“选择测试对象”对话框的测试对象树中，选择“Promotion Order”节点：



- c. 单击“确定”将步骤添加到测试。
UFT 为洞察对象向测试添加了一个新步骤。

5. 运行测试并查看运行结果。

- a. 选择“录制”>“录制和运行设置”。将打开“录制和运行设置”对话框。
- b. 在“Windows 应用程序”选项卡中，选择“仅在以下应用程序上录制和运行:”选项。
- c. 在“仅在以下应用程序上录制和运行:”选项下，选择“以下指定的应用程序”选项。应该保存了上一个运行会话的应用程序详细信息。

备注: 如果需要重新输入应用程序详细信息，请参阅[练习 4a:运行测试 \(第 62 页\)](#) 了解必要的详细信息。

- d. 单击“确定”保存更改并应用设置。
- e. 在工具栏中，单击“运行”按钮 。将打开“运行”对话框。
- f. 在“运行”对话框的“结果位置”选项卡中，选择“新建运行结果文件夹”。接受默认文件夹名。
- g. 单击“运行”以启动测试运行。

重要事项: 请确保执行测试运行前航班预订应用程序已关闭。

UFT 将打开航班预订应用程序并按顺序运行测试步骤。当 UFT 运行包含洞察对象的步骤时，将使用洞察对象标识机制查找对象。

测试运行完成后，将打开运行结果。

- h. 在“测试流”中，选择“Promotion Order.Click”节点。运行结果将显示步骤的摘要。
- i. 您可以在步骤详细信息中看到测试运行期间 UFT 使用洞察对象标识所标识的对象：

步骤详细信息

步骤

Promotion Object.Click

执行时间

2015-09-25 11:34:43



测试对象

 InsightObject: "Promotion Object"

存储库

Local

对象路径

WpfWindow("HP MyFlight Sample Application").InsightObject("Promotion Object")

第 4 部分: 创建和运行自动 API 测试

除了测试应用程序的用户界面以外，还应测试非 GUI (服务) 层，以确保运行应用程序的 API 正常工作。使用 UFT，您可以创建为您执行测试的自动 API 测试。

创建 API 测试时，您将创建总体测试流，该测试流包含单独测试应用程序的 API 流程的测试步骤。您提供这些测试步骤的输入和检查点属性，然后 UFT 在应用程序中运行测试。测试运行结束后，您可以查看结果，了解您应用程序的运行情况。

在此部分中，您将学习如何为各种不同类型的应用程序和服务创建 API 测试。

此部分包含以下内容：

· 课程 1: 创建 API 测试	125
· 课程 2: 创建简单的 API 测试步骤	126
· 课程 3: 使用标准活动创建 API 测试步骤	130
· 课程 4: 参数化 API 测试步骤	135
· 课程 5: 运行 API 测试	154
· 课程 6: 创建并运行 Web Service 的 API 测试	158
· 课程 7: 创建并运行 REST 服务的 API 测试	170
· 课程 8: 创建并运行 Web 应用程序服务 (WADL) 的 API 测试	183

课程 1: 创建 API 测试

在创建应用程序服务层的测试内容之前，必须首先创建测试并创建测试结构。

1. 在 UFT 工具栏上，单击“新建”按钮向下箭头  并选择“新建测试”。
2. 在“新建测试”对话框中，选择“API 测试”。
3. 为测试输入以下详细信息：

名称: Basic

位置: 默认情况下，UFT 将文档保存在 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 中。对于本课程，您无需修改此路径。

4. 单击创建。

在带有测试流选项卡的画布中将打开一个空测试（名为 **Basic**）。测试流画布内部是一个空的测试流。

此测试在解决方案浏览器窗格中也显示为 **Untitled Solution** 节点的子节点。（这是创建测试但未命名解决方案时使用的常规解决方案名称。）

现在您可以随时开始设计 API 测试。请继续[课程 2: 创建简单的 API 测试步骤 \(第 126 页\)](#) 学习如何创建 API 测试步骤。

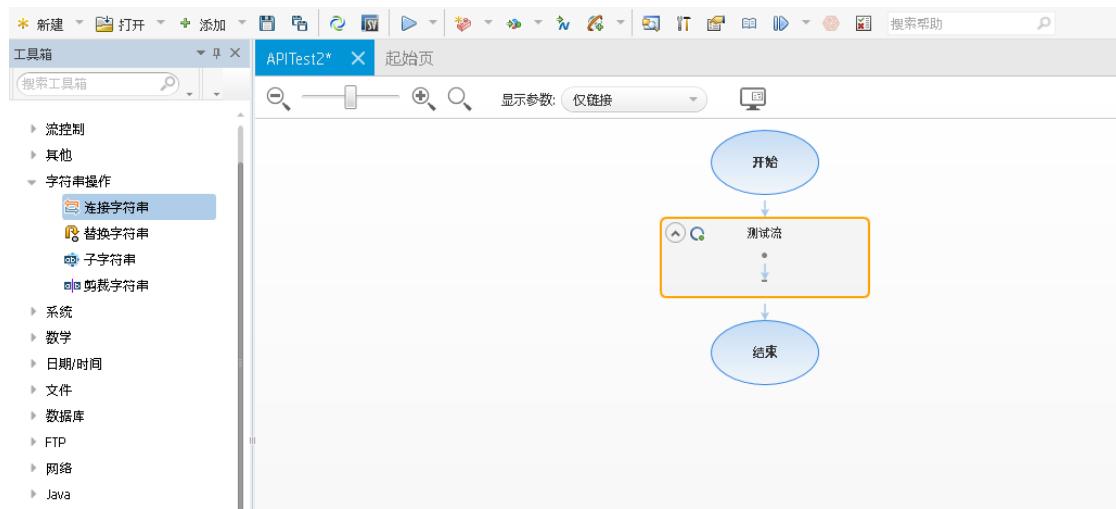
课程 2: 创建简单的 API 测试步骤

在 UFT API 测试中，创建测试的过程是可见的。测试步骤显示在可显示整个主测试流的画布上。

创建实际测试步骤包含两个主要部分：

1. 通过将相应的活动拖到画布中的测试流来创建测试步骤。

“工具箱”窗格包含可在测试中使用的所有活动。从画布中包含的活动列表中，可以将这些活动拖到画布上并将其添加到测试流中的任何位置：



2. 添加步骤属性 (步骤的输入和检查点)

将测试步骤拖到画布上后，每次选择画布中的步骤时，步骤的属性将显示在“属性”窗格中：



每个步骤都具有两个不同的属性类型：

输入	这些输入由 UFT 用于设置活动运行步骤所需的值。
检查点	步骤运行后，将这些值与实际值进行比较，以确认步骤是否正常运行。 运行测试时，检查点属性是可选的。

在本课程中，您将使用这些基本功能创建基本测试。

1. 启动 UFT 打开 Book Flights 测试。

- a. 如果 UFT 当前未打开，请按[为所有测试创建解决方案 \(第 20 页\)](#) 中所述将其打开。
 - b. 在起始页上的“最近使用的测试/组件”区域中，单击“基本”。
- “基本”测试在文档窗格中打开。

2. 向画布添加一个活动并定义其属性。

- a. 在工具栏中, 单击“工具箱”按钮 。“工具箱”窗格将打开并显示活动的不同类别。
- b. 在“工具箱”窗格中, 展开“字符串操作”节点。
- c. 从“字符串操作”活动列表中, 将“连接字符串”活动拖到画布上。
一个新块将添加到名为“连接字符串”的画布上, 并且“属性”窗格打开到测试步骤属性。
- d. 在“属性”窗格中, 选择“输入/检查点”选项卡 .
- e. 在“输入/检查点”选项卡的“输入”部分(顶部)中, 为步骤属性输入以下值:
 - A: Hello (后跟一个空格)
 - B: World.
 - 检查点: Hello world.

3. 向画布添加另一个活动并定义其属性。

除定义输入和检查点属性之外, 还可以定义测试步骤的其他属性。

- a. 在“工具箱”窗格中, 展开“字符串操作”节点。
- b. 从“字符串操作”活动列表中, 双击“替换字符串”活动以将其添加到画布上。“输入/检查点”选项卡在“属性”窗格中打开, 其中显示步骤的输入和检查点属性。
- c. 在“属性”窗格中, 打开“常规”选项卡 .
- d. 在“名称”属性行中, 将 Replace String 更改为 Change Text 并按“Enter”。画布中步骤的名称将更改为 Change Text。
- e. 打开“输入/检查点”选项卡。
- f. 在“输入/检查点”选项卡中, 为属性输入以下值:
 - 源字符串: Hello world.
 - 搜索字符串: Hello
 - 替换字符串: Goodbye
 - 区分大小写: False

4. 将检查点属性添加到 Change Text 步骤。

- a. 在“输入/检查点”选项卡的“检查点”部分的“结果”行中, 选中“验证”列中的复选框。这样可以启用此步骤的检查点
- b. 在“预期值”列中, 输入预期结果: Goodbye world.

5. 保存测试。

选择文件 > 保存。

6. 运行测试并查看结果。

- a. 在工具栏中, 单击“运行”按钮 。将打开“运行”对话框。
- b. 在“运行”对话框中, 单击选项展开对话框。
- c. 在“结果位置”选项卡中, 选择“临时运行结果文件夹”选项。
- d. 单击运行以编译并运行测试。

在 UFT 运行测试时，“输出”窗格将显示编译日志。在编译日志中所显示的行中，可以看到您在之前的步骤中输入和检查点属性值：



The screenshot shows the UFT Output window titled "User Logger". The log entries are as follows:

```
输出
User Logger
步骤“连接字符串7”：输出参数“结果”，值：“”
步骤“连接字符串7”已成功结束
步骤“连接字符串8”：步骤“连接字符串8”已启动
步骤“连接字符串8”：输入参数“前缀”，值：“Hello。”
步骤“连接字符串8”：输入参数“后缀”，值：“World。”
步骤“连接字符串8”：输出参数“结果”，值：“Hello World。”
步骤“连接字符串8”已成功结束
步骤“迭代 1”已成功结束
步骤“测试流”已成功结束
步骤“结束”：步骤“结束”已启动
步骤“结束”已成功结束
结束
```

测试运行完成后，将打开运行结果。

- e. 在运行结果中，选择“测试流”以显示测试中的步骤。
- f. 在测试步骤树中，选择“连接字符串”步骤节点。运行结果将显示步骤的摘要。
- g. 在步骤摘要区域中，单击“Concatenate Strings.xml”链接。一个单独的选项卡将在文档窗格中打开。
- h. 在新选项卡中，查看“前缀”和“后缀”行。您将看到 UFT 恰好使用您在之前的步骤中提供的输入和输出值来运行步骤。
- i. 在测试步骤树中，选择“Change Text”步骤。
- j. 在步骤摘要区域中，单击“Change Text.xml”链接。
- k. 在新选项卡中，再次查看您用于此步骤的输入属性。
- l. 在 Change Text 步骤下，选择“Checkpoints”步骤。
- m. 在步骤摘要区域中，单击“Checkpoint.xml”链接。
- n. 在新选项卡中，记录测试步骤运行的详细信息。请注意，结果显示通过的检查点（有绿色复选标记）和您在之前的步骤中输入的预期字符串。
- o. 查看完结果后，请关闭显示有运行结果的选项卡。

课程 3:使用标准活动创建 API 测试步骤

创建 API 测试时，您会通过测试了解到应用程序的非 GUI (服务) 层是否正常运行。运行应用程序的不可见流程可能涉及多种操作：调用数据库、调用 Web Service、打开程序、通过 Web 发送消息等等。

为帮助您创建测试，UFT 提供了各种可在设计测试步骤时使用的标准 API 活动。在本课程中，您将使用标准活动创建基本测试。

此课程包含以下内容：

- [UFT API 测试标准活动 - 概述](#) 131
- [练习 3a:使用标准活动创建测试](#) 131

UFT API 测试标准活动 - 概述

创建 API 测试时，为所有测试提供各种标准活动。这些活动可以测试常见的应用程序进程，包括：

- “流控制”活动，例如“等待”、“中断”和“条件”步骤。
- “字符串操作”活动，如“连接字符串”和“替换字符串”
- 使用文件系统执行的流程的“文件”系统活动
- “数据库”活动
- “FTP”活动
- **网络活动**，例如 **HTTP 请求** 和 **SOAP 请求**
- 涉及与 XML 或 JSON 字符串/文件交互的应用程序进程的“JSON”和“XML”活动
- **数学和日期/时间活动**
- “其他”活动，包括“自定义代码”活动、“运行程序”和“结束程序”活动以及“报告”活动。

有多种特定于技术的活动：

- “调用 Java 类”活动，用于测试应用程序中使用的 Java 进程。
- “JMS”(Java 消息服务)活动
- **IBM Websphere MQ** 活动
- “SAP”活动，用于访问存储在 SAP 服务器上的 SAP iDOC 或 RFC
- “负载测试”活动，可帮助您使用 HP LoadRunner 测试运行(转换为 LoadRunner 脚本后)
- “HP 自动测试工具”活动，使您能够从 UFT、QuickTest Professional、Service Test 或 LoadRunner 调用 GUI 测试或操作、API 测试或操作或者 Virtual User Generator 脚本以作为测试的一部分

还有许多可导入测试中的自定义活动，但是这些类型的活动将在后面的课程中详细讨论。

练习 3a: 使用标准活动创建测试

在 [课程 2: 创建简单的 API 测试步骤 \(第 126 页\)](#) 中，您应先熟悉 UFT API 测试用户界面并了解如何使用该界面创建和运行测试步骤。

在本课程中，您将通过该知识使用标准活动创建基本 API 测试。

1. 新建测试。

- a. 在工具栏中，单击“新建”按钮 。将打开“新建测试”对话框。
- b. 在“新建测试”对话框中，选择“API 测试”。
- c. 为测试输入以下详细信息：
 - **名称:** Standard
 - **位置:** C:\%HOMEPATH%\My Documents\Unified Functional Testing

d. **单击创建。**

画布中将打开空白测试和空白测试流。此测试还将在“解决方案浏览器”窗格中显示为“Solution Untitled”解决方案的子节点。

2. 在测试流中创建步骤。

在此步骤中，您将创建应用程序进程的测试，该进程可查找特定字符串、替换它，然后将字符串替换的结果写入文件。

对于此测试，您将需要三个活动：

- 替换字符串
- 创建用于保存结果的文件
- 将结果写入文件

这些活动都与 UFT 的标准活动一起提供。

- a. 如果“工具箱”窗格尚未打开，请通过单击 UFT 窗口左下角的“工具箱”打开它。
- b. 在“工具箱”窗格中，展开“字符串操作”活动节点。
- c. 从“字符串操作”活动中，将“替换字符串”活动拖到画布上。一个新块将添加到画布中的测试流中，并且“属性”窗格将显示“替换字符串”步骤的“输入/检查点”选项卡 。
- d. 在“工具箱”窗格中，展开“文件”活动。
- e. 在“文件”活动中，将以下活动拖到画布上：
 - **文件创建**
 - **文件写入**

两个新块将添加到每个活动的画布上，并且每个活动的块显示有红色警报图标：



将活动添加到画布上后，测试流应如下所示：



3. 输入“替换字符串”活动的属性。

- 在画布中，选择“替换字符串”活动。“输入/检查点”选项卡将在“属性”窗格中打开。
- 在“输入/检查点”选项卡中，输入步骤的输入属性：
 - 源字符串: Hello world.
 - 搜索字符串: Hello
 - 替换字符串: Goodbye

备注: 您无需使用此活动的检查点属性。

4. 输入“文件创建”步骤的属性。

此步骤将允许您创建一个文件，以写入上一步的“替换字符串”操作的结果。在要用于后续步骤的指定目录中创建文件。

- 在画布中，选择“文件创建”步骤。“输入/检查点”选项卡将在“属性”窗格中打开。
- 在文件系统中，打开 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 文件夹。
- 在上一步提及的 Unified Functional Testing 文件夹中，创建名为 **Tutorial_Files** 的文件夹。您将使用此文件夹在下一步中创建文件。

- d. 在“输入/检查点”选项卡中，输入步骤的输入属性：
- **文件夹路径:** C:\%HOMEPATH%\My Documents\Unified Functional Testing\Tutorial_Files

备注: 还可以通过按“浏览”按钮并直接导航到文件夹来进入此文件夹。

- **文件名:** Standard Test Result
- **覆盖:** true

- e. 在“输入/检查点”选项卡的“检查点”部分的“结果”行中，选中“验证”列中的复选框。将默认值保留为原样。

这样，您便可以在运行测试和查看运行结果时看到文件创建步骤是否已成功完成。

5. 输入“文件写入”步骤的属性。

此步骤会将“替换字符串”步骤指定的字符串写入您在“文件创建”步骤中创建的文件中。

- a. 在画布中，选择“文件写入”步骤。“输入/检查点” 选项卡将在“属性”窗格中打开。
- b. 在“输入/检查点”选项卡的“输入”部分中，输入步骤的输入属性：
 - **内容:** Goodbye world.
 - **文件路径:** 您将使用上一步中创建的文件。必须手动输入以下路径:C:\%HOMEPATH%\My Documents\Unified Functional Testing\Tutorial_Files\Standard Test Result
 - 将所有其他属性保留为默认值。
- c. 在“输入/检查点”选项卡的“检查点”部分的“结果”行中，选中“验证”列中的复选框(将默认值保留原样)。

这样，您便可以在运行测试和查看运行结果时看到文件写入操作是否已成功完成。

6. 保存测试。

选择“文件”>“保存”。

7. 运行测试。

- a. 在工具栏中，单击“运行”按钮。将打开“运行”对话框。
- b. 在“运行”对话框的“结果位置”选项卡中，确保“临时运行结果文件夹”仍然处于选中状态。
- c. 单击运行以编译并运行测试。

测试运行完成后，将打开运行结果。

- d. 在文件系统中，打开C:\%HOMEPATH%\My Documents\Unified Functional Testing\Tutorial_Files\Standard Test Result文件。

在此文件中，您应该在文档的第一行中看到字符串“Goodbye world”。这表示UFT在真实应用程序上运行了测试步骤，正如使用UFT附带的标准API活动在任何应用程序运行测试步骤一样。

请注意，您已使用标准活动创建了一组基本应用程序测试进程，请继续[课程4:参数化API测试步骤\(第135页\)](#)学习如何在测试中使用数据以及参数化测试步骤。

课程 4:参数化 API 测试步骤

在[课程 2:创建简单的 API 测试步骤 \(第 126 页\)](#) 和[课程 3:使用标准活动创建 API 测试步骤 \(第 130 页\)](#)中, 您已学会如何通过手动输入所需输入和检查点值来为 API 测试步骤提供值。

但是, 您还可以从其他来源提供输入和检查点值:

- 测试中附带的数据源 (包括 Excel 文件、XML、数据库数据源或本地创建的表)
- 之前步骤的输出
- 所有以上来源的组合: 手动输入所需数据、数据源和之前步骤的输出

使用这些数据源填充步骤输入和输出值可模拟应用程序的工作方式, 因为应用程序进程的输入可以来自数据源和之前的应用程序进程的结果等。

在本课程中, 您将学会如何使用不同的方法参数化 API 测试步骤。

此课程包含以下内容:

· 参数化 API 测试步骤 - 概述	136
· 练习 4a:从数据源参数化测试步骤	136
· 练习 4b:从以前步骤的输出参数化测试步骤	145
· 练习 4c:使用自定义表达式参数化具有多个源的测试	148

参数化 API 测试步骤 - 概述

当在 API 测试中为测试步骤属性提供值时，提供这些值的默认方式是在“属性”窗格中手动输入这些值。

但是，使用此方法并不一定提供应用程序的实际测试。在许多应用程序中，内部 API 过程通常会自动从数据源、其他测试步骤输出或各种此类来源接收信息。

因此，UFT 提供了多种提供（参数化）步骤值的不同方式：

手动输入	当手动输入步骤值时，选择每个步骤，然后为输入和检查点属性值输入或选择相应的值。 此方法未提供简单的测试维护，每次更改应用程序的属性时，都必须更新测试中的每个步骤和每个属性。
链接到数据源	当将步骤属性链接到数据源时，UFT 将在测试运行过程中从数据源获取值，并使用数据源中提供的值。如果您的数据源有多组数据，则可以使用多次迭代运行测试，为输入和检查点提供不同的值，以查看应用程序如何使用不同的数据输入运行。 此方法提供了较简单的测试维护，您仅需更新数据源的值即可，无需更新每个测试步骤。
链接到以前步骤的输出	当将步骤属性链接到以前步骤的输出时，UFT 将从步骤的输出获取值，并在测试运行过程中使用这些值。这样您可以模拟真实应用程序行为，其中应用程序 API 过程的输出通常会将值作为输入传递到其他过程。
链接到多个源	如果应用程序的输入和检查点由多个源提供 - 静态字符串、数据和其他步骤/过程的输出，则可以创建自定义表达式以在测试中执行。UFT 将使用自定义表达式提供值，并在测试运行过程中使用这些值。

练习 4a:从数据源参数化测试步骤

如您在[参数化 API 测试步骤 - 概述 \(第 136 页\)](#) 中所见，为测试步骤提供值的其中一种方式是将步骤属性值链接到数据源。这样，您可以使用多个不同值运行测试步骤，具体取决于数据源的结构。

在 API 测试中，您可以添加多个不同类型的数据源：

- Excel 工作表
- XML 文件或架构
- 数据库
- 本地存储数据表

每种数据源均可添加到测试，使其可用于测试中的所有步骤。

在本课程中，将测试步骤链接到数据源。

1. 启动 UFT 并打开标准测试。

- a. 如果 UFT 当前未打开，请按[为所有测试创建解决方案 \(第 20 页\)](#) 中所述将其打开。
- b. 在起始页上的“最近使用的测试/组件”区域中，单击“标准”。

您在[课程 3:使用标准活动创建 API 测试步骤 \(第 130 页\)](#) 中创建的标准测试将在文档窗格的单独选项卡中打开。它还将在“解决方案浏览器”中显示为“Solution Untitled”解决方案的子节点。

2. 将“连接字符串”步骤添加到测试。

- a. 选择“查看”>“工具箱”以打开“工具箱”窗格。
- b. 在“工具箱”窗格中，展开“字符串操作”节点。
- c. 在“字符串操作”节点中，将“连接字符串”活动拖动到画布，放置在“替换字符串”活动之上。

