



HP Unified Functional Testing

소프트웨어 버전: 12.50

Windows[®] 운영 체제

자습서

문서 릴리스 날짜: 2015년 7월

소프트웨어 릴리스 날짜: 2015년 7월

법적 고지

보증

HP 제품 및 서비스에 대한 모든 보증 사항은 해당 제품 및 서비스와 함께 제공된 명시적 보증서에 규정되어 있습니다. 여기에 수록된 어떤 내용도 추가 보증을 구성하는 것으로 해석될 수 없습니다. HP는 여기에 수록된 기술적 또는 편집상의 오류나 누락에 대해 책임지지 않습니다.

본 정보는 예고 없이 변경될 수 있습니다.

제한된 권리 범례

기밀 컴퓨터 소프트웨어, 소유, 사용 또는 복사하기 위해서는 HP로부터 유효한 라이선스를 확보해야 합니다. FAR 12.211 및 12.212에 의거하여 상용 컴퓨터 소프트웨어, 컴퓨터 소프트웨어 문서 및 상용 품목에 대한 기술 데이터는 공급업체의 표준 상용 라이선스 아래에서 미국 정부에 사용이 허가되었습니다.

상표 공고

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

저작권 공고

Adobe® 및 Acrobat® 은 Adobe Systems Incorporated의 상표입니다.

Google™ 및 Google Maps™은 Google Inc.의 상표입니다.

Intel® 및 Pentium® 은 미국 및 기타 국가에서 Intel Corporation의 상표입니다.

Microsoft®, Windows®, Windows® XP 및 Windows Vista® 는 Microsoft Corporation의 미국 등록 상표입니다.

Oracle 및 Java는 Oracle 및/또는 그 자회사의 등록 상표입니다.

설명서 업데이트

이 문서의 제목 페이지에는 다음과 같은 식별 정보가 있습니다.

- 소프트웨어 버전을 의미하는 소프트웨어 버전 번호
- 문서가 업데이트될 때마다 변경되는 문서 릴리스 날짜
- 이 소프트웨어 버전의 릴리스 날짜를 나타내는 소프트웨어 릴리스 날짜

최근 업데이트를 확인하거나 문서의 최신 버전을 사용하고 있는지 확인하려면 다음 사이트로 이동합니다.

<https://softwaresupport.hp.com>.

이 사이트를 사용하려면 HP Passport 사용자로 등록하여 로그인해야 합니다. HP Passport ID를 등록하려면

<https://softwaresupport.hp.com>으로 이동하여 등록을 클릭하십시오.

지원

다음 HP Software Support Online 웹 사이트를 방문하십시오. <https://softwaresupport.hp.com>

이 웹 사이트에서는 연락처 정보를 비롯하여 HP 소프트웨어에서 제공하는 제품, 서비스 및 지원에 대한 자세한 내용을 확인할 수 있습니다.

온라인 지원을 통해 사용자가 스스로 문제를 해결할 수 있습니다. 또한 업무 관리에 필요한 대화식 기술 지원 도구에 신속하고 효율적으로 액세스할 수 있습니다. 소중한 지원 고객으로서 지원 웹 사이트를 통해 다음과 같은 혜택을 누릴 수 있습니다.

- 관심 있는 지식 문서를 검색할 수 있습니다.
- 지원 사례 및 개선 요청을 제출하고 추적할 수 있습니다.
- 소프트웨어 패치를 다운로드할 수 있습니다.
- 지원 계약을 관리할 수 있습니다.
- HP 지원 연락처를 조회할 수 있습니다.
- 사용 가능한 서비스에 대한 정보를 검토할 수 있습니다.
- 다른 소프트웨어 고객과의 토론에 참여할 수 있습니다.
- 소프트웨어 교육을 조사하고 등록할 수 있습니다.

대부분의 지원 영역을 이용하려면 HP Passport 사용자로 등록하여 로그인해야 합니다. 이 영역에서는 지원 계약이 필요할 수도 있습니다. HP Passport ID를 등록하려면 다음 웹 사이트를 방문하십시오.

<https://softwaresupport.hp.com>. 여기에서 등록을 클릭하십시오.

액세스 수준에 대한 자세한 내용을 보려면 다음 웹 사이트를 방문하십시오.

<https://softwaresupport.hp.com/web/softwaresupport/access-levels>.

HP Software Solutions & 통합 및 모범 사례

HP 소프트웨어 카탈로그의 여러 제품이 함께 작동하고 정보를 교환하고 비즈니스 요구 사항을 해결하는 방법에 대해 알아보려면 **HP Software Solutions Now**(<https://h20230.www2.hp.com/sc/solutions/index.jsp>)를 방문하십시오.

다양한 모범 사례 문서와 자료를 활용하려면 **Cross Portfolio Best Practices Library**

(<https://hpln.hp.com/group/best-practices-hpsw>)를 방문하십시오.

목차

UFT 자습서 시작	7
UFT 안내서 및 참조	8
추가 온라인 리소스	9
1부: Unified Functional Testing 소개	11
자동화 테스트의 이점	12
UFT 테스트 프로세스	13
UFT 기본 창	15
2부: 응용 프로그램 분석 및 테스트 만들기	21
응용 프로그램 알아보기	22
Flight Reservation Application 탐색	23
모든 테스트에 대한 솔루션 만들기	25
3부: 자동화 GUI 테스트 만들기 및 실행	27
1단원: GUI 테스트 및 수행 만들기	28
2단원: 개체 저장소 만들기	32
UFT 테스트 개체 인식 - 개요	33
연습 2a: 응용 프로그램의 개체 추가	34
연습 2b: 탐색 및 알아보기를 사용하여 개체 저장소 만들기	40
3단원: 테스트에 단계 추가	44
GUI 테스트에서 테스트 단계 추가 - 개요	45
연습 3a: 키워드 보기에서 로그인 수행에 단계 추가	45
키워드 보기 및 편집기에서 로그인 수행 분석	50
연습 3b: 기록을 통해 FlightFinder 수행에 단계 추가	52
연습 3c: 도구 상자 창을 사용하여 Select Flight 수행에 단계 추가	55
연습 3d: 단계 생성기를 사용하여 Book Flight 수행에 단계 추가	58
고급 연습 3e(선택 사항) - 편집기를 사용하여 단계 추가	63
4단원: GUI 테스트 실행 및 분석	69
연습 4a: 테스트 실행	70
연습 4b: 실행 결과 탐색	72
연습 4c: 실행 결과 분석	74
5단원: 단계 및 개체 매개 변수화	75
테스트, 수행 및 개체 매개 변수화 - 개요	76
연습 5a: 매개 변수화에 대한 테스트 만들기	76
연습 5b: 데이터 테이블 매개 변수 정의	77
연습 5c: 데이터 테이블에 매개 변수 값 추가	81

연습 5d: 매개 변수화된 테스트 실행	83
6단원: 검사점 및 출력 값 만들기	89
검사점 및 출력 값 유형 이해	90
연습 6a: Checkpoint 테스트 만들기	91
연습 6b: 개체 값 확인	93
연습 6c: 테이블 값 확인	96
연습 6d: 텍스트 값 확인	100
연습 6e: 개체 저장소에서 검사점 관리	106
연습 6f: 검사점이 있는 테스트 실행 및 분석	108
연습 6g: 출력 값 테스트 만들기	111
연습 6h: 출력 값 단계 추가	112
7단원: 함수 및 함수 라이브러리 만들기	120
함수 및 함수 라이브러리 - 개요	121
연습 7a: 함수 만들기	121
연습 7b: 테스트에 함수 라이브러리 연결	123
연습 7c: 함수를 사용하여 검사 수행	124
8단원: 테스트에서 이미지 인식 사용	129
이미지 인식 개체 식별 - 개요	130
연습 8a: 이미지 인식 개체에 대한 테스트 만들기	130
연습 8b: 개체 저장소에 이미지 인식 개체 추가	131
연습 8c: 테스트에서 이미지 인식 개체 사용	134
4부: 자동화 API 테스트 만들기 및 실행	137
1단원: API 테스트 만들기	138
2단원: 단순 API 테스트 단계 만들기	139
3단원: 표준 작업을 사용하여 API 테스트 단계 만들기	144
UFT API 테스트 표준 작업 - 개요	145
연습 3a: 표준 작업으로 테스트 만들기	145
4단원: API 테스트 단계 매개 변수화	150
API 테스트 단계 매개 변수화 - 개요	151
연습 4a: 데이터 원본에서 테스트 단계 매개 변수화	151
연습 4b: 이전 단계의 출력에서 테스트 단계 매개 변수화	161
연습 4c: 사용자 지정 식을 사용하여 여러 원본이 있는 테스트 매개 변수화	164
5단원: API 테스트 실행	171
연습 5a: 테스트 실행	172
연습 5b: 실행 결과 탐색	173
연습 5c: 실행 결과 분석	174
6단원: 웹 서비스의 API 테스트 만들기 및 실행	176
연습 6a: 웹 서비스 테스트 만들기	177
연습 6b: 웹 서비스 가져오기	177
연습 6c: 웹 서비스 테스트 빌드 및 매개 변수화	179

연습 6d: 웹 서비스 테스트 실행	185
7단원: REST 서비스의 API 테스트 만들기 및 실행	188
연습 7a: REST 서비스 테스트 만들기	189
연습 7b: REST 서비스 구조 만들기	189
연습 7c: REST 서비스 메서드를 사용하여 테스트 만들기	194
연습 7d: REST 서비스 테스트 실행	196
연습 7e: REST 서비스 충돌 해결	198
8단원: 웹 응용 프로그램 서비스(WADL)의 API 테스트 만들기 및 실행	202
연습 8a: 웹 응용 프로그램 서비스에 대한 테스트 만들기	203
연습 8b: 웹 응용 프로그램 서비스 모델 가져오기	203
연습 8d: 웹 응용 프로그램 서비스 메서드로 테스트 빌드	205
연습 8e: 웹 응용 프로그램 서비스 테스트 실행	209
5부: 단일 테스트에서 GUI 및 API 테스트 만들기 및 실행	212
1단원: GUI 및 API 테스트를 함께 실행할 테스트 만들기	213
2단원: GUI 테스트에서 API 테스트 호출	214
3단원: API 테스트를 호출하는 GUI 테스트 실행	218
다음 단계	220
피드백 보내기	223

UFT 자습서 시작

UFT 자습서는 UFT를 통해 응용 프로그램 테스트에 대한 기본 사항을 알려주는 자가 학습 안내서입니다. 이를 통해 자동화 GUI 및 API 테스트를 만들고 실행하며 실행 결과를 분석하는 프로세스에 친숙해질 수 있습니다.

자습서를 완료하고 나면 사용자가 배운 기술을 고유한 응용 프로그램을 테스트하는 데 적용할 수 있습니다.

참고: 웹 응용 프로그램의 GUI 테스트 만들기 및 실행에 대한 자세한 내용은 <UFT 설치 폴더>\help 폴더에서 사용할 수 있는 웹 응용 프로그램의 UFT GUI 테스트 자습서를 참조하십시오.

자습서 대상 및 범위

이 자습서는 UFT를 처음 사용하는 사용자를 대상으로 합니다. UFT, QuickTest 또는 Service Test에 대한 사전 지식이 필요하지 않습니다. 테스트 개념과 기능 테스트 프로세스에 대한 일반적인 이해가 도움이 될 수 있지만 필수는 아닙니다. UFT를 사용하여 GUI 테스트, API 테스트, 비즈니스 프로세스 테스트 및 GUI/API 테스트나 테스트 호출이 포함된 복합 테스트를 만들 수 있습니다.

참고: 이 자습서에서는 Windows 7 운영 체제에 관련된 파일 시스템 경로를 참조합니다. 다른 운영 체제의 경로는 조금 다를 수 있습니다.

BPT와 함께 UFT 사용

테스트 이외에, UFT에서는 BPT를 지원하는 ALM 서버에 연결된 경우 비즈니스 프로세스 테스트에서 사용할 키워드 기반, 스크립팅된 또는 API 비즈니스 구성 요소를 만들 수 있습니다. 이 자습서에 설명된 절차는 GUI 및 API 테스트를 만들도록 설계되었지만 이러한 절차 대부분은 키워드 기반 구성 요소, 스크립팅된 구성 요소 또는 API 구성 요소를 만드는 데 적용할 수도 있습니다. 비즈니스 구성 요소 및 BPT에 대한 자세한 내용은 *HP Unified Functional Testing User Guide* 및 *HP Business Process Testing User Guide*를 참조하십시오.

참고: 별도로 지정되지 않은 경우 이 안내서의 **Application Lifecycle Management** 또는 **ALM** 참조는 현재 지원되는 모든 버전의 ALM 및 Quality Center에 적용됩니다. 특정 버전의 ALM 또는 Quality Center에서는 일부 기능 및 옵션이 지원되지 않을 수 있습니다.

지원되는 ALM 또는 Quality Center 버전 목록은 [HP Support Matrix 페이지](#)(HP Passport 필요)에서 사용할 수 있는 *HP Unified Functional Testing Product Availability Matrix*를 참조하십시오.

ALM 또는 Quality Center 버전에 대한 자세한 내용은 *HP Application Lifecycle Management 사용자 안내서* 또는 *HP Quality Center User Guide*를 참조하십시오.

UFT 안내서 및 참조

다음 표에서는 UFT 안내서, 온라인 도움말 및 참조 목록을 제공합니다.

참고: 아래 설명서의 최근 업데이트를 확인하려면 HP 소프트웨어 제품 설명서 웹 사이트 (<https://softwaresupport.hp.com/group/softwaresupport/search-result?keyword=>)를 방문하십시오.

시작하기

참조	설명
새로운 기능	Unified Functional Testing 최신 버전의 최신 기능에 대해 설명합니다. Unified Functional Testing 도움말 메뉴에서 새로운 기능 에 액세스할 수도 있습니다.
제품 동영상	링크를 클릭하거나 도움말 > 제품 기능 동영상 을 선택하여 주 제품 기능을 보여주는 짧은 동영상을 볼 수 있습니다.
추가 정보	Unified Functional Testing에 대한 최신 뉴스 및 정보를 제공합니다. 최신 추가 정보 파일을 보려면 HP 소프트웨어 설명서 웹 사이트(HP Passport 필요)(https://softwaresupport.hp.com/group/softwaresupport/search-result?keyword=)로 이동합니다.
UFT PAM	PAM(Product Availability Matrix)은 이 UFT 버전에 대해 지원되는 기술 및 통합에 대한 최신 정보를 제공합니다.
UFT 자습서	UFT 자습서는 복합 응용 프로그램의 GUI, API 및 전체 UFT 테스트를 만드는 프로세스를 스스로 진행하도록 설계된 인쇄 가능한 자가 학습 안내서입니다.
웹 응용 프로그램에 대한 GUI 테스트 자습서	웹 응용 프로그램에 대한 GUI 테스트 자습서는 웹 응용 프로그램의 GUI 테스트를 만드는 프로세스를 스스로 진행하고 테스트 환경에 친숙해지도록 설계된 인쇄 가능한 자가 학습 안내서입니다.

PDF 안내서

안내서	설명
UFT 사용자 안내서	HP Unified Functional Testing 사용자 안내서에서는 UFT를 사용하여 응용 프로그램을 테스트하는 방법에 대해 설명합니다. 테스트를 만들고, 디버깅하며, 실행하고, 테스트 프로세스에서 발견된 결함을 보고하는 단계별 지침을 제공합니다.
Run Results Viewer	HP Run Results Viewer 사용자 안내서에서는 Run Results Viewer를 사용하여 GUI 또는 API 테스트의 테스트 결과를 해석하고 사용하는 방법을 설명합니다.
UFT 설치 안내서	HP Unified Functional Testing 설치 안내서에서는 독립 실행형 컴퓨터에서 UFT를 설치 및 설정하는 방법에 대한 전체 단계별 지침을 제공합니다.
UFT QuickStart	UFT Installation QuickStart Sheet에서는 UFT의 기본 설치를 수행하는 단계를 설명합니다.
UFT Add-ins Guide	HP Unified Functional Testing Add-ins Guide에서는 UFT Add-in 지원 및 표준 Windows 테

안내서	설명
	스트 지원을 설정하는 방법을 설명합니다. Add-in을 통해 GUI 테스트 및 비즈니스 구성 요소를 사용하여 지원되는 환경을 테스트할 수 있습니다.
새로운 기능	Unified Functional Testing 최신 버전의 최신 기능에 대해 설명합니다.
Security Reference	UFT 사용에 대한 보안 관련 정보 및 추가 정보에 대한 참조를 포함합니다.
런타임 엔진 안내서	UFT IDE 없이 런타임 엔진 설치를 사용하여 테스트를 실행하는 방법에 대해 설명합니다.

참조

참조 링크는 UFT 온라인 도움말 홈 페이지에서 사용할 수 있습니다.

참조	설명
개체 모델 참조	GUI 테스트에 대한 개체 모델 참조에는 각 UFT 테스트 개체에 대한 설명, 메서드 및 속성 목록, 구문, 예제, 식별 속성이 포함됩니다.
VBScript 참조	VBScript 스크립트를 작성하는 경우 사용할 수 있는 개체, 메서드, 속성, 함수 및 기타 요소에 대해 설명하는 Microsoft의 Visual Basic Scripting 설명서입니다.
자동화 개체 모델 참조	다른 응용 프로그램 내에서 UFT를 제어할 수 있는 개체, 메서드, 속성을 나열합니다.
개체 저장소 자동화 참조	UFT 외부에서 UFT 공유 개체 저장소 및 해당 콘텐츠를 조작할 수 있는 개체에 대해 설명합니다.
실행 결과 스키마 참조	실행 결과 XML 스키마의 구조에 대한 세부 정보를 제공하고 XML 보고서에서 사용되는 요소 및 특성에 대해 설명합니다.
테스트 개체 스키마 참조	UFT 확장성 프로젝트를 만들 때 사용할 테스트 개체 구성 XML 콘텐츠를 만드는 데 사용할 수 있는 요소 및 특성을 설명하는 참조입니다.
개체 저장소 스키마 참조	개체 저장소 스키마에 대해 정의된 요소 및 복합 유형을 설명합니다.

추가적인 온라인 참조에 대해서는 PDF 안내서의 시작 섹션을 참조하십시오.

추가 온라인 리소스

추가로 사용할 수 있는 온라인 리소스는 다음과 같습니다.

리소스	설명
HP 소프트웨어 온라인 지원	<p>HP 소프트웨어 지원 웹 사이트(www.hp.com/go/hpsoftwaresupport)입니다. 액세스하려면 도움말 > HP 소프트웨어 지원을 선택합니다.</p> <p>참고: 대부분의 지원 영역을 이용하려면 HP Passport 사용자로 등록하여 로그인해야 합니다. 지원 계약이 필요할 수도 있습니다. HP Passport 사용자 ID를 등록하려면 다음 웹 사이트를 방문하십시오. http://h20229.www2.hp.com/passport-registration.html</p>
테스트 포럼	<ul style="list-style-type: none"> GUI 테스트: http://h30499.www3.hp.com/t5/Unified-Functional-Testing/bd-p/sws-Fun_TEST_SF API 테스트: http://h30499.www3.hp.com/t5/Service-Test-Support-and-News/bd-p/sws-Serv_TEST_SF

리소스	설명
	<ul style="list-style-type: none"> BPT: http://h30499.www3.hp.com/t5/Business-Process-Validation/bd-p/sws-BPT_SF
UFT 제품 페이지	HP Unified Functional Testing 제품 페이지(http://www8.hp.com/us/en/software-solutions/unified-functional-testing-automated-testing/index.html)로, UFT에 대한 정보와 관련 링크가 있습니다. 액세스하려면 도움말 > 유용한 링크 > 제품 페이지 를 선택합니다.
문제 해결 및 기술 자료	HP Software Self-solve 기술 자료를 검색할 수 있는 HP 소프트웨어 지원 웹 사이트의 문제 해결 페이지(http://h20230.www2.hp.com/troubleshooting.jsp)입니다. 액세스하려면 도움말 > 기술 자료 또는 도움말 > 문제 해결 을 선택합니다.
HP 소프트웨어 커뮤니티	HP IT 전문가 커뮤니티 사이트(http://h10124.www1.hp.com/campaigns/IT_Experts/pages/home.html)로, 다른 HP 소프트웨어 사용자와 의견을 나누거나 HP 소프트웨어에 대한 자료 및 블로그를 읽고 기타 소프트웨어 제품 다운로드에 액세스할 수 있습니다.
HP 설명서 사이트	HP 소프트웨어 제품 설명서 웹 사이트(http://support.openview.hp.com/selfsolve/manuals)로, 선택한 HP 소프트웨어 제품에 대한 최신 설명서를 검색할 수 있습니다. 액세스하려면 도움말 > 유용한 링크 > HP 설명서 사이트 를 선택합니다.
새로운 기능	이 버전의 UFT에서 제공하는 새 기능 및 향상된 점에 대해 설명하는 UFT 새로운 기능 도움말입니다.
제품 동영상	UFT HPLN(HP Live Networks) 페이지(https://hpln.hp.com/page/uft-120-videos)로, 모든 제품 동영상 목록이 포함됩니다.
HP 소프트웨어 웹 사이트	HP 소프트웨어 웹 사이트(www.hp.com/go/software)입니다. 이 사이트에서는 HP 소프트웨어 제품에 대한 최신 정보를 제공합니다. 새 소프트웨어 릴리스, 세미나 및 박람회, 고객 지원 등의 정보가 여기에 포함됩니다.

1부: Unified Functional Testing 소개

이 단원에서는 자동화 테스트 및 UFT에 대해 개략적으로 살펴봅니다. 테스트 준비도 시작합니다.

참고: 테스트 단계를 복사하여 UFT에 붙여넣도록 요청하는 섹션이 있기 때문에 이 자습서의 소프트 복사본을 사용하는 것이 좋습니다. 하지만 이 자습서의 선택한 부분에서 UFT가 작업을 기록하거나 마우스로 가리키는 개체를 인식할 것임에 유의하십시오. 해당 세션 중에 PDF 또는 도움말 창으로 포커스를 전환하지 않아야 합니다.

이 섹션의 내용은 다음과 같습니다.

· 자동화 테스트의 이점	12
· UFT 테스트 프로세스	13
· UFT 기본 창	15

자동화 테스트의 이점

응용 프로그램이나 웹 사이트를 수동으로 테스트한 적이 있다면 이러한 응용 프로그램에 대한 수동 테스트의 단점을 알고 있을 것입니다.

수동 테스트는 시간이 오래 걸리고 지루하며 인력이 많이 필요한 작업입니다. 가장 나쁜 점은 시간 제약 때문에 응용 프로그램을 릴리스하기 전에 수동으로 모든 기능을 철저히 테스트하는 것이 불가능하다는 것입니다. 이 경우 심각한 버그를 발견하지 못하고 지나칠 우려가 있습니다.

UFT를 사용한 응용 프로그램의 GUI 및 비 GUI(서비스) 레이어 자동화 테스트에서는 테스트 프로세스 속도를 높여서 수동 테스트 관련 문제를 해결합니다. 응용 프로그램이나 웹 사이트의 모든 측면을 확인하는 테스트를 만들고 응용 프로그램이나 웹 사이트가 변경될 때마다 이러한 테스트를 실행할 수 있습니다. 따라서 테스트를 한 번 만들고, 각 응용 프로그램 업데이트에 대한 테스트를 업데이트할 필요 없이 이후에 응용 프로그램이 변경될 때마다 이러한 테스트를 실행합니다.

UFT에서는 이러한 테스트를 실행할 때 응용 프로그램이나 웹 사이트에서 커서를 이동하거나, 응용 프로그램 GUI에서 개체를 클릭하거나, 키보드 입력을 기록하거나, 응용 프로그램 API 프로세스를 실행하여 사용자의 수행을 에뮬레이트합니다. 그러나 수동 테스트와 달리 UFT는 이러한 작업을 사용자보다 더 빨리 수행합니다.

자동화 테스트의 이점은 다음과 같이 다양합니다.

자동화 테스트의 이점	
신속	자동화 테스트는 사용자가 수행하는 수동 테스트보다 훨씬 더 빠릅니다.
안정적	자동화 테스트는 실행할 때마다 동일한 작업을 정확하게 수행하므로 수동 테스트에서 발생하는 사용자 오류가 발생하지 않습니다.
반복 가능	같은 작업을 반복적으로 수행한 후 응용 프로그램이나 웹 사이트가 반응하는 방식을 테스트할 수 있습니다.
프로그램 가능	복잡하지만 중요한 시나리오를 다양하게 테스트하고 일반적인 수동 테스트를 통해 쉽게 식별할 수 없는 문제나 결함을 찾는 정교한 테스트를 프로그래밍할 수 있습니다.
포괄적	응용 프로그램이나 웹 사이트의 모든 기능을 다루는 테스트 집합을 빌드할 수 있습니다.
다시 사용 가능	사용자 인터페이스나 내부 응용 프로그램 API가 변경되더라도 응용 프로그램 또는 웹 사이트의 다양한 버전에서 자동화 테스트를 다시 사용할 수 있습니다.

UFT 테스트 프로세스

UFT 테스트를 만들 때 수행할 다양한 단계는 다음과 같습니다.

1. 응용 프로그램 분석

테스트를 계획하는 첫 번째 단계는 다음과 같이 응용 프로그램을 분석하여 테스트 요구를 결정하는 것입니다.

<p>응용 프로그램 개발 환경은 무엇입니까?</p>	<p>응용 프로그램의 사용자 인터페이스 개체 작업을 하기 위해서는 적합한 UFT GUI 테스트 Add-in을 로드해야 합니다. 예를 들어 응용 프로그램은 웹, .NET 또는 Java 환경에서 빌드될 수 있습니다. 따라서 이러한 환경에서 작업하려면 UFT에 웹, .NET 또는 Java Add-in이 필요합니다.</p>
<p>어떤 비즈니스 프로세스 및 기능을 테스트할 수 있습니까?</p>	<p>이를 위해서는 사용자가 응용 프로그램에서 수행할 작업 및 이러한 비즈니스 프로세스를 수행하기 위해 응용 프로그램이 수행해야 하는 내부 수행을 고려합니다. 사용자 인터페이스에서의 사용자 수행을 모방하는 GUI 테스트 단계를 만듭니다. 응용 프로그램이 백그라운드에서 실행하는 프로세스를 수행하는 API 테스트 단계를 만듭니다.</p>
<p>응용 프로그램에서 표준 응용 프로그램 작업을 사용합니까 아니면 사용자 설계 서비스를 사용합니까?</p>	<p>응용 프로그램 API에서 실행하는 기능에 따라 UFT API 테스트와 함께 제공되는 기본 제공 작업을 사용하거나 사용자 지정 작업을 가져오거나 만듭니다.</p>
<p>테스트를 작은 테스트 가능 단위로 어떻게 구분할 수 있습니까?</p>	<p>GUI 테스트에서 UFT 수행을 만들 수 있도록 테스트할 프로세스 및 기능을 더 작은 작업으로 구분해야 합니다. 이러한 더 작고 많은 모듈식 수행을 통해 테스트를 더욱 쉽게 읽고 추적할 수 있으며 장기 실행 시 쉽게 유지 관리할 수 있습니다.</p>

이 단계에서도 테스트 골격을 만들고 수행을 GUI 테스트에 추가하는 작업을 시작할 수 있습니다.

2. 테스트 인프라 준비

테스트 요구 사항에 따라 어떤 리소스가 필요한지 결정하고 이에 따라 이러한 리소스를 만들어야 합니다.

- GUI 테스트의 경우, 이러한 리소스에는 응용 프로그램에서 개체를 나타내는 테스트 개체가 들어 있는 **공유 개체 저장소**, 테스트에서 사용할 사용자 지정 함수가 들어 있는 **함수 라이브러리**, 응용 프로그램에 문제가 있을 때 응답 방법을 UFT에 지시하는 **복구 시나리오**, 공통 환경 변수에 대한 정의가 들어 있는 **환경 변수 파일** 또는 테스트 단계를 매개 변수화하는 데 사용되는 **외부 데이터 테이블**이 포함됩니다.
- API 테스트의 경우 이러한 리소스에는 응용 프로그램 서비스의 메서드를 설명하는 **WSDL** 또는 **WADL** 파일, 프로토타입 또는 응용 프로그램 REST 프로세스로 사용되도록 만드는 **REST 서비스**, **외부 데이터 원본**, 서비스 호출에서 사용되는 **가상화 프로젝트**, 테스트 단계에서 참조되는 **.NET 어셈블리** 또는 테스트 단계에서 사용되는 **Java 클래스**가 포함됩니다. 이러한 리소스는 UFT에서 가져오거나 만들어야 합니다.

또한 UFT에서 필요한 추가 작업(예: 테스트를 실행할 때마다 결과 보고서 표시, 테스트 실행에 대한 디버깅 사용 여부 설정 등)을 수행하도록 UFT 설정을 구성해야 합니다.

3. 테스트 빌드 및 각 테스트에 단계 추가

테스트 인프라와 리소스가 준비되면 테스트 빌드를 시작할 수 있습니다.

- GUI 테스트에서는 하나 이상의 빈 테스트를 만들고 테스트에 수행을 추가하여 테스트 골격을 만들 수 있습니다. 그런 다음, 개체 저장소를 관련 수행과 연결하여 키워드 기반 방법론이나 스크립트 작성을 통해 단계를 삽입할 수 있습니다.
- API 테스트에서는 하나 이상의 빈 테스트를 만들고, 작업을 테스트 캔버스에 끌어와 이러한 테스트에 테스트 단계를 추가한 후, 해당 단계에 대한 입력, 출력 및 검사점 속성을 정의할 수 있습니다. 여러 번 실행되는 단계를 모아 개별 테스트 단계로 실행될 수 있는 한 번의 수행으로 그룹화할 수도 있습니다.
- 모든 테스트를 단일 솔루션에 추가할 수도 있습니다. 솔루션을 사용하면 다른 테스트를 열기 전에 한 테스트를 닫을 필요 없이 관련 테스트를 함께 저장, 관리 및 편집할 수 있습니다.
- 테스트 기본 설정 및 설정(GUI 테스트)이나 테스트 관련 속성(API 테스트)을 구성할 수도 있습니다.

4. 테스트 향상

다음과 같은 여러 가지 방법으로 테스트를 향상시킬 수 있습니다.

<p>GUI 테스트의 경우...</p>	<ul style="list-style-type: none"> • 검사점을 테스트 단계로 삽입하여 응용 프로그램이 제대로 작동하는지 확인합니다. 예를 들어 이들 검사점에서 페이지, 개별 테스트 개체 또는 텍스트 문자열의 특정 값을 확인할 수 있습니다. • 여러 데이터 집합으로 테스트 단계 값을 매개 변수화하여 응용 프로그램에서 서로 다른 값의 같은 작업을 어떻게 수행하는지 확인합니다. 이 작업을 하려면 테스트 단계의 고정 값을 매개 변수로 바꿉니다. • VBScript를 사용하여 프로그래밍 문과 조건 문 또는 루프 문 및 기타 프로그래밍 논리를 테스트에 추가할 수 있습니다.
<p>API 테스트의 경우...</p>	<ul style="list-style-type: none"> • 검사점 속성을 선택하고 단계 속성의 예상 값을 입력하여 테스트 단계 및 테스트 단계의 개별 속성에 대한 유효성을 검사할 수 있습니다. • 여러 데이터 집합으로 테스트 단계 속성을 매개 변수화하여 응용 프로그램에서 서로 다른 값의 같은 프로세스를 어떻게 수행하는지 확인할 수 있습니다. 이 작업을 하려면 고정 값을 매개 변수로 바꿉니다. • 사용자 지정 코드 작업, 이벤트 처리기 또는 UFT 작업 마법사로 만든 사용자 지정 작업을 통해 테스트 단계에 기능을 추가할 수 있습니다.

5. 테스트 디버깅, 실행 및 분석

UFT의 디버깅 기능을 사용하여 테스트를 디버그하면 중단되지 않고 원활하게 작동되도록 할 수 있습니다. 테스트가 제대로 작동하고 나면 테스트를 실행하여 응용 프로그램의 동작을 확인할 수 있습니다. 실행하는 동안 UFT에서는 응용 프로그램 사용자 인터페이스에서 각 단계를 수행하거나 (GUI 테스트를 실행하는 동안) 응용 프로그램의 API 프로세스를 실행합니다(API 테스트를 실행하는 동안).

6. 결함 보고

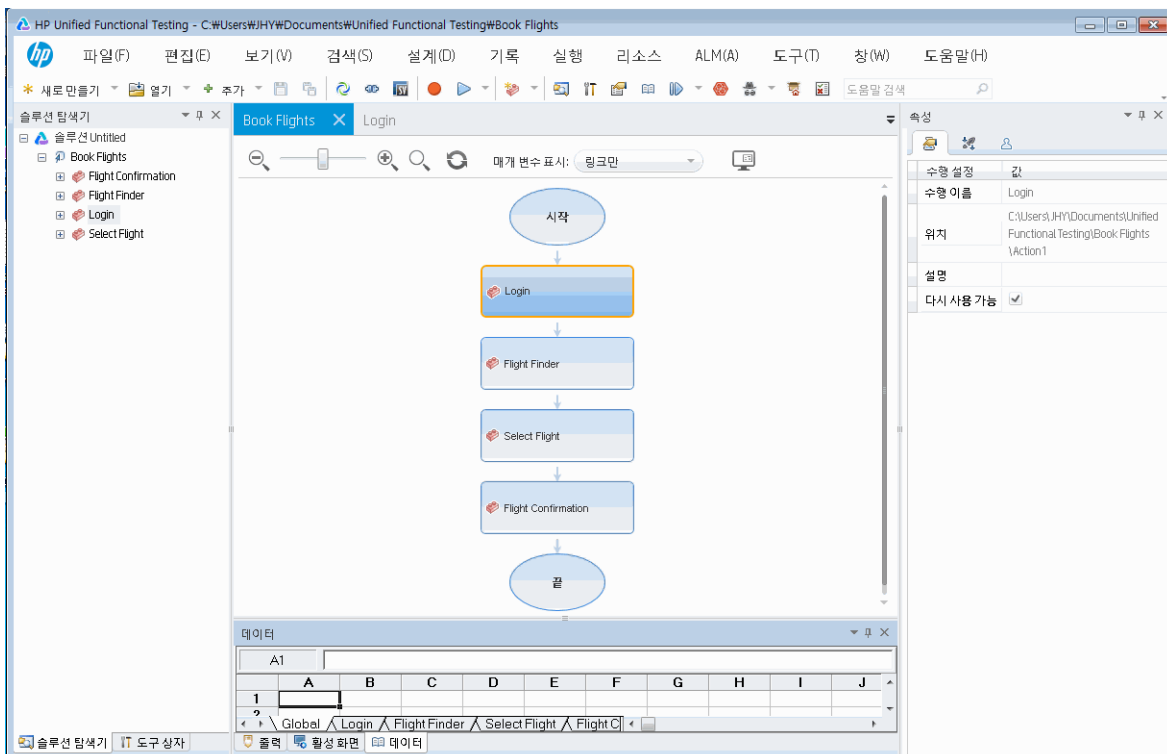
ALM 서버에 액세스할 수 있으면 발견하는 결함을 ALM 프로젝트에 보고할 수 있습니다. 액세스할

수 없으면 결함을 자체 결함 데이터베이스에 수동으로 보고할 수 있습니다.

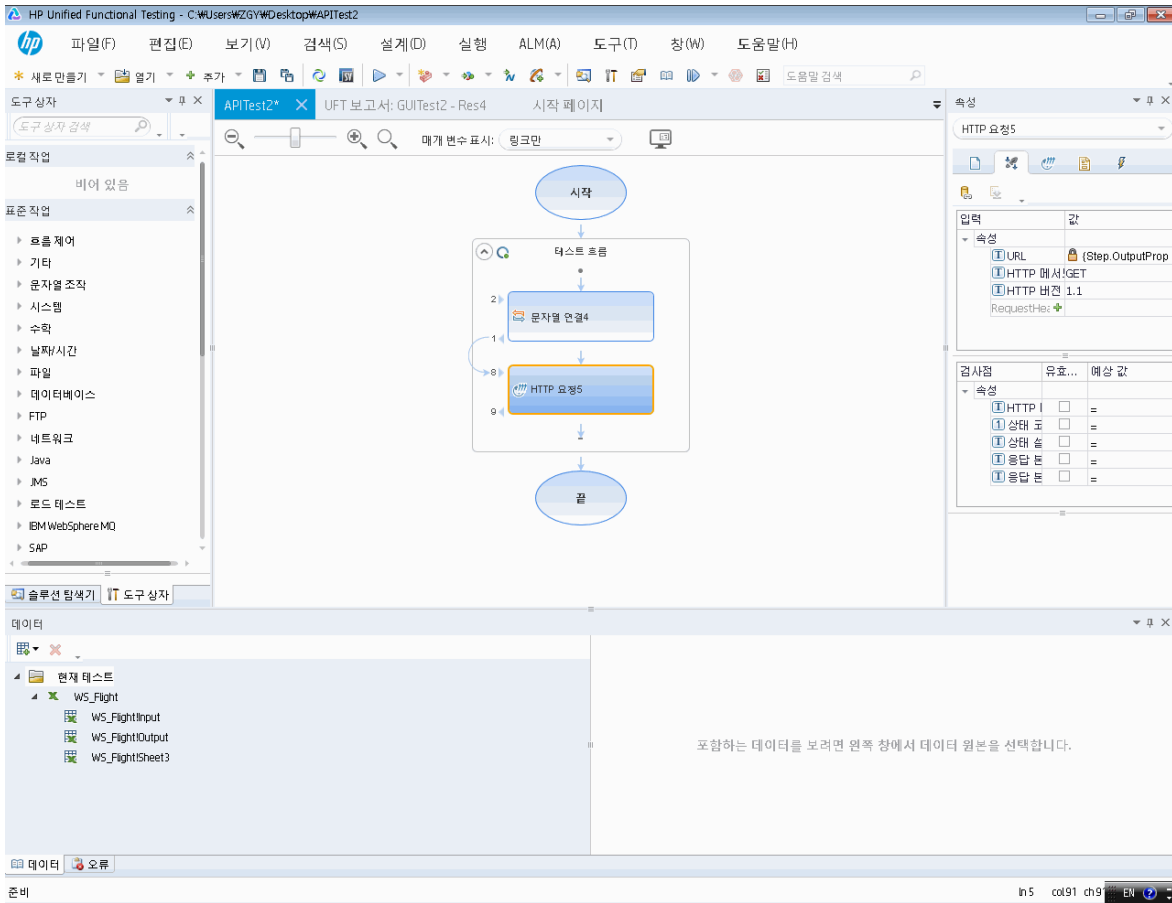
UFT 기본 창

UFT 작업 및 테스트 만들기를 시작하기 전에 기본 UFT 창에 친숙해져야 합니다.

아래 이미지는 GUI 테스트를 만든 후 테스트 흐름이 캔버스에 표시되고 도구 모음, 솔루션 탐색기, 데이터 창, 속성 창이 표시된 UFT 창을 보여줍니다.



아래 이미지는 API 테스트를 만든 후 테스트 흐름이 캔버스에 표시되고 도구 모음, 도구 상자 창, 데이터 창, 속성 창이 표시된 UFT 창을 보여줍니다.



기본 창에 다음과 같은 여러 요소가 표시됩니다.

테스트 문서

UFT에서 문서 창에 열린 문서를 표시합니다(UFT 창 가운데). 도구 모음 바로 아래에 있는 문서 탭을 사용하여 열려 있는 문서로 이동한 후 포커스를 문서에 맞출 수 있습니다.

문서 창에 표시할 수 있는 파일 유형은 다음과 같습니다.

<p>테스트</p>	<p>GUI 및 API 테스트는 둘 다 캔버스에서 자체 탭에 표시됩니다. GUI 테스트에는 테스트에 포함된 각 수행에 대한 개별 항목과 함께 테스트 흐름이 표시됩니다. API 테스트에는 테스트에 포함된 다양한 단계의 테스트 흐름이 표시됩니다.</p> <p>BPT 테스트는 표로 표시되며 테스트에 포함된 개별 구성 요소, 그룹 또는 흐름이 나열됩니다.</p>
<p>GUI 테스트 수행</p>	<p>각 GUI 테스트에는 개별 수행 또는 다른 수행에 대한 호출이 포함됩니다. 다음 보기 중 하나에서 각 수행을 볼 수 있습니다.</p> <ul style="list-style-type: none"> • 키워드 보기: 각 단계(및 개체 계층 구조)는 아이콘 기반의 모듈식 테이블로 표시됩니다. • 편집기: 각 단계는 텍스트/코드 편집기에 표시되는 VBScript 줄로 표시됩니다. 개체 기반 단계에서는 VBScript 줄이 개체 계층 구조를 정의합니다.

<p>비즈니스 구성 요소</p>	<p>비즈니스 구성 요소를 사용하여 응용 프로그램의 각 비즈니스 프로세스에 대한 단일 모듈식 테스트 "단위"를 설계할 수 있습니다. 이러한 구성 요소를 비즈니스 프로세스 테스트에 추가하여 응용 프로그램 시나리오로 함께 실행할 수 있습니다.</p> <p>다음 비즈니스 구성 요소 유형 중 하나를 표시할 수 있습니다.</p> <ul style="list-style-type: none"> • 키워드 GUI 구성 요소: 이러한 구성 요소는 키워드 보기에만 표시됩니다. • 스크립팅된 GUI 구성 요소: 이러한 구성 요소는 키워드 보기 또는 편집기에 표시될 수 있습니다. (기본적으로 편집기에서 열림) • API 구성 요소: 이러한 구성 요소는 테스트 캔버스와 함께 표시됩니다(API 테스트와 같은 방식으로).
<p>함수 라이브러리</p>	<p>함수 라이브러리를 사용하여 테스트에서 사용할 함수를 만들고 편집 및 수정할 수 있습니다. 이러한 함수를 단일 함수 라이브러리에 기록할 수 있으며 그런 다음 함수 라이브러리를 각 테스트와 연결하여 여러 테스트에서 사용할 수 있습니다.</p> <p>함수 라이브러리는 편집기에서 표시됩니다.</p>
<p>응용 프로그램 영역</p>	<p>각 GUI 비즈니스 구성 요소에는 응용 프로그램 영역도 있습니다. 응용 프로그램 영역은 구성 요소의 개체 저장소, 함수 라이브러리 및 구성 설정용 컨테이너로 사용됩니다. 각 응용 프로그램 영역을 여러 비즈니스 구성 요소와 연결할 수도 있습니다.</p> <p>응용 프로그램 영역 사용자 인터페이스는 일련의 사이드바 탭으로 표시됩니다.</p>
<p>사용자 코드 파일</p>	<p>API 테스트에서 특수 이벤트 처리기 코드 또는 사용자 지정 코드 파일을 추가할 수 있습니다. 이 코드를 사용하여 API 테스트 단계의 기본 제공 기능을 보충 및 확장할 수 있습니다. 이벤트 처리기 코드는 기본 API 테스트와 함께 제공된 TestUserCode.cs 파일에 들어 있습니다.</p> <p>이러한 파일은 편집기에 표시됩니다.</p>
<p>시작 페이지</p>	<p>이 페이지에서는 UFT에 대해 소개하고 최근 파일의 링크, 새 기능 설명, 제품 포럼, 기타 지원 링크를 제공합니다. 바로 가기 버튼을 사용하여 문서를 만들거나 기존 문서를 열 수 있습니다.</p>
<p>내부 브라우저 페이지</p>	<p>포럼에 대한 인터넷 페이지를 보고 시작 페이지 또는 도움말 메뉴에서 액세스할 수 있는 자료 등 기타 제품 관련 자료를 볼 수 있습니다.</p>

도구 모음 및 메뉴





문서 창 이외에 UFT 창에 포함된 요소는 다음과 같습니다.

- **제목 표시줄.** 현재 테스트 또는 솔루션의 경로를 표시합니다.
- **메뉴 모음.**
- **UFT 도구 모음.**

창

UFT 창에는 테스트 문서 만들기 및 설계를 지원하도록 설계된 다양한 창이 있습니다.

이러한 일부 창과 도구 모음 옵션은 이후 단원에서 자세히 설명합니다. 기타 창과 도구 모음 옵션에 대한 자세한 내용은 *HP Unified Functional Testing User Guide*를 참조하십시오.

이름	도구 모음 버튼	설명	기본 위치
솔루션 탐색기		솔루션에 포함되거나 현재 열린 모든 테스트, 구성 요소 및 응용 프로그램 영역을 표시하고 현재 테스트 및 구성 요소와 연결된 모든 리소스를 표시합니다. 솔루션 탐색기를 사용하여 이러한 리소스를 관리할 수 있습니다.	UFT 창 왼쪽에 있는 탭입니다.
도구 상자		<ul style="list-style-type: none"> GUI 테스트 및 구성 요소의 경우: 테스트에 사용할 수 있는 모든 키워드를 표시하고 이를 통해 도구 모음 창에서 테스트로 개체 또는 함수 호출을 끌어오 수 있습니다. API 테스트 및 구성 요소의 경우: 테스트에 사용할 수 있는 모든 작업을 표시하며 해당 작업을 캔버스에 끌어오 수 있습니다. 	UFT 창 왼쪽에 있는 탭입니다.
문서 창	해당 없음	열려 있는 문서를 모두 표시합니다. 각 문서에는 클릭하여 문서에 포커스를 맞출 수 있는 탭이 있습니다.	UFT 창의 가운데에 있는 레이블이 지정되지 않은 창입니다. 각 문서 탭은 문서 이름으로 레이블이 지정되어 있습니다. 표시하려면: 테스트 문서를 엽니다.
속성		<p>GUI 테스트 및 구성 요소의 경우: 현재 선택된 테스트, 수행, 구성 요소 또는 응용 프로그램 영역에 대한 모든 속성을 표시합니다.</p> <p>API 테스트 및 구성 요소의 경우: 선택한 테스트 단계/테스트 흐름 또는 선택한 데이터 원본에 대한 모든 속성(데이터 창)을 표시합니다.</p>	UFT 창 오른쪽에 있는 창입니다. 표시하려면 다음을 수행합니다. <ul style="list-style-type: none"> 보기 > 속성을 선택합니다. 도구 모음에서 속성 버튼을 클릭합니다. 캔버스에서 API 테스트 단계를 두 번 클릭합니다. 캔버스에서 API 테스트 단계를 마우스 오른쪽 버튼으로 클릭하고 속성을 선택합니다.
데이터		테스트를 매개 변수화하도록 지원합니다.	UFT 창 아래쪽에 있는 탭입니다.
출력	해당 없음	실행 세션 동안의 정보를 표시합니다.	UFT 창 아래쪽에 있는 탭입니다. 표시하려면 보기 > 출

			력 을 선택합니다.
오류	해당 없음	테스트 또는 구성 요소와 관련된 문제 목록, 즉 테스트에서의 참조 누락(예: GUI 테스트에서 개체 저장소 또는 복구 시나리오 누락, 외부 파일에 대한 참조 누락 또는 API 테스트에 대한 속성 값 누락)과 같은 문제를 표시합니다.	UFT 창 아래쪽에 있는 탭입니다. 표시하려면 보기 > 오류 를 선택합니다.
활성 화면	해당 없음	기록 세션 동안 특정 단계를 수행할 당시의 응용 프로그램에 대한 스냅샷을 제공합니다. 이 창은 API 테스트에 사용되지 않습니다.	UFT 창 아래쪽에 있는 탭입니다. 표시하려면 보기 > 활성 화면 을 선택합니다.
디버그 창		테스트를 디버깅하도록 지원합니다. 다음과 같이 다양한 디버그 창이 있습니다. <ul style="list-style-type: none"> • 중단점 • 호출 스택 • 로컬 변수 • 콘솔 • 조사 • 스레드(API 테스트에만 해당) • 로드된 모듈(API 테스트에만 해당) 	UFT 창 아래쪽에 있는 탭입니다.
작업	해당 없음	현재 테스트에 대해 정의된 작업을 표시하고 관리할 수 있습니다. 이 창에는 테스트 수행, 함수 라이브러리 또는 사용자 코드 파일의 TODO 주석 단계가 표시됩니다.	UFT 창 아래쪽에 있는 탭입니다. 표시하려면 보기 > 작업 을 선택합니다.
검색 결과	해당 없음	찾기 대화 상자나 다른 검색 메뉴 항목을 사용하여 정의한 검색 기준과 일치하는 모든 항목을 표시합니다.	UFT 창 아래쪽에 있는 탭입니다. 표시하려면 다음을 수행합니다. <ul style="list-style-type: none"> • 보기 > 검색 결과를 선택합니다. • 검색을 수행합니다.
책갈피	해당 없음	테스트 문서에 포함된 책갈피 목록 및 위치를 표시합니다.	UFT 창 아래쪽에 있는 탭입니다. 표시하려면 보기 > 책갈피 를 선택합니다.
단계 실행 결과	해당 없음	개별 API 테스트 단계에 대한 테스트 실행의 실행 결과를 표시합니다. 이 창은 GUI 테스트에 사용되지 않습니다.	UFT 창 아래쪽에 있는 탭입니다. 표시하려면 다음을 수행합니다. <ul style="list-style-type: none"> • 보기 > 실행 결과를 선택합니다.

			<ul style="list-style-type: none">• API 테스트 단계를 마우스 오른쪽 버튼으로 클릭하고 단계 실행을 선택하여 단계를 실행합니다.
--	--	--	---

2부: 응용 프로그램 분석 및 테스트 만들기

"Unified Functional Testing 소개"(11페이지)에서는 자동화 테스트 및 UFT에 대해 개략적으로 살펴봅니다.

이 단원에서는 응용 프로그램을 분석하여 테스트할 사항을 확인합니다.

이 섹션의 내용은 다음과 같습니다.

- 응용 프로그램 알아보기 22
- Flight Reservation Application 탐색 23
- 모든 테스트에 대한 솔루션 만들기 25

응용 프로그램 알아보기

응용 프로그램의 테스트 만들기를 시작하기 전에 응용 프로그램에서 테스트할 사항을 정확히 결정해야 합니다. 이를 위해서는 특정 작업을 완료하기 위해 응용 프로그램에서 수행하는 별도의 작업인 응용 프로그램 프로세스 측면에서 응용 프로그램을 분석해야 합니다.

이 자습서에서는 항공편 예약 응용 프로그램을 테스트합니다. 이 응용 프로그램은 항공편 정보 및 예약 서비스를 에뮬레이트합니다.

응용 프로그램은 다음 두 가지 구성 요소로 구성됩니다.

- **Book Flights 레이어.** 이 레이어는 응용 프로그램의 사용자 인터페이스이고 항공편을 찾아서 예약하는 데 사용되는 인터페이스를 제공합니다.
- **Flights API 레이어.** 이 응용 프로그램은 항공편 예약 응용 프로그램의 서비스(API) 쪽을 제공합니다. 서비스의 메서드 및 작업에 대한 자세한 내용은 Flights API 창에서 **도움말 페이지 열기** 버튼을 클릭합니다.

Book Flights 레이어를 사용하여 응용 프로그램의 GUI 테스트를 만듭니다. Flights API 레이어를 사용하여 응용 프로그램의 API 테스트를 만듭니다.

항공편 예약 응용 프로그램의 테스트를 계획할 때 다음을 고려하십시오.

<p>테스트해야 할 비즈니스 프로세스</p>	<p>이는 기초적인 질문입니다. 다음을 고려하십시오.</p> <ul style="list-style-type: none"> • 응용 프로그램이 수행해야 하는 프로세스는 무엇입니까? • 결정한 프로세스를 토대로 이러한 프로세스를 수행하기 위해 사용자가 사용할 수행은 무엇입니까?
<p>응용 프로그램은 어떻게 구성됩니까?</p>	<ul style="list-style-type: none"> • 각 사용자 작업에 대한 응용 프로그램의 개별 섹션/페이지/모듈이 있습니까? • 사용자가 수행할 수 있는 작업은 무엇입니까? • 응용 프로그램에서 이러한 섹션/페이지/모듈은 어디에 있습니까? • 이러한 사용자 작업의 예상 결과는 무엇입니까? • 이러한 사용자 작업을 지원하는 백그라운드 프로세스는 무엇입니까?
<p>각 응용 프로그램의 섹션/페이지/모듈에서 어떤 사용자 인터페이스 요소를 테스트해야 합니까?</p>	<ul style="list-style-type: none"> • 각 영역에서 어떤 사용자 인터페이스 개체를 테스트해야 합니까? • 테스트에서 어떤 사용자 수행을 시뮬레이트해야 합니까?
<p>여러 시나리오에서 어떤 활동을 사용할 수 있습니까?</p>	<p>어떤 특정 사용자 수행 또는 응용 프로그램 프로세스를 반복적으로 수행합니까? 예를 들어 응용 프로그램에 로그인하는 작업은 사용자가 반복적으로 수행하는 작업일 수 있으며, 사용자 자격 증명 데이터베이스에 연결하는 작업은 응용 프로그램에서 반복적으로 수행하는 작업입니다.</p>
<p>테스트를 위해 어떤 개발 환경을 지원해야 합니까?</p>	<p>UFT에서는 다양한 테스트 환경을 지원하기 위한 Add-in을 제공합니다. 응용 프로그램에 적절한 테스트 환경을 로드하려면 UFT가 기술을 지원하도록 응용 프로그램 개발에 사용되는 기술을 고려해야 합니다. 또한 UFT를 열고 테스트를 만들 때 이러한 Add-in을 로드해야 합니다. 이렇게 하면 UFT가 테스트를 만들고 실행할 때 응용 프로그램에서 개체를 제대로 인식합니다.</p>

Flight Reservation Application 탐색

테스트 만들기를 시작하기 전에 샘플 응용 프로그램을 탐색하고 응용 프로그램에 포함된 사용자 수행과 응용 프로그램 흐름을 지원하는 프로세스를 확인해야 합니다.

응용 프로그램을 탐색하고 사용할 때 [이전 단원](#)에서 제시된 질문을 고려하십시오. 이러한 질문을 사용하여 응용 프로그램에서 테스트를 만드는 방법을 생각해 볼 수 있습니다.

1. 항공편 예약 응용 프로그램을 엽니다.

Book Flights(GUI) 레이어와 Flights API(서비스) 레이어를 둘 다 엽니다.

- Book Flights 레이어는 [시작 > 모든 프로그램 > HP 소프트웨어 > HP Unified Functional Testing > Sample Applications > Flight GUI](#)에서 사용할 수 있습니다.
- Flights API 레이어는 [시작 > 모든 프로그램 > HP 소프트웨어 > HP Unified Functional Testing > Sample Applications > Flight API](#)에서 사용할 수 있습니다.

Windows 8 및 Windows Server 2012에서 UFT 및 UFT 도구와 파일에 액세스하는 방법은 "[Windows 8.X 이상 운영 체제에서의 UFT 액세스](#)"(222페이지)를 참조하십시오.

2. Book Flights 응용 프로그램에 로그인합니다.

Book Flights 응용 프로그램 시작 페이지에서 사용자 이름으로 John을 입력하고 비밀번호로 hp를 입력합니다.

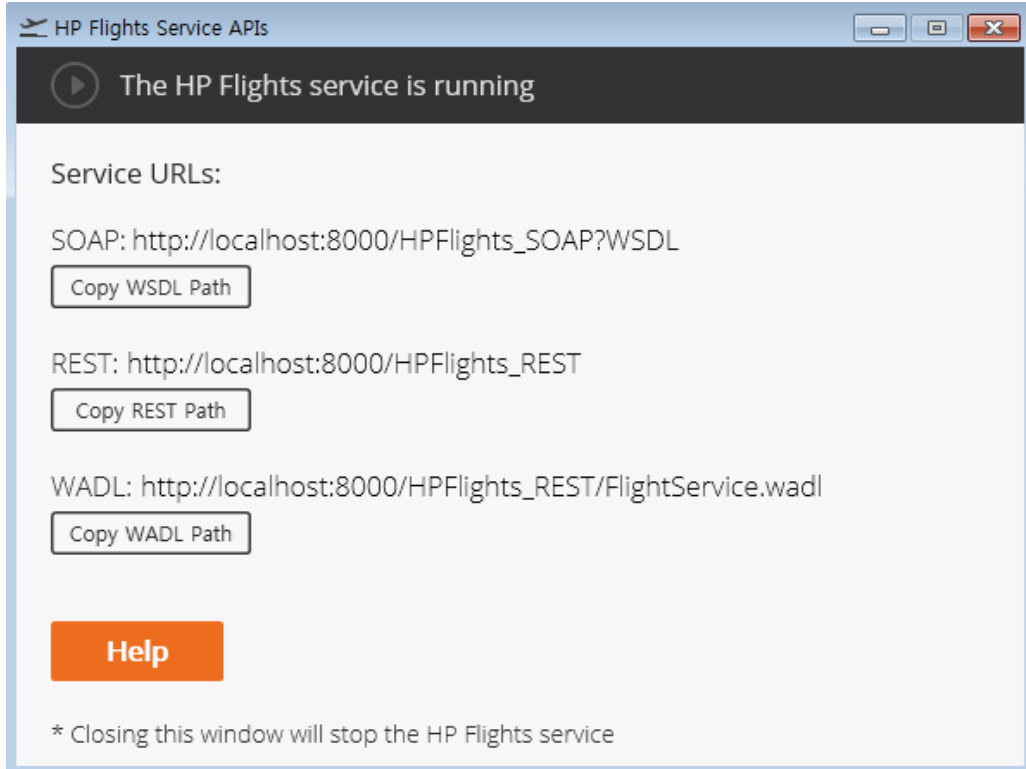
참고: Flights API 레이어에는 로그인 정보가 필요하지 않습니다.

성공적으로 로그인한 후 응용 프로그램 레이아웃 표시는 다음과 같아야 합니다.

Book Flights(GUI 레이아웃)



Flights API 레이아웃



3. 응용 프로그램 레이어를 탐색합니다.

Book Flights 레이어의 경우...	예약 프로세스에 따라 각 페이지에서 요청된 정보 또는 선택 사항을 입력합니다. 응용 프로그램을 탐색할 때 어떤 수행을 테스트할지 그리고 테스트를 설정하려면 어떤 개체를 만들어야 하는지 고려하십시오.
Flights API 레이어의 경우...	도움말 버튼을 클릭하여 응용 프로그램에 포함된 메서드에 대한 설명을 확인합니다. 응용 프로그램에 포함된 메서드 목록을 검색하면서 각 메서드에 제공된 속성 세부 정보를 확인합니다. 나중에 이러한 메서드에 대한 속성 값을 제공하는 테스트를 설계할 때 이 데이터를 사용할 수 있습니다.

4. 응용 프로그램 찾아보기 세션을 종료합니다.

- **Book Flights 레이어의 경우:** Order Details 창에 항공편 예약 주문 세부 정보가 표시된 후 **New Search**를 클릭하면 응용 프로그램이 시작 화면으로 돌아갑니다.
- **Flights API 레이어의 경우:** Flights API 창을 최소화합니다. 서비스가 중단되므로 이 창을 닫지 않도록 합니다.

이제 이러한 응용 프로그램을 사용하여 응용 프로그램의 각 레이어에 대한 테스트를 만들 준비가 되었습니다. "[모든 테스트에 대한 솔루션 만들기](#)"(25페이지)로 이동하여 이 자습서의 과정에서 사용할 테스트에 대한 솔루션을 만듭니다.

모든 테스트에 대한 솔루션 만들기

UFT에서 테스트용 컨테이너로 사용할 **솔루션**을 만들 수 있습니다. 솔루션이 만들어지면 테스트, 비즈니스 구성 요소 또는 응용 프로그램 영역을 솔루션에 추가할 수 있습니다.

솔루션을 사용하면 편리하게 테스트를 함께 그룹화할 수 있습니다. 예를 들어 특정 응용 프로그램의 모든 테스트가 포함된 솔루션을 만들거나, 응용 프로그램 중 일부의 GUI 또는 API 테스트만 포함된 솔루션을 만들 수 있습니다.

모든 유형의 UFT 문서를 솔루션에 추가할 수 있고 테스트를 여러 솔루션에 추가할 수 있습니다.

이 연습에서는 이 자습서 과정에서 만들 테스트를 포함할 새 솔루션을 만듭니다.

1. UFT를 시작합니다.

다음 중 하나를 수행합니다.

UFT가 현재 열려 있지 않은 경우	<ol style="list-style-type: none"> UFT를 엽니다. <ul style="list-style-type: none"> 바탕 화면이나 시작 화면(Windows 8.x 또는 Windows Server 2012)에서 UFT 아이콘을 두 번 클릭합니다. Windows 7 또는 Windows Server 2008 R2에서 시작 > 모든 프로그램 > HP 소프트웨어 > HP Unified Functional Testing을 선택합니다. Add-in 관리자에서 WPF Add-in만 선택되었는지 확인합니다. 기타 Add-in을 모두 지웁니다. 확인을 클릭하여 Add-in 관리자를 닫고 UFT를 엽니다.
----------------------------	---

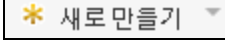
	UFT에서 선택된 Add-in을 로드하는 동안 UFT 시작 화면이 표시됩니다.
UFT가 현재 열려 있는 경우	<p>a. 도움말 > HP Unified Functional Testing 정보를 선택하여 로드되는 Add-in을 확인합니다. 정보 대화 상자에서 로드된 Add-in 이름 옆에 확인란이 표시됩니다.</p> <p>b. WPF Add-in이 로드되지 않으면 UFT를 종료하고 다시 시작해야 합니다. Add-In 관리자가 다시 열리면 WPF를 선택합니다.</p> <p>UFT를 시작할 때 Add-in 관리자가 열리지 않으면 UFT가 열린 후 도구 > 옵션을 선택합니다. 그리고 나서 시작 옵션 창(도구 > 옵션 > 일반 탭 > 시작 옵션 노드)에서 시작 시 Add-in 관리자 표시 옵션을 선택합니다.</p>


2. 시작 페이지를 살펴봅니다.

시작 페이지에는 최근에 사용한 파일 링크, UFT의 새로운 기능에 관한 정보를 비롯하여 유용한 지원 및 커뮤니티 포럼에 대한 링크가 제공됩니다. 페이지의 오른쪽 위 모서리에서 UFT를 사용하는 동안 시작 페이지 표시 및 닫기에 대한 옵션을 정의할 수 있습니다.

참고: 시작 페이지가 표시되지 않으면 **보기 > 시작 페이지**를 선택하여 시작 페이지를 표시할 수 있습니다.

3. 새 솔루션을 만듭니다.

- 도구 모음에서 **새로 만들기** 아래쪽 화살표 를 클릭하고 **새 솔루션**을 선택합니다.
- 솔루션에 대한 세부 정보를 입력합니다.
 - 이름:** Flight Reservation Application
 - 위치:** 기본적으로 모든 솔루션과 테스트는 **C:\%HOMEPATH%\My Documents\Unified Functional Testing**에 저장됩니다. 이 자습서에서는 이 경로를 수정할 필요가 없습니다.
- 만들기**를 클릭합니다.
이제 솔루션 탐색기 창에서 솔루션 이름이 창의 위쪽에 표시됨을 알 수 있습니다. 이 솔루션에 추가하는 테스트는 이 솔루션의 하위 노드로 표시됩니다.

참고: 솔루션 탐색기가 숨겨져 있는 경우에는 도구 모음에서 솔루션 탐색기 버튼 을 클릭하거나 **보기 > 솔루션 탐색기**를 선택하여 표시합니다.

이제 "**자동화 GUI 테스트 만들기 및 실행**"(27페이지)의 설명대로 GUI 테스트를 만들거나 "**자동화 API 테스트 만들기 및 실행**"(137페이지)의 설명대로 API 테스트를 만들기 시작할 수 있습니다.

3부: 자동화 GUI 테스트 만들기 및 실행

응용 프로그램을 분석하고 테스트 목적을 계획한 후에는 응용 프로그램 테스트를 만듭니다. 이 작업의 주요 부분은 응용 프로그램의 사용자 인터페이스(GUI) 테스트를 만드는 것입니다. 이 작업을 수행하면 응용 프로그램의 컨트롤 및 개체가 설계대로 작동하게 됩니다.

GUI 테스트 만들기에는 다음과 같은 다양한 개별 프로세스가 포함됩니다.

- 응용 프로그램에서 개체에 대한 테스트 개체가 포함된 개체 저장소 만들기
- 테스트에서 함수 라이브러리 내부에 사용할 보조 함수 만들기
- 응용 프로그램에서 사용자 수행을 나타내는 단계를 테스트에 추가
- 응용 프로그램에서 특정 개체의 유효성을 검사하기 위한 검사점 만들기
- 테스트 개체 값을 매개 변수화하여 응용 프로그램이 여러 입력 값에 반응하는 방식 확인
- 테스트 실행 및 테스트 실행 결과 분석

다음 단원에서는 이러한 프로세스를 소개하고 자세히 설명합니다.

이 섹션의 내용은 다음과 같습니다.

• 1단원: GUI 테스트 및 수행 만들기	28
• 2단원: 개체 저장소 만들기	32
• 3단원: 테스트에 단계 추가	44
• 4단원: GUI 테스트 실행 및 분석	69
• 5단원: 단계 및 개체 매개 변수화	75
• 6단원: 검사점 및 출력 값 만들기	89
• 7단원: 함수 및 함수 라이브러리 만들기	120
• 8단원: 테스트에서 이미지 인식 사용	129

1단원: GUI 테스트 및 수행 만들기

응용 프로그램의 GUI를 테스트할 단계를 만들기 전에 테스트를 만들고 테스트 구조를 제공하는 수행을 만들어야 합니다.

각 UFT GUI 테스트는 수행에 대한 호출로 구성됩니다. 수행은 테스트를 논리적 섹션으로 구분하는 테스트 내의 단위입니다. 테스트를 여러 수행으로 구분하면 더욱 이해하기 쉽고 효율적인 모듈식 테스트를 설계할 수 있습니다.

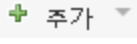
테스트에는 다양한 유형의 수행이 포함될 수 있습니다.

<p>내부 및 외부 수행</p>	<ul style="list-style-type: none"> • 내부 수행은 로컬 테스트(원본 테스트라고도 함)에 저장된 수행입니다. • 외부 수행은 다른 테스트에 저장된 수행에 대한 참조된 호출입니다. <p>참고: 테스트에 의해 호출된 외부 수행은 솔루션 탐색기에서 테스트 노드 아래 개별 노드로 표시됩니다.</p> <p>예를 들어 여러 테스트에서 사용할 수행이 있으면 하나의 테스트에서 수행을 내부 수행으로 저장하고 다른 테스트에서 해당 수행에 대한 호출을 삽입합니다. 수행을 호출하는 다른 테스트에서 해당 수행을 외부 수행으로 사용할 수 있습니다.</p>
<p>다시 사용할 수 있는 수행</p>	<p>새 수행에 대한 호출을 삽입하면 해당 호출은 기본적으로 다시 사용 가능하므로 모든 테스트에서 수행을 호출할 수 있습니다.</p> <p>다시 사용할 수 있는 수행을 사용할 때 원래 테스트와 함께 저장된 기존 수행을 업데이트하지만 하면 됩니다. 원래 수행을 수정하면 수행에 대한 호출을 포함하는 모든 테스트가 업데이트됩니다. 다시 사용할 수 있는 수행을 호출하는 테스트에서 이러한 다시 사용할 수 있는 수행은 읽기 전용입니다.</p> <p>다시 사용할 수 있는 수행은 다음과 같은 경우에 유용할 수 있습니다.</p> <ul style="list-style-type: none"> • 응용 프로그램 로그인과 같이 여러 테스트에 포함해야 할 수 있는 프로세스가 있는 경우 • 사용자가 응용 프로그램의 보안 부분에 액세스하려고 할 때마다 열리는 사용자 자격 증명 대화 상자에 입력하는 등 같은 테스트에 여러 번 삽입해야 하는 프로세스가 있는 경우 <p>수행이 다른 테스트에서 사용되지 않게 하려면 수행을 다시 사용 불가능으로 설정합니다.</p> <p>팁: 같거나 비슷한 수행을 여러 테스트에서 사용하려면 다시 사용 가능한 수행을 저장할 저장소 테스트 만들기를 고려할 수 있습니다. 그런 다음 다른 테스트에서 이러한 기존 수행에 대한 호출을 삽입합니다.</p>
<p>복사된 수행</p>	<p>수행 단계를 수정하려는 경우 테스트에 수행 복사본을 삽입할 수도 있습니다. 수행을 복사하면 해당 수행은 복사된 대상 테스트의 내부 수행이 됩니다. 이러한 복사본은 원본 테스트에 링크되어 있지 않으므로 원래 수행의 모든 변경 사항이 복사본에서 업데이트되지 않습니다.</p>

다음 방법으로 테스트에 수행을 추가합니다.

- **새 수행 호출.** 새롭고 빈 수행을 테스트에 삽입합니다. 수행을 포함하는 테스트의 내부 수행입니다.
- **수행 복사본 호출.** 기존 수행의 복사본을 테스트에 삽입합니다.
- **기존 수행 호출.** 기존 수행에 대한 호출을 삽입합니다(현재 테스트에 대한 외부 수행).

1. 새 테스트를 만들고 솔루션에 테스트를 추가합니다.

- a. UFT 도구 모음에서 **추가** 버튼 아래쪽 화살표  를 클릭하고 **새 테스트 추가**를 선택합니다.
- b. 솔루션에 새 테스트 추가 대화 상자에서 **GUI 테스트**를 선택합니다.
- c. 테스트에 대해 다음 세부 정보를 입력합니다.

- **이름:** **Book Flights**
- **위치:** 기본적으로 UFT에서는 문서를 **C:\%HOMEPATH%\My Documents\Unified Functional Testing**에 저장합니다. 이 단원에서는 이 경로를 수정할 필요가 없습니다.

- d. **추가**를 클릭합니다.

테스트 흐름에 대한 탭이 있는 빈 테스트가 캔버스에서 열리고(이름: **Book Flights**) 빈 수행 탭이 문서 창에서 열립니다(이름: **Action 1**).

이 테스트는 솔루션 탐색기 창에서 Flight Reservation Application 솔루션 노드의 하위 노드으로도 표시됩니다.

2. 논리 이름을 가지도록 **Action 1** 이름을 바꿉니다.

- a. 캔버스에서 **Action1** 상자를 마우스 오른쪽 버튼으로 클릭하고 **수행 속성**을 선택합니다.
- b. 수행 속성 대화 상자의 일반 탭에서 **Action1** 이름을 **Login**으로 바꾸고 **확인**을 클릭합니다.
- c. 경고가 열리면 **예**를 클릭합니다.

이제 캔버스의 수행 블록이 **Login**이라는 이름으로 표시됩니다.


3. **Flight Finder**라는 새 수행을 만듭니다.

- a. 캔버스에서 아무 곳이나 마우스 오른쪽 버튼으로 클릭하고 **새 수행 호출**을 선택합니다.

새 수행 호출 삽입 대화 상자가 열립니다.

- b. 이름 상자에 수행 이름으로 **Flight Finder**를 입력합니다.
- c. **다시 사용할 수 있는 수행 및 테스트의 끝** 옵션이 선택되었는지 확인합니다.
- d. **확인**을 클릭합니다.
이름이 **Flight Finder**인 다른 블록이 캔버스에 추가됩니다.

4. **추가 수행을 테스트에 추가합니다.**

- a. UFT 도구 모음에서 **새 수행 호출 삽입** 버튼  을 클릭합니다.
- b. 새 수행 호출 삽입 대화 상자에서 이름 상자에 **Select Flight**를 입력합니다.
- c. **다시 사용할 수 있는 수행 및 테스트의 끝** 옵션이 선택되었는지 확인합니다.
- d. **확인**을 클릭하여 수행을 테스트에 추가합니다.
이름이 **Select Flight**인 또 다른 블록이 캔버스에 추가됩니다.

5. **최종 수행을 만듭니다.**

위에서 사용된 메서드를 사용하여 **Flight Confirmation**이라는 또 다른 수행을 테스트에 추가합니다.
이제 테스트에 응용 프로그램을 테스트하는 데 필요한 모든 수행이 포함되어 있습니다.

6. **테스트를 저장합니다.**


UFT 도구 모음에서 **저장** 버튼  을 클릭합니다.

저장을 클릭하기 전에 문서 창의 Book Flights 탭과 솔루션 탐색기의 Book Flights 노드에는 별표(*)가 표시됩니다. 이러한 별표는 문서에 저장되지 않은 콘텐츠가 있음을 나타내기 위해 표시됩니다. 테스트를 저장하면 수행 탭의 모든 변경 사항도 저장됩니다.

7. 캔버스에서 테스트 흐름을 살펴봅니다.

캔버스에서 **Book Flights** 탭을 클릭하여 전체 테스트 흐름을 봅니다.

- 마우스 또는 캔버스 위에 있는 확대/축소 막대를 사용하여 테스트 흐름을 확대 또는 축소할 수 있습니다.
- 테스트 흐름이 최대 크기로 확대되고 캔버스에 더 이상 맞춰지지 않으면 캔버스 오른쪽 아래에 있는 미니 맵을 사용하여 테스트 흐름을 탐색할 수도 있습니다.

미니 맵이 표시되지 않으면 캔버스 위에 있는 미니 맵 버튼  을 클릭하여 표시합니다. 미니 맵은 테스트에 많은 수행이나 중첩된 수행이 포함된 경우 유용할 수 있습니다.

테스트 구조를 만들었으므로 이제 테스트에 대한 개체 저장소 만들기를 시작할 수 있습니다. 계속하여 ["2단원: 개체 저장소 만들기"\(32페이지\)](#)로 이동하십시오.

2단원: 개체 저장소 만들기

GUI 테스트는 응용 프로그램 사용자 인터페이스를 테스트하는 데 사용되는 테스트 개체 컬렉션에 기반을 둡니다. UFT에서는 이러한 개체를 알아보고 테스트와 연결된 개체 저장소에 저장합니다.

수행을 만들어서 테스트와 해당 테스트 구조를 만들었으므로 이제 테스트에서 사용할 테스트 개체를 만들어야 합니다. 이 단원에서는 테스트에서 사용되는 테스트 개체, 런타임 개체 및 개체 저장소의 기본 개념을 소개합니다.

이 단원의 내용은 다음과 같습니다.

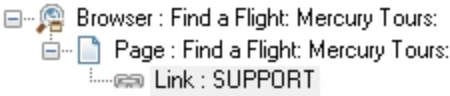
- [UFT 테스트 개체 인식 - 개요](#) 33
- [연습 2a: 응용 프로그램의 개체 추가](#) 34
- [연습 2b: 탐색 및 알아보기를 사용하여 개체 저장소 만들기](#) 40

UFT 테스트 개체 인식 - 개요

GUI 테스트를 만들고 실행할 때 UFT에서는 **테스트 개체**를 사용하여 응용 프로그램에서 개체를 인식하고 응용 프로그램 개체에 따라 테스트 단계를 만듭니다. 이러한 테스트 개체는 UFT의 **테스트 개체 모델**에 기반을 둡니다.

테스트 개체 모델은 UFT가 응용 프로그램에서 개체를 표현하는 데 사용하는 커다란 개체 유형 또는 클래스 집합입니다. 각 테스트 개체 클래스마다 UFT에서 개체에 대해 알아볼 수 있는 식별 속성 목록, 해당 클래스에서 개체를 고유하게 식별할 수 있는 이러한 속성의 부분 집합, 그리고 UFT에서 개체에 대해 수행할 수 있는 관련 작업 집합이 있습니다.

테스트를 설계 및 실행할 때 개체에는 두 가지 유형이 있습니다.

<p>테스트 개체</p>	<p>테스트 개체는 UFT가 응용 프로그램에서 실제 개체를 표현하기 위해 만드는 저장된 표현입니다. UFT에서는 응용 프로그램에서 개체 속성 및 값의 선택 집합을 인식하여 테스트 개체를 만듭니다. 그리고 나서 UFT에서는 실행 세션 중에 개체를 식별하고 확인하는 데 도움이 될 개체 정보를 저장하고 해당 데이터를 사용하여 실행 세션 중에 응용 프로그램 개체를 인식합니다.</p> <p>각 테스트 개체는 더 큰 테스트 개체 계층 구조의 일부입니다. 예를 들어 Link 개체는 (웹) Browser 개체 내부의 Page 개체 일부일 수 있습니다.</p>  <p>Browser 개체와 같은 최상위 개체는 Page 또는 Frame 개체와 같은 하위 수준 개체를 포함할 수 있으므로 컨테이너 개체라고도 합니다.</p>
<p>런타임 개체</p>	<p>런타임 개체는 UFT가 실행 세션 중에 실행(메서드)을 수행하는 응용 프로그램의 실제 개체입니다. UFT에서는 런타임 개체의 속성을 알아보고 테스트 개체로 변환합니다.</p>

UFT는 응용 프로그램에서 개체를 인식하면 해당하는 테스트 개체를 **개체 저장소**에 추가합니다. 이 개체 저장소는 테스트 개체의 참고 역할을 합니다. UFT에서는 테스트를 실행할 때 테스트 단계에 포함된 개체에 대한 테스트 개체 저장소를 확인합니다.

개체 저장소에 개체를 추가할 경우 UFT는 다음과 같이 작동합니다.

- 응용 프로그램에서 알아본 개체를 나타내는 UFT 테스트 개체 클래스를 식별하고 적당한 테스트 개체를 만듭니다.
- 응용 프로그램에서 개체 속성의 현재 값을 읽고 식별 속성 및 값 목록을 테스트 개체와 함께 저장합니다.
- 테스트 개체의 고유 이름을 선택합니다.

개체 저장소에는 다음 두 가지 유형이 있습니다.

<p>공유 개체 저장소</p>	<p>공유 개체 저장소는 개별 테스트와 독립적으로 존재하는 개체 저장소입니다. 공유 개체 저장소의 테스트 개체는 여러 테스트/수행에서 사용할 수 있습니다. 이렇게 하면 이 개체 저장소 유형이 테스트 개체 저장 및 유지 관리를 위한 기본 설정 저장소 유형으로 설정되므로 테스트 개체에 대한 업데이트가 해당 공유 개체 저장소를 사용하는 모든 테스트에 적용됩니다.</p>
<p>로컬 개체 저장소</p>	<p>로컬 개체 저장소에는 특정 수행의 컨텍스트에서 사용되는 테스트 개체가 있습니다. 이러한 개체 저장소 유형은 기타 수행에서는 사용할 수 없습니다. 기본적으로 모든 수행에는 로컬 개체 저장소가 있습니다.</p>

개체 저장소를 만들 때 테스트에 필요한 테스트 개체만 포함하는 것이 좋습니다. 이렇게 하면 개체 저장소가 비교적 작게 유지되고 유지 관리 및 개체 선택을 간소화할 수 있습니다. 또한 다른 사용자가 테스트를 만들거나 수정할 때 올바른 개체를 손쉽게 선택할 수 있도록 논리 이름을 제공해야 합니다.

개체 저장소는 검사점 및 출력 개체를 포함할 수도 있습니다. 검사점 개체 유형은 "[5단원: 단계 및 개체 매개 변수화](#)"(75페이지)에서 설명합니다.

연습 2a: 응용 프로그램의 개체 추가

"[UFT 테스트 개체 인식 - 개요](#)"(33페이지)에서는 UFT의 테스트 개체 모델 및 UFT가 개체를 알아보고 저장하는 방법을 알아보았습니다.

이 연습에서는 UFT의 개체 인식 기능을 사용하여 개체를 인식하고 개체 저장소를 만듭니다.

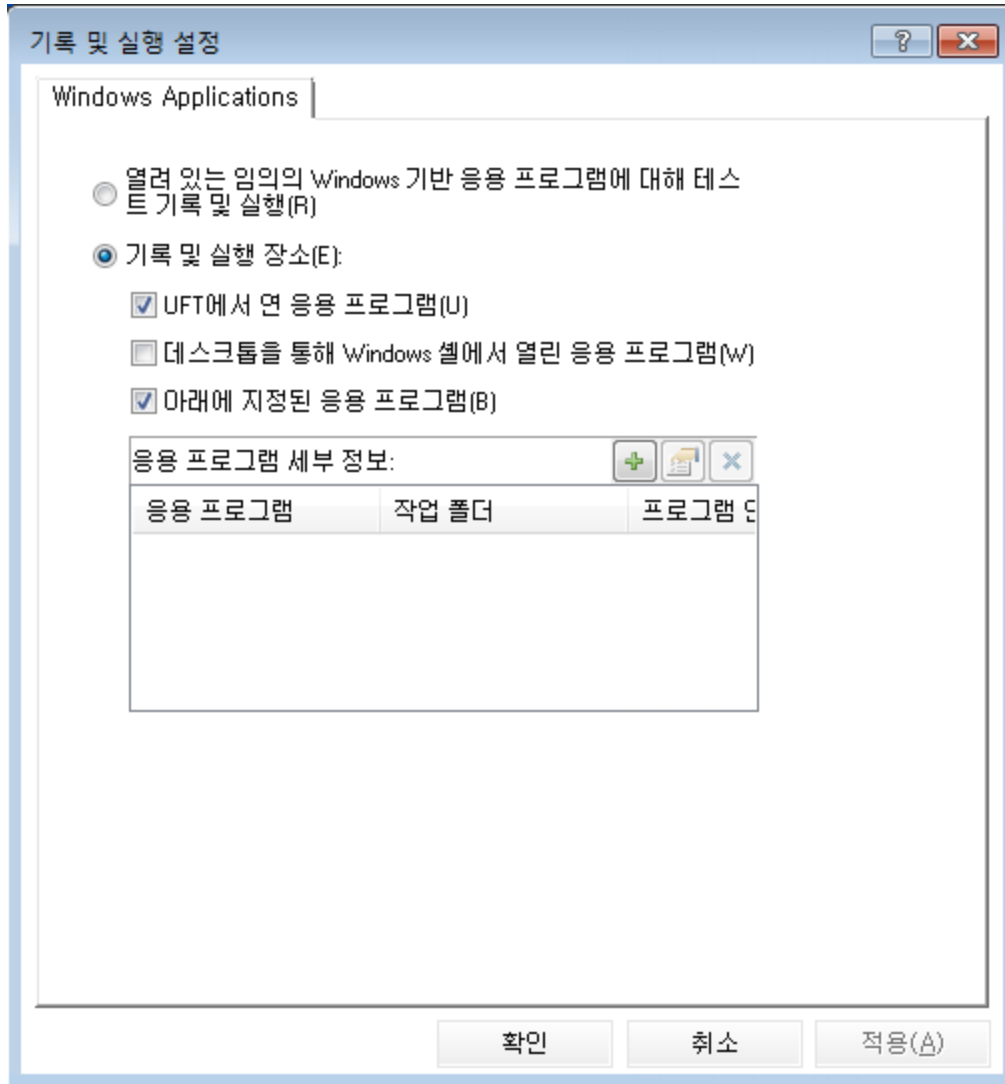
1. UFT를 시작하고 Book Flights 테스트를 엽니다.


- a. UFT가 현재 열려 있지 않으면 "[모든 테스트에 대한 솔루션 만들기](#)"(25페이지)에 설명된 대로 UFT를 엽니다.
- b. 시작 페이지의 **최근 테스트/구성 요소** 영역에서 **Book Flights**를 클릭합니다.
Book Flights 테스트가 열리고 "[1단원: GUI 테스트 및 수행 만들기](#)"(28페이지)에서 만든 Book Flights 테스트 및 해당 수행이 표시됩니다.

2. UFT에 대한 알아보기 설정을 지정합니다.

- a. **기록 > 기록 및 실행 설정**을 선택합니다. 기록 및 실행 설정 대화 상자가 열립니다.
- b. 기록 및 실행 설정 대화 상자에서 **Windows 응용 프로그램** 탭을 선택합니다.
- c. Windows 응용 프로그램 탭에서 **기록 및 실행 장소** 옵션을 선택합니다.

- d. 기록 및 실행 장소 옵션에서 **UFT에서 연 응용 프로그램 및 아래 지정된 응용 프로그램** 옵션의 확인란을 선택합니다.




- e. 응용 프로그램 세부 정보 영역에서 추가 버튼  을 클릭합니다.
- f. 응용 프로그램 세부 정보 대화 상자에서 응용 프로그램 경로 및 응용 프로그램 작업 폴더를 입력합니다.
- **응용 프로그램:** <UFT 설치 폴더>\samples\Flights Application\FlightsGUI.exe
 - **작업 폴더:** <UFT 설치 폴더>\samples\Flights Application
- g. **응용 프로그램 시작** 옵션을 선택하고 **확인**을 클릭합니다.
- h. 기본 기록 및 실행 설정 대화 상자에서 **확인**을 클릭합니다. 나중에 응용 프로그램에 대한 단계를 기록하거나 응용 프로그램에 대한 테스트를 실행할 때 UFT가 응용 프로그램을 사용할 수 있습니다.

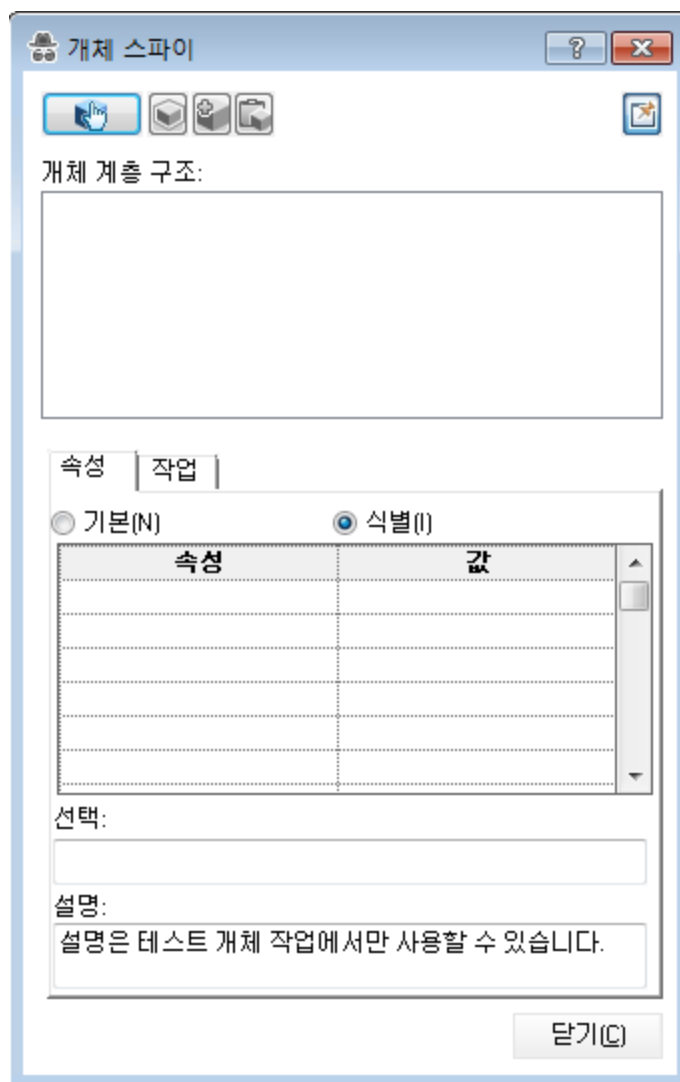
3. **항공편 예약 응용 프로그램을 엽니다.**


아직 열려 있지 않으면 항공편 예약 응용 프로그램을 엽니다.

참고: UFT를 열기 전이나 후에 항공편 예약 응용 프로그램을 열 수 있습니다.

4. 항공편 예약 응용 프로그램의 일부 개체에 대해 사용 가능한 속성과 작업을 봅니다.

- a. 도구 모음에서 개체 스파이 버튼  을 클릭합니다. 개체 스파이 대화 상자가 열립니다.



- b. 개체 스파이 대화 상자를 응용 프로그램 쪽으로 끕니다. 그러면 스파이 작업을 수행할 응용 프로그램의 개체를 더욱 분명하게 확인할 수 있습니다.
- c. 스파이 작업 시 개체 스파이를 최상위에 두기 전환 버튼  을 눌렀는지 확인합니다.

- d. 손모양 아이콘 버튼  을 클릭합니다.


손모양 아이콘을 누르면 UFT가 숨겨지고 개체 스파이 대화 상자가 항공편 예약 응용 프로그램 위에 표시됩니다.

팁: 항공편 예약 응용 프로그램, UFT 또는 현재 열린 창 사이에서 앞뒤로 전환해야 하면 **Ctrl** 키를 눌러 손모양 아이콘을 일반적인 Windows 포인터로 변경합니다. 포인터가 필요한 만큼 **CTRL** 버튼을 길게 누른 다음 손모양 아이콘을 사용할 준비가 되면 버튼을 놓습니다.

- e. 페이지에 있는 다양한 개체를 마우스로 가리키고 개체 스파이 대화 상자에서 발생하는 내용을 지켜봅니다.

참고: UFT가 올바른 위치의 개체를 인식하지 않으면 응용 프로그램이나 페이지를 100%로 보고 있고 현재 보기를 확대 또는 축소하지 않았는지 확인합니다.

예를 들어 페이지를 90% 또는 120%로 보고 있으면 개체를 인식하기 위해 실제 개체의 왼쪽 또는 오른쪽 영역을 클릭하거나 선택해야 할 수 있습니다.

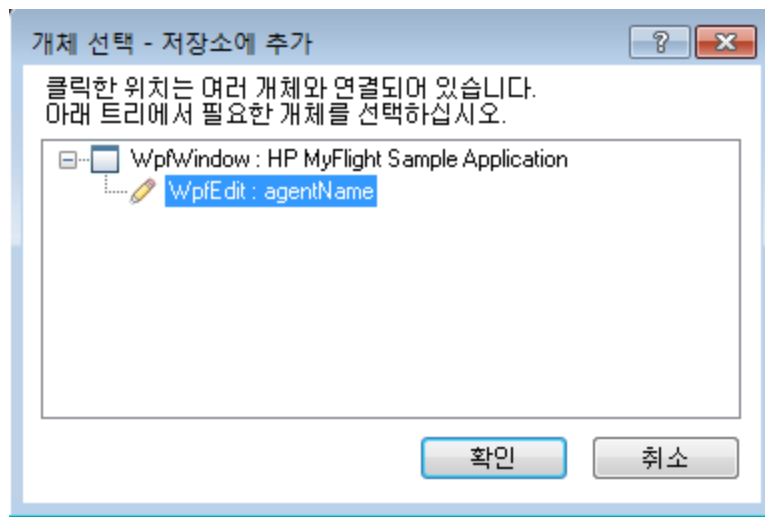
- f. **사용자 이름** 편집 상자를 클릭합니다. 해당 개체의 테스트 개체 계층 구조가 표시됩니다. 계층 구조 상자에서 개체 이름은 개체 클래스 유형인 WebEdit입니다.
- g. 응용 프로그램 **Login** 창에서 **Username** 편집 상자 내부를 클릭합니다. 그러면 응용 프로그램에서 개체가 활성화됩니다.
- h. UFT의 개체 스파이 대화 상자에서 손모양 아이콘 버튼  을 다시 클릭합니다. 개체 계층 구조 상자에서 개체 스파이에 **agentName**이 표시되는지 확인합니다.
- i. 개체 스파이 대화 상자를 닫습니다.

5. 테스트에 필요한 개체를 개체 저장소에 추가합니다.

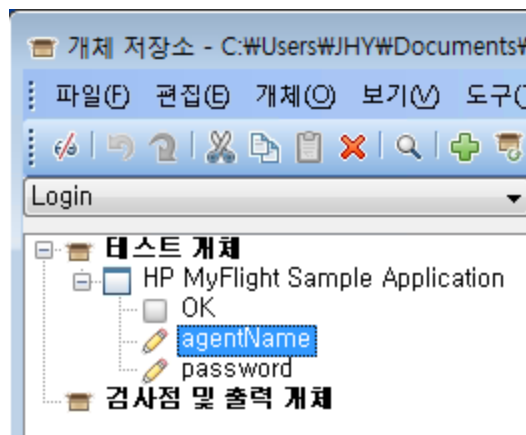
이 단계에서는 "스파이" 프로세스 단계를 추가로 수행하고 테스트에 필요한 개체만 알아보고 개체 저장소에 추가하도록 UFT에 지시합니다.

- a. 응용 프로그램 **Login** 창에서 **Username** 편집 상자 내부를 클릭합니다.
- b. UFT에서 **리소스 > 개체 저장소 관리자**를 선택합니다. 개체 저장소 관리자 창이 열립니다.
- c. 개체 저장소 관리자 창에서 **개체 > 개체 추가**를 선택합니다. UFT 및 개체 저장소 관리자가 둘 다 숨겨집니다.

- d. **Login** 페이지에서 **Username** 편집 상자를 다시 클릭합니다. 개체 선택 - 저장소에 추가 대화 상자가 열립니다.

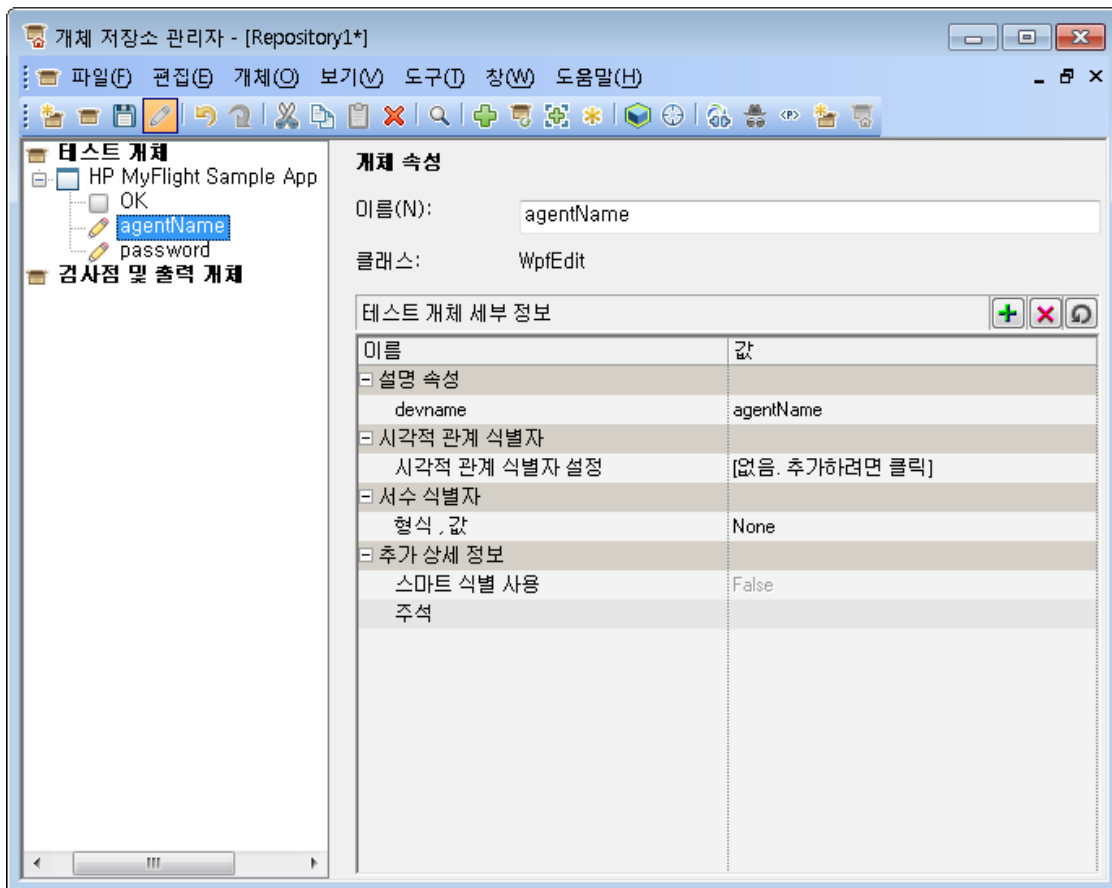


- e. 개체 선택 대화 상자에서 **agentName** 개체를 선택하고 **확인**을 클릭합니다. **agentName** 개체가 상위 개체인 **Login** 창 개체와 함께 개체 저장소에 추가됩니다.
- f. 위 프로세스를 반복하여 **Password** 편집 상자 및 **OK** 버튼에 대한 개체를 추가합니다. 모든 개체를 개체 저장소에 추가하고 나면 저장소가 다음과 같이 표시됩니다.




- 6. **UFT**에서 개체 중 하나에 대해 인식한 내용을 확인합니다.

테스트 개체 트리에서 **agentName** 개체를 선택하면 개체 속성이 개체 저장소의 오른쪽 창에 표시됩니다. 이 설명 속성은 UFT에서 실행 세션 중에 개체를 식별하는 데 사용됩니다.



7. 개체 저장소를 저장합니다.

- a. 개체 저장소 관리자 창에서 저장  을 클릭합니다.
- b. 솔루션과 테스트가 저장된 폴더인 **C:\%HOMEPATH%\My Documents\Unified Functional Testing**으로 이동합니다. 해당 폴더에서 **Tutorial_Object Repositories**라는 새 폴더를 만들고 엽니다.
- c. Tutorial_Object Repositories 폴더에서 개체 저장소를 **Login**이라는 이름으로 저장하고 확인을 클릭합니다.

참고: 이 단계에서는 개체 저장소만 저장합니다. 테스트에서 UFT 기본 창을 연 경우에도 개체 저장소를 저장할 때 개체 저장소가 테스트와 연결되지 않습니다.

8. 개체 저장소를 로그인 수행과 연결합니다.

개체 저장소를 수행과 연결하면 해당 저장소의 개체를 연결된 수행의 모든 단계에서 사용할 수 있습니다.

참고: 같은 개체 저장소를 여러 테스트 및 수행과 연결할 수 있습니다.

- a. 기본 UFT 창을 엽니다.
 - b. 기본 UFT 창에서 솔루션 탐색기를 엽니다.
 - c. 솔루션 탐색기의 **Book Flights** 노드에서 **Login** 노드를 확장합니다.
 - d. **로그인** 수행을 마우스 오른쪽 버튼으로 클릭하고 **수행에 저장소 연결**을 선택합니다. 공유 개체 저장소 열기 대화 상자가 열립니다.
 - e. 공유 개체 저장소 열기 대화 상자에서 이전 단계에서 개체 저장소를 저장한 폴더로 이동하고 **Login.tsr** 파일을 선택합니다. **열기**를 클릭합니다.
 - f. 열린 대화 상자에서 **예**를 클릭합니다.
- 이제 개체 저장소가 **Login** 수행에 연결되고 솔루션 탐색기에서 수행의 하위 요소로 표시됩니다.

9. 테스트를 저장합니다.

파일 > 저장을 선택하여 테스트를 저장합니다.

응용 프로그램 개체를 인식하도록 UFT에 지시하고, 해당 개체가 포함된 개체 저장소를 만들고, 개체 저장소를 테스트와 연결했으므로 응용 프로그램의 기타 영역에 대한 추가 개체 저장소를 만들 수 있습니다. 자세한 내용은 "[연습 2b: 탐색 및 알아보기를 사용하여 개체 저장소 만들기](#)"(40페이지)에서 계속됩니다.

연습 2b: 탐색 및 알아보기를 사용하여 개체 저장소 만들기

"[연습 2a: 응용 프로그램의 개체 추가](#)"(34페이지)에서는 UFT를 사용하여 응용 프로그램에 개체를 추가하고 항공편 예약 응용 프로그램의 Login 페이지에 대한 공유 개체 저장소를 만드는 방법을 알아보고 이 개체 저장소를 테스트와 연결했습니다.

이 단원에서는 사이트의 나머지 페이지 각각에 대한 공유 개체 저장소를 만듭니다. 한 번에 응용 프로그램의 한 페이지 또는 섹션에 있는 모든 개체를 인식할 수 있는 탐색 및 알아보기 프로세스를 사용합니다.


팁: 항상 응용 프로그램 또는 웹 사이트의 각 섹션/페이지에 대한 별도의 공유 개체 저장소를 만드는 것이 좋습니다. 이렇게 하면 테스트 단계를 추가하거나 수정할 때 또는 유지 관리 작업을 수행할 때 올바른 개체를 더욱 손쉽게 찾을 수 있습니다.

이 연습에서는 각 페이지에서 모든 개체를 알아보도록 UFT에 지시합니다. 이때 개체 저장소를 특정 수행과 연결할 필요가 없습니다.

1. **항공편 예약 응용 프로그램의 Flight Finder 페이지에 로그인합니다.**


- a. 아직 열려 있지 않으면 항공편 예약 응용 프로그램을 엽니다.
- b. Login 페이지에서 로그인 자격 증명을 입력합니다.
 - o 사용자 이름: john
 - o 비밀번호: hp
- c. 확인을 클릭합니다. Book Flight 페이지가 열립니다.

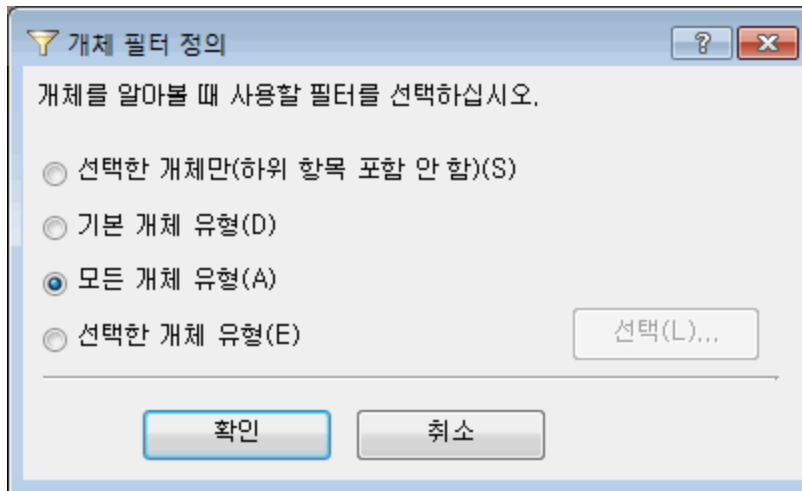
2. **새 공유 개체 저장소를 만듭니다.**

- a. 개체 저장소 관리자가 달혀 있으면 UFT에서 리소스 > 개체 저장소 관리자를 선택하여 엽니다.
- b. 개체 저장소 관리자 창에서 새로 만들기 를 클릭합니다. 빈 공유 개체 저장소가 열립니다.

3. **개체 필터 정의를 설정하여 탐색 및 알아보기 프로세스를 시작합니다.**

이전 연습에서 추가한 것처럼 개체를 개별적으로 추가할 뿐 아니라 탐색 및 알아보기 메커니즘을 사용하여 하나의 프로세스에서 응용 프로그램의 모든 개체를 인식하고 추가할 수 있습니다.

- a. 개체 저장소 관리자 창에서 개체 > 탐색 및 알아보기를 선택합니다. UFT 및 개체 저장소 관리자가 둘 다 숨겨집니다.
- b. 탐색 및 알아보기 도구 모음에서 개체 필터 정의 버튼 을 클릭합니다. 개체 필터 정의 대화 상자가 열립니다.
- c. 개체 필터 정의 대화 상자에서 모든 개체 유형을 선택하고 확인을 클릭합니다.



4. **Flight Finder 페이지에서 모든 개체를 확인합니다.**

이 단계에서는 Book Flight 페이지에서 필터와 일치하는 모든 개체를 인식하고 공유 개체 저장소에 추가하도록 UFT에 지시합니다.


- a. 항공편 예약 응용 프로그램의 Book Flight 페이지에서 응용 프로그램 제목 표시줄을 클릭하여 UFT가 알아보도록 할 페이지에 포커스를 설정합니다.

- b. 탐색 및 알아보기 도구 모음에서 **확인**을 클릭합니다. 응용 프로그램이 깜박이고 UFT에서 페이지의 개체 표현을 새 개체 저장소에 추가하기 시작하면 개체 추가 메시지 상자가 표시됩니다.

참고: 이러한 개체 추가에는 몇 초가 걸립니다. 탐색 및 알아보기 프로세스가 실행되는 동안 응용 프로그램과 상호 작용하지 마십시오.

- c. 탐색 및 알아보기 도구 모음을 닫습니다. UFT 및 개체 저장소 관리자 창이 다시 표시됩니다.

5. 공유 개체 저장소를 저장합니다.

- a. 개체 저장소 관리자 창에서 저장  을 클릭합니다. 공유 개체 저장소 저장 대화 상자가 열립니다.
- b. 이전 연습에서 만든 **Tutorial_Object Repositories** 폴더로 이동합니다.
- c. 이 개체 저장소의 이름을 Flight Finder로 지정하고 **확인**을 클릭합니다.

6. 나머지 응용 프로그램 페이지에 대한 개체 저장소를 만듭니다.

- a. 이전 단계에서 설명한 프로세스를 사용하여 다음 페이지 각각에 대한 새 공유 개체 저장소를 만듭니다.
 - **Select Flight**
 - **Flight Details**

중요: Flight Details 페이지에서 개체를 인식하기 전에 **Passenger Name** 상자에 문자열을 입력해야 합니다. 그러면 **Order** 버튼이 활성화되고 UFT가 버튼을 제대로 알아볼 수 있습니다. 이 버튼은 다른 연습에서 필요합니다.

- b. 이름을 각각 개체 저장소 Select Flight 및 Flight Confirmation으로 지정합니다.

7. Flight Finder 개체 저장소를 Flight Finder 수행과 연결합니다.

- a. UFT 창으로 전환합니다. 솔루션 탐색기가 열려 있지 않으면 UFT 창의 왼쪽 아래에서 **솔루션 탐색기** 탭을 클릭하여 엽니다.
- b. 솔루션 탐색기의 Book Flights 노드에서 **Flight Finder** 노드를 마우스 오른쪽 버튼으로 클릭하고 **수행에 저장소 연결**을 선택합니다. 공유 개체 저장소 열기 대화 상자가 열립니다.
- c. **Tutorial_ObjectRepositories** 폴더로 이동합니다.
- d. Tutorial_ObjectRepositories 폴더에서 **Flight Finder.tsr** 파일을 선택하고 **열기**를 클릭합니다.
- e. 열린 대화 상자에서 **예**를 클릭합니다.

이제 개체 저장소가 **Flight Finder** 수행에 연결되고 솔루션 탐색기에서 수행의 하위로 표시됩니다.


8. 나머지 개체 저장소를 관련 수행과 연결합니다.

개체 저장소를 다음과 같이 수행과 연결합니다.

수행	개체 저장소
Select Flight	Select Flight.tsr
Flight Confirmation	Flight Confirmation.tsr

나중에 각 수행에 단계를 추가할 때 모든 필요한 테스트 개체를 사용할 수 있습니다.

9. 테스트를 저장합니다.

저장  을 클릭합니다.

개체 저장소를 만들고 테스트와 연결했으므로 이제 이러한 개체를 사용하여 테스트 단계를 만들 수 있습니다. ["3단원: 테스트에 단계 추가"\(44페이지\)](#)로 이동하여 테스트 단계 만들기를 시작하십시오.

3단원: 테스트에 단계 추가

"1단원: GUI 테스트 및 수행 만들기"에서 항공편 예약 응용 프로그램 테스트에 대한 구조를 제공하는 테스트 및 수행을 만들었습니다. "2단원: 개체 저장소 만들기"에서는 응용 프로그램에 대한 테스트 개체가 있는 개체 저장소를 만들었습니다.

이 단원에서는 응용 프로그램 테스트를 만들기 위한 최종 필수 단계에 대해 알아봅니다. 사용자 인터페이스에서 사용자 수행의 정확한 테스트를 실행할 수 있도록 하는 테스트 단계를 GUI 테스트에 추가하는 방법을 알아봅니다.

이 단원의 내용은 다음과 같습니다.

- GUI 테스트에서 테스트 단계 추가 - 개요 45
- 연습 3a: 키워드 보기에서 로그인 수행에 단계 추가 45
- 연습 3b: 기록을 통해 FlightFinder 수행에 단계 추가 52
- 연습 3c: 도구 상자 창을 사용하여 Select Flight 수행에 단계 추가 55
- 연습 3d: 단계 생성기를 사용하여 Book Flight 수행에 단계 추가 58
- 고급 연습 3e(선택 사항) - 편집기를 사용하여 단계 추가 63

GUI 테스트에서 테스트 단계 추가 - 개요

GUI 테스트에서 테스트 단계를 만들려면 테스트 단계에서 개체를 사용하고 테스트 개체에서 수행할 수행을 UFT에 지시해야 합니다. 이렇게 하면 UFT가 응용 프로그램에서 테스트 개체 메서드(수행)를 런타임 개체에 대한 수행으로 변환하여 응용 프로그램에 대한 수행을 재생할 수 있습니다.

이 작업을 지원하기 위해 UFT에서는 테스트 단계를 추가하는 다음과 같은 다양한 방법을 제공합니다.

키워드 보기	키워드 보기를 사용하여 단계 표에서 테스트 개체를 선택하고 이러한 테스트 개체에 필요한 수행(메서드)을 추가할 수 있습니다. 필요한 경우 키워드 보기에서 개체 계층 구조를 자동으로 정렬합니다. 적절한 개체 및 메서드를 선택하면 개체 이름, 개체 메서드, 추가된 매개 변수, 단계의 설명서 요약물 보여 주는 표에 테스트 단계가 표시됩니다.
편집기	편집기에서 필요에 따라 개체에 필요한 계층 구조를 포함한 개체와 함께 개체 메서드 및 매개 변수를 입력할 수 있습니다. 응용 프로그램 코드를 작성한 적이 있으면 이 방법으로 테스트 단계를 더 쉽게 만들 수 있습니다.
기록	UFT에는 응용 프로그램 개체에서 수행할 수행 및 응용 프로그램의 테스트 개체를 자동으로 인식하는 기본 제공 기록 메커니즘이 있습니다. 응용 프로그램에 기록하면 UFT에서는 수행을 테스트 단계로 변환하여 개체 이름과 개체에서 수행되는 수행(메서드)을 표시합니다. 이를 사용하여 사용자처럼 테스트를 수행하고 차례로 UFT 내부에서 테스트를 수동으로 편집하는 대신 UFT가 테스트를 자동으로 만들도록 지시할 수 있습니다.
도구 상자 창	문서 창에서 GUI 수행 탭을 선택하면 UFT에서는 도구 상자 창에서 해당 수행에 대한 연결된 개체 및 함수를 자동으로 표시합니다. 그리고 나서 이러한 테스트 개체(또는 함수)를 키워드 보기 또는 편집기로 끌 수 있고 UFT에서는 해당 개체가 있는 단계를 자동으로 만듭니다. (그러나 도구 상자 창에서 테스트 개체를 끈 후에 개체에 대한 메서드를 제공해야 합니다.)
단계 생성기	단계 생성기 대화 상자를 사용하면 단일 대화 상자에서 테스트 단계에 대한 모든 세부 정보를 선택하고 제공할 수 있습니다. 단계 생성기에서는 현재 선택된 수행과 연결된 테스트 개체, 해당 수행의 메서드 및 필요한 매개 변수를 선택할 수 있습니다. 이 정보를 선택하면 UFT가 선택한 위치에 모든 세부 정보가 포함된 단계를 삽입합니다.

다음 연습에서는 이러한 각 메서드를 사용하여 테스트 단계를 만듭니다.


연습 3a: 키워드 보기에서 로그인 수행에 단계 추가

이 연습에서는 키워드 보기를 사용하여 로그인 수행에 단계를 삽입합니다.

1. UFT를 시작하고 Book Flights 테스트를 엽니다.

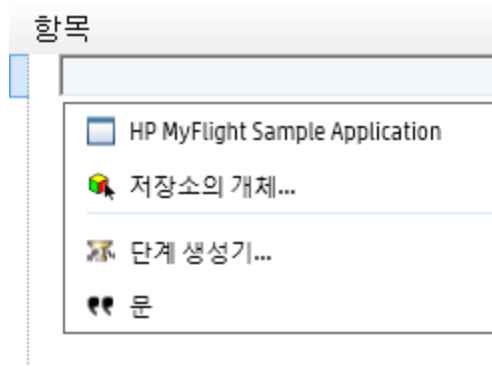
- a. UFT가 현재 열려 있지 않으면 "[모든 테스트에 대한 솔루션 만들기](#)"(25페이지)에 설명된 대로 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- b. 시작 페이지의 최근에 사용한 솔루션 영역에서 **Flight Reservation Application**을 클릭합니다.
["1단원: GUI 테스트 및 수행 만들기"](#)(28페이지)에서 만든 Book Flights 테스트가 표시된 Flight Reservation Application 솔루션이 열립니다.

2. 로그인 수행을 열고 키워드 보기를 표시합니다.

- a. 테스트 흐름 캔버스가 현재 열려 있지 않으면 문서 창에서 **Book Flights** 탭을 클릭하여 캔버스를 표시합니다.
- b. 캔버스에서 **로그인** 수행을 두 번 클릭합니다.
로그인 수행이 문서 창에서 개별 탭으로 열립니다.
- c. 편집기가 표시되면 도구 모음에서 **키워드 보기** 버튼  을 클릭하여 키워드 보기를 표시합니다.

3. 항공편 예약 응용 프로그램에 로그인하는 첫 번째 단계를 추가합니다.

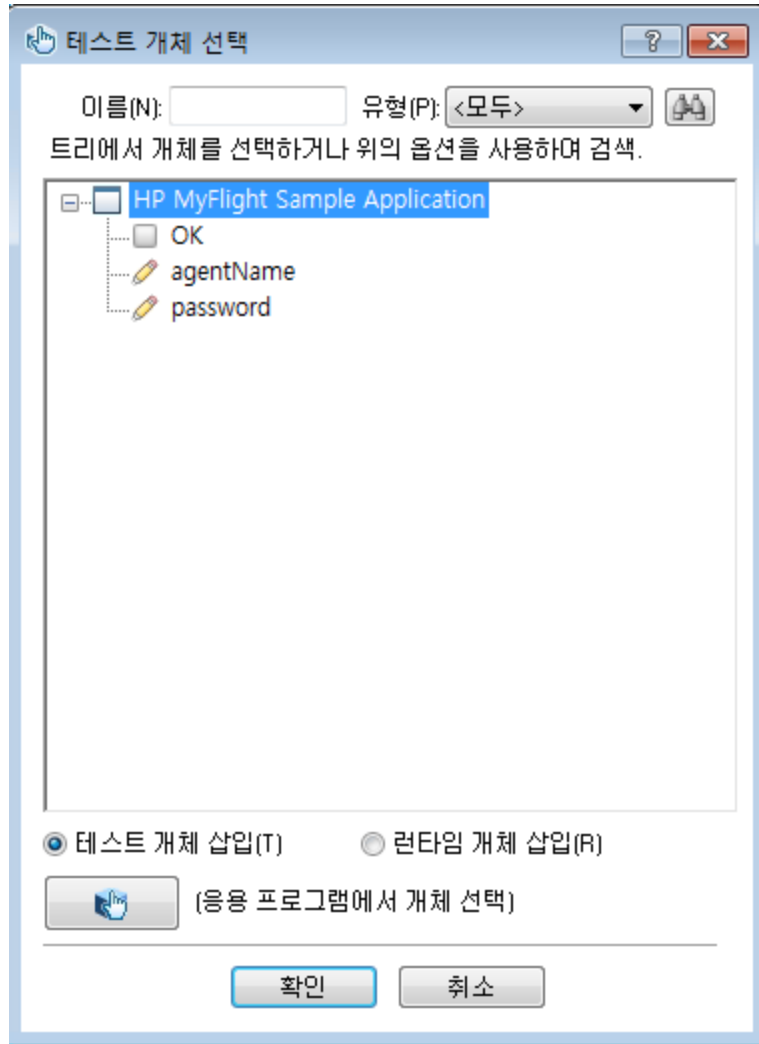
- a. 키워드 보기의 항목 열에서 **새 단계** 버튼을 클릭합니다. 항목 열의 표에서 빈 편집 필드가 열립니다.
- b. 항목 열에서 빈 편집 필드를 클릭합니다. 항목 목록 드롭다운 목록이 열리고 연결된 개체 저장소에 최상위 수준 상위 개체가 표시되고 항목 선택을 지시하는 메시지가 표시됩니다.



이 단계에서 상위 개체는 **Login** 창입니다. 기본 창에서 작업을 수행하지 않을 것이므로 이 개체를 선택하지 *마십시오*. 작업을 수행할 개체에 대한 단계만 삽입하면 됩니다.

- c. **저장소의 개체**를 선택하여 테스트 개체 선택 대화 상자를 엽니다.

- d. 테스트 개체 선택 대화 상자에서 테스트 개체 트리를 확장합니다.



- e. 테스트 개체 트리에서 **agentName**을 선택하고 **확인**을 클릭합니다.
테스트 개체 선택 대화 상자가 닫히고 수행에 한 단계가 추가됩니다.
키워드 보기에는 3개의 행이 추가됩니다. UFT에서는 이러한 개체에서 작업을 수행하지 않더라도 각 상위 테스트 개체에 대한 행을 추가합니다. 이러한 행은 단계가 수행되는 개체에 대한 경로의 일부입니다.
실행 세션 중에 UFT에서는 상위 개체를 사용하여 작업을 수행해야 하는 실제 개체를 식별합니다.

이 단계에서는 마지막 3개의 새로운 행이 제공됩니다.

- 선택된 **agentName** WebEdit 개체가 **항목** 셀에 추가됩니다.
- 기본 메서드 **Set**가 **작업** 셀에 추가됩니다.
- 이 단계에서 편집 상자의 텍스트가 지워짐을 나타내는 텍스트가 **설명** 셀에 추가됩니다. 이는 이 단계에서 **값** 셀에 필수 값이 없고 사용자 이름을 사용하여 업데이트해야 하기 때문입니다.

항목	작업	값	설명
HP MyFlight Sample Application			
agentName	Set		"agentName" edit box의 텍스트를 지웁니다.
+ 새 단계			

f. **값** 셀을 클릭하고 john을 입력합니다. 문자열을 입력하고 나서 **Enter** 키를 누릅니다.

이 값을 삽입하면 단계가 완료됩니다. 키워드 보기에서 다른 영역을 클릭하면 이 단계에 대한 설명이 **설명** 셀에서 업데이트됩니다.

항목	작업	값	설명
HP MyFlight Sample Application			
agentName	Set	john	john을(를) "agentName" edit box에 입력합니다.
+ 새 단계			


팁: 값 열에 입력한 값 주위에 따옴표가 자동으로 추가되어 이 값이 String 값을 나타냅니다. 메서드가 Index 값을 지원하고 Index 값을 입력했다면 따옴표가 추가되지 않습니다.

g. **보기 > 편집기**를 선택하여 VBScript로 단계 구문을 표시하는 편집기를 표시합니다.

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

이 단계는 agentName이라는 **WpfEdit**(편집 상자) 테스트 개체에서 수행되며 다음에 유의하십시오.

- WpfEdit(편집 상자) 테스트 개체에 대한 테스트 개체 계층 구조는 테스트 개체 앞에 표시됩니다. 이 단계에서 계층 구조에는 WpfWindow 개체가 포함됩니다.
- 개체에서 수행할 메서드가 테스트 개체 바로 뒤에 표시됩니다. 이 단계에서 메서드는 **Set**입니다.
- **agentName** 편집 상자에 입력할 텍스트는 **Set** 메서드 바로 뒤에 표시됩니다. 텍스트는 따옴표 안에 표시되어 이 값이 문자열임을 나타냅니다. 이 단계에서 입력할 텍스트는 john입니다.
- 전체 중지(마침표)로 단계의 각 부분을 구분합니다.

h. **키워드 보기** 버튼  을 클릭하여 키워드 보기로 돌아갑니다.

4. 다음 단계를 추가합니다.

a. **항목** 열의 **agentName** 행 바로 아래에서 **새 단계** 버튼을 클릭합니다. 편집 필드가 있는 다른 행이 추가됩니다.

- b. 빈 편집 필드를 클릭합니다. 항목 목록이 열리고 이전 단계 테스트 개체의 동일 수준 개체가 나열됩니다.
- c. 항목 목록에서 **password**를 선택합니다. 이때 개체가 이전 단계와 같은 상위 개체를 공유하므로 새 행이 하나만 추가됩니다.

이 단계에서:

- **password** WpfEdit 테스트 개체가 **항목** 셀에 추가됩니다.
- 기본 메서드 **Set**가 **작업** 셀에 추가됩니다. 비밀번호를 인코딩해야 하므로 이 메서드를 변경해야 합니다.
- 이 단계에서 편집 상자의 텍스트가 지워짐을 나타내는 텍스트가 **설명** 셀에 추가됩니다. 이는 이 단계에서 **값** 셀에 아직 필수 값이 없고 비밀번호를 사용하여 업데이트해야 하기 때문입니다.
- d. **작업** 셀을 클릭하여 아래쪽 화살표를 표시하고 아래쪽 화살표를 클릭하여 선택된 테스트 개체에 대해 사용 가능한 메서드 목록을 표시합니다.
- e. 메서드 목록에서 **SetSecure**를 선택합니다. 이 메서드를 통해 암호화된 텍스트를 사용할 수 있습니다.

5. HP 비밀번호 인코더 응용 프로그램을 사용하여 인코딩된 비밀번호를 생성합니다.

- a. 시작 > 모든 프로그램 > HP 소프트웨어 > HP > HP Unified Functional Testing > 도구 > 비밀번호 인코더 또는 <UFT 설치 폴더>\bin\CryptonApp.exe를 선택합니다.
- b. 비밀번호 인코더의 **비밀번호** 상자에 hp를 입력합니다.
- c. **생성**을 클릭합니다. 비밀번호 인코더에서 비밀번호를 암호화하고 **인코딩된 문자열** 상자에 표시합니다.
- d. **복사**를 클릭합니다.
- e. UFT의 로그인 수행에서 비밀번호 단계에 대한 **값** 셀에 인코딩된 값을 붙여넣고 **Enter** 키를 누릅니다.
이 단계의 **설명서** 열에 업데이트된 설명서가 표시됩니다.
- f. 비밀번호 인코더 대화 상자를 닫습니다.
이때 수행을 실행하면 UFT에서는 항공편 예약 응용 프로그램을 자동으로 열고 지정한 값을 **Username** 및 **Password** 상자에 삽입합니다.

6. 로그인 수행에 마지막 단계를 삽입합니다.

- a. 마지막 단계 아래의 **항목** 열에서 **새 단계**를 클릭하여 다음 단계를 삽입합니다. 빈 편집 필드가 열립니다.
- b. 편집 필드를 클릭합니다. 항목 목록이 열리고 이전 단계 테스트 개체의 동일 수준 개체가 나열됩니다.
- c. **항목** 목록에서 **확인**을 선택합니다.
이 단계에서는 응용 프로그램의 사용자 이름 및 비밀번호를 입력한 후 **확인**을 클릭하도록 UFT에 지시합니다.

7. 테스트를 저장합니다.

파일 > 저장을 선택합니다.

키워드 보기 및 편집기에 대해 자세히 알아보려면 "[키워드 보기 및 편집기에서 로그인 수행 분석](#)"(50페이지)을 계속 진행하십시오.

계속해서 테스트에 단계를 추가하려면 "[연습 3b: 기록을 통해 FlightFinder 수행에 단계 추가](#)"(52페이지)로 이동하십시오.

키워드 보기 및 편집기에서 로그인 수행 분석

일부 테스트 단계를 만들었으므로 키워드 보기 및 편집기에서 이러한 단계가 어떻게 표시되는지 살펴보겠습니다.


키워드 보기

단계를 추가하면 키워드 보기가 다음과 같이 표시됩니다.

항목	작업	값	설명
HP MyFlight Sample Application			
agentName	Set	"john"	"john"을(를) "agentName" edit box에 입력합니다.
password	Set	"53b3cd55b30d8acf6af1"	"53b3cd55b30d8acf6af1"을(를) "password" edit box에 입력합니다.
OK	Click		"OK" button을(를) 클릭합니다.
+ 새 단계			

보는 것처럼 테스트의 단계는 개체 계층 구조 및 테스트 단계 시퀀스를 보여 주는 아이콘 기반 표에 정렬됩니다. 키워드 보기의 각 줄은 서로 다른 정보 조각을 나타냅니다.

줄	설명
HP MyFlight Sample Application	HP MyFlight Sample Application 창 개체는 이 수행에 포함된 테스트 개체의 상위 개체입니다. 다음 단계에 포함된 모든 테스트 개체는 이 개체에 대한 하위 노드로 표시됩니다.
agentName Set "john"	agentName 은 UFT가 수행을 실행하는 편집 상자의 이름입니다. Set 메서드는 agentName 개체에서 실행되는 수행입니다. UFT에서는 편집 상자에서 john을 텍스트로 입력합니다.
password SetSecure	password 는 UFT가 수행을 실행하는 편집 상자의 이름입니다. SetSecure 메서드는 passwordWatermakr 개체에서 실행되는 수행입니다.

	UFT에서는 password 편집 상자에서 숫자 문자 열을 텍스트로 입력합니다.
	<p>OK는 agentName 및 password 편집 상자에 정보가 입력된 후 UFT가 클릭하는 버튼의 이름입니다.</p> <p>Click 메시드는 버튼에서 실행되는 수행입니다.</p>

키워드 보기의 각 단계에는 다음과 같은 다양한 요소가 있습니다.

키워드 보기 요소	설명
항목	단계에 대한 항목(테스트 개체, 유틸리티 개체, 함수 호출 또는 문)입니다. 이 항목은 계층 구조식 아이콘 기반 트리에 표시됩니다.
작업	항목에서 수행할 작업입니다(예: Click , Set 또는 Select).
값	선택된 작업의 인수 값입니다(필요한 경우). 예를 들어 편집 상자에 입력할 텍스트 또는 이미지를 클릭할 때 사용할 마우스 버튼입니다.
설명서	단계가 수행하는 작업에 대한 문을 이해하기 쉬운 문장으로 자동으로 제공합니다. 예: Click the "OK" button.
할당	나중에 테스트에서 사용할 수 있도록 값을 변수에 할당하거나 변수의 값을 할당합니다. 이 열은 기본적으로 표시되지 않습니다.
주석	단계에 관해 추가할 텍스트 정보입니다. 예를 들어 Return to page used in first step of test 주석을 추가할 수 있습니다. 이 열은 기본적으로 표시되지 않습니다.

팁: 키워드 보기에서 열 머리글을 마우스 오른쪽 버튼으로 클릭하고 목록에서 열 이름을 선택하여 개별 열을 숨기거나 표시할 수 있습니다.

키워드 보기에 대한 자세한 내용은 *HP Unified Functional Testing User Guide*에서 키워드 보기 섹션을 참조하십시오.