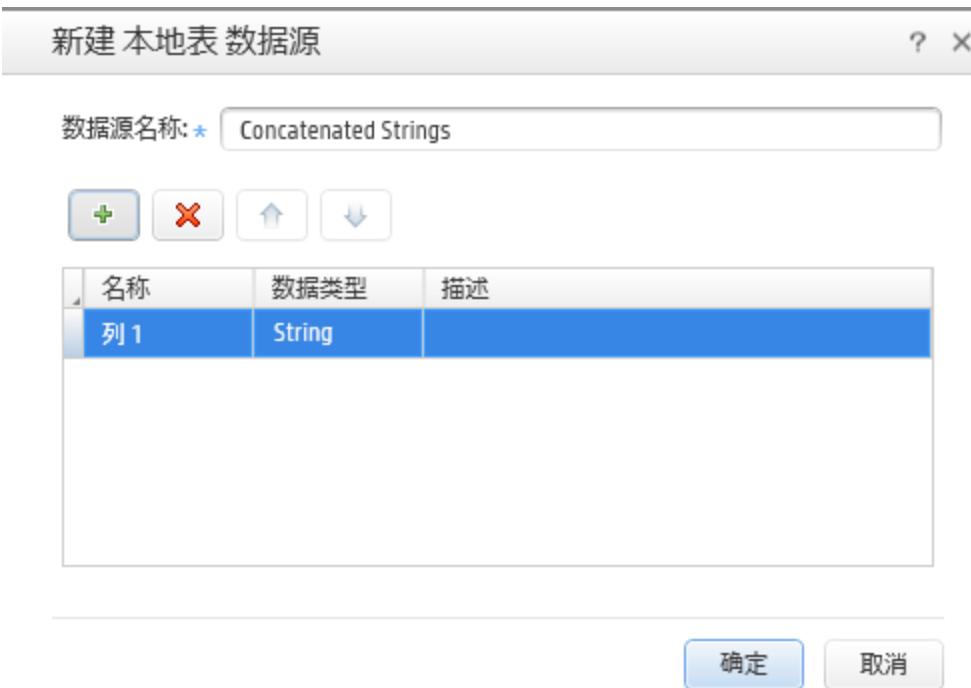
3. 将数据源添加到测试。

- a. 如需要，请选择“查看”>“数据”打开“数据”窗格。
- b. 在“数据”窗格中，单击“新建数据源”按钮 ，然后选择“本地表”。将打开“新建本地表数据源”对话框：



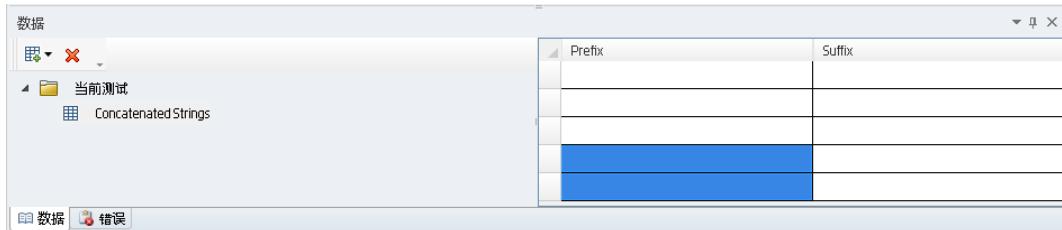
- c. 在“数据源名称”字段中，将表命名为 Concatenated Strings。

- d. 在“数据源”名称字段下方，单击“添加”按钮 。一个新行将添加到表网格中。



- e. 在网格中，在“名称”字段中，输入 Prefix。将“数据类型”保留为“String”。
f. 再次单击“添加”按钮 。第二行将添加到表中。
g. 在网格中，在第二行的“名称”字段中，输入 Suffix。将“数据类型”保留为“String”。
h. 单击“确定”关闭对话框，并将表添加到测试。

该表将添加到测试中，并在“数据”窗格中显示为“当前测试”数据源的子节点：



4. 向数据表添加值。

为了让测试步骤使用测试步骤的值，还必须确保该数据源有可用的数据。对于本地创建和存储的表，必须添加数据。

- 在“数据”窗格中，在“当前测试”节点下方，选择“连接字符串”节点。“数据”窗格更新显示选定数据源的数据。(现在，表中没有任何数据。)
- 在数据网格中，单击“Prefix”列。
- 在“Prefix”列中，输入 Hello。确保 Hello 之后空一格。
- 单击“Suffix”列，并输入 World。

e. 输入其他行:

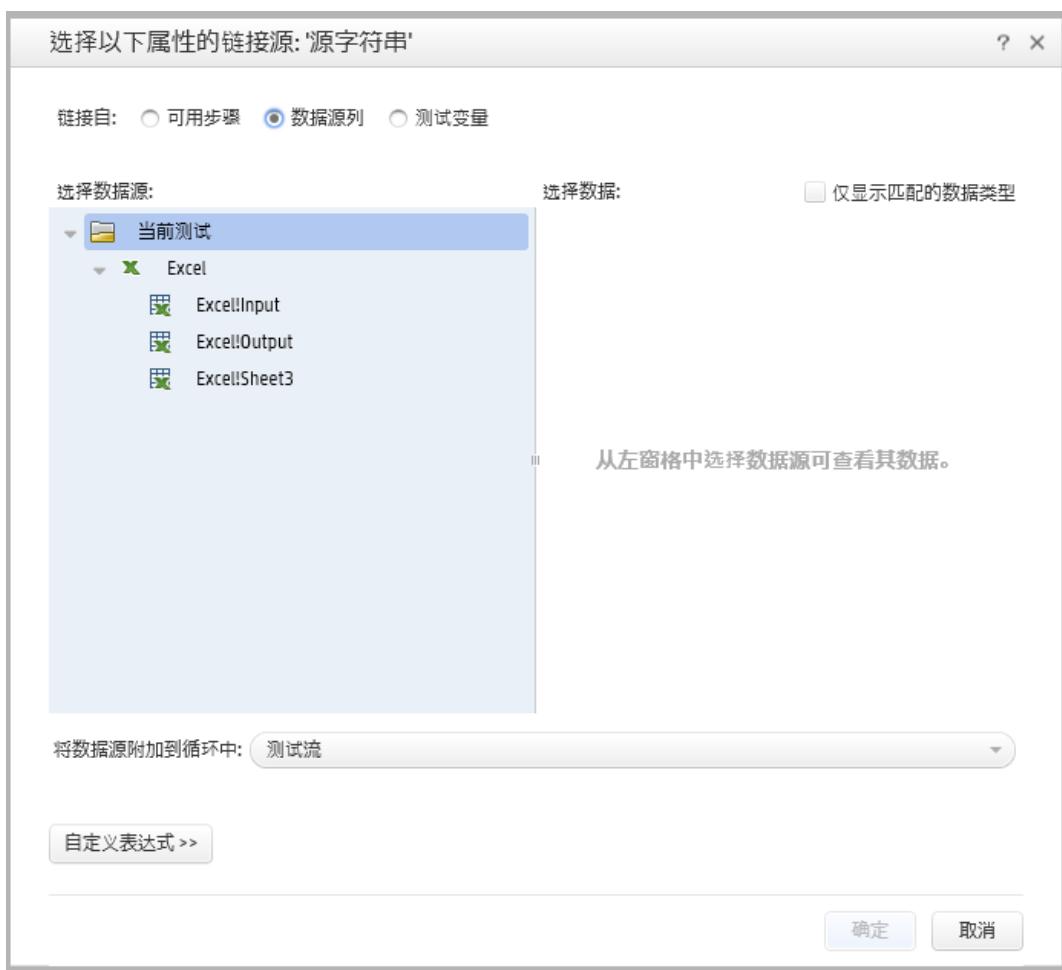
Prefix	Suffix
Welcome	to UFT.
I am running	API tests.

备注: 确保在 A 列中的字符串后面输入空格。

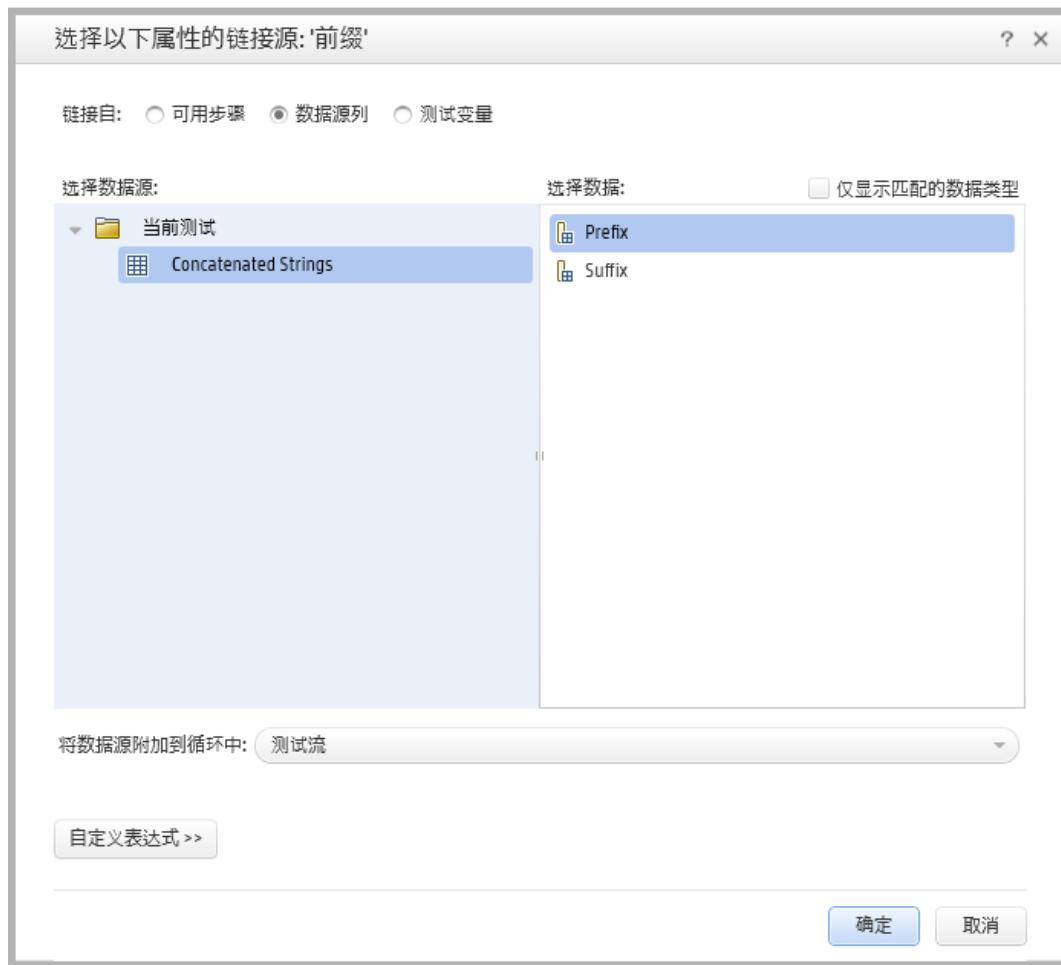
5. 将测试步骤连接到数据源。

为了在运行测试时使用数据源值，需要将测试步骤属性链接到数据源。

- 在画布中，选择“连接字符串”步骤。“输入/检查点”选项卡  将在“属性”窗格中打开。
- 在“输入/检查点”选项卡的“输入”部分中，单击“Prefix”属性的“值”单元格。
- 在“值”单元格中，单击“链接至数据源”按钮 。将打开“选择链接源”对话框。
- 在“选择链接源”对话框中，选择“数据源列”单选按钮。在对话框左侧将显示所有测试的数据源列表：



- e. 在左侧的“选择数据源”窗格中，选择“连接字符串”节点。(这是您在上一步中创建的数据源。)右窗格中将显示数据源参数的列表:

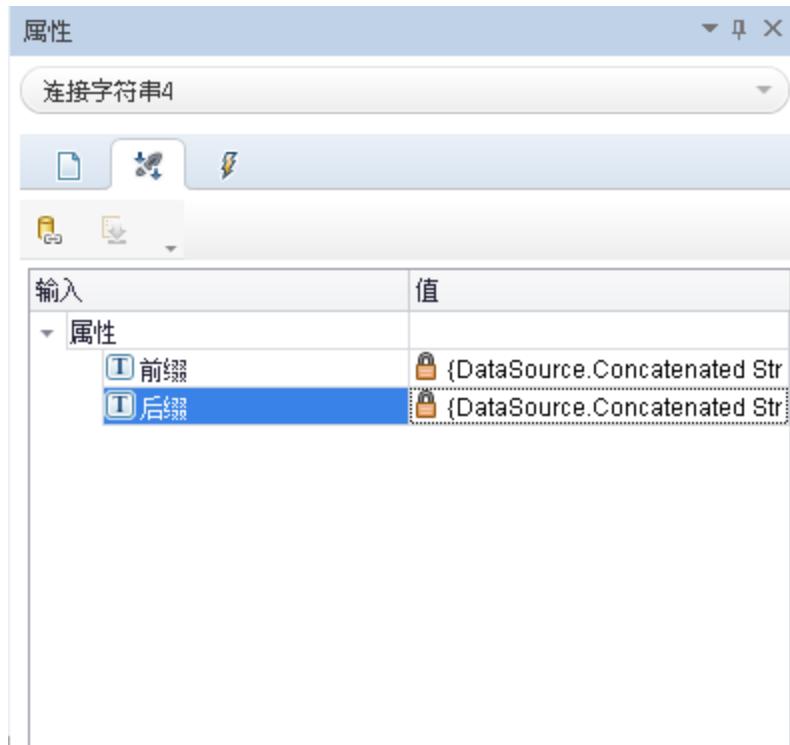


- f. 在右侧的“选择数据”窗格中，选择“Prefix”节点并单击“确定”。
“输入/检查点”选项卡中的“值”列将使用表达式更新，从而显示到数据源的链接。如果将鼠标悬停在“值”列之上，表达式将显示出来:

```
{DataSource.Concatenated Strings.Prefix}
```

- g. 重复上述过程，将“Suffix”属性链接到数据表的“Suffix”列。

完成操作后，可在“输入/检查点”选项卡查看到数据表的链接:

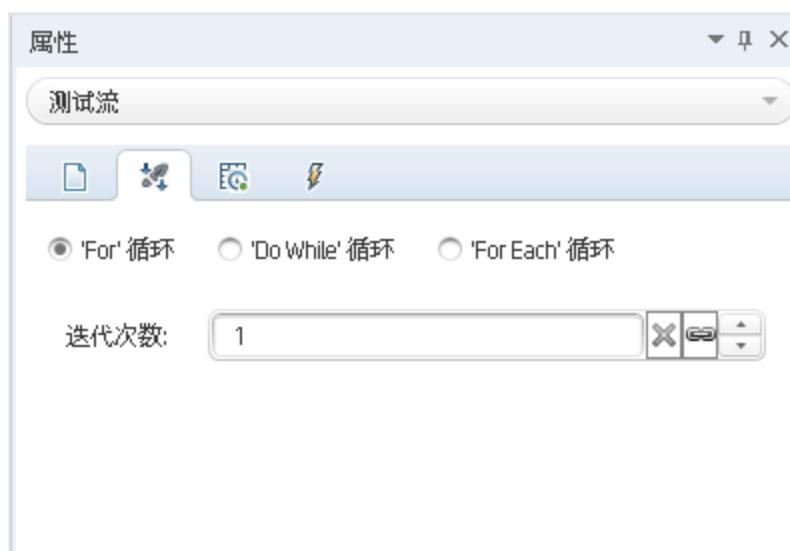


6. 设置测试的迭代次数。

如果现在运行整个测试，则将使用数据表第一行的数据仅运行一次迭代。

因为您输入了三行，所以应运行三次测试迭代，以查看 UFT 如何处理不同的数据。

- 在画布中，选择“测试流”。 “输入”选项卡 将在“属性”窗格中打开:



- 在“输入”选项卡中，选择“'For' 循环”单选按钮。

- c. 在“迭代次数”中，输入“3”作为迭代次数。

现在，当 UFT 运行测试时，它将运行三次测试迭代，每次均使用数据表新行的数据。

备注: 您无需运行与数据源中行数相等的迭代次数。但是，默认情况下，UFT 将从数据表中的第一行开始，每次后续迭代均使用新行的数据运行，直到测试结束。

7. 为数据表设置数据导航策略。

- 在画布中，选择“测试流”。 “输入”选项卡将在“属性”窗格中打开。
- 在“属性”窗格中，选择**数据源**选项卡 。将打开“数据导航”网格。
- 在“数据源”选项卡中，在“数据导航”网格的“数据源名称”列中，选择“连接字符串”数据源并单击“编辑”。将打开“数据导航”对话框：



- 在“数据导航”对话框中，设置以下属性：

起始于	第一行
行	1
移动行数	向前 1 行

结束于	最后一行
在到达最后一行时	环绕

- e. 单击“确定”关闭对话框并更新数据导航策略。

8. 运行测试。

- 在工具栏上，单击“运行”按钮。将打开“运行”对话框。
- 在“运行”对话框中，单击“选项”栏展开对话框。
- 在“结果位置”选项卡中，选择“临时运行结果文件夹”选项。
- 单击“运行”开始运行测试。

UFT 使用“连接字符串”活动输入的数据表中的值顺序运行步骤。当 UFT 运行测试时，您可以在“输出”窗格中看到 UFT 从数据表获取的值：



The screenshot shows the UFT Output window with the title bar "输出" and the tab "User Logger". The window displays a log of test steps:

```
开始
步骤“开始”：步骤“开始”已启动
步骤“开始”已成功结束
步骤“测试流”：步骤“测试流”已启动
步骤“迭代 1”：步骤“迭代 1”已启动
步骤“连接字符串4”：步骤“连接字符串4”已启动
步骤“连接字符串4”：输入参数“前缀”，值：“Hello”
步骤“连接字符串4”：输入参数“后缀”，值：“World”
步骤“连接字符串4”：输出参数“结果”，值：“HelloWorld”
步骤“连接字符串4”已成功结束
步骤“迭代 1”已成功结束
步骤“测试流”已成功结束
步骤“结束”：步骤“结束”已启动
```

A red box highlights the step where the prefix "Hello" and suffix "World" are being input into the "连接字符串4" (String Connect) activity.

测试运行完成后，运行结果将打开。

9. 分析运行结果。

- 在测试结果中，可在“测试流”中查看测试结果树。

请注意此处有三个“摘要”节点:



这与您之前设置的三次迭代相对应。

- 展开“操作迭代: 第 1 行”节点，然后选择“连接字符串”节点。运行结果将显示该步骤的摘要。
- 捕获的数据显示“连接字符串”活动输入值所使用的数据:

Name	Value
类型	HP.ST.Ext.BasicActivities.ConcatenateStringsActivity
步骤 ID	ConcatenateStringsActivity4
消息	已成功连接字符串
前缀	'Hello'
后缀	'World'
结果	'HelloWorld'
名称	'连接字符串4'

使用的值与数据表中第一行的值匹配。

- 对迭代 2 和迭代 3 “连接字符串”活动重复此过程。您会发现作为步骤的捕获数据部分显示的值与数据表中第二行和第三行的值匹配。

10. 保存测试。

选择文件 > 保存。

现在，您已学会如何将测试步骤连接到数据源。请继续[练习 4b:从以前步骤的输出参数化测试步骤 \(第 145 页\)](#) 学习如何将步骤属性与以前步骤的输出相链接。

练习 4b:从以前步骤的输出参数化测试步骤

在[练习 4a:从数据源参数化测试步骤 \(第 136 页\)](#) 中，您学会了如何将选定步骤的属性值链接到数据源。

但是，除了从数据提供属性值，您还可以从以前步骤的输出获取属性值。在本课程中，您将学习如何使用以前步骤的输出链接步骤值。

1. 启动 UFT 并打开标准测试。

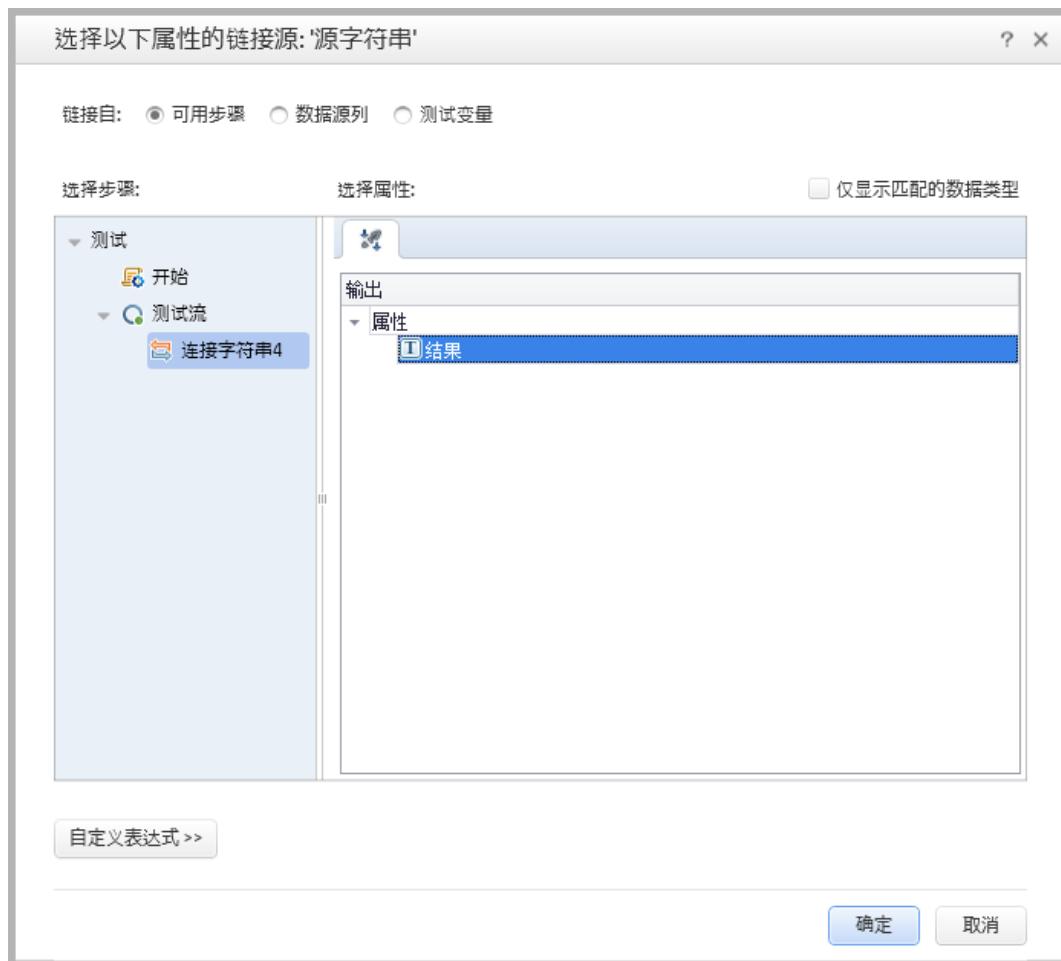
- a. 如果 UFT 当前未打开，请按[为所有测试创建解决方案 \(第 20 页\)](#) 中所述将其打开。
- b. 在起始页上的“最近使用的测试/组件”区域中，单击“标准”。