편집기

단계를 추가하면 편집기가 다음과 같이 표시됩니다.



```

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

키워드 보기와 달리 편집기의 각 단계는 다음 형식의 스크립트 줄로 표현됩니다.

<object hierarchy>.<method> <method parameters>

따라서 각 단계 및 스크립트의 각 줄에 해당하는 다양한 요소가 표시됩니다.

<p>테스트 개체 계층 구조</p>	<p>다음에 포함하여 편집기의 각 단계에 대한 전체 개체 계층 구조를 가져옵니다.</p> <ul style="list-style-type: none"> • 테스트 개체 유형 • 각 개체의 개체 이름(UFT에서 식별된 대로 표시) <p>위에 표시된 예의 첫 번째 줄에서 다음 요소를 둘 다 확인할 수 있습니다.</p> <ul style="list-style-type: none"> • WpfWindow는 테스트 개체 유형입니다. • "Login"은 개체 이름입니다(UFT에서 식별된 대로 표시).
<p>개체 메서드</p>	<p>개체 계층 구조 다음에 메서드(개체에서 실행할 수행)가 표시됩니다. 각 메서드는 굵게 표시된 텍스트로 표시됩니다.</p> <p>위에 표시된 예의 첫 번째 줄에서 개체는 Set 메서드를 수행합니다.</p>
<p>메서드 매개 변수</p>	<p>대부분 메서드에는 필수 또는 선택적 매개 변수를 제공해야 합니다. 이러한 매개 변수는 편집기에서 메서드 이름 뒤에 나열됩니다.</p> <p>위에 표시된 예의 첫 번째 줄에서 Set 메서드는 John 매개 변수를 입력합니다.</p>

편집기에 대한 자세한 내용은 UFT에서 코드 및 텍스트 문서 편집 섹션을 참조하십시오.

연습 3b: 기록을 통해 FlightFinder 수행에 단계 추가

"연습 3a: 키워드 보기에서 로그인 수행에 단계 추가"에서는 키워드 보기를 사용하여 단계를 만드는 방식으로 항공편 예약 응용 프로그램의 Login 페이지에서 실행할 단계를 테스트에 추가했습니다.

이 연습에서는 응용 프로그램의 Flight Finder 페이지용으로 만든 **Flight Finder** 수행에 대한 단계를 기록합니다. 이 수행에서는 Flight Finder 공유 개체 저장소에 포함된 테스트 개체를 사용합니다.

팁: 기록 세션을 시작하기 전에 화면에서 응용 프로그램 창과 이 자습서 창을 나란히 배치할 수 있습니다. 이렇게 하면 기록 중에 자습서를 읽을 수 있습니다.

1. UFT를 시작하고 Book Flights 테스트를 엽니다.

- a. UFT가 현재 열려 있지 않으면 **"모든 테스트에 대한 솔루션 만들기"**(25페이지)에 설명된 대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- b. **파일 > 열기 > 솔루션**을 선택합니다. 솔루션 열기 대화 상자가 열립니다.
- c. 솔루션 열기 대화 상자에서 **C:\%HOMEPATH%\My Documents\Unified Functional Testing**에 있는 **Flight Reservation Application.ftsln** 파일로 이동하고 **열기**를 클릭합니다.
"1단원: GUI 테스트 및 수행 만들기"(28페이지)에서 만든 Book Flights 테스트가 표시된 Flight Reservation Application 솔루션이 열립니다.

2. Flight Finder 수행을 엽니다.

- a. 테스트 흐름 캔버스가 현재 열려 있지 않으면 문서 창에서 **Book Flights** 탭을 클릭하여 캔버스를 표시합니다.

- b. 캔버스에서 **Flight Finder** 수행을 두 번 클릭합니다.
Flight Finder 수행이 문서 창에서 개별 탭으로 열립니다.

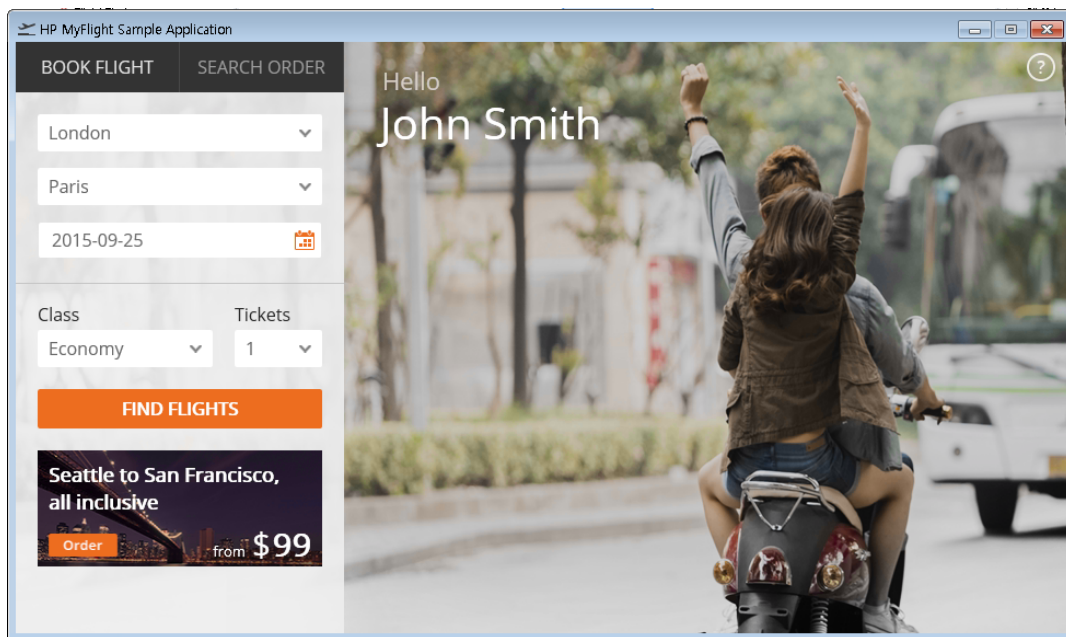
3. **항공편 예약 응용 프로그램을 엽니다.**

- a. 아직 열려 있지 않으면 "**Flight Reservation Application 탐색**"(23페이지)의 설명대로 항공편 예약 응용 프로그램을 엽니다.

항공편 예약 응용 프로그램의 **Login** 페이지가 열립니다.

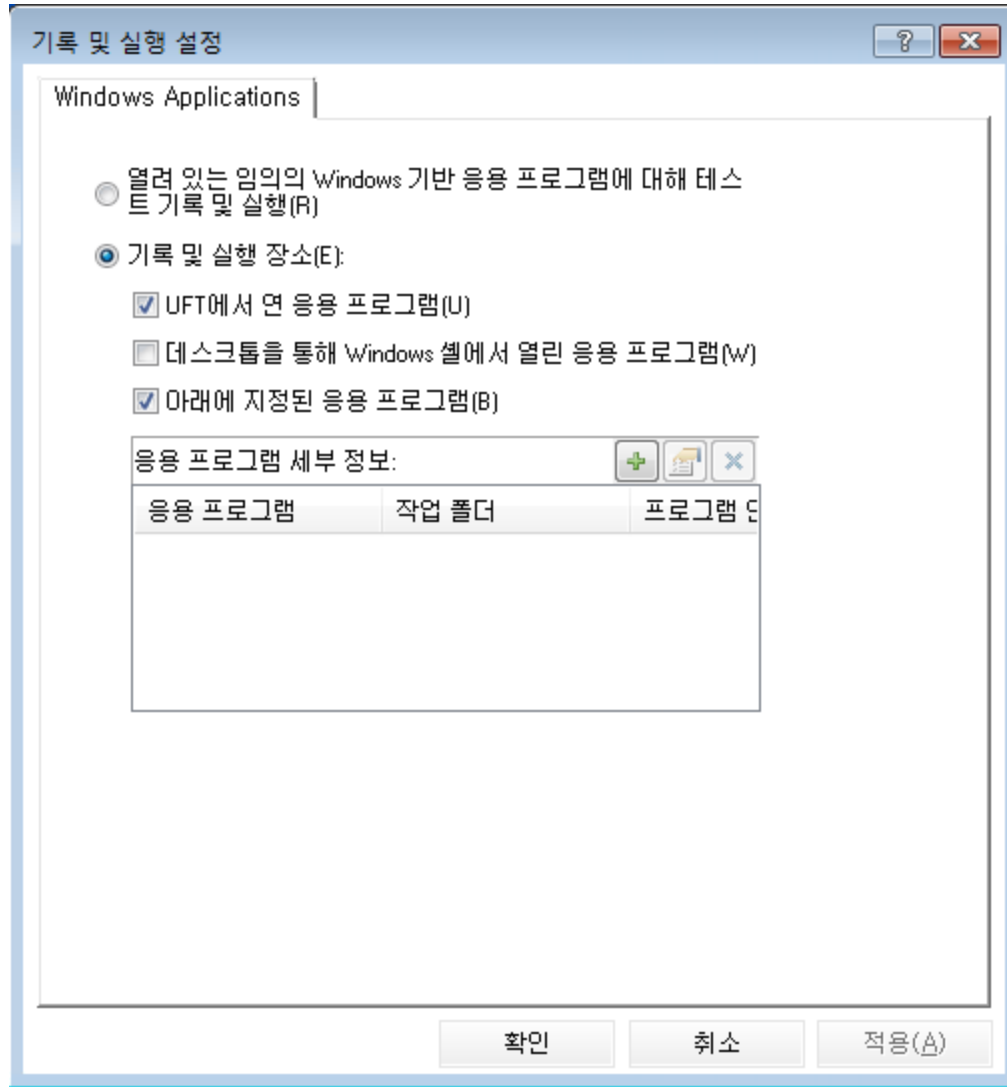
- b. 응용 프로그램에 로그인합니다. 사용자 이름으로 john을 사용하고 비밀번호로 hp를 사용합니다.


응용 프로그램의 Flight Finder 페이지가 열립니다.



- c. UFT에서 기록 > 기록 및 실행 설정을 선택합니다. 기록 및 실행 설정 대화 상자가 열립니다.

- d. 기록 및 실행 설정 대화 상자의 **Windows 응용 프로그램** 탭에서 **열려 있는 임의의 Windows 응용 프로그램에서 테스트 기록 및 실행**을 선택합니다. 그러면 UFT가 열린 응용 프로그램에서 실행될 수 있습니다.



- e. UFT에서 기록 버튼  을 클릭합니다. 기록 도구 모음이 화면 위쪽에 나타나고 UFT가 기록을 시작합니다.

UFT 창이 사라지고 응용 프로그램의 Flight Finder 페이지가 표시됩니다.

- f. Flight Finder 페이지에서 다음 항목 중 선택하여 다음 선택 사항을 변경합니다.
- **출발지:** Los Angeles
 - **도착지:** Sydney
 - **날짜:** 내일 날짜

중요: 편집 상자에 날짜를 입력합니다. **M/D/YYYY** 형식을 사용하십시오.

- **클래스:** Business
- **티켓:** 2

참고: 월말 또는 연말에 이 자습서를 수행하고 있는 경우 기록하는 동안 다른 월 또는 연도를 선택하십시오. UFT는 응용 프로그램에서 내용을 변경할 때만 작업을 기록하므로 이렇게 하면 단계가 기록됩니다. 기본값을 수락하거나 기본값을 다시 선택하면 UFT가 작업을 기록하지 않습니다.

- 선택을 완료하면 **FIND FLIGHTS**를 클릭하여 계속합니다. Select Flight 페이지가 열립니다.
- 기록 도구 모음에서 **중지** 버튼을 클릭하여 기록을 중지합니다.

이제 로스앤젤레스에서 시드니로 가는 가상 티켓을 예약했습니다. UFT에서 기록 버튼을 클릭한 때부터 기록 도구 모음에서 중지를 클릭한 때까지 UFT에서 응용 프로그램의 수행을 기록했습니다.

4. 테스트를 저장합니다.

저장  을 클릭합니다.

계속해서 다른 수행에 단계를 추가하므로 테스트를 닫지 마십시오. "[연습 3c: 도구 상자 창을 사용하여 Select Flight 수행에 단계 추가](#)"(55페이지)로 이동하여 다른 수행에 다른 단계를 추가하십시오.

연습 3c: 도구 상자 창을 사용하여 Select Flight 수행에 단계 추가


"[연습 3b: 기록을 통해 FlightFinder 수행에 단계 추가](#)"(52페이지)에서 UFT의 기록 기능을 사용하여 Flight Finder 수행에 단계를 추가했습니다. 이를 통해 항공편 예약 응용 프로그램의 사용자 인터페이스에서 실행한 수행에 따라 단계를 만들었습니다.

이 연습에서는 도구 상자 창에서 관련 테스트 개체를 수행으로 끕니다.

참고: 도구 상자 창 항목은 문서 창에 포커스가 맞춰져 있는 수행에 따라 나열됩니다. 테스트 흐름 탭 또는 함수 라이브러리 탭에 포커스가 맞춰지거나 테스트를 전혀 열지 않으면 도구 상자 창이 비어 있습니다.

1. UFT를 시작하고 Book Flights 테스트를 엽니다.

- UFT가 현재 열려 있지 않으면 "[모든 테스트에 대한 솔루션 만들기](#)"(25페이지)에 설명된 대로 UFT를 엽니다.

- 열기** 버튼 아래쪽 화살표  열기 를 클릭하고 **솔루션 열기**를 선택합니다. 솔루션 열기 대화 상자가 열립니다.

- c. 솔루션 열기 대화 상자에서 **C:\%HOMEPATH%\My Documents\Unified Functional Testing**에 있는 **Flight Reservation Application.ftsln** 파일로 이동하고 열기를 클릭합니다.

"1단원: GUI 테스트 및 수행 만들기"(28페이지)에서 만든 Book Flights 테스트가 표시된 Flight Reservation Application 솔루션이 열립니다.

2. Select Flight 수행을 엽니다.

솔루션 탐색기 창에서 **Select Flight** 수행을 두 번 클릭합니다.

Select Flight 수행이 문서 창에서 개별 탭으로 열립니다.

3. 도구 상자 창을 표시합니다.

도구UFT모음에서 도구 상자 버튼  을 클릭합니다. UFT 창 오른쪽에서 도구 상자 창이 열립니다.

참고: 도구 상자 창에는 키워드 보기 또는 편집기에 수행이 표시되는지와 관계없이 수행의 테스트 개체 및 함수가 표시됩니다.

4. 표에서 예약할 항공편을 선택합니다.


- a. 도구 상자 창에서 **테스트 개체** 트리를 확장합니다.
- b. 항목 열에서 **새 단계** 버튼을 클릭합니다. 빈 편집 필드가 열립니다.
- c. 도구 상자 창에서 **flightsDataGrid** 개체를 찾아서 키워드 보기 또는 편집기(열려 있는 보기에 따라)의 편집 필드로 끕니다.

참고: 이 개체 저장소에는 Login 저장소보다 더 많은 개체가 있을 수 있습니다. 이는 이 개체 저장소를 만들 때 관련 없는 개체를 삭제하지 않았기 때문입니다.

선택한 테이블 테스트 개체가 기본 메서드 **SelectCell**과 함께 단계에 추가됩니다.

- 메서드의 필수 매개 변수를 제공하지 않았으므로 키워드 보기에서 **설명서** 셀이 비어 있습니다. 상위 테스트 개체가 단계의 일부이므로 이 단계는 키워드 보기에서 3개의 행에 표시되어야 합니다.
- 편집기에는 단계가 다음과 같이 표시됩니다.

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

- d. 키워드 보기가 열려 있지 않으면 **보기 > 키워드 보기**를 선택하여 엽니다.
- e. flightsDataGrid 개체에 대한 **값** 열의 중간에서 **값 구성** 아이콘  을 클릭합니다.

참고: 이 아이콘을 클릭하면 row, Column이라는 도구 설명이 표시됩니다. 이는 row 매개 변수에 대한 값을 설정하고 있음을 나타냅니다. 메서드 매개 변수의 값을 구성할 때 항상 설정 중인 매개 변수를 알리는 도구 설명이 표시됩니다.

- f. row 매개 변수 값으로 0을 입력합니다.

g. 값 열의 오른쪽에서 값 구성 아이콘 <#D> 을 클릭합니다.

참고: 아이콘을 클릭하면 row, Column이라는 도구 설명이 표시됩니다. 이는 **Column** 매개 변수에 대한 값을 설정하고 있음을 나타냅니다.

h. Column 매개 변수로 0을 입력합니다.

이 매개 변수를 입력하면 아래 예와 같이 첫 번째 행에서 항공편을 선택하도록 UFT에 지시한 것 입니다.

Price	From: London	To: Paris	Date	Flight
USD 138.60	06:15 PM	07:55 PM	25 Sep, 2015	11820 NW
USD 155.00	01:27 PM	03:07 PM	25 Sep, 2015	12274 NW
USD 165.50	08:00 AM	10:00 AM	25 Sep, 2015	12534 AF
USD 162.40	10:24 AM	12:24 PM	25 Sep, 2015	12538 AF
USD 165.60	12:48 PM	02:48 PM	25 Sep, 2015	12542 AF
USD 165.70	03:12 PM	05:12 PM	25 Sep, 2015	12546 AF
USD 168.20	09:51 AM	11:31 AM	25 Sep, 2015	12935 NW

메서드 매개 변수를 추가하면 키워드 보기 및 편집기에서 단계가 업데이트됩니다.

- 키워드 보기의 값 열에는 단계 값으로 "0", "0"이 표시됩니다.
- 편집기에는 **SelectCell** 메서드 뒤에 "0", "0"이 추가됩니다. 편집기에서 문이 다음과 같이 표시됩니다.

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

- 설명서 열에는 단계 수행을 설명하는 문이 있습니다.

5. Select Flight 버튼을 클릭하는 단계를 추가합니다.

항공편 목록에서 셀을 선택하고 나서 항공편 예약 프로세스를 계속하려면 Select Flight 버튼을 클릭해야 합니다.

- 도구 상자 창에서 **SELECT FLIGHT** 개체를 찾습니다.
- 항목 열에서 새 단계 버튼을 클릭합니다. 빈 편집 필드가 열립니다.
- 도구 상자 창에서 SELECT FLIGHT 개체를 키워드 보기에 있는 단계 표의 편집 필드로 끌어서 **flightsDataGrid** 개체 아래에 놓습니다.

SELECT FLIGHT 개체가 포함된 새 단계가 테스트에 추가됩니다.

항목	작업	값	설명
HP MyFlight Sample Application			
flightsDataGrid	SelectCell	"0","0"	"flightsDataGrid" table의 행 "0", 열 "0"의 셀을 선택합니다.
SELECT FLIGHT	Click		"SELECT FLIGHT" object을(를) 클릭합니다.
+ 새 단계			

이 경우 SELECT FLIGHT 개체의 기본 메서드인 **Click**이 테스트에 사용할 메서드입니다. 편집기에서 수행 단계가 다음과 같이 표시됩니다.

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

6. 테스트를 저장합니다.

파일 > 저장을 선택합니다.

다른 수행에 단계를 추가해야 하므로 테스트를 닫지 마십시오. ["연습 3d: 단계 생성기를 사용하여 Book Flight 수행에 단계 추가"\(58페이지\)](#)로 이동하여 단계 생성기를 통해 단계를 추가하십시오.

연습 3d: 단계 생성기를 사용하여 Book Flight 수행에 단계 추가

"[연습 3c: 도구 상자 창을 사용하여 Select Flight 수행에 단계 추가"\(55페이지\)](#)에서는 도구 상자 창에 표시된 개체를 사용하여 테스트 단계를 만들었습니다.

이 단원에서는 테스트 단계를 만드는 또 다른 방법인 단계 생성기를 사용합니다. 단계 생성기에서는 키워드 보기의 다양한 열에서 단계의 여러 부분을 삽입하는 대신 하나의 대화 상자에서 전체 단계를 정의할 수 있습니다.

1. Flight Confirmation 수행을 엽니다.


솔루션 탐색기의 **Book Flights** 노드 아래에서 **Flight Confirmation** 수행을 두 번 클릭합니다. Flight Confirmation 수행이 문서 창에서 개별 탭으로 열립니다.

2. 단계 생성기를 사용하여 단계를 추가합니다.

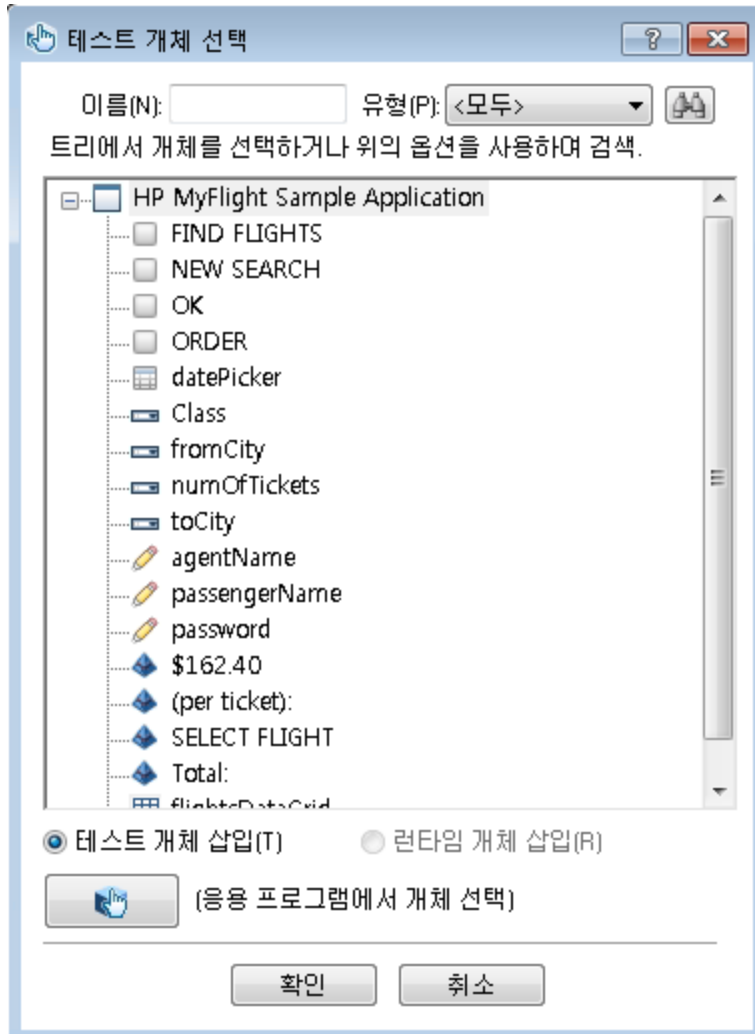
a. 편집기가 표시되어 있지 않으면 보기 > 편집기를 선택하여 편집기를 표시합니다.

- b. 편집기의 첫 번째 줄에서 마우스 오른쪽 버튼을 클릭하고 단계 삽입 > 단계 생성기를 선택합니다. 단계 생성기 대화 상자가 열립니다.

이름	유형	값
BUTTON	상수	<값 없음>

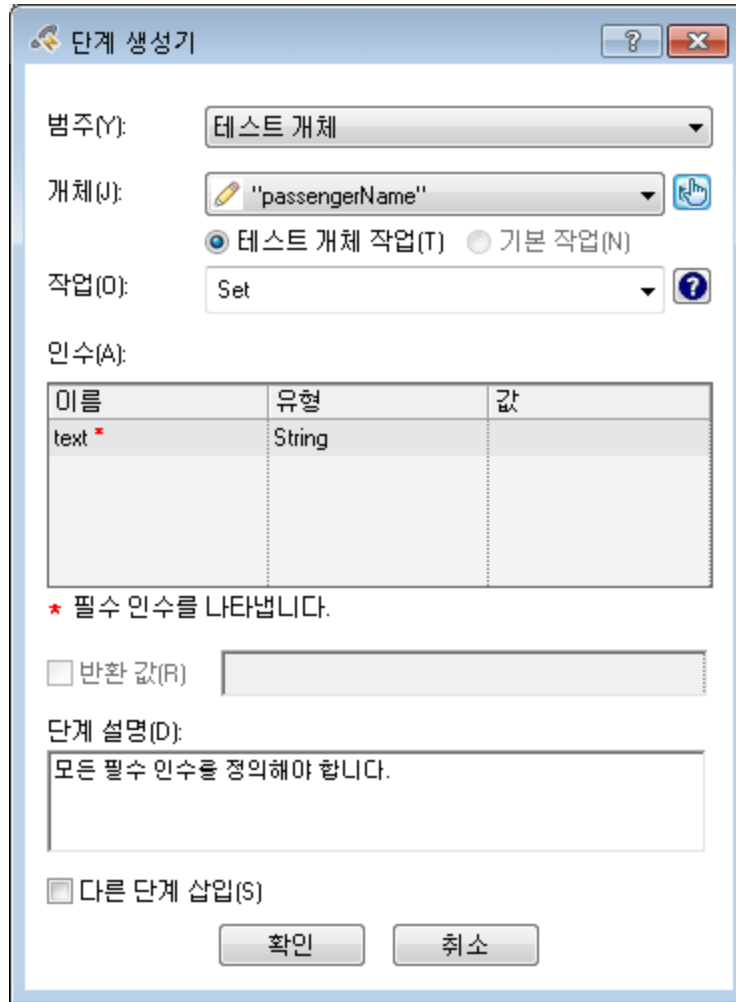
- c. 개체 드롭다운에서 "Book Flights"를 선택합니다.
- d. 개체 선택 버튼  을 클릭합니다. 테스트 개체 선택 대화 상자가 열립니다.

- e. 테스트 개체 선택 대화 상자에서 **Book Flights** 노드를 확장합니다.



- f. **passengerName** 편집 상자  를 선택하고 **확인**을 클릭합니다.

단계 생성기에서 **passengerName** 개체에 대한 기본 옵션을 표시합니다.



- g. 키워드 보기에서 정의하는 것처럼 단계 생성기에서 인수와 값을 정의합니다.
 - **작업** 드롭다운 목록에서 작업을 **Set**로 유지합니다. 개체 드롭다운 목록에서 테스트 개체를 선택할 경우 기본 작업이 표시됩니다. 필요에 따라 이 개체의 다른 작업을 선택할 수 있습니다. 그러나 이 연습에서는 기본 작업을 그대로 사용합니다.
 - **인수** 테이블에서 키워드 보기에서와 같이 **값** 열 내부를 클릭하고 선택한 이름을 입력합니다. 필수 인수인 경우 인수 이름 옆에 빨간색 별표가 표시됩니다.
 - **단계 설명**에는 키워드 보기의 **설명** 셀에 표시되는 대로 이 단계에 대한 지침이 표시됩니다.
 - **다른 단계 삽입** 확인란을 선택하여 이 단계를 추가한 후 다시 단계 생성기 대화 상자를 엽니다.
- h. **확인**을 클릭합니다. 단계 생성기는 열린 상태로 남아 있지만 백그라운드에서 입력한 세부 정보를 사용하여 편집기에 단계가 추가됩니다.

3. **응용 프로그램 창의 진행 표시줄이 로드될 때까지 대기하는 단계를 추가합니다.**

Flight Details 창에서 개체 저장소에 **progBar**라는 진행 표시줄 개체가 있음을 알 수 있습니다. 테스트

트할 때 응용 프로그램이 제대로 작동하려면 주문을 완료하도록 버튼을 클릭하기 전에 이 개체가 로드되는지 확인하는 단계를 추가해야 합니다.

- a. 단계 생성기의 **테스트 개체** 드롭다운 목록에서 **progBar** 개체를 선택합니다. 단계 생성기에서는 기본 메서드 **Value**를 비롯해 **progBar** 개체에 대한 기본 속성을 사용하여 대화 상자 필드를 업데이트합니다.
- b. **작업** 드롭다운 메뉴에서 **WaitProperty**를 선택합니다. 이 메서드는 특정 속성이 지정된 상태가 될 때까지 테스트 실행 중에 대기하도록 UFT에 지시합니다.
- c. **인수** 테이블에서 다음 정보를 입력합니다.

항목	값
PropertyName	value
PropertyValue	100

- d. **다른 단계 삽입 확인란**을 선택하고 **확인**을 클릭합니다.
백그라운드에서 또 다른 단계가 키워드 보기에 삽입되고 단계 생성기 상자는 열린 상태로 남아 있습니다.

4. **단계 생성기를 사용하여 주문을 완료하는 단계를 테스트에 추가합니다.**

주문 세부 정보를 정의했으므로 주문을 완료하려면 Flight Details 창에서 **ORDER** 및 **NEW SEARCH** 버튼을 클릭하는 단계를 제공해야 합니다. 다시 단계 생성기를 사용하여 이 문을 삽입합니다.

참고: **ORDER** 버튼을 클릭하는 단계는 이전 단계에서 만든 진행 표시줄 개체가 있는 단계 앞에 삽입되어야 합니다.

- a. 이전 단계의 프로세스를 사용하여 다음 세부 정보를 입력합니다.

	ORDER 버튼 단계	NEW SEARCH 버튼 단계
개체	ORDER(WpfButton)	NEW SEARCH(WpfButton)
작업	클릭	클릭
인수	공백 유지	공백 유지
다른 단계 삽입 확인란	선택	Clear

- b. **확인**을 클릭합니다. 단계 생성기가 닫히고 단계가 키워드 보기에 추가됩니다.
두 단계를 모두 테스트에 추가했으므로 키워드 보기가 다음과 같이 표시됩니다.

항목	작업	값	설명
HP MyFlight Sample Application			
passengerName	Set	"John Smih"	"John Smih"을(를) "passengerName" edit box에 입력합니다.
ORDER	Click		"ORDER" button을(를) 클릭합니다.
progBar	WaitProperty	"value", "100"	다음 단계를 계속하기 전에 "progBar" progress bar의 "valu...
NEW SEARCH	Click		"NEW SEARCH" button을(를) 클릭합니다.
+ 새 단계			

편집기에는 단계가 다음과 같이 표시됩니다.

```
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. 테스트를 저장합니다.

저장  을 클릭합니다.

첫 번째 테스트를 만들었으므로 테스트를 실행할 수 있습니다. ["4단원: GUI 테스트 실행 및 분석"\(69페이지\)](#)을 계속 진행하여 테스트 실행에 대해 자세히 알아보십시오.

고급 단원을 진행하기 원하는 경우 ["고급 연습 3e\(선택 사항\) - 편집기를 사용하여 단계 추가"\(63페이지\)](#)로 이동하여 편집기에서 단계를 추가하는 방법을 알아보십시오.

고급 연습 3e(선택 사항) - 편집기를 사용하여 단계 추가

키워드 보기, 도구 상자 창 또는 단계 생성기를 통해 테스트에 단계를 추가할 뿐 아니라 단계를 편집기에 바로 추가할 수도 있습니다.

그러나 편집기를 통해 단계를 추가할 경우 응용 프로그램과 테스트 개체를 더 잘 이해하고 있어야 합니다. 키워드 보기, 도구 상자 창 및 단계 생성기에서 테스트 개체에 대한 모든 정보는 UFT에서 대화 상자에 제공됩니다. 편집기에서는 다음과 같은 다양한 요소를 알아야 합니다.

- 테스트에 대한 전체 테스트 개체 계층 구조
- 테스트 개체 이름(개체 저장소에 기록된 대로 표시)
- 테스트 개체 유형(예: WpfWindow, WpfButton 등)
- 사용할 메서드

이 정보를 사용하여 편집기에서 줄을 만듭니다. 이 단원에서는 이 정보를 찾을 위치와 정보를 편집기에 입력하여 테스트 단계를 만드는 방법을 알아봅니다.

편집기에서 이미 테스트 단계가 있는 수행에 대한 문을 만듭니다.

1. UFT를 시작하고 Book Flights 테스트를 엽니다.

- UFT가 현재 열려 있지 않으면 ["모든 테스트에 대한 솔루션 만들기"\(25페이지\)](#)에 설명된 대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- 파일 > 열기 > 솔루션**을 선택합니다. 솔루션 열기 대화 상자가 열립니다.

- c. 솔루션 열기 대화 상자에서 **C:\%HOMEPATH%\My Documents\Unified Functional Testing**에 있는 **Flight Reservation Application.ftsln** 파일로 이동하고 **열기**를 클릭합니다.

"1단원: GUI 테스트 및 수행 만들기"(28페이지)에서 만든 Book Flights 테스트가 표시된 Flight Reservation Application 솔루션이 열립니다.

2. Flight Confirmation 수행을 엽니다.

솔루션 탐색기 창에서 **Flight Confirmation** 수행을 두 번 클릭합니다.

Flight Confirmation 수행이 문서 창에서 개별 탭으로 열립니다. 수행에서 기존 테스트 단계를 삭제하지 마십시오.

3. 상위 개체에 대한 개체 세부 정보를 알아봅니다.

- a. 편집기가 열려 있지 않으면 **보기 > 편집기**를 선택하여 편집기를 엽니다.

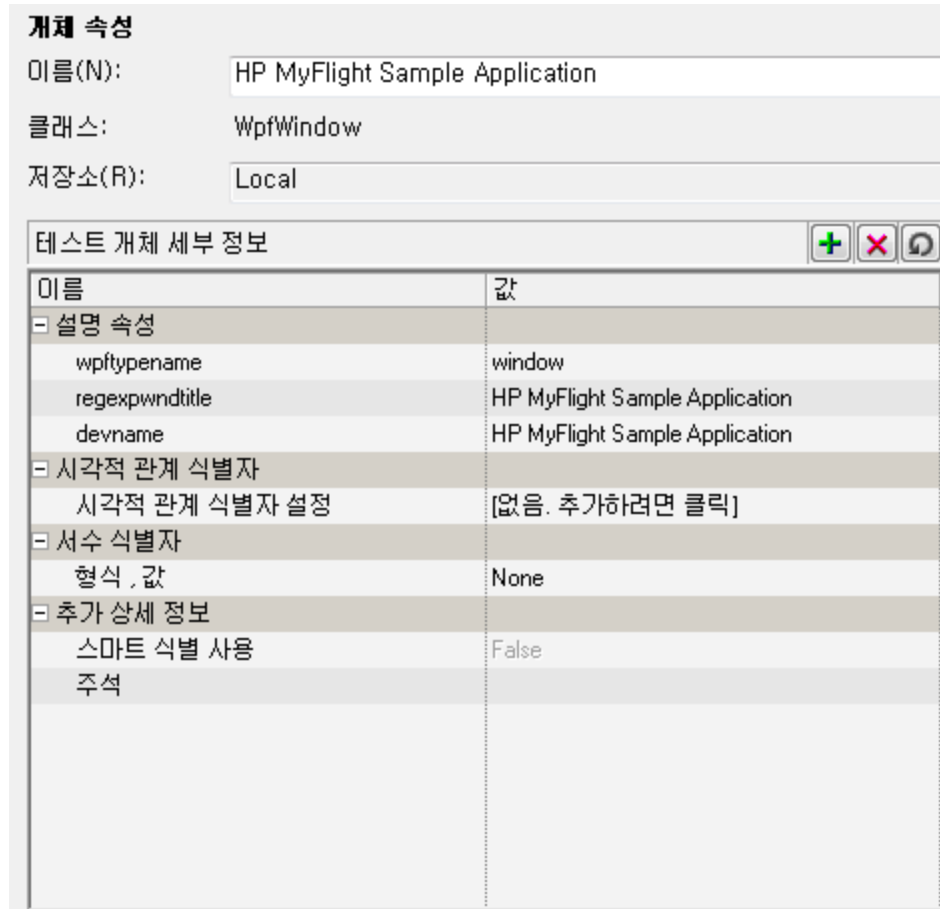
기존 단계가 다음과 같이 표시됩니다.

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

이러한 문에 있는 각 요소에 대한 설명은 "[키워드 보기 및 편집기에서 로그인 수행 분석](#)"(50페이지)에서 **편집기** 섹션을 참조하십시오.

- b. 솔루션 탐색기의 **Book Flights** 노드에서 **Flight Confirmation** 수행 노드를 확장합니다.
- c. **Flight Confirmation.tsr** 개체 저장소를 두 번 클릭합니다. 항공편 예약 응용 프로그램의 Flight Details 페이지에 대한 개체가 포함된 개체 저장소 관리자가 열립니다.
- d. 개체 저장소 관리자에서 **HP MyFlight Sample Application** 개체(최상위 노드)를 선택합니다. 이는 Flight Details 페이지에 있는 모든 개체의 상위 개체입니다.

개체의 세부 정보와 속성은 **개체 속성** 영역(창 오른쪽)에 표시됩니다.



e. **HP MyFlight Sample Application** 개체에 대해 다음 세부 정보를 기록합니다.

- **이름:** HP MyFlight Sample Application
- **클래스:** WpfWindow

편집기에서 문을 만들 때 이 정보가 필요합니다. 이 정보는 이 페이지에서 테스트 개체를 사용하는 단계에 대한 모든 문의 첫 번째 부분입니다.

4. **하위 개체에 대한 개체 세부 정보를 알아봅니다.**

"연습 3d: 단계 생성기를 사용하여 Book Flight 수행에 단계 추가"(58페이지)(이 연습의 기반이 되는 연습)에서 다음 두 단계를 만들었습니다. 한 단계에서는 항공편 주문의 이름을 입력하고 두 번째 단계에서는 **ORDER** 버튼을 클릭합니다. 이러한 단계를 편집기에서 만들기 위해서는 이러한 단계에 포함된 개체에 대한 개체 세부 정보도 알아야 합니다.

- a. 개체 저장소 관리자에서 **passengerName** 개체를 선택합니다. 테스트 개체 세부 정보가 개체 저장소 관리자의 **개체 속성** 섹션에 표시됩니다.



- b. 개체에 대해 다음 속성을 기록합니다.
 - o **이름:** passengerName
 - o **클래스:** WpfEdit
- c. **ORDER** 개체에 대해 같은 프로세스를 수행합니다.

5. 편집기에서 단계에 대한 문을 만듭니다.

이 수행 단계에 포함된 개체에 대한 개체 속성을 확인한 후 다음 정보를 제공해야 합니다.

개체	이름	클래스
HP MyFlight Sample Application 창(상위 개체)	HP MyFlight Sample Application	WpfWindow
Passenger Name (편집 상자)	passengerName	WpfEdit
ORDER 버튼	ORDER	WpfButton

개체 세부 정보를 사용하여 개체에서 수행할 메서드(수행) 및 개체 계층 구조를 포함하는 문을 만들어야 합니다. (각 개체에는 테스트 개체에 사용할 다양한 지원 메서드가 있습니다. 사용 가능한 모든 개체 및 해당 메서드에 대한 자세한 내용은 자습서 연습을 완료한 후 *HP UFT Object Model Reference for GUI Testing*을 참조하십시오.

- a. 첫 번째 새 줄에서 다음 형식을 사용하여 첫 번째 단계에 대한 상위 개체 계층 구조를 입력합니다(주문에 대한 고객 이름 입력).

```
<object class>("<object name>").
```

이 단계에서 다음을 입력해야 합니다.

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

- b. 같은 형식을 사용하여 첫 번째 단계에 대한 하위 개체(**passengerName**)를 입력합니다. 이 단계에서 다음을 입력해야 합니다.

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

- c. **WpfEdit("passengerName")** 개체 뒤에 **passengerName** 개체에 대한 **Set** 메서드를 입력합니다.

단계가 다음과 같이 표시됩니다.

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

- d. **Set** 메서드 뒤에 **passengerName** 개체에 대해 입력할(Set) 문자열로 "John Smith"를 입력합니다.

단계가 다음과 같이 표시됩니다.

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

- e. 위에서 설명한 같은 프로세스를 사용하여 두 번째 단계에 대한 개체 계층 구조를 추가합니다 (**ORDER** 버튼 클릭).

두 번째 단계에 대한 개체 계층 구조를 추가하면 문이 다음과 같이 표시됩니다.

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

- f. **WpfButton("ORDER")** 개체 뒤에 **Click** 메서드를 추가합니다. 문이 다음과 같이 표시됩니다.

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

참고: **Click** 메서드에는 매개 변수가 필요하지 않으므로 메서드 이름 뒤에 다른 정보를 추가할 필요가 없습니다.

두 문을 모두 완료하면 편집기에 다음 내용이 표시됩니다.

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

6. 추가 줄을 제거합니다.

이전 단계를 수행한 후 총 4개 문에 대해 두 개의 추가 줄을 수행에 추가했습니다. 수행에서 마지막 두 개의 문을 제거하여 테스트가 성공적으로 실행되는지 확인합니다.

7. 테스트를 저장합니다.

파일 > 저장을 선택합니다.

4단원: GUI 테스트 실행 및 분석

"3단원: 테스트에 단계 추가"(44페이지)에서는 여러 수행 및 각 수행의 테스트 단계를 만들어서 항공편 예약 응용 프로그램을 테스트했습니다. 테스트가 완료되었으므로 이제 테스트를 실행하여 항공편 응용 프로그램이 어떻게 작동하는지 확인할 수 있습니다.

이 단원에서는 테스트를 실행하고 실행 결과를 보는 방법을 알아봅니다.

이 단원의 내용은 다음과 같습니다.

- 연습 4a: 테스트 실행 70
- 연습 4b: 실행 결과 탐색 72
- 연습 4c: 실행 결과 분석 74

연습 4a: 테스트 실행

"3단원: 테스트에 단계 추가"(44페이지)에서 항공편을 예약하기 위해 항공편 예약 응용 프로그램을 통해 실행되는 기본 테스트를 만들었습니다.

이 연습에서는 방금 마친 테스트를 실행하는 방법을 알아봅니다.

1. UFT를 시작하고 Book Flights 테스트를 엽니다.

- UFT가 현재 열려 있지 않으면 "모든 테스트에 대한 솔루션 만들기"(25페이지)에 설명된 대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- 파일 > 열기 > 솔루션**을 선택합니다. 솔루션 열기 대화 상자가 열립니다.
- 솔루션 열기 대화 상자에서 **C:\%HOMEPATH%\My Documents\Unified Functional Testing**에 있는 **Flight Reservation Application.ftsln** 파일로 이동하고 **열기**를 클릭합니다.
Flight Reservation Application 솔루션이 솔루션 탐색기에서 열리고 Book Flights 테스트가 문서 창에서 개별 탭으로 열립니다.


2. 모든 이미지를 실행 결과에 저장하도록 UFT를 구성합니다.

테스트 실행을 수행하면 UFT에서는 모든 이미지를 테스트 결과에 저장하는 옵션을 제공합니다.

- 도구 > 옵션 > GUI 테스트 탭 > 화면 캡처** 노드를 선택합니다. 화면 캡처 옵션 창이 열립니다.
- 화면 캡처 옵션 창에서 **정지 이미지 캡처를 결과에 저장** 확인란을 선택하고 드롭다운 메뉴에서 **항상**을 선택합니다.
- 확인**을 클릭하여 옵션 대화 상자를 닫습니다.

3. 테스트에 대한 기록 및 실행 설정을 구성합니다.


경우에 따라 테스트 실행 시작 시 UFT에서 응용 프로그램을 자동으로 열도록 해야 할 수 있습니다. 이 경우 기록 및 실행 설정을 지정하여 이 기능을 사용하도록 설정할 수 있습니다.

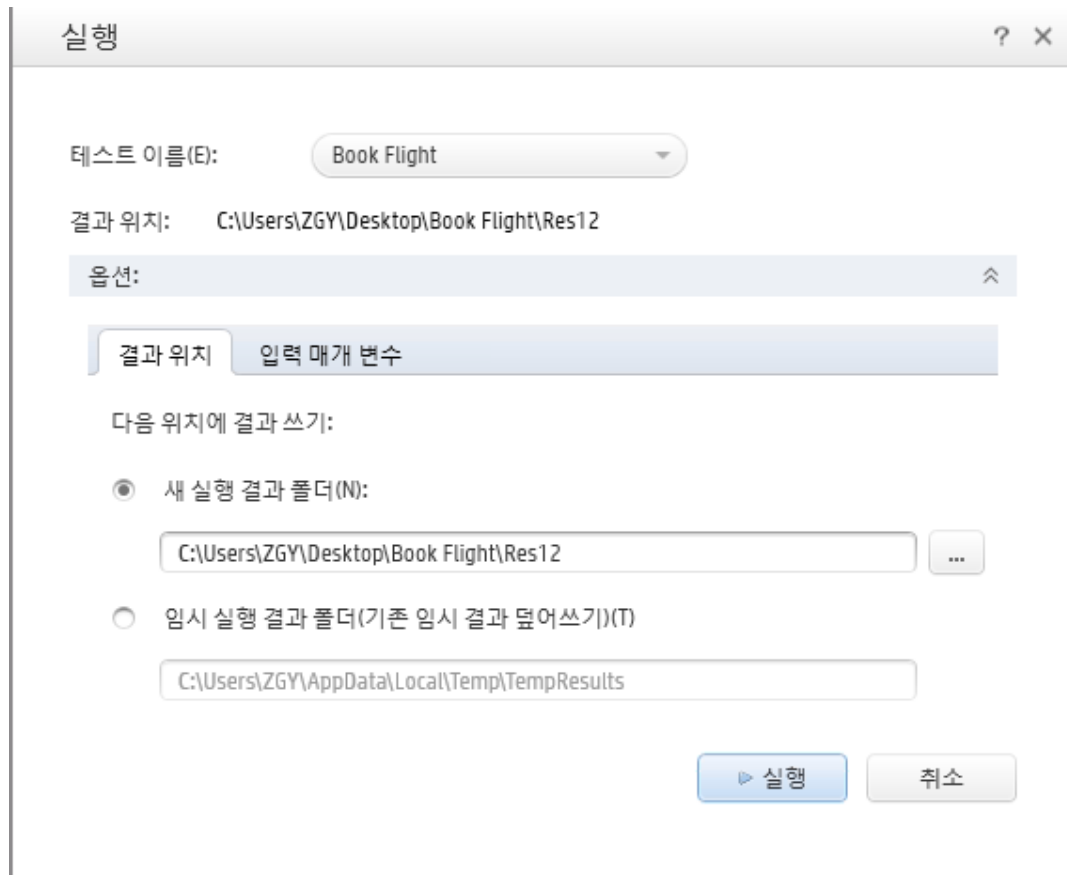
- 기록 > 기록 및 실행 설정**을 선택합니다. 기록 및 실행 설정 대화 상자가 열립니다.
- 기록 및 실행 설정 대화 상자의 Windows 응용 프로그램 탭에서 **기록 및 실행 장소**: 옵션을 선택합니다.
- 기록 및 실행 장소 옵션에서 **아래 지정된 응용 프로그램** 옵션을 선택합니다.
- 응용 프로그램 세부 정보 영역에서 **추가** 버튼 을 클릭합니다. 응용 프로그램 세부 정보 대화 상자가 열립니다.
- 응용 프로그램 세부 정보 대화 상자에서 응용 프로그램 세부 정보를 입력합니다.

응용 프로그램	C:\%HOMEPATH%\Unified Functional Testing\samples\Flights Application\FlightsGUI.exe
작업 폴더	C:\%HOMEPATH%\Unified Functional Testing\samples\Flights Application\

- f. **확인**을 클릭하여 응용 프로그램 세부 정보 대화 상자를 닫습니다.
- g. 기록 및 실행 설정 대화 상자에서 **적용**을 클릭하고 **확인**을 클릭하여 설정을 사용하도록 설정하고 대화 상자를 닫습니다.

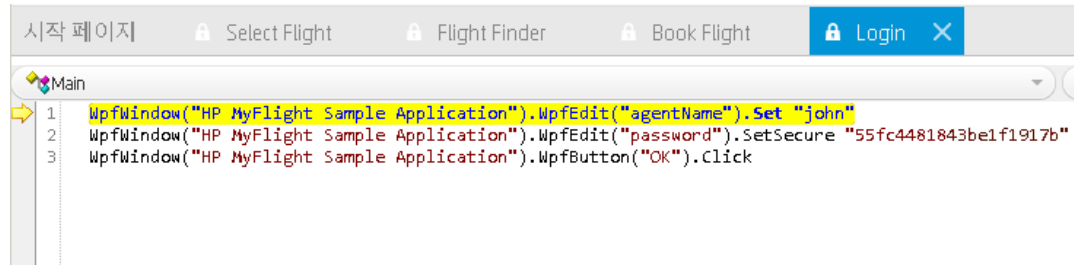
4. **테스트 실행을 시작합니다.**

- a. 솔루션 탐색기에서 **Book Flights** 테스트 노드를 선택합니다.
- b. 도구 모음에서 **실행** 버튼  을 클릭합니다. 테스트 실행 대화 상자가 열립니다.
- c. 테스트 실행 대화 상자에서 **옵션** 막대를 클릭하여 테스트 실행 옵션 영역을 확장하고 **새 실행 결과 폴더** 옵션이 선택되었는지 확인합니다. 기본 폴더 이름을 그대로 사용합니다.



- d. **실행**을 클릭하여 실행 대화 상자를 닫고 테스트 실행을 시작합니다.

UFT에서 응용 프로그램을 열고 테스트 실행을 시작할 때 주의해서 지켜보십시오. 응용 프로그램에서 UFT가 삽입된 각 단계를 수행하는지 확인할 수 있습니다. 키워드 보기의 왼쪽 가장자리에 있는 노란색 화살표와 강조 표시된 행은 UFT에서 실행 중인 단계를 나타냅니다.



오류가 나타나면 테스트에서 오류 메시지에 표시된 지점으로 이동하여 단계가 관련 연습에 설명된 대로 구성되었는지 확인합니다.

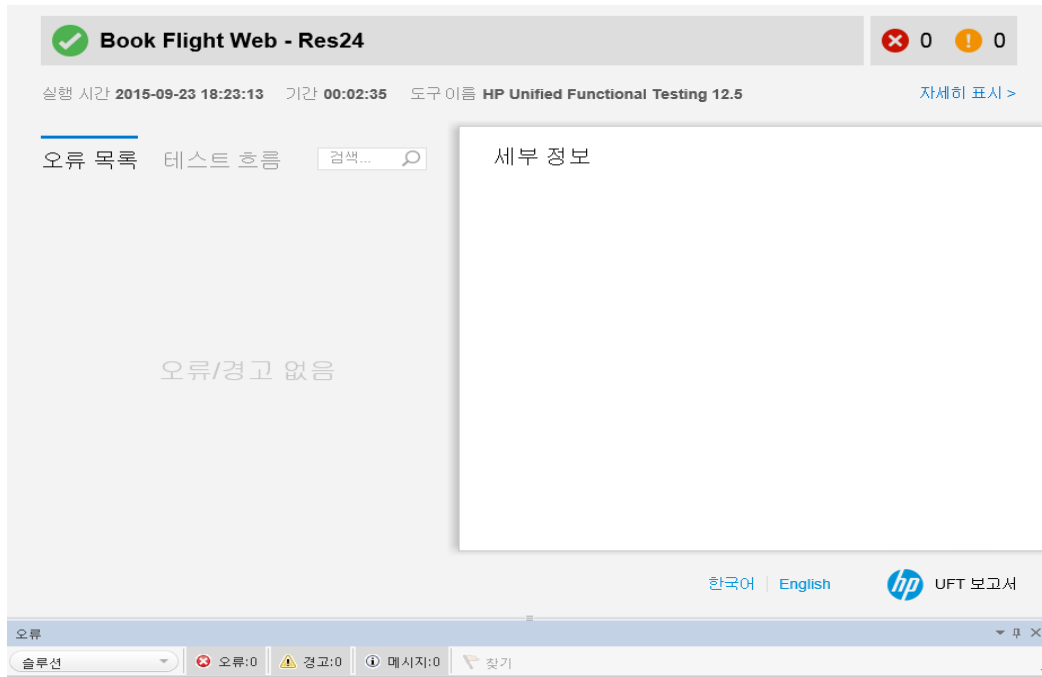
테스트 실행이 완료되면 실행 결과가 문서 창에서 개별 탭으로 열립니다. "[연습 4b: 실행 결과 탐색](#)" (72페이지)으로 이동하여 실행 결과에 대해 자세히 알아보십시오.

연습 4b: 실행 결과 탐색

"[연습 4a: 테스트 실행](#)" (70페이지)에서는 직접 만든 Book Flights 테스트를 실행했습니다. 테스트 실행이 완료되면 실행 결과에 이 테스트 실행에 대한 결과가 자동으로 표시됩니다.

참고: 기본적으로 실행 결과는 HTML 기반 보고서로 표시됩니다. 실행 결과가 Run Results Viewer에 표시되는 옵션 대화 상자의 실행 세션 창(도구 > 옵션 > 일반 탭 > 실행 세션 노드)에 표시되도록 선택할 수도 있습니다. 이 자습서의 단원은 HTML 기반 보고서에 기반을 둡니다.

실행 결과가 열리면 UFT에서 다음 내용을 표시합니다.



기본적으로 실행 결과에는 다음 내용이 표시됩니다.

<p>테스트 흐름</p>	<p>테스트 실행 중에 방문한 응용 프로그램 페이지 및 수행에 따라 구성된, 트리에 있는 결과의 그래픽 표시입니다. 실행 중에 수행되는 단계는 트리에서 아이콘으로 표시되고, 트리를 확장하여(화살표) 각 단계를 볼 수 있습니다. 각 실행에서 여러 데이터 집합을 사용하여 두 번 이상 테스트나 수행을 실행하도록 UFT에 지시할 수 있습니다. 각 실행을 반복이라고 하며 각 반복은 번호가 매겨집니다. 실행한 테스트에는 하나의 반복만 포함됩니다.</p>
<p>오류 목록</p>	<p>목록으로 제공되는 모든 오류 및 경고입니다.</p>
<p>단계 요약 정보</p>	<p>테스트에 대한 일반 정보, 통과하거나 실패한 단계, 각 테스트 단계에 대한 세부 정보를 포함하는 상위 수준 결과 개요 보고서입니다.</p>
<p>외부 리소스 링크</p>	<p>다음을 포함하여 테스트 또는 테스트 실행에서 사용되는 외부 리소스 링크입니다.</p> <ul style="list-style-type: none"> • 데이터 테이블 • 런타임 동영상 • 응용 프로그램 로그

이 연습에서는 추가한 단계에 따라 UFT에서 항공편 예약 응용 프로그램을 탐색할 수 있으므로 테스트 실행에 성공했습니다. 오류가 발생하고 테스트가 성공적으로 실행되지 않으면 오류가 실행 결과에 표시됩니다. 이 경우 돌아가서 이 자습서의 설명대로 단계가 정확히 구성되었는지 확인합니다.

실행 결과에 표시되는 내용을 알아보았으므로 이제 "연습 4c: 실행 결과 분석"(74페이지)으로 이동하여 실행 결과의 세부 정보를 알아보십시오.

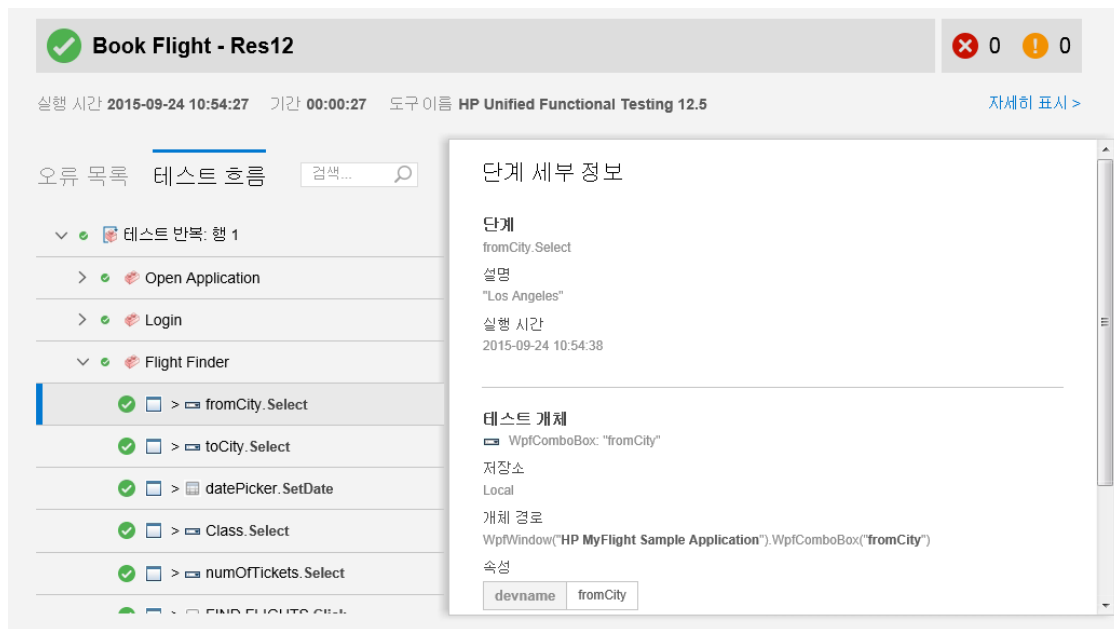
연습 4c: 실행 결과 분석

이 연습에서는 "연습 4a: 테스트 실행"(70페이지)에서 테스트를 실행할 때 UFT에서 수행된 단계를 검사합니다. 각 단계에 대한 응용 프로그램 창의 스냅샷을 볼 수 있습니다.

1. 특정 단계에 대한 실행 결과를 봅니다.

테스트 흐름의 결과 트리에서 **Flight Finder** 노드를 찾고 항공편 예약 응용 프로그램의 Flight Finder 페이지에서 수행된 모든 단계를 확인합니다.

결과 트리에서 **fromCity.Select** 단계를 선택합니다.



실행 결과에 다음 정보가 표시됩니다.

- 테스트 흐름 및 강조 표시된 단계
- 강조 표시된 단계의 세부 정보가 표시된 테스트 단계 요약

2. 실행 결과를 닫습니다.

문서 창에서 실행 결과가 포함된 탭을 닫습니다.

첫 번째 테스트를 설정하고 실행했으므로 계속해서 테스트를 향상하는 다양한 방법을 알아볼 수 있습니다. 다음 내용 중 하나를 선택하여 자세히 알아보십시오.

- "5단원: 단계 및 개체 매개 변수화"(75페이지)
- "6단원: 검사점 및 출력 값 만들기"(89페이지)
- "7단원: 함수 및 함수 라이브러리 만들기"(120페이지)
- "8단원: 테스트에서 이미지 인식 사용"(129페이지)

5단원: 단계 및 개체 매개 변수화

"3단원: 테스트에 단계 추가"(44페이지)에서는 테스트 단계를 만들어서 항공편 예약 응용 프로그램에서 수행되는 일련의 단계가 원활하게 실행되는지 확인했습니다. "4단원: GUI 테스트 실행 및 분석"(69페이지)에서는 단일 데이터 집합만 사용하여 테스트를 실행했습니다. 하지만 사용자 응용 프로그램을 테스트할 때 여러 데이터 집합을 사용하여 수행되는 동일한 작업을 확인할 수도 있습니다.

예를 들어 10개의 서로 다른 데이터 집합을 사용하여 응용 프로그램에 대한 테스트를 실행할 수 있습니다. 각각 고유한 데이터 집합이 포함된 10개의 개별 테스트를 만들거나 10개의 매개 변수 집합을 단일 테스트에 추가할 수 있습니다. 매개 변수를 추가하면 테스트가 10회 실행되고 매번 다른 데이터 집합을 사용합니다.

이 단원에서는 테스트에 매개 변수를 추가하고 여러 데이터 집합을 사용하여 테스트를 실행합니다.

이 단원의 내용은 다음과 같습니다.


- [테스트, 수행 및 개체 매개 변수화 - 개요](#)76
- [연습 5a: 매개 변수화에 대한 테스트 만들기](#)76
- [연습 5b: 데이터 테이블 매개 변수 정의](#)77
- [연습 5c: 데이터 테이블에 매개 변수 값 추가](#) 81
- [연습 5d: 매개 변수화된 테스트 실행](#)83

테스트, 수행 및 개체 매개 변수화 - 개요

데이터를 사용하여 테스트를 매개 변수화하면 여러 위치에서 데이터 원본을 제공할 수 있습니다.

- **데이터 테이블:** 매개 변수 이름 및 값이 포함된 Excel 스프레드시트입니다.
- **환경 변수:** 고정 값으로 테스트에 설정된 변수입니다.
- **임의의 숫자:** 테스트 실행 시 UFT에서 생성되는 임의의 숫자입니다.

이러한 매개 변수 중 가장 일반적인 매개 변수는 데이터 테이블 매개 변수입니다. 데이터 테이블은 데이터 창에서 UFT 창 아래쪽에 표시되는 Excel 스프레드시트입니다.

참고: 데이터 창이 표시되지 않으면 [보기 > 데이터](#)를 선택하거나 도구 모음에서 **데이터 버튼**  을 클릭합니다.

데이터 테이블에는 다음과 같은 두 가지 유형의 시트가 있습니다.

<p>글로벌 데이터 시트</p>	<p>글로벌 데이터 시트에는 테스트의 모든 수행에 사용되고 사용할 수 있는 데이터 매개 변수와 데이터가 포함되어 있습니다. 매개 변수가 글로벌 시트에 삽입되면 수행 및 테스트의 수행 단계에서 이 매개 변수를 사용할 수 있습니다.</p> <p>테스트에서는 글로벌 데이터 시트에 있는 행 수와 같은 수만큼 반복을 실행합니다. 따라서 5개의 데이터 행이 있으면 테스트에서는 5개 반복을 실행합니다.</p>
<p>수행 시트</p>	<p>UFT에서는 테스트의 각 수행에 대해 해당 수행이 포함된 시트를 수행과 같은 이름으로 추가합니다. 데이터 매개 변수 및 데이터는 해당 수행의 단계에만 사용할 수 있습니다.</p> <p>한 수행 시트 내에서 여러 데이터 행을 사용하면 UFT에서는 하나의 테스트 반복 내에서 데이터 시트에 있는 행 수와 같은 횟수만큼 수행을 실행합니다.</p>

이 단원에서는 데이터 테이블 매개 변수만 사용합니다. 기타 매개 변수 유형에 대한 자세한 내용은 *HP Unified Functional Testing User Guide*의 매개 변수화 섹션을 참조하십시오.

연습 5a: 매개 변수화에 대한 테스트 만들기

"[연습 3b: 기록을 통해 FlightFinder 수행에 단계 추가](#)"(52페이지)에서 로스앤젤레스에서 시드니로 가는 항공편을 예약했습니다. 그러한 단계에서 로스앤젤레스 및 시드니 값은 상수 값입니다. 이는 UFT가 테스트를 실행할 때마다 **Los Angeles** 및 **Sydney**를 출발 및 도착 도시로 사용함을 의미합니다.

이 연습에서는 출발 및 도착 도시를 매개 변수로 정의하여 각 테스트 실행에 대해 다른 출발 및 도착 도시를 사용할 수 있는 새 테스트를 만듭니다.

1. UFT를 시작하고 Book Flights 테스트를 엽니다.

- a. "[모든 테스트에 대한 솔루션 만들기](#)"(25페이지)의 설명대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- b. 시작 페이지의 **최근에 사용한 솔루션** 영역에서 **Flight Reservation Application**을 클릭합니다.

"1단원: GUI 테스트 및 수행 만들기"(28페이지)에서 만든 **Book Flights** 테스트가 포함된 Flight Reservation Application 솔루션이 열립니다.

2. Book Flights를 Book Flights Parameter로 저장합니다.

- a. 솔루션 탐색기에서 **Book Flights** 노드를 선택합니다.
- b. **파일 > 다른 이름으로 저장**을 선택합니다. 다른 이름으로 저장 대화 상자에서 **C:\%HOMEPATH%\My Documents\Unified Functional Testing**으로 이동하고 테스트를 Book Flights Parameter로 저장합니다.
 솔루션 탐색기에서 **Book Flights** 테스트가 새 **Book Flights Parameter** 테스트로 대체됩니다. Book Flights 테스트는 파일 시스템에 별도로 저장되어 있습니다.

3. Book Flights 테스트를 다시 솔루션에 추가합니다.

두 테스트가 모두 같은 솔루션에서 참조된다면 모든 테스트를 동시에 열 수 있습니다. 이를 통해 테스트를 비교하거나 편집하려는 경우 테스트 사이에서 앞뒤로 전환할 수 있습니다. 단일 테스트를 한 번만 실행할 수 있습니다.

- a. **파일 > 추가 > 기존 테스트**를 선택합니다.
- b. 기존 테스트 추가 대화 상자에서 **C:\%HOMEPATH%\My Documents\Unified Functional Testing**으로 이동하고 **Book Flights** 테스트를 선택합니다.
- c. **추가**를 클릭하여 테스트를 솔루션에 추가합니다.
 Book Flights 테스트가 다시 솔루션 탐색기에 표시됩니다. 테스트는 알파벳순으로 나열되므로 이 테스트는 방금 만든 Book Flights Parameter 테스트 위에 표시됩니다.

매개 변수화에 대한 테스트를 만들었으므로 "**연습 5b: 데이터 테이블 매개 변수 정의**"(77페이지)를 계속 진행하여 테스트에서 데이터 테이블 매개 변수를 만드십시오.

연습 5b: 데이터 테이블 매개 변수 정의

이 단원에서는 출발 및 도착 도시를 매개 변수로 정의하므로 각 테스트 실행에 대해 다른 출발 도시를 사용할 수 있습니다.

1. UFT를 시작하고 필요한 경우 Book Flights Parameter 테스트를 엽니다.

- a. "**모든 테스트에 대한 솔루션 만들기**"(25페이지)의 설명대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- b. **파일 > 열기 > 솔루션**을 선택합니다. 솔루션 열기 대화 상자가 열립니다.
- c. **C:\%HOMEPATH%\My Documents\Unified Functional Testing**에 있는 **Flight Reservation Application.ftsln** 파일로 이동하고 **열기**를 클릭합니다.
 "**연습 5a: 매개 변수화에 대한 테스트 만들기**"(76페이지)에서 만든 Book Flights Parameter 테스트가 포함된 항공편 예약 솔루션이 열립니다.
- d. 솔루션 탐색기에서 **Book Flights Parameter** 노드를 선택합니다.

2. **데이터 창이 표시되는지 확인합니다.**

UFT 창 아래쪽에 데이터 창이 표시되지 않으면 보기 > 데이터를 선택합니다.

3. **Flight Finder 수행을 엽니다.**

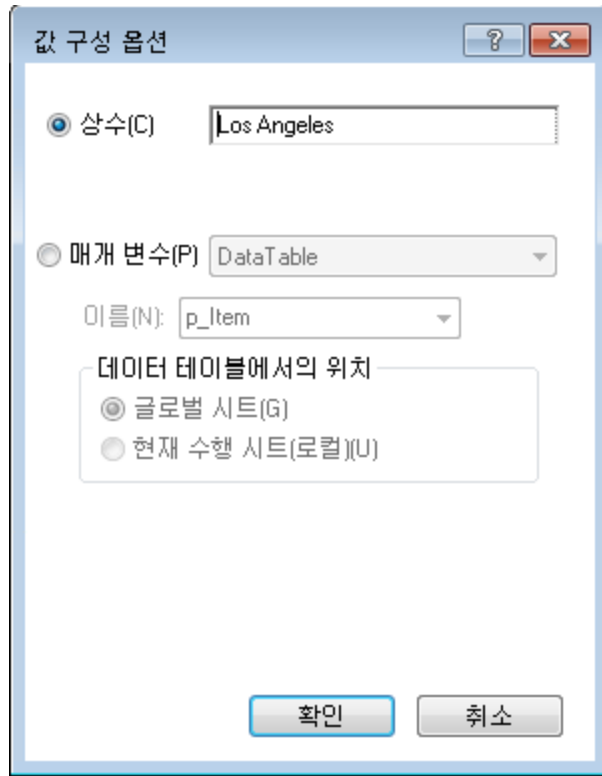
- a. 캔버스에서 **Flight Finder** 수행을 두 번 클릭합니다. Flight Finder 수행이 문서 창에서 개별 탭으로 표시됩니다.
- b. 필요한 경우 보기 > 키워드 보기를 선택합니다.

4. **매개 변수화할 텍스트를 선택합니다.**

키워드 보기의 **fromCity** 행에서 **값** 셀을 클릭하고 매개 변수화 버튼 을 클릭합니다. 매개 변수 목록이 표시됩니다.

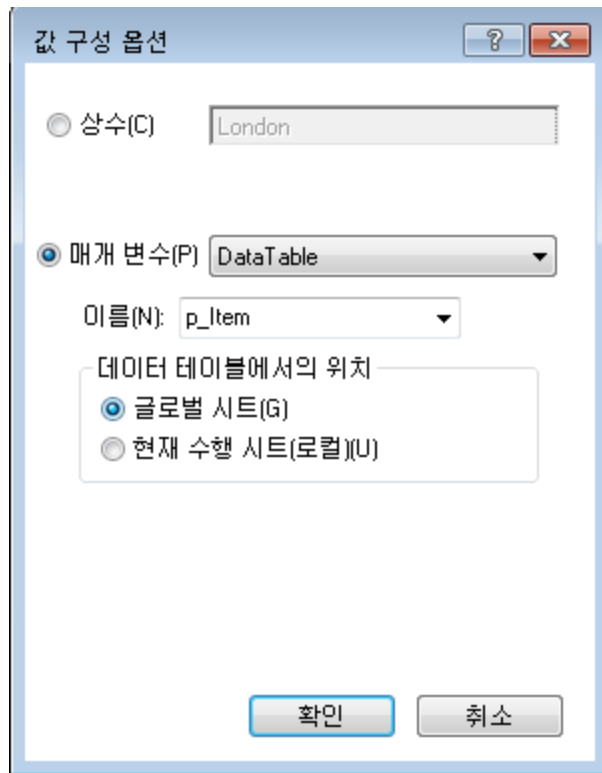
5. **매개 변수화 속성을 설정합니다.**

- a. 매개 변수 목록에서 **데이터 테이블** 탭을 선택합니다. 이를 통해 상수 값(**London**)을 매개 변수로 바꿀 수 있습니다. 아직 데이터 테이블 매개 변수를 만들지 않았으므로 매개 변수가 표시되지 않습니다.
- b. 매개 변수 목록의 데이터 테이블 탭에서 **새 매개 변수 추가** 버튼을 클릭합니다. 값 구성 옵션 대화 상자가 열립니다.



- c. 값 구성 옵션 대화 상자에서 매개 변수 라디오 버튼을 선택합니다.

- d. 매개 변수 드롭다운 메뉴에서 **DataTable** 옵션이 선택되었는지 확인합니다. 이는 매개 변수 값을 UFT 데이터 창에서 가져옴을 의미합니다. 이름 상자가 활성화되고 **p_Item**이 표시됩니다.



- e. **p_Item** 매개 변수를 삭제하고 **fromCity**를 입력합니다.
- f. **확인**을 클릭하여 대화 상자를 닫습니다.

UFT에서 **fromCity** 매개 변수를 데이터 창에 새 열로 추가하고 Los Angeles(이전 상수 값)를 열의 첫 번째 행에 삽입합니다.

Los Angeles는 UFT에서 응용 프로그램의 테스트 실행 중에 사용할 여러 출발 도시 중 첫 번째 도시가 됩니다.

항목	작업	값	설명
HP MyFlight Sample Application			
fromCity	Select	DataTable("fromCity", dtGlobalS...	"fromCity" list에서 <the 'fromCity' Data Table column의 값> ...
toCity	Select	Sydney	"toCity" list에서 Sydney 항목을 선택합니다.
datePicker	SetDate	"22-Sep-2016"	"datePicker" calendar의 날짜 또는 날짜 범위를 "22-Sep-20...
Class	Select	"Business"	"Class" list에서 "Business" 항목을 선택합니다.
numOfTickets	Select	"3"	"numOfTickets" list에서 "3" 항목을 선택합니다.
FIND FLIGHTS	Click		"FIND FLIGHTS" button을(를) 클릭합니다.
+ 새 단계			

키워드 보기에서 단계 모양의 변경 사항을 확인합니다. 이전에는 단계가 fromCity Select Los Angeles로 표시되었습니다. 이제 **값** 셀을 클릭하면 다음 정보가 표시되어 값이 fromCity라는 데이터 창 매개 변수를 사용하여 매개 변수화됨을 나타냅니다.

DataTable("fromCity", dtGlobalS...

6. toCity 단계에 대한 데이터 테이블 매개 변수를 추가합니다.

이전 단계에서 설명한 프로세스를 사용하여 toCity 개체에 대한 데이터 테이블 매개 변수를 toCity 라는 이름으로 추가합니다.

작업이 완료되면 테스트가 다음과 같이 표시됩니다.

항목	작업	값	설명
HP MyFlight Sample Application			
fromCity	Select	DataTable("fromCity", dtGlobal...	"fromCity" list에서 <the 'fromCity' Data Table column의 값> ...
toCity	Select	DataTable("toCity", dtGlobalSh...	"toCity" list에서 <the 'toCity' Data Table column의 값> 항목...
datePicker	SetDate	"22-Sep-2016"	"datePicker" calendar의 날짜 또는 날짜 범위를 "22-Sep-20...
Class	Select	"Business"	"Class" list에서 "Business" 항목을 선택합니다.
numOfTickets	Select	"3"	"numOfTickets" list에서 "3" 항목을 선택합니다.
FIND FLIGHTS	Click		"FIND FLIGHTS" button을(를) 클릭합니다.
+ 새 단계			

7. 테스트를 저장합니다.


파일 > 저장을 선택합니다.

"연습 5c: 데이터 테이블에 매개 변수 값 추가"(81페이지)를 계속 진행하여 데이터 테이블을 테스트 실행에 사용된 값으로 채우는 방법을 알아보십시오.

연습 5c: 데이터 테이블에 매개 변수 값 추가

"연습 5b: 데이터 테이블 매개 변수 정의"(77페이지)에서 설명한 대로 UFT에서는 매개 변수 값을 데이터 창에 표시합니다. 이 연습에서는 데이터 창에 fromCity 개체에 대한 다른 출발 도시를 추가하므로 UFT에서 이 데이터로 응용 프로그램을 테스트할 수 있습니다.

1. UFT를 시작하고 필요한 경우 Book Flights Parameter 테스트를 엽니다.

- "모든 테스트에 대한 솔루션 만들기"(25페이지)의 설명대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- 열기 아래쪽 화살표  열기 를 클릭하고 솔루션 열기를 선택합니다. 솔루션 열기 대화 상자가 열립니다.
- C:\%HOMEPATH%\My Documents\Unified Functional Testing에 있는 Flight Reservation Application.ftsln 파일로 이동하고 열기를 클릭합니다.
"연습 5a: 매개 변수화에 대한 테스트 만들기"(76페이지)에서 만든 Book Flights Parameter 테스트가 포함된 항공편 예약 솔루션이 열립니다.
- 솔루션 탐색기에서 Book Flights Parameter 노드를 선택합니다.

2. Flight Finder 수행을 엽니다.

솔루션 탐색기에서 **Flight Finder** 수행을 두 번 클릭합니다.

Flight Finder 수행이 문서 창에서 개별 탭으로 열립니다.

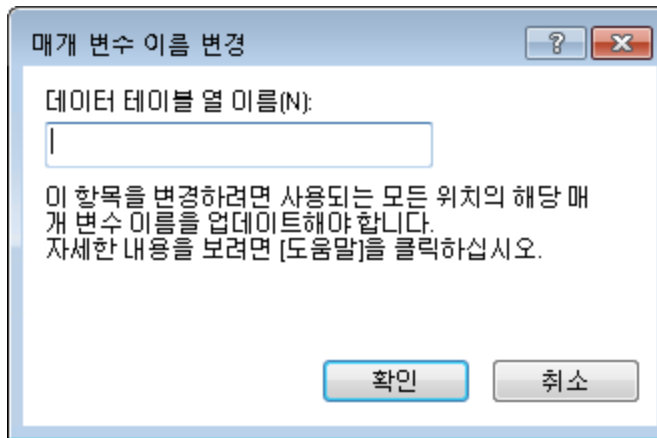
3. **fromCity 열에 추가 도시를 입력합니다.**

데이터 창에서 **fromCity** 매개 변수에 대한 데이터 창에 다음을 입력합니다.

행	값
2	Denver
3	Frankfurt
4	London

4. **toCity 개체에 대한 데이터 테이블 매개 변수 및 값을 만듭니다.**

- a. 데이터 창에서 **B** 열에 대한 머리글 행을 두 번 클릭합니다. 매개 변수 이름 변경 대화 상자가 열립니다.



- b. 매개 변수 이름 변경 대화 상자에서 매개 변수 이름으로 **toCity**를 입력하고 **확인**을 클릭합니다. 이전에 **B**였던 열 머리글이 새 매개 변수 이름으로 업데이트됩니다.

	fromCity	toCity	C
1	Los Angeles		
2			
3			
4			
5			

Global / Login / Flight Finder /

- c. 다음과 같이 **toCity** 매개 변수의 값을 입력합니다.


행	값
1	Sydney
2	Los Angeles

3	London
4	Frankfurt

두 번째 매개 변수 및 해당 값을 추가하면 데이터 창이 다음과 같이 표시됩니다.

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

5. toCity 단계를 매개 변수화합니다.

- Flight Finder 수행의 **toCity** 행에서 **값** 셀을 클릭하고 매개 변수화 버튼  을 클릭합니다. 값 구성 옵션 대화 상자가 열립니다.
- 값 구성 옵션 대화 상자에서 **매개 변수** 라디오 버튼을 선택합니다.
- 매개 변수** 유형 드롭다운 목록에서 **DataTable**을 선택합니다.
- 데이터 테이블에서의 위치 영역에서 **글로벌 시트**를 선택합니다. 이름 드롭다운 메뉴가 변경되어 글로벌 데이터 시트의 매개 변수가 반영됩니다.
- 이름 상자에서 toCity 매개 변수를 선택하고 **확인**을 클릭합니다.
키워드 보기에서 **toCity** 개체에 대한 **값** 셀이 업데이트되어 다음 매개 변수화를 보여 줍니다.



6. 테스트를 저장합니다.

도구 모음에서 **저장**  을 클릭합니다.

매개 변수 및 값을 추가하고 테스트 단계를 이러한 값과 링크했으므로 이제 매개 변수화된 테스트를 실행할 준비가 되었습니다. "[연습 5d: 매개 변수화된 테스트 실행\(83페이지\)](#)"으로 이동하십시오.


연습 5d: 매개 변수화된 테스트 실행

"연습 5b: 데이터 테이블 매개 변수 정의"(77페이지) 및 "연습 5c: 데이터 테이블에 매개 변수 값 추가"(81페이지)에서는 Flight Finder 수행에서 **toCity** 및 **fromCity** 개체에 대한 데이터 테이블 매개 변수를 만들었습니다. 이를 사용하여 상수 개체 값을 테스트 데이터 테이블의 변경되는 값으로 대체할 수 있습니다.

그러나 현재 테스트를 실행한다면 테스트는 글로벌 데이터 시트, 첫 번째 행의 데이터를 사용하여 한 번만 실행됩니다. 매개 변수화의 목적은 다양한 데이터 집합을 사용할 때 응용 프로그램이 어떻게 실행되는지 확인하는 것이므로 테스트를 여러 번 실행하도록 UFT에 지시해야 합니다.

이 단원에서는 전체 테스트가 여러 번 실행되고 테스트 데이터 테이블의 데이터를 사용하도록 UFT 및 테스트를 구성합니다.

1. **UFT를 시작하고 필요한 경우 Book Flights Parameter 테스트를 엽니다.**

- a. "모든 테스트에 대한 솔루션 만들기"(25페이지)의 설명대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- b. 열기 아래쪽 화살표  열기 를 클릭하고 솔루션 열기를 선택합니다. 솔루션 열기 대화 상자가 열립니다.
- c. C:\%HOMEPATH%\My Documents\Unified Functional Testing에 있는 Flight Reservation Application.ftsln 파일로 이동하고 열기를 클릭합니다.
"연습 5a: 매개 변수화에 대한 테스트 만들기"(76페이지)에서 만든 Book Flights Parameter 테스트가 포함된 항공편 예약 솔루션이 열립니다.
- d. 솔루션 탐색기에서 Book Flights 테스트 노드를 클릭합니다. 테스트 흐름 캔버스가 문서 창에서 개별 탭으로 열립니다.

2. **항공편 예약 응용 프로그램이 자동으로 열리지 않도록 기록 및 실행 설정을 변경합니다.**

Book Flights 테스트(이 테스트를 만드는 기준으로 사용된 테스트)에서 테스트 실행 시작 시 항공편 예약 응용 프로그램을 자동으로 열도록 실행 및 기록 설정을 구성했습니다. 이 테스트 실행에서는 UFT가 응용 프로그램을 테스트 단계 일부로 열도록 합니다.

- a. 기록 > 기록 및 실행 설정을 선택합니다. 기록 및 실행 설정 대화 상자가 열립니다.
- b. 기록 및 실행 설정 대화 상자의 Windows 응용 프로그램 탭에서 열려 있는 임의의 Windows 응용 프로그램에서 테스트 기록 및 실행 옵션을 선택하고 적용을 클릭합니다.
- c. 확인을 클릭하여 대화 상자를 닫습니다.


3. **응용 프로그램 열기 및 닫기에 대한 수행을 추가합니다.**

매개 변수화된 테스트를 실행하면 UFT에서는 데이터 테이블의 행 수에 따라 전체 테스트를 여러 번 실행합니다. 그러나 이 작업을 수행하려면 UFT가 각각의 4개 응용 프로그램 페이지(Login, Flight Finder, Select Flight 및 Flight Details/Confirmation)에 대한 수행을 실행하도록 응용 프로그램을 열고 닫는 단계를 추가해야 합니다.

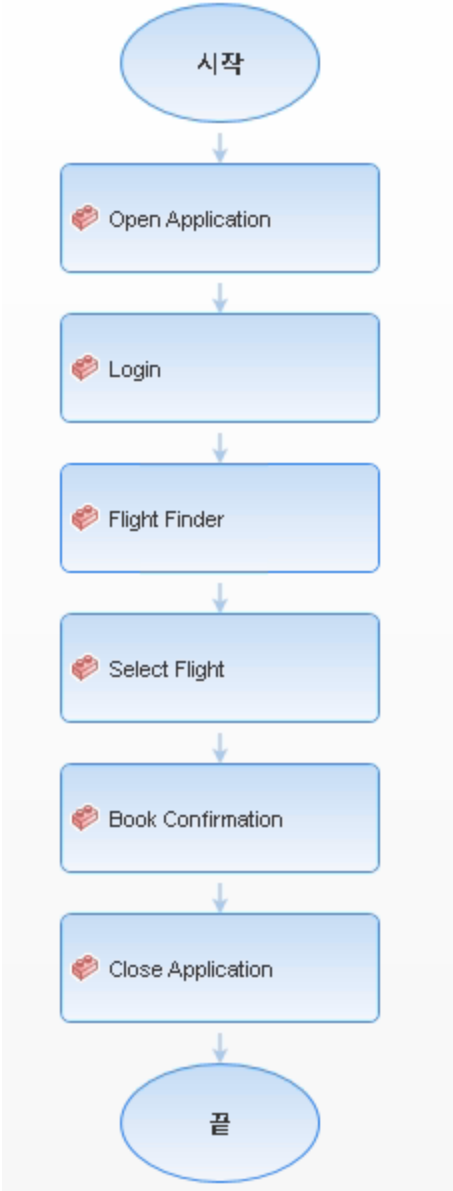
- a. 문서 창에서 테스트 흐름 캔버스가 포함된 Book Flights 탭을 선택합니다.
- b. 도구 모음에서 새 수행 호출 삽입 버튼  을 클릭합니다. 새 수행 호출 삽입 대화 상자가 열립니다.
- c. 새 수행 호출 삽입 대화 상자에서 새 수행 이름을 Open Application으로 지정합니다. 다른 모든 설정 및 옵션을 기본값으로 유지합니다.
이름이 Open Application인 새 수행 블록이 테스트 흐름 끝에 추가됩니다.
- d. 테스트 흐름 캔버스가 포함된 Book Flights 탭에서 Open Application 수행을 마우스 오른쪽 버튼으로 클릭하고 위로 이동을 선택합니다. Open Application 수행 블록이 Flight Confirmation 수행 위로 이동합니다.

- e. 마우스 오른쪽 버튼을 클릭하고 Open Application 블록이 테스트의 첫 번째 수행이 될 때까지 위로 이동을 선택합니다.

참고: 필요에 따라 테스트 흐름에서 수행 블록을 끌어서 놓을 수도 있습니다.

- f. 도구 모음에서 새 수행 호출 삽입 버튼  을 다시 클릭합니다.
- g. 새 수행 호출 삽입 대화 상자에서 새 수행 이름을 Close Application으로 지정합니다. 다른 모든 설정 및 옵션을 기본값으로 유지합니다.

두 가지 수행을 삽입하면 테스트 흐름이 다음과 같이 표시됩니다.



4. 응용 프로그램을 열고 닫는 문을 추가합니다.

Book Flights 테스트를 만들 때 테스트에 대한 실행 설정을 사용하여 응용 프로그램을 자동으로 열도록 UFT에 지시했습니다. 이 테스트에서는 응용 프로그램 열기 및 닫기를 개별 단계로 추가해야 합니다. 이 작업을 수행하기 위해서는 **SystemUtil** 문을 사용합니다.

- a. 솔루션 탐색기에서 **Open Application** 수행 노드를 두 번 클릭합니다. Open Application 수행이 문서 창에서 개별 탭으로 열립니다.
- b. **보기 > 편집기**를 선택하여 편집기를 엽니다.
- c. 편집기에서 다음 줄을 붙여넣습니다.

```
SystemUtil.Run "C:\Program Files (x86)\HP\Unified Functional Testing\samples\Flights Application\FlightsGUI.exe"
```

참고: 자습서의 PDF 복사본에서 이 줄을 붙여넣을 경우 하나의 줄에서 이 메서드만 사용하도록 붙여넣은 테스트를 편집해야 합니다.

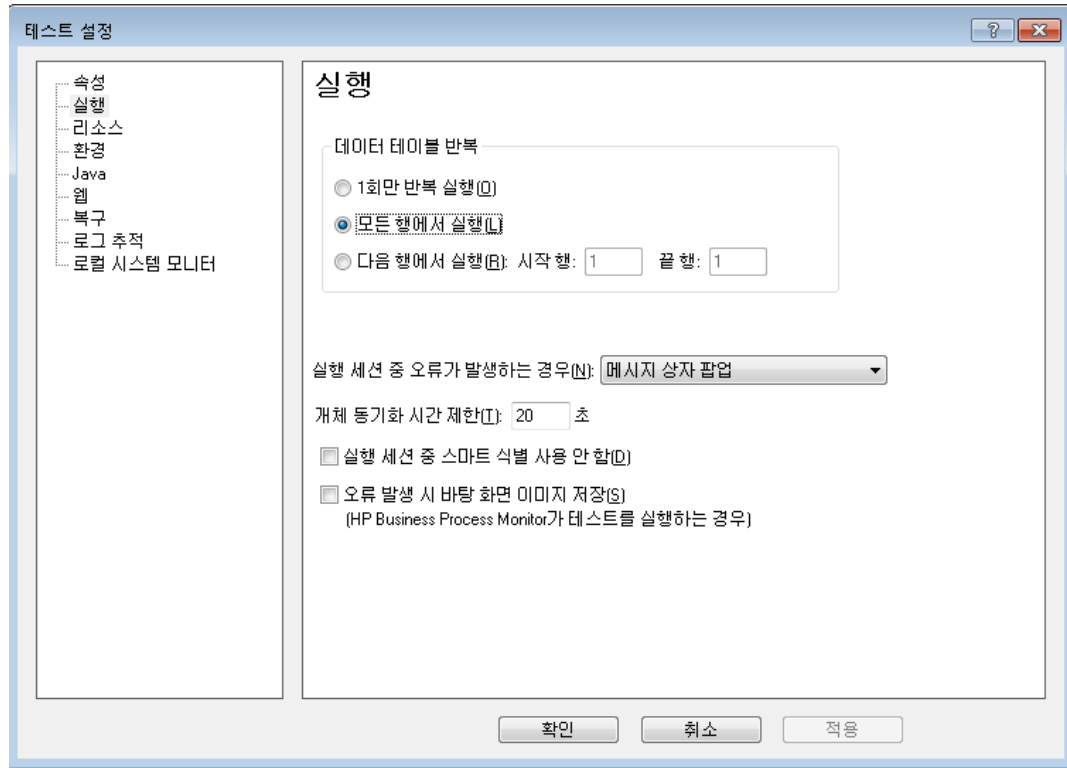
- d. 솔루션 탐색기에서 **Close Application** 수행 노드를 두 번 클릭합니다. Close Application 수행도 문서 창에서 개별 탭으로 열립니다.
- e. 편집기에서 다음 줄을 붙여넣습니다.

```
SystemUtil.CloseDescendentProcesses
```

5. 데이터 테이블의 각 행에 대해 반복을 실행하도록 UFT에 지시합니다.


- a. **파일 > 설정**을 선택합니다. 설정 대화 상자가 열립니다.
- b. 설정 대화 상자에서 **실행** 노드를 선택합니다.

- c. **데이터 테이블 반복** 섹션에서 **모든 행 실행** 옵션을 선택합니다. 이렇게 하면 UFT가 글로벌 데이터 시트의 각 행에 대해 테스트 반복을 실행합니다.



이제 테스트를 실행하면 UFT에서는 글로벌 데이터 시트의 4개 행에 해당하는 여러 테스트 반복을 실행합니다.

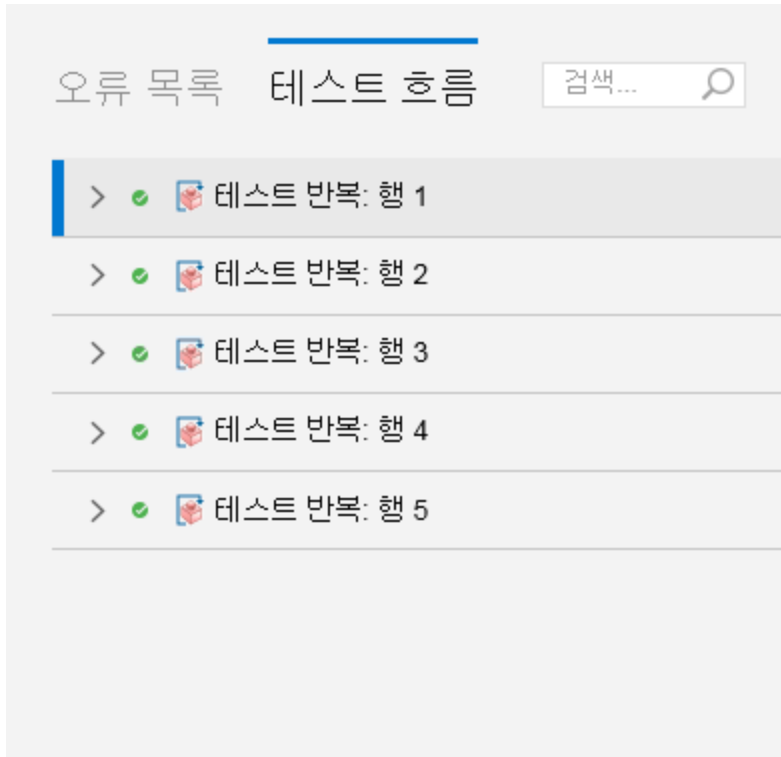
6. Book Flights Parameter 테스트를 실행합니다.

- a. 실행 버튼  을 클릭합니다. 실행 대화 상자가 열립니다.
- b. 실행 대화 상자의 **결과 위치** 탭에서 **새 실행 결과 폴더**를 선택하고 기본 폴더 이름을 그대로 사용합니다.
- c. **확인**을 클릭합니다. 테스트 실행이 완료되면 실행 결과가 열립니다.

7. 실행 결과를 분석합니다.

Run Results Viewer에서 결과 트리의 위쪽 노드를 마우스 오른쪽 버튼으로 클릭하고 **모두 확장**을 선택합니다.

결과에는 테스트의 각 반복에 대한 5개 노드가 표시됩니다. 이는 글로벌 데이터 테이블의 여러 행에 해당합니다.



Flight Finder Summary 노드에서 **fromCity.Select** 또는 **toCity.Select** 단계가 보일 때까지 검색하면 단계에 대한 단계 세부 정보가 데이터 테이블의 값과 일치하도록 수정되었음을 알 수 있습니다.

8. 실행 결과를 닫습니다.

문서 창에서 실행 결과가 포함된 탭을 닫습니다.

데이터를 사용하여 테스트를 매개 변수화하는 방법을 알아보았으므로 이제 ["6단원: 검사점 및 출력 값 만들기"\(89페이지\)](#)로 이동하여 검사점 및 출력 값을 사용하여 테스트를 향상하는 방법을 알아보십시오.

6단원: 검사점 및 출력 값 만들기

"4단원: GUI 테스트 실행 및 분석"(69페이지)에서는 이전 단원에서 만든 테스트를 실행하여 항공편 예약 응용 프로그램에서 수행되는 일련의 단계가 원활하게 실행되는지 확인했습니다.

기본 테스트 단계를 만들고 나서 만들 수 있는 향상된 기능 중 하나는 테스트 및 테스트 단계에 대해 검사점과 출력 값을 추가하는 것입니다. 검사점은 테스트가 실행되는 동안 예상된 정보가 응용 프로그램에 표시되는지 확인합니다. 출력 값은 테스트의 다른 위치에서 매개 변수로 사용할 값을 내보냅니다.

이 단원에서는 검사점을 삽입하고 함수를 사용하여 항공편 예약 응용 프로그램 검사점에서 일부 개체의 유효성을 확인합니다.

이 단원의 내용은 다음과 같습니다.

- 검사점 및 출력 값 유형 이해90
- 연습 6a: Checkpoint 테스트 만들기 91
- 연습 6b: 개체 값 확인 93
- 연습 6c: 테이블 값 확인96
- 연습 6d: 텍스트 값 확인 100
- 연습 6e: 개체 저장소에서 검사점 관리 106
- 연습 6f: 검사점이 있는 테스트 실행 및 분석 108
- 연습 6g: 출력 값 테스트 만들기 111
- 연습 6h: 출력 값 단계 추가 112

검사점 및 출력 값 유형 이해

UFT에서 **검사점**을 삽입하여 응용 프로그램이 제대로 실행되는지 확인할 수 있습니다. 이러한 검사점은 전체 테스트 흐름에서 개별 테스트 단계로 실행됩니다. **출력 값**을 사용하여 특정 단계 또는 개체에서 생성된 값을 가져오고 이 값을 다른 단계에 전달합니다.

검사점

검사점을 사용하여 다양한 응용 프로그램 개체를 확인할 수 있습니다.

개체 유형	검사점 설명	사용 예
표준	개체 속성 값을 검사합니다.	라디오 버튼이 선택되었는지 검사합니다.
이미지	이미지의 속성 값을 검사합니다. 표준 검사점 옵션을 선택하고 이미지 개체를 선택하여 이미지를 검사합니다.	이미지 원본 파일이 올바른지 검사합니다.
테이블	테이블에서 정보를 검사합니다. 표준 검사점 옵션을 선택하고 테이블 개체를 선택하여 테이블을 검사합니다.	테이블 셀의 값이 올바른지 검사합니다.
페이지	웹 페이지의 특성을 검사합니다. 표준 검사점 옵션을 선택하고 브라우저에서 웹 페이지를 선택하여 테이블을 검사합니다.	웹 페이지를 로드하는 데 소요되는 시간이나 웹 페이지에 끊긴 링크가 있는지 검사합니다.
텍스트	텍스트 문자열이 응용 프로그램의 해당 위치에 표시되는지 검사합니다.	테스트 개체의 예상 위치에 예상 텍스트 문자열이 표시되는지 검사합니다.
텍스트 영역	텍스트 문자열이 Windows 기반 응용 프로그램의 정의된 영역 내에 표시되는지 검사합니다.	대화 상자의 영역에 응용 프로그램의 다른 부분에 입력된 텍스트가 포함되는지 검사합니다.
비트맵	응용 프로그램을 비트맵으로 캡처하고 나서 응용 프로그램의 영역을 검사합니다.	웹 페이지(또는 웹 페이지의 부분)가 예상대로 표시되는지 검사합니다.
데이터베이스	응용 프로그램이나 웹 사이트에서 액세스하는 데이터베이스의 콘텐츠를 검사합니다.	데이터베이스 쿼리의 값이 올바른지 검사합니다.
액세스 가능성	웹 사이트 영역을 식별하여 색션 508 준수를 검사합니다.	웹 페이지의 이미지에 W3C 웹 콘텐츠 액세스 가능성 지침에서 요구하는 ALT 속성이 포함되는지 검사합니다.
파일 콘텐츠	실행 세션 중에 생성되거나 액세스된 문서의 텍스트를 검사합니다.	동적으로 생성된 PDF의 머리글에 지역 본사 연락처 정보가 올바르게 표시되는지 검사합니다.
XML	XML 문서의 데이터 콘텐츠를 검사합니다.	요소의 내용을 검사하여 태그, 특성 및 값이 변경되지 않았

		<p>는지 검사합니다.</p> <p>참고: XML 파일 검사점은 지정된 XML 파일을 검사하는 데 사용되고, XML 응용 프로그램 검사점은 웹 페이지 내에서 XML 문서를 검사하는 데 사용됩니다.</p>
--	--	--

출력 값

다음과 같이 다양한 유형의 출력 값을 사용할 수 있습니다.

개체 유형	설명	예
표준	응용 프로그램에서 대부분 개체의 값을 가져오고 저장합니다.	편집 필드의 문자열 출력을 가져옵니다.
파일 콘텐츠	선택한 파일 또는 선택한 파일 일부의 출력을 가져옵니다.	HTML 페이지의 출력을 가져옵니다.
테이블	테이블 개체의 선택한 셀 또는 셀 출력을 가져옵니다.	테이블 개체에서 행 1, 열 1의 셀 출력을 가져옵니다.
텍스트/텍스트 영역	응용 프로그램에서 개체 또는 영역의 텍스트 출력을 가져옵니다.	오류 메시지의 텍스트 출력을 가져옵니다.
데이터베이스	선택한 데이터베이스 셀 또는 데이터베이스 셀의 출력을 가져옵니다.	응용 프로그램에서 개체가 액세스한 데이터베이스의 출력을 가져옵니다.
XML	XML 문서에 포함된 요소의 출력을 가져옵니다.	제품 가격을 정의하는 XML에서 <price> 특성의 출력을 가져옵니다.

대부분 검사점 및 출력 값은 키워드 보기 또는 편집기의 편집 단계나 기록 중에 추가할 수 있습니다. 다음 연습에서는 위에 설명된 일부 검사점을 만드는 방법을 설명합니다.

UFT에서는 검사점 또는 출력 값을 만들 때 검사점 또는 출력 값 내부 정보에 따라 이름을 지정합니다(예: 선택된 값). 검사점 또는 출력 값 이름은 이름 지정 기준으로 사용된 정보를 수정하더라도 변경되지 않습니다. 키워드 보기에 표시된 검사점 또는 출력 값을 찾을 때 이를 유의하십시오. 또한 UFT에서는 키워드 보기에 표시된 이름을 짧게 만들 수 있음에 유의하십시오.

검사점 및 출력 값에 대한 자세한 내용은 *HP Unified Functional Testing User Guide*를 참조하십시오.

검사점을 사용할 테스트를 만들려면 "[연습 6a: Checkpoint 테스트 만들기](#)"(91페이지)로 이동하십시오.


출력 값을 추가하려면 "[연습 6g: 출력 값 테스트 만들기](#)"(111페이지)를 계속 진행하십시오.

연습 6a: Checkpoint 테스트 만들기

이 연습에서는 "[4단원: GUI 테스트 실행 및 분석](#)"(69페이지)에서 새 테스트로 실행한 Book Flights 테스트를 저장하여 검사점을 만듭니다.

참고: 검사점을 개별 테스트로 관리할 필요는 없습니다. 여기서는 이 자습서 학습을 위해 새 테스트를 만드는 것입니다 일반 작업 프로세스 중에 검사점을 테스트에 추가할 수 있습니다.

1. UFT를 시작하고 Book Flights 테스트를 엽니다.

- a. UFT가 현재 열려 있지 않으면 "모든 테스트에 대한 솔루션 만들기"(25페이지)에 설명된 대로 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- b. 열기 버튼 아래쪽 화살표  열기 를 클릭하고 솔루션 열기를 선택합니다. 솔루션 열기 대화 상자가 열립니다.
- c. C:\%HOMEPATH%\My Documents\Unified Functional Testing에 있는 Flight Reservation Application.ftsln 파일로 이동하고 열기를 클릭합니다.
 "1단원: GUI 테스트 및 수행 만들기"(28페이지)에서 만든 Book Flights 테스트가 포함된 Flight Reservation Application 솔루션이 열립니다.

2. 테스트를 Book Flights Checkpoint로 저장합니다.

- a. 솔루션 탐색기에서 Book Flights 테스트 노드를 마우스 오른쪽 버튼으로 클릭하고 다른 이름으로 저장을 선택합니다.
- b. 다른 이름으로 테스트 저장 대화 상자에서 C:\%HOMEPATH%\My Documents\Unified Functional Testing 디렉터리로 이동하고 테스트를 Book Flight Checkpoint로 저장합니다.
 솔루션 탐색기에서 Book Flights 테스트가 새 Book Flights Checkpoint 테스트로 대체됩니다. Book Flights 테스트는 파일 시스템에 별도로 저장되어 있습니다.

3. Book Flights 테스트를 다시 솔루션에 추가합니다.

Book Flights 및 Book Flights Checkpoint 테스트가 같은 솔루션에 포함되어 있으면 두 테스트를 모두 동시에 열 수 있습니다. 이를 통해 테스트를 비교하거나 편집하려는 경우 테스트 사이에서 앞뒤로 전환할 수 있습니다.

참고: 단일 테스트를 한 번만 실행할 수 있습니다.

- a. 파일 > 추가 > 기존 테스트를 선택합니다.
- b. C:\%HOMEPATH%\My Documents\Unified Functional Testing에 있는 Book Flights 테스트로 이동하고 추가를 클릭합니다.
 Book Flights 테스트가 다시 솔루션 탐색기에 추가됩니다. 테스트는 솔루션 탐색기에서 알파벳 순으로 나열됩니다.
 솔루션이 자동으로 저장됩니다.

"연습 6b: 개체 값 확인"(93페이지)으로 이동하여 검사점 삽입을 시작하십시오.

연습 6b: 개체 값 확인

이 연습에서는 "연습 6a: Checkpoint 테스트 만들기"(91페이지)에서 만든 테스트에 표준 검사점을 추가합니다. 이 검사점은 Flight Details 창의 **Passenger Name** 필드에 입력된 값을 확인합니다.

참고: 검사점을 삽입하려면 먼저 항공편 예약 응용 프로그램의 Flight Details 페이지를 열어야 합니다.

1. UFT를 시작하고 Book Flights Checkpoint 테스트를 엽니다.

- 필요한 경우 "모든 테스트에 대한 솔루션 만들기"(25페이지)의 설명대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- 시작 페이지의 **최근에 사용한 솔루션** 영역에서 **Flight Reservation Application**을 클릭합니다. "연습 6a: Checkpoint 테스트 만들기"(91페이지)에서 만든 Book Flights Parameter 테스트를 포함하여 항공편 예약 솔루션이 열립니다.
- 솔루션 탐색기에서 **Book Flights Checkpoint** 노드를 두 번 클릭합니다. Book Flights Checkpoint 테스트가 문서 창에서 개별 탭으로 열립니다.

2. 검사점을 추가할 수행을 표시합니다.

Passenger Name 편집 상자의 속성 값을 확인하는 검사점을 추가하려면 테스트에서 승객 이름을 자동으로 입력한 후 테스트의 적절한 수행에 해당 이름을 추가해야 합니다.


캔버스에서 **Flight Confirmation** 수행을 두 번 클릭하여 엽니다.

3. 항공편 예약 응용 프로그램의 Flight Details 페이지를 엽니다.

- "Flight Reservation Application 탐색"(23페이지)의 설명대로 항공편 예약 응용 프로그램을 엽니다.
- 로그인 정보를 입력합니다.
 - **사용자 이름:** john
 - **비밀번호:** hp
- 확인**을 클릭합니다. Flight Finder 페이지가 열립니다.
- 항공편 검색 세부 정보를 입력합니다.
 - **출발 도시:** Los Angeles
 - **도착 도시:** Sydney
 - **날짜:** 내일 날짜
 - **클래스:** Business
 - **티켓:** 2
- Find Flights** 버튼을 클릭합니다. Select Flight 페이지가 열립니다.

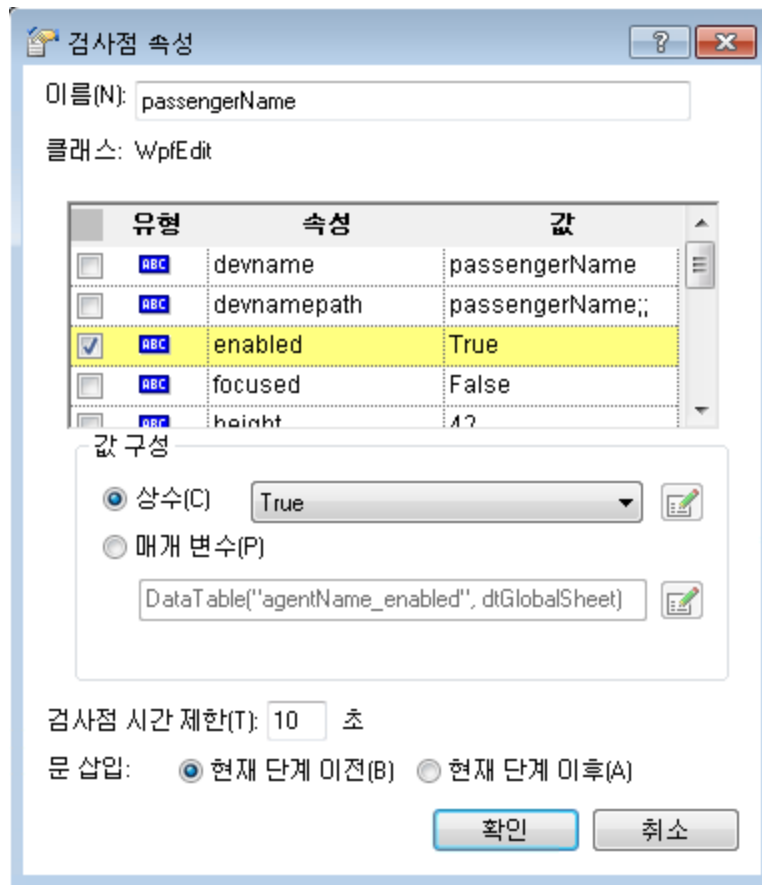
- f. Select Flight 페이지에서 첫 번째 행을 선택하고 **Select Flight**를 클릭합니다. Flight Details 페이지가 열립니다.

4. 표준 검사점을 만듭니다.

- a. 편집기가 표시되면 **키워드 보기 버튼**  을 클릭하여 키워드 보기를 표시합니다.
- b. 키워드 보기에서 표의 오른쪽 여백을 클릭하여 **passengerName** 행을 선택합니다.

참고: 항목 열을 클릭하지 마십시오. 클릭하면 개체만 선택됩니다. 검사점을 추가하려면 전체 단계를 선택해야 합니다.

- c. **설계 > 검사점 > 표준 검사점**을 선택합니다. 검사점 속성 대화 상자가 열립니다.



이 대화 상자에 passengerName 개체 속성이 표시됩니다.

- **이름**은 응용 프로그램에 정의된 개체의 이름입니다. 이 경우 이름은 **passengerName**입니다.
- **클래스**는 개체 유형입니다. 이 경우 유형은 **WpfEdit**으로, 개체 유형이 편집 상자임을 나타냅니다.

- 유형 열의 **ABC** 아이콘은 속성 값이 상수임을 나타냅니다.

검사점을 삽입할 때 UFT에서는 각 개체 클래스의 기본 속성 확인을 권장합니다.

속성	값	설명
enabled	True	개체가 현재 사용하도록 설정되어 있는지 여부를 확인합니다.
isreadonly	False	정보를 편집 상자에 입력할 수 있는지 여부를 확인합니다. 현재 개체는 텍스트 문자열 입력을 허용하도록 설정되어 있습니다.
text	기본값이 지정되지 않음	개체에 입력된 텍스트를 확인합니다. 현재 값이 비어 있습니다. 이 수행의 첫 번째 단계에서 passengerName 편집 상자에 대해 지정한 동일한 값을 입력해야 합니다.

- 검사점 속성 대화 상자의 **이름** 상자에서 새 검사점 이름으로 CheckName을 입력합니다.
- 개체 속성 영역에서 아래로 스크롤하고 속성 이름 **text**가 포함된 행을 선택합니다. 행이 노란색으로 바뀌어 이 행을 선택했음을 보여 줍니다.
- text** 속성 행에서 **값** 열을 클릭합니다.
- 개체 속성 표 아래 **값** 구성 영역에서 **상수** 라디오 버튼을 클릭합니다.
- 상수** 값 편집 상자에서 John Smith를 입력합니다. (이는 수행의 첫 번째 단계에서 Passenger Name 상자에 입력한 값의 이름입니다.) 개체 속성 표가 이 값으로 업데이트되는지 확인합니다.
- 검사점 속성 대화 상자 아래쪽의 문 삽입 영역에서 **현재 단계 뒤**를 선택합니다. 그러면 **passengerName Set** 단계 뒤에 검사점이 삽입됩니다.
- 나머지 설정을 기본값으로 수락하고 **확인**을 클릭합니다.

UFT에서 표준 검사점 단계를 선택한 단계 아래 테스트에 추가합니다.

항목	작업	값	설명
HP MyFlight Sample Application			
passengerName	Set	"John Sminth"	"John Sminth"을(를) "passengerName" edit box에 입...
passengerName	Check	Checkpoint("CheckName")	"passengerName" edit box이(가) 선택한 속성에 대...
ORDER	Click		"ORDER" button을(를) 클릭합니다.
progBar	WaitProp...	value,100	다음 단계를 계속하기 전에 "progBar" progress bar...
Order 94 completed	Check	Checkpoint("CheckOrderComplete")	"Order 94 completed" object의 텍스트가 예상 텍스...
NEW SEARCH	Click		"NEW SEARCH" button을(를) 클릭합니다.
+ 새 단계			

5. 테스트를 저장합니다.


도구 모음에서 **저장**  을 클릭합니다.

이 프로세스를 사용하여 다양한 검사점 유형을 삽입할 수 있습니다. **"연습 6c: 테이블 값 확인"(96페이지)**으로 이동하여 응용 프로그램에서 테이블 개체를 확인하는 방법을 알아보십시오.

연습 6c: 테이블 값 확인

"연습 6b: 개체 값 확인"(93페이지)에서는 응용 프로그램에서 개체에 대한 검사점을 추가했습니다. 이 연습에서는 테스트에 테이블 검사점을 추가합니다. 테이블 검사점은 Select Flights 페이지의 항공편 표에서 값을 확인합니다.

1. UFT를 시작하고 Book Flights Checkpoint 테스트를 엽니다.

- 필요한 경우 "모든 테스트에 대한 솔루션 만들기"(25페이지)의 설명대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- 열기 버튼 아래쪽 화살표  열기 를 클릭하고 솔루션 열기를 선택합니다. 솔루션 열기 대화 상자가 열립니다.
- 솔루션 열기 대화 상자에서 C:\%HOMEPATH%\My Documents\Unified Functional Testing 에 있는 **Flight Reservation Application.ftsln** 파일로 이동하고 열기를 클릭합니다.
"연습 6a: Checkpoint 테스트 만들기"(91페이지)에서 만든 Book Flights Parameter 테스트가 포함된 Flight Reservation Application 솔루션이 열립니다.
- 솔루션 탐색기에서 **Book Flights Checkpoint** 노드를 두 번 클릭합니다.

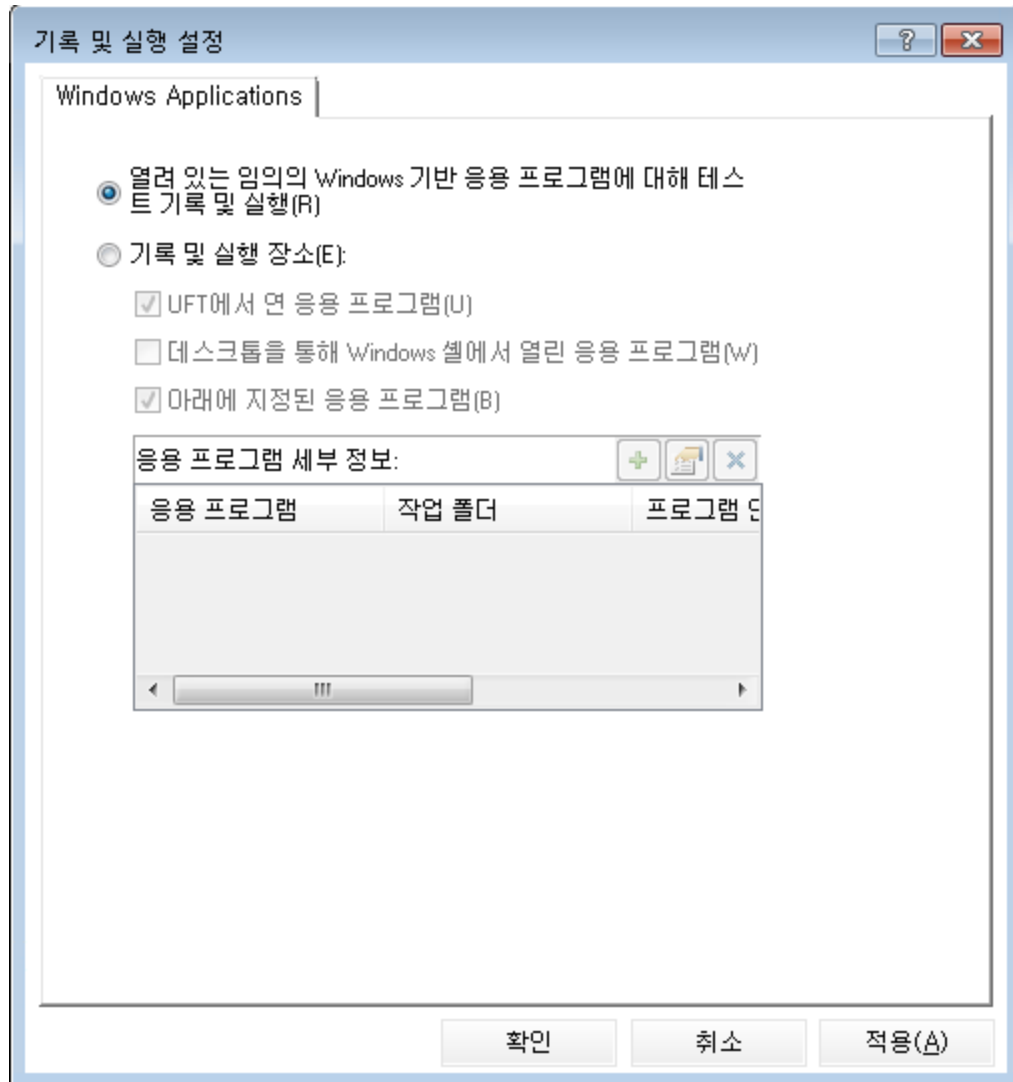
2. 테이블 검사점을 추가할 단계를 찾습니다.


- Select Flight 수행이 열려 있지 않으면 솔루션 탐색기에서 **Select Flight** 수행 노드를 두 번 클릭합니다. 수행이 문서 창에서 개별 탭으로 표시됩니다.
- 편집기가 표시되면 보기 > 키워드 보기를 선택하여 키워드 보기를 표시합니다.
- 키워드 보기에서 **flightsDataGrid** 단계(예약할 항공편을 선택하는 단계)를 선택합니다.


3. 항공편 예약 응용 프로그램의 Select Flight 페이지를 엽니다.

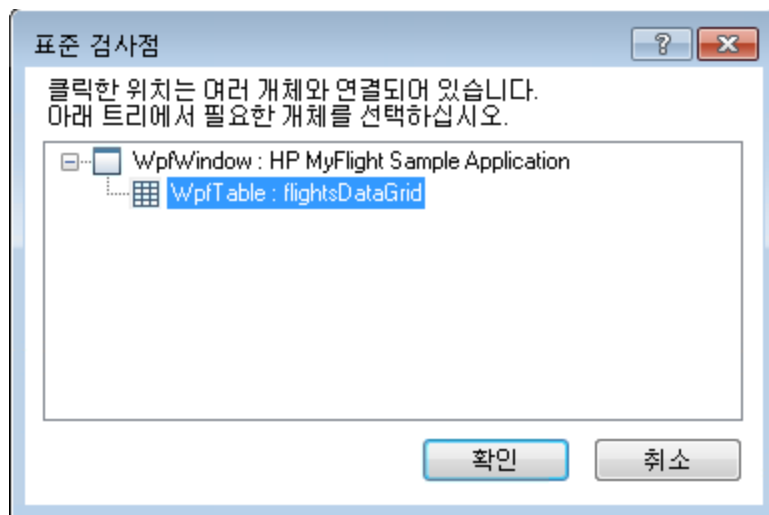
- "Flight Reservation Application 탐색"(23페이지)의 설명대로 항공편 예약 응용 프로그램을 엽니다.
- 로그인 정보를 입력합니다.
 - 사용자 이름: john
 - 비밀번호: hp
- 확인을 클릭합니다. Flight Finder 페이지가 열립니다.
- 항공편 검색 세부 정보를 입력합니다.
 - 출발 도시: Los Angeles
 - 도착 도시: Sydney
 - 날짜: 내일 날짜
 - 클래스: Business
 - 티켓: 2

- e. **Find Flights** 버튼을 클릭합니다. Select Flight 페이지가 열립니다.
4. **열린 응용 프로그램 페이지에서 기록하도록 UFT를 구성합니다.**
- a. UFT에서 기록 > 기록 및 실행 설정을 선택합니다. 기록 및 실행 설정 대화 상자가 열립니다.

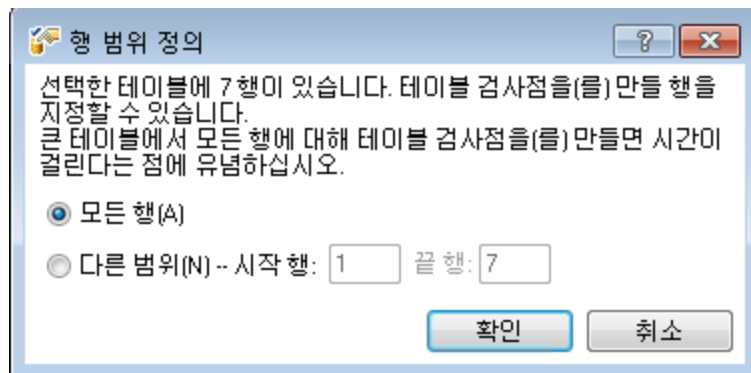


- b. Windows 응용 프로그램 탭에서 **열려 있는 임의의 Windows 기반 응용 프로그램에 대해 테스트 기록 및 실행** 옵션을 선택합니다.
 - c. **확인**을 클릭하여 대화 상자를 닫습니다.
5. **테이블 검사점을 만듭니다.**
- a. 도구 모음에서 기록 버튼  을 클릭합니다. UFT에서 기록 세션을 시작하고 기본 UFT 창이 숨겨집니다.

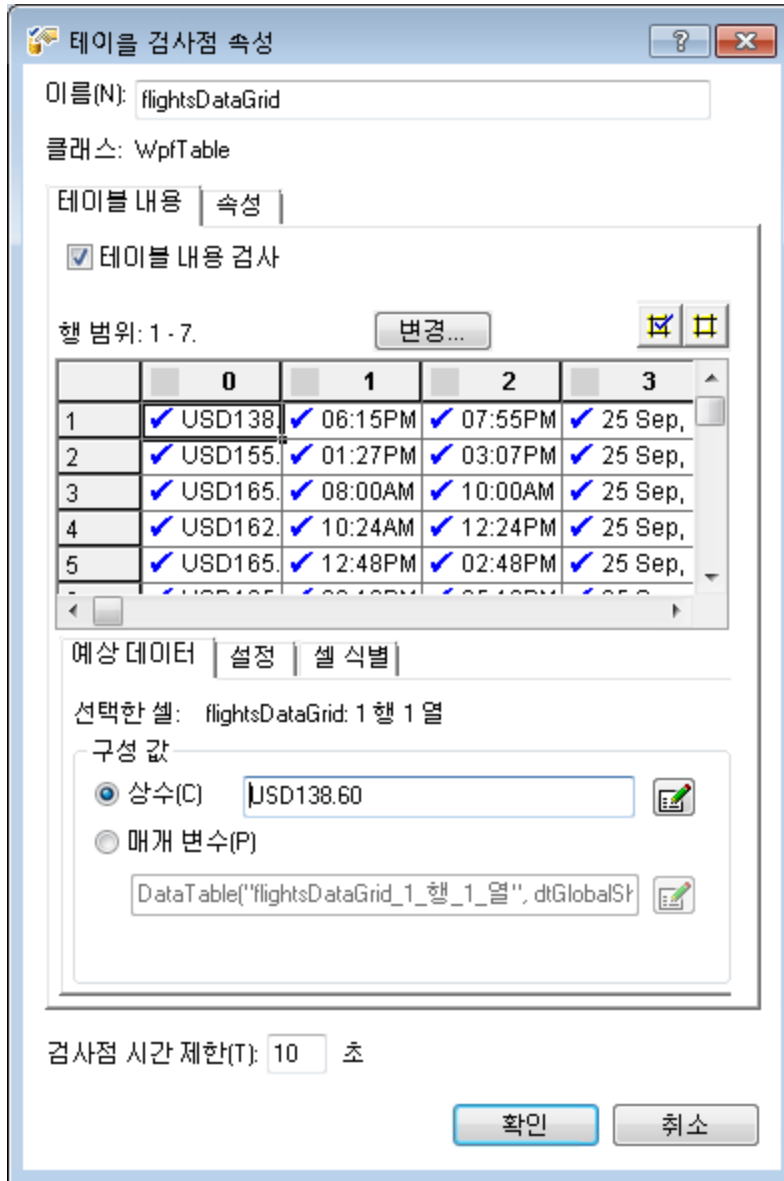
- b. 기록 도구 모음에서 **검사점 또는 출력 값 삽입** 드롭다운 화살표  를 클릭하고 **표준 검사점** 을 선택합니다. 마우스 포인터가 가리키는 손 모양으로 바뀝니다.
- c. 항공편 예약 응용 프로그램에서 항공편 테이블을 클릭합니다. 표준 검사점 개체 선택 대화 상자가 열립니다.



- d. 표준 검사점 선택 대화 상자에서 **WpfTable: flightsDataGrid** 개체를 선택하고 **확인**을 클릭합니다. 행 범위 정의 대화 상자가 열립니다.



- e. 행 범위 정의 대화 상자에서 **모든 행** 라디오 버튼을 선택하고 **확인**을 클릭합니다. 테이블 검사점 속성 대화 상자가 열립니다.



기본적으로 확인 표시가 모든 셀에 나타납니다. 셀을 두 번 클릭하여 셀을 선택 또는 선택 취소 하거나, 행 또는 열 머리글을 두 번 클릭하여 해당 행이나 열의 모든 셀을 선택 또는 선택 취소할 수 있습니다.

- f. 테이블 검사점 속성 대화 상자의 **이름** 상자에 새 검사점 이름으로 CheckCost를 입력합니다.
- g. 표에서 각 열 머리글을 두 번 클릭하여 확인 표시를 지웁니다.

참고: 모든 테이블 개체 열을 보려면 표에서 오른쪽으로 스크롤해야 합니다.

- h. 표에서 행 1, 열 0을 두 번 클릭하여 이 셀을 선택합니다. (UFT에서는 확인 표시가 있는 셀만 확인합니다.)

	0	1	2	3
1	✓ USD138.	06:15PM	07:55PM	25 Sep,
2	USD155.	01:27PM	03:07PM	25 Sep,
3	USD165.	08:00AM	10:00AM	25 Sep,
4	USD162.	10:24AM	12:24PM	25 Sep,
5	USD165.	12:48PM	02:48PM	25 Sep,

참고: 테이블에 표시된 데이터는 날짜를 구분합니다. 한 날짜에 대해 이 검사점을 만들었는데 돌아가서 다른 날짜에 대해 이 테스트를 실행할 경우 검사점을 통과하게 하려면 위 단계를 사용하여 이 검사점을 업데이트해야 합니다.

- i. 행과 열을 스크롤하여 행 1, 열 0의 셀만 선택되었는지 확인합니다. 선택된 다른 셀이 있으면 두 번 클릭하여 선택을 제거합니다.
- j. 나머지 설정을 기본값으로 수락하고 **확인**을 클릭합니다.

6. 기록 세션을 중지합니다.

기록 도구 모음에서 **중지**  를 클릭합니다.

테이블 개체의 검사점 속성을 정의한 후 UFT에서 테이블 검사점 단계를 테스트에 추가했습니다. 테이블 검사점 단계는 키워드 보기에서 flightsDataGrid 개체 단계 아래에 새 단계로 표시됩니다.

항목	작업	값	설명
HP MyFlight Sample Application			
flightsDataGrid	SelectCell	"1","1"	"flightsDataGrid" table의 행 "1", 열 "1"의 셀을 선택합니다.
flightsDataGrid	Check	CheckPoint("CheckCost")	Check whether the content of specified cells in the "flightsDataG...
SELECT FLIGHT	Click		"SELECT FLIGHT" button을(를) 클릭합니다.
+ 새 단계			

7. 테스트를 저장합니다.

도구 모음에서 **저장**  을 클릭합니다.

테이블 개체에 대한 검사점을 추가했으므로 이제 계속하여 **"연습 6d: 텍스트 값 확인"(100페이지)**에서 검사점을 추가하십시오.


연습 6d: 텍스트 값 확인

이전 연습에서는 일반 테스트 개체 및 테이블 개체에 검사점을 추가했습니다. 이 개체에서는 텍스트 검사점을 테스트에 추가하여 개체 내부에서 주문 프로세스 종료 시 나타나는 텍스트를 확인합니다.

1. **UFT를 시작하고 Book Flights Checkpoint 테스트를 엽니다.**

- a. 필요한 경우 "[모든 테스트에 대한 솔루션 만들기](#)"(25페이지)의 설명대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- b. 시작 페이지의 최근에 사용한 솔루션 영역에서 **Flight Reservation Application** 솔루션을 클릭합니다.
 "[연습 6a: Checkpoint 테스트 만들기](#)"(91페이지)에서 만든 Book Flights Parameter 테스트가 포함된 Flight Reservation Application 솔루션이 열립니다.
- c. 솔루션 탐색기에서 **Book Flights Checkpoint** 노드를 두 번 클릭합니다.
 Book Flights Checkpoint 테스트가 문서 창에서 개별 탭으로 열립니다.

2. **텍스트 검사점을 추가할 단계를 찾습니다.**

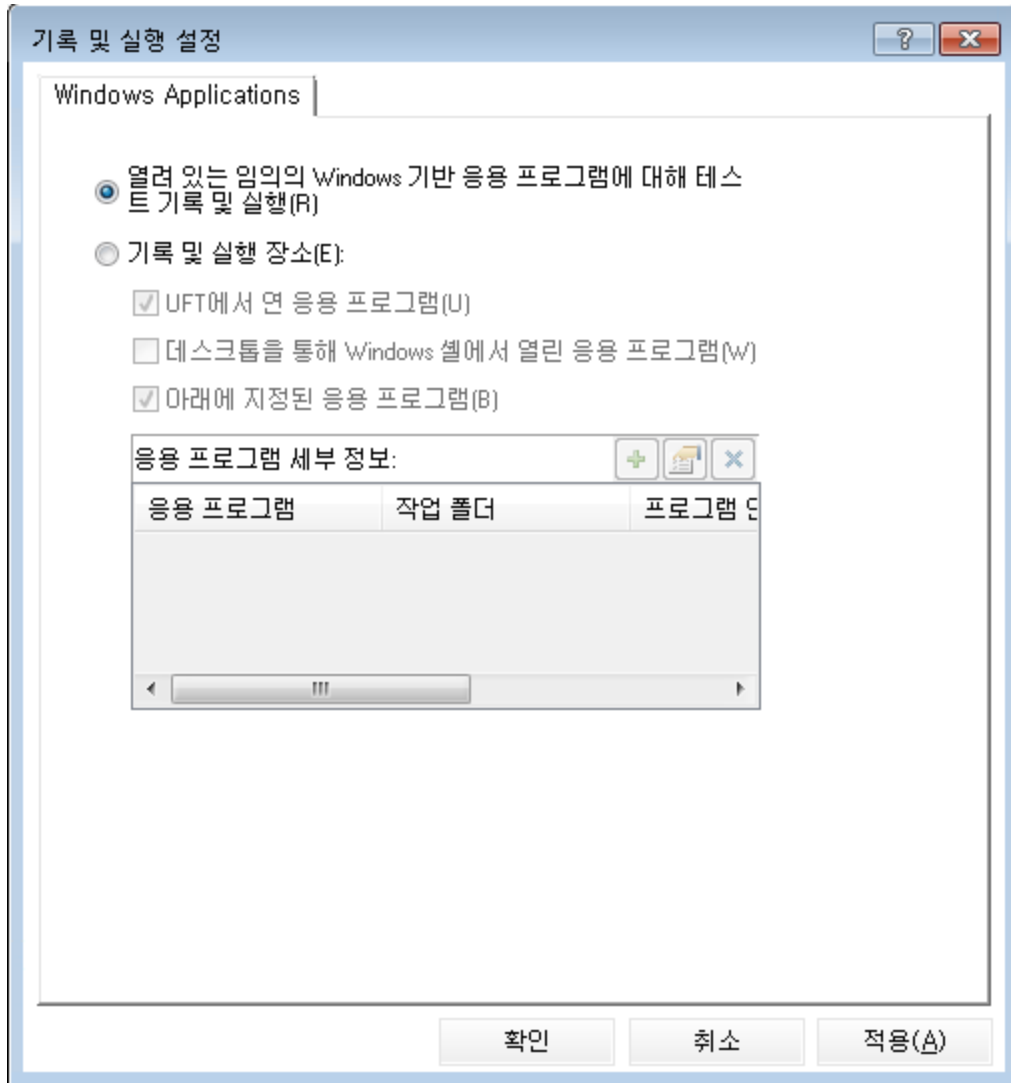
- a. 솔루션 탐색기에서 **Flight Confirmation** 수행 노드를 두 번 클릭합니다. Flight Confirmation 수행이 문서 창에서 개별 탭으로 열립니다.
- b. 편집기가 열리면 **키워드 보기 버튼**  을 클릭하여 키워드 보기를 표시합니다.
- c. 키워드 보기에서 **progBar** 단계를 강조 표시합니다(모든 단계를 완전히 확장한 경우 마지막에서 두 번째 행에서).

3. **항공편 예약 응용 프로그램의 Flight Details 페이지를 엽니다.**

- a. "[Flight Reservation Application 탐색](#)"(23페이지)의 설명대로 항공편 예약 응용 프로그램을 엽니다.
- b. 로그인 정보를 입력합니다.
 - o **사용자 이름:** john
 - o **비밀번호:** hp
- c. **확인**을 클릭합니다. Flight Finder 페이지가 열립니다.
- d. 항공편 검색 세부 정보를 입력합니다.
 - o **출발 도시:** Los Angeles
 - o **도착 도시:** Sydney
 - o **날짜:** 내일 날짜
 - o **클래스:** Business
 - o **티켓:** 2
- e. **Find Flights** 버튼을 클릭합니다. Select Flight 페이지가 열립니다.
- f. Select Flight 페이지에서 첫 번째 행을 선택하고 **Select Flight**를 클릭합니다. Flight Details 페이지가 열립니다.
- g. Flight Details 페이지의 **Passenger Name** 상자에 John Smith를 입력하고 **ORDER**를 클릭합니다.
 창 중간에 주문 완료를 알리는 상자가 표시됩니다. 응용 프로그램을 이처럼 유지합니다.



4. 열린 응용 프로그램 페이지에서 기록하도록 UFT를 구성합니다.

- a. UFT에서 기록 > 기록 및 실행 설정을 선택합니다. 기록 및 실행 설정 대화 상자가 열립니다.
- b. Windows 응용 프로그램 탭에서 열려 있는 임의의 Windows 기반 응용 프로그램에 대해 텍스트 기록 및 실행이 선택되었는지 확인합니다.

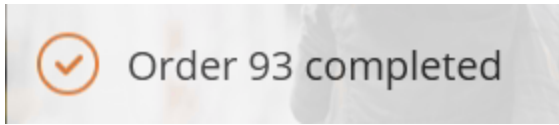


- c. 확인을 클릭하여 대화 상자를 닫습니다.

5. 텍스트 검사점을 만듭니다.

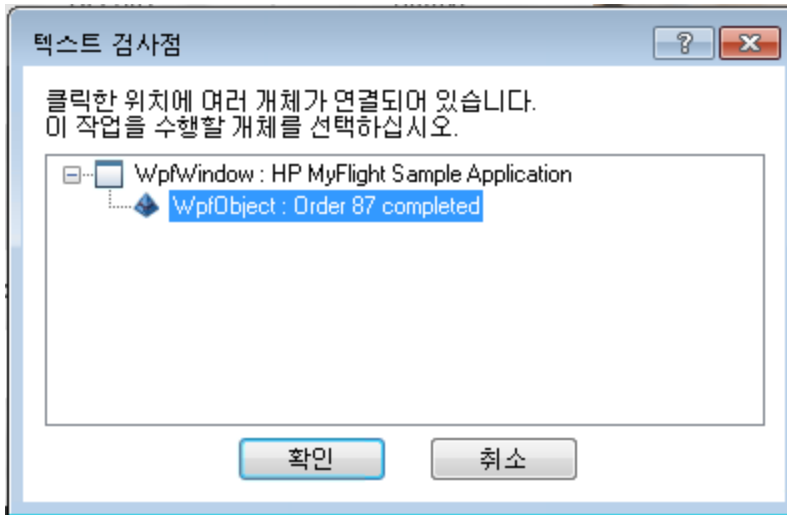
- a. 도구 모음에서 기록 버튼  을 클릭합니다. UFT 창이 숨겨지고 기록 도구 모음이 창 위쪽에서 열립니다.
- b. 기록 도구 모음에서 검사점 또는 출력 값 삽입 버튼  을 클릭하고 텍스트 검사점을 선택합니다. 마우스 포인터가 가리키는 손 모양으로 변경됩니다.