您在[课程 3:使用标准活动创建 API 测试步骤 \(第 130 页\)](#) 中创建的标准测试将在文档窗格的单独选项卡中打开。还会作为“Solution Untitled”解决方案的子节点在解决方案浏览器中显示。

2. 将“替换字符串”步骤的属性链接到“连接字符串”测试。

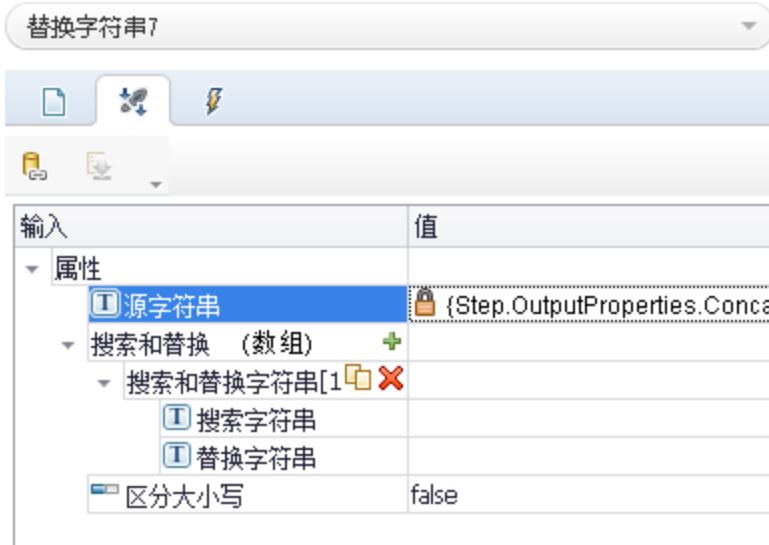
- a. 在画布中，选择“替换字符串”步骤。“输入/检查点” 选项卡将在“属性”窗格中打开。
- b. 在“输入/检查点”选项卡的“输入”区域中，选择“源字符串”行。
- c. 在“源字符串”行的“值”单元格中，单击“链接至数据源”按钮。将打开“选择链接源”对话框。

- d. 在“选择链接源”对话框中，选择“可用步骤”单选按钮。然后，将在对话框左侧显示可用步骤的列表：



- e. 在左侧的“选择步骤”窗格中，选择“连接字符串”。此时将显示“连接字符串”活动的输出属性列表。
f. 在“选择属性”窗格(右侧)中，选择“结果”行。
g. 单击“确定”链接属性并关闭对话框。

“属性”窗格中“源字符串”行的“值”列此时将更新，从而显示到以前步骤的输出的链接:



如果将鼠标悬停在“值”列之上，您可以看到完整的链接说明:

```
{Step.OutputProperties.ConcatenateStringsActivity7.Result}
```

3. 运行测试并查看链接步骤的结果。

- a. 在画布中，选择“测试流”。 “输入”选项卡 将在“属性”窗格中打开。
- b. 在“输入”选项卡中，确保选中“'For' 循环”选项。
- c. 在“迭代次数”中，输入 1。
- d. 在工具栏中，单击“运行”按钮 。将打开“运行”对话框。
- e. 在“运行”对话框中，在“结果位置”选项卡中，确保选中“临时运行结果文件夹”选项。
- f. 单击“运行”以启动测试运行。
UFT 使用“替换字符串”活动输入的“连接字符串”步骤输出运行测试步骤。
测试运行完成后，将打开运行结果。
- g. 在运行结果中，显示“测试流”。
- h. 在“概要”节点下，查找“替换字符串”节点并将其选择为可见。运行结果将显示“替换字符串”步骤的详细信息。
- i. 在步骤详细信息中，请注意此测试运行所使用的源字符串:

Step Properties

Name	
类型	HP.ST.Ext.BasicActivities.ReplaceStringActivity
步骤 ID	ReplaceStringActivity5
源字符串	'HelloWorld' ←
搜索字符串 1	'Hello'
替换字符串 1	'Goodbye'

此源字符串是“连接字符串”活动的输出，在[练习 4a:从数据源参数化测试步骤 \(第 136 页\)](#)中输入。

4. 保存测试。

选择文件 > 保存。

现在，您已学会如何将测试步骤彼此相链接，请继续[练习 4c:使用自定义表达式参数化具有多个源的测试 \(第 148 页\)](#)学习如何通过使用手动输入值组合、链接到数据源以及链接到以前步骤的输出来参数化步骤。

练习 4c:使用自定义表达式参数化具有多个源的测试

在之前的练习中，您已学会如何将测试步骤属性值链接到数据源或上一步的输出。

但是，有时步骤值来自各种位置：手动输入静态值、数据源和/或上一步的输出。在这些情况下，您可以创建自定义表达式以链接到多个源。

在此练习中，您将创建自定义表达式以将字符串替换操作的结果写入文件，该操作使用测试的数据表中的数据、手动输入的静态文本字符串和其他测试步骤的输出。

1. 启动 UFT 并打开标准测试。

- 如果 UFT 当前未打开，请按[为所有测试创建解决方案 \(第 20 页\)](#)中所述将其打开。
- 在起始页上的“最近使用的测试/组件”区域中，单击“标准”。

您在[课程 3:使用标准活动创建 API 测试步骤 \(第 130 页\)](#)中创建的标准测试将在文档窗格的单独选项卡中打开。它还将在“解决方案浏览器”中显示为“Solution Untitled”解决方案的子节点。

2. 链接“文件写入”测试步骤的输入属性。

在[课程 3:使用标准活动创建 API 测试步骤 \(第 130 页\)](#)中，您使用三种活动创建了测试：

- 替换字符串，选择一个字符串并使用另一个字符串替换该字符串的一部分
- 文件创建，创建一个用于写入替换字符串的文件
- 文件写入，写入替换的字符串

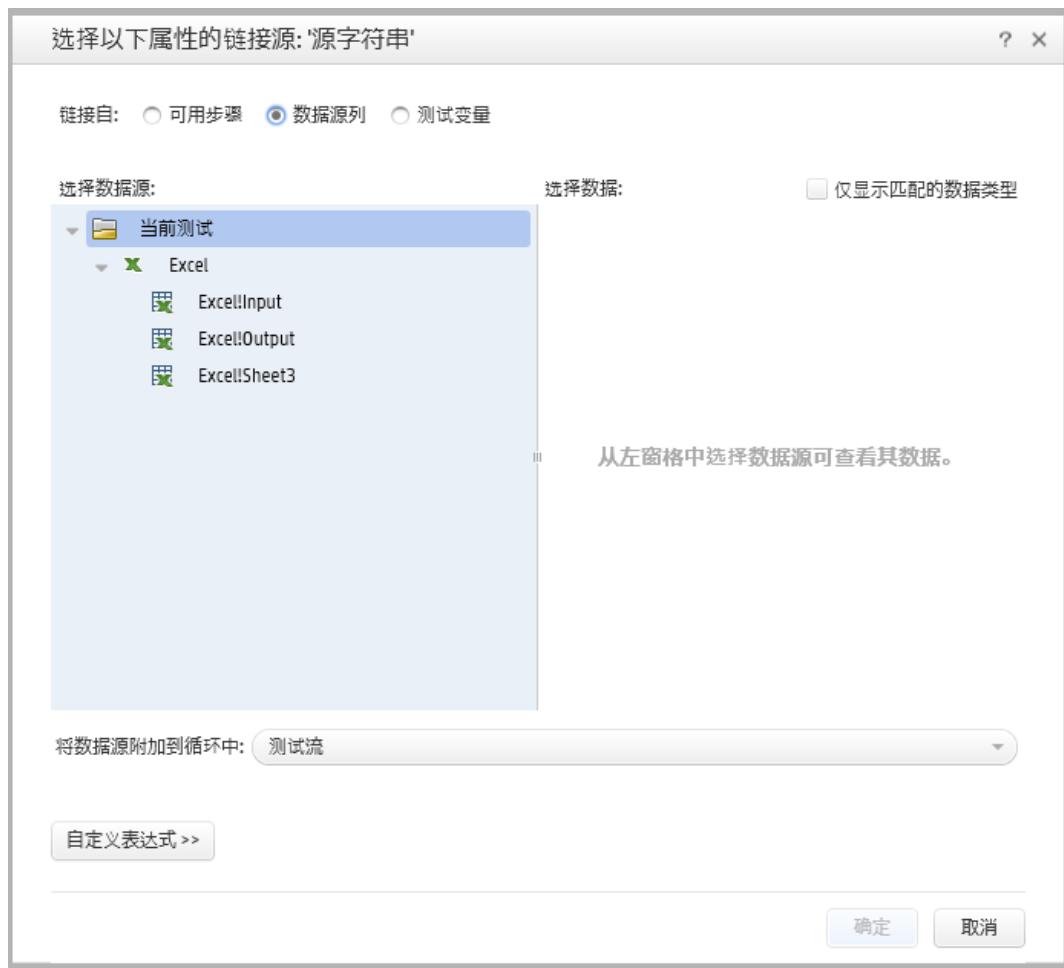
备注: 您已在练习 4a:从数据源参数化测试步骤 (第 136 页) 中添加了“连接字符串”步骤。

在此步骤和后续步骤中，您将使用“文件写入”活动。

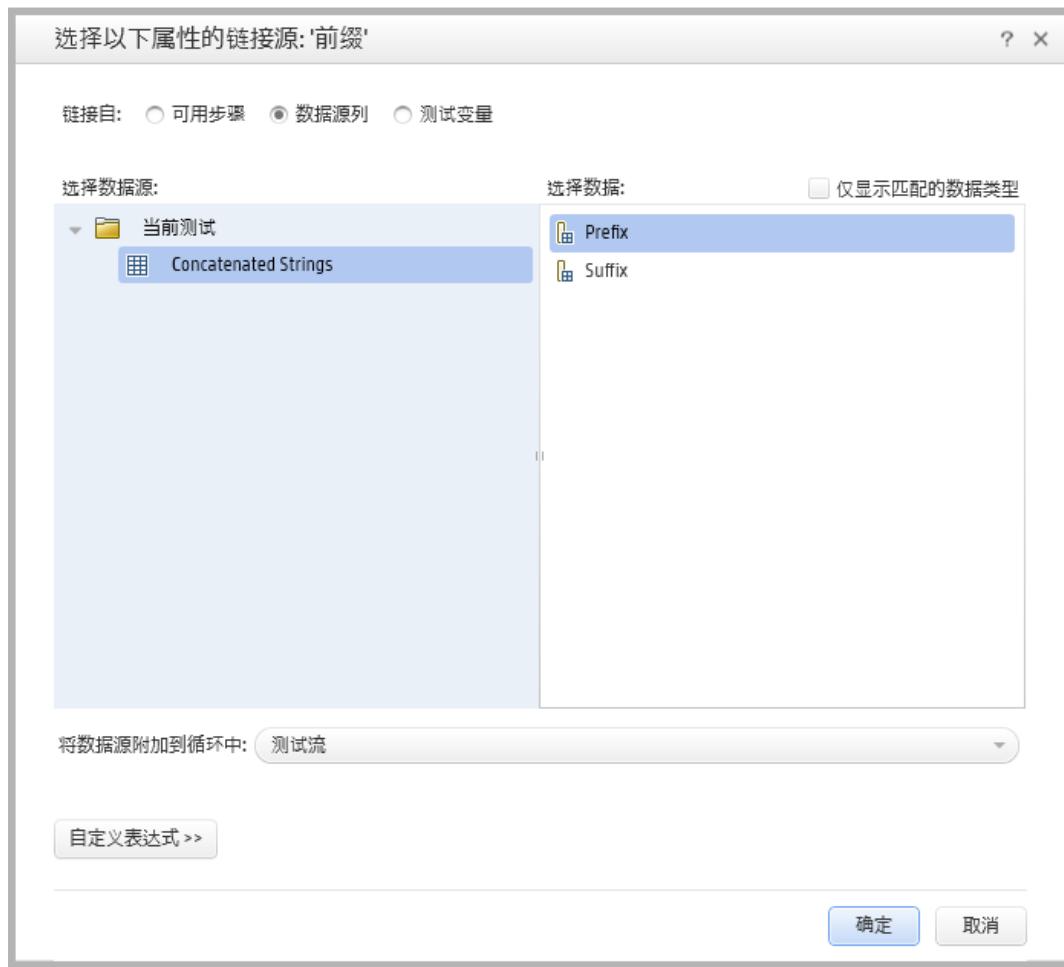
- a. 在画布中，选择“文件写入”步骤。“输入/检查点”选项卡  将在“属性”窗格中打开。
- b. 在“输入/检查点”选项卡的“输入”部分中，选择“内容”行。
- c. 在“内容”行的“值”列中，单击“链接至数据源”按钮。将打开“选择链接源”对话框。

3. 从数据表为内容属性值创建自定义表达式的第一个部分

- a. 在“选择链接源”对话框中，选择“数据源列”选项。“选择链接源”对话框将显示所有测试的数据源的列表：



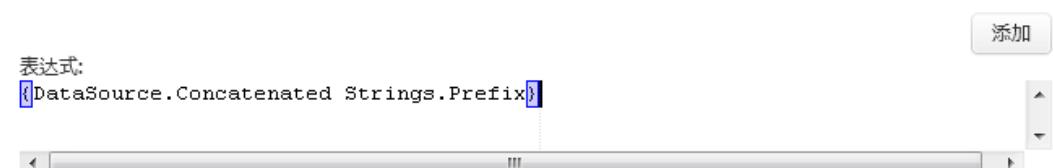
- b. 在“选择数据源”窗格(左侧)中，选择“连接字符串”节点。对话框将显示“连接字符串”数据表中的数据参数的列表:



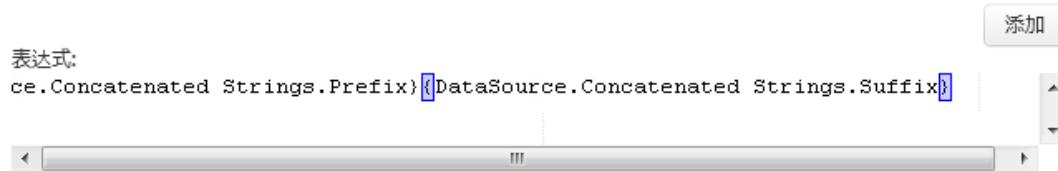
- c. 在“选择数据”窗格(右侧)中，选择“前缀”节点。
d. 在对话框底部，单击“自定义表达式”按钮。对话框将展开以显示“表达式”区域:



- e. 在“表达式”区域上方，当“前缀”节点处于选中状态时，单击“添加”按钮。UFT 会将“前缀”参数的表达式添加到以下表达式:



- f. 在“选择数据”窗格中，选择“后缀”节点。
- g. 在“表达式”区域中，再次单击“添加”按钮。UFT 会将“后缀”参数的表达式添加到以下表达式：



4. 通过手动输入字符串添加自定义表达式的中间部分

对于自定义表达式的中间部分，您将手动添加静态文本字符串。

- a. 在“选择链接”对话框中的“表达式”区域中，在`{DataSource.Concatenated Strings.Suffix}`表达式后面输入一个空格。
- b. 输入文本 was replaced by，后跟另一个空格。

备注: 输入此字符串后不要单击“添加”。如果单击“添加”按钮，UFT 将添加在对话框顶部的窗格中选择的任何元素。

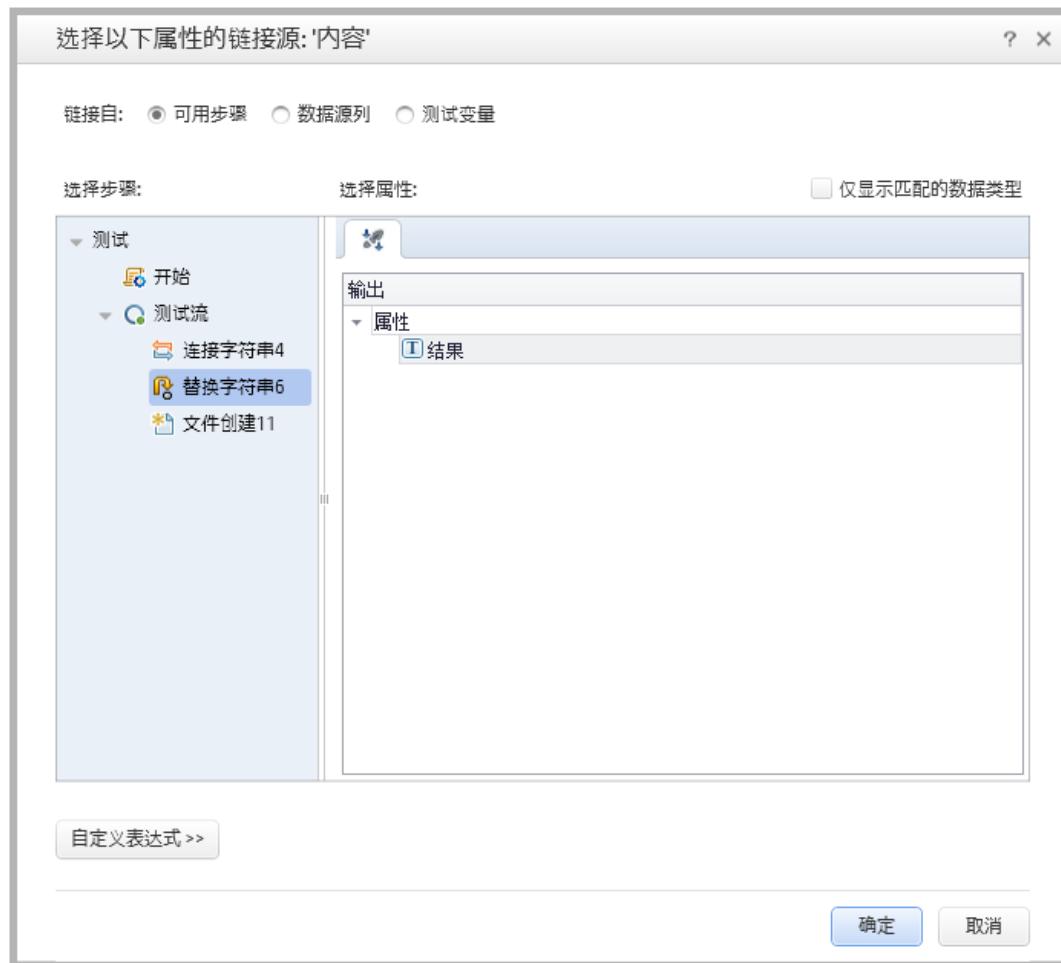
现在，在输入静态文本字符串之后，自定义表达式应如下所示：

```
{DataSource.Concatenated Strings.Prefix}{DataSource.Concatenated Strings.Suffix}  
was replaced by
```

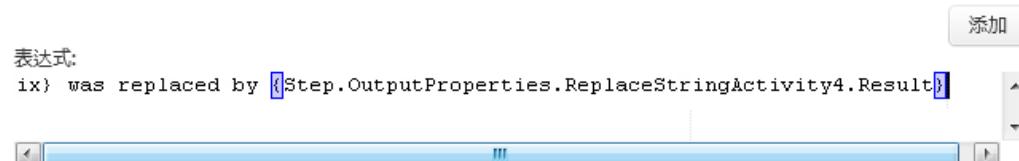
5. 通过将其链接到上一步的输入来添加自定义表达式的最后部分。

对于自定义表达式的最后一个元素，您将链接到上一步的输出。

- a. 在“选择链接源”对话框的顶部，选择“可用步骤”选项。该对话框将显示所有之前步骤的列表：



- b. 在“选择步骤”窗格(左侧)中，选择“替换字符串”节点。将显示可用属性的列表:
- c. 在“选择属性”窗格(右侧)中，选择“结果”行。
- d. 在“表达式”区域中，单击“添加”。将向上一个表达式中再添加一个表达式，以反映“替换字符串”步骤输出的链接:



- e. 单击“确定”关闭对话框，并添加此表达式作为“文件写入”步骤的“内容”属性的值。
“内容”属性的“值”列现在将显示更新后的表达式:

```
{DataSource.Concatenated Strings.Prefix}{DataSource.Concatenated Strings.Suffix}
was replaced by {Step.OutputProperties.ReplaceStringActivity4.Result}
```

6. 设置测试运行的迭代次数。

- a. 在画布中，选择“测试流”。“输入”选项卡  将在“属性”窗格中打开。
- b. 在“输入”选项卡中，确保选中“'For' 循环”选项。
- c. 在“迭代次数”中，输入 1。

7. 运行测试并查看运行结果。

- a. 在工具栏中，单击“运行”按钮 。将打开“运行”对话框。
- b. 在“运行”对话框的“结果位置”选项卡中，确保选中“临时运行结果”选项。
- c. 单击“运行”以启动测试运行。
UFT 将运行步骤，从您在自定义表达式中创建的链接获取“文件写入”步骤的“内容”属性的值。
测试运行完成后，将打开运行结果。
- d. 在运行结果中，显示“测试流”。
- e. 在“测试流”中，找到“文件写入”节点。运行结果将显示步骤相关信息的摘要。
- f. 在步骤摘要中，单击“Write to File.xml”链接。将打开单独的选项卡，其中显示步骤捕获的数据。
- g. 在单独的选项卡中，记录用于此测试运行的“内容”属性值：

Name	
类型	HP.ST.Ext.BasicActivities.FileWrite
步骤 ID	FileWriteActivity10
内容	'Goodbyeworld'
文件路径	'True'
编码	'ASCII'
模式	'附加到现有文件'
附加换行符	False

8. 保存测试。

单击“保存” 。

课程 5:运行 API 测试

在[练习 3a:使用标准活动创建测试 \(第 131 页\)](#) 中，您使用标准活动创建了基本 API 测试。因此您可以运行已创建的此测试了。

在本课程中，您将学习如何运行测试以及查看运行结果。

此课程包含以下内容：

- [练习 5a:运行测试](#) 155
- [练习 5b:导航运行结果](#) 156
- [练习 5c: 分析运行结果](#) 157

练习 5a:运行测试

在[练习 3a:使用标准活动创建测试 \(第 131 页\)](#)中，您使用标准 API 测试活动创建了测试。然后，在[课程 4:参数化 API 测试步骤 \(第 135 页\)](#)中，使用一些不同方法对测试进行了参数化。

在本课程中，您将学习如何为运行测试准备 UFT 以及如何运行 API 测试。

1. 启动 UFT 并打开 Book Flight 解决方案。

- 如[为所有测试创建解决方案 \(第 20 页\)](#)中所述打开 UFT。
- 在起始页上的“最近使用的测试”区域，单击“Flight Reservation Application”。

将打开 Flight Reservation Application 解决方案，其中包含您在[练习 3a:使用标准活动创建测试 \(第 131 页\)](#)中创建的 Book Flights 测试。

2. 设置测试的运行模式。

运行测试之前，必须告知 UFT 您要如何运行该测试。您可以在快速运行测试的“发布”模式中运行测试，或者，当 UFT 需要在测试运行之前加载调试工具时，在测试运行较慢的“调试”模式中运行测试。

- 选择“工具”>“选项”。将打开“选项”对话框。
- 在“选项”对话框中，选择“API 测试”选项卡。
- 在“API 测试”选项卡中，选择“常规”节点。
- 在“常规”窗格中，选择“以调试模式运行测试”选项。

3. 开始运行测试。

- 在工具栏中，单击“运行”按钮 。将打开“运行”对话框。
- 在“运行”对话框中，单击“选项”栏展开对话框。
- 在“选项”区域中，单击“结果位置”选项卡。这样，您可以指定保存测试结果的位置。
- 在“结果位置”中，选择“新运行结果文件夹”选项。接受默认文件夹名。
- 单击运行以关闭“运行”对话框并开始运行测试。

在 UFT 运行测试时，它会在“输出”窗格中提供所执行步骤的日志，其中包含输入和输出参数，以及任何检查点运行的结果：



日志还会对测试运行过程中发生的任何错误进行报告。您可以返回到相关步骤解决这些错误。

完成测试运行后，在文档窗格中，运行结果将在单独的选项卡中打开。请继续[练习 5b:导航运行结果 \(第 156 页\)](#)学习更多有关运行结果的内容。

练习 5b:导航运行结果

在[练习 5a:运行测试 \(第 155 页\)](#)中，您运行了所创建的标准测试。完成测试运行后，运行结果将自动显示此测试运行的结果。

备注: 默认情况下，运行结果显示在基于 HTML 的报告中。还可以选择将运行结果显示在“选项”对话框“运行会话”窗格的 Run Results Viewer 中（“工具”>“选项”>“常规”选项卡>“运行会话”节点）。本教程的此课程以基于 HTML 的报告为基础。

当打开运行结果时，将显示以下内容：

The screenshot shows the UFT software interface with the title bar "UFT 报告: APITest18 - Result2". Below the title bar, there are three tabs: "错误列表" (Error List), "测试流" (Test Flow), and "详细信息" (Detailed Information). A search bar with the placeholder "搜索..." is located between the tabs. On the right side of the interface, there is a large panel titled "详细信息" (Detailed Information) which currently displays the message "无错误/警告" (No errors/warnings).