- c. 항공편 예약 응용 프로그램의 Flight Details 창에서 Flight Details 창의 중간에 있는 Order # Completed 그래픽을 클릭합니다.

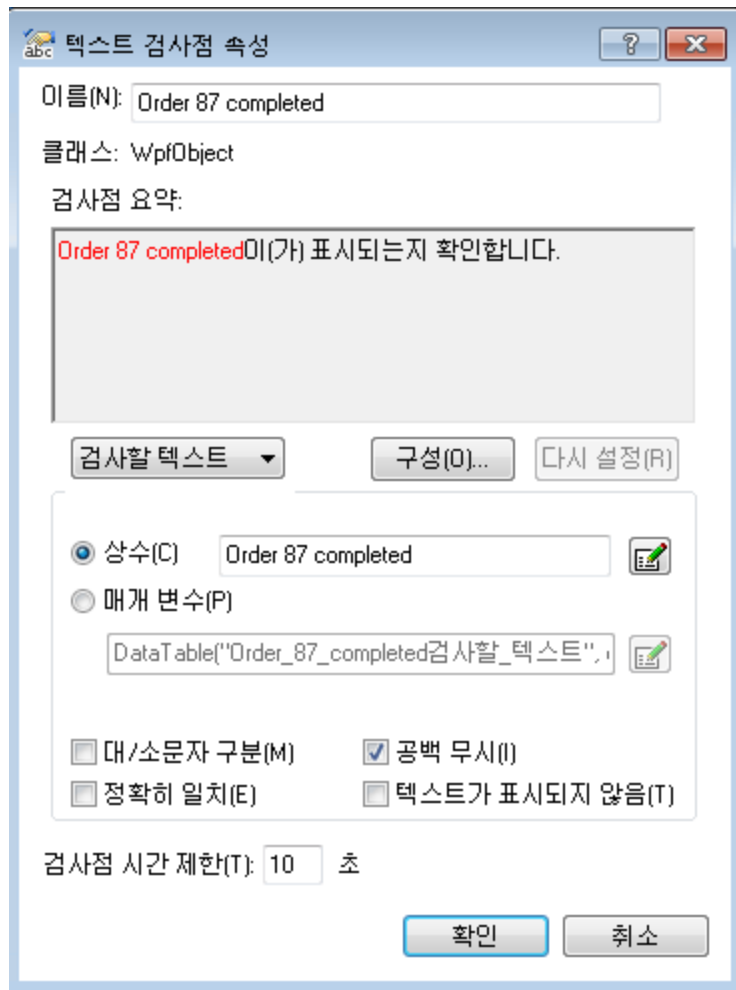


참고: 이전에 항공편 예약 응용 프로그램을 실행했는지 여부에 따라 주문 번호가 다를 수 있습니다.

텍스트 검사점 개체 선택 대화 상자가 열립니다.

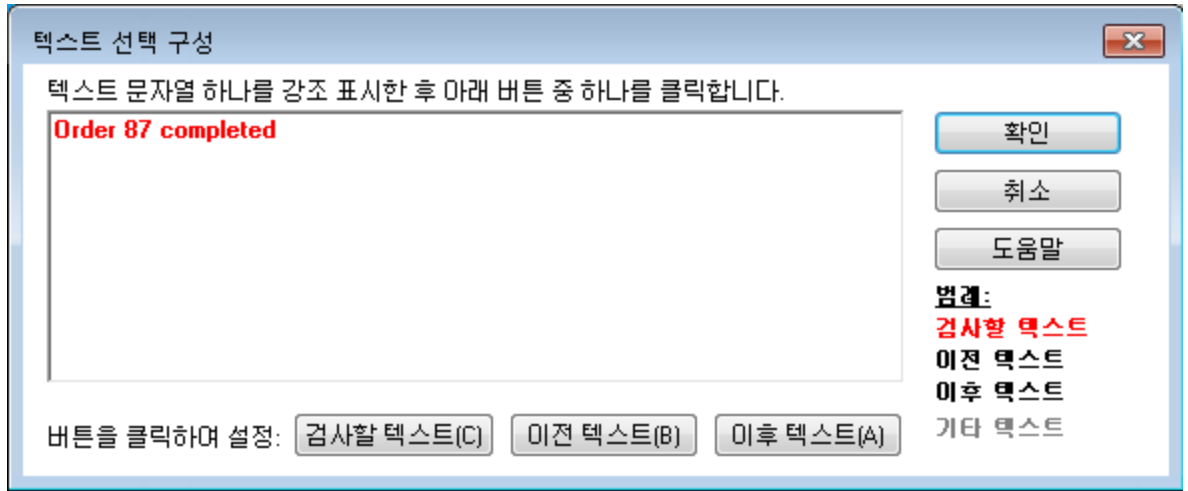


- d. 텍스트 검사점 개체 선택 대화 상자에서 **WpfObject: Order # Completed** 개체를 선택하고 **확인** 을 클릭합니다. 텍스트 검사점 속성 대화 상자가 열립니다.

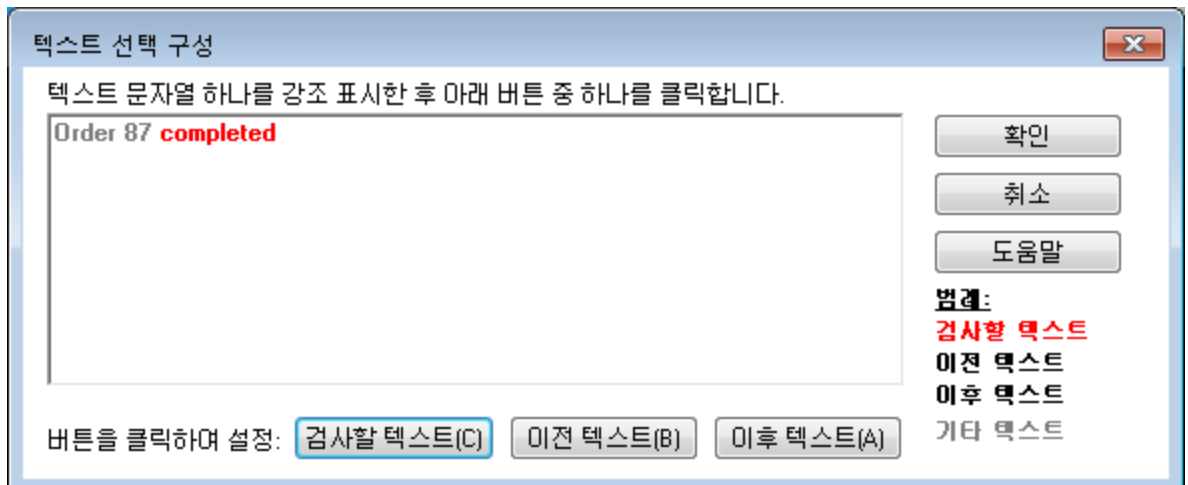


- e. 텍스트 검사점 속성 대화 상자의 **이름** 상자에서 새 검사점 이름으로 **CheckOrderCompletedText**를 입력합니다.

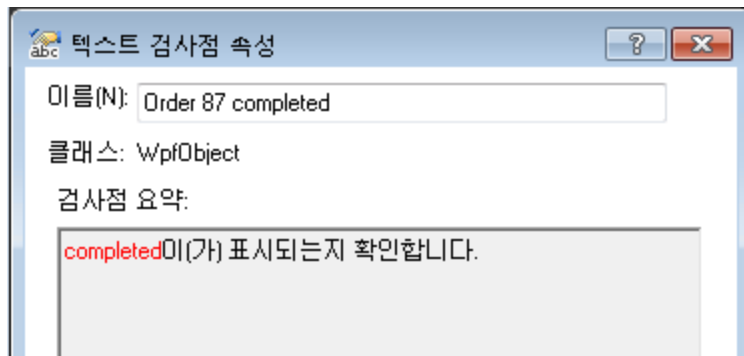
- f. **검사점 요약 영역에서 구성 버튼을 클릭합니다.** 텍스트 선택 구성 대화 상자가 열립니다.



- g. 텍스트 선택 구성 대화 상자에서 **completed** 문자열을 강조 표시하고 **검사할 텍스트**를 클릭합니다. **Order #** 텍스트 문자열이 빨간색에서 회색으로 바뀝니다.



- h. **확인**을 클릭하여 텍스트 선택 구성 대화 상자를 닫습니다.
텍스트 검사점 속성 대화 상자에서 검사점 요약 영역이 업데이트되어 선택 사항이 반영됩니다.

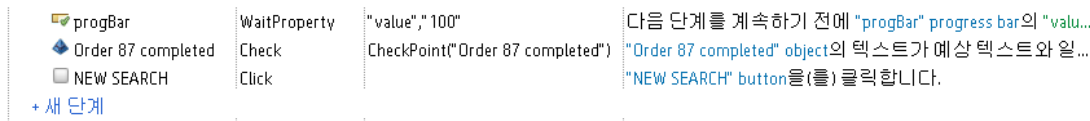


i. 나머지 설정을 기본값으로 수락하고 **확인**을 클릭합니다.

6. 기록 세션을 중지합니다.

기록 도구 모음에서 **중지**  를 클릭하여 기록을 중지합니다.

UFT에서는 텍스트 검사점이 있는 단계를 테스트에서 **progBar** 개체가 포함된 단계 아래에 추가합니다. 이 단계는 키워드 보기에서 **Order # Completed** 개체에 대한 검사점 작업으로 표시됩니다.



편집기에 문이 다음과 같이 표시됩니다.

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

7. 테스트를 저장합니다.

저장  을 클릭합니다.

다양한 검사점을 추가했으므로 이제 "**연습 6e: 개체 저장소에서 검사점 관리**"(106페이지)에서 검사점 관리에 대해 자세히 알아보십시오.

연습 6e: 개체 저장소에서 검사점 관리

이전 연습에서 수행의 다양한 검사점 유형을 추가했습니다. 특정 수행의 컨텍스트에서 검사점을 사용하고 볼 수 있을 뿐만 아니라 개체 저장소에서 검사점을 보고 해당 속성을 수정할 수도 있습니다.

개체 저장소에서 검사점을 수정하면 테스트에 있는 둘 이상의 위치에서 같은 검사점을 사용할 수 있습니다. 예를 들어 조직 로고가 응용 프로그램의 모든 페이지에 나타나는지 확인하려면 검사점을 만들고 테스트의 여러 수행 또는 위치에서 검사점을 삽입할 수 있습니다.


이 연습에서는 검사점을 다시 사용하지 않습니다.

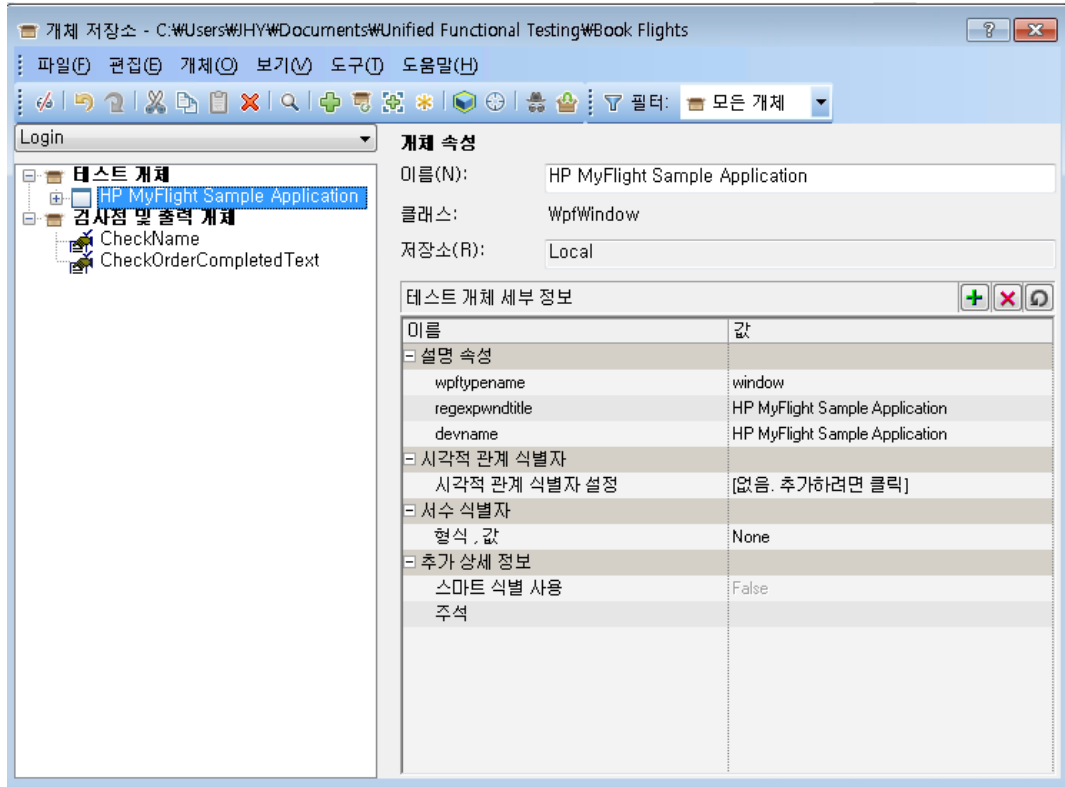
1. UFT를 시작하고 Book Flights Checkpoint 테스트를 엽니다.

- 필요한 경우 "**모든 테스트에 대한 솔루션 만들기**"(25페이지)의 설명대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- 파일 > 열기 > 솔루션**을 선택합니다. 솔루션 열기 대화 상자가 열립니다.
- 솔루션 열기 대화 상자에서 **C:\%HOMEPATH%\My Documents\Unified Functional Testing**에 있는 **Flight Reservation Application** 솔루션으로 이동하고 **열기**를 클릭합니다.

"**연습 6a: Checkpoint 테스트 만들기**"(91페이지)에서 만든 Book Flights Parameter 테스트가 포함된 Flight Reservation Application 솔루션이 열립니다.

2. 개체 저장소 창을 엽니다.

- a. 솔루션 탐색기에서 **Flight Confirmation** 수행을 두 번 클릭합니다. Flight Confirmation 수행이 문서 창에서 개별 탭으로 열립니다.
- b. 도구 모음에서 **개체 저장소 버튼**  을 클릭합니다. 개체 저장소 창이 열리고 현재 수행의 모든 테스트 개체 및 모든 검사점/출력 개체 트리가 표시됩니다.
트리에에는 모든 로컬 개체 및 수행과 연결된 공유 개체 저장소의 모든 개체가 포함됩니다.



3. 검사점을 볼 수행을 선택합니다.

- a. 개체 트리 바로 위의 수행 드롭다운 메뉴에서 테스트 개체, 검사점 개체 및 출력 값 개체를 표시할 수행을 선택합니다.
- b. 작업을 마치면 개체 저장소 창을 닫습니다.

참고: 이 연습에서는 개체 또는 검사점 속성을 수정할 필요가 없습니다.

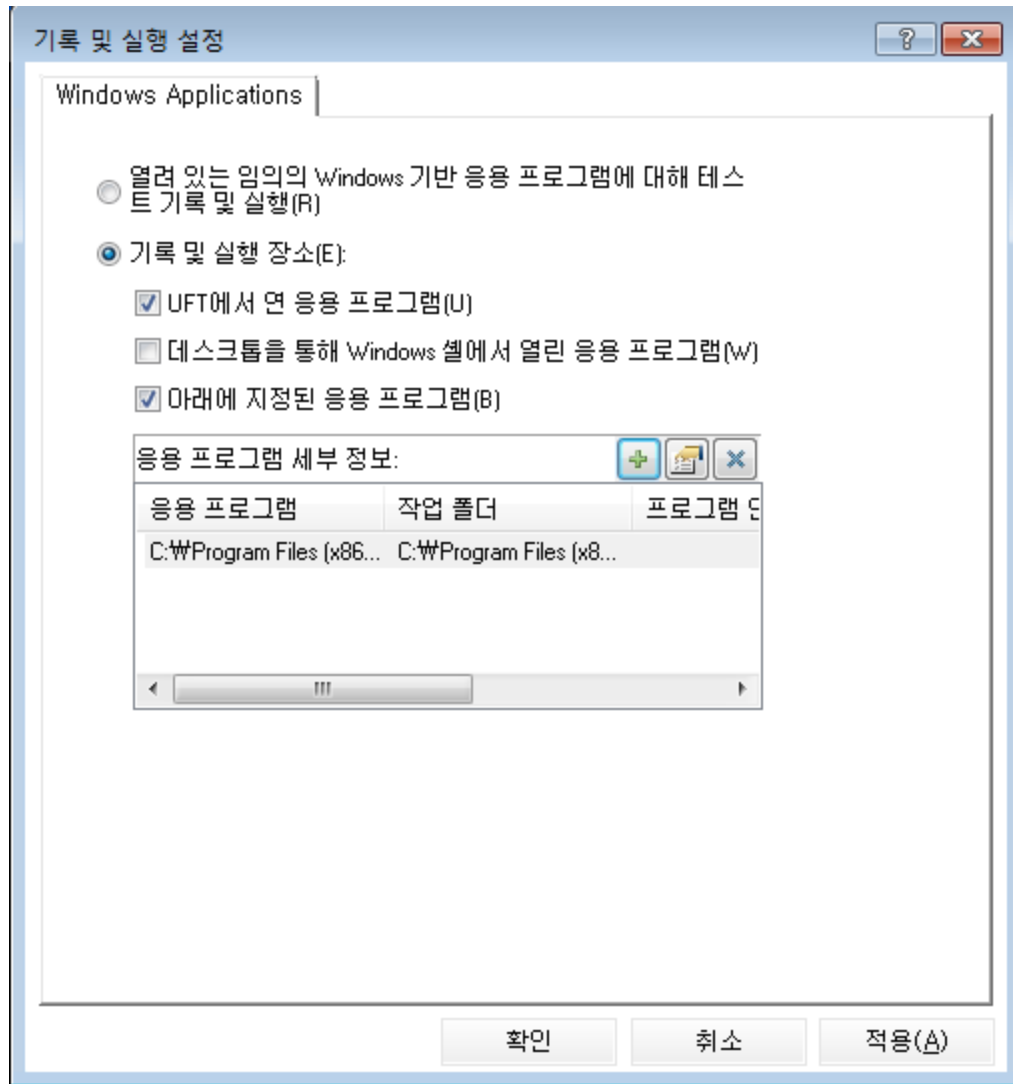
이제 검사점이 있는 테스트를 실행할 준비가 되었습니다. "[연습 6f: 검사점이 있는 테스트 실행 및 분석](#)" (108페이지)으로 이동하여 검사점이 있는 테스트에 대한 테스트를 실행하고 실행 결과를 확인하는 방법을 알아보십시오.

연습 6f: 검사점이 있는 테스트 실행 및 분석

검사점을 사용하여 테스트를 만들었으므로 이제 테스트를 실행하여 검사점 작동 방식을 확인해야 합니다. 이 연습에서는 테스트를 실행하고 검사점 결과를 분석합니다.

1. 항공편 예약 응용 프로그램을 열도록 UFT를 구성합니다.


- a. UFT에서 기록 > 기록 및 실행 설정을 선택합니다. 기록 및 실행 설정 대화 상자가 열립니다.
- b. Windows 응용 프로그램 탭에서 기록 및 실행 장소: 옵션을 선택합니다.



참고: 응용 프로그램 세부 정보가 "4단원: GUI 테스트 실행 및 분석"(69페이지)에서 설정한 대로 저장되어야 합니다.

- c. 확인을 클릭하여 대화 상자를 닫습니다.

2. 테스트 실행을 시작합니다.

- a. 도구 모음에서 실행 버튼  을 클릭합니다. 실행 대화 상자가 열립니다.
- b. 결과 위치 탭에서 새 실행 결과 폴더가 선택되었는지 확인합니다. 기본 결과 폴더 이름을 그대로 사용합니다.
- c. 확인을 클릭합니다.
UFT에서는 항공편 예약 응용 프로그램을 열고 단계를 수행합니다. 테스트 실행이 끝나면 실행 결과가 열립니다.

3. 실행 결과를 봅니다.

실행 결과가 표시될 때 실행 결과는 모든 검사점을 통과했음을 나타내는 통과여야 합니다. 검사점의 하나 이상이 실패하면 테스트 실행이 실패로 나열됩니다.

4. 표준 검사점의 결과를 봅니다.

- a. 테스트 흐름에서 **Flight Confirmation** 노드를 찾습니다.
- b. Flight Confirmation 노드 아래의 **passengerName.Set** 노드에서 표준 검사점: "**CheckName**" 노드를 선택합니다.
요약에 검사된 속성과 해당 값을 포함하여 표준 검사점의 세부 정보가 표시됩니다. 개체 속성의 실제 값이 예상 값과 일치하기 때문에 검사점을 통과했습니다.

s

5. 테이블 검사점의 결과를 봅니다.

- a. 결과 트리에서 **Select Flight** 노드를 확장합니다.
- b. 수행: Select Flight 노드에서 **Check Cost** 노드를 찾습니다.
- c. Check Cost 노드를 확장하고 표준 검사점: **CheckCost** 노드를 선택합니다.

요약에 테이블 검사점 세부 정보가 표시됩니다.

오류 세부 정보

실행 오류 - 단계

浏览 JMS 队列消息6

설명

浏览 JMS 队列消息6 비정상 종료

실행 시간

2015-09-28 10:52:58

경우에 따라 단계 요약에 테이블 개체에서 캡처된 데이터를 포함하여 테이블 검사점에 대한 세부 정보도 표시됩니다.

6. 텍스트 검사점의 결과를 봅니다.

- a. 테스트 흐름에서 **Flight Confirmation** 노드를 찾습니다.
- b. Flight Confirmation 노드 아래의 **progBar.Wait** 단계에서 **Text Checkpoint: CheckOrderCompleted** 노드를 선택합니다.

단계 요약에 검사점 세부 정보가 표시됩니다. 실제 텍스트가 예상 텍스트와 일치하기 때문에 검사점을 통과했습니다.

The screenshot shows a test runner interface with a list of test steps on the left and a detailed view of a text checkpoint on the right.

오류 목록 테스트 흐름 검색...

- SELECT FLIGHT.Click
- Flight Confirmation
 - passengerName.Set
 - 표준 검사점: "passengerName"
 - ORDER.Click
 - progBar.WaitProperty
 - 텍스트 검사점: "Order 91 co..."**
 - 표준 출력: "passengerName..."

검사점 세부 정보

텍스트 검사점
"Order 91 completed"

설명
텍스트 검사점: "completed"이(가) 캡처될 때/소문자 구분: OFF 정확히 일치: OFF 공백 무시: ON

실행 시간
2015-09-24 15:25:01

테스트 개체
WpfObject: "Order 91 completed"

저장소
Local

개체 경로
WpfWindow("HP MyFlight Sample Application").WpfObject("Order 91")


7. 실행 결과를 닫습니다.

문서 창에서 실행 결과가 포함된 탭을 닫습니다.

연습 6g: 출력 값 테스트 만들기

이 연습에서는 출력 값 단계를 추가할 테스트를 만듭니다. 이 테스트는 "1단원: GUI 테스트 및 수행 만들기"(28페이지)에서 만든 Book Flights 테스트에 기반을 둡니다.

1. UFT를 시작하고 Book Flights 테스트를 엽니다.

- 필요한 경우 "모든 테스트에 대한 솔루션 만들기"(25페이지)의 설명대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- 열기 버튼 아래쪽 화살표  열기 를 클릭하고 솔루션 열기를 선택합니다. 솔루션 열기 대화 상자가 열립니다.
- C:\%HOMEPATH%\My Documents\Unified Functional Testing에 있는 Flight Reservation Application.ftsln 파일로 이동하고 열기를 클릭합니다.
"1단원: GUI 테스트 및 수행 만들기"(28페이지)에서 만든 Book Flights 테스트가 포함된 Flight Reservation Application 솔루션이 열립니다.

2. 테스트를 Book Flights Output Value로 저장합니다.

- 솔루션 탐색기에서 Book Flights 테스트 노드를 선택하고 파일 > 다른 이름으로 저장을 선택합니다.
- 다른 이름으로 테스트 저장 대화 상자에서 C:\%HOMEPATH%\My Documents\Unified Functional Testing 디렉터리로 이동하고 테스트를 Book Flights Output Value로 저장합니다.
솔루션 탐색기에서 Book Flights 테스트가 새 Book Flights Output Value 테스트로 대체됩니다. 원래 Book Flights 테스트는 파일 시스템에 저장되어 있습니다.

3. Book Flights 테스트를 다시 솔루션에 추가합니다.

Book Flights 및 Book Flights Output Value 테스트가 같은 솔루션에 포함되어 있으면 두 테스트를 모두 동시에 열 수 있습니다. 이를 통해 테스트를 비교하거나 편집하려는 경우 테스트 사이에서 앞뒤로 전환할 수 있습니다.

참고: 단일 테스트를 한 번만 실행할 수 있습니다.

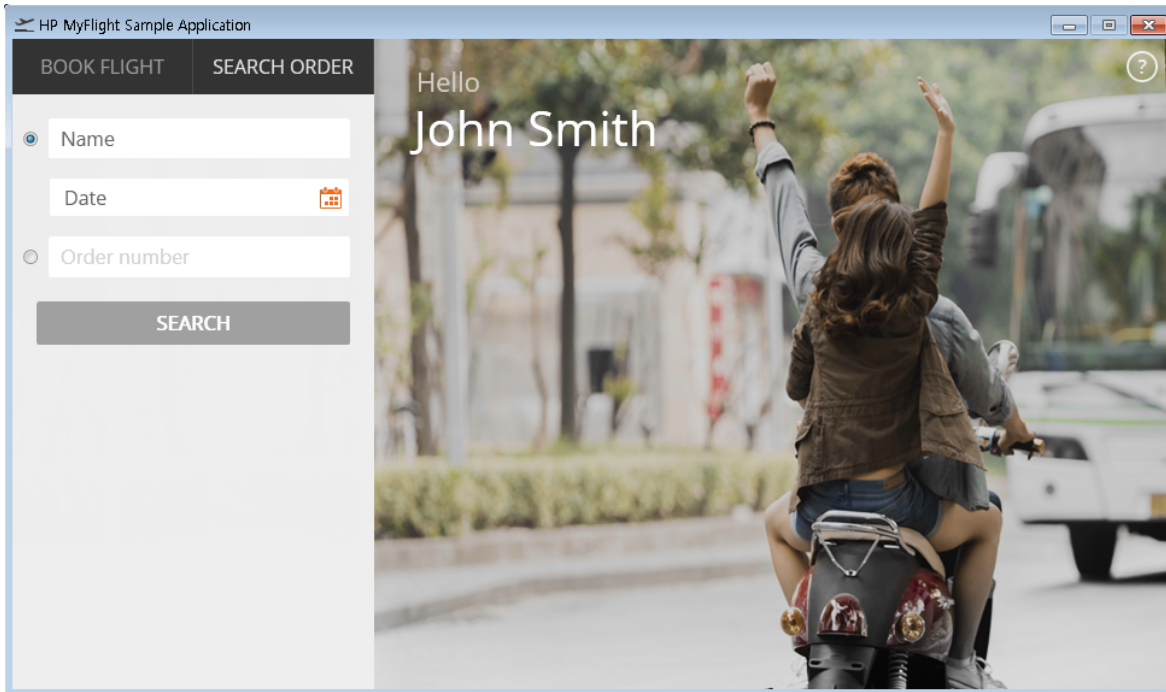
- 파일 > 추가 > 기존 테스트를 선택합니다.
- 기존 테스트 추가 대화 상자에서 C:\%HOMEPATH%\My Documents\Unified Functional Testing에 저장된 Book Flights 테스트로 이동하고 추가를 클릭합니다.
Book Flights 테스트가 다시 솔루션 탐색기에서 개별 노드로 표시됩니다.

출력 값을 추가할 테스트를 만들었으므로 이제 "연습 6h: 출력 값 단계 추가"(112페이지)로 이동하여 출력 값 단계를 테스트에 추가하십시오.

연습 6h: 출력 값 단계 추가


원래 Book Flights 테스트를 만들 때 응용 프로그램에 로그인하고, 출발 및 도착 세부 정보를 입력하고, 항공편을 선택하고, 특정 고객에 대한 항공편을 예약하는 기본 응용 프로그램 흐름에 대한 수행을 만들었습니다.

항공편 예약 응용 프로그램에는 이전에 생성된 항공편 주문을 모두 검색할 수 있는 추가 응용 프로그램 영역이 있습니다.



이 연습에서는 테스트의 Flight Confirmation 수행을 통해 Flight Details 페이지에서 단계 출력을 가져오고 이 출력을 검색 페이지에서 개체 매개 변수로 사용하는 출력 값 단계를 만듭니다.

1. 검색 페이지에서 테스트 단계에 대한 수행을 만듭니다.

- 솔루션 탐색기에서 **Book Flights** 테스트 노드를 클릭합니다. 테스트 흐름 캔버스가 문서 창에서 개별 탭으로 열립니다.
- 도구 모음에서 새 수행 호출 삽입 버튼  을 클릭합니다. 새 수행 호출 삽입 대화 상자가 열립니다.
- 새 수행 호출 삽입 대화 상자에서 새 수행의 이름을 Flight Order Search로 입력합니다.
- 기타 설정을 기본값으로 유지하고 확인을 클릭합니다.

Flight Order Search라는 새 수행 블록이 테스트 흐름 종료 시 캔버스에 추가되고 Flight Order Search 수행이 문서 창에서 개별 탭으로 열립니다.

2. 기타 응용 프로그램 페이지에 대한 개체 저장소를 만듭니다.

Book Flights 테스트를 만들 때 기본 응용 프로그램 페이지에 대한 개체 저장소만 만들었습니다. 검색 페이지에 대한 테스트 단계를 만들려면 검색 페이지에 대한 개체 저장소를 추가로 만들어야 합니다.

- a. "Flight Reservation Application 탐색"(23페이지)의 설명대로 항공편 예약 응용 프로그램을 엽니다.
- b. 응용 프로그램에 대한 로그인 자격 증명을 입력합니다.
 - **사용자 이름:** john
 - **비밀번호:** hp
- c. **확인**을 클릭하여 로그인합니다. Flight Finder 페이지가 열립니다.
- d. Flight Finder 페이지에서 창의 왼쪽 위에 있는 **Search Order** 버튼을 클릭합니다. Search Details 페이지가 열립니다.
- e. UFT에서 리소스 > 개체 저장소 관리자를 선택합니다. 개체 저장소 관리자 창이 열립니다.
- f. 개체 저장소 관리자에서 "연습 2b: 탐색 및 알아보기를 사용하여 개체 저장소 만들기"(40페이지)에 설명된 대로 탐색 및 알아보기 프로세스를 사용합니다.
- g. 이 페이지의 모든 개체에 대해 알아보고 **파일 > 저장**을 클릭합니다.
- h. **C:\%HOMEPATH%\My Documents\Unified Functional Testing\Tutorial_Object Repositories** 폴더로 이동하고 개체 저장소를 **Search**라는 이름으로 저장합니다.
- i. 항공편 예약 응용 프로그램의 Search Details 페이지에서 **Name** 상자에 John Smith를 입력하고 **Search**를 클릭합니다. John Smith에 대해 예약된 항공편 목록이 별도의 페이지에 표시됩니다.
- j. UFT에서 개체 저장소 관리자 창을 다시 엽니다.
- k. Select Order 페이지에서 이동 및 알아보기 프로세스를 사용하여 이 페이지의 개체에 대해 알아봅니다.
 - l. Select Order 페이지의 모든 개체에 대해 알아보고 **파일 > 저장**을 클릭합니다.
- m. 개체 저장소 저장 대화 상자에서 다시 **C:\%HOMEPATH%\My Documents\Unified Functional Testing\Tutorial_Object Repositories** 폴더로 이동하고 개체 저장소를 Search_Results라는 이름으로 저장합니다.
- n. 개체 저장소 관리자를 닫습니다.

3. 새 개체 저장소를 Flight Order Search 수행과 연결합니다.

- a. UFT 기본 창의 솔루션 탐색기에서 Book Flights Output Value 테스트 노드 아래에 있는 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 수행 노드를 마우스 오른쪽 버튼으로 클릭하고 **수행과 저장소 연결**을 선택합니다.
- b. 공유 개체 저장소 열기 창에서 **C:\%HOMEPATH%\My Documents\Unified Functional Testing\Tutorial_Object Repositories** 폴더로 이동하고 **Search.tsr** 파일을 선택합니다.
- c. **열기**를 클릭하여 개체 저장소를 연결합니다.

- d. 프로세스를 반복하여 **C:\%HOMEPATH%\My Documents\Unified Functional Testing\Tutorial_Object Repositories** 폴더에 저장된 Search Results.tsr 개체 저장소를 연결합니다.

Search.tsr 및 **Search Results.tsr** 개체 저장소 파일이 Flight Order Search 수행의 하위 노드로 표시됩니다.

4. Search and Flight Orders 페이지를 사용할 단계를 추가합니다.

- a. 솔루션 탐색기에서 **Flight Order Search** 수행 노드를 두 번 클릭합니다. Flight Order Search 수행이 문서 창에서 개별 탭으로 표시됩니다.
- b. 키워드 보기가 표시되면 **보기 > 편집기**를 클릭하여 편집기를 표시합니다.
- c. 편집기에서 다음 줄을 붙여넣습니다.

```
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
```

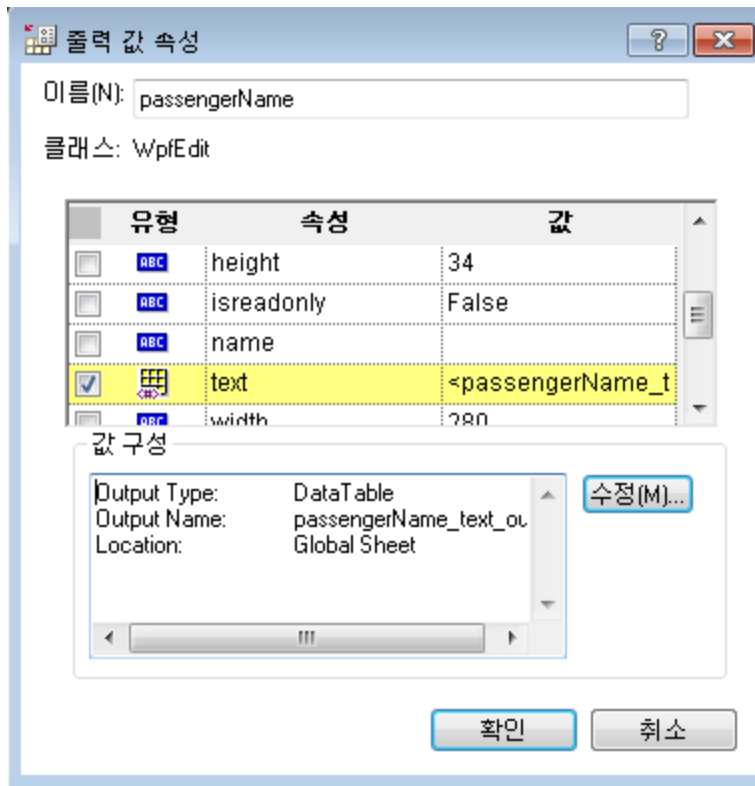
5. 항공편 예약 응용 프로그램의 Flight Details 페이지를 엽니다.

- a. 항공편 예약 응용 프로그램의 Search Results 페이지에서 **뒤로** 버튼을 클릭합니다. Search Details 페이지가 표시됩니다.
- b. Search Details 페이지의 왼쪽 위에서 **Book Flight** 버튼을 클릭합니다. Flight Finder 페이지가 열립니다.
- c. Flight Finder 페이지에서 항공편 세부 정보를 입력합니다.
 - o **출발 도시:** Los Angeles
 - o **도착 도시:** Sydney
 - o **날짜:** 내일 날짜
 - o **클래스:** Business
 - o **티켓:** 2
- d. **Find Flights** 버튼을 클릭합니다. Select Flight 페이지가 열립니다.
- e. Select Flight 페이지에서 첫 번째 행을 선택하고 **Select Flight**를 클릭합니다. Flight Details 페이지가 열립니다.

6. Flight Confirmation 수행에 출력 값 단계를 추가합니다.

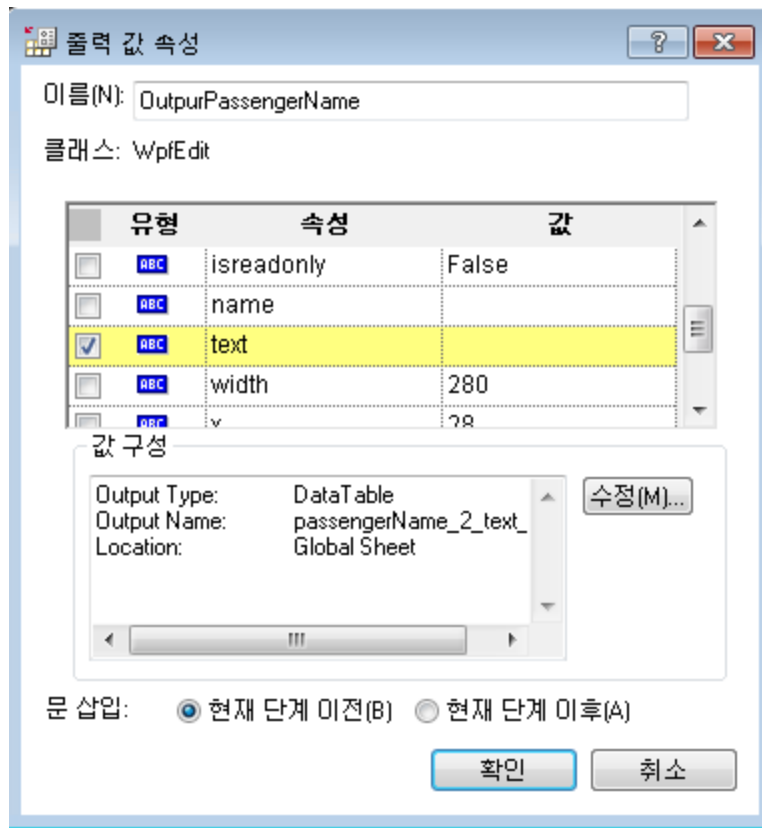
- a. UFT에서 솔루션 탐색기의 Book Flights 테스트 노드 아래에 있는 **Flight Confirmation** 수행 노드를 두 번 클릭합니다. Flight Confirmation 수행이 문서 창에서 개별 탭으로 열립니다.
- b. 편집기가 표시되면 **보기 > 키워드 보기**를 클릭하여 키워드 보기를 표시합니다.

- c. 키워드 보기에서 byName 단계를 마우스 오른쪽 버튼으로 클릭하고 출력 값 삽입을 선택합니다. 출력 값 속성 대화 상자가 열립니다.

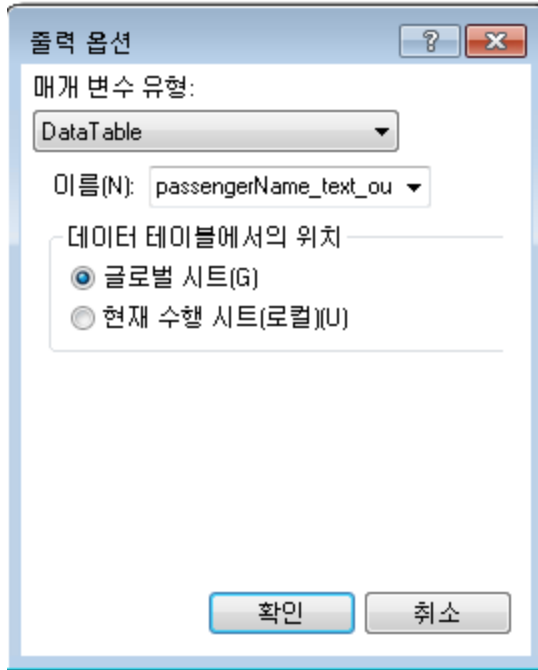


- 대화 상자에 출력 값을 삽입할 속성이 표시됩니다.
- 이름은 응용 프로그램에 정의된 개체의 이름입니다(이 경우 **byName**).
 - 클래스는 개체가 편집 상자임을 나타내는 개체 유형입니다(이 경우 **WpfEdit**).
 - 유형 열의 **ABC** 아이콘은 속성 값이 상수임을 나타냅니다.
 - 표에는 출력으로 전송할 수 있는 개체의 속성 이름 및 값을 포함하여 출력하도록 선택할 수 있는 개체 속성이 나열됩니다.
- d. 이름 상자에 새 출력 값 이름으로 OutputPassengerName을 입력합니다.

- e. 개체 속성 표에서 속성을 스크롤하고 텍스트 속성 행을 선택합니다. 이 속성에 제공된 값이 없는지 확인합니다.



- f. 속성 표 아래에서 수정 버튼을 클릭합니다. 출력 옵션 대화 상자가 열립니다. (이 대화 상자에서 이 테스트 단계의 출력을 저장할 위치를 결정할 수 있습니다.)



출력 값을 만들 때 출력 값을 저장할 위치를 선택할 수 있습니다.

- DataTable 매개 변수
- 테스트/수행 매개 변수(이미 만든 경우)
- 환경 변수
- 구성 요소 매개 변수(이미 만든 경우)

이 자습서에서는 출력 값을 데이터 테이블에 저장합니다.

- g. **매개 변수 유형** 드롭다운 메뉴에서 **DataTable**을 선택합니다. UFT에서는 대화 상자의 필드를 업데이트하고 매개 변수의 기본 이름을 제안합니다.
- h. **이름** 필드에 `passengerName_text`를 입력합니다.
- i. **데이터 테이블에서의 위치** 영역에서 **글로벌 시트**가 선택되었는지 확인합니다. 그러면 출력이 글로벌 데이터 시트에 저장되므로 값을 테스트의 다른 수행에서 액세스할 수 있습니다.
- j. **확인**을 클릭하여 대화 상자를 닫습니다. UFT에서 출력 값 속성 대화 상자의 값을 업데이트하여 DataTable 매개 변수 저장소 옵션을 반영합니다.

유형	속성	값
<input type="checkbox"/>	isreadonly	False
<input type="checkbox"/>	name	
<input checked="" type="checkbox"/>	text	<passengerName_
<input type="checkbox"/>	width	280
<input type="checkbox"/>	x	28

- k. **문 삽입 영역에서 현재 단계 뒤 옵션을 선택하고 확인을 클릭합니다.**
UFT에서 출력 단계를 passengerName.Set 단계 바로 뒤에 삽입합니다.

항목	작업	값	설명
HP MyFlight Sample Application			
passengerName	Set	"John Smith"	"John Smith"을(를) "passengerName" edit box에 입력합니다.
passengerName	Output	Checkpoint("passengerN...	스크립트에서 나중에 사용할 수 있도록 "passengerName" e
ORDER	Click		"ORDER" button을(를) 클릭합니다.
progBar	WaitProperty	"value", "100"	다음 단계를 계속하기 전에 "progBar" progress bar의 "valu...
NEW SEARCH	Click		"NEW SEARCH" button을(를) 클릭합니다.
+ 새 단계			

편집기에 단계가 다음과 같이 표시됩니다.

```

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
    
```

데이터 테이블의 글로벌 시트도 이에 따라 업데이트됩니다.

데이터	
A1	
	passengerName_text_out
1	
2	
3	
4	
5	

Global Login Flight Finder

7. 저장된 출력 값을 사용하여 검색 수행을 매개 변수화합니다.

- a. 솔루션 탐색기에서 **Flight Order Search** 수행 노드를 두 번 클릭합니다. Flight Order Search 수 행이 문서 창에서 개별 탭으로 열립니다.
- b. Flight Order Search 탭에서 **byName** 행을 선택합니다.
- c. **byName** 행의 값 열에서 값 구성 버튼 **<#>** 을 클릭합니다. 값 구성 옵션 대화 상자가 열립니 다.
- d. 값 구성 옵션 대화 상자에서 **매개 변수** 라디오 버튼을 선택합니다.
- e. **매개 변수** 드롭다운 목록에서 **DataTable**을 선택합니다. 대화 상자에서 기타 필드가 이에 따라 업데이트됩니다.
- f. **데이터 테이블에서의 위치** 영역에서 **글로벌 시트** 옵션을 선택합니다.
- g. **이름** 드롭다운 목록에서 **passengerName_text** 매개 변수를 선택하고 **확인**을 클릭합니다.

UFT에서 **byName** 행을 업데이트하여 이제 값이 데이터 테이블 매개 변수에서 제공됨을 반영합니다.

항목	작업	값	설명
HP MyFlight Sample Application			
WpfTabStrip	Select	"SEARCH ORDER"	"WpfTabStrip" tab strip의 "SEARCH ORDER" 탭을 선택합니다.
byName	Set	DataTable("passengerName_...")	<the 'passengerName_text_out' Data Table column의 값>을(를)
SEARCH	Click		"SEARCH" button을(를) 클릭합니다.
ordersDataGrid	SelectCell	1,1	"ordersDataGrid" table의 행 1, 열 1의 셀을 선택합니다.
SELECT ORDER	Click		"SELECT ORDER" button을(를) 클릭합니다.
+ 새 단계			


편집기에 문이 다음과 같이 표시됩니다.

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

8. 테스트를 저장합니다.

저장  을 클릭합니다.

9. 테스트를 실행하고 실행 결과를 봅니다.

- 기록 > 기록 및 실행 설정을 선택합니다. 기록 및 실행 설정 대화 상자가 열립니다.
- 기록 및 실행 설정 대화 상자의 Windows 응용 프로그램 탭에서 기록 및 실행 장소: 옵션을 선택합니다. 이전 테스트 실행의 응용 프로그램 세부 정보를 저장해야 합니다.
- 확인을 클릭하여 설정을 저장하고 대화 상자를 닫습니다.
- 도구 모음에서 실행 버튼  을 클릭합니다.
- 결과 위치 탭에서 새 실행 결과 폴더 옵션이 선택되었는지 확인합니다. 기본 결과 폴더 이름을 그대로 사용합니다.
- 확인을 클릭합니다. 테스트 실행이 완료되면 실행 결과가 열립니다.
- 테스트 흐름에서 **Flight Confirmation** 수행 노드를 찾습니다.
- Flight Confirmation Summary 노드에서 **Standard Output: "OutputPassengerName"** 노드를 선택합니다. 실행 결과에 테스트 단계 요약이 표시됩니다.
요약에는 출력 값 세부 정보가 표시됩니다.
- 테스트 흐름에서 **Flight Order Search** 노드를 찾습니다.
- Flight Order Search 노드에서 **byName.Set** 노드를 선택합니다. 실행 결과에 테스트 단계 요약이 표시됩니다.
요약에는 Set 작업에 사용된 값을 포함하여 이 단계의 결과가 표시됩니다. 이 값은 출력 값이어야 합니다.

10. 실행 결과를 닫습니다.

실행 결과 확인을 마치면 실행 결과를 보여 주는 탭을 닫습니다.

7단원: 함수 및 함수 라이브러리 만들기

UFT에서는 다양한 테스트 요구 사항을 충족할 수 있는 많은 기본 제공 함수 및 메서드를 제공합니다. 하지만 경우에 따라 특정 테스트 개체 클래스에 대해 기본적으로 사용할 수 없는 특정 작업을 수행해야 할 수 있습니다. 이 경우 이 작업에 대한 사용자 정의 함수를 만들 수 있습니다. 테스트와 연결된 함수 라이브러리 파일에 이 함수를 저장하고 작업을 수행해야 할 때마다 함수 호출을 단계로 삽입합니다.

"2단원: 개체 저장소 만들기"(32페이지)에서 공유 개체 저장소를 만들고 테스트의 수행과 연결했습니다. 이 단원에서는 함수 및 함수 라이브러리를 만들고 함수 라이브러리를 테스트와 연결하여 비슷한 프로세스를 사용합니다. 함수 라이브러리를 테스트와 연결하면 테스트에서 모든 함수를 호출할 수 있습니다.

이 단원의 내용은 다음과 같습니다.

- 함수 및 함수 라이브러리 - 개요121
- 연습 7a: 함수 만들기121
- 연습 7b: 테스트에 함수 라이브러리 연결123
- 연습 7c: 함수를 사용하여 검사 수행 124

함수 및 함수 라이브러리 - 개요

UFT에서 함수를 만들어서 UFT의 표준 클래스 및 메서드에서 지원하지 않는 특수 작업을 수행할 수 있습니다. 함수는 기본적으로 적합한 메서드가 없는 특정 작업을 수행하는 코딩된 단계 집합입니다. 테스트에 해당 작업을 포함하고 이 작업을 여러 번 반복할 수도 있습니다. 따라서 함수에 손쉽게 액세스할 수 있어야 합니다.

함수를 만들면 함수 라이브러리에 함수를 저장할 수 있습니다. 이러한 함수 라이브러리는 사용자 정의 함수의 저장소로 사용됩니다. 각 함수 라이브러리는 하나 이상의 테스트에 할당할 수 있습니다. 그러면 테스트에서 함수를 테스트 단계로 호출할 수 있습니다.

이 단원에서는 항공편 예약 응용 프로그램에서 생성된 페이지에서 날짜 형식을 확인하는 함수를 만들고 해당 함수 호출을 테스트에 추가합니다.

연습 7a: 함수 만들기

이 연습에서는 테스트에서 호출할 함수를 만듭니다. 이 함수는 날짜가 적절한 형식으로 표시되는지 확인합니다. 또한 함수는 날짜가 유효한지 확인합니다. 예를 들어 월이 12를 초과하지 않는지 또는 날짜가 31을 초과하지 않는지 확인합니다.

1. UFT를 시작하고 Book Flights 테스트를 엽니다.

- a. UFT가 현재 열려 있지 않으면 "모든 테스트에 대한 솔루션 만들기"(25페이지)에 설명된 대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- b. 파일 > 열기 > 솔루션을 선택합니다. 솔루션 열기 대화 상자가 열립니다.
- c. 솔루션 열기 대화 상자에서 C:\%HOMEPATH%\My Documents\Unified Functional Testing에 있는 Flight Reservation Application.ftsln 파일로 이동하고 열기를 클릭합니다.
Book Flights 테스트를 포함하여 테스트가 솔루션 탐색기에 표시됩니다.

2. 함수 라이브러리를 만듭니다.

- a. 도구 모음에서 새로 만들기 버튼 아래쪽 화살표 * 새로 만들기 를 클릭하고 함수 라이브러리를 선택합니다. 새 함수 라이브러리 대화 상자가 열립니다.
- b. 새 함수 라이브러리 대화 상자에서 다음과 같이 함수 라이브러리 세부 정보를 입력합니다.

찾는 위치:	<ol style="list-style-type: none"> i. C:\%HOMEPATH%\My Documents\Unified Functional Testing 폴더로 이동합니다. ii. 이 폴더에서 Tutorial_Function Libraries라는 새 폴더를 만듭니다. iii. 이 폴더를 엽니다.
파일 이름:	CheckDate Function

- c. 만들기를 클릭합니다. UFT에서는 빈 함수 라이브러리를 문서 창의 개별 탭으로 엽니다.

3. 함수를 만듭니다.

CheckDate 함수 라이브러리에서 다음 코드를 붙여넣습니다.

```
'The following function checks whether a date string (dateStr)
'has the characters representing DD-<MMM string>-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. 함수 라이브러리를 저장합니다.

저장  을 클릭합니다.

5. 함수 라이브러리를 닫습니다.

파일 > 닫기를 선택합니다. 함수 라이브러리 탭이 닫히고 테스트는 열려 있습니다.

함수를 만들었으므로 이제 이러한 함수를 테스트 단계에서 사용하려면 함수를 테스트와 연결해야 합니다. "[연습 7b: 테스트에 함수 라이브러리 연결](#)"(123페이지)을 계속 진행하여 함수 라이브러리를 테스트와 연결하는 방법을 알아보십시오.

연습 7b: 테스트에 함수 라이브러리 연결


"[연습 7a: 함수 만들기](#)"(121페이지)에서 함수 및 함수 라이브러리를 만들어서 테스트 개체에서 날짜를 실행했습니다. 그러나 테스트에서 이 함수를 사용하려면 함수 라이브러리를 테스트와 연결해야 합니다.

이 연습에서는 함수 라이브러리를 테스트와 연결합니다.

1. UFT를 시작하고 Book Flights 테스트를 엽니다.

- UFT가 현재 열려 있지 않으면 "[모든 테스트에 대한 솔루션 만들기](#)"(25페이지)에 설명된 대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- 파일 > 열기 > 솔루션을 선택합니다. 솔루션 열기 대화 상자가 열립니다.
- 솔루션 열기 대화 상자에서 C:\%HOMEPATH%\My Documents\Unified Functional Testing에 있는 **Flight Reservation Application.ftsln** 파일로 이동하고 열기를 클릭합니다.
Book Flights 테스트를 포함하여 테스트가 솔루션 탐색기에 표시됩니다.

2. CheckDate Function.qfl을 Book Flights 테스트와 연결합니다.

- 솔루션 탐색기가 열려 있지 않으면 도구 모음에서 솔루션 탐색기 버튼  을 클릭합니다.
- 솔루션 탐색기에서 **Book Flights** 테스트 노드를 마우스 오른쪽 버튼으로 클릭하고 추가 > 함수 라이브러리 연결을 선택합니다. 함수 라이브러리 열기 대화 상자가 열립니다.

- c. 함수 라이브러리 열기 대화 상자에서 **C:\%HOMEPATH%\Unified Functional Testing\Tutorial_Function Libraries**에 있는 **CheckDate Function.qfl** 파일로 이동하고 열기를 클릭합니다.
- d. 자동 상대 경로 변환 대화 상자에서 **예**를 클릭합니다. (그러면 함수 라이브러리 경로가 상대 경로로 변환됩니다.)
이제 함수 라이브러리가 Book Flights 테스트와 연결되고 솔루션 탐색기에서 Book Flights 테스트의 **Function Libraries** 폴더에 표시됩니다.

참고: 상대 경로를 사용하면 폴더 계층 구조가 동일하게 유지되는 한 테스트와 기타 파일이 포함된 폴더의 위치를 이동해도 경로가 계속 유효합니다.

3. 테스트를 저장합니다.

문서 창에서 Book Flights 캔버스 탭을 선택하고 저장  을 클릭합니다.

함수 라이브러리를 테스트와 연결했으므로 이제 테스트 단계에서 함수를 사용할 수 있습니다. "[연습 7c: 함수를 사용하여 검사 수행\(124페이지\)](#)"을 계속 진행하여 테스트 단계에서 함수를 사용하십시오.

연습 7c: 함수를 사용하여 검사 수행

"[6단원: 검사점 및 출력 값 만들기\(89페이지\)](#)"에서는 항공편 예약 응용 프로그램의 개체를 확인하는 여러 검사점을 만들었습니다.

이 연습에서는 UFT 사용자 인터페이스를 사용하여 검사점을 만드는 대신 "[연습 7a: 함수 만들기\(121페이지\)](#)"에서 만든 함수를 사용하여 Flight Finder 수행에서 달력 개체의 날짜 형식을 확인합니다.

1. UFT를 시작하고 Book Flights 테스트를 엽니다.

- a. UFT가 현재 열려 있지 않으면 "[모든 테스트에 대한 솔루션 만들기\(25페이지\)](#)"에 설명된 대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- b. **파일 > 열기 > 솔루션**을 선택합니다. 솔루션 열기 대화 상자가 열립니다.
- c. 솔루션 열기 대화 상자에서 **C:\%HOMEPATH%\My Documents\Unified Functional Testing**에 있는 **Flight Reservation Application.ftsln** 파일로 이동하고 열기를 클릭합니다.
Book Flights 테스트를 포함하여 테스트가 솔루션 탐색기에 표시됩니다.

2. 테스트를 Book Flights Function으로 저장합니다.


- a. 솔루션 탐색기에서 **Book Flights** 노드를 마우스 오른쪽 버튼으로 클릭하고 **다른 이름으로 저장**을 선택합니다. 다른 이름으로 테스트 저장 대화 상자가 열립니다.
- b. 다른 이름으로 테스트 저장 대화 상자에서 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 폴더로 이동합니다.
- c. 테스트 이름으로 **Book Flights Function**을 입력합니다.

솔루션 탐색기에서 **Book Flights** 테스트가 새 **Book Flights Function** 테스트로 대체됩니다. Book Flights 테스트는 파일 시스템에 별도로 저장되어 있습니다.

3. Book Flights 테스트를 다시 솔루션에 추가합니다.

Book Flights 및 Book Flights Function 테스트가 같은 솔루션에 포함되어 있으면 두 테스트를 동시에 열 수 있습니다. 이를 통해 테스트를 비교하거나 편집하려는 경우 테스트 사이에서 앞뒤로 전환할 수 있습니다.

참고: 단일 테스트를 한 번만 실행할 수 있습니다.



- 도구 모음에서 **추가** 버튼 드롭다운 화살표  **추가** 를 클릭하고 **기존 테스트 추가**를 선택합니다. 솔루션에 기존 테스트 추가 대화 상자가 열립니다.
- 솔루션에 기존 테스트 추가 대화 상자에서 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 폴더로 이동합니다.
- 폴더에서 **Book Flights** 테스트를 선택하고 **열기**를 클릭합니다.
Book Flights 테스트 노드가 다시 솔루션 탐색기에 추가됩니다. 테스트가 알파벳순으로 구성되었는지 확인합니다.

4. 항공편 예약 응용 프로그램의 Flight Finder 페이지를 표시합니다.

- "[Flight Reservation Application 탐색](#)"(23페이지)의 설명대로 항공편 예약 응용 프로그램을 엽니다.
- 로그인 화면에서 로그인 사용자 자격 증명을 입력합니다.
 - 사용자 이름:** john
 - 비밀번호:** hp
- 확인**을 클릭하여 로그인합니다. Flight Finder 페이지가 열립니다.

5. 검사점 개체에 대한 개체 속성에서 스파이 작업을 수행합니다.

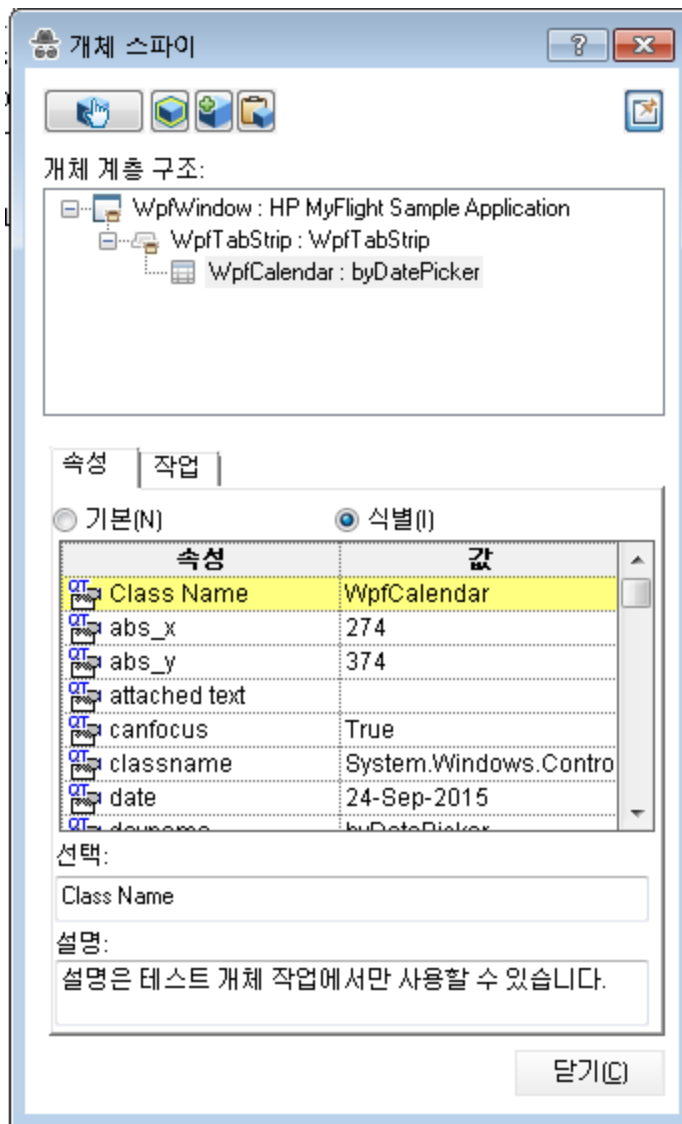
이 연습에서는 함수를 사용하여 달력 개체의 날짜 형식을 확인합니다. 이 작업을 하려면 검사점을 설정할 개체의 속성을 알아야 합니다.

- UFT 창의 도구 모음에서 **개체 스파이** 버튼  을 클릭합니다. 개체 스파이 대화 상자가 열립니다.
- 개체 스파이 대화 상자가 열려 표시된 상태로 스파이 작업을 수행할 수 있도록 개체 스파이 대화 상자를 창 가장자리로 이동합니다.
- 개체 스파이 대화 상자에서 손모양 아이콘 버튼  을 클릭합니다. 손모양 아이콘을 누르면 UFT가 숨겨지고 개체 스파이 대화 상자가 항공편 예약 응용 프로그램 위에 표시됩니다.
- 항공편 예약 응용 프로그램에서 날짜 입력 필드 개체를 클릭합니다.

2014-07-17



개체 스파이 대화 상자에 개체 속성이 표시됩니다.



- e. 속성 표(대화 상자의 아래쪽)에서 날짜 값(DD-MMM-YYYY 형식)이 있는 속성을 찾을 때까지 속성을 스크롤합니다. 나중에 테스트에 삽입되는 함수 호출 단계에 속성이 필요하므로 이 속성의 이름을 기록해 둡니다.
 - f. 닫기를 클릭하여 개체 스파이를 닫고 테스트로 돌아갑니다.
6. **함수 호출에 대한 수행을 엽니다.**
- 솔루션 탐색기에서 **Book Flights Function** 테스트 노드를 확장하고 **Flight Finder** 수행을 두 번 클릭합니다.
- Flight Finder 수행이 문서 창에서 개별 탭으로 열립니다.
7. **편집기에서 함수 호출을 추가합니다.**

이 단계에서는 "연습 7a: 함수 만들기"(121페이지)에서 함수 라이브러리에 추가한 `check_data_validity` 함수를 추가합니다.

- 키워드 보기가 표시되면 **보기 > 편집기**를 클릭하여 편집기를 표시합니다.
- 편집기에서 커서를 **WpfCalendar.SetDate** 단계 뒤에 놓고 **Enter** 키를 누릅니다.
- 편집기에 단계를 추가하여 **datePicker** 개체에 대한 날짜 속성을 검색합니다.

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

이 단계에서는 UFT가 속성을 확인할 때 해당 개체에 대한 검사점 함수를 실행하도록 날짜 속성의 값을 검색합니다.

- 편집기에 또 다른 단계를 추가하여 검사점 함수를 호출합니다.

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

이러한 단계를 추가하면 수행이 다음과 같이 표시됩니다(사용하는 날짜가 다른 경우에도 마찬가지임).