最初，运行结果会显示：

测试流	结果在树中的图形表示，根据测试中的步骤进行组织。可以指示 UFT 在每次运行中使用不同数据集多次运行测试。每次运行称为一个迭代，每个迭代都有编号。
错误列表	所有错误和警告的列表，以列表形式显示。
步骤摘要信息	高级别的结果概述报告，包括测试的常规信息、通过或失败的步骤、每个测试步骤的详细信息。摘要还包括打开该测试步骤所捕获数据的链接。

测试运行之所以会成功，是因为 UFT 能够根据您创建的步骤和提供的属性正确执行所有步骤。如果发生错误且测试未成功运行，则该错误将在“输出”窗格的日志中列出。在这种情况下，请返回并确保完全按照此教程中所述配置了步骤。

现在，您已了解了运行结果所显示的内容。请继续[练习 5c: 分析运行结果 \(第 157 页\)](#)学习有关运行结果的详细信息。

练习 5c: 分析运行结果

在此练习中，您将检查在运行练习 5a:运行测试（第 155 页）中的测试时 UFT 执行的步骤。

1. 查看特定步骤的结果。

- a. 在测试流的结果树中，查找“测试流”>“摘要”>“Iteration 1”节点，以查看此测试中所有步骤的执行情况。
- b. 在“Iteration 1”节点下，选择“替换字符串”节点：



现在，运行结果显示以下信息：

- 测试流，且突出显示步骤
- 测试步骤摘要，显示已突出显示步骤的详细信息
- 查看选定步骤的捕获数据的链接。

2. 关闭运行结果。

在文档窗格中，关闭包含运行结果的选项卡。

课程 6:创建并运行 Web Service 的 API 测试

在[课程 3:使用标准活动创建 API 测试步骤 \(第 130 页\)](#) 中, 您学习了如何使用标准 API 活动创建测试。

但是, 标准活动可能会与您应用程序执行的进程不匹配。在这种情况下, 您需要使用导入到 UFT 中的自定义活动。使用较广泛的服务活动类型之一就是 Web Service。在 UFT 中, 您将服务及其方法导入到 UFT 中, 使其可用于测试。

在本课程中, 您将学习如何创建 Web Service 测试并运行测试。

此课程包含以下内容:

- [练习 6a: 创建 Web Service 测试](#) 159
- [练习 6b: 导入 Web Service](#) 159
- [练习 6c:构建并参数化 Web Service 测试](#) 161
- [练习 6d:运行 Web Service 测试](#) 167

练习 6a: 创建 Web Service 测试

在[为所有测试创建解决方案 \(第 20 页\)](#) 中，您为航班预订应用程序测试创建了解决方案。在[创建和运行自动 GUI 测试 \(第 23 页\)](#) 中，您创建了各种测试航班预订用户界面性能的 GUI 测试。

在本练习中，您将创建测试航班预订应用程序的服务 (API) 层所必需的第一个 API 测试。

1. 启动 UFT 并打开 Flight Reservation Application 解决方案。
 - a. 如[为所有测试创建解决方案 \(第 20 页\)](#) 中所述打开 UFT。
 - b. 在起始页上的“最近使用的解决方案”区域，单击“Flight Reservation Application”。

将打开 Flight Reservation Application 解决方案，其中包含您在[课程 1: 创建 GUI 测试和操作 \(第 24 页\)](#) 中创建的 Book Flights 测试。
2. 将新 API 测试添加到解决方案。
 - a. 在工具栏上，选择“添加”下拉箭头  添加 并选择“添加新测试”。将打开“添加新测试”对话框。
 - b. 在“添加新测试”对话框中，选择“API 测试”。
 - c. 在“名称”字段中，将测试命名为 Book Flights Web Service。
 - d. 在“位置”字段中，单击“浏览”按钮并导航到 **C:\%HOMEPATH%\Unified Functional Testing** 文件夹。
 - e. 单击“添加”创建测试，并将其添加到解决方案。

Book Flights Web Service 测试将作为一个独立节点添加到 Flight Reservation Application 解决方案中，并在文档窗格的单独选项卡中打开。

备注: 解决方案会自动保存。

现在，您已创建了测试。您随时可以使用 Web Service 及其方法。请继续[练习 6b: 导入 Web Service \(第 159 页\)](#) 学习如何将 Web Service 导入到测试中。

练习 6b: 导入 Web Service

测试 Web Service 之前，必须将服务描述及其方法导入到 UFT 中。一般而言，服务描述存储在 WSDL (Web Service 描述语言) 文件中。此文件定义服务的元数据，以及服务的操作/方法。然后，UFT 读取此 WSDL 文件，并在“工具箱”窗格中如同创建活动那样创建服务的方法。

在本练习中，您会将航班预订应用程序的服务 WSDL 文件导入到 UFT 中。

1. 启动 Flights API 应用程序。

启动 HP Flights Service API 应用程序，如[浏览航班预订应用程序 \(第 18 页\)](#) 中所述。

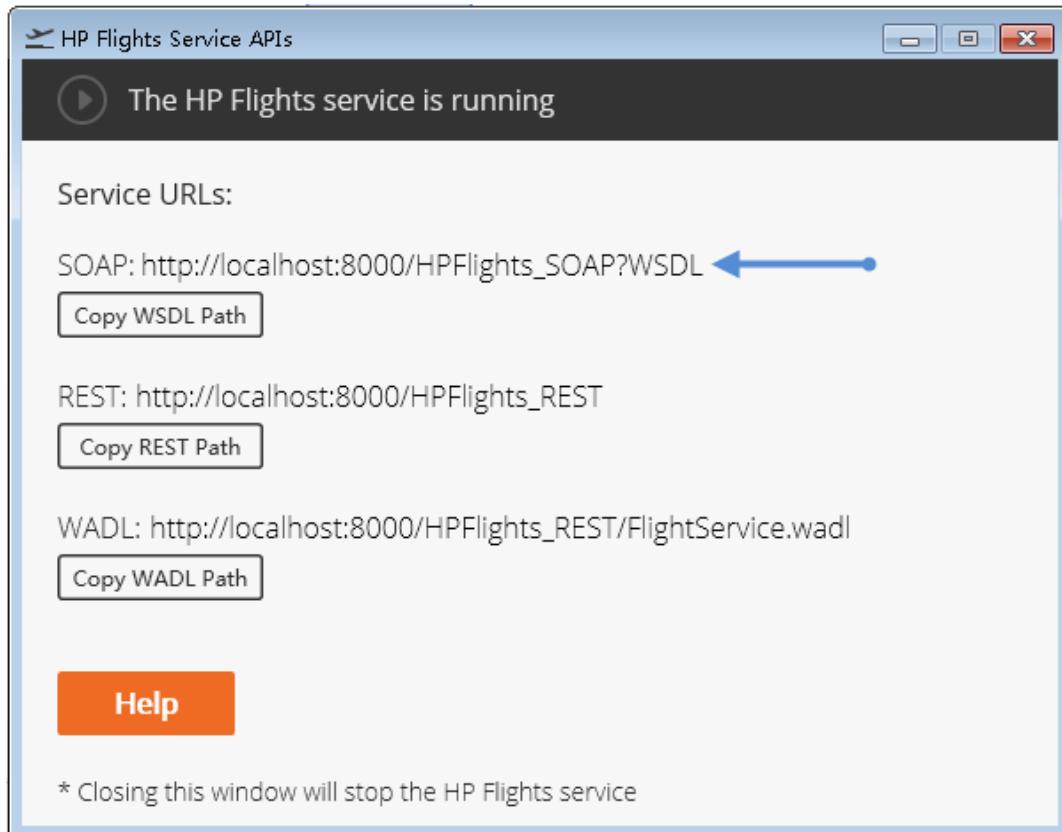
备注: 在使用本教程时，确保此应用程序保持打开状态，因为 UFT 在编辑和运行测试时，必须能够访问此应用程序。

2. 导入 WSDL 文件。

- a. 在 UFT 的工具栏上, 按“导入 WSDL”按钮并选择“从 URL 或 UDDI 导入 WSDL”。将打开“从 URL 或 UDDI 导入 WSDL”对话框。

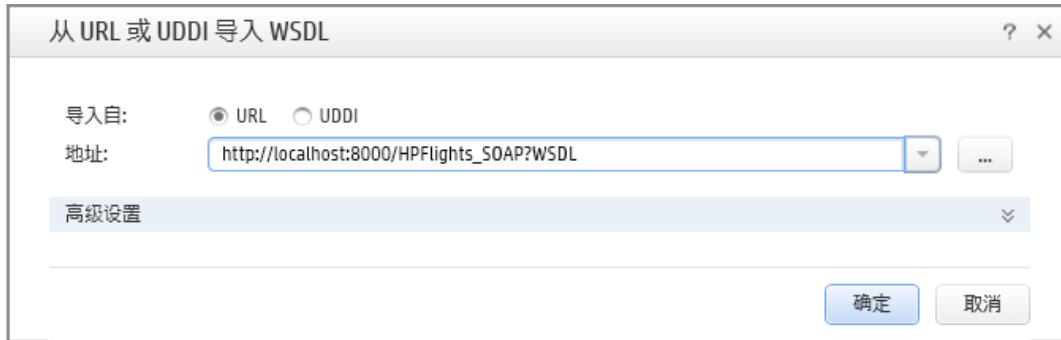
备注: 如果您本地或 ALM 项目中保存有 WSDL 文件的副本, 还可以直接将文件导入到 UFT 中。

- b. 在“从 URL 或 UDDI 导入 WSDL”对话框中, 选择“URL”选项。
- c. 在 HP Flights Service API 应用程序窗口中, 查找基于 SOAP 服务的 URL:



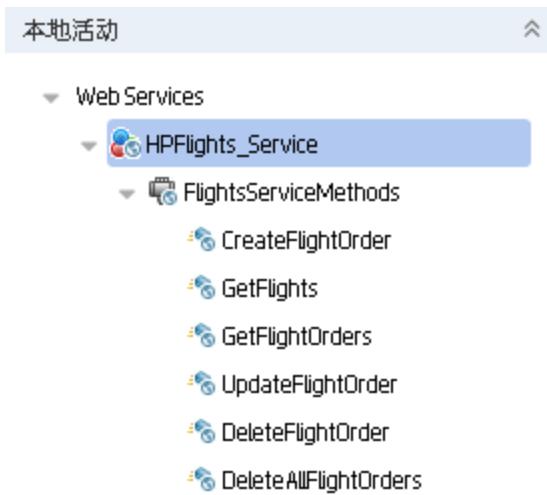
- d. 在 HP Flights Service API 应用程序窗口中, 单击“Copy WSDL Path”按钮。此操作将保存 WSDL 文件的 URL, 这样, 您便可以将其复制到“从 URL 或 UDDI 导入 WSDL”对话框中。

- e. 在 UFT 中, 在“从 URL 或 UDDI 导入 WSDL”对话框的“地址”字段中, 粘贴从应用程序窗口复制的 URL:



- f. 单击“确定”将服务导入到 UFT。

此服务将导入到 UFT 并且其方法显示在“工具箱”窗格中“本地活动”部分下方:



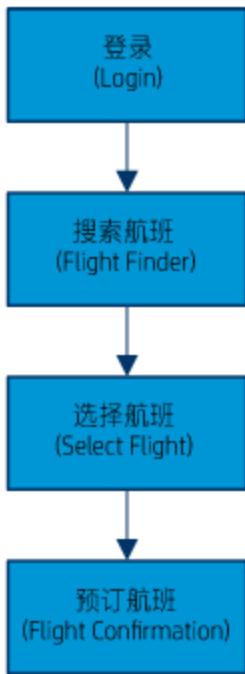
现在, 您已将服务及其方法导入到了测试中。您随时可以创建 Web Service 的测试。继续[练习 6c: 构建并参数化 Web Service 测试 \(第 161 页\)](#)学习如何构建并参数化 Web Service 的测试。

练习 6c: 构建并参数化 Web Service 测试

在[练习 6b: 导入 Web Service \(第 159 页\)](#)中, 您导入了包含 Web Service 详细信息的 WSDL 文件。导入服务后, UFT 在“工具箱”窗格中显示方法。从“工具箱”窗格, 您可以使用任意这些方法创建测试。

在本课程中, 您将创建 Web Service 测试并将其参数化, 以查看如何使用 UFT API 测试界面创建 Web Service 测试。

在您在[课程 3:向测试添加步骤 \(第 38 页\)](#) 中创建的 Book Flights GUI 测试中，应用程序窗口的顺序如下所示:



在创建相同应用程序的 API 测试时，您希望使这些步骤尽可能地与应用程序的流程匹配。在从 WSDL 文件导入的方法列表中，您具有以下内容:

- **CreateFlightOrder**
- **GetFlights**
- **GetFlightOrders**
- **UpdateFlightOrder**
- **DeleteFlightOrder**
- **DeleteAllFlightOrders**

为了与用户界面的流程匹配，您需要创建查找航班的 API 测试步骤，然后根据客户的输入创建航班订单。

在本练习中，您将创建两个测试步骤: **GetFlights** 和 **CreateFlightOrder**。

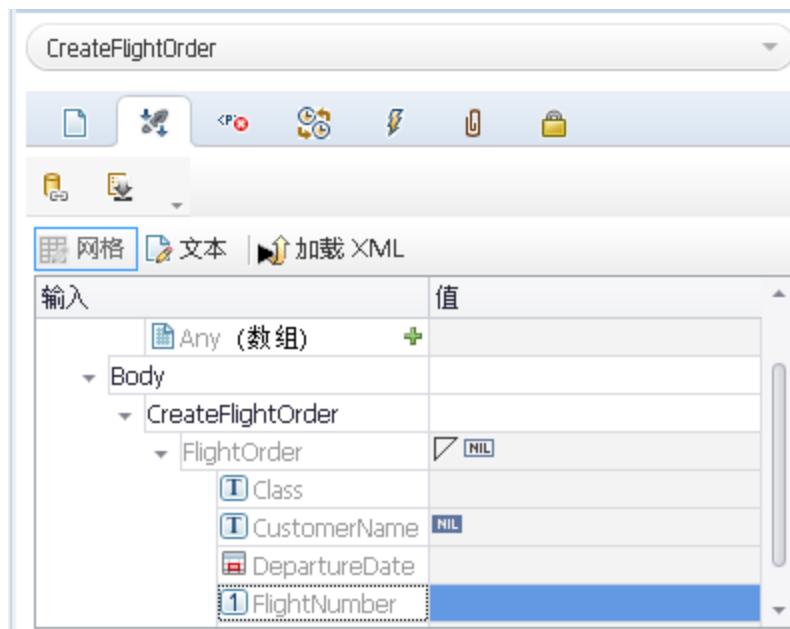
1. 创建测试步骤。

- a. 在“工具箱”窗格的“本地活动”部分，依次展开 Web Service、“HP_Flights Services”和“FlightsServiceMethods”节点。“工具箱”窗格将显示完整的可用方法（共六个）列表。
- b. 从 FlightsServiceMethods 列表，将“GetFlights”方法拖动到画布中。
名为 GetFlights 的新步骤块将添加到画布中。“输入/检查点”选项卡将在“属性”窗格中打开。
- c. 在“工具箱”窗格中，再次从“FlightsServiceMethods”节点，将“CreateFlightOrder”方法拖动到画布中。

2. 将 CreateFlightOrder 步骤的 FlightNumber 属性链接到 GetFlights 步骤的输

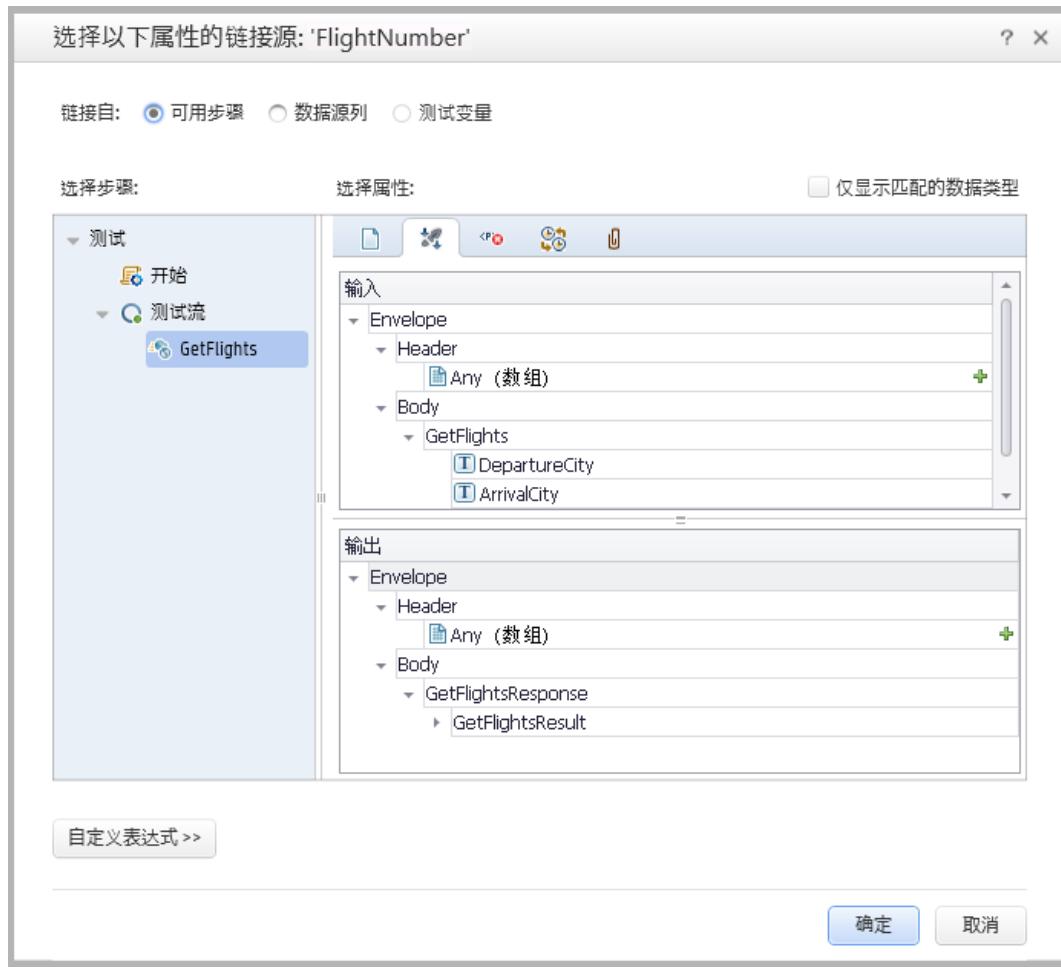
出。

- a. 在画布中，选择 **CreateFlightOrder** 步骤。“输入/检查点”选项卡  将在“属性”窗格中打开。
- b. 在“输入/检查点”选项卡的“输入”部分，选择“FlightNumber”行。



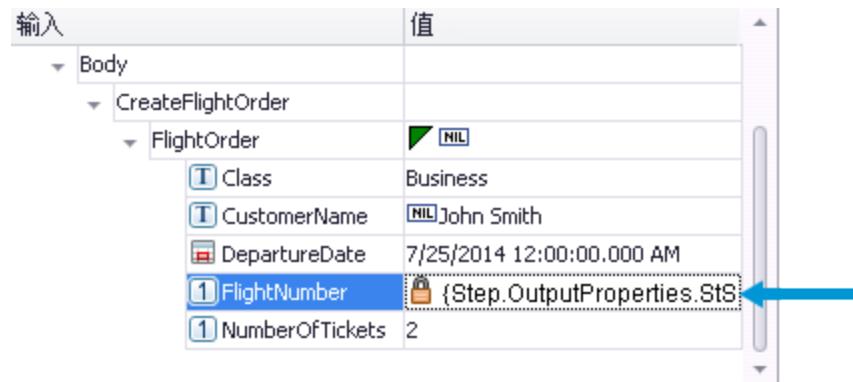
- c. 在 FlightNumber 行的“值”列中，单击“链接至数据源”按钮 。将打开“选择链接源”对话框。
- d. 在“选择链接源”对话框中，选择可用步骤选项。“选择步骤:”窗格（左侧）中的可用步骤列表将会进行更新。

- e. 在“选择步骤:”窗格中，选择“GetFlights”步骤。“选择属性:”窗格(右侧)中的步骤属性将进行更新：



- f. 在“选择步骤:”窗格中，在“输入/检查点”选项卡 的“输出”部分，展开“GetFlightsResult”节点。
- g. 在“GetFlightsResult”节点下方，在“Flight (数组)”行，单击“添加”按钮 。一个新输出数组将添加到输出属性。
- h. 在“Flight”(数组)行下，展开“Flight[1]”数组。GetFlights 步骤的所有输出属性列表将显示出来。
- i. 在输出属性列表中，选择“FlightNumber”属性，并单击“确定”。当系统提示是否要将目标步骤包括在循环中时，选择“否”。

UFT 更新 FlightNumber 属性的“值”列以反映该链接:



The screenshot shows the 'Input/Output Properties' window in UFT. The 'Body' section contains a 'CreateFlightOrder' step, which in turn contains a 'FlightOrder' step. Under 'FlightOrder', the 'FlightNumber' field is selected and its value is set to '{Step.OutputProperties.Step}'. A blue arrow points to this value field.