```
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
```

- 단계를 붙여넣고 나서 `WpfWindow("HP MyFlight Sample Application").WpfComboBox("Class").Select "Business"` 단계가 붙여넣은 단계 뒤의 자체 줄에서 시작되는지 확인합니다. 그렇지 않으면 이 단계 앞에 커서를 놓고 **Enter** 키를 누릅니다.
UFT가 응용 프로그램에서 다음 필드를 편집하기 전에 이 함수를 실행하도록 하려 합니다.
- 수행에서 줄이 `departureDate`로 시작하고 ("`date`")로 끝나도록 코드의 **departureDate** 줄에서 줄 바꿈을 제거합니다.

8. 키워드 보기에서 이러한 단계를 확인합니다.

보기 > 키워드 보기를 선택하여 키워드 보기로 전환합니다.


GetROProperty 단계에서 개별 단계로 표시된 함수 호출 단계를 확인합니다.

항목	작업	값	설명
HP MyFlight Sample Application			
fromCity	Select	"Los Angeles"	"fromCity" list에서 "Los Angeles" 항목을 선택합니다.
toCity	Select	Sydney	"toCity" list에서 Sydney 항목을 선택합니다.
datePicker	SetDate	"17-Jul-2014"	"datePicker" calendar의 날짜 또는 날짜 범위를 "17-Jul-2014"(으)로 설정...
datePicker	GetROProperty	"date"	"datePicker" calendar에 대한 "date" 속성의 현재 값을 검색합니다. 변수 'd...
함수 호출	check_data_validity	departureDate	
reporter	ReportEvent	micPass,"Date is v...	departureDate을(를) 실행 결과로 보고하고 "Date is valid" 단계의 상태를
Class	Select	"Business"	"Class" list에서 "Business" 항목을 선택합니다.
numOfTickets	Select	"2"	"numOfTickets" list에서 "2" 항목을 선택합니다.
FIND FLIGHTS	Click		"FIND FLIGHTS" button을(를) 클릭합니다.

9. 테스트를 저장합니다.

파일 > 저장을 선택합니다.

10. 테스트를 실행합니다.

- a. 도구 모음에서 **실행 버튼**  을 클릭합니다. 실행 대화 상자가 열립니다.

참고: 테스트를 실행하기 전에 항공편 예약 응용 프로그램이 닫혔는지 확인합니다.

- b. 실행 대화 상자의 **결과 위치** 탭에서 **새 실행 결과 폴더** 옵션이 선택되었는지 확인합니다. 기본 결과 폴더 이름을 그대로 사용합니다.
- c. **실행**을 클릭합니다. UFT에서 단계를 순서대로 실행합니다.
테스트 실행이 완료되면 실행 결과가 열립니다.

11. 실행 결과를 분석합니다.

- a. 실행 결과, 테스트 흐름의 **Flight Finder** 노드 아래에서 **Date is valid node**를 선택합니다.
단계 이름 옆에 녹색 확인 표시가 표시되는지 확인합니다. 이는 추가한 함수에 관한 검사점을 통과했음을 나타냅니다.

8단원: 테스트에서 이미지 인식 사용

테스트와 테스트 개체를 만들 때 일반적인 개체 식별이 테스트 중에 응용 프로그램에서 개체를 식별하는 데 도움이 되지 않는 경우가 있습니다. 표준 개체 식별은 응용 프로그램 또는 브라우저 창에서의 위치, 창에 표시된 시간 또는 기타 다양한 속성과 같은 개체 속성에 의존하므로 테스트 중에 응용 프로그램에서 개체를 식별하는 데 도움이 되지 않습니다.

일반적인 개체 식별이 제대로 작동하지 않거나 요구 사항에 맞지 않는 경우를 위해 UFT에는 **이미지 인식**이라는 이미지 기반 개체 인식 메커니즘이 있습니다. 이 메커니즘을 통해 이미지 스냅샷을 캡처하고 캡처된 이미지를 테스트 실행 중에 개체로 사용하여 개체를 식별할 수 있습니다.

이 단원에서는 이미지 인식을 사용하여 개체를 식별하고 이러한 개체를 테스트에서 사용하는 방법을 알아봅니다.

이 단원의 내용은 다음과 같습니다.

- [이미지 인식 개체 식별 - 개요](#) 130
- [연습 8a: 이미지 인식 개체에 대한 테스트 만들기](#) 130
- [연습 8b: 개체 저장소에 이미지 인식 개체 추가](#) 131
- [연습 8c: 테스트에서 이미지 인식 개체 사용](#) 134

이미지 인식 개체 식별 - 개요

UFT에서 이미지 기반 식별 기능인 **이미지 인식**을 사용하여 응용 프로그램에서 개체를 인식할 수 있습니다. 이미지 인식 기능은 응용 프로그램/개체 설계의 일부인 개체 속성을 사용하지 않고 표시되는 모양을 기반으로 개체를 식별합니다. 이미지 인식 개체 식별은 응용 프로그램이 UFT에서 지원하지 않는 기술이나 원격 컴퓨터에서 실행되는 응용 프로그램을 사용하여 설계된 경우 특히 유용합니다.

이미지 인식 개체 식별을 사용할 때 UFT에서는 개체 이미지를 UFT에서 만드는 이미지 인식 테스트 개체 일부로 저장합니다. 그리고 나서 테스트를 실행하면 UFT에서는 이미지를 기본 개체 속성으로 사용하여 런타임에 응용 프로그램에서 개체를 식별합니다.

개체 저장소에서도 기록 시 이미지 인식 개체를 만들 수 있습니다. 이 단원에서는 개체 저장소에서 이미지 인식 개체를 추가하는 작업만 수행합니다.

연습 8a: 이미지 인식 개체에 대한 테스트 만들기

이 연습에서는 이미지 인식 개체를 사용하여 테스트 단계를 추가하는 테스트를 만듭니다. 이 테스트는 **Book Flights** 테스트가 포함된 솔루션과 별도로 저장됩니다.

1. UFT를 시작합니다.

- UFT가 열려 있지 않으면 "**모든 테스트에 대한 솔루션 만들기**"(25페이지)에 설명된 대로 엽니다. WPF Add-in이 열렸는지 확인합니다.
- 도구 모음에서 **새로 만들기 버튼** * 새로 만들기 을 클릭합니다. 새 테스트 대화 상자가 열립니다.
- 새 테스트 대화 상자에서 **GUI 테스트**를 선택합니다.
- 테스트 이름을 **Insight**로 지정합니다. 기본 위치를 그대로 사용합니다.

참고: 이 테스트에는 솔루션 이름을 입력하지 마십시오. 이 테스트는 이 자습서의 다른 단원에서 사용되는 솔루션과 별도로 저장됩니다.

- 만들기**를 클릭하여 테스트를 만듭니다.
테스트가 문서 창에서 개별 탭으로 열립니다.

2. 개체 저장소를 수행과 연결합니다.

이 테스트에서는 모든 테스트 단계를 하나의 수행으로 유지합니다. 그러나 적절한 테스트 개체를 포함하려면 여러 개체 저장소가 필요합니다.

이 테스트에서는 항공편 예약 응용 프로그램에 로그인하고 판촉 이미지에서 링크를 클릭하여 항공편을 주문합니다. 따라서 응용 프로그램의 Login 페이지와 Flight Finder 페이지에 대한 개체 저장소를 포함해야 합니다.

- a. 솔루션 탐색기에서 **Action1** 노드를 마우스 오른쪽 버튼으로 클릭하고 **수행에 저장소 연결**을 선택합니다. 공유 개체 저장소 열기 대화 상자가 열립니다.
- b. 공유 개체 저장소 열기 대화 상자에서 **C:\%HOMEPATH%\Unified Functional Testing\Tutorial_Object Repositories**에 있고 **"2단원: 개체 저장소 만들기"(32페이지)**에서 만든 **Login.tsr** 개체 저장소 파일로 이동하고 **열기**를 클릭합니다.
Login.tsr 개체 저장소는 **Action1** 노드의 하위 노드로 표시됩니다.
- c. 위에 설명된 프로세스를 반복하여 **Flight Finder.tsr** 개체 저장소 파일을 Action1에 추가합니다.

3. 테스트를 저장합니다.


저장  을 클릭합니다.

이미지 인식 개체를 사용할 테스트를 만들었으므로 **"연습 8b: 개체 저장소에 이미지 인식 개체 추가"(131페이지)**로 계속하여 이미지 인식 개체를 테스트 단계에 사용할 수 있도록 개체 저장소에 추가하십시오.

연습 8b: 개체 저장소에 이미지 인식 개체 추가

"연습 8a: 이미지 인식 개체에 대한 테스트 만들기"(130페이지)에서는 이미지 인식 개체에 사용할 테스트의 구조를 만들었습니다. 그러나 이러한 개체를 테스트에 추가하려면 먼저 개체 저장소에 추가해야 합니다. 이 단원에서는 개체 저장소 기능을 사용하여 이미지 인식 개체를 테스트에 포함하는 방법을 알아봅니다.

1. UFT를 시작합니다.

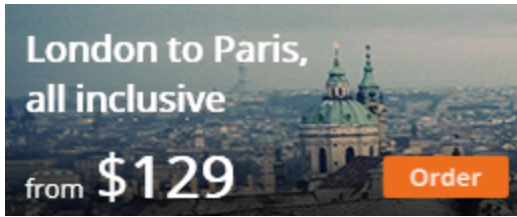
- a. UFT가 열려 있지 않으면 **"모든 테스트에 대한 솔루션 만들기"(25페이지)**에 설명된 대로 엽니다. WPF Add-in이 열렸는지 확인합니다.
- b. 도구 모음에서 **열기** 드롭다운 화살표  열기 를 선택하고 **테스트 열기**를 선택합니다. 테스트 열기 대화 상자가 열립니다.
- c. 새 테스트 대화 상자에서 **C:%HOMEPATH%\Unified Functional Testing**에 저장된 **이미지 인식 테스트**로 이동하고 **열기**를 클릭합니다.
테스트가 문서 창에서 개별 탭으로 열립니다.

2. 새 개체 저장소를 만듭니다.

- a. **리소스 > 개체 저장소 관리자**를 선택합니다. 개체 저장소 관리자 대화 상자가 열립니다.
- b. 개체 저장소 관리자 창에서 **파일 > 저장**을 선택합니다. 공유 개체 저장소 저장 대화 상자가 열립니다.
- c. 공유 개체 저장소 저장 대화 상자에서 **C:%HOMEPATH%\Unified Functional Testing**에 있는 **Tutorial_Object Repositories** 폴더로 이동합니다.
- d. 이 개체 저장소의 이름을 **Insight**로 지정하고 **저장**을 클릭합니다.

3. 항공편 예약 응용 프로그램의 Flight Finder 페이지를 엽니다.


항공편 예약 응용 프로그램의 Flight Finder 페이지에는 특별 항공편에 대한 광고를 표시하며 변경되는 개체가 있습니다.

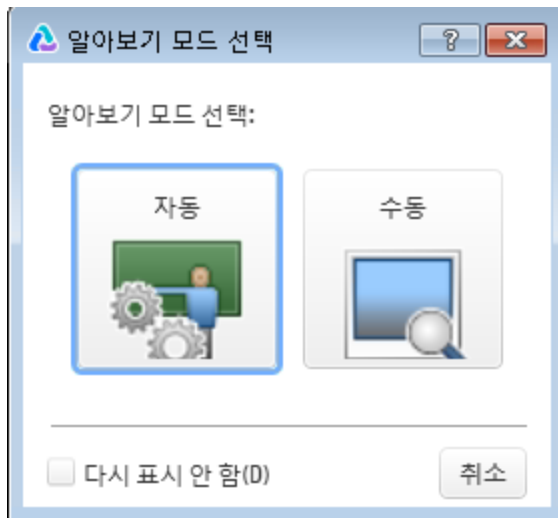


기존 개체 인식은 다른 개체에 상대적인 화면상 위치, 서수 식별자 등의 요소에 의존하는 것에 반해 이 이미지는 응용 프로그램에서 주기적으로 변경되므로 이미지 인식 개체 인식을 통해 테스트할 수 있는 개체 유형입니다. 또한 이미지 내부의 **Order** 버튼이 개별 개체로 식별되지 않습니다. 예를 들어 개체 스파이를 사용하여 **Order** 버튼을 식별하면 UFT가 이 버튼을 식별할 수 없습니다.

- a. "Flight Reservation Application 탐색"(23페이지)의 설명대로 항공편 예약 응용 프로그램을 엽니다.
- b. Login 창에서 사용자 자격 증명을 입력합니다.
 - o 사용자 이름: john
 - o 비밀번호: hp
- c. 확인을 클릭하여 응용 프로그램에 로그인합니다. Flight Finder 페이지가 열립니다.

4. 개체 저장소에 이미지 인식 개체를 추가합니다.

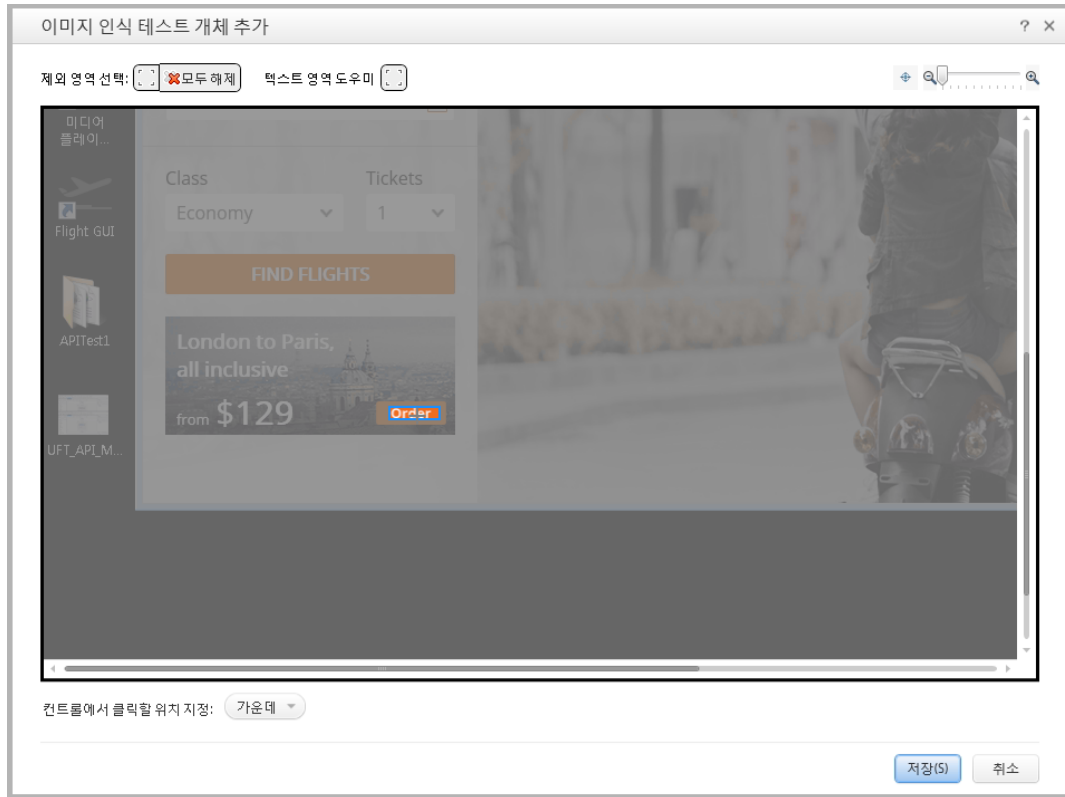
- a. 개체 저장소 관리자의 도구 모음에서 이미지 인식 개체 추가 버튼  을 클릭합니다. 알아보기 모드 선택 대화 상자가 열립니다.



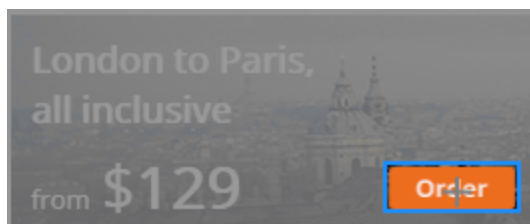
- b. 알아보기 모드 선택 대화 상자에서 **자동** 버튼을 클릭합니다. UFT가 보기에서 숨겨지고 응용 프로그램이 표시됩니다.

참고: 자동 모드를 선택하면 UFT가 자동으로 영역/개체를 이미지 인식 개체로 선택할 수 있습니다. 수동을 선택하면 이미지 인식 개체로 사용할 영역/개체를 지정할 수 있습니다.

- c. 항공편 예약 응용 프로그램에서 항공편 탐색 이미지 내부에서 **Order** 버튼을 클릭합니다. 이미 지 인식 테스트 개체 추가 대화 상자가 열립니다.




- d. 이미지 편집기에서(대화 상자의 가운데) 전체 Order 버튼을 포함하도록 개체 선택 상자를 끕니다.



- e. 컨트롤에서 클릭할 위치 지정 옵션에서 가운데를 선택합니다.
- f. 저장을 클릭하여 개체 저장소에 개체를 추가합니다.
이미지 인식 개체가 개체 저장소에서 최상위 개체로 추가됩니다.



- g. 개체 저장소 관리자의 개체 속성 창(오른쪽)에서 개체 이름을 **InsightObject**에서 **Promotion Order**로 바꿉니다.


- h. 도구 모음에서 저장  을 클릭하여 개체 이름을 변경하고 개체 저장소를 저장합니다.

이미지 인식 이미지를 포함한 개체 저장소를 만들었으므로 이제 이러한 개체를 테스트에서 사용할 수 있습니다. "[연습 8c: 테스트에서 이미지 인식 개체 사용](#)"(134페이지)으로 이동하여 이미지 인식 개체를 테스트에서 사용하고 테스트를 실행하십시오.


연습 8c: 테스트에서 이미지 인식 개체 사용

"[연습 8b: 개체 저장소에 이미지 인식 개체 추가](#)"(131페이지)에서 응용 프로그램의 개체(이미지)에 대한 이미지 인식 개체를 만들었습니다. 이 연습에서는 테스트에서 이러한 개체를 사용하고 이미지 인식 개체 식별을 사용할 때 테스트를 실행하는 방법을 알아봅니다.

1. UFT를 시작합니다.

- a. UFT가 열려 있지 않으면 "[모든 테스트에 대한 솔루션 만들기](#)"(25페이지)에 설명된 대로 엽니다. WPF Add-in이 열렸는지 확인합니다.
- b. 도구 모음에서 열기 드롭다운 화살표  열기 를 선택하고 테스트 열기를 선택합니다. 테스트 열기 대화 상자가 열립니다.
- c. 새 테스트 대화 상자에서 **C:%HOMEPATH%\Unified Functional Testing**에 저장된 **이미지 인식** 테스트로 이동하고 열기를 클릭합니다.
테스트가 문서 창에서 개별 탭으로 열립니다.

2. 이미지 인식 개체 저장소를 테스트와 연결합니다.

- a. 솔루션 탐색기에서 **Action1** 노드를 마우스 오른쪽 버튼으로 클릭하고 **수행에 저장소 연결**을 선택합니다. 공유 개체 저장소 열기 대화 상자가 열립니다.
- b. 공유 개체 저장소 열기 대화 상자에서 **C:%HOMEPATH%\Unified Functional Testing\Tutorial_Object Repositories**에 저장된 **Insight.tsr** 개체 저장소 파일로 이동하고 열기를 클릭합니다.
이제 **Insight.tsr** 파일이 솔루션 탐색기에서 **Action1** 노드의 하위 노드로 표시되며 해당 개체를 테스트에서 사용할 수 있습니다.
- c. 저장  을 클릭하여 변경 사항을 저장합니다.

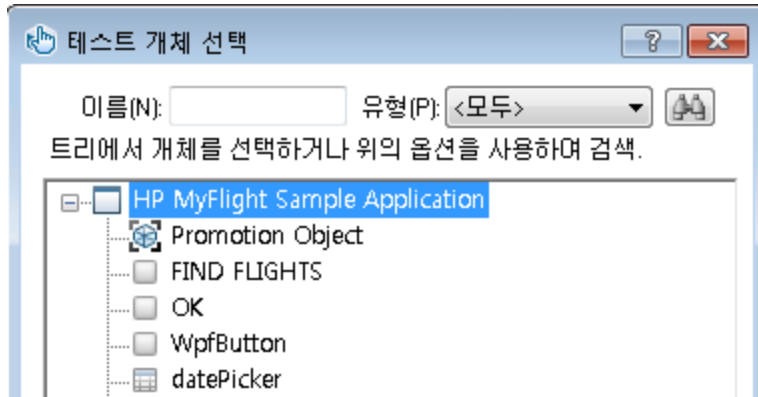
3. 로그인 단계를 테스트에 추가합니다.

"[연습 3a: 키워드 보기에서 로그인 수행에 단계 추가](#)"(45페이지)의 설명대로 항공편 예약 응용 프로그램에 로그인하는 단계를 테스트에 추가합니다.

4. 테스트에 이미지 인식 개체를 추가합니다.

- a. 키워드 보기의 **확인** 버튼 단계 아래에서 항목 열을 클릭하고 드롭다운 목록에서 **저장소의 개체**

- 를 선택합니다. 테스트 개체 선택 대화 상자가 열립니다.
- b. 테스트 개체 선택 대화 상자의 테스트 개체 트리에서 **Promotion Order** 노드를 선택합니다.



- c. **확인**을 클릭하여 단계를 테스트에 추가합니다.
UFT에서는 이미지 인식 개체에 대한 새 단계를 테스트에 추가합니다.

5. 테스트를 실행하고 실행 결과를 봅니다.

- a. 기록 > 기록 및 실행 설정을 선택합니다. 기록 및 실행 설정 대화 상자가 열립니다.
- b. **Windows 응용 프로그램** 탭에서 기록 및 실행 장소 옵션을 선택합니다.
- c. 기록 및 실행 장소 옵션에서 **아래 지정된 응용 프로그램** 옵션을 선택합니다. 이전 실행 세션의 응용 프로그램 세부 정보가 저장되어야 합니다.

참고: 응용 프로그램 세부 정보를 다시 입력해야 하면 "[연습 4a: 테스트 실행\(70페이지\)](#)에서 필요한 세부 정보를 참조하십시오.

- d. **확인**을 클릭하여 변경 사항을 저장하고 설정을 적용합니다.
- e. 도구 모음에서 실행 버튼 을 클릭합니다. 실행 대화 상자가 열립니다.
- f. 실행 대화 상자의 **결과 위치** 탭에서 새 실행 결과 폴더를 선택합니다. 기본 폴더 이름을 그대로 사용합니다.
- g. **실행**을 클릭하여 테스트 실행을 시작합니다.

중요: 테스트 실행을 수행하기 전에 항공편 예약 응용 프로그램이 닫혔는지 확인합니다.

- UFT에서 항공편 예약 응용 프로그램을 열고 테스트 단계를 순서대로 실행합니다. UFT에서는 이미지 인식 개체가 포함된 단계를 실행할 때 이미지 인식 개체 식별 메커니즘을 사용하여 개체를 찾습니다.
- 테스트 실행이 완료되면 실행 결과가 열립니다.
- h. 테스트 흐름에서 **Promotion Order.Click** 노드를 선택합니다. 실행 결과에 단계 요약이 표시됩니다.
 - i. 단계 세부 정보에서 UFT가 테스트 실행 중에 이미지 인식 개체 식별을 사용하여 식별한 개체를 확인할 수 있습니다.

실행 시간

2015-09-24 14:29:15

테스트 개체

 InsightObject: "Promotion Object"


저장소

Local

개체 경로

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

컨텍스트 정보

 HP MyFlight Sample Application

WpfWindow

4부: 자동화 API 테스트 만들기 및 실행

응용 프로그램의 사용자 인터페이스 테스트 이외에 비 GUI(서비스) 레이어를 테스트하여 응용 프로그램을 실행하는 API가 제대로 작동하는지 확인해야 합니다. UFT를 사용하면 자동으로 이 작업을 수행하는 자동화 API 테스트를 만들 수 있습니다.

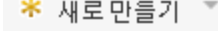
API 테스트를 만들 때 응용 프로그램의 API 프로세스를 개별적으로 테스트하는 테스트 단계로 구성된 전체 테스트 흐름을 만듭니다. 이러한 테스트 단계에 대한 입력 및 검사점 속성을 제공하면 UFT가 응용 프로그램에서 테스트를 실행합니다. 테스트 실행이 완료되면 결과를 확인하여 응용 프로그램이 어떻게 작동하는지 확인할 수 있습니다.

이 부분에서는 응용 프로그램 및 서비스의 다양한 유형에 대한 API 테스트를 만드는 방법을 알아봅니다. 이 섹션의 내용은 다음과 같습니다.

- 1단원: API 테스트 만들기138
- 2단원: 단순 API 테스트 단계 만들기 139
- 3단원: 표준 작업을 사용하여 API 테스트 단계 만들기144
- 4단원: API 테스트 단계 매개 변수화 150
- 5단원: API 테스트 실행 171
- 6단원: 웹 서비스의 API 테스트 만들기 및 실행 176
- 7단원: REST 서비스의 API 테스트 만들기 및 실행 188
- 8단원: 웹 응용 프로그램 서비스(WADL)의 API 테스트 만들기 및 실행 202

1단원: API 테스트 만들기

응용 프로그램 서비스 레이어의 테스트 콘텐츠를 만들기 전에 테스트를 만들고 테스트 구조를 만들어야 합니다.

1. UFT 도구 모음에서 **새로 만들기** 버튼 아래쪽 화살표  * 새로 만들기 를 클릭하고 **새 테스트**를 선택합니다.
2. 새 테스트 대화 상자에서 **API 테스트**를 선택합니다.
3. 테스트에 대해 다음 세부 정보를 입력합니다.

이름: Basic

위치: 기본적으로 UFT에서는 문서를 C:\%HOMEPATH%\My Documents\Unified Functional Testing에 저장합니다. 이 단원에서는 이 경로를 수정할 필요가 없습니다.

4. **만들기**를 클릭합니다.

테스트 흐름에 대한 탭이 있는 빈 테스트가 캔버스에서 열립니다(이름: **Basic**). 테스트 흐름 캔버스 내부에는 빈 테스트 흐름이 있습니다.

이 테스트가 솔루션 탐색기 창에서 **Untitled Solution** 노드의 하위 노드로도 표시됩니다. (이는 테스트가 생성될 때 사용되는 일반적인 솔루션 이름이지만 이름이 지정된 솔루션은 아닙니다.)

API 테스트 설계를 시작할 준비가 되었습니다. "[2단원: 단순 API 테스트 단계 만들기\(139페이지\)](#)"로 이동하여 API 테스트 단계를 만드는 방법을 알아보십시오.

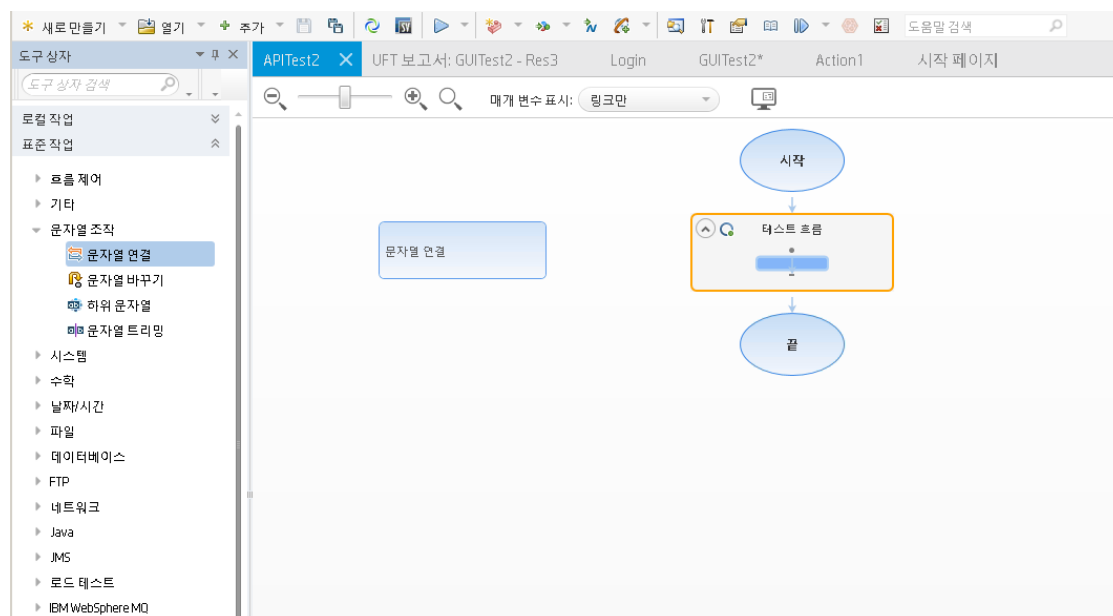
2단원: 단순 API 테스트 단계 만들기

UFT API 테스트에서 테스트 만들기 프로세스는 시각적인 프로세스입니다. 테스트 단계는 전체 마스터 테스트 흐름을 표시하는 캔버스에 표시됩니다.

실제 테스트 단계 만들기는 다음 두 기본 부분으로 구성됩니다.

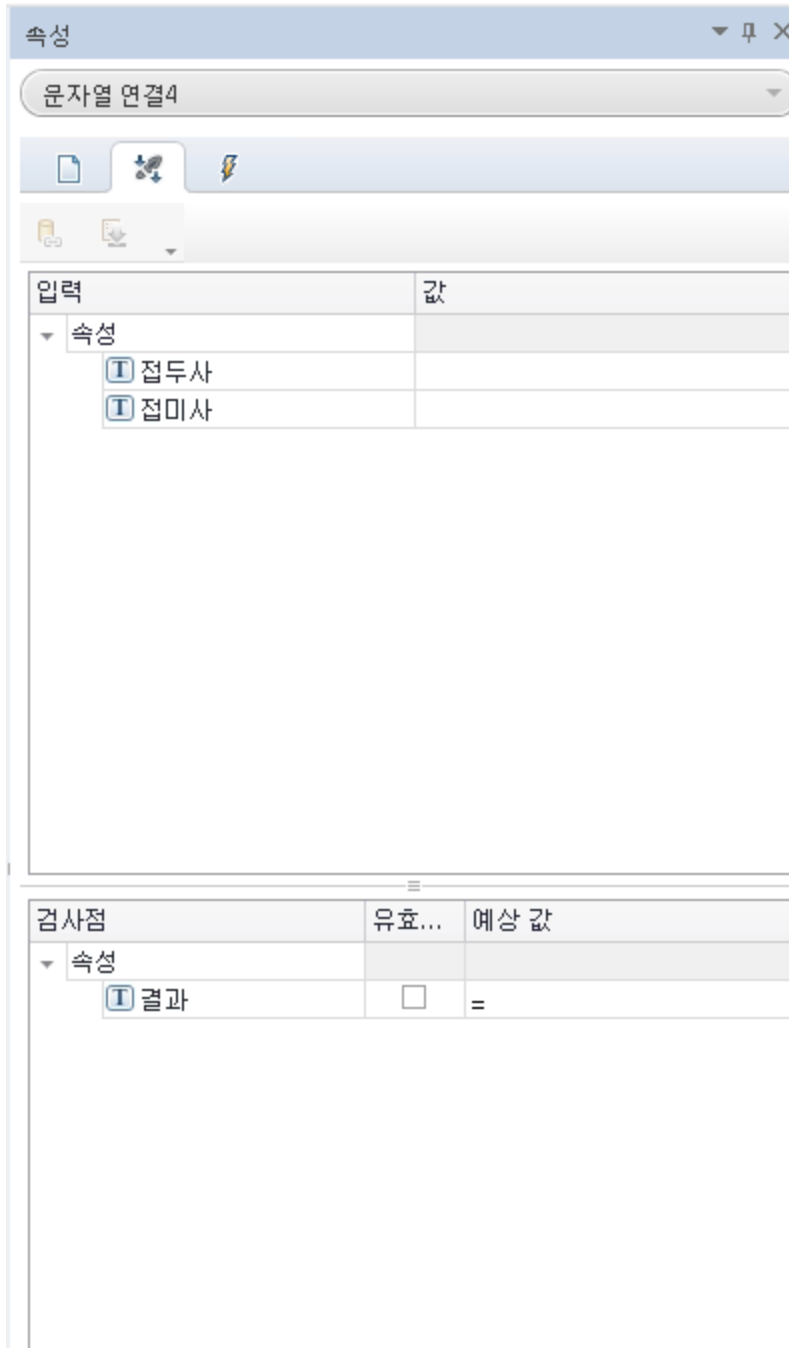
1. 해당 작업을 캔버스의 테스트 흐름으로 끌어서 테스트 단계 만들기

도구 상자 창에는 테스트에서 사용할 수 있는 모든 작업이 들어 있습니다. 캔버스에 포함된 작업 목록에서 작업을 캔버스로 끌고 임의 위치에서 테스트 흐름에 추가할 수 있습니다.



2. 단계 속성 추가(단계의 입력 및 검사점)

테스트 단계를 캔버스로 끌면 캔버스에서 단계를 선택할 때마다 단계 속성이 속성 창에 표시됩니다.



각 단계에는 두 가지 속성이 있습니다.

입력	UFT에서는 이러한 속성을 사용하여 작업이 단계를 실행하는 데 필요한 값을 설정합니다.
검사점	단계가 제대로 실행되는지(또는 제대로 실행되지 않는지) 확인하기 위해 단계가 실행된 후에 이



	<p>값을 실제 값과 비교합니다.</p> <p>테스트를 실행할 때 검사점 속성은 선택 사항입니다.</p>
--	--

이 단원에서는 이러한 기본 기능을 사용하여 기본 테스트를 만듭니다.

1. UFT를 시작하고 Book Flights 테스트를 엽니다.


- a. UFT가 현재 열려 있지 않으면 "[모든 테스트에 대한 솔루션 만들기](#)"(25페이지)에 설명된 대로 엽니다.
- b. 시작 페이지의 최근에 사용한 테스트/구성 요소 영역에서 **기본**을 클릭합니다.
기본 테스트가 문서 창에서 열립니다.

2. 캔버스에 작업을 추가하고 해당 속성을 정의합니다.

- a. 도구 모음에서 **도구 상자 버튼**  을 클릭합니다. 도구 상자 창이 열리고 여러 작업 범주가 표시됩니다.
- b. 도구 상자 창에서 **문자열 조작** 노드를 확장합니다.
- c. 문자열 조작 작업 목록에서 **문자열 연결** 작업을 캔버스에 끕니다.
새 블록이 **문자열 연결** 캔버스에 추가되고 테스트 단계 속성에 대한 속성 창이 열립니다.
- d. 속성 창에서 **입력/검사점** 탭  을 선택합니다.
- e. 입력/검사점 탭의 **입력** 섹션(위쪽 부분)에서 단계 속성에 대해 다음 값을 입력합니다.
 - o **A:** Hello(뒤에 공백 있음)
 - o **B:** World.
 - o **검사점:** Hello world.

3. 캔버스에 다른 작업을 추가하고 해당 속성을 정의합니다.

입력 및 검사점 속성 정의 이외에 테스트 단계에 대한 다른 속성을 정의할 수 있습니다.

- a. 도구 상자 창에서 **문자열 조작** 노드를 확장합니다.
- b. 문자열 조작 작업 목록에서 **문자열 바꾸기** 작업을 두 번 클릭하여 캔버스에 추가합니다. **입력/검사점** 탭이 속성 창에서 열리고 단계에 대한 입력 및 검사점 속성이 표시됩니다.
- c. 속성 창에서 **일반** 탭  을 엽니다.
- d. 이름 속성 행에서 Replace String을 Change Text로 변경하고 **Enter**를 누릅니다. 캔버스에서 단계 이름이 **Change Text**로 변경됩니다.
- e. 입력/검사점 탭을 엽니다.

f. 입력/검사점 탭에서 속성에 대해 다음 값을 입력합니다.

- 원본 문자열: Hello world.
- 검색 문자열: Hello
- 바꾸기 문자열: Goodbye
- 대/소문자 구분: False


4. **Change Text 단계에 검사점 속성을 추가합니다.**

- a. 입력/검사점 탭의 검사점 섹션에 있는 **결과 행에서 유효성 검사 열의 확인란**을 선택합니다. 그러면 이 단계에 대한 검사점이 사용하도록 설정됩니다.
- b. **예상 값 열**에 예상 결과 Goodbye world를 입력합니다.

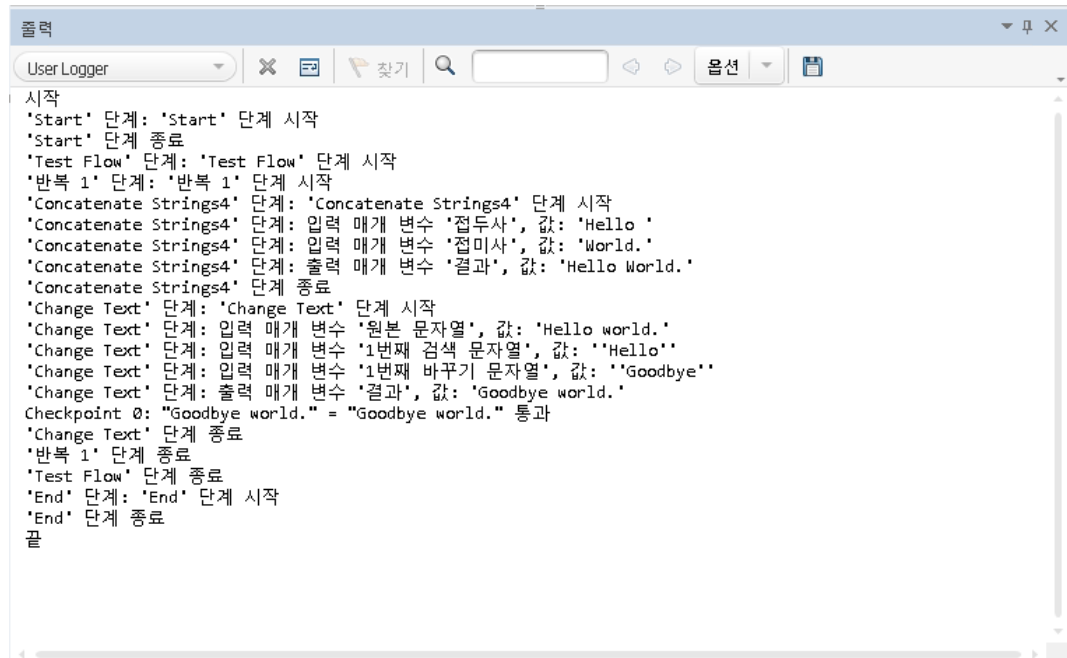
5. **테스트를 저장합니다.**

파일 > 저장을 선택합니다.

6. **테스트를 실행하고 결과를 확인합니다.**

- a. 도구 모음에서 **실행** 버튼  을 클릭합니다. 실행 대화 상자가 열립니다.
- b. 실행 대화 상자에서 **옵션**을 클릭하여 대화 상자를 확장합니다.
- c. **결과 위치** 탭에서 **임시 실행 결과** 폴더 옵션을 선택합니다.
- d. **실행**을 클릭하여 테스트를 컴파일 및 실행합니다.

UFT에서 테스트를 실행하는 동안 출력 창에 컴파일 로그가 표시됩니다. 컴파일 로그에 표시된 줄에는 이전 단계에서 입력한 입력 및 검사점 속성 값이 표시될 수 있습니다.



테스트 실행이 완료되면 실행 결과가 열립니다.

- e. 실행 결과에서 **테스트 흐름**을 선택하여 테스트에서 단계를 표시합니다.
- f. 테스트 단계 트리에서 **문자열 연결** 단계 노드를 선택합니다. 실행 결과에 단계 요약이 표시됩니다.
- g. 단계 요약 영역에서 **Concatenate Strings.xml** 링크를 클릭합니다. 개별 탭이 문서 창에서 열립니다.
- h. 새 탭에서 **접두사** 및 **접미사** 행을 확인합니다. UFT에서 이전 단계에서 제공한 것과 동일한 입력 및 출력 값을 사용하여 단계를 실행했음이 표시됩니다.
- i. 테스트 단계 트리에서 **Change Text** 단계를 선택합니다.
- j. 단계 요약 영역에서 **Change Text.xml** 링크를 클릭합니다.
- k. 새 탭에서 이 단계에 사용한 입력 속성을 다시 확인합니다.
- l. Change Text 단계에서 **검사점** 단계를 선택합니다.
- m. 단계 요약 영역에서 **Checkpoint.xml** 링크를 클릭합니다.
- n. 새 탭에서 테스트 단계 실행에 대한 세부 정보를 확인합니다. 결과에 통과된 검사점(녹색 확인 표시 있음)과 이전 단계에서 입력한 예상 문자열이 표시되는지 확인합니다.
- o. 결과 검토를 마치면 실행 결과가 있는 탭을 닫습니다.

3단원: 표준 작업을 사용하여 API 테스트 단계 만들기

API 테스트를 만들어 테스트를 통해 응용 프로그램의 비 GUI(서비스) 레이어가 제대로 작동하는지 확인합니다. 응용 프로그램을 실행하는 표시되지 않는 프로세스는 데이터베이스 호출, 웹 서비스 호출, 프로그램 열기, 웹을 통해 메시지 보내기 등의 다양한 작업이 될 수 있습니다.

테스트 만들기를 지원하기 위해 UFT에서는 테스트 단계 설계에 사용할 다양한 표준 API 작업을 제공합니다. 이 단원에서는 표준 작업을 사용하여 기본 테스트를 만듭니다.

이 단원의 내용은 다음과 같습니다.

- [UFT API 테스트 표준 작업 - 개요](#) 145
- [연습 3a: 표준 작업으로 테스트 만들기](#) 145

UFT API 테스트 표준 작업 - 개요

API 테스트를 만들 때 모든 테스트에 사용되는 여러 표준 작업이 제공됩니다. 이러한 테스트 공유 응용 프로그램 프로세스는 다음과 같습니다.

- 흐름 제어 작업(예: 대기, 중단 및 조건 단계)
 - 문자열 조작 작업(예: 문자열 연결 및 문자열 바꾸기)
 - 파일 시스템을 사용하여 수행된 프로세스에 대한 파일 시스템 작업
 - 데이터베이스 작업
 - FTP 작업
 - 네트워크 작업(예: HTTP 요청 및 SOAP 요청)
 - XML 또는 JSON 문자열/파일과의 상호 작용과 관련된 응용 프로그램 프로세스에 대한 JSON 및 XML 작업
 - 수학 및 날짜/시간 작업
 - 사용자 지정 코드 작업, 프로그램 실행 및 프로그램 종료 작업, 보고서 작업을 포함한 다른 기타 작업
- 다음과 같은 다양한 기술 관련 작업이 있습니다.

- 응용 프로그램에서 사용되는 Java 프로세스를 테스트하는 Java 클래스 호출 작업
- JMS(Java Message Service) 작업
- IBM Websphere MQ 작업
- SAP 서버에 저장된 SAP iDOC 또는 RFC에 액세스하기 위한 SAP 작업
- LoadRunner 스크립트로 변환한 후 HP LoadRunner를 통해 테스트를 실행하도록 돕는 로드 테스트 작업
- UFT, QuickTest Professional, Service Test 또는 LoadRunner에서 GUI 테스트 또는 수행, API 테스트 또는 수행, Virtual User Generator 스크립트를 호출하여 테스트의 일부로 사용할 수 있는 HP 자동화 테스트 도구 작업

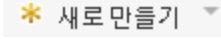
테스트로 가져올 수 있는 사용자 지정 작업도 많으며, 이러한 작업 유형은 이후 단원에서 자세히 설명합니다.

연습 3a: 표준 작업으로 테스트 만들기

"2단원: 단순 API 테스트 단계 만들기"(139페이지)에서는 UFT API 테스트 사용자 인터페이스를 숙지하고 이를 사용하여 테스트 단계를 만들고 실행하는 방법을 알아보았습니다.

이 단원에서는 이 지식을 기반으로 표준 작업을 사용하여 기본 API 테스트를 만듭니다.

1. 새 테스트를 만듭니다.

- 도구 모음에서 새로 만들기 버튼  을 클릭합니다. 새 테스트 대화 상자가 열

립니다.

- b. 새 테스트 대화 상자에서 **API 테스트**를 선택합니다.
- c. 테스트에 대해 다음 세부 정보를 입력합니다.
 - **이름:** Standard
 - **위치:** C:\%HOMEPATH%\My Documents\Unified Functional Testing
- d. **만들기**를 클릭합니다.

캔버스에서 테스트 흐름이 비어 있는 빈 테스트가 열립니다. 테스트가 솔루션 탐색기 창에서 **Solution Untitled** 솔루션의 하위 노드로도 표시됩니다.


2. 테스트 흐름에서 단계를 만듭니다.

이 단계에서는 특정 문자열을 찾고, 바꾸고, 문자열 바꾸기 결과를 파일에 쓰는 응용 프로그램 프로세스의 테스트를 만듭니다.

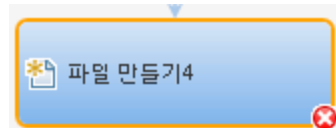
이 테스트에는 다음 세 가지 작업이 필요합니다.

- 문자열 바꾸기
- 결과를 저장할 파일 만들기
- 결과를 파일에 쓰기

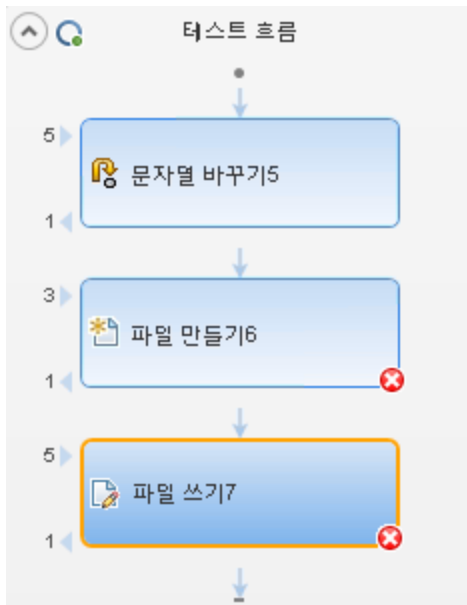
이러한 작업은 모두 UFT 표준 작업과 함께 제공됩니다.

- a. 도구 상자 창의 아직 열려 있지 않으면 UFT 창의 왼쪽 아래에서 **도구 상자** 탭을 클릭하여 도구 상자 창을 엽니다.
- b. 도구 상자 창에서 **문자열 조작** 작업 노드를 확장합니다.
- c. 문자열 조작 작업에서 **문자열 바꾸기** 작업을 캔버스로 끕니다. 새 블록이 캔버스의 테스트 흐름에 추가되고 속성 창에는 문자열 바꾸기 단계에 대한 **입력/검사점** 탭  이 표시됩니다.
- d. 다시 도구 상자에서 **파일작업**을 확장합니다.
- e. 파일 작업에서 다음 작업을 캔버스로 끕니다.
 - **파일 만들기**
 - **파일에 쓰기**


새 블록 두 개가 각 작업에 대해 캔버스에 추가되고 작업의 각 블록이 빨간색 경보 아이콘과 함께 다음과 같이 표시됩니다.



작업을 캔버스에 추가하면 테스트 흐름이 다음과 같이 표시됩니다.




3. 문자열 바꾸기 작업의 속성을 입력합니다.

- a. 캔버스에서 **문자열 바꾸기** 작업을 선택합니다. **입력/검사점 탭**  이 속성 창에서 열립니다.
- b. 입력/검사점 탭에서 단계에 대한 입력 속성을 입력합니다.
 - o **원본 문자열:** Hello world.
 - o **검색 문자열:** Hello
 - o **바꾸기 문자열:** Goodbye

참고: 이 작업에 대한 검사점 속성을 사용할 필요는 없습니다.

4. 파일 만들기 단계의 속성을 입력합니다.


이 단계에서는 이전 단계의 문자열 바꾸기 작업 결과를 쓸 파일을 만들 수 있습니다. 지정된 디렉터리에서 다음 단계에서 사용할 파일을 만듭니다.

- a. 캔버스에서 **파일 만들기** 단계를 선택합니다. 속성 창에서 **입력/검사점 탭**  이 열립니다.
- b. 파일 시스템에서 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 폴더를 엽니다.
- c. 이전 단계에서 언급한 Unified Functional Testing 폴더 내부에 **Tutorial_Files** 폴더를 만듭니다. 이 폴더를 사용하여 다음 단계에서 파일을 만듭니다.

- d. 입력/검사점 탭에서 단계에 대한 입력 속성을 입력합니다.
 - **폴더 경로:** C:\%HOMEPATH%\My Documents\Unified Functional Testing\Tutorial_Files
- 참고:** 찾아보기 버튼을 누르고 폴더로 바로 이동하여 이 폴더를 입력할 수도 있습니다.
- **파일 이름:** Standard Test Result
 - **덮어쓰기:** true
- e. 입력/검사점 탭의 **검사점** 섹션에 있는 **결과** 행에서 **유효성 검사** 열의 확인란을 선택합니다. 기본값을 그대로 유지합니다.
- 이렇게 하면 테스트를 실행하고 실행 결과를 확인하여 파일 만들기 단계가 성공적으로 완료되었는지 확인할 수 있습니다.

5. 파일에 쓰기 단계의 속성을 입력합니다.


이 단계에서는 **문자열 바꾸기** 단계의 문자열 결과를 **파일 만들기** 단계에서 만든 파일에 씁니다.

- a. 캔버스에서 **파일에 쓰기** 단계를 선택합니다. 속성 창에서 입력/검사점 탭  이 열립니다.
 - b. 입력/검사점 탭의 입력 섹션에서 단계에 대한 입력 속성을 입력합니다.
 - **컨텐츠:** Goodbye world.
 - **파일 경로:** 이전 단계에서 만든 파일을 사용합니다. 다음을 수동으로 입력해야 합니다.
C:\%HOMEPATH%\My Documents\Unified Functional Testing\Tutorial_Files\Standard Test Result
 - 다른 모든 속성을 기본값으로 유지합니다.
 - c. 입력/검사점 탭의 **검사점** 섹션에 있는 **결과** 행에서 **유효성 검사** 열의 확인란을 선택합니다(기본값을 그대로 유지).
- 이렇게 하면 테스트를 실행하고 실행 결과를 확인할 때 파일에 쓰기 작업이 성공적으로 완료되었는지 확인할 수 있습니다.

6. 테스트를 저장합니다.

파일 > 저장을 선택합니다.

7. 테스트를 실행합니다.

- a. 도구 모음에서 **실행** 버튼  을 클릭합니다. 실행 대화 상자가 열립니다.
 - b. 실행 대화 상자의 결과 위치 탭에서 **임시 실행 결과 폴더**가 선택되어 있는지 확인합니다.
 - c. **실행**을 클릭하여 테스트를 컴파일 및 실행합니다.
테스트 실행이 완료되면 실행 결과가 열립니다.
 - d. 파일 시스템에서 **C:\%HOMEPATH%\My Documents\Unified Functional Testing\Tutorial_Files\Standard Test Result** 파일을 엽니다.
- 이 파일에서 문서의 첫 번째 줄에 **"Goodbye world"** 문자열이 표시됩니다. 이는 UFT에서 UFT와 함께 제공된 표준 API 작업을 사용하여 임의의 응용 프로그램에서 실행되는 경우와 동일하게 실제 응용 프로그램에서 테스트 단계를 실행했음을 나타냅니다.

자습서

3단원: 표준 작업을 사용하여 API 테스트 단계 만들기

표준 작업을 사용하여 응용 프로그램 프로세스의 기본 테스트를 만들었으므로 이제 ["4단원: API 테스트 단계 매개 변수화"\(150페이지\)](#)로 이동하여 테스트에서 데이터를 사용하여 테스트 단계를 매개 변수화하는 방법을 알아보십시오.

4단원: API 테스트 단계 매개 변수화

"2단원: 단순 API 테스트 단계 만들기"(139페이지) 및 "3단원: 표준 작업을 사용하여 API 테스트 단계 만들기"(144페이지)에서는 필수 입력 및 검사점 값을 수동으로 입력하여 API 테스트 단계의 값을 제공하는 방법을 알아보았습니다.

그러나 다음과 같은 다른 원본의 입력 및 검사점 값을 제공할 수도 있습니다.

- 테스트와 함께 포함되는 데이터 원본(Excel 파일, XML, 데이터베이스 데이터 원본 또는 로컬로 생성된 테이블 등)
- 이전 단계 출력
- 위 모든 항목의 조합: 필수 데이터, 데이터 원본 및 이전 단계 출력 수동 입력

이러한 데이터 원본을 사용하여 단계 입력 및 출력 값을 채우면 응용 프로그램 프로세스 입력을 데이터 원본, 이전 응용 프로그램 프로세스 결과 등에서 가져올 수 있으므로 응용 프로그램 작동 방식을 모방할 수 있습니다.

이 단원에서는 다양한 방법으로 API 테스트 단계를 매개 변수화하는 방법을 설명합니다.

이 단원의 내용은 다음과 같습니다.

• API 테스트 단계 매개 변수화 - 개요	151
• 연습 4a: 데이터 원본에서 테스트 단계 매개 변수화	151
• 연습 4b: 이전 단계의 출력에서 테스트 단계 매개 변수화	161
• 연습 4c: 사용자 지정 식을 사용하여 여러 원본이 있는 테스트 매개 변수화	164

API 테스트 단계 매개 변수화 - 개요

API 테스트에서 테스트 단계 속성의 값을 제공할 때 이러한 값을 제공하는 기본적인 방법은 속성 창에서 값을 수동으로 입력하는 것입니다.

그러나 이 방법을 사용할 때 응용 프로그램의 실제 테스트를 제공할 필요는 없습니다. 대부분 응용 프로그램에서는 내부 API 프로세스가 때때로 동적으로 데이터 원본, 기타 테스트 단계 출력 또는 두 항목에서 모두 해당 정보를 수신합니다.

따라서 UFT에서는 단계 값을 제공(매개 변수화)하는 다양한 방법을 제공합니다.

<p>수동 입력</p>	<p>단계 값을 수동으로 입력할 때 각 단계 및 유형을 선택하거나 입력 및 검사점 속성 값의 적절한 값을 선택합니다.</p> <p>응용 프로그램 속성이 변경될 때마다 테스트의 각 단계 및 각 속성을 업데이트해야 하므로 간편한 테스트 유지 관리를 위해 이 방법은 제공되지 않습니다.</p>
<p>데이터 원본으로 링크</p>	<p>단계 속성을 데이터 원본에 링크하면 UFT에서는 테스트 실행 중에 데이터 원본에서 값을 가져오고 데이터 원본에서 제공된 값을 사용합니다. 데이터 원본에 여러 데이터 집합이 있으면 입력 및 검사점에 대해 서로 다른 값을 제공하여 서로 다른 데이터 입력을 통해 응용 프로그램이 어떻게 작동하는지 확인하는 테스트를 여러 번 반복 실행할 수 있습니다.</p> <p>각 테스트 단계 대신에 데이터 원본 값을 업데이트하면 되므로 이 방법으로 테스트를 더 쉽게 유지 관리할 수 있습니다.</p>
<p>이전 단계 출력으로 링크</p>	<p>단계 속성을 이전 단계의 출력에 링크하면 UFT에서는 단계 출력에서 값을 가져오고 테스트 실행 중에 이러한 값을 사용합니다. 이렇게 하면 응용 프로그램 API 프로세스의 출력이 값을 다른 프로세스에 입력으로 전달할 수 있는 실제 응용 프로그램 동작을 모방할 수 있습니다.</p>
<p>여러 원본으로 링크</p>	<p>응용 프로그램 입력 및 검사점이 다른 단계/프로세스의 정적 문자열, 데이터 및 출력과 같은 여러 원본에서 제공되면 사용자 지정 식을 만들어서 테스트에서 이 작업을 수행할 수 있습니다. 그러면 UFT에서는 사용자 지정 식을 사용하여 값을 제공하고 테스트 실행 중에 값을 사용합니다.</p>

연습 4a: 데이터 원본에서 테스트 단계 매개 변수화

"API 테스트 단계 매개 변수화 - 개요"(151페이지)에서 살펴본 대로 테스트 단계에 값을 제공하는 한 가지 방법은 단계 속성 값을 데이터 원본에 링크하는 것입니다. 그러면 데이터 원본 구조에 따라 여러 가지 값을 사용하여 테스트 단계를 실행할 수 있습니다.

API 테스트에서 여러 가지 유형의 데이터 원본을 추가할 수 있습니다.

- Excel 시트
- XML 파일 또는 스키마
- 데이터베이스
- 로컬로 저장된 데이터 테이블

이러한 항목을 각각 테스트에 추가하여 테스트에서 모든 단계에 사용할 수 있습니다.

이 단계에서는 테스트 단계를 데이터 원본에 링크합니다.


1. UFT를 시작하고 표준 테스트를 엽니다.

- a. UFT가 현재 열려 있지 않으면 "모든 테스트에 대한 솔루션 만들기"(25페이지)에 설명된 대로 엽니다.
- b. 시작 페이지의 **최근 테스트/구성 요소** 영역에서 **표준**을 클릭합니다.
 "3단원: 표준 작업을 사용하여 API 테스트 단계 만들기"(144페이지)에서 만든 표준 테스트가 문서 창에서 개별 탭으로 열립니다. 솔루션 탐색기에서 **Solution Untitled** 솔루션의 하위 노드으로도 표시됩니다.

2. 문자열 연결 단계를 테스트에 추가합니다.

- a. **보기 > 도구 상자**를 선택하여 도구 상자 창을 엽니다.
- b. 도구 상자 창에서 **문자열 조작** 노드를 확장합니다.
- c. 문자열 조작 노드에서 **문자열 연결** 작업을 캔버스의 **문자열 바꾸기** 작업 위로 끕니다.

3. 데이터 원본을 테스트에 추가합니다.

- a. 필요하면 **보기 > 데이터**를 선택하여 데이터 창을 엽니다.
- b. 데이터 창에서 **새 데이터 원본** 버튼  을 클릭하고 **로컬 테이블**을 선택합니다. 새 로컬 테이블 데이터 원본 대화 상자가 열립니다.

? ×

새 로컬 테이블 데이터 원본

데이터 원본 이름: *

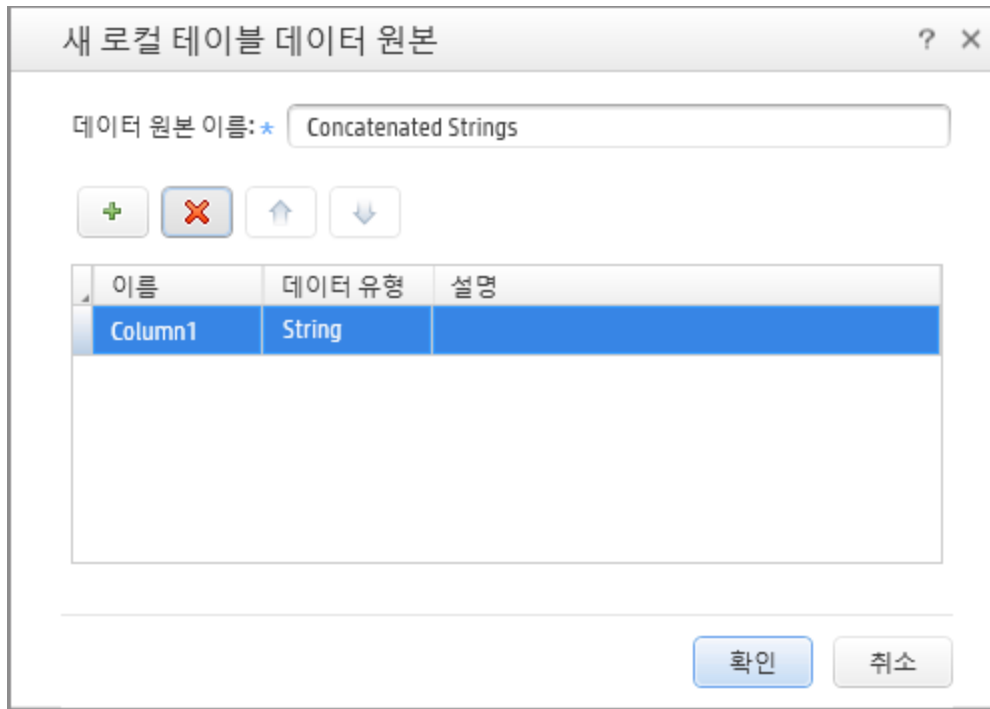
+
×
↑
↓


이름	데이터 유형	설명

확인
취소

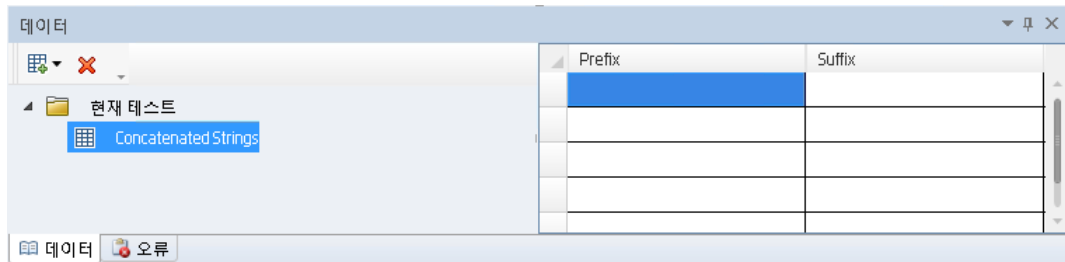
- c. **데이터 원본 이름** 필드에서 테이블 이름을 Concatenated Strings로 지정합니다.

- d. 데이터 원본 이름 필드 아래에서 **추가** 버튼  을 클릭합니다. 새 행이 테이블 표에 추가됩니다.



- e. 표의 **이름** 필드에서 Prefix를 입력합니다. **데이터 유형**을 문자열로 남겨 둡니다.
- f. **추가** 버튼  을 다시 클릭합니다. 두 번째 행이 테이블에 추가됩니다.
- g. 표에서 두 번째 행의 **이름** 필드에 Suffix를 입력합니다. **데이터 유형**을 문자열로 남겨 둡니다.
- h. **확인**을 클릭하여 대화 상자를 닫고 테이블을 테스트에 추가합니다.

테이블이 테스트에 추가되고 데이터 창에 현재 테스트 데이터 원본의 하위 노드로 표시됩니다.



4. 데이터 테이블에 값을 추가합니다.

테스트 단계에서 테스트 단계 값을 사용하려면 데이터 원본에 사용할 수 있는 데이터가 있는지 확인해야 합니다. 로컬로 생성되고 저장된 테이블의 경우 데이터를 추가해야 합니다.

- a. 데이터 창의 **현재 테스트** 노드에서 **Concatenated Strings** 노드를 선택합니다. 데이터 창이 업데이트되어 선택한 데이터 원본의 데이터가 표시됩니다. (현재 테이블에는 데이터가 없습니다.)



- b. 데이터 표에서 **접두사** 열을 클릭합니다.
- c. **접두사** 열에서 Hello를 입력합니다. Hello 뒤에 공백을 두어야 합니다.
- d. **접미사** 열을 클릭하고 World를 입력합니다.
- e. 추가 행을 입력합니다.

접두사	접미사
Welcome	to UFT.
I am running	API tests.

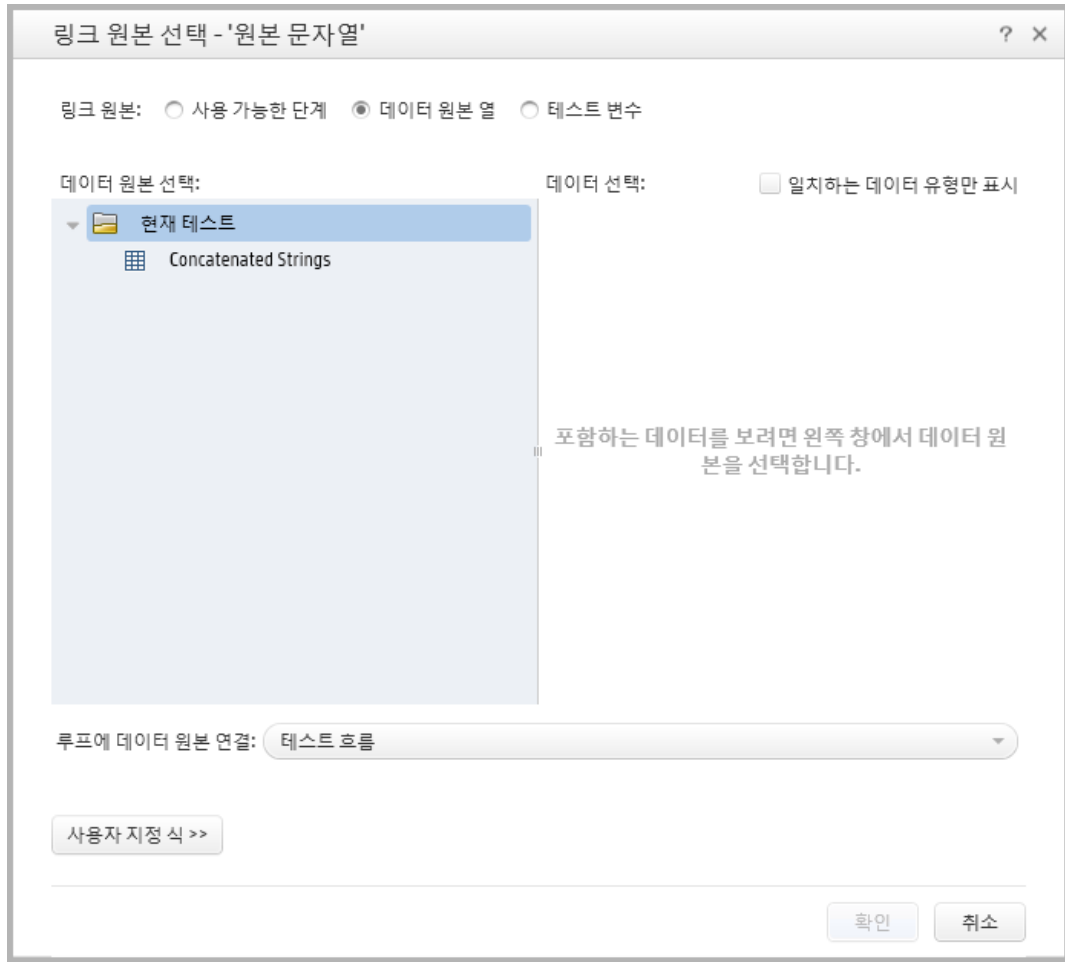
참고: A 열에서 문자열 뒤에 공백을 입력해야 합니다.

5. **테스트 단계를 데이터 원본에 연결합니다.**

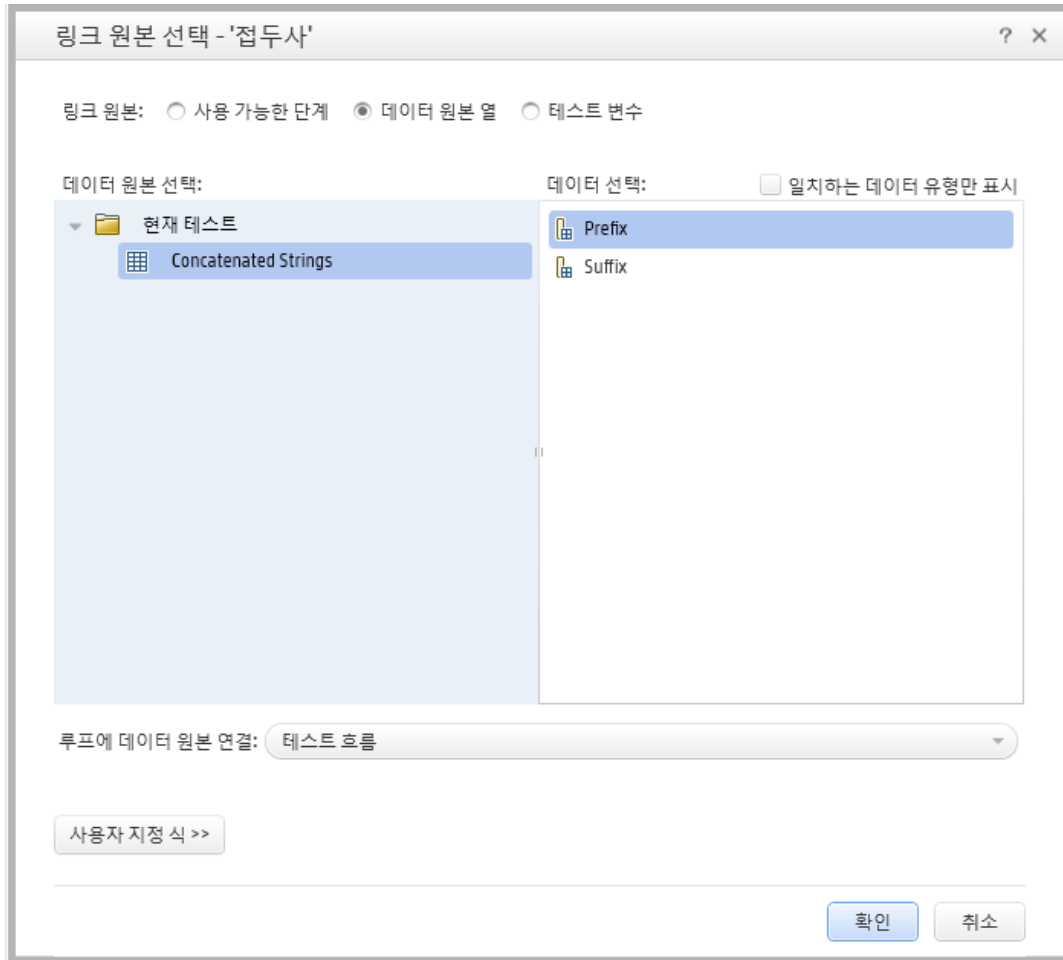
테스트를 실행할 때 데이터 원본 값을 사용하려면 테스트 단계 속성을 데이터 원본에 링크해야 합니다.

- a. 캔버스에서 **문자열 연결** 단계를 선택합니다. **입력/검사점** 탭  이 속성 창에서 열립니다.
- b. 입력/검사점 탭의 입력 섹션에서 **접두사** 속성에 대한 **값** 셀을 클릭합니다.
- c. **값** 셀에서 **데이터 원본으로 링크** 버튼  을 클릭합니다. 링크 원본 선택 대화 상자가 열립니다.

- d. 링크 원본 선택 대화 상자에서 **데이터 원본 열** 라디오 버튼을 선택합니다. 모든 테스트 데이터 원본 목록이 대화 상자 왼쪽에 표시됩니다.



- e. **데이터 원본 선택** 창(왼쪽)에서 **Concatenated Strings** 노드를 선택합니다. (이는 이전 단계에서 만든 데이터 원본입니다.) 데이터 원본 매개 변수 목록이 오른쪽 창에 표시됩니다.

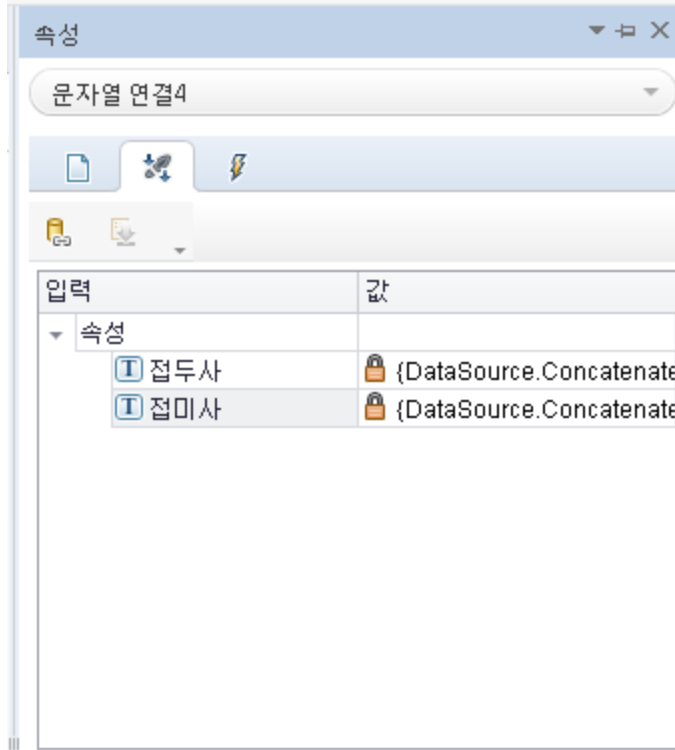


- f. **데이터 선택** 창(오른쪽)에서 **접두사** 노드를 선택하고 **확인**을 클릭합니다.
식을 통해 입력/검사점 탭의 값 열이 업데이트되고 데이터 원본 링크가 표시됩니다. 마우스로 값 열을 가리키면 식이 표시됩니다.

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

- g. 위에 설명된 프로세스를 반복하여 **접미사** 속성을 데이터 테이블의 **접미사** 열에 링크합니다.

작업을 완료하면 입력/검사점 탭에 데이터 테이블 링크가 반영됩니다.

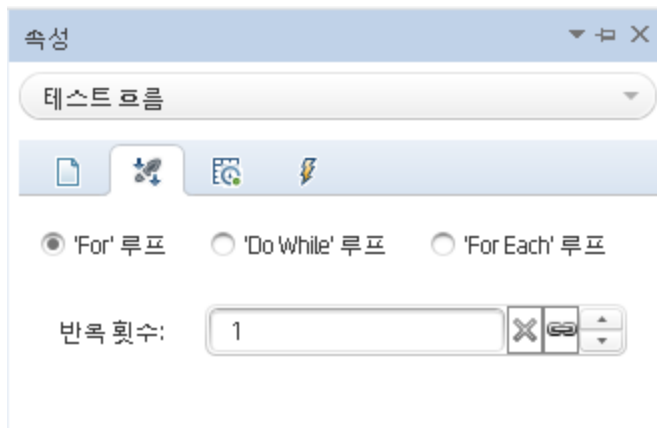


6. 테스트 반복 수를 설정합니다.

지금부터 전체 테스트를 실행하면 데이터 테이블 첫 번째 행의 데이터를 사용하여 한 번의 반복만 실행됩니다.

3개 행을 입력했으므로 테스트를 세 번 반복 실행하여 UFT에서 다른 데이터를 처리하는 방법을 확인해야 합니다.

- a. 캔버스에서 **테스트 흐름**을 선택합니다. **입력 탭**  이 속성 창에서 열립니다.




- b. 입력 탭에서 **'For'** 루프 라디오 버튼을 선택합니다.

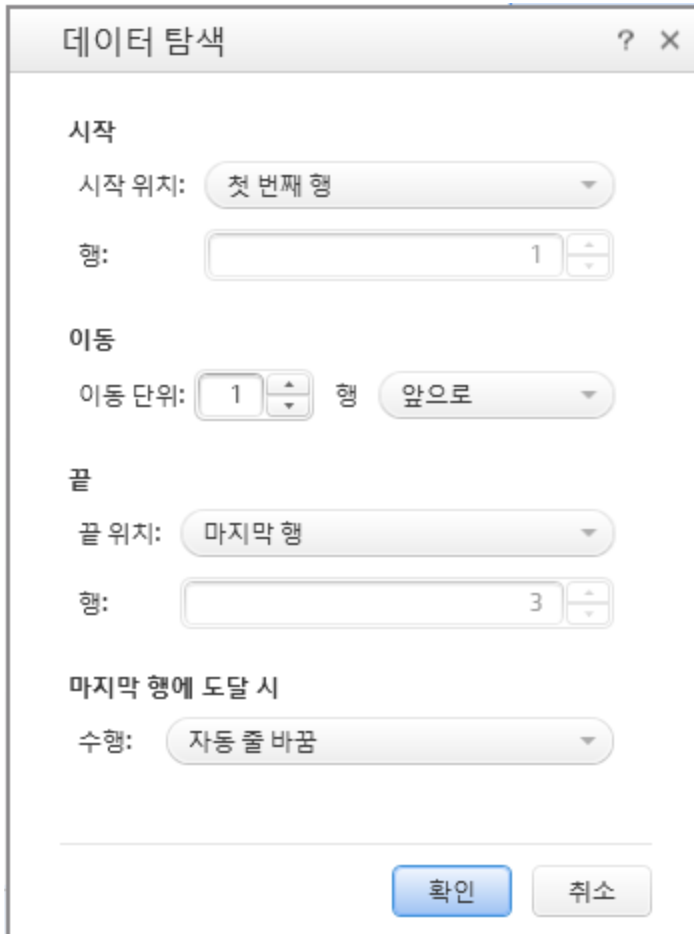
- c. 반복 수에서 반복 수로 3을 입력합니다.

이제 UFT에서 테스트를 실행할 때마다 데이터 테이블의 새 행을 사용하여 3개의 테스트 반복을 실행합니다.

참고: 데이터 원본에 있는 행과 같은 수만큼 반복을 실행할 필요는 없습니다. 그러나 기본적으로 UFT에서는 데이터 테이블의 첫 번째 행에서 시작하고 테스트가 끝날 때까지 각 후속 반복에서 새 행을 사용합니다.

7. 데이터 테이블에 대한 데이터 탐색 정책을 설정합니다.

- a. 캔버스에서 테스트 흐름을 선택합니다. 입력 탭이 속성 창에서 열립니다.
- b. 속성 창에서 데이터 원본 탭  을 선택합니다. 데이터 탐색 표가 열립니다.
- c. 데이터 원본 탭, 데이터 탐색 표의 데이터 원본 이름 열에서 **Concatenated Strings** 데이터 원본을 선택하고 편집을 클릭합니다. 데이터 탐색 대화 상자가 열립니다.




- d. 데이터 탐색 대화 상자에서 다음 속성을 설정합니다.

시작 위치	첫 번째 행
-------	--------

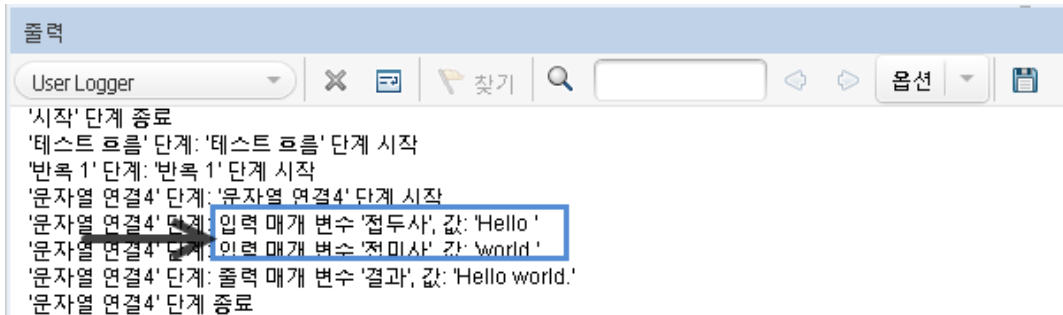
행	1
위로 이동	1 rows Forward
종료 위치	Last row
마지막 행에 도달할 경우	줄 바꿈

e. **확인**을 클릭하여 대화 상자를 닫고 데이터 탐색 정책을 업데이트합니다.

8. 테스트를 실행합니다.

- a. 도구 모음에서 **실행** 버튼  을 클릭합니다. 실행 대화 상자가 열립니다.
- b. 실행 대화 상자에서 **옵션** 막대를 클릭하여 대화 상자를 확장합니다.
- c. **결과 위치** 탭에서 **임시 실행 결과 폴더** 옵션을 선택합니다.
- d. **실행**을 클릭하여 테스트 실행을 시작합니다.

UFT에서는 문자열 연결 작업의 입력으로 데이터 테이블의 값을 사용하여 단계를 순서대로 실행합니다. UFT에서 테스트를 실행하는 동안 UFT가 데이터 테이블에서 가져온 값을 출력 창에서 확인할 수 있습니다.

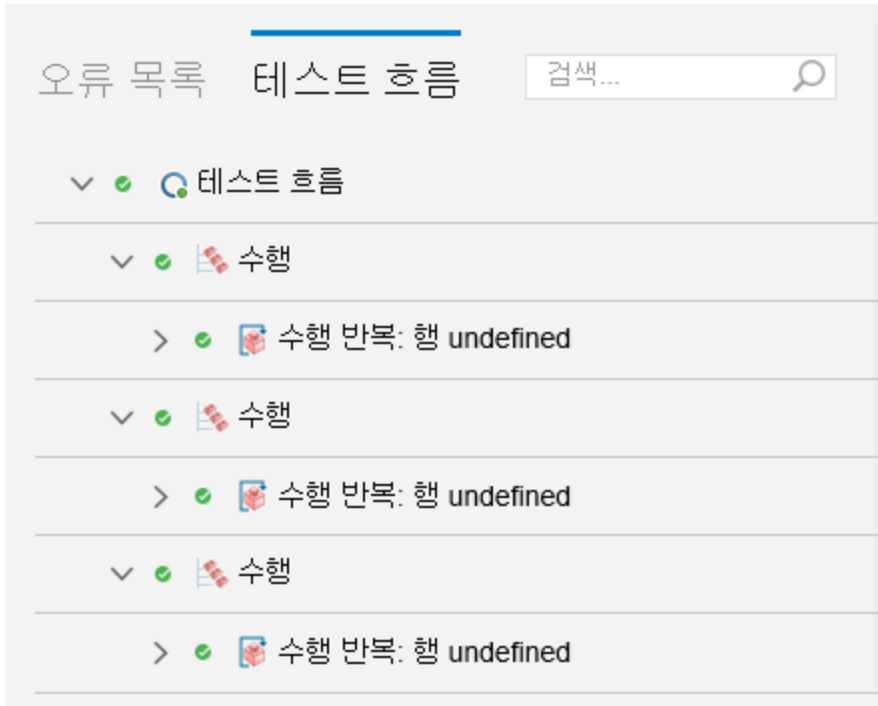


테스트 실행이 완료되면 실행 결과가 열립니다.

9. 실행 결과를 분석합니다.

- a. 실행 결과의 테스트 흐름에서 테스트 결과 트리를 확인합니다.

다음 세 가지 요약 노드가 있는지 확인합니다.



이는 이전에 설정한 세 가지 반복에 해당합니다.

- b. 수행 반복: 행 1 노드를 확장하고 문자열 연결 노드를 선택합니다. 실행 결과에 단계 요약이 표시됩니다.
- c. 캡처된 데이터는 문자열 연결 작업의 입력 값으로 사용된 데이터를 보여 줍니다.

Name	Value
유형	HP.ST.Ext.BasicActivities.ConcatenateStringsActivity
단계 ID	ConcatenateStringsActivity4
메시지	문자열 연결 성공
접두사	'Hello '
접미사	'world.'
결과	'Hello world.'
이름	'문자열 연결 4'
주석	"

사용된 값이 데이터 테이블의 첫 번째 행과 일치합니다.

- d. 반복 2 및 반복 3 문자열 연결 작업에 대해 이 프로세스를 반복합니다. 단계의 캡처된 데이터 일부로 표시된 값이 데이터 테이블의 두 번째 및 세 번째 행과 일치함을 확인합니다.

10. 테스트를 저장합니다.

파일 > 저장을 선택합니다.

테스트 단계를 데이터 원본에 연결하는 방법을 알아보았으므로 이제 "연습 4b: 이전 단계의 출력에서 테스트 단계 매개 변수화"(161페이지)로 이동하여 단계 속성을 이전 단계 출력과 링크하는 방법을 알아보십시오.

연습 4b: 이전 단계의 출력에서 테스트 단계 매개 변수화

"연습 4a: 데이터 원본에서 테스트 단계 매개 변수화"(151페이지)에서는 선택한 단계의 속성 값을 데이터 원본에 링크하는 방법을 알아보았습니다.

그러나 데이터의 속성 값을 제공할 뿐 아니라 이전 단계의 출력에서 속성 값을 가져올 수도 있습니다. 이 단원에서는 이전 단계 출력을 사용하여 단계 값을 링크하는 방법을 학습합니다.


1. UFT를 시작하고 표준 테스트를 엽니다.

a. UFT가 현재 열려 있지 않으면 "모든 테스트에 대한 솔루션 만들기"(25페이지)에 설명된 대로 엽니다.


b. 시작 페이지의 **최근 테스트/구성 요소** 영역에서 **표준**을 클릭합니다.

"3단원: 표준 작업을 사용하여 API 테스트 단계 만들기"(144페이지)에서 만든 표준 테스트가 문서 창에서 개별 탭으로 열립니다. 솔루션 탐색기에서 Solution Untitled 솔루션의 하위 노드로도 표시됩니다.

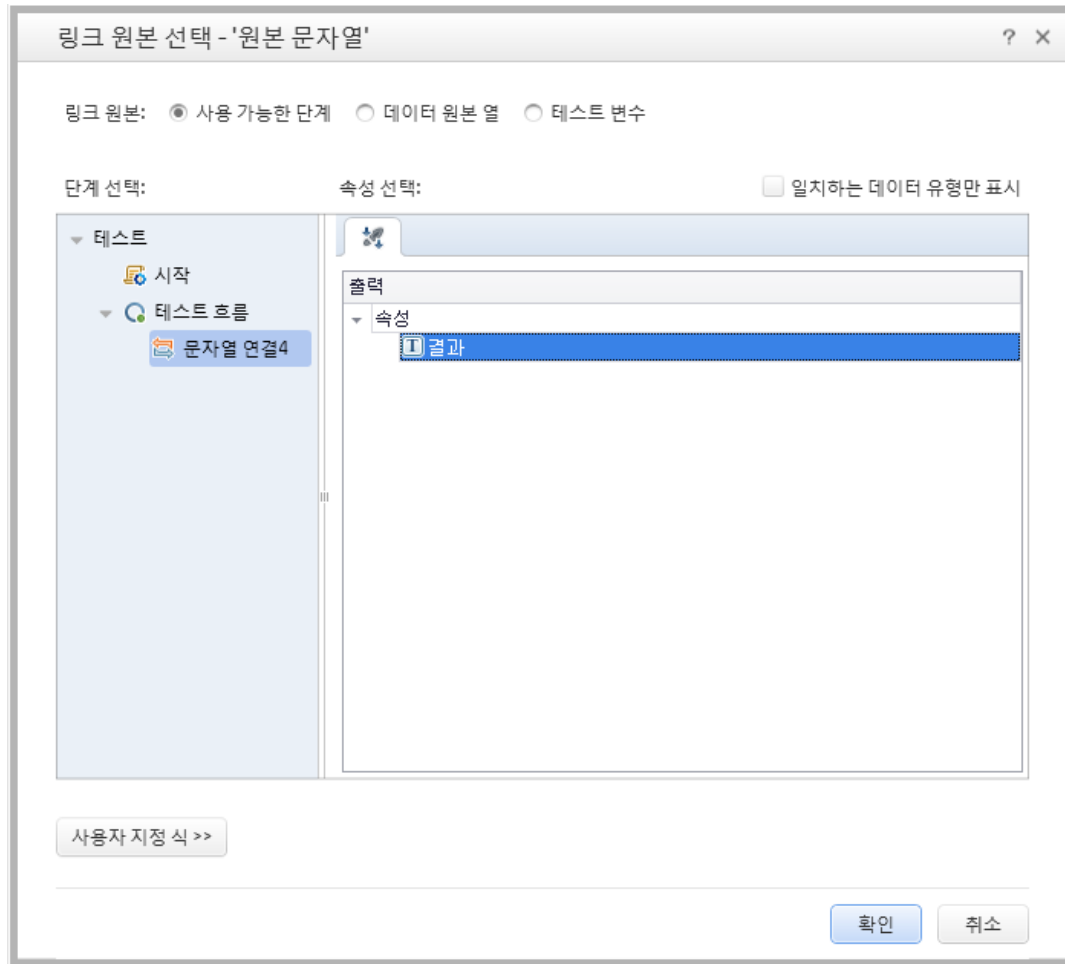
2. 문자열 바꾸기 단계의 속성을 문자열 연결 테스트에 연결합니다.

a. 캔버스에서 **문자열 바꾸기** 단계를 선택합니다. 속성 창에서 **입력/검사점**  탭이 열립니다.

b. 입력/검사점 탭의 **입력** 영역에서 **원본 문자열** 행을 선택합니다.

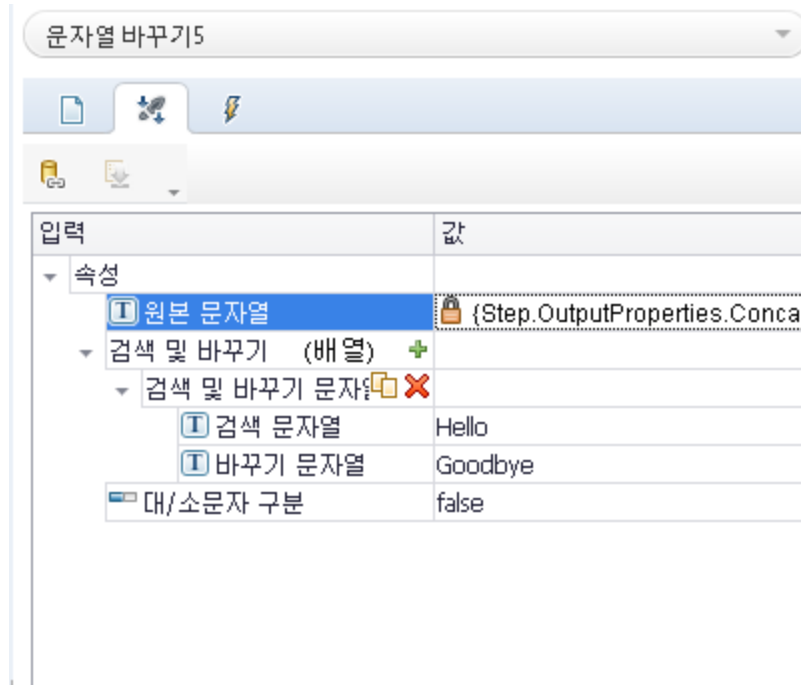
c. 원본 문자열 행의 **값** 셀에서 **데이터 원본으로 링크** 버튼  을 클릭합니다. 링크 원본 선택 대화 상자가 열립니다.

- d. 링크 원본 선택 대화 상자에서 **사용 가능한 단계** 라디오 버튼을 선택합니다. 대화 상자 왼쪽에 사용 가능한 단계 목록이 표시됩니다.



- e. **단계 선택** 창(왼쪽)에서 **문자열 연결**을 선택합니다. 문자열 연결 작업에 대한 출력 속성 목록이 표시됩니다.
- f. **속성 선택** 창(오른쪽)에서 **결과** 행을 선택합니다.
- g. **확인**을 클릭하여 속성을 링크하고 대화 상자를 닫습니다.



원본 문자열 행(속성 창)의 값 열이 업데이트되어 이전 단계 출력의 링크가 반영됩니다.



마우스로 값 열을 가리키면 전체 링크 설명을 확인할 수 있습니다.

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

3. 테스트를 실행하고 링크된 단계의 결과를 확인합니다.

- a. 캔버스에서 테스트 흐름을 선택합니다. 입력 탭  이 속성 창에서 열립니다.
- b. 입력 탭에서 'For' 루프 옵션이 선택되었는지 확인합니다.
- c. 반복 수에서 1을 입력합니다.
- d. 도구 모음에서 실행 버튼  을 클릭합니다. 실행 대화 상자가 열립니다.
- e. 실행 대화 상자의 결과 위치 탭에서 임시 실행 결과 폴더 옵션이 선택되었는지 확인합니다.
- f. 실행을 클릭하여 테스트 실행을 시작합니다.
 UFT에서는 문자열 바꾸기 작업의 입력으로 문자열 연결 단계의 출력을 사용하여 테스트 단계를 실행합니다.
 테스트 실행이 완료되면 실행 결과가 열립니다.
- g. 실행 결과에 테스트 흐름을 표시합니다.
- h. 요약 노드에서 문자열 바꾸기 노드를 찾아서 선택합니다. 실행 결과에 문자열 바꾸기 단계의 세부 정보가 표시됩니다.
- i. 단계 세부 정보에서 이 테스트 실행에 사용된 원본 문자열을 확인합니다.

Step Properties

Name	
유형	HP.ST.Ext.BasicActivities.ReplaceStringAc
단계 ID	ReplaceStringActivity5
원본 문자열	'Hello world.'
1번째 검색 문자열	'Hello'
1번째 바꾸기 문자열	'Goodbye'

원본 문자열은 "연습 4a: 데이터 원본에서 테스트 단계 매개 변수화"(151페이지)에서 입력한 문자열 연결 작업의 출력입니다.

4. 테스트를 저장합니다.

파일 > 저장을 선택합니다.

테스트 단계를 서로 링크하는 방법을 알아보았으므로 이제 "연습 4c: 사용자 지정 식을 사용하여 여러 원본이 있는 테스트 매개 변수화"(164페이지)로 이동하여 수동으로 값을 입력하고, 데이터 원본에 링크하고, 이전 단계의 출력에 링크하여 단계를 매개 변수화하는 방법을 알아보십시오.

연습 4c: 사용자 지정 식을 사용하여 여러 원본이 있는 테스트 매개 변수화

이전 연습에서는 테스트 단계 속성 값을 데이터 원본 또는 이전 단계 출력에 링크하는 방법을 살펴보았습니다.

그러나 정적 값 수동 입력, 데이터 원본 및/또는 이전 단계 출력과 같은 다양한 위치에서 단계 값을 가져오는 경우가 있습니다. 이러한 경우 여러 원본에 링크할 사용자 지정 식을 만들 수 있습니다.

이 연습에서는 사용자 지정 식을 만들어서 테스트 데이터 테이블, 정적 텍스트 문자열 수동 입력 및 다른 테스트 단계 출력에서 가져오는 데이터를 사용하는 문자열 바꾸기 작업의 결과를 파일에 씁니다.

1. UFT를 시작하고 표준 테스트를 엽니다.

- a. UFT가 현재 열려 있지 않으면 "모든 테스트에 대한 솔루션 만들기"(25페이지)에 설명된 대로 엽니다.
- b. 시작 페이지의 최근 테스트/구성 요소 영역에서 표준을 클릭합니다.

"3단원: 표준 작업을 사용하여 API 테스트 단계 만들기"(144페이지)에서 만든 표준 테스트가 문서 창에서 개별 탭으로 열립니다. 솔루션 탐색기에서 **Solution Untitled** 솔루션의 하위 노드으로도 표시됩니다.


2. 파일에 쓰기 테스트 단계의 입력 속성을 링크합니다.

"3단원: 표준 작업을 사용하여 API 테스트 단계 만들기"(144페이지)에서는 다음 세 가지 작업이 있는 테스트를 만들었습니다.

- 문자열 바꾸기 - 문자열을 가져오고 그 일부를 다른 문자열로 바꾸는 경우
- 파일 만들기 - 파일을 만들어서 바뀐 문자열을 쓰는 경우
- 파일에 쓰기 - 바뀐 문자열을 쓴 경우

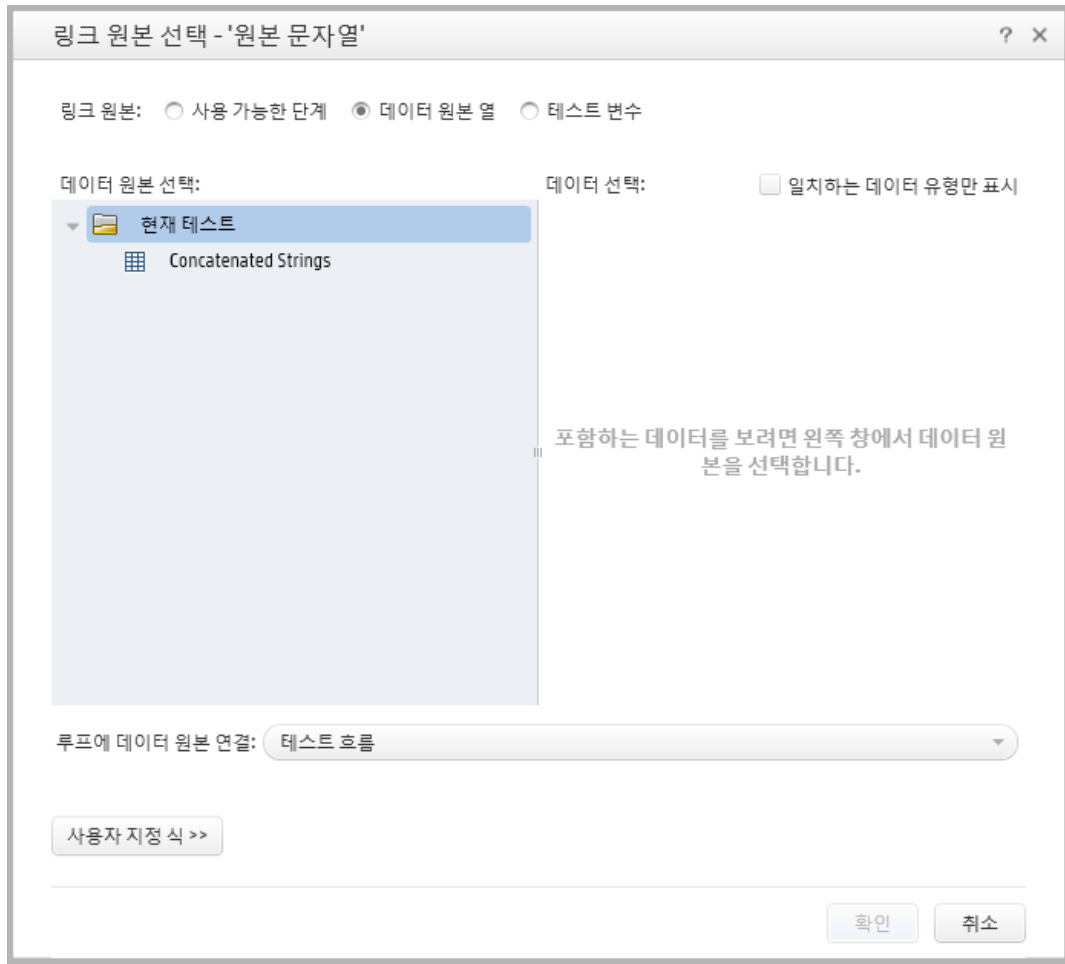
참고: "연습 4a: 데이터 원본에서 테스트 단계 매개 변수화"(151페이지)에서 문자열 연결 단계를 추가했습니다.

이 단계와 다음 단계에서 **파일에 쓰기** 작업을 사용합니다.

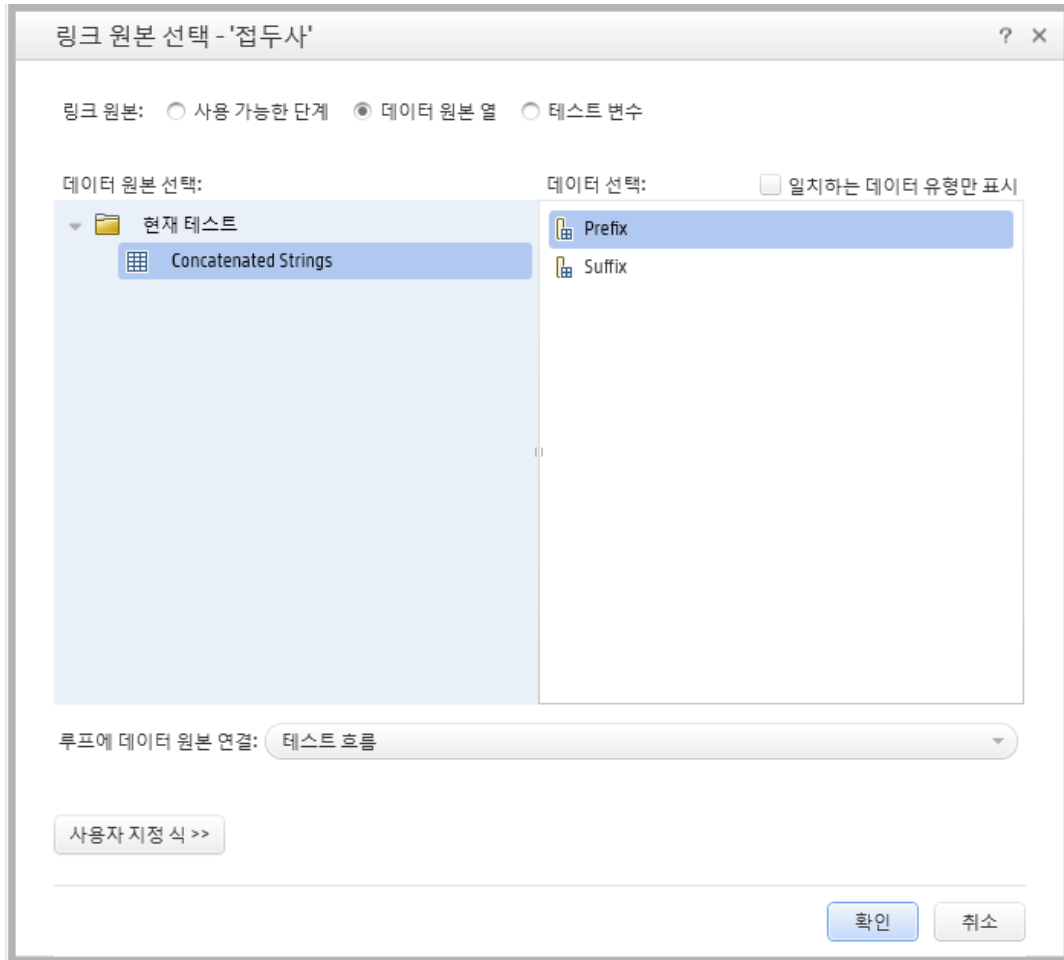
- a. 캔버스에서 **파일에 쓰기** 단계를 선택합니다. **입력/검사점** 탭  이 속성 창에서 열립니다.
- b. 입력/검사점 탭의 **입력** 섹션에서 **컨텐츠** 행을 선택합니다.
- c. 컨텐츠 행의 **값** 열에서 **데이터 원본으로 링크** 버튼을 클릭합니다. 링크 원본 선택 대화 상자가 열립니다.

3. 데이터 테이블에서 컨텐츠 속성 값에 대한 사용자 지정 식의 첫 번째 부분을 만듭니다.

- a. 링크 원본 선택 대화 상자에서 **데이터 원본** 열 옵션을 선택합니다. 링크 원본 선택 대화 상자에 모든 테스트 데이터 원본 목록이 표시됩니다.



- b. **데이터 원본 선택** 창(왼쪽)에서 Concatenated Strings 노드를 선택합니다. 대화 상자에 문자열 연결 데이터 테이블의 데이터 매개 변수 목록이 표시됩니다.



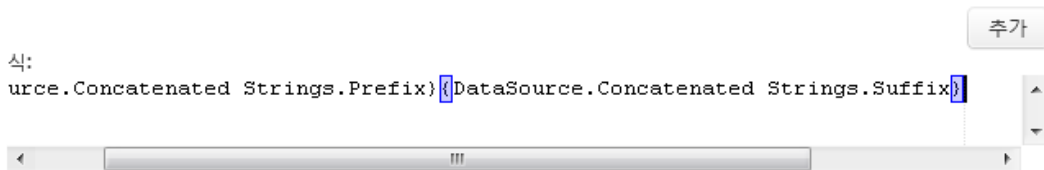
- c. **데이터 선택** 창(오른쪽)에서 점두사 노드를 선택합니다.
- d. 대화 상자 아래쪽에서 **사용자 지정 식** 버튼을 클릭합니다. 대화 상자가 확장되고 식 영역이 표시됩니다.



- e. **접두사** 노드가 선택된 상태에서 식 영역 위에 있는 **추가** 버튼을 클릭합니다. UFT에서 접두사 매개 변수에 대한 식을 다음 식에 추가합니다.



- f. **데이터 선택** 창에서 **접미사** 노드를 선택합니다.
- g. 식 영역에서 **추가** 버튼을 다시 클릭합니다. UFT에서 접미사 매개 변수에 대한 식을 다음 식에 추가합니다.



4. **문자열을 수동으로 입력하여 사용자 지정 식의 중간 부분을 추가합니다.**

사용자 지정 식의 중간 부분에 대해 정적 텍스트 문자열을 수동으로 추가합니다.

- a. 링크 선택 대화 상자의 식 영역에서 **{DataSource.Concatenated Strings.Suffix}** 식 뒤에 공백을 입력합니다.
- b. `was replaced by` 텍스트를 입력하고 그 뒤에 공백을 하나 더 입력합니다.

참고: 이 문자열을 입력한 후에는 추가를 클릭하지 마십시오. 추가 버튼을 클릭하면 UFT에서 창에 선택된 요소를 대화 상자의 위쪽 부분에 추가합니다.

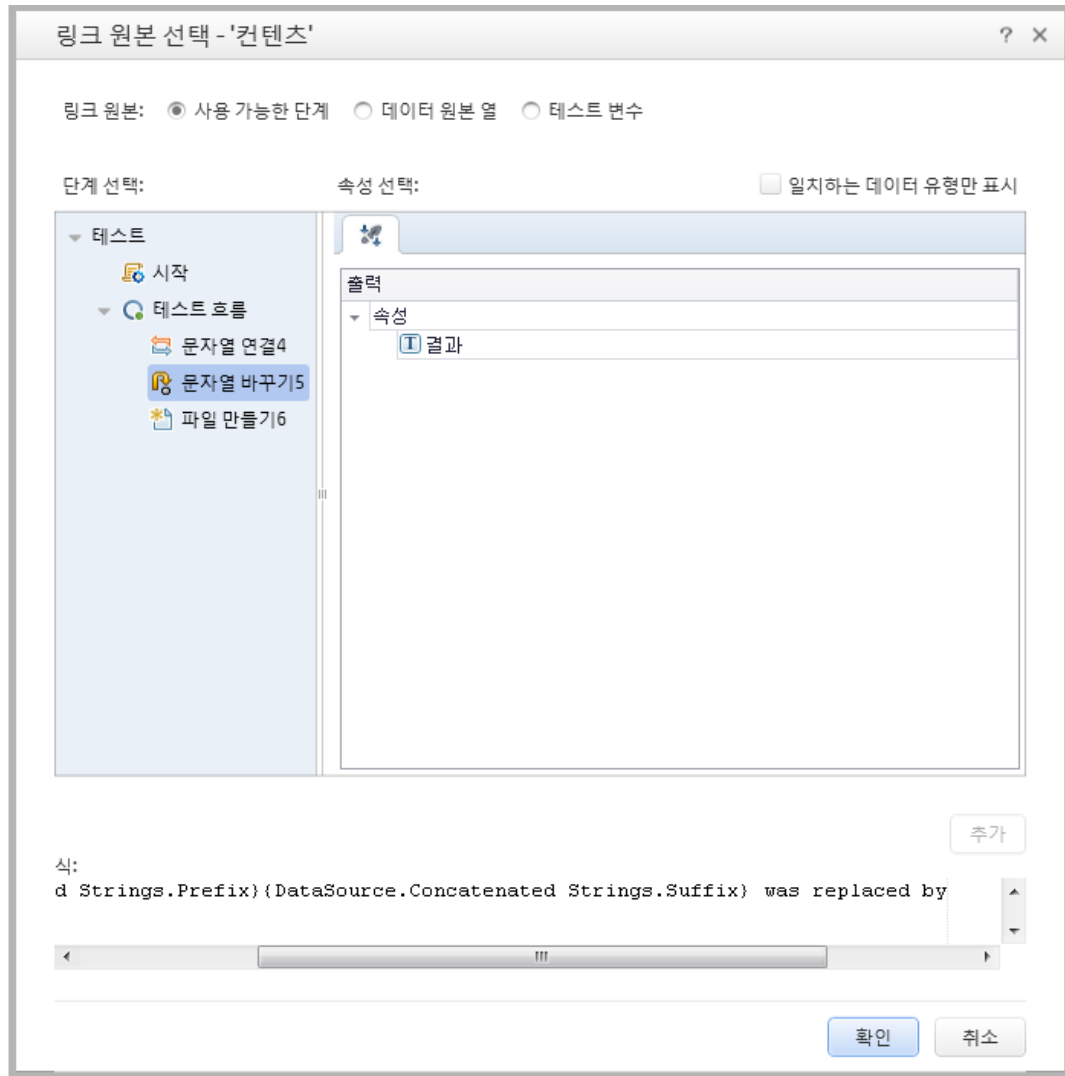
정적 텍스트 문자열을 입력하고 나면 사용자 지정 식이 다음과 같이 표시됩니다.

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

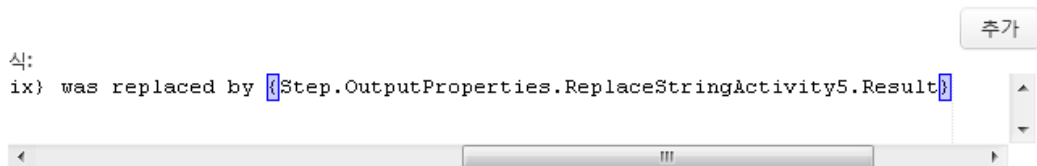
5. **이전 단계 입력에 이 식을 링크하여 사용자 지정 식의 마지막 부분을 추가합니다.**

사용자 지정 식의 마지막 요소인 경우 이전 단계 출력에 링크합니다.

- a. 링크 원본 선택 대화 상자 위쪽에서 **사용 가능한 단계** 옵션을 선택합니다. 대화 상자에 모든 이전 단계 목록이 표시됩니다.



- b. 단계 선택 창(왼쪽)에서 문자열 바꾸기 노드를 선택합니다. 사용 가능한 속성 목록이 표시됩니다.
- c. 속성 선택 창(오른쪽)에서 결과 행을 선택합니다.
- d. 식 영역에서 추가를 클릭합니다. 추가 식이 이전 식에 추가되고 문자열 바꾸기 단계 출력의 링크가 반영됩니다.




- e. 확인을 클릭하여 대화 상자를 닫고 이 식을 파일에 쓰기 단계의 콘텐츠 속성에 대한 값으로 추가합니다.


이제 콘텐츠 속성의 값 열에 업데이트된 식이 표시됩니다.

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

6. 테스트 실행 반복 수를 설정합니다.


- a. 캔버스에서 테스트 흐름을 선택합니다. 입력 탭  이 속성 창에서 열립니다.
- b. 입력 탭에서 'For' 루프 옵션이 선택되었는지 확인합니다.
- c. 반복 수에서 1을 입력합니다.

7. 테스트를 실행하고 실행 결과를 봅니다.

- a. 도구 모음에서 실행 버튼  을 클릭합니다. 실행 대화 상자가 열립니다.
- b. 실행 대화 상자의 결과 위치 탭에서 임시 실행 결과 옵션이 선택되었는지 확인합니다.
- c. 실행을 클릭하여 테스트 실행을 시작합니다.
UFT에서는 단계를 실행하여 사용자 지정 식에서 만든 링크를 통해 파일에 쓰기 단계의 콘텐츠 속성에 대한 값을 가져옵니다.
테스트 실행이 완료되면 실행 결과가 열립니다.
- d. 실행 결과에 테스트 흐름을 표시합니다.
- e. 테스트 흐름에서 파일에 쓰기 노드를 찾습니다. 실행 결과에 단계에 대한 관련 정보의 요약이 표시됩니다.
- f. 단계 요약에서 파일에 쓰기 링크를 클릭합니다. 단계에 대해 캡처된 데이터가 포함된 개별 탭이 열립니다.
- g. 개별 탭에서 이 테스트 실행에 사용된 콘텐츠 속성 값을 확인합니다.

Name	Value
유형	HP.ST.Ext.BasicActivities.FileWriteActivity
단계 ID	FileWriteActivity7
콘텐츠	'Hello world. was replaced by Goodbye world.'
파일 경로	'True'
인코딩	'ASCII'
모드	'기존 파일에 추가'
새 줄 추가	False

8. 테스트를 저장합니다.

저장  을 클릭합니다.

5단원: API 테스트 실행

"연습 3a: 표준 작업으로 테스트 만들기"(145페이지)에서는 표준 작업을 사용하여 기본 API 테스트를 만들었습니다. 이 테스트를 만들었으므로 이제 테스트를 실행할 수 있습니다.

이 단원에서는 테스트를 실행하고 실행 결과를 보는 방법을 알아봅니다.

이 단원의 내용은 다음과 같습니다.

- [연습 5a: 테스트 실행](#)172
- [연습 5b: 실행 결과 탐색](#) 173
- [연습 5c: 실행 결과 분석](#)174

연습 5a: 테스트 실행

"연습 3a: 표준 작업으로 테스트 만들기"(145페이지)에서는 표준 API 테스트 작업을 사용하여 테스트를 만들었습니다. 그리고 나서 "4단원: API 테스트 단계 매개 변수화"(150페이지)에서 다양한 메서드를 사용하여 테스트를 매개 변수화했습니다.

이 단원에서는 테스트 실행을 위해 UFT를 준비하는 방법 및 API 테스트를 실행하는 방법을 알아봅니다.

1. UFT를 시작하고 Book Flights 솔루션을 엽니다.


- "모든 테스트에 대한 솔루션 만들기"(25페이지)의 설명대로 UFT를 엽니다.
- 시작 페이지의 최근에 사용한 테스트 영역에서 **Flight Reservation Application**을 클릭합니다.
"연습 3a: 표준 작업으로 테스트 만들기"(145페이지)에서 만든 **Book Flights** 테스트가 포함된 Flight Reservation Application 솔루션이 열립니다.

2. 테스트에 대한 실행 모드를 설정합니다.

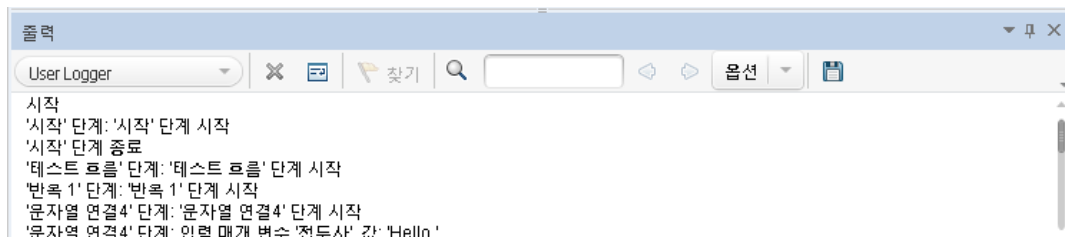
테스트를 실행하기 전에 테스트 실행 방법을 UFT에 지시해야 합니다. 테스트를 빠르게 실행하는 릴리스 모드 또는 UFT에서 테스트 실행 전에 디버깅 도구를 로드해야 하므로 더 느리게 실행하는 디버그 모드에서 테스트를 실행할 수 있습니다.

- 도구 > 옵션을 선택합니다. 옵션 대화 상자가 열립니다.
- 옵션 대화 상자에서 **API 테스트** 탭을 선택합니다.
- API 테스트 탭에서 **일반** 노드를 선택합니다.
- 일반 창에서 **디버깅 모드에서 테스트 실행** 옵션을 선택합니다.

3. 테스트 실행을 시작합니다.

- 도구 모음에서 실행 버튼  을 클릭합니다. 실행 대화 상자가 열립니다.
- 실행 대화 상자에서 옵션 막대를 클릭하여 대화 상자를 확장합니다.
- 옵션 영역에서 **결과 위치** 탭을 클릭합니다. 이렇게 하면 테스트 결과가 저장되는 위치를 지정할 수 있습니다.
- 결과 위치에서 **새 실행 결과 폴더** 옵션을 선택합니다. 기본 폴더 이름을 그대로 사용합니다.
- 실행을 클릭하여 실행 대화 상자를 닫고 테스트 실행을 시작합니다.

UFT에서는 테스트를 실행할 때 출력 및 출력 매개 변수와 검사점 실행 결과를 포함하여 수행되는 단계에 대한 로그를 출력 창에 제공합니다.



테스트 실행 중에 발생하는 오류는 로그 일부로 보고됩니다. 관련 단계로 돌아가서 이러한 오류를 수정할 수 있습니다.

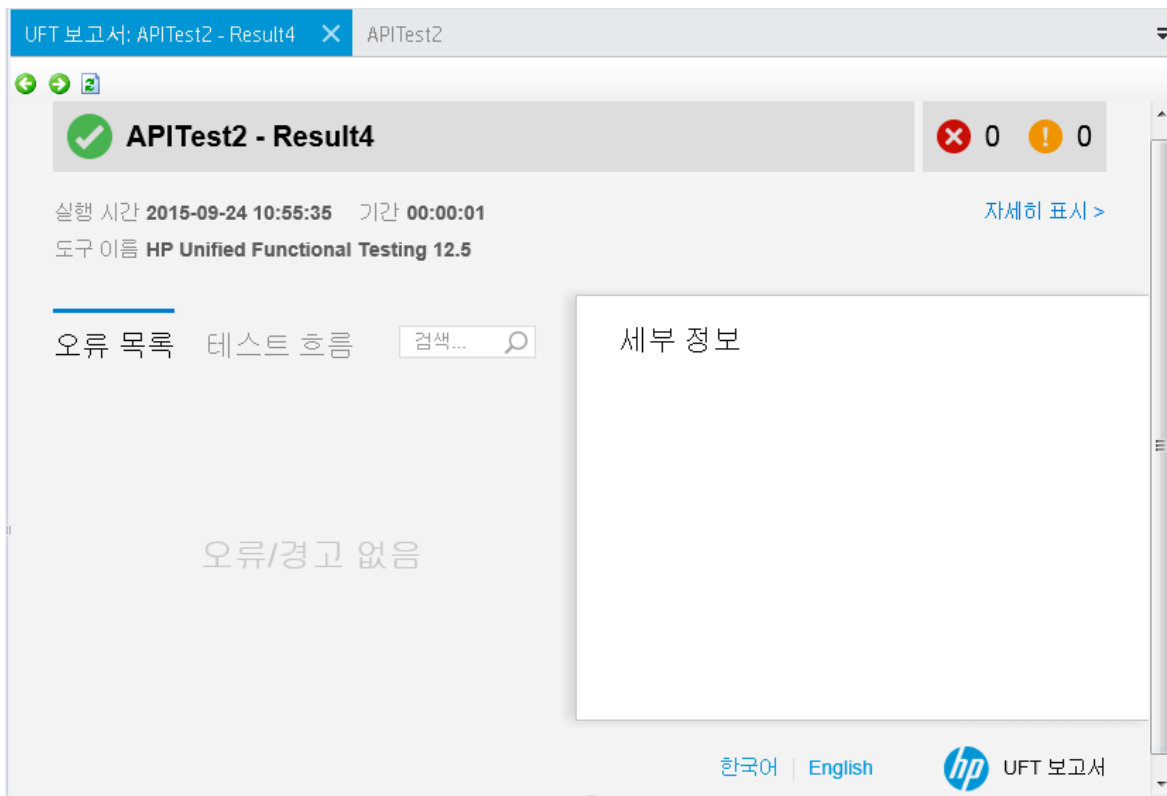
테스트 실행이 완료되면 실행 결과가 문서 창에서 개별 탭으로 열립니다. "[연습 5b: 실행 결과 탐색](#)"(173 페이지)으로 이동하여 실행 결과에 대해 자세히 알아보십시오.

연습 5b: 실행 결과 탐색

"[연습 5a: 테스트 실행](#)"(172페이지)에서는 직접 만든 표준 테스트를 실행했습니다. 테스트 실행이 완료되면 실행 결과에 이 테스트 실행에 대한 결과가 자동으로 표시됩니다.

참고: 기본적으로 실행 결과는 HTML 기반 보고서로 표시됩니다. 실행 결과가 Run Results Viewer에 표시되는 옵션 대화 상자의 **실행 세션 창**(**도구 > 옵션 > 일반 탭 > 실행 세션** 노드)에 표시되도록 선택할 수도 있습니다. 이 자습서의 단원은 HTML 기반 보고서에 기반을 둡니다.

실행 결과가 열리면 실행 결과에 다음 내용이 표시됩니다.



처음에 실행 결과에는 다음 내용이 표시됩니다.

테스트 흐름	트리에 표시되는 결과의 그래픽 표현으로, 테스트의 단계에 따라 구성됩니다. 각 실행에서 여러 데이터 집합을 사용하여 두 번 이상 테스트를 실행하도록 UFT에 지시할 수 있습니다. 각 실행을 반복이라고 하며 각 반복은 번호가 매겨집니다.
오류 목록	목록으로 제공되는 모든 오류 및 경고입니다.

<p>단계 요약 정보</p>	<p>테스트에 대한 일반 정보, 통과하거나 실패한 단계, 각 테스트 단계에 대한 세부 정보를 포함하는 상위 수준 결과 개요 보고서입니다.</p> <p>요약에는 해당 테스트 단계에 대해 캡처된 데이터를 열 수 있는 링크도 포함됩니다.</p>
------------------------	---

UFT가 직접 만든 단계와 제공한 속성에 따라 모든 단계를 제대로 수행할 수 있었으므로 테스트 실행에 성공했습니다. 오류가 발생하여 테스트 실행에 실패한 경우 오류가 출력 창의 로그에 나열됩니다. 이 경우 돌아가서 이 자습서의 설명대로 단계가 정확히 구성되었는지 확인합니다.

실행 결과에 표시되는 내용을 알아보았으므로 이제 **"연습 5c: 실행 결과 분석"(174페이지)**으로 이동하여 실행 결과의 세부 정보를 알아보십시오.

연습 5c: 실행 결과 분석

이 연습에서는 **"연습 5a: 테스트 실행"(172페이지)**에서 테스트를 실행할 때 UFT에서 수행된 단계를 검사합니다.

1. 특정 단계에 대한 결과를 봅니다.

- 테스트 흐름의 결과 트리에서 **테스트 흐름 > 요약 > 반복 1** 노드를 찾아서 이 테스트에서 수행된 모든 단계를 확인합니다.
- 반복 1 노드에서 **문자열 바꾸기** 노드를 선택합니다.

The screenshot displays the UFT test results interface. At the top, it shows 'APITest2 - Result4' with a success icon and zero errors/warnings. Below this, it indicates the execution time (2015-09-24 10:55:35) and duration (00:00:01). A search bar is present for filtering test steps. The test flow tree on the left shows the following steps: 시작, 테스트 흐름, 수행, 수행 반복: 행 1, 문자열 연결4, and 문자열 바꾸기5. The '문자열 바꾸기5' step is highlighted. The '단계 세부 정보' panel on the right provides details for this step, including its name '문자열 바꾸기5', execution time '2015-09-24 10:55:35', and a section for '캡처된 데이터' (Captured Data) with 'Step Properties'.

실행 결과에 다음 정보가 표시됩니다.

- 테스트 흐름 및 강조 표시된 단계
- 강조 표시된 단계의 세부 정보가 표시된 테스트 단계 요약
- 선택한 단계에 대해 캡처된 데이터를 볼 수 있는 링크

2. 실행 결과를 달습니다.

문서 창에서 실행 결과가 포함된 탭을 달습니다.

6단원: 웹 서비스의 API 테스트 만들기 및 실행

"3단원: 표준 작업을 사용하여 API 테스트 단계 만들기"(144페이지)에서는 표준 API 작업을 사용하여 테스트를 만드는 방법을 알아보았습니다.

그러나 표준 작업이 응용 프로그램에서 수행하는 프로세스와 일치하지 않는 경우도 있습니다. 그러한 경우에는 UFT로 가져오는 사용자 지정 작업을 사용해야 합니다. 널리 사용되는 서비스 작업 유형의 하나는 웹 서비스입니다. UFT에서 서비스와 해당 메시지를 UFT로 가져와서 테스트에 사용할 수 있습니다.

이 단원에서는 웹 서비스 테스트를 만들고 실행하는 방법을 알아봅니다.

이 단원의 내용은 다음과 같습니다.

- 연습 6a: 웹 서비스 테스트 만들기 177
- 연습 6b: 웹 서비스 가져오기 177
- 연습 6c: 웹 서비스 테스트 빌드 및 매개 변수화 179
- 연습 6d: 웹 서비스 테스트 실행 185

연습 6a: 웹 서비스 테스트 만들기

"모든 테스트에 대한 솔루션 만들기"(25페이지)에서는 항공편 예약 응용 프로그램 테스트에 대한 솔루션을 만들었으며, "자동화 GUI 테스트 만들기 및 실행"(27페이지)에서는 항공편 예약 사용자 인터페이스의 성능을 테스트하는 다양한 GUI 테스트를 만들었습니다.

이 연습에서는 항공편 예약 응용 프로그램의 서비스(API) 레이어를 테스트하는 데 필요한 첫 번째 API 테스트를 만듭니다.

1. **UFT를 시작하고 항공편 예약 응용 프로그램 솔루션을 엽니다.**
 - a. "모든 테스트에 대한 솔루션 만들기"(25페이지)의 설명대로 UFT를 엽니다.
 - b. 시작 페이지의 **최근에 사용한 솔루션** 영역에서 **Flight Reservation Application**을 클릭합니다. "1단원: GUI 테스트 및 수행 만들기"(28페이지)에서 만든 **Book Flights** 테스트가 포함된 Flight Reservation Application 솔루션이 열립니다.
2. **솔루션에 API 테스트를 새로 추가합니다.**
 - a. 도구 모음에서 **추가** 드롭다운 화살표  **추가** 를 선택하고 **새 테스트 추가**를 선택합니다. 새 테스트 추가 대화 상자가 열립니다.
 - b. 새 테스트 추가 대화 상자에서 **API 테스트**를 선택합니다.
 - c. **이름** 필드에서 테스트 이름을 Book Flights Web Service로 지정합니다.
 - d. **위치** 필드에서 **찾아보기** 버튼을 클릭하고 **C:\%HOMEPATH%\Unified Functional Testing** 폴더로 이동합니다.
 - e. **추가**를 클릭하여 테스트를 만들고 솔루션에 추가합니다.
Book Flights Web Service 테스트가 Flight Reservation Application 솔루션에서 개별 노드로 추가되고 문서 창에서 개별 탭으로 열립니다.

참고: 솔루션이 자동으로 저장됩니다.

테스트를 만들었으므로 이제 웹 서비스와 해당 메서드를 사용할 준비가 되었습니다. "연습 6b: 웹 서비스 가져오기"(177페이지)로 이동하여 웹 서비스를 테스트로 가져오는 방법을 알아보십시오.

연습 6b: 웹 서비스 가져오기

웹 서비스를 테스트하기 전에 서비스 설명과 해당 메서드를 UFT로 가져와야 합니다. 일반적으로 서비스 설명은 WSDL(Web Service Description Language) 파일에 저장됩니다. 이 파일에서는 서비스 작업/메서드 및 서비스에 대한 메타데이터를 정의합니다. 그러면 UFT에서는 이 WSDL 파일을 읽고 도구 상자 창에서 서비스 메서드를 작업으로 만듭니다.

이 연습에서는 항공편 예약 응용 프로그램의 서비스 WSDL 파일을 UFT로 가져웁니다.

1. Flights API 응용 프로그램을 시작합니다.

"Flight Reservation Application 탐색"(23페이지)의 설명대로 HP Flights Service API 응용 프로그램을 시작합니다.

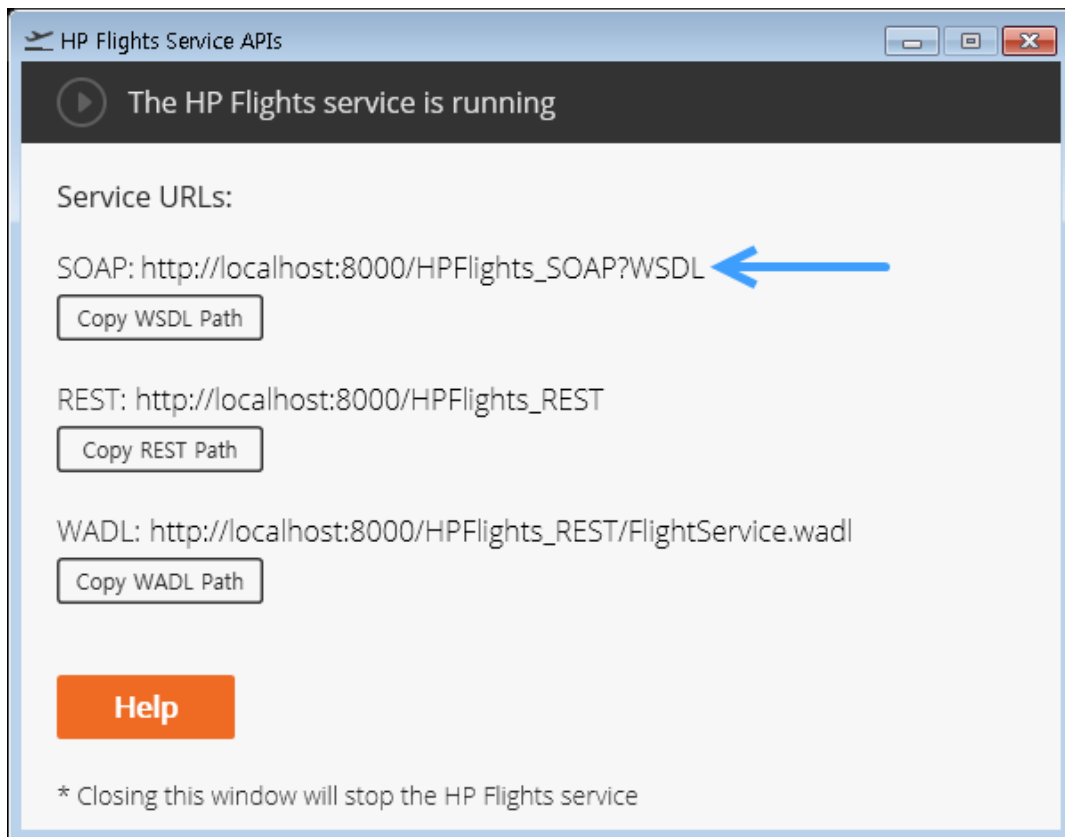
참고: 테스트를 편집하고 실행할 때 UFT에서 응용 프로그램에 액세스할 수 있어야 하므로 자습서를 사용할 때 이 응용 프로그램이 열려 있어야 합니다.

2. WSDL 파일을 가져옵니다.

- a. UFT의 도구 모음에서 **WSDL 가져오기** 버튼을 누르고 **URL** 또는 **UDDI**에서 **WSDL 가져오기**를 선택합니다. URL 또는 UDDI에서 WSDL 가져오기 대화 상자가 열립니다.

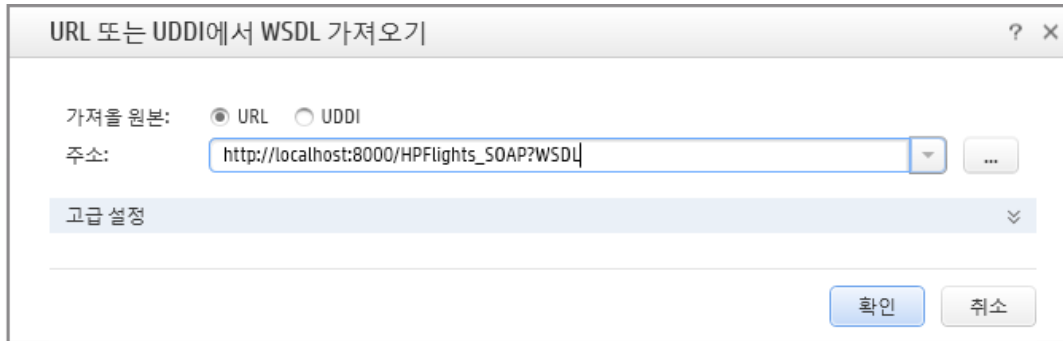
참고: WSDL 파일 복사본을 로컬로 저장하거나 ALM 프로젝트에서 저장했으면 파일을 UFT로 바로 가져올 수 있습니다.

- b. URL 또는 UDDI에서 WSDL 가져오기 대화 상자에서 **URL** 옵션을 선택합니다.
- c. HP Flights Service API 응용 프로그램 창에서 SOAP 기반 서비스의 URL을 찾습니다.

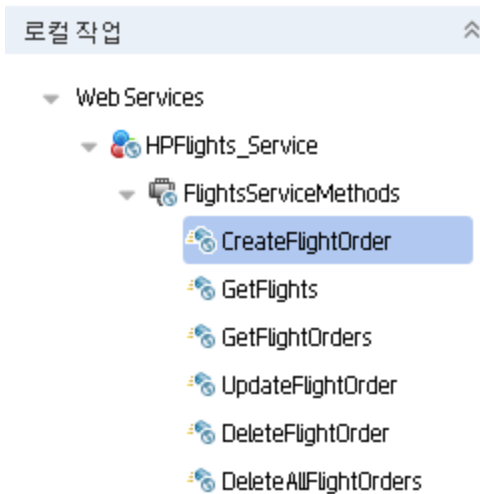


- d. HP Flights Service API 응용 프로그램 창에서 **WSDL 경로 복사** 버튼을 클릭합니다. 그러면 WSDL 파일의 URL이 저장되므로 URL을 URL 또는 UDDI에서 WSDL 가져오기 대화 상자로 복사할 수 있습니다.

- e. UFT의 URL 또는 UDDI에서 WSDL 가져오기 대화 상자에 있는 주소 필드에 응용 프로그램 창에서 복사한 URL을 붙여넣습니다.



- f. **확인**을 클릭하여 서비스를 UFT로 가져옵니다.
서비스를 UFT로 가져오고 해당 메시드가 로컬 작업 섹션 아래 도구 상자 창에 표시됩니다.



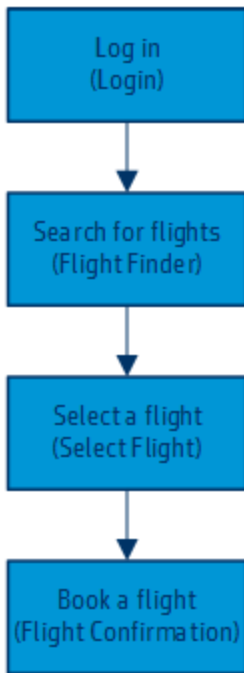
서비스 및 해당 메시지를 테스트로 가져왔으므로 이제 웹 서비스 테스트를 만들 준비가 되었습니다. "[연습 6c: 웹 서비스 테스트 빌드 및 매개 변수화](#)"(179페이지)로 이동하여 웹 서비스 테스트를 빌드 및 매개 변수화하십시오.

연습 6c: 웹 서비스 테스트 빌드 및 매개 변수화

"[연습 6b: 웹 서비스 가져오기](#)"(177페이지)에서는 웹 서비스 세부 정보가 포함된 WSDL 파일을 가져왔습니다. 서비스를 가져온 후 UFT에서는 도구 상자 창에 메시지를 표시했습니다. 도구 상자 창에서 이러한 메시지를 사용하여 테스트를 만들 수 있습니다.

이 단원에서는 웹 서비스 테스트를 만들고 매개 변수화하여 UFT API 테스트 인터페이스를 사용해 웹 서비스 테스트를 만드는 방법을 알아봅니다.

"3단원: 테스트에 단계 추가"(44페이지)에서 만든 Book Flights GUI 테스트에서 응용 프로그램 창 순서는 다음과 같습니다.



같은 응용 프로그램의 API 테스트를 만들 때 단계가 가능한 한 응용 프로그램 흐름과 일치하도록 만들 수 있습니다. WSDL 파일에서 가져온 메서드 목록에는 다음이 포함됩니다.

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


사용자 인터페이스 흐름과 일치시키려면 항공편을 찾는 API 테스트 단계를 만든 다음, 고객 입력을 기반으로 한 항공편 주문을 만들어야 합니다.

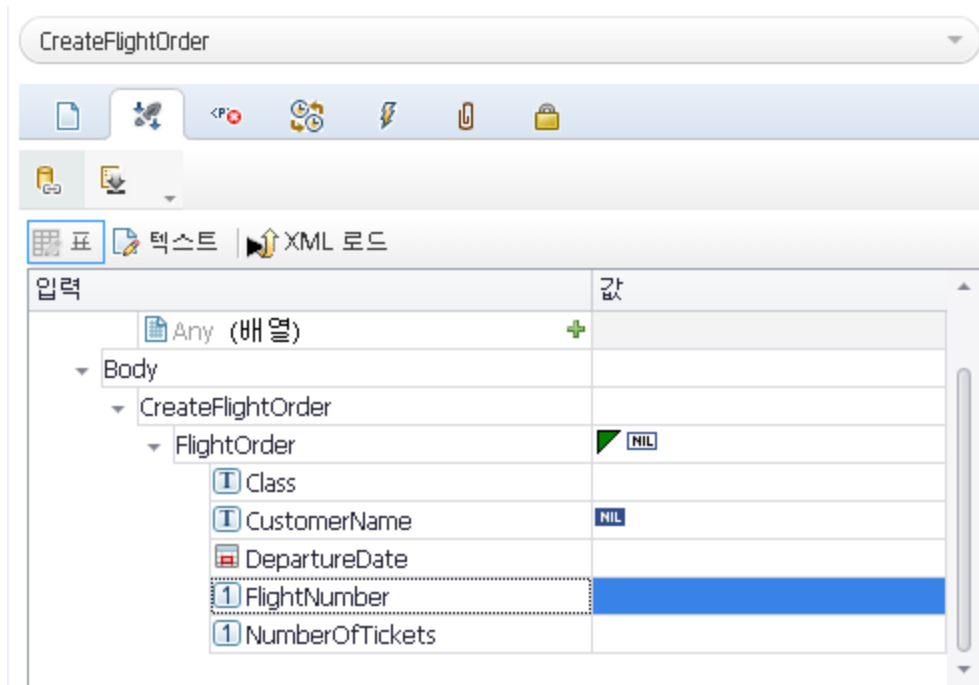
이 연습에서는 두 가지 테스트 단계 **GetFlights** 및 **CreateFlightOrder**를 만듭니다.


1. 테스트 단계를 만듭니다.

- a. 도구 상자 창의 **로컬 작업** 섹션에서 웹 서비스를 확장하고 **HP_Flights Services** 및 **FlightsServiceMethods** 노드를 확장합니다. 도구 상자 창에 사용 가능한 메서드(모두 6개)의 전체 목록이 표시됩니다.
- b. FlightsServiceMethods 목록에서 **GetFlights** 메서드를 캔버스로 끕니다.
GetFlights라는 새 단계 블록이 캔버스에 추가됩니다. **입력/검사점** 탭이 속성 창에 열립니다.
- c. 다시 도구 상자 창의 **FlightsServiceMethods** 노드에서 **CreateFlightOrder** 메서드를 캔버스로 끕니다.

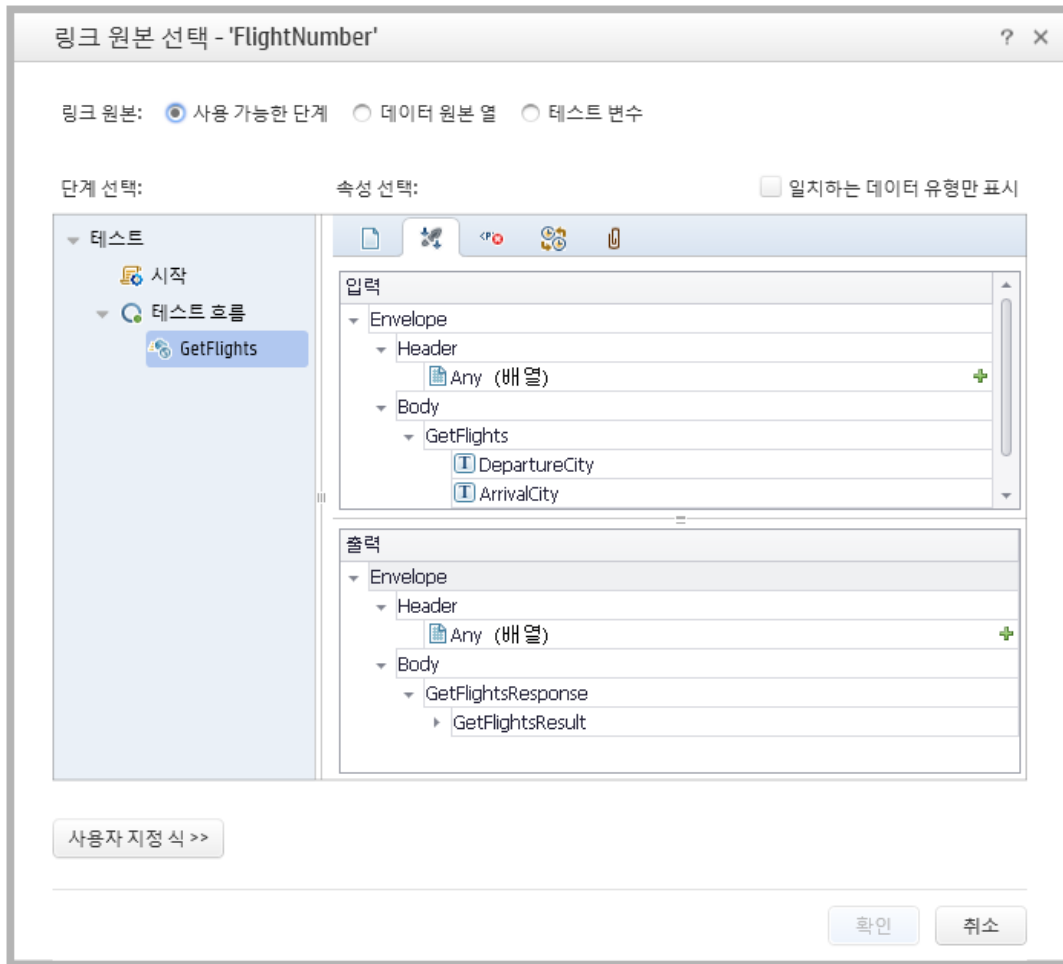
2. ReserveOrder 단계의 FlightNumber 속성을 CreateFlightOrder 단계의 출력에 링크합니다.



- a. 캔버스에서 CreateFlightOrder 단계를 선택합니다. 입력/검사점 탭  이 속성 창에서 열립니다.
- b. 입력/검사점 탭의 입력 섹션에서 FlightNumber 행을 선택합니다.



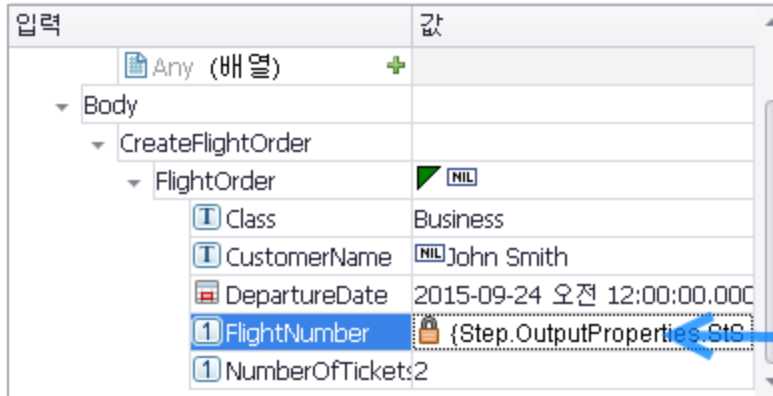
- c. FlightNumber 행의 값 열에서 데이터 원본으로 링크 버튼  을 클릭합니다. 링크 원본 선택 대화 상자가 열립니다.
- d. 링크 원본 선택 대화 상자에서 사용 가능한 단계 옵션을 선택합니다. 단계 선택: 창(왼쪽)이 사용 가능한 단계 목록으로 업데이트됩니다.

- e. 단계 선택: 창에서 **GetFlights** 단계를 선택합니다. 속성 선택: 창(오른쪽)이 단계 속성으로 업데이트됩니다.



- f. 속성 선택: 창의 **입력/검사점** 탭  에 있는 **출력** 섹션에서 **GetFlightsResult** 노드를 확장합니다.
- g. GetFlightsResult 노드 아래의 **항공편(배열)** 행에서 **추가** 버튼  을 클릭합니다. 새 출력 배열이 출력 속성에 추가됩니다.
- h. 항공편(배열) 행에서 **항공편[1]** 배열을 확장합니다. GetFlights 단계에 대한 모든 출력 속성 목록이 표시됩니다.
- i. 출력 속성 목록에서 **FlightNumber** 속성을 선택하고 **확인**을 클릭합니다. 대상 단계를 루프에 포함할지 묻는 메시지가 표시되면 **아니요**를 선택합니다.

UFT에서 FlightNumber 속성에 대한 값 열을 업데이트하여 링크를 반영합니다.



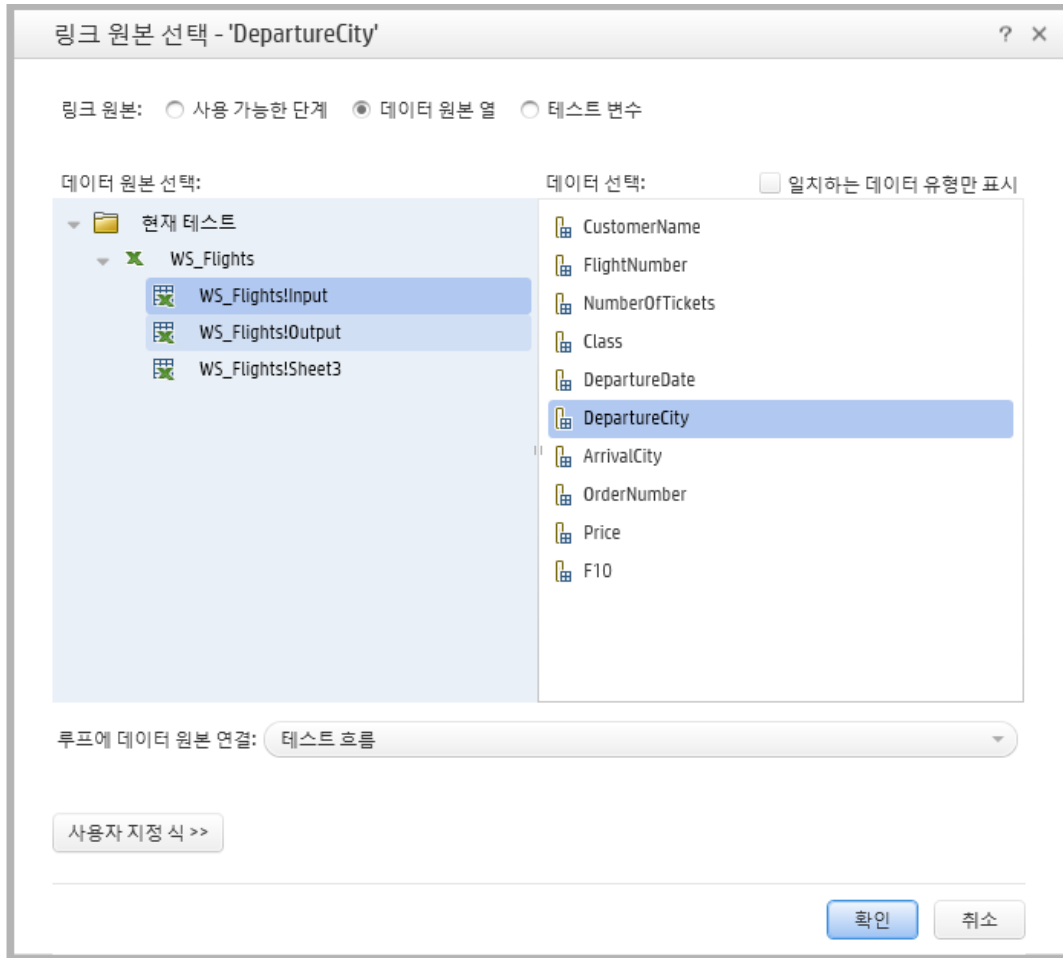
3. 테스트에서 사용할 데이터 원본을 추가합니다.

- 필요한 경우 보기 > 데이터를 선택하여 데이터 창을 표시합니다.
- 데이터 창에서 새 데이터 원본 버튼 을 클릭하고 Excel을 선택합니다. 새 Excel 데이터 원본 대화 상자가 열립니다.
- 새 데이터 원본 대화 상자의 Excel 파일 경로 필드에서 찾아보기 버튼을 클릭합니다.
- 열기 대화 상자에서 <UFT 설치 디렉터리>\samples\Flights Application에 저장된 응용 프로그램 Excel 파일로 이동하고 확인을 클릭합니다.
- 새 Excel 데이터 원본 대화 상자에서 파일 이름을 WS_Flights로 지정합니다.
- 원래 위치의 Excel 파일에 링크 옵션을 선택합니다.
- 확인을 클릭하여 데이터 원본 정보를 저장하고 Excel 데이터를 테스트에 추가합니다.

4. 테스트 단계의 입력 속성을 데이터 원본에 링크합니다.

- 캔버스에서 GetFlights 단계를 선택합니다. 입력/검사점 탭 이 속성 창에서 열립니다.
- 입력/검사점 탭의 입력 섹션에서 GetFlights 노드를 확장합니다.
- GetFlights 노드에서 DepartureCity 행을 선택합니다.
- DepartureCity 행의 값 열에서 데이터 원본으로 링크 버튼 을 클릭합니다. 링크 원본 선택 대화 상자가 열립니다.
- 링크 원본 선택 대화 상자에서 데이터 원본 열 옵션을 선택합니다. 데이터 원본 선택 창(왼쪽)에 데이터 원본 목록이 표시됩니다(이 경우 Excel 파일만).
- 데이터 원본 선택 창에서 WS_Flights!Input 노드를 선택합니다. 데이터 선택 창(오른쪽)에 사용 가능한 모든 데이터 매개 변수(열) 목록이 표시됩니다.

g. 데이터 선택 창에서 **DepartureCity** 열을 선택합니다.



h. **확인**을 클릭하여 데이터 테이블에서 이 열에 속성을 링크합니다.

UFT에서 데이터 원본에 대한 링크를 보여 주는 문으로 값 열을 업데이트합니다.

i. 단계의 기타 속성에 대해 위의 프로세스를 반복합니다.

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


j. CreateFlightOrder 단계의 속성에 대해 같은 프로세스를 반복합니다.

- **클래스**
- **CustomerName**

참고: 이 속성에 대한 값 열에는 안에 **NIL**이 기록된 파란색 상자가 있습니다. 속성을 링크하기 전에 이 상자를 클릭하고 **NIL**을 제거해야 합니다(상자가 흰색으로 바뀜).

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

5. 데이터 원본에 대한 탐색 설정을 설정합니다.

- a. 캔버스에서 **테스트 흐름**을 선택합니다(테스트 흐름의 단계가 아님).
- b. 속성 창에서 **데이터 원본 탭** 을 선택합니다.
- c. 데이터 원본 탭의 연결된 데이터 원본 목록에서 테이블의 **WS_Flights!Input** 항목을 선택하고 **편집**을 클릭합니다. 데이터 탐색 대화 상자가 열립니다.
- d. 데이터 탐색 대화 상자에서 다음과 같이 데이터 탐색 세부 정보를 지정합니다.

시작 위치:	First row
위로 이동:	3 rows Forward
종료 위치:	Last row
마지막 행에 도달할 경우:	Wrap around

- e. **확인**을 클릭하여 데이터 탐색 속성을 지정하고 대화 상자를 닫습니다.

항공편 예약 응용 프로그램의 웹 서비스에 대한 테스트를 만들었으므로 이제 테스트를 실행하고 UFT가 웹 서비스에 대한 실행 결과를 어떻게 실행 및 보고하는지 확인할 수 있습니다. "[연습 6d: 웹 서비스 테스트 실행](#)"(185페이지)으로 이동하여 자세히 알아보십시오.

연습 6d: 웹 서비스 테스트 실행

"[연습 6c: 웹 서비스 테스트 빌드 및 매개 변수화](#)"(179페이지)에서는 가져온 메서드를 기반으로 웹 서비스 테스트를 만들고 단계 중 하나를 매개 변수화했습니다. 이 단원에서는 테스트를 실행하여 UFT가 웹 서비스 응용 프로그램을 테스트할 때 실행 결과를 보고하는 방식을 확인합니다.

1. UFT를 시작하고 Book Flights 솔루션을 엽니다.


- a. "[모든 테스트에 대한 솔루션 만들기](#)"(25페이지)의 설명대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- b. 시작 페이지의 **최근에 사용한 솔루션** 영역에서 **Flight Reservation Application**을 클릭합니다. "[1단원: GUI 테스트 및 수행 만들기](#)"(28페이지)에서 만든 **Book Flights** 테스트가 포함된 Flight Reservation Application 솔루션이 열립니다.

2. Flights API 응용 프로그램을 시작합니다.

필요한 경우 "[Flight Reservation Application 탐색](#)"(23페이지)의 설명대로 HP Flights Service API 응용 프로그램을 시작합니다.


참고: 테스트를 편집하고 실행할 때 UFT에서 응용 프로그램에 액세스할 수 있어야 하므로 자습서를 사용할 때 이 응용 프로그램이 열려 있어야 합니다.

3. 테스트 반복 수를 설정합니다.

- a. 캔버스에서 **테스트 흐름**을 선택합니다(테스트 흐름의 단계가 아님). **입력 탭** 이 속성 창에서 열립니다.

- b. 입력 탭에서 'For' 루프 옵션을 선택합니다.
- c. 반복 수 필드에서 4를 입력합니다.

4. 테스트를 실행합니다.

- a. 도구 모음에서 실행 버튼  을 클릭합니다. 실행 대화 상자가 열립니다.
- b. 실행 대화 상자의 결과 위치 탭에서 **임시 실행 결과 폴더**가 선택되어 있는지 확인합니다.
- c. 실행을 클릭하여 테스트를 컴파일 및 실행합니다.
테스트 실행이 완료되면 실행 결과가 열립니다.

5. 실행 결과를 봅니다.

- a. 테스트 흐름의 반복 노드 아래에서 **GetFlights** 노드를 선택합니다. 단계 요약 세부 정보가 표시 됩니다.
- b. 캡처된 데이터에서 웹 서비스 호출 HTTP 스냅샷 영역이 표시될 때까지 아래로 스크롤합니다.

Web Service Call HTTP Snapshot

Request	Response
HTTP Header	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
<pre><Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/"> <Body> <GetFlights xmlns="HP.SOAQ.SampleApp"> <DepartureCity>Los Angeles</DepartureCity> <ArrivalCity>Sydney</ArrivalCity> </GetFlights> </Body> </Envelope></pre>	<pre><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></pre>

이 영역에서 UFT는 웹 서비스 호출에 대한 HTTP 요청 및 응답 정보를 제공합니다.

요청에 대한 SOAP 창에서 GetFlights 단계에 대해 전송된 입력 속성을 확인할 수 있습니다.