输入	值
Body	
CreateFlightOrder	
FlightOrder	NIL
Class	Business
CustomerName	John Smith
DepartureDate	7/25/2014 12:00:00.000 AM
FlightNumber	{Step.OutputProperties.Step}
NumberOfTickets	2

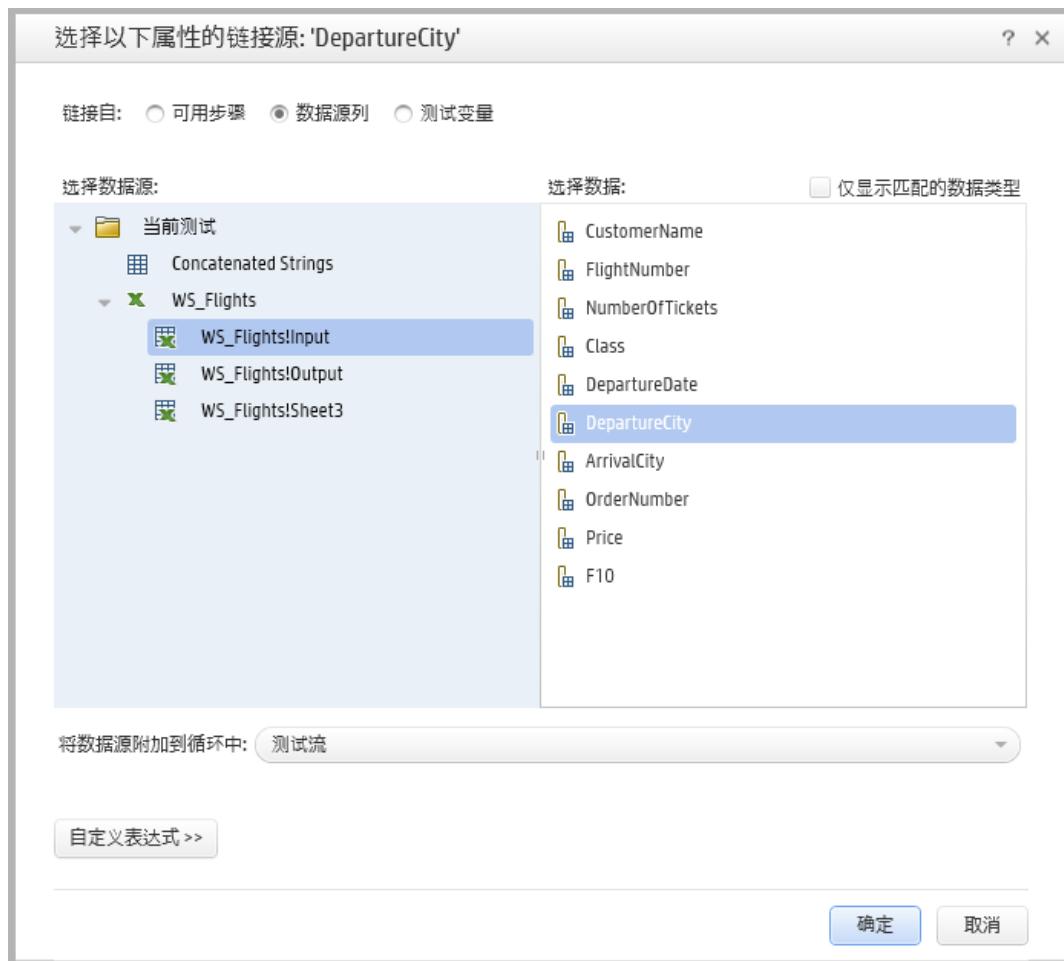
3. 添加用于测试的数据源。

- 如需要,请选择“查看”>“数据”显示“数据”窗格。
- 在“数据”窗格中,单击“新建数据源”按钮  ,然后选择“Excel”。将打开“新建 Excel 数据源”对话框。
- 在“新建数据源”对话框中,在“Excel 文件路径”字段上,单击“浏览”按钮。
- 在“打开”对话框中,导航到保存在 <UFT 安装目录>\samples\Flights Application 中的应用程序 Excel 文件,然后单击“确定”。
- 在“新建 Excel 数据源”对话框中,将文件命名为 WS_Flights。
- 选择“链接到 Excel 文件的原始位置”选项。
- 单击“确定”保存数据源信息,并将 Excel 数据添加到测试。

4. 将测试步骤的输入属性链接到数据源。

- 在画布中,选择**GetFlights** 步骤。“输入/检查点”选项卡  将在“属性”窗格中打开。
- 在“输入/检查点”选项卡的“输入”部分,展开“GetFlights”节点。
- 在“GetFlights”节点中,选择“DepartureCity”行。
- 在“DepartureCity”行的“值”列中,单击“链接至数据源”按钮  。将打开“选择链接源”对话框。
- 在“选择链接源”对话框中,选择“数据源列”选项。数据源列表(这种情况下是 Excel 文件)将显示在“选择数据源:”窗格(左侧)中。
- 在“选择数据源:”窗格中,选择“WS_Flights!Input”节点。所有可用数据参数(列)列表将显示在“选择数据”窗格(右侧)中。

- g. 在“选择数据”窗格中，选择“DepartureCity”列:



- h. 单击“确定”将属性链接到数据表的此列中。

UFT 值列中的语句将进行更新，以显示到数据源的链接。

- i. 对其他步骤属性重复执行以上过程:

- **ArrivalCity**
- **FlightDate**

- j. 对 CreateFlightOrder 步骤属性重复执行相同过程:

- **Class**
- **CustomerName**

备注: 在此属性的“值”列中，有一个其中写有“NIL”的蓝框。您需要在链接属性之前，先单击此框并删除“NIL”(蓝框将变为白色)。

- **DepartureDate**
- **NumberofTickets**

5. 为数据源设置导航设置。

- a. 在画布中，选择“测试流”(而非测试流中的步骤)。

- b. 在“属性”窗格中，选择**数据源**选项卡 。
- c. 在“数据源”选项卡中，在相关联的数据源列表中，选择表中的“WS_Flights!Input”条目，并单击“编辑”。将打开“数据导航”对话框。
- d. 在“数据导航”对话框中，指定数据导航详细信息：

开始于：	第一行
移动行数：	向前 3 行
结束于：	最后一行
在到达最后一行时：	环绕

- e. 单击“确定”指定数据导航属性并关闭对话框。

现在，你已为航班预定应用程序的 Web Service 创建了测试。您可以运行该测试、查看 UFT 的运行方式并报告 Web Service 的运行结果。请继续[练习 6d:运行 Web Service 测试 \(第 167 页\)](#)学习更多内容。

练习 6d: 运行 Web Service 测试

在[练习 6c:构建并参数化 Web Service 测试 \(第 161 页\)](#)中，您从导入的方法创建了 Web Service 测试，然后参数化了其中一个步骤。在本课程中，您将运行该测试，查看在测试 Web Service 应用程序时，UFT 如何报告运行结果。

1. **启动 UFT 并打开 Book Flight 解决方案。**
 - a. 如[为所有测试创建解决方案 \(第 20 页\)](#)中所述打开 UFT。确保已加载 WPF 插件。
 - b. 在起始页上的“最近使用的解决方案”区域，单击“Flight Reservation Application”。

将打开 Flight Reservation Application 解决方案，其中包含您在[课程 1: 创建 GUI 测试和操作 \(第 24 页\)](#)中创建的 **Book Flights** 测试。
2. **启动 Flights API 应用程序。**

如需要，请启动 HP Flights Service API 应用程序。如[浏览航班预订应用程序 \(第 18 页\)](#)中所述。

备注: 在使用本教程时，确保此应用程序保持打开状态，因为 UFT 在编辑和运行测试时，必须能够访问此应用程序。
3. **设置测试的迭代次数。**
 - a. 在画布中，选择“测试流”(而非测试流中的步骤)。“输入”选项卡  将在“属性”窗格中打开。
 - b. 在“输入”选项卡中，选择“'For' 循环”选项。
 - c. 在“迭代次数”字段中，输入 4。
4. **运行测试。**
 - a. 在工具栏中，单击“运行”按钮 。将打开“运行”对话框。

b. 在“运行”对话框中的“结果位置”选项卡中，确保“临时运行结果文件夹”仍然选中。

c. 单击运行以编译并运行测试。

测试运行完成后，将打开运行结果。

5. 查看运行结果。

a. 在测试流中，在这些迭代的任何节点下方，选择“GetFlights”节点。将显示步骤摘要详细信息。

b. 在捕获图中，向下滚动，直到可以看到“Web Service 调用 HTTP 快照”区域：

Web Service Call HTTP Snapshot

Request	Response
HTTP Header	
SOAPAction: HP.SOAQ.SampleApp/IFlightsSoapService/GetFlights Content-Type: text/xml; charset=utf-8 Host: localhost:8000 Content-Length: 210 Expect: 100-continue Connection: Close	Connection: close Content-Length: 14696 Content-Type: text/xml; charset=utf-8 Date: Mon, 28 Jul 2014 07:21:49 GMT Server: Microsoft-HTTPAPI/2.0
SOAP	SOAP
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/"> <Body> <GetFlights xmlns="HP.SOAQ.SampleApp"> <DepartureCity>Los Angeles</DepartureCity> <ArrivalCity>Sydney</ArrivalCity> </GetFlights> </Body> </Envelope>	<DepartureCity>Los Angeles</DepartureCity> <DepartureTime>09:04 AM</DepartureTime> <FlightNumber>13930</FlightNumber> <Price>117</Price> </Flight> <Flight> <Airline>LH</Airline> <ArrivalCity>Sydney</ArrivalCity> <ArrivalTime>10:02 AM</ArrivalTime>

请注意，在此区域中，UFT 为 Web Service 调用提供 HTTP 请求和响应信息。

在“请求”的“SOAP”窗口中，您可以看到为 GetFlights 步骤发送的输入属性：

```

<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">  

<Body>  

<GetFlights xmlns="HP.SOAQ.SampleApp">  

<DepartureCity>Los Angeles</DepartureCity>  

<ArrivalCity>Sydney</ArrivalCity>  

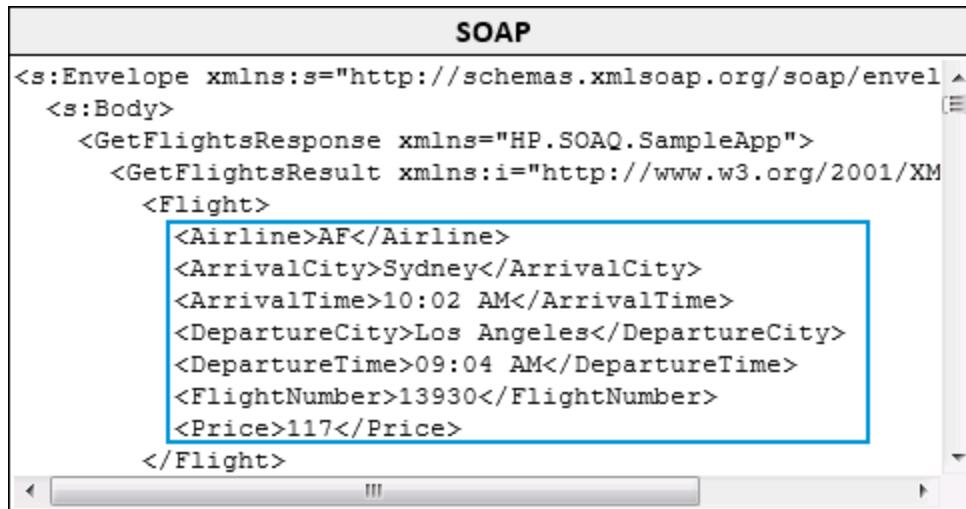
</GetFlights>  

</Body>  

</Envelope>

```

同样地，在“响应”区域，您可以看到步骤的输出属性：



The screenshot shows a UFT interface displaying a SOAP response. The title bar says "SOAP". The XML content is as follows:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope">
<s:Body>
<GetFlightsResponse xmlns="HP.SOAP.SampleApp">
<GetFlightsResult xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
<Flight>
<Airline>AF</Airline>
<ArrivalCity>Sydney</ArrivalCity>
<ArrivalTime>10:02 AM</ArrivalTime>
<DepartureCity>Los Angeles</DepartureCity>
<DepartureTime>09:04 AM</DepartureTime>
<FlightNumber>13930</FlightNumber>
<Price>117</Price>
</Flight>
</GetFlightsResult>
</GetFlightsResponse>
</s:Body>
</s:Envelope>
```

A blue box highlights the flight details section: <Flight> ... </Flight>.

- c. 在“GetFlights”节点下方，选择“检查点”节点。运行结果将显示该检查点的摘要。
- d. 在检查点的步骤详细信息中，UFT 将显示结果（无论检查点是通过还是失败）、实际值和预期值以及检查点类型：

Name	Result	Property	Actual Result	Evaluation Style	Expected Values	Details
"检查点 1"	✓		""	""	结构验证	""

6. 保存测试。

选择文件 > 保存。

课程 7:创建并运行 REST 服务的 API 测试

除了测试 UFT 中的 Web Service, API 测试还测试应用程序基于 REST 的服务或基于 REST 的服务层。在 UFT 中创建服务的原型模型, 然后使用创建的方法构建测试。

本课程为您提供创建 REST 服务原型模型和使用这些方法模型创建测试的基本步骤。

此课程包含以下内容:

· 练习 7a:创建 REST 服务测试	171
· 练习 7b:创建 REST 服务结构	171
· 练习 7c: 使用 REST 服务方法创建测试	175
· 练习 7d:运行 REST 服务测试	178
· 练习 7e:解决 REST 服务冲突	179

练习 7a: 创建 REST 服务测试

在[课程 6: 创建并运行 Web Service 的 API 测试 \(第 158 页\)](#)中，您为航班预订应用程序 API 端创建了 Web Service 测试。在本练习中，您将为航班预订应用程序 API 端的 REST 服务组件添加测试。

1. **启动 UFT 并打开 Flight Reservation Application 解决方案。**
 - a. 如[为所有测试创建解决方案 \(第 20 页\)](#)中所述打开 UFT。
 - b. 在起始页上的“最近使用的解决方案”区域，单击“Flight Reservation Application”。
将打开 Flight Reservation Application 解决方案。
2. **将新 API 测试添加到解决方案。**
 - a. 在工具栏上，选择“添加”下拉箭头  + 添加 并选择“添加新测试”。将打开“添加新测试”对话框。
 - b. 在“添加新测试”对话框中，选择“API 测试”。
 - c. 在“名称”字段中，将测试命名为 Book Flights REST Service。
 - d. 在“位置”字段中，单击“浏览”按钮并导航到 **C:\%HOMEPATH%\Unified Functional Testing** 文件夹。
 - e. 单击“添加”创建测试，并将其添加到解决方案。
Book Flights REST Service 测试将作为一个独立节点添加到 Flight Reservation Application 解决方案中，并在文档窗格的单独选项卡中打开。

备注: 解决方案会自动保存。

现在，您已创建了测试。您随时可以使用 REST 服务模型及其方法。请继续[练习 7b: 创建 REST 服务结构 \(第 171 页\)](#)学习如何在测试中创建 REST 服务模型。

练习 7b: 创建 REST 服务结构

在测试中使用 REST 服务活动之前，必须在 UFT 内创建必需方法及其属性的模型。然后，UFT 获取有关服务和方法的信息，并将之用作测试步骤以测试真实服务的性能。

1. **启动 UFT 并打开 Book Flight 解决方案。**
 - a. 如[为所有测试创建解决方案 \(第 20 页\)](#)中所述打开 UFT。
 - b. 在起始页上的“最近使用的解决方案”区域，单击“Flight Reservation Application”。
此时将打开 Flight Reservation Application 解决方案，其中包含您在[练习 7a: 创建 REST 服务测试 \(第 171 页\)](#)中创建的 Book Flights REST Service 测试。
2. **启动 Flights API 应用程序。**
确保 HP Flights Service API 应用程序正在运行，如[浏览航班预订应用程序 \(第 18 页\)](#)中所述。
3. **打开 REST 服务方法属性帮助文档。**

在 HP Flights Service API 窗口中，单击“HELP”按钮。将打开显示方法信息的浏览器窗口。

4. 创建 REST 服务模型。

- 在工具栏中，单击“添加 REST 服务”按钮。将打开“添加 REST 服务”对话框。
- 在“添加 REST 服务”对话框中，将“新建服务”名称更改为 Flights REST Service。

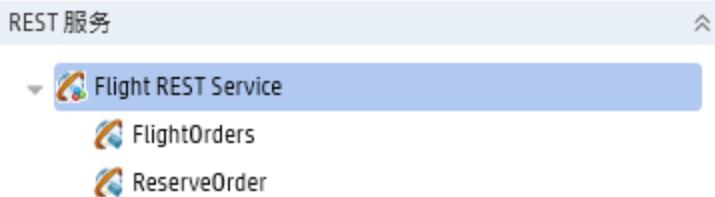
5. 将资源添加到 REST 服务模型。

- 在“添加 REST 服务”对话框中，单击工具栏上的“添加资源”按钮 。一个子节点将添加到 Flights REST Service 节点。
- 将资源名称更改为“FlightOrders”。

6. 将方法添加到 REST 服务模型。

- 在“添加 REST 服务”对话框中，单击工具栏上的“添加方法”按钮 。一个子节点将添加到 FlightOrders 资源。
- 将资源名称更改为 ReserveOrder。

现在，您已添加了服务、资源和方法，您应有三级层次结构：



7. 配置 REST 服务方法模型 URL。

为了使 REST 服务模型方法准确测试应用程序，您必须提供应用程序服务的 URL。此 URL 在“服务”、“资源”和“方法”级别提供。

- 在“添加 REST 服务”对话框中，选择“Flights REST Service”节点。“常规”选项卡  将在右窗格中打开。
- 在“常规”选项卡中，在“URL”属性的“值”列中，输入 `http://localhost:8000`。
- 在左窗格中，选择“Flight Orders”行。“常规”窗格  再次在右窗格中打开。请注意，您在“常规”窗格中添加的 URL 在选定 Flights REST Service 时显示出来。
- 在“常规”选项卡中，在“相对 URL”属性的“值”列中，输入 `HPFlights_REST`。

添加此部分 URL 后，UFT 通过将“相对 URL”值添加到“URL”值构造连接
URL:`http://localhost:8000/HPFlights_REST`。

- 在左窗格中，选择 **ReserveOrder** 节点。“常规”窗格  将在右窗格中打开。
- 在右窗格中，选择“HTTP 输入/检查点”选项卡 。
- 在“相对 URL”属性的“值”列，输入 `/FlightOrders`。

UFT 将此部分 URL 与从 **Flights REST Service** 和 **Flight Orders** 级别传递来的 URL 相连接。

8. 配置 ReserveOrders 方法模型的其他 HTTP 属性。

a. 在“添加 REST 服务”对话框中，选择“ReserveOrder”节点。“常规”窗格  将在右窗格中打开。

b. 在右窗格中，选择“HTTP 输入/检查点”选项卡 .

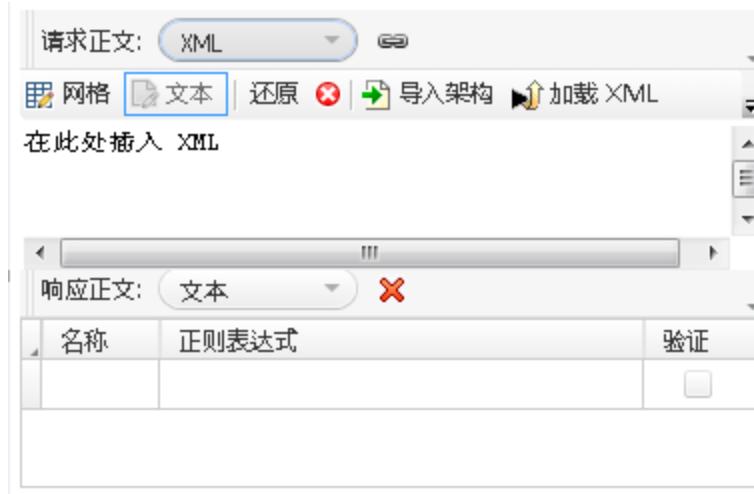
c. 在“HTTP 方法”属性的“值”列中，将 HTTP 类型设置为 POST。

9. 为 ReserveOrder 方法模型添加响应信息。

a. 在“添加 REST 服务”对话框中，选择 ReserveOrder 节点。“常规”选项卡  将在右窗格中打开。

b. 在右窗格中，选择“HTTP”选项卡 .

c. 在“请求”正文部分，在“请求”下拉列表中，选择“XML”。将打开文本编辑器，用于输入 XML：



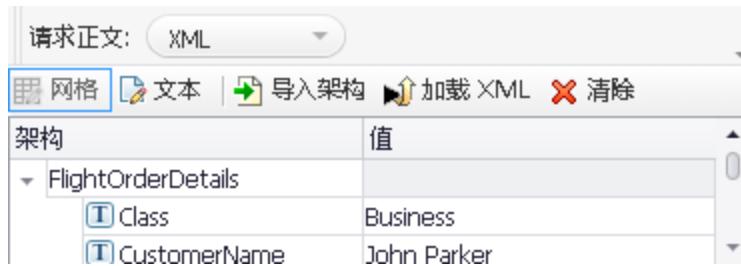
d. 在“请求”正文类型下方，确保已选择“XML”。

e. 在文本编辑器区域中，输入以下 XML：

```
<FlightOrderDetails xmlns="HP.SOAQ.SampleApp">
<Class>Business</Class>
<CustomerName>John Parker</CustomerName>
<DepartureDate>2015-05-27</DepartureDate>
<FlightNumber>1042</FlightNumber>
<NumberOfTickets>1</NumberOfTickets>
</FlightOrderDetails>
```

备注: 还可以将此 XML 保存在文件中，然后通过单击“加载 XML”按钮输入 XML。

如果单击“网格”按钮，还可以看到您在 XML 中输入的属性以网格形式显示出来：



The screenshot shows a software interface with a toolbar at the top containing buttons for '请求正文' (Request Body), 'XML' dropdown, '网格' (Grid) button (highlighted in blue), '文本' (Text) button, '导入架构' (Import Architecture) button, '加载 XML' (Load XML) button, and '清除' (Clear) button. Below the toolbar is a table titled '架构' (Architecture) with columns '架构' (Architecture) and '值' (Value). Under the 'FlightOrderDetails' node, there are two entries: 'Class' with value 'Business' and 'CustomerName' with value 'John Parker'.

架构	值
FlightOrderDetails	
Class	Business
CustomerName	John Parker

- f. 在右窗格中，再次选择“HTTP 输入/检查点”选项卡 。
- g. 在“HTTP 输入/检查点”选项卡的“输入”部分，依次展开“Request Headers”节点和“Request Headers [1]”节点。

请注意响应的设置：

- **名称:** Content - Type
- **值:** text/xml



The screenshot shows the 'Input' tab of a configuration dialog. It displays the following settings under '属性' (Properties):

- URL: /FlightOrders
- HTTP 方法: POST
- HTTP 版本: 1.1
- RequestHeaders (数组):
 - RequestHeaders[1]:
 - 名称: Content-Type
 - 值: text/xml

输入	值
属性	
URL	/FlightOrders
HTTP 方法	POST
HTTP 版本	1.1
RequestHeaders (数组)	+ 
RequestHeaders[1]	 
名称	Content-Type
值	text/xml

10. 为 ReserveOrder 方法模型创建输出属性。

- a. 在“添加 REST 服务”对话框中，选择“ReserveOrder”节点。“常规”选项卡  将在右窗格中打开。
- b. 在右窗格中，选择“自定义输入/检查点”选项卡 .
- c. 在“自定义输入/检查点”选项卡中，单击“添加”按钮，然后选择“添加新输出属性”。将打开“添加新输出属性”对话框。
- d. 在“添加新输出属性”对话框中，在“名称”字段中，输入 Total_Price。
- e. 在“类型”下拉菜单中，选择“Int”，并单击“确定”以添加输出属性。新输出属性将添加到“自定义输入/检查点”选项卡的“输出”部分。
- f. 使用同一过程，添加另一个 Int 类型的名为 Order_Number 的输出属性。

ReserveOrder 方法的“自定义输入/检查点”选项卡现在将显示您所创建的所有输出属性:

检查点	验证	值
属性	<input type="checkbox"/>	
① Total_Price	<input type="checkbox"/>	= 0
① Order_Number	<input type="checkbox"/>	= 0

11. 测试 ReserveOrder 方法模型。

- 在“添加 REST 服务”对话框中，选择“ReserveOrder”方法节点。
- 在工具栏上，单击“运行方法”按钮 .

UFT 运行方法，并在“添加 REST 服务”对话框的底部窗格中提供结果:

线上时间	00:39
响应正文	<pre> 1 <?xml version='1.0' encoding='utf-8'?> 2 <wsdl:definitions name='HPFlights_Service' targetNamespace='http://tempuri.org' xmlns:wsdl='http://schemas.xmlsoap.org/wsdl/' xmlns:xsd='http://www.w3.org/2001/XMLSchema' xmlns:soapenc='http://schemas.xmlsoap.org/soap/encoding/'> 3 <wsdl:import namespace='HPSOAP SampleApp' location='http://localhost:8000/HPSOAP_SOAP?wsdl?wsdl?'/> 4 <wsdl:types> 5 <wsdl:types> 6 <wsdl:port name='HPFlights_Service' binding='i0 FlightsServiceMethods' /> 7 <wsdl:port name='FlightsServiceMethods' binding='i0 FlightsServiceMethods' /> 8 </wsdl:types> 9 </wsdl:types> 10 <wsdl:port name='FlightOrders' location='http://localhost:8000/HPSOAP_SOAP?' /> 11 <wsdl:port name='ReserveOrder' /> 12</wsdl:definitions> </pre>
XML 响应	
响应类型	Text

12. 将服务模型及其方法添加到工具箱。

在“添加 REST 服务”对话框中，单击“确定”按钮。

UFT 将 REST 服务模型、其资源和方法添加在“工具箱”窗格的“本地活动”节点下:



从“工具箱”窗格，您可以将方法拖放到画布，并编辑其步骤属性。

现在您已创建 REST 服务的原型模型。您随时可以使用方法创建测试。请继续[练习 7c: 使用 REST 服务方法创建测试 \(第 175 页\)](#)学习在测试中使用方法。

练习 7c: 使用 REST 服务方法创建测试

在[练习 7b: 创建 REST 服务结构 \(第 171 页\)](#)中，您创建了用于创建测试步骤的 REST 服务的原型模型，其中包含方法及其属性。

在本练习中，您将使用测试流中的 REST 服务模型方法创建测试。

1. 启动 UFT 并打开 Book Flight 解决方案。

- 如[为所有测试创建解决方案 \(第 20 页\)](#)中所述打开 UFT。

- b. 在起始页上的“最近使用的解决方案”区域，单击“Flight Reservation Application”。
此时将打开 Flight Reservation Application 解决方案，其中包含您在[练习 7a: 创建 REST 服务测试 \(第 171 页\)](#)中创建的 Book Flights REST Service 测试。

2. 将用于测试的数据源导入。

- a. 如需要，请选择“查看”>“数据”显示“数据”窗格。
- b. 在“数据”窗格中，单击“新建数据源”按钮 ，然后选择“Excel”。将打开“新建 Excel 数据源”对话框。
- c. 在“新建数据源”对话框中，在“Excel 文件路径”字段上，单击“浏览”按钮。
- d. 在“打开”对话框中，导航到保存在 <UFT 安装目录>\samples\Flights Application 中的应用程序 Excel 文件，然后单击“确定”。
- e. 在“新建 Excel 数据源”对话框中，将文件命名为 Flights_REST。
- f. 选择“链接到 Excel 文件的原始位置”选项。
- g. 单击“确定”保存数据源信息，并将 Excel 数据添加到测试。

3. 创建步骤以测试 ReserveOrder 方法。

- a. 在工具栏上，单击“工具箱”按钮 。将打开“工具箱”窗格。
- b. 在“工具箱”窗格中，在“本地活动”部分，展开“Flights REST Service”节点。
- c. 在“本地活动”节点下，将“ReserveOrder”步骤拖动到画布中。
UFT 向测试流添加新块，其中包含方法名称 (ReserveOrder)。

4. 将方法的 HTTP 请求属性链接到数据源。

- a. 在画布中，选择 ReserveOrder 步骤。
- b. 在“属性”窗格中，选择“HTTP”选项卡 。
- c. 在“HTTP”选项卡中，单击“请求”部分的“网格”按钮。
- d. 在 Class 属性的“值”列中，单击“链接到数据源”按钮 。将打开“选择链接源”对话框。
- e. 在“选择链接源”对话框中，选择**数据源列**选项。“选择数据源:”窗格(左窗格)将显示所有可用数据源的列表。
- f. 在“选择数据源:”窗格中，选择“Flights_REST!Input”节点。“选择数据:”窗格(右窗格)将显示所有数据列/参数的列表。
- g. 在“选择数据:”窗格中，选择“Class”节点并单击“确定”。

UFT 更新 Class 属性的值以反映到数据源的链接:

架构	值
FlightOrderDetails	
Class	🔒 {DataSource.Flights_REST!Input.Class} 
CustomerName	John Parker
DepartureDate	2115/5/27
FlightNumber	1042
NumberOfTickets	1

- h. 对其他 HTTP 请求属性重复执行相同过程:

- **CustomerName**
- **DepartureDate**
- **FlightNumber**
- **NumberofTickets**

5. 为 ReserveOrder 步骤设置检查点。

- a. 在画布中，再次选择“ReserveOrder”步骤。
- b. 在“属性”窗格中，选择“HTTP”选项卡 。
- c. 在文本编辑器中，粘贴以下 XML:

```
<?xml version="1.0"?>
<CreatedOrderParams xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="HP.SOAQ.SampleApp">
  <OrderNumber>93</OrderNumber>
  <TotalPrice>374.400024</TotalPrice>
</CreatedOrderParams>
```

- d. 将文本编辑器中的文件以名称 **response.xml** 保存在您所选择的目录下。
- e. 在 UFT 中，在“属性”窗格“HTTP”选项卡的“响应”部分，从下拉列表中，选择“XML”。
- f. 在“响应正文”部分中，单击“加载 XML”按钮。
- g. 在“打开”对话框中，导航到您在之前步骤中保存的 **response.xml** 文件，然后单击“打开”。

UFT 将 XML 架构从 **response.xml** 文件加载到“HTTP”选项卡的“响应正文”部分:

The screenshot shows the 'Response XML' configuration in UFT. The 'Response XML' dropdown is set to 'XML'. Below it, there are buttons for 'Import Structure' (green), 'Load XML' (blue), and 'Clear' (red). A table lists the XML structure with validation and value settings:

架构	验证	值
CreatedOrderParams	<input type="checkbox"/>	=
① OrderNumber	<input checked="" type="checkbox"/>	> 93
② TotalPrice	<input checked="" type="checkbox"/>	< 374.400024

- h. 在 **OrderNumber** 属性的“值”列中，单击下拉箭头并选择“>”。
- i. 输入 **10** 作为值。
- j. 在 **TotalPrice** 属性的“值”列中，重复执行相同过程，并输入 **<** 和 **500** 作为值。

现在，您已创建了测试步骤，并提供了输入和检查点属性。您随时可以运行测试和查看运行结果。请继续[练习 7d:运行 REST 服务测试 \(第 178 页\)](#)学习更多内容。

练习 7d: 运行 REST 服务测试

在[练习 7c: 使用 REST 服务方法创建测试 \(第 175 页\)](#)中，您使用 REST 服务模型方法创建了测试。在本课程中，您将学习如何运行测试以及如何查看运行结果。

1. 启动 UFT 并打开 Book Flight 解决方案。

- a. 如[为所有测试创建解决方案 \(第 20 页\)](#)中所述打开 UFT。
 - b. 在起始页上的“最近使用的解决方案”区域，单击“Flight Reservation Application”。
- 此时将打开 Flight Reservation Application 解决方案，其中包含您在[练习 7a: 创建 REST 服务测试 \(第 171 页\)](#)中创建的 Book Flights REST Service 测试。

2. 设置测试的迭代次数。

因为您的测试有多行数据源，所以您必须指定要运行的迭代次数。

- a. 在画布中，选择“测试流”。 “输入”选项卡 将在“属性”窗格中打开。
- b. 在“输入”选项卡中，选择“‘For’循环”选项。
- c. 在“迭代次数”字段中，输入 8。

3. 设置数据源的数据导航属性。

- a. 在画布中，选择“测试流”。 “输入/检查点”选项卡 将在“属性”窗格中打开。
- b. 在“属性”窗格中，选择数据源选项卡 。将显示与测试流相关联的所有数据源列表
- c. 在数据源列表中，选择“Flights_REST!Input”数据源并单击“编辑”。将打开“数据导航”对话框。
- d. 在“数据导航”对话框中，配置以下数据导航属性：

开始于:	第一行
------	-----

移动行数:	向前 1 行
结束于:	最后一行
在到达最后一行时	环绕

4. 运行测试。

- 确保 HP Flights Service API 应用程序处于打开状态。
- 在工具栏中，单击“运行”按钮 。

UFT 运行测试步骤，提供数据源的属性值。测试运行日志将在“输出”窗格中显示。

测试运行完成后，将打开运行结果。

5. 分析运行结果。

- 在“运行结果”树窗格（左窗格）中，展开“Action: Book Flights REST Service”节点。
- 在运行结果树中，选择“ReserveOrder”节点。将显示步骤详细信息。
- 在捕获的数据中，请注意“请求”和“响应”信息。您可以通过单击“请求正文”和“响应正文”单元格，在浏览器窗口中打开 XML 响应和请求信息：

```
<?xml version="1.0"?>
- <CreatedOrderParams xmlns:i="http://www.w3.org/2001/XMLSchema-instance" xmlns="HP.SOAQ.SampleApp">
  <OrderNumber>96</OrderNumber>
  <TotalPrice>374.400024</TotalPrice>
</CreatedOrderParams>
```

- 在 ReserveOrder 节点下方，选择“Checkpoints”节点。

在捕获的数据中，请注意检查点的状态。在这种情况下，检查点处于已通过状态，因为真实值处于预期值限制范围内：

Name	Result	Property	Actual Result	Evaluation Style	Expected Values	Details
"检查点 1"	✓	"CreatedOrderParams[1]/OrderNumber[1]"	"96"	>	"10"	""
"检查点 2"	✓	"CreatedOrderParams[1]/TotalPrice[1]"	"374.400024"	<	"500"	""

练习 7e: 解决 REST 服务冲突

在[练习 7b: 创建 REST 服务结构（第 171 页）](#)中，您创建了带有 ReserveOrder 方法的原型 REST 服务模型。此方法具有特定的属性，如 URL 和属性名称。如果在创建测试后更改了服务模型的属性，则测试将不再匹配模型。因此，UFT 提供了“解决冲突”向导，使用此向导您可以解决方法属性中的变更问题。

在本练习中，您将使用“解决冲突”向导来帮助解决这些差异。

1. 启动 UFT 并打开 Book Flight 解决方案。

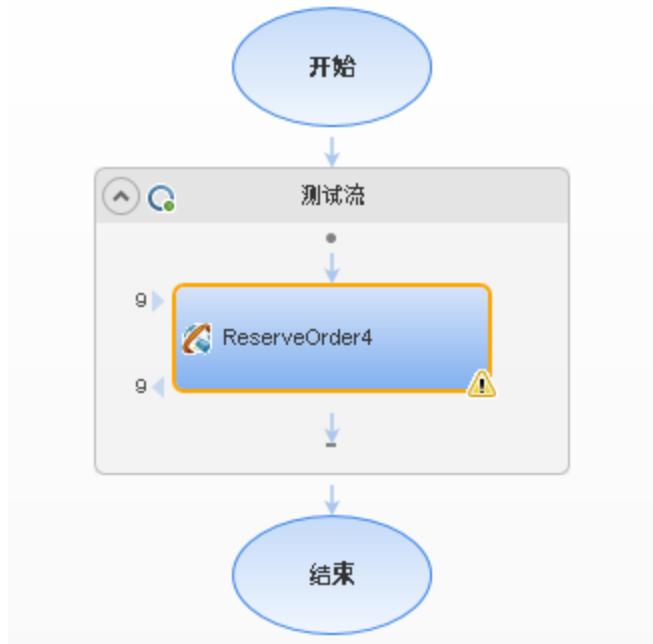
- 如[为所有测试创建解决方案（第 20 页）](#)中所述打开 UFT。

- b. 在起始页上的“最近使用的解决方案”区域，单击“Flight Reservation Application”。
此时将打开 Flight Reservation Application 解决方案，其中包含您在[练习 7a: 创建 REST 服务测试 \(第 171 页\)](#)中创建的 Book Flights REST Service 测试。

2. 编辑服务模型属性。

- a. 在工具栏上，单击“工具箱”按钮 。
- b. 在“工具箱”的“本地活动”部分，展开“REST Services”节点下的节点。
- c. 右键单击“Flights REST Service”节点，并选择“编辑服务”。将打开“编辑 REST 服务”对话框。
- d. 在“编辑 REST 服务”对话框中，选择“ReserveOrder”节点。“常规”选项卡  将在右窗格中打开。
- e. 在右窗格中，选择“自定义输入/检查点”选项卡 .
- f. 在“自定义输入/检查点”选项卡中，在“检查点”部分，选择“Total_Price”属性，并单击“编辑属性”按钮 。将打开“编辑属性”对话框。
- g. 在“编辑属性”对话框中，将属性的名称更改为 TotalPrice 并单击“确定”。“检查点”部分中的属性名称将被修改。
- h. 重复执行相同过程将 Order_Number 属性更改为 OrderNumber。
- i. 在“编辑 REST 服务”对话框中，单击“确定”保存您对服务的更改。

UFT 将更改保存到 REST 服务模型中。但是，在画布的 ReserveOrder 步骤中，将显示一个警报：



这说明在服务模型和测试中的步骤之间存在冲突，必须进行解决。

3. 解决冲突。

- a. 在画布中，单击“警报”图标。UFT 将显示消息：必须解析此步骤。解析步骤。
- b. 单击警报文本。将打开“解决 REST 方法”向导：



- c. 在“选择步骤”屏幕（第一个屏幕）中，选中“ReserveOrder”步骤的复选框，并单击“下一步”。将打开“解决冲突”屏幕。

备注: 使用此向导时，“选择步骤”屏幕将显示所有存在冲突的步骤。因此，您可以一次解决多个步骤冲突。

- d. 在“解决冲突”屏幕中，在“输出属性”部分（底部区域）的“更改后”框中，选择“Total_Price”属性（已标红）：

更改后:

架构	验证	值
属性		
① Total_Price	<input checked="" type="checkbox"/>	255
① Order_Number	<input checked="" type="checkbox"/>	10
① TotalPrice	<input type="checkbox"/>	0

- e. 单击“删除”。
- f. 重复执行相同过程删除 Order_Number 属性（也已标红）。
- g. 单击“下一步”继续操作。将打开“完成”屏幕。
- h. “完成”屏幕将显示服务模型中的冲突的状态。在这种情况下，当前已没有冲突。

单击“完成”退出向导。UFT 针对“解决 REST 方法”向导中选定的变更更新测试。在本练习中，**ReserveOrder** 步骤的输出属性现在已在“属性”窗格中得以更新：

检查点	验证	值
① 属性		
① Total_Price	<input type="checkbox"/>	= 0
① Order_Number	<input type="checkbox"/>	= 0

4. 保存测试。

选择文件 > 保存。

课程 8:创建并运行 Web 应用程序服务 (WADL) 的 API 测试

在[课程 3:使用标准活动创建 API 测试步骤 \(第 130 页\)](#)中，您学习了如何使用标准 API 活动创建测试。

但是，标准活动可能会与您应用程序执行的进程不匹配。在这种情况下，您需要使用导入到 UFT 中的自定义活动。其他服务活动类型之一就是 Web 应用程序服务。在 UFT 中，您将服务的描述及其方法导入到 UFT 中，使其可用于测试。

在本课程中，您将学习 Web 应用程序服务描述的用法以及如何将其导入到 UFT 中。

此课程包含以下内容：

· 练习 8a:为 Web 应用程序服务创建测试	184
· 练习 8b:导入 Web 应用程序服务模型	184
· 练习 8c:编辑 Web 应用程序服务方法	186
· 练习 8d:构建带有 Web 应用程序服务方法的测试	189
· 练习 8e:运行 Web 应用程序服务测试	193

练习 8a: 为 Web 应用程序服务创建测试

在[为所有测试创建解决方案 \(第 20 页\)](#) 中，您创建了航班预订应用程序的解决方案，并向该解决方案添加了应用程序用户界面的 GUI 测试，以及应用程序 Web Service 和 REST 服务的 API 测试。

在本练习中，您将为航班预订应用程序的 Web 应用程序服务创建测试。

1. 启动 UFT 并打开 Flight Reservation Application 解决方案。
 - a. 如[为所有测试创建解决方案 \(第 20 页\)](#) 中所述打开 UFT。
 - b. 在起始页上的“最近使用的解决方案”区域，单击“Flight Reservation Application”。
Flight Reservation Application 解决方案将在解决方案浏览器中打开。
2. 将新 API 测试添加到解决方案。
 - a. 在工具栏上，选择“添加”下拉箭头  并选择“新建测试”。
 - b. 在“添加新测试”对话框中，选择“API 测试”。
 - c. 在“名称”字段中，将测试命名为 Flights WADL。
 - d. 在“位置”字段中，单击“浏览”按钮并导航到 **C:\%HOMEPATH%\Unified Functional Testing** 文件夹。
 - e. 单击“添加”创建测试，并将其添加到解决方案。

Book Flights WADL 测试将作为一个独立节点添加到 Flight Reservation Application 解决方案中，并在文档窗格的单独选项卡中打开。

备注: 解决方案会自动保存。

现在，您已创建了测试。您随时可以使用 Web 应用程序服务模型及其方法。请继续[练习 8b: 导入 Web 应用程序服务模型 \(第 184 页\)](#) 学习如何将服务描述导入到 UFT 中。

练习 8b: 导入 Web 应用程序服务模型

测试 Web 应用程序服务之前，必须先导入服务描述（包括其资源结构及其方法）。Web 应用程序服务描述存储在 WADL（Web 应用程序描述语言）文件中。UFT 读取此文件，然后创建服务、资源和方法的层次结构（与 REST 服务层次结构类似）。只有导入服务描述，才能使用这些方法创建测试。

在本练习中，您会将航班预订应用程序的服务 WADL 文件导入到 UFT 中。

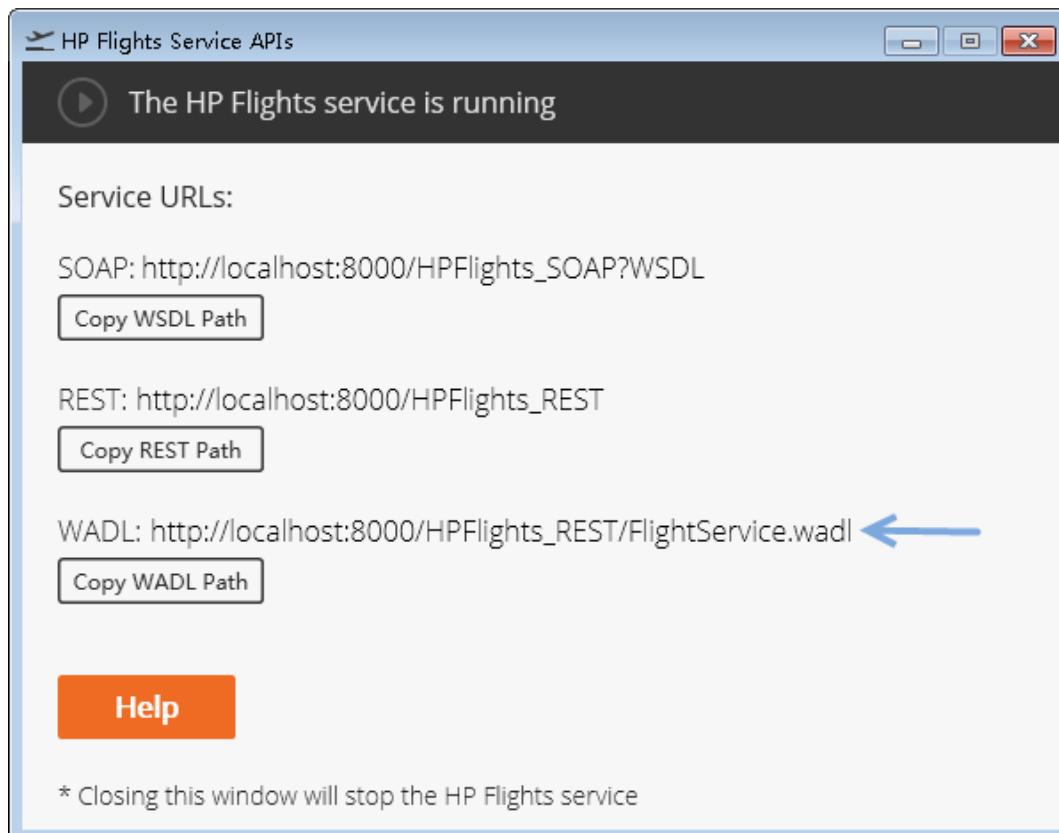
1. 启动 Flights API 应用程序。
启动 HP Flights Service API 应用程序，如[浏览航班预订应用程序 \(第 18 页\)](#) 中所述。

备注: 在使用本教程时，确保此应用程序保持打开状态，因为 UFT 在编辑和运行测试时，必须能够访问此应用程序。
2. 导入 WADL 文件。
 - a. 在 UFT 的工具栏上，单击“添加 REST 服务”下拉箭头，然后选择“从 URL 导入 WADL”。

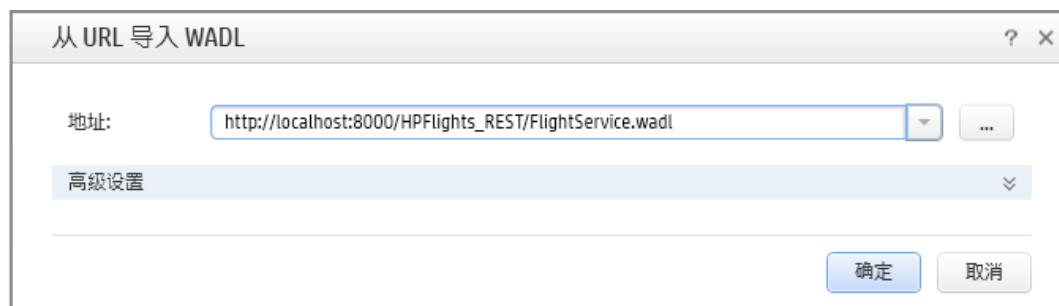
将打开“从 URL 导入 WADL”对话框。

备注: 如果您本地保存有 WADL 文件的副本，则可以将 WADL 文件导入到 UFT 中。

- b. 在 HP Flights Service API 应用程序窗口中，查找 WADL 服务的 URL:



- c. 在 HP Flights Service API 应用程序窗口中，单击“Copy WADL Path”按钮。此操作将保存 WADL 文件的 URL，这样，您便可以将其复制到“从 URL 导入 WADL”对话框中。
- d. 在 UFT 中，在“从 URL 导入 WADL”对话框的“地址”字段中，粘贴从应用程序窗口复制的 URL:



- e. 单击“确定”将服务描述导入到 UFT。

服务描述被导入到 UFT 中，其资源和方法层次结构将在“本地活动”部分的“工具箱”窗格中显示：



现在，您已将服务描述及其方法导入到了 UFT 中。您随时可以使用这些方法创建测试。请继续[练习 8c: 编辑 Web 应用程序服务方法 \(第 186 页\)](#) 学习如何在 UFT 中编辑服务描述方法。

练习 8c: 编辑 Web 应用程序服务方法

在[练习 8b: 导入 Web 应用程序服务模型 \(第 184 页\)](#) 中，您将 Web 应用程序服务的描述和方法导入到了 UFT 中，用于创建服务、资源和方法的层次结构。

您导入的 WADL 文件中包含服务资源和方法的详细描述，其中包括服务的 URL 及其方法，以及各种方法的参数。在将 WADL 文件导入到 UFT 中后，您将无法更改这些属性。

但是，您可以为服务描述添加附加信息，如方法的其他参数、方法的请求和响应信息。此附加信息也保存在服务模型的一部分，并在将方法拖动到画布时用作原型信息。

在本课程中，您将学习如何在 UFT 中编辑服务模型的属性。您将向 GetFlights 和 ReserveOrder 方法添加信息，用于在下一个练习中创建测试。

1. 启动 UFT 并打开 Flight Reservation Application 解决方案。

- 如需要，请如[为所有测试创建解决方案 \(第 20 页\)](#) 中所述打开 UFT。
- 在起始页上的“最近使用的解决方案”区域，单击“Flight Reservation Application”。

Flight Reservation Application 解决方案将在解决方案浏览器中打开，其中包含您在[练习 8a: 为 Web 应用程序服务创建测试 \(第 184 页\)](#) 中创建的 Flights WADL 测试。

2. 打开 Web 应用程序服务模型描述。

在编辑 Web 应用程序服务描述时，您将会使用与用于创建和编辑 REST 服务模型描述相同的对话框和工具。

在“工具箱”窗格中，在“本地活动”部分下方，右键单击“Flight Service”节点，并选择“编辑服务”。将打开“编辑 REST 服务”对话框。

3. 编辑 GetFlights 方法的响应信息。

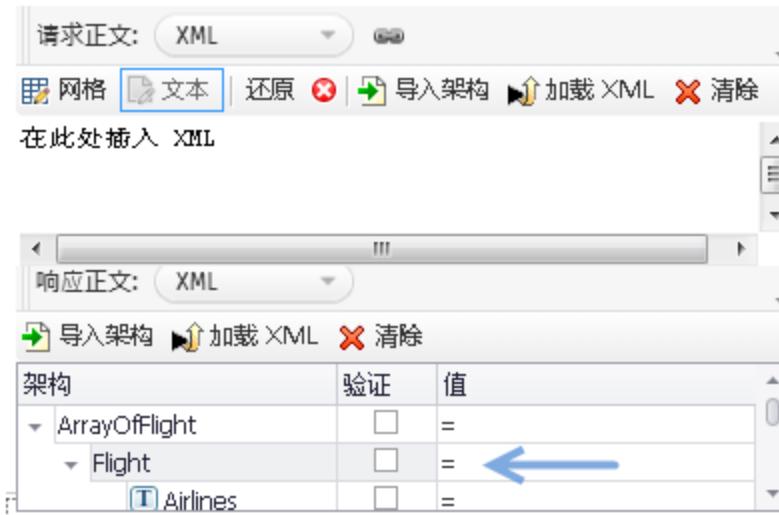
- a. 在“编辑 REST 服务”对话框中，在“Flights”节点下方，选择“GetFlights”节点。右窗格中此方法的信息将进行更新。
- b. 在右窗格中，选择“HTTP”选项卡 。
- c. 在文本编辑器中，复制以下 XML:

```
<ArrayOfFlight xmlns="HP.SOAQ.SampleApp"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Flight>
    <Airlines>AA</Airlines>
    <ArrivalCity>Denver</ArrivalCity>
    <ArrivalTime>01:23 PM</ArrivalTime>
    <DepartureCity>London</DepartureCity>
    <DepartureTime>06:12 AM</DepartureTime>
    <FlightNumber>20279</FlightNumber>
    <Price>112.2</Price>
  </Flight>
</ArrayOfFlight>
```

备注: 这种情况下，必须将值作为 XML 的一部分保留，因为它会向 UFT 说明每个参数必须使用的值类型。

- d. 将文件以名称 **response.xml** 保存在您所选择的目录下。
- e. 在 UFT 的“编辑 REST 服务”对话框中，选中“GetFlights”节点，在“HTTP”选项卡的“响应”部分，从下拉列表中，选择“XML”。
- f. 在“响应正文”部分中，单击“加载 XML”按钮。将打开“打开”对话框。
- g. 在“打开”对话框中，导航至您保存 **response.xml** 文件的目录，并选择 **response.xml** 文件。
- h. 单击“打开”将 XML 架构添加到服务描述中。

UFT 在“响应”网格中添加基于 XML 的响应属性:



4. 添加 ReserveOrders 方法的请求信息。

- 在“编辑 REST 服务”对话框中，在“FlightOrders”节点下，选择“ReserveOrder”节点。右窗格中 ReserveOrder 方法的信息将进行更新。
- 在右窗格中，选择“HTTP”选项卡 .
- 在“HTTP”选项卡的“请求”部分，从“请求正文”下拉列表中选择“XML”。将打开文本编辑器，其中显示消息: 在此处插入 XML。
- 在文本输入区域中，粘贴以下 XML:

```
<?xml version="1.0" encoding="utf-8"?>
<FlightOrderDetails xmlns="HP.SOAQ.SampleApp">
<Class>Business</Class>
<CustomerName>John Doe</CustomerName>
<DepartureDate>2012-12-12</DepartureDate>
<FlightNumber>1304</FlightNumber>
<NumberOfTickets>21</NumberOfTickets>
</FlightOrderDetails>
```

备注: 这种情况下，必须将值作为 XML 的一部分保留，因为它会向 UFT 说明每个参数必须使用的值类型。

如果单击“网格”按钮，您可以看到 XML 请求的参数在“属性”网格中显示出来：

The screenshot shows the UFT interface with the 'Properties' dialog open. At the top, there's a dropdown for 'Request Body' set to 'XML'. Below it is a toolbar with icons for 'Grid' (selected), 'Text', 'Import Structure', 'Load XML', and 'Clear'. The main area is a grid table with two columns: '架构' (Structure) and '值' (Value). Under '架构', 'FlightOrderDetails' is expanded to show 'Class' (Business) and 'CustomerName' (John Doe). Below the grid is another section for 'Response Body' with a 'Text' dropdown and a red 'X' button.

- e. 单击“确定”保存变更。

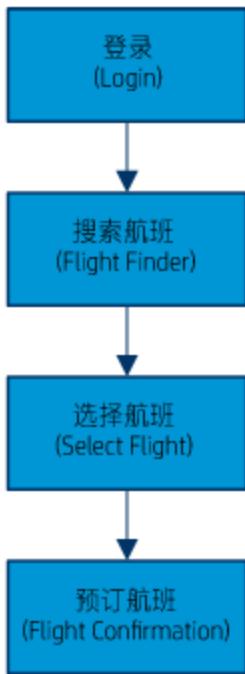
UFT 将变更保存到服务模型，并更新在“工具箱”窗格中已导入的方法的属性。

现在，您已编辑了 Web 应用程序方法，使其包含附加信息。您随时可将这些 Web 应用程序方法用于测试。请继续[练习 8d:构建带有 Web 应用程序服务方法的测试 \(第 189 页\)](#)学习使用从 WADL 导入的方法构建测试。

练习 8d:构建带有 Web 应用程序服务方法的测试

在[练习 8b:导入 Web 应用程序服务模型 \(第 184 页\)](#)中，您导入了 WADL 文件，其中包含用于 Web 应用程序服务的方法描述。在[练习 8c:编辑 Web 应用程序服务方法 \(第 186 页\)](#)中，您向这些方法添加了未在 WADL 文件中定义的其他属性信息。

在您在[课程 3:向测试添加步骤 \(第 38 页\)](#) 中创建的 Book Flights GUI 测试中，应用程序窗口的顺序如下所示:



在创建相同应用程序的 API 测试时，您希望使这些步骤尽可能地与应用程序的流程匹配。在从 WADL 文件导入的方法列表中，您具有以下某些内容:

- **GetFlights**
- **GetFlightOrders**
- **UpdateFlightOrder**
- **DeleteFlightOrder**
- **DeleteAllFlightOrders**
- **ReserveOrder**

为了与用户界面的流程匹配，您需要创建查找航班的 API 测试步骤，然后根据客户的输入创建航班订单。

在本练习中，您将创建两个测试步骤: **GetFlights** 和 **ReserveOrder**。

1. **启动 UFT 并打开 Flight Reservation Application 解决方案。**
 - a. 如需要，请如[为所有测试创建解决方案 \(第 20 页\)](#) 中所述打开 UFT。
 - b. 在起始页上的“最近使用的解决方案”区域，单击“Flight Reservation Application”。

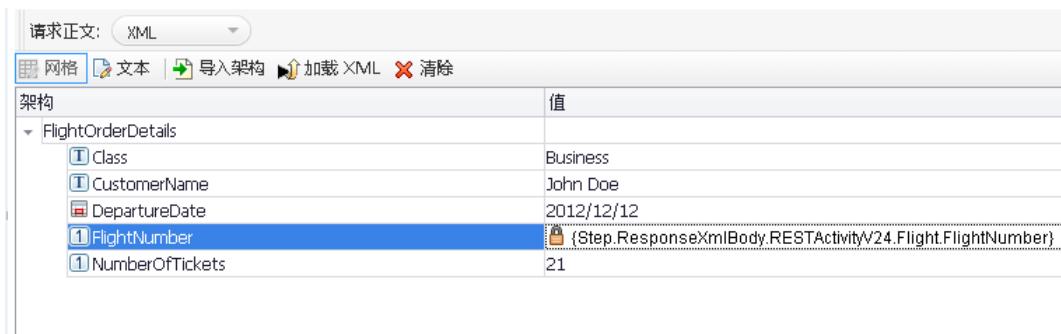
Flight Reservation Application 解决方案将在解决方案浏览器中打开，其中包含您在[练习 8a: 为 Web 应用程序服务创建测试 \(第 184 页\)](#) 中创建的 Flights WADL 测试。
2. **创建测试步骤。**
 - a. 在“工具箱”窗格中，在“本地活动”部分，展开“Flights”节点 (位于“Flight Service”和“HPFlights_REST”节点下方)。

- b. 在 Flights 节点下方, 将 “GetFlights” 方法拖动到画布中。UFT 将在测试流中添加一个名为 **GetFlights** 的测试块。
- c. 在 FlightOrders 节点下方, 将 “ReserveOrder” 节点拖动到画布中。UFT 在测试流中 (在 GetFlights 测试块下方) 添加另一个名为 **ReserveOrder** 的测试块。

3. 将 ReserveOrder 步骤的 FlightNumber 属性链接到 GetFlights 步骤的输出。

- a. 在画布中, 选择 **ReserveOrder** 步骤。“输入/检查点”选项卡  将在“属性”窗格中打开。
- b. 在“属性”窗格中, 选择“HTTP”选项卡 .
- c. 在“HTTP”选项卡的“请求正文”部分中, 在“FlightNumber”属性的“值”单元格上, 单击“链接至数据源”按钮。将打开“选择链接源”对话框。
- d. 在“选择链接源”对话框中, 选择可用步骤选项。“选择步骤:”窗格(左窗格)中的可用步骤列表将会进行更新。
- e. 在“选择步骤:”窗格中, 选择“GetFlights”步骤。“选择属性:”窗格(右窗格)中的可用属性列表将会进行更新。
- f. 在“选择属性:”窗格中, 选择“HTTP”选项卡 。将显示 HTTP 属性的列表。
- g. 在属性列表中, 选择“FlightNumber”属性, 并单击“确定”。

UFT 将更新 ReserveOrder 步骤中的 FlightNumber 属性的值, 以反映到 GetFlights 步骤输出的链接:



架构	值
FlightOrderDetails	
Class	Business
CustomerName	John Doe
DepartureDate	2012/12/12
FlightNumber	 {Step.Response.XmlBody.RESTActivityV24.Flight.FlightNumber}
NumberOfTickets	21

4. 添加用于测试的数据源。

- a. 如需要, 请选择“查看”>“数据”显示“数据”窗格。
- b. 在“数据”窗格中, 单击“新建数据源”按钮  , 然后选择“Excel”。将打开“新建 Excel 数据源”对话框。
- c. 在“新建数据源”对话框中, 在“Excel 文件路径”字段上, 单击“浏览”按钮。
- d. 在“打开”对话框中, 导航到保存在 <UFT 安装目录>\samples\Flights Application 中的应用程序 Excel 文件, 然后单击“确定”。
- e. 在“新建 Excel 数据源”对话框中, 将文件命名为 WADL_Flights。
- f. 选择“链接到 Excel 文件的原始位置”选项。
- g. 单击“确定”保存数据源信息, 并将 Excel 数据添加到测试。

5. 将 GetFlights 步骤的输入属性链接到数据源。

- a. 在画布中, 选择 **GetFlights** 步骤。 “输入/检查点” 选项卡  将在“属性”窗格中打开。
- b. 在“输入/检查点”选项卡的“DepartureCity”属性“值”单元格中, 选择“链接至数据源”按钮 。将打开“选择链接源”对话框。
- c. 在“选择链接源”对话框中, 选择**数据源列**选项。“选择数据源:”窗格(左窗格)将显示所有可用数据源的列表。
- d. 在“选择数据源:”窗格中, 选择“WADL_Flights!Input”节点。“选择数据:”窗格(右窗格)将显示所有可用数据参数(列)的列表。
- e. 在“选择数据:”窗格中, 选择“DepartureCity”参数并单击“确定”。UFT 将更新“属性”窗格中“DepartureCity”属性的“值”列, 以反映到数据源的链接。
- f. 重复执行相同过程, 将 **ArrivalCity** 和 **Date** 属性链接到数据源。

UFT 通过显示 GetFlights 属性的值来显示到数据源的链接:

输入	值
属性	
DepartureCity	 {DataSource.WADL_Flights!Input.DepartureCity}
ArrivalCity	 {DataSource.WADL_Flights!Input.ArrivalCity}
Date	 {DataSource.WADL_Flights!Input.DepartureDate}

6. 将 ReserveOrder 步骤的 HTTP 请求属性链接到数据源。

- a. 在画布中, 选择“ReserveOrder”步骤。“输入/检查点”选项卡  将在“属性”窗格中打开。
- b. 在“属性”窗格中, 选择“HTTP”选项卡 。将显示 HTTP 请求和响应属性的列表。
- c. 在“Class”属性的“值”单元格中, 单击“链接至数据源”按钮 。将打开“选择链接源”对话框。
- d. 在“选择链接源”对话框中, 选择**数据源列**选项。“选择数据源:”(左窗格)将显示所有可用数据源的列表。
- e. 在“选择数据源:”窗格中, 选择“WADL_Flights!Input”节点。“选择数据:”窗格(右窗格)将显示所有可用数据参数(列)的列表。
- f. 在“选择数据:”窗格中, 选择“Class”列, 并单击“确定”。UFT 将更新 Class 属性的值以反映到数据源的链接。
- g. 对其他 HTTP 请求属性重复执行相同过程:
 - **CustomerName**
 - **DepartureDate**
 - **NumberOfTickets**

将所有属性（“FlightNumber”属性除外）链接到数据源之后，“属性”窗格中的值将进行更新以反映该链接：

架构	值
FlightOrderDetails	
Class	{DataSource.WADL_Flights!Input.Class}
CustomerName	{DataSource.WADL_Flights!Input.CustomerName}
DepartureDate	{DataSource.WADL_Flights!Input.DepartureDate}
FlightNumber	{Step.ResponseXmlBody.RESTActivityV24.Flight.F}
NumberOfTickets	{DataSource.WADL_Flights!Input.NumberOfTickets}

7. 保存测试。

在工具栏上，单击“保存”。

现在，您已使用从 WADL 文件导入的方法创建了测试。您随时可以运行该测试并查看运行结果。请继续[练习 8e:运行 Web 应用程序服务测试 \(第 193 页\)](#)学习运行测试。

练习 8e:运行 Web 应用程序服务测试

在[练习 8d:构建带有 Web 应用程序服务方法的测试 \(第 189 页\)](#)中，您使用 Web 应用程序服务模型的方法创建了测试，这些方法是您在之前的练习中导入和编辑的。

在本练习中，您将运行所创建的测试以查看其结果。

1. 启动 UFT 并打开 Flight Reservation Application 解决方案。

- 如需要，请如[为所有测试创建解决方案 \(第 20 页\)](#)中所述打开 UFT。
- 在起始页上的“最近使用的解决方案”区域，单击“Flight Reservation Application”。

Flight Reservation Application 解决方案将在解决方案浏览器中打开，其中包含您在[练习 8a:为 Web 应用程序服务创建测试 \(第 184 页\)](#)中创建的 Flights WADL 测试。

2. 设置测试的迭代次数。

- 在画布中，选择“测试流”。 “输入”选项卡 将在“属性”窗格中打开。
- 在“输入”窗格中，选择“‘For’循环”选项。
- 在“迭代次数”字段中，输入 8。

3. 设置数据源的数据导航属性。

- 在画布中，选择“测试流”。 “输入/检查点”选项卡 将在“属性”窗格中打开。
- 在“属性”窗格中，选择数据源选项卡。将显示与测试流相关联的所有数据源列表

- c. 在数据源列表中，选择“WADL_Flights!Input”数据源并单击“编辑”。将打开“数据导航”对话框。
- d. 在“数据导航”对话框中，配置以下数据导航属性：

开始于:	第一行
移动行数:	向前 1 行
结束于:	最后一行
在到达最后一行时	环绕

4. 运行测试。

- a. 确保 HP Flights Service API 应用程序处于打开状态。

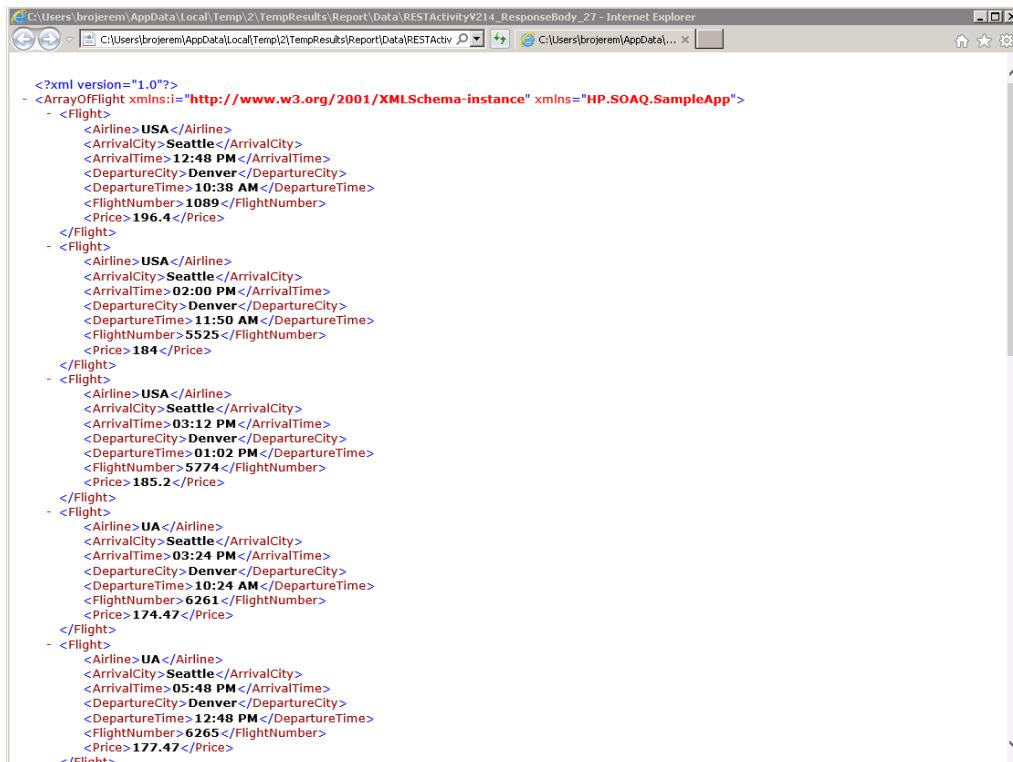
b. 在工具栏中，单击“运行”按钮 。

UFT 运行测试步骤，提供数据源的属性值。测试运行日志将在“输出”窗格中显示。

测试运行完成后，将打开运行结果。

5. 分析运行结果。

- a. 在测试流中，找到“ReserveOrder”步骤。运行结果将显示步骤的摘要。
- b. 在捕获的数据中，请注意“请求”和“响应”信息。您可以通过单击“请求正文”和“响应正文”单元格，在浏览器窗口中打开 XML 响应和请求信息：



```

<?xml version="1.0"?>
<ArrayOfFlight xmlns:i="http://www.w3.org/2001/XMLSchema-instance" xmlns="HP.SOAQ.SampleApp">
  - <Flight>
    <Airline>USA</Airline>
    <ArrivalCity>Seattle</ArrivalCity>
    <ArrivalTime>12:48 PM</ArrivalTime>
    <DepartureCity>Denver</DepartureCity>
    <DepartureTime>10:38 AM</DepartureTime>
    <FlightNumber>1089</FlightNumber>
    <Price>196.4</Price>
  </Flight>
  - <Flight>
    <Airline>USA</Airline>
    <ArrivalCity>Seattle</ArrivalCity>
    <ArrivalTime>02:00 PM</ArrivalTime>
    <DepartureCity>Denver</DepartureCity>
    <DepartureTime>11:50 AM</DepartureTime>
    <FlightNumber>5525</FlightNumber>
    <Price>184</Price>
  </Flight>
  - <Flight>
    <Airline>USA</Airline>
    <ArrivalCity>Seattle</ArrivalCity>
    <ArrivalTime>03:12 PM</ArrivalTime>
    <DepartureCity>Denver</DepartureCity>
    <DepartureTime>01:02 PM</DepartureTime>
    <FlightNumber>5774</FlightNumber>
    <Price>185.2</Price>
  </Flight>
  - <Flight>
    <Airline>UA</Airline>
    <ArrivalCity>Seattle</ArrivalCity>
    <ArrivalTime>03:24 PM</ArrivalTime>
    <DepartureCity>Denver</DepartureCity>
    <DepartureTime>10:24 AM</DepartureTime>
    <FlightNumber>6261</FlightNumber>
    <Price>174.47</Price>
  </Flight>
  - <Flight>
    <Airline>UA</Airline>
    <ArrivalCity>Seattle</ArrivalCity>
    <ArrivalTime>05:48 PM</ArrivalTime>
    <DepartureCity>Denver</DepartureCity>
    <DepartureTime>12:48 PM</DepartureTime>
    <FlightNumber>6265</FlightNumber>
    <Price>177.47</Price>
  </Flight>
</ArrayOfFlight>

```

第 5 部分: 在单个测试中创建和运行 GUI 和 API 测试

备注: 您必须先执行[创建和运行自动 GUI 测试 \(第 23 页\)](#) 和[创建和运行自动 API 测试 \(第 124 页\)](#) 中包含的教程课程, 才能执行教程的这部分内容。

测试应用程序时, 为了执行全面的测试, 必须对用户界面 (GUI) 和服务层 (API) 进行测试。执行这种测试的挑战之一就是维护和运行应用程序每个部分的单独测试。

不过在 UFT 中, 尽管您仍然必须创建和维护应用程序的 GUI 和 API 测试, 但您可以运行统一的测试, 在单个统一测试运行中同时测试应用程序的 GUI 和 API。您只需从 GUI 测试调用 API 测试, UFT 即可在单个测试运行中运行这两层。然后, 在测试运行完成后, 运行结果将显示统一的视图, 在单个报告中同时报告 GUI 和 API 层的性能。

在此部分中, 您将学习如何创建和运行将 GUI 和 API 测试包含在单个测试运行中的测试。

此部分包含以下内容:

- [课程 1: 创建测试以同时运行 GUI 和 API 测试](#) 196
- [课程 2: 从 GUI 测试中调用 API 测试](#) 197
- [课程 3: 运行一个调用 API 测试的 GUI 测试](#) 201

课程 1: 创建测试以同时运行 GUI 和 API 测试

在本课程中，您将创建一个单独的测试，以运行同时包含 GUI 和 API 测试的统一测试。

1. 启动 UFT 打开 Book Flights 测试。

- a. 如[为所有测试创建解决方案 \(第 20 页\)](#) 中所述打开 UFT。确保已加载 WPF 插件。
- b. 在起始页上的“最近使用的解决方案”区域，单击“Flight Reservation Application”。
将打开 Flight Reservation Application 解决方案，该方案包含您在[创建和运行自动 GUI 测试 \(第 23 页\)](#) 中创建的 Book Flights GUI 测试和您在[创建和运行自动 API 测试 \(第 124 页\)](#) 中创建的 API 测试。

2. 将 Book Flights 测试另存为 Flight Reservation Application。

- a. 在解决方案浏览器中，右键单击“Book Flights”节点并选择“另存为”。将打开“另存为”对话框。
- b. 在“将测试另存为”对话框中，浏览到 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 并将测试另存为 Flight Reservation Application。
在解决方案浏览器中，Book Flights 测试将替换为新的 Flight Reservation Application 测试。Book Flights 测试仍单独保存在文件系统中。

3. 将 Book Flights 测试添加回解决方案。

- a. 选择“文件”>“添加”>“现有测试”。将打开“将测试添加到解决方案中”对话框。
- b. 在“将测试添加到解决方案中”对话框中，浏览到 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 目录，然后选择 Book Flights 测试。
- c. 单击“添加”使 Book Flights 测试回到解决方案中。

解决方案浏览器中将再次显示 Book Flights 测试。

现在您已经拥有一个可同时运行 GUI 和 API 测试的独立测试，可立即构建测试，将这两种类型的测试包含在内。继续[课程 2:从 GUI 测试中调用 API 测试 \(第 197 页\)](#) 学习如何构建包含这两种测试的测试。

课程 2:从 GUI 测试中调用 API 测试

要从 GUI 测试运行 API 测试，您必须先调用 API 测试。在此课程中，您将学习如何从 GUI 测试添加对 API 测试的调用，从而在单个统一的测试运行中同时运行这两个测试。

1. 启动 UFT 打开 Book Flights 测试。

- a. 如需要，请如[为所有测试创建解决方案 \(第 20 页\)](#) 中所述打开 UFT。确保已加载 WPF 插件。
- b. 在起始页上的“最近使用的解决方案”区域，单击“Flight Reservation Application”。

将打开 Flight Reservation Application 解决方案，其中包含您在[课程 1:创建测试以同时运行 GUI 和 API 测试 \(第 196 页\)](#) 中创建的 Flight Reservation Application 测试。

2. 为 API 测试调用新建操作。

调用 API 测试时，UFT 将完整地运行调用的 API 测试。您可以将 API 测试调用作为操作中独立的步骤，或作为独立的操作。

- a. 在工具栏中，单击“调用新操作”下拉箭头  并选择“调用新操作”。将打开“插入对新操作的调用”对话框。
- b. 在“插入对新操作的调用”对话框中，将操作命名为 API Test Call。
- c. 确保已选中“测试结束时”选项。其他选项保留默认设置。
- d. 单击“确定”将操作添加到测试。
- e. 在画布中，右键单击“API Test Call”操作并选择“上移”。测试流中的 API Test Call 操作将移动到 Flight Confirmation 操作上方。

3. 添加对 API 测试的调用。

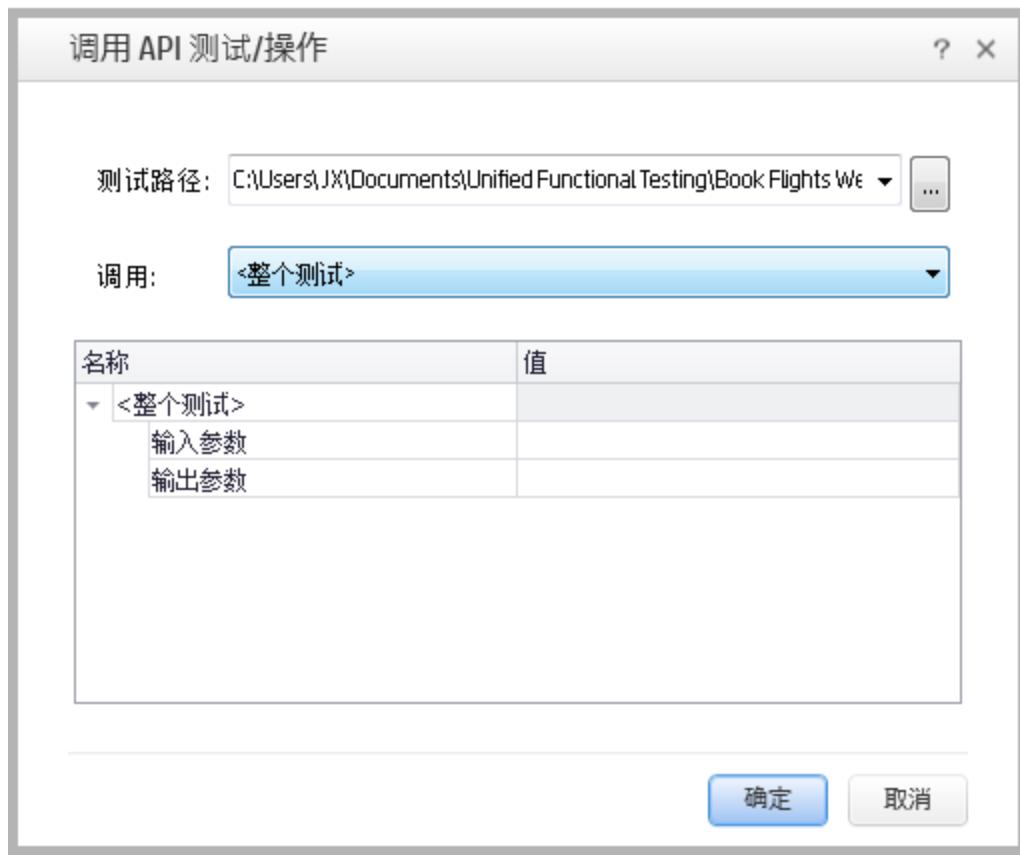
- a. 在画布中双击“API Test Call”操作。该操作将在文档窗格的单独选项卡中打开。

- b. 在工具栏中，单击“调用新操作”下拉箭头  并选择“调用现有 API 测试/操作”。将打开“调用 API 测试/操作”对话框：



- c. 在“调用 API 测试/操作”对话框的“测试路径”字段中，单击“浏览”按钮。将打开“打开测试”对话框。
- d. 在“打开测试”对话框中，导航到包含 **Flight Reservation Application** 解决方案测试的文件夹，该文件夹存储在 **C:\%HOMEPATH%\Unified Functional Testing** 中。

- e. 在此目录中, 选择“Book Flights Web Service”测试并单击“打开”。UFT 会将测试名称及其参数添加到“调用 API 测试/操作”对话框中:



备注: 如果您的测试有输出参数, 则可以使用此对话框指定存储 API 测试输出参数的位置。在此例中, API 测试没有输出参数, 因此您将不执行此步骤。

- f. 单击“确定”将调用添加为测试中的步骤。

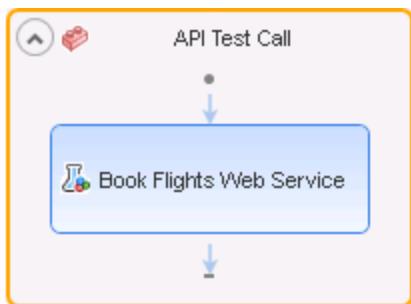
UFT 在 API Test Call 操作中添加了一个 API 测试调用步骤。在编辑器中, 该步骤显示如下:

RunAPITest "Book Flights Web Service"

在“关键字视图”中, 该步骤显示如下:

项	操作	值	文档
函数调用	RunAPITest	"Book Flights Web Service"	运行 "Book Flights Web Service" API 测试...

UFT 将在画布中向 API Test Call 操作添加一个操作调用的可视指标:



4. 保存测试。

选择文件 > 保存。

您已经创建了测试步骤以调用 API 测试，现在即可运行该测试。继续课程 3:运行一个调用 API 测试的
[GUI 测试 \(第 201 页\)](#) 学习运行测试并查看运行结果。

课程 3:运行一个调用 API 测试的 GUI 测试

在[课程 2:从 GUI 测试中调用 API 测试 \(第 197 页\)](#) 中, 您向现有 GUI 测试添加了对 API 测试的调用, 使您能够在一个测试运行中运行航班预订应用程序的统一测试。

在本课程中, 您将运行测试并查看结果。

1. 启动 UFT 打开 Book Flights 测试。

a. 如需要, 请如[为所有测试创建解决方案 \(第 20 页\)](#) 中所述打开 UFT。确保已加载 WPF 插件。

b. 在起始页上的“最近使用的解决方案”区域, 单击“Flight Reservation Application”。

将打开 Flight Reservation Application 解决方案, 其中包含您在[课程 1:创建测试以同时运行 GUI 和 API 测试 \(第 196 页\)](#) 中创建的 Flight Reservation Application 测试。

2. 更新 Book Flights Web Service 测试的迭代次数。

请记住, 创建 Book Flights Web Service 测试后, 会将测试运行多次迭代。但在 Flight Reservation Application 测试(调用了 Book Flights Web Service 测试)中, 只有一次测试迭代。因此, 您应修改 API 测试, 使其运行相同的迭代次数。

a. 在解决方案浏览器中, 展开位于“Book Flights Web Service”节点下方的节点。

b. 在“Book Flights Web Service”节点下, 双击“Flow”节点。“Book Flights Web Service Test Flow”将在文档窗格的单独选项卡中打开。

c. 在画布中, 选择“测试流”。 “输入”选项卡  将在“属性”窗格中打开。

d. 在“输入”选项卡的“迭代次数”字段中, 将数字更改为“1”。

e. 选择“文件”>“保存”保存修改后的设置。

3. 设置 Flight Reservation Application 测试的运行设置。

a. 在文档窗格中, 选择“Flight Reservation Application”选项卡。

如果“Flight Reservation Application”选项卡已关闭, 请在解决方案浏览器中双击打开“Flight Reservation Application”节点。

b. 选择“录制”>“录制和运行设置”。将打开“录制和运行设置”对话框。

c. 在“Windows 应用程序”选项卡中, 确保已选中“仅在以下应用程序上录制和运行:”和“以下指定的应用程序”选项。应用程序详细信息应已在[从课程 4:运行和分析 GUI 测试 \(第 61 页\)](#) 运行原始 Book Flights 测试时保存。

d. 单击确定关闭对话框。

4. 运行 Flight Reservation Application 测试。

a. 运行测试之前, 请确保已关闭“HP MyFlight Sample Application”(航班预订应用程序的用户界面)窗口。

b. 确保“HP Flights Service APIs”窗口处于打开状态。

c. 在工具栏中, 单击“运行”按钮 。将打开“运行”对话框。

d. 在“运行”对话框的“结果位置”选项卡中，选择“新建运行结果文件夹”选项。保留默认文件夹名称。

e. 单击“运行”以启动测试运行。

UFT 将打开“HP MyFlight Sample Application”窗口并对应用程序对象执行在 GUI 测试中创建的步骤。当测试进入 API 测试调用时，GUI 测试将暂停，API 测试成为焦点。

API 测试运行时，您可以在“输出”窗格中查看 API 测试运行的进度。

API 测试运行完成后，将运行剩余的 GUI 测试。

整个测试运行完成后，将打开运行结果并显示测试结果。

5. 查看运行结果。

在运行结果中，显示“测试流”。

请注意，GUI 测试中的每个操作均有单独的节点。但您还可以查看作为运行结果一部分的 API 测试运行结果：

步骤详细信息

步骤
RunAPITest

描述
APITest1

执行时间
2015-09-18 09:40:20

External Tests/Actions
链接: [Report](#)

选择 API 测试结果中的一个节点，以在步骤摘要中查看步骤详细信息

6. 关闭运行结果。

查看运行结果后，关闭包含运行结果的选项卡。

接下来要执行的操作

现在您已经了解如何使用 UFT，包括创建自动 GUI 测试、自动化 API 测试以及同时运行 GUI 和 API 测试的测试，现在即可使用 UFT 测试您自己的应用程序。

我们建议您在测试自己的应用程序时按照以下步骤操作。

1. 分析应用程序。

- 确定开发环境。这使您能够加载相关 UFT 插件并为应用程序中的对象提供支持。
- 确定用户将执行的业务流程。然后相应地计划测试和操作。
- 确定如何组织测试和要包含的操作。考虑测试的目标，并确认应用程序和 UFT 的设置是否符合测试的需要。

在此阶段，您可以开始创建在测试应用程序时要使用的框架测试和操作。

2. 准备测试基础结构。

决定如何在测试中存储对象。可将每个操作的对象存储在其对应的本地对象存储库中，也可以将每个操作的对象存储在一个或多个共用（共享）对象存储库中。还可以对多个操作使用相同的共享对象存储库。

- 如果您对测试不熟悉，则可能要对每个操作使用本地对象存储库。这是默认设置，并且每个操作的所有对象都将被自动添加到本地存储库中。
- 如果您对测试熟悉，则使用可用于一个或多个操作的共享对象存储库时工作效率通常会更高。对象信息保存在单个中央位置，当应用程序中的对象发生更改时，您可以在此中央位置中为多个测试中的多个操作更新对象。

尽管本教程中未加说明，但您还可以将测试对象从本地对象存储库导出到共享对象存储库，并且可以合并多个对象存储库。

您可能还想创建函数库以增强 UFT 功能。

有关详细信息，请参见《HP Unified Functional Testing 用户指南》。

3. 构建测试。

创建测试步骤时，请按照您希望用户在应用程序中导航时执行的步骤操作。

4. 增强测试。

- 添加检查点，以搜索页面、对象、文本字符串或表单元格的特定值。
- 用参数替换测试中的固定值，以检查应用程序如何使用多组数据执行相同的操作。

可以使用增强测试逻辑性的编程、条件语句和循环语句进一步增强测试。有关详细信息，请参见《HP Unified Functional Testing 用户指南》。

5. 调试测试。

对测试进行调试，以检查它是否顺利运行，未中断。有关详细信息，请参见《HP Unified Functional Testing 用户指南》。

6. 运行测试。

在应用程序上运行测试，以检查应用程序功能是否按预期运行。

7. 分析运行结果。

检查测试的结果，以查明应用程序中的缺陷。（请参考本教程的相应部分，了解在检查点或参数的运行结果中查找哪些方面的缺陷。）

在 Windows 8.X 或更高版本操作系统中访问 UFT

备注: 默认情况下, Windows 8.x 或更高版本中的“开始”和“应用”屏幕设置为以 Metro 模式打开 Internet Explorer。但是,如果计算机上的“用户帐户控制”已关闭,则 Windows 8 将不会以 Metro 模式打开 Internet Explorer。因此,如果尝试从“开始”或“应用”屏幕打开 HTML 快捷方式,例如 UFT 帮助或自述文件,则将显示错误。

要解决此问题,您可以更改 Internet Explorer 的默认行为,以便在以 Metro 模式打开时不会再出现此问题。在 **Internet 属性对话框 > 程序选项卡** 中,在**选择打开链接的方式**选项中选择**始终在桌面上的 Internet Explorer 中**。有关更多详细信息,请参见

<http://support.microsoft.com/kb/2736601> 和

<http://blogs.msdn.com/b/ie/archive/2012/03/26/launch-options-for-internet-explorer-10-on-windows-8.aspx>。

向我们发送反馈



我们可以如何改进教程?
告知我们方式: sw-doc@hp.com