```
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Body>
    <GetFlights xmlns="HP.SOAQ.SampleApp">
      <DepartureCity>Los Angeles</DepartureCity>
      <ArrivalCity>Sydney</ArrivalCity>
    </GetFlights>
  </Body>
</Envelope>
```

마찬가지로 **응답 영역**에서 단계의 출력 속성을 확인할 수 있습니다.

```
SOAP
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envel
  <s:Body>
    <GetFlightsResponse xmlns="HP.SOAQ.SampleApp">
      <GetFlightsResult xmlns:i="http://www.w3.org/2001/XM
        <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>
```

- c. **GetFlights** 노드에서 **검사점** 노드를 선택합니다. 실행 결과에 검사점 요약이 표시됩니다.
- d. 검사점에 대한 단계 세부 정보에서 UFT는 결과를 표시합니다(검사점 통과 또는 실패 여부, 실제 및 예상 값, 검사점 유형).

Name	Result	Property	Actual Result	Evaluation Style	Expected Values	Details
"검사점 1"	✓	""	""	구조 유효성 검사	""	

6. 테스트를 저장합니다.

파일 > 저장을 선택합니다.

7단원: REST 서비스의 API 테스트 만들기 및 실행

UFT에서 웹 서비스를 테스트할 뿐만 아니라 API 테스트를 사용하여 응용 프로그램의 REST 기반 서비스 또는 REST 기반 서비스 레이어를 테스트할 수도 있습니다. UFT에서 서비스의 프로토타입 모델을 만들고 생성된 메서드를 사용하여 테스트를 구조화합니다.

이 단원에서는 REST 서비스 프로토타입 모델을 만들고 이러한 메서드 모델을 사용하여 테스트를 만드는 기본 단계를 설명합니다.

이 단원의 내용은 다음과 같습니다.

- 연습 7a: REST 서비스 테스트 만들기 189
- 연습 7b: REST 서비스 구조 만들기 189
- 연습 7c: REST 서비스 메서드를 사용하여 테스트 만들기 194
- 연습 7d: REST 서비스 테스트 실행 196
- 연습 7e: REST 서비스 충돌 해결 198

연습 7a: REST 서비스 테스트 만들기

"6단원: 웹 서비스의 API 테스트 만들기 및 실행"(176페이지)에서는 Flight Reservation Application의 API 쪽에 대한 웹 서비스 테스트를 만들었습니다. 이 연습에서는 항공편 예약 응용 프로그램의 API 쪽에 대한 REST 서비스 구성 요소 테스트를 추가합니다.

1. UFT를 시작하고 항공편 예약 응용 프로그램 솔루션을 엽니다.

- "모든 테스트에 대한 솔루션 만들기"(25페이지)의 설명대로 UFT를 엽니다.
- 시작 페이지의 **최근에 사용한 솔루션** 영역에서 **Flight Reservation Application**을 클릭합니다. Flight Reservation Application 솔루션이 열립니다.

2. 솔루션에 API 테스트를 새로 추가합니다.

- 도구 모음에서 **추가** 드롭다운 화살표  **추가**  를 선택하고 **새 테스트 추가**를 선택합니다. 새 테스트 추가 대화 상자가 열립니다.
- 새 테스트 추가 대화 상자에서 **API 테스트**를 선택합니다.
- 이름** 필드에서 테스트 이름을 Book Flights REST Service로 지정합니다.
- 위치** 필드에서 **찾아보기** 버튼을 클릭하고 **C:\%HOMEPATH%\Unified Functional Testing** 폴더로 이동합니다.
- 추가**를 클릭하여 테스트를 만들고 솔루션에 추가합니다. Book Flights REST Service 테스트가 Flight Reservation Application 솔루션에서 개별 노드로 추가되고 문서 창에서 개별 탭으로 열립니다.

참고: 솔루션이 자동으로 저장됩니다.

테스트를 만들었으므로 REST 서비스 모델과 해당 메서드를 사용할 준비가 되었습니다. 이제 "[연습 7b: REST 서비스 구조 만들기](#)"(189페이지)로 이동하여 테스트에서 REST 서비스 모델을 만드는 방법을 알아 봅니다.

연습 7b: REST 서비스 구조 만들기

필요한 메서드 및 해당 속성의 모델을 UFT 내부에 만들어야 REST 서비스 작업을 테스트에 사용할 수 있습니다. UFT에서는 서비스 및 메서드에 대한 정보를 가져오고 이를 테스트 단계로 사용하여 실제 서비스 성능을 테스트합니다.

1. UFT를 시작하고 Book Flights 솔루션을 엽니다.

- "모든 테스트에 대한 솔루션 만들기"(25페이지)의 설명대로 UFT를 엽니다.
- 시작 페이지의 **최근에 사용한 솔루션** 영역에서 **Flight Reservation Application**을 클릭합니다.

"연습 7a: REST 서비스 테스트 만들기"(189페이지)에서 만든 **Book Flights REST Service** 테스트가 포함된 Flight Reservation Application 솔루션이 열립니다.

2. Flights API 응용 프로그램을 시작합니다.

HP Flights Service API 응용 프로그램이 "Flight Reservation Application 탐색"(23페이지)에 설명된 대로 실행 중인지 확인합니다.


3. REST 서비스 메서드 속성 도움말 문서를 엽니다.

HP Flights Service API 창에서 도움말 버튼을 클릭합니다. 메서드 정보가 있는 브라우저 창이 열립니다.


4. REST 서비스 모델을 만듭니다.

- a. 도구 상자에서 **REST 서비스 추가** 버튼을 클릭합니다. REST 서비스 추가 대화 상자가 열립니다.
- b. REST 서비스 추가 대화 상자에서 **새 서비스 이름**을 Flights REST Service로 변경합니다.

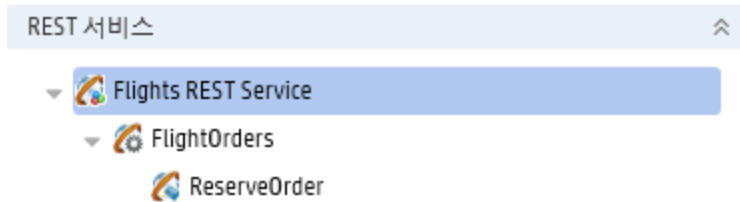
5. REST 서비스 모델에 리소스를 추가합니다.

- a. REST 서비스 추가 대화 상자의 도구 모음에서 **리소스 추가** 버튼  을 클릭합니다. 하위 노드가 Flights REST Service 노드에 추가됩니다.
- b. 리소스 이름을 **FlightOrders**로 변경합니다.

6. REST 서비스 모델에 메서드를 추가합니다.

- a. REST 서비스 추가 대화 상자의 도구 모음에서 **메서드 추가** 버튼  을 클릭합니다. 하위 노드가 FlightOrders 리소스에 추가됩니다.
- b. 리소스 이름을 ReserveOrder로 변경합니다.








서비스, 리소스 및 메서드를 추가했으므로 이제 다음 3개 수준 계층 구조가 생성됩니다.

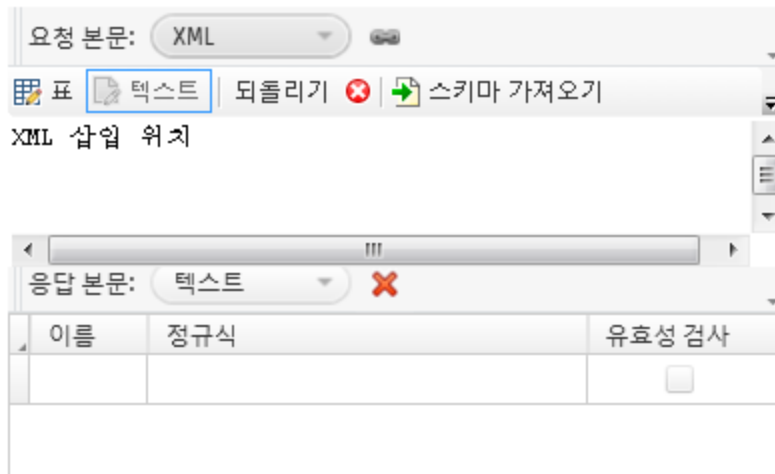


7. REST 서비스 메서드 모델 URL을 구성합니다.

REST 서비스 모델 메서드가 응용 프로그램을 정확히 테스트하도록 하려면 응용 프로그램 서비스의 URL을 제공해야 합니다. 이 URL은 **서비스, 리소스 및 메서드** 수준에서 제공됩니다.

- a. REST 서비스 추가 대화 상자에서 **Flights REST Service** 노드를 선택합니다. **일반 탭**  이 오른쪽 창에서 열립니다.

- b. 일반 탭의 **URL** 속성에 대한 **값** 열에서 `http://localhost:8000`을 입력합니다.
 - c. 왼쪽 창에서 **Flight Orders** 행을 선택합니다. **일반 창**  이 다시 오른쪽 창에서 열립니다. Flights REST Service가 선택될 때 일반 창에서 추가한 URL이 표시되는지 확인합니다.
 - d. 일반 탭의 **상대 URL** 속성에 대한 **값** 열에서 `HPFlights_REST`를 입력합니다.
URL의 이 부분을 추가하면 UFT에서는 **상대 URL** 값을 **URL** 값에 추가하여 연결된 URL `http://localhost:8000/HPFlights_REST`를 만듭니다.
 - e. 왼쪽 창에서 **ReserveOrder** 노드를 선택합니다. **일반 창**  이 오른쪽 창에 열립니다.
 - f. 오른쪽 창에서 **HTTP 입력/검사점** 탭  을 선택합니다.
 - g. **상대 URL** 속성에 대한 **값** 열에서 `/FlightOrders`를 입력합니다.
UFT에서는 URL의 이 부분을 **Flights REST Service** 및 **Flight Orders** 수준에서 전달된 URL과 연결합니다.
8. **ReserveOrders 메서드 모델에 대한 추가 HTTP 속성을 구성합니다.**
- a. REST 서비스 추가 대화 상자에서 **ReserveOrder** 노드를 선택합니다. **일반 창**  이 오른쪽 창에 열립니다.
 - b. 오른쪽 창에서 **HTTP 입력/검사점** 탭  을 선택합니다.
 - c. **HTTP 메서드** 속성에 대한 **값** 열에서 HTTP 유형을 `POST`로 설정합니다.
9. **ReserveOrder 메서드 모델에 대한 응답 정보를 추가합니다.**
- a. REST 서비스 추가 대화 상자에서 **ReserveOrder** 노드를 선택합니다. **일반 탭**  이 오른쪽 창에서 열립니다.
 - b. 오른쪽 창에서 **HTTP** 탭  을 선택합니다.
 - c. **요청 본문** 섹션의 **요청** 드롭다운 목록에서 **XML**을 선택합니다. XML을 입력할 수 있도록 텍스트 편집기가 열립니다.




- d. 요청 본문 유형에서 **XML**이 선택되었는지 확인합니다.
- e. 텍스트 편집기 영역에서 다음 XML을 입력합니다.

```
<FlightOrderDetails xmlns="HP.SOAQ.SampleApp">
  <Class>Business</Class>
  <CustomerName>John Parker</CustomerName>
  <DepartureDate>2115-05-27</DepartureDate>
  <FlightNumber>1042</FlightNumber>
  <NumberOfTickets>1</NumberOfTickets>
</FlightOrderDetails>
```

참고: 이 XML을 파일로 저장하고 **XML 로드** 버튼을 클릭하여 XML을 입력할 수도 있습니다.

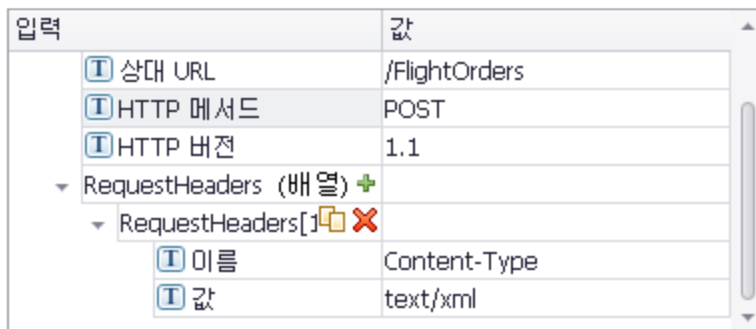
표 버튼을 클릭하면 표 양식에 표시된 XML에 입력한 속성을 확인할 수도 있습니다.



- f. 오른쪽 창에서 **HTTP 입력/검사점** 탭  을 다시 선택합니다.
- g. HTTP 입력/검사점 탭의 입력 섹션에서 **요청 헤더** 노드, **요청 헤더 [1]** 노드를 차례로 확장합니다.


응답에 대한 설정을 확인합니다.

- o **이름:** Content - Type
- o **값:** text/xml




10. ReserveOrder 메서드 모델에 대한 출력 속성을 만듭니다.

- a. REST 서비스 추가 대화 상자에서 **ReserveOrder** 노드를 선택합니다. **일반** 탭  이 오른쪽 창에서 열립니다.

- b. 오른쪽 창에서 사용자 지정 입력/검사점 탭  을 선택합니다.
 - c. 사용자 지정 입력/검사점 탭에서 추가 버튼을 클릭하고 출력 속성 추가를 선택합니다. 출력 속성 추가 대화 상자가 열립니다.
 - d. 출력 속성 추가 대화 상자의 이름 필드에 Total_Price를 입력합니다.
 - e. 유형 드롭다운 메뉴에서 Int를 선택하고 확인을 클릭하여 출력 속성을 추가합니다. 새 출력 속성이 사용자 지정 입력/검사점 탭의 출력 섹션에 추가됩니다.
 - f. 같은 프로세스를 사용하여 Int 유형의 Order_Number라는 다른 출력 속성을 추가합니다.
- ReserveOrder 메시드에 대한 사용자 지정 입력/검사점 탭에는 직접 만든 모든 출력 속성이 표시됩니다.

검사점	유효...	값
속성	<input type="checkbox"/>	
1 Total_Price 	<input type="checkbox"/>	= 0
1 Order_Number	<input type="checkbox"/>	= 0

11. ReserveOrder 메시드 모델을 테스트합니다.

- a. REST 서비스 추가 대화 상자에서 ReserveOrder 메시드 노드를 선택합니다.
- b. 도구 모음에서 메시드 실행 버튼  을 클릭합니다.

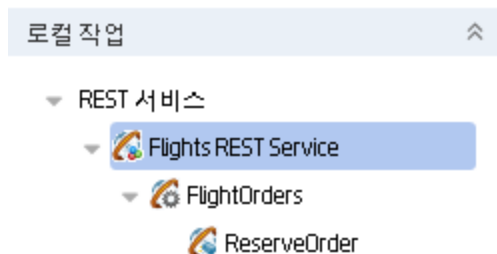
UFT에서 메시드를 실행하고 REST 서비스 추가 대화 상자의 아래쪽 창에 결과를 제공합니다.

연결 시간	0:0:170
응답 본문	<pre> 1 <CreatedOrderParams xmlns="HP.SOAQ.SampleApp" xmlns:i="http://www.w3.org/2001/XMLSchema" 2 <OrderNumber>88</OrderNumber> 3 <TotalPrice>249.6</TotalPrice> </pre>
XML 응답	
응답 유형	Text

12. 서비스 모델과 해당 메시드를 도구 상자에 추가합니다.

REST 서비스 추가 대화 상자에서 확인 버튼을 클릭합니다.

UFT에서 REST 서비스 모델, 해당 리소스 및 메시드를 도구 상자 창의 로컬 작업 노드 아래에 추가합니다.



도구 상자 창에서 메시드를 캔버스로 끌고 단계 속성을 편집할 수 있습니다.

REST 서비스의 프로토타입 모델을 만들었으므로 메서드를 사용하여 테스트를 만들 준비가 되었습니다. 이제 ["연습 7c: REST 서비스 메서드를 사용하여 테스트 만들기"\(194페이지\)](#)로 이동하여 테스트에서 메서드를 사용하십시오.

연습 7c: REST 서비스 메서드를 사용하여 테스트 만들기

"[연습 7b: REST 서비스 구조 만들기"\(189페이지\)](#)에서는 메서드 및 해당 속성을 포함하여 테스트 단계 만들기에서 사용할 REST 서비스의 프로토타입 모델을 만들었습니다.

이 연습에서는 테스트 흐름에서 REST 서비스 모델 메서드를 사용하여 테스트를 만듭니다.


1. UFT를 시작하고 Book Flights 솔루션을 엽니다.

- "[모든 테스트에 대한 솔루션 만들기"\(25페이지\)](#)의 설명대로 UFT를 엽니다.
- 시작 페이지의 **최근에 사용한 솔루션** 영역에서 **Flight Reservation Application**을 클릭합니다.
"[연습 7a: REST 서비스 테스트 만들기"\(189페이지\)](#)에서 만든 **Book Flights REST Service** 테스트가 포함된 Flight Reservation Application 솔루션이 열립니다.

2. 테스트에서 사용할 데이터 원본을 가져옵니다.



- 필요한 경우 **보기 > 데이터**를 선택하여 데이터 창을 표시합니다.
- 데이터 창에서 **새 데이터 원본** 버튼  을 클릭하고 **Excel**을 선택합니다. 새 Excel 데이터 원본 대화 상자가 열립니다.
- 새 데이터 원본 대화 상자의 **Excel 파일 경로** 필드에서 **찾아보기** 버튼을 클릭합니다.
- 열기 대화 상자에서 **<UFT 설치 디렉터리>\samples\Flights Application**에 저장된 응용 프로그램 Excel 파일로 이동하고 **확인**을 클릭합니다.
- 새 Excel 데이터 원본 대화 상자에서 파일 이름을 **Flights_REST**로 지정합니다.
- 원래 위치의 Excel 파일에 링크** 옵션을 선택합니다.
- 확인**을 클릭하여 데이터 원본 정보를 저장하고 Excel 데이터를 테스트에 추가합니다.


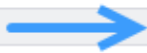







3. ReserveOrder 메서드를 테스트하는 단계를 만듭니다.

- 도구 모음에서 **도구 상자** 버튼  을 클릭합니다. 도구 상자 창이 열립니다.
- 도구 상자의 **로컬 작업** 섹션에서 **Flights REST Service** 노드 아래의 노드를 확장합니다.
- 로컬 작업 노드에서 **ReserveOrder** 단계를 캔버스로 끕니다.
UFT에서는 메서드 이름(**ReserveOrder**)을 사용하여 새 블록을 테스트 흐름에 추가합니다.

4. 메서드의 HTTP 요청 속성을 데이터 원본에 링크합니다.


- 캔버스에서 **ReserveOrder** 단계를 선택합니다.

- b. 속성 창에서 **HTTP** 탭  을 선택합니다.
- c. HTTP 탭의 **요청** 섹션에서 **표** 버튼을 클릭합니다.
- d. 클래스 속성의 값 열에서 **데이터 원본으로 링크** 버튼  을 클릭합니다. 링크 원본 선택 대화 상자가 열립니다.
- e. 링크 원본 선택 대화 상자에서 **데이터 원본 열** 옵션을 선택합니다. **데이터 원본 선택:** 창(왼쪽 창)에 사용 가능한 모든 데이터 원본 목록이 표시됩니다.
- f. 데이터 원본 선택 창에서 **Flights_REST!Input** 노드를 선택합니다. **데이터 선택:** 창(오른쪽 창)에 모든 데이터 열/매개 변수 목록이 표시됩니다.
- g. 데이터 선택: 창에서 **클래스** 노드를 선택하고 **확인**을 클릭합니다.
UFT에서는 클래스 속성 값이 업데이트되어 데이터 소스의 링크가 반영됩니다.

스키마	값
<ul style="list-style-type: none"> FlightOrderDetails <ul style="list-style-type: none">  Class   CustomerName  DepartureDate  FlightNumber  NumberOfTickets 	<ul style="list-style-type: none">  {DataSource.Flights_REST!Input.Class}   John Parker 2115-05-27 1042 1

- h. 다른 HTTP 요청 속성에 대해 같은 프로세스를 반복합니다.
 - o **CustomerName**
 - o **DepartureDate**
 - o **FlightNumber**
 - o **NumberofTickets**

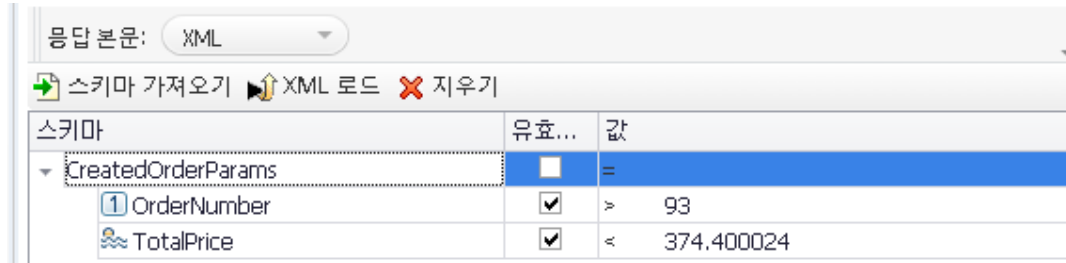
5. **ReserveOrder** 단계에 대한 검사점을 설정합니다.

- a. 캔버스에서 **ReserveOrder** 단계를 다시 선택합니다.
- b. 속성 창에서 **HTTP** 탭  을 선택합니다.
- c. 텍스트 편집기에서 다음 XML을 붙여넣습니다.

```
<?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. 텍스트 편집기에서 파일을 선택한 디렉터리에 **response.xml**로 저장합니다.
- e. UFT의 속성 창에서, **HTTP** 탭의 **응답** 섹션에 있는 드롭다운 목록에서 **XML**을 선택합니다.

- f. 요청 본문 섹션에서 **XML 로드** 버튼을 클릭합니다.
- g. 열기 대화 상자에서 이전 단계에서 저장한 **response.xml** 파일로 이동하고 **열기**를 클릭합니다. UFT에서는 **response.xml** 파일에서 HTTP 탭의 **Response** 본문 섹션으로 XML 스키마를 로드합니다.



- h. **OrderNumber** 속성의 **값** 열에서 드롭다운 화살표를 클릭하고 **>**을 선택합니다.
- i. 값으로 **10**을 입력합니다.
- j. **TotalPrice** 속성에 대한 **값** 열에서 같은 프로세스를 반복하고 값으로 **<** 및 **500**을 입력합니다.

테스트 단계를 만들고 입력 및 검사점 속성을 제공했으므로 이제 테스트를 실행하고 실행 결과를 볼 준비가 되었습니다. "[연습 7d: REST 서비스 테스트 실행](#)"(196페이지)으로 이동하여 자세히 알아보십시오.

연습 7d: REST 서비스 테스트 실행


"[연습 7c: REST 서비스 메서드를 사용하여 테스트 만들기](#)"(194페이지)에서 REST 서비스 모델 메서드를 사용하여 테스트를 만들었습니다. 이 단원에서는 테스트를 실행하고 실행 결과를 보는 방법을 알아봅니다.

1. UFT를 시작하고 Book Flights 솔루션을 엽니다.

- a. "[모든 테스트에 대한 솔루션 만들기](#)"(25페이지)의 설명대로 UFT를 엽니다.
- b. 시작 페이지의 **최근에 사용한 솔루션** 영역에서 **Flight Reservation Application**을 클릭합니다. "[연습 7a: REST 서비스 테스트 만들기](#)"(189페이지)에서 만든 **Book Flights REST Service** 테스트가 포함된 Flight Reservation Application 솔루션이 열립니다.


2. 테스트 반복 수를 설정합니다.

테스트에 여러 행이 있는 데이터 원본이 있으므로 실행할 반복 수를 지정해야 합니다.

- a. 캔버스에서 **테스트 흐름**을 선택합니다. **입력 탭**  이 속성 창에서 열립니다.
- b. 입력 탭에서 **'For'** 루프 옵션을 선택합니다.
- c. **반복 수** 필드에서 8을 입력합니다.


3. 데이터 원본에 대한 데이터 탐색 속성을 설정합니다.

- a. 캔버스에서 테스트 흐름을 선택합니다. **입력/검사점 탭**  이 속성 창에서 열립니다.

- b. 속성 창에서 **데이터 원본 탭**  을 선택합니다. 테스트 흐름과 연결된 모든 데이터 원본 목록이 표시됩니다.
- c. 데이터 원본 목록에서 **Flights_REST!Input** 데이터 원본을 선택하고 **편집**을 클릭합니다. 데이터 탐색 대화 상자가 열립니다.
- d. 데이터 탐색 대화 상자에서 다음 데이터 탐색 속성을 구성합니다.

시작 위치:	First row
위로 이동:	1 rows Forward
종료 위치:	Last row
마지막 행에 도달할 경우	Wrap around

4. 테스트를 실행합니다.

- a. HP Flights Service API 응용 프로그램이 열려 있는지 확인합니다.
- b. 도구 모음에서 **실행 버튼**  을 클릭합니다.
 UFT에서 테스트 단계를 실행하여 데이터 원본의 속성 값을 제공합니다. 테스트 실행 로그가 출력 창에 표시됩니다.
 테스트 실행이 완료되면 실행 결과가 열립니다.

5. 실행 결과를 분석합니다.

- a. 실행 결과 트리 창(왼쪽 창)에서 **수행: Book Flights REST Service** 노드를 확장합니다.
- b. 실행 결과 트리에서 **ReserveOrder** 노드를 선택합니다. 단계 세부 정보가 표시됩니다.
- c. 캡처된 데이터에서 요청 및 응답 정보를 확인합니다. **요청 본문** 및 **응답 본문** 셀에서 링크를 클릭하여 XML 응답 및 요청 정보를 브라우저 창에서 엽니다.

```
<?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>
```

- d. ReserveOrder 노드 아래에서 **검사점** 노드를 선택합니다.
 캡처된 데이터에서 검사점 상태를 확인합니다. 이 경우 실제 값이 예상 값 한도 내에 있으므로 검사점을 통과했습니다.

Name	Result	Property	Actual Result	Evaluation Style	Expected Values	Details
"검사점 1"	✓	"CreatedOrderParams [1]/OrderNumber[1]"	"96"	>	"10"	""
"검사점 2"	✓	"CreatedOrderParams [1]/TotalPrice[1]"	"374.400024"	<	"500"	""

연습 7e: REST 서비스 충돌 해결





"연습 7b: REST 서비스 구조 만들기"(189페이지)에서는 ReserveOrder 메서드를 사용하여 프로토타입 REST 서비스 모델을 만들었습니다. 이 메서드에는 URL 및 속성 이름과 같은 특정 속성이 있습니다. 테스트를 만든 후 서비스 모델의 속성을 변경하면 테스트가 더 이상 모델과 일치하지 않게 됩니다. 따라서 UFT에는 메서드 속성의 변경 사항을 해결할 수 있는 충돌 해결 마법사가 있습니다.

이 단원에서는 충돌 해결 마법사를 사용하여 이러한 차이점을 해결합니다.

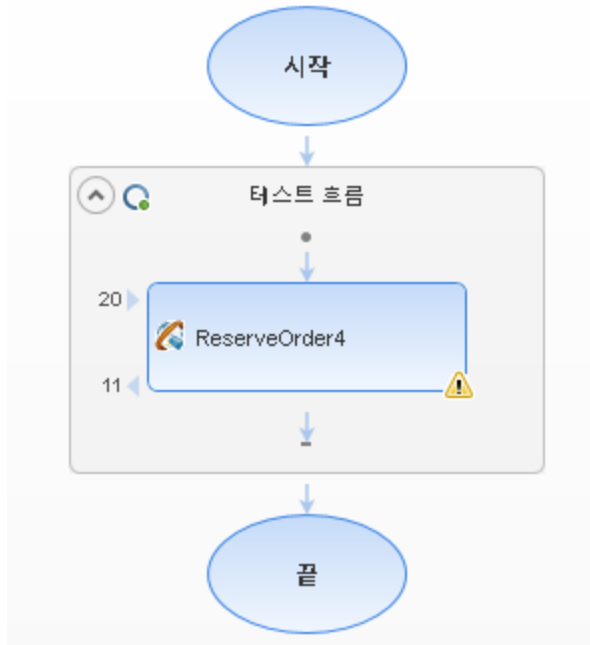
1. UFT를 시작하고 Book Flights 솔루션을 엽니다.

- a. "모든 테스트에 대한 솔루션 만들기"(25페이지)의 설명대로 UFT를 엽니다.
- b. 시작 페이지의 **최근에 사용한 솔루션** 영역에서 **Flight Reservation Application**을 클릭합니다.
"연습 7a: REST 서비스 테스트 만들기"(189페이지)에서 만든 **Book Flights REST Service** 테스트가 포함된 Flight Reservation Application 솔루션이 열립니다.

2. 서비스 모델 속성을 편집합니다.

- a. 도구 모음에서 **도구 상자 버튼** 을 클릭합니다.
- b. 도구 상자의 **로컬 작업** 섹션에서 **REST 서비스** 노드 아래의 노드를 확장합니다.
- c. **Flights REST Service** 노드를 마우스 오른쪽 버튼으로 클릭하고 **서비스 편집**을 선택합니다. REST 서비스 편집 대화 상자가 열립니다.
- d. REST 서비스 편집 대화 상자에서 **ReserveOrder** 노드를 선택합니다. **일반 탭** 이 오른쪽 창에서 열립니다.
- e. 오른쪽 창에서 **사용자 지정 입력/검사점 탭** 을 선택합니다.
- f. 사용자 지정 입력/검사점 탭의 **검사점** 섹션에서 **Total_Price** 속성을 선택하고 **속성 편집 버튼** 을 클릭합니다. 속성 편집 대화 상자가 열립니다.
- g. 속성 편집 대화 상자에서 속성 이름을 TotalPrice로 변경하고 **확인**을 클릭합니다. 검사점 섹션에서 속성 이름이 수정됩니다.
- h. 같은 프로세스를 반복하여 **Order_Number** 속성을 OrderNumber로 변경합니다.
- i. REST 서비스 편집 대화 상자에서 **확인**을 클릭하여 서비스에 대한 변경 사항을 저장합니다.

UFT에서는 REST 서비스 모델에 대한 변경 사항을 저장합니다. 그러나 캔버스의 ReserveOrder 단계에는 경보가 표시됩니다.



이는 서비스 모델과 테스트의 단계 간에 해결해야 하는 충돌이 있음을 나타냅니다.

3. 충돌을 해결합니다.

- a. 캔버스에서 경고 아이콘을 클릭합니다. UFT에서 This step should be resolved. Resolve step. 이라는 메시지를 표시합니다.

- b. 경보 텍스트를 클릭합니다. REST 메서드 확인 마법사가 열립니다.



- c. 단계 선택 화면(첫 번째 화면)에서 **ReserveOrder** 단계의 확인란을 선택하고 **다음**을 클릭합니다. 충돌 해결 화면이 열립니다.

참고: 이 마법사를 사용하면 단계 선택 화면에는 충돌이 있는 모든 단계가 표시됩니다. 따라서 여러 단계 충돌을 한 번에 해결할 수 있습니다.

- d. 충돌 해결 화면의 출력 속성 섹션(아래쪽 영역)에 있는 **변경 후** 상자에서 **Total_Price** 속성(빨간색으로 표시)을 선택합니다.

변경 후:

스키마	유효성 ...	값
속성		
1 Total_Price	<input type="checkbox"/>	255
1 Order_Number	<input type="checkbox"/>	10
1 TotalPrice	<input type="checkbox"/>	0

- e. **제거**를 클릭합니다.
 f. 같은 프로세스를 반복하여 **Order_Number** 속성(빨간색으로 표시)을 제거합니다.
 g. **다음**을 클릭하여 계속 진행합니다. 마침 화면이 열립니다.
 h. 마침 화면에는 서비스 모델의 충돌 상태가 표시됩니다. 이 경우 기존 충돌이 없습니다.

마침을 클릭하여 마법사를 종료합니다. UFT에서는 REST 메서드 확인 마법사에서 선택한 변경 사항으로 테스트를 업데이트합니다. 이 연습에서는 **ReserveOrder** 단계에 대한 출력 속성이 속성 창에서 업데이트됩니다.

검사점	유효...	값
속성		
1 TotalPrice	<input type="checkbox"/>	= 0
1 OrderNumber	<input type="checkbox"/>	= 0

4. **테스트를 저장합니다.**

파일 > 저장을 선택합니다.

8단원: 웹 응용 프로그램 서비스(WADL)의 API 테스트 만들기 및 실행

"3단원: 표준 작업을 사용하여 API 테스트 단계 만들기"(144페이지)에서는 표준 API 작업을 사용하여 테스트를 만드는 방법을 알아보았습니다.

그러나 표준 작업이 응용 프로그램에서 수행하는 프로세스와 일치하지 않는 경우도 있습니다. 그러한 경우에는 UFT로 가져오는 사용자 지정 작업을 사용해야 합니다. 기타 서비스 작업 유형 중 하나는 웹 응용 프로그램 서비스입니다. UFT에서 사용자는 서비스 설명과 해당 메서드를 UFT로 가져와서 테스트에 사용할 수 있습니다.

이 단원에서는 웹 응용 프로그램 서비스 설명을 UFT로 가져와서 사용하는 방법을 알아봅니다.

이 단원의 내용은 다음과 같습니다.

- 연습 8a: 웹 응용 프로그램 서비스에 대한 테스트 만들기 203
- 연습 8b: 웹 응용 프로그램 서비스 모델 가져오기 203
- 연습 8d: 웹 응용 프로그램 서비스 메서드로 테스트 빌드 205
- 연습 8e: 웹 응용 프로그램 서비스 테스트 실행 209

연습 8a: 웹 응용 프로그램 서비스에 대한 테스트 만들기

"모든 테스트에 대한 솔루션 만들기"(25페이지)에서는 항공편 예약 응용 프로그램에 대한 솔루션을 만들고 이 솔루션에 응용 프로그램 사용자 인터페이스의 GUI 테스트 및 응용 프로그램 웹 서비스와 REST 서비스의 API 테스트를 추가했습니다.

이 연습에서는 항공편 예약 응용 프로그램의 웹 응용 프로그램 서비스에 대한 테스트를 만듭니다.

1. UFT를 시작하고 항공편 예약 응용 프로그램 솔루션을 엽니다.

- "모든 테스트에 대한 솔루션 만들기"(25페이지)의 설명대로 UFT를 엽니다.
- 시작 페이지의 **최근에 사용한 솔루션** 영역에서 **Flight Reservation Application**을 클릭합니다. Flight Reservation Application 솔루션이 솔루션 탐색기에서 열립니다.

2. 솔루션에 API 테스트를 새로 추가합니다.

- 도구 모음에서 **추가** 드롭다운 화살표  **추가** 를 선택하고 **새 테스트**를 선택합니다.
- 새 테스트 추가 대화 상자에서 **API 테스트**를 선택합니다.
- 이름** 필드에서 테스트 이름을 Flights WADL로 지정합니다.
- 위치** 필드에서 **찾아보기** 버튼을 클릭하고 **C:\%HOMEPATH%\Unified Functional Testing** 폴더로 이동합니다.
- 추가**를 클릭하여 테스트를 만들고 솔루션에 추가합니다.
Book Flights WADL 테스트가 Flight Reservation Application 솔루션에서 개별 노드로 추가되고 문서 창에서 개별 탭으로 열립니다.

참고: 솔루션이 자동으로 저장됩니다.

테스트를 만들었으므로 이제 웹 응용 프로그램 서비스 모델과 해당 메서드를 사용할 준비가 되었습니다. "연습 8b: 웹 응용 프로그램 서비스 모델 가져오기"(203페이지)로 이동하여 서비스 설명을 UFT로 가져오는 방법을 알아보십시오.

연습 8b: 웹 응용 프로그램 서비스 모델 가져오기

웹 응용 프로그램 서비스를 테스트하기 전에 서비스 설명을 가져와야 합니다(리소스 및 해당 메서드의 구조 포함). 웹 응용 프로그램 서비스 설명은 WADL (Web Application Description Language) 파일에 저장됩니다. UFT에서는 이 파일을 읽고 서비스, 리소스 및 메서드의 계층 구조를 만듭니다(REST 서비스 계층 구조와 유사). 서비스 설명을 가져오면 메서드를 사용하여 테스트를 만들 수 있습니다.

이 연습에서는 항공편 예약 응용 프로그램의 서비스 WADL 파일을 UFT로 가져웁니다.

1. **Flights API 응용 프로그램을 시작합니다.**

"Flight Reservation Application 탐색"(23페이지)의 설명대로 HP Flights Service API 응용 프로그램을 시작합니다.

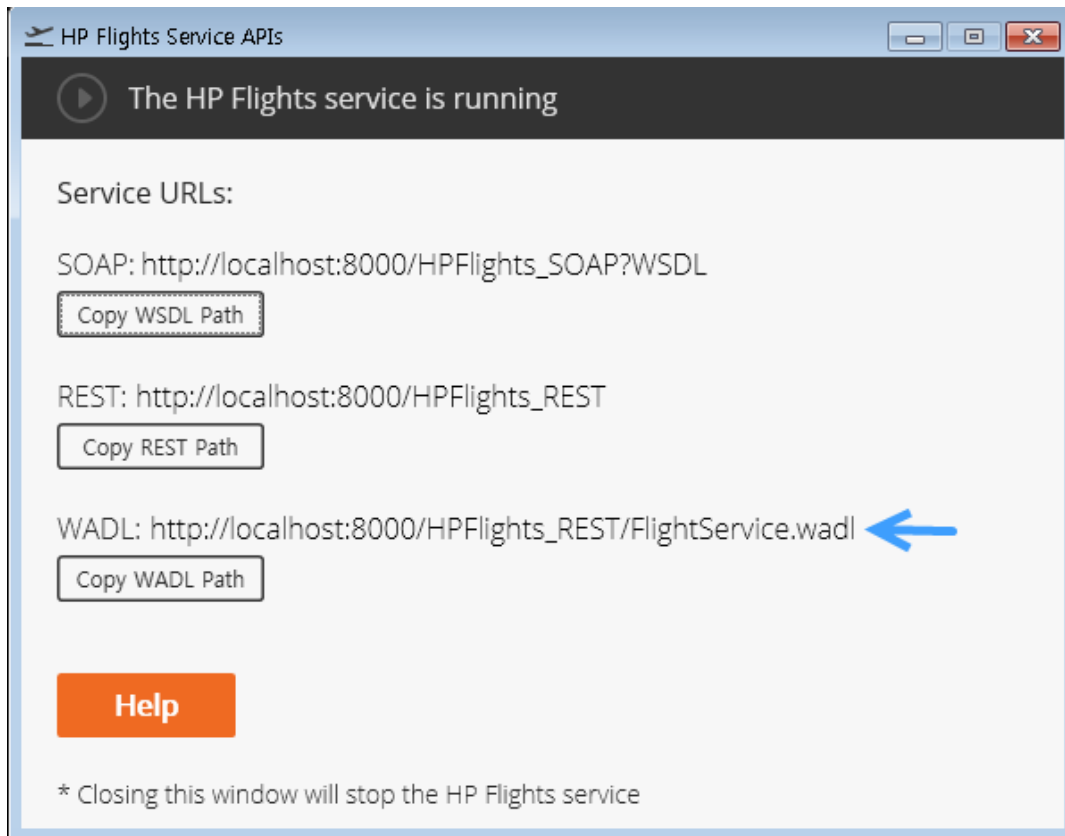
참고: 테스트를 편집하고 실행할 때 UFT에서 응용 프로그램에 액세스할 수 있어야 하므로 자습서를 사용할 때 이 응용 프로그램이 열려 있어야 합니다.

2. **WADL 파일을 가져옵니다.**

- a. UFT의 도구 모음에서 **REST 서비스 추가** 드롭다운 화살표를 클릭하고 **URL에서 WADL 가져오기**를 선택합니다. URL 에서 WADL 가져오기 대화 상자가 열립니다.

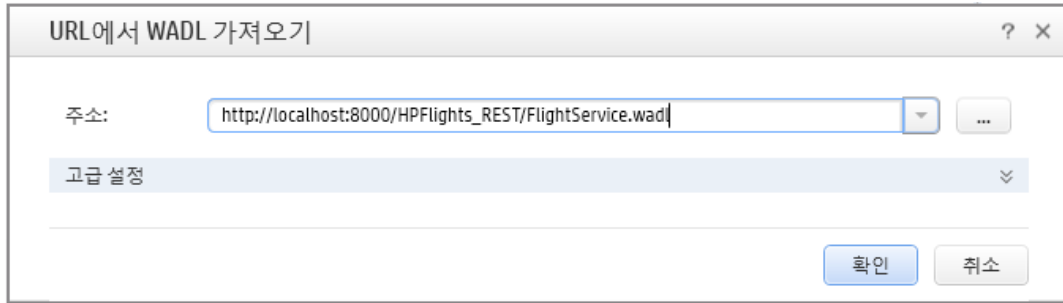
참고: WADL 파일 복사본을 로컬로 저장했으면 WADL 파일을 UFT로 가져올 수 있습니다.

- b. HP Flights Service API 응용 프로그램 창에서 WADL 서비스의 URL을 찾습니다.



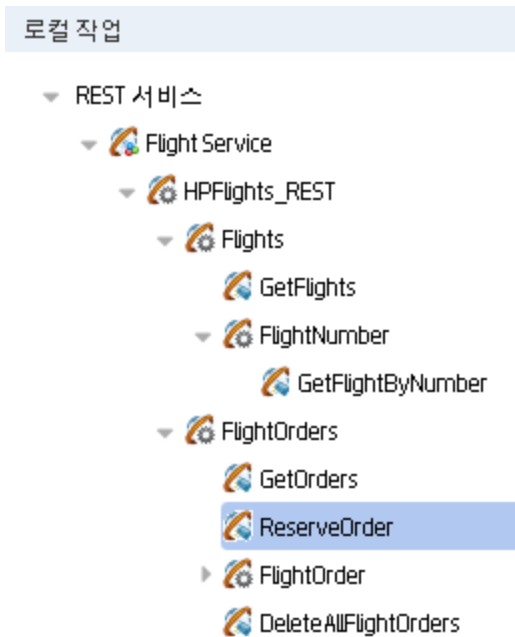
- c. HP Flights Service API 응용 프로그램 창에서 **WADL 경로 복사** 버튼을 클릭합니다. 그러면 WADL 파일의 URL이 저장되므로 URL에서 WADL 가져오기 대화 상자로 복사할 수 있습니다.

- d. UFT의 URL에서 WADL 가져오기 대화 상자에 있는 주소 필드에 응용 프로그램 창에서 복사한 URL을 붙여넣습니다.



- e. **확인**을 클릭하여 서비스 설명을 UFT로 가져옵니다.

서비스 설명이 UFT에 추가되고 해당 리소스 및 메서드 계층 구조가 로컬 작업 섹션의 도구 상자 창에 표시됩니다.



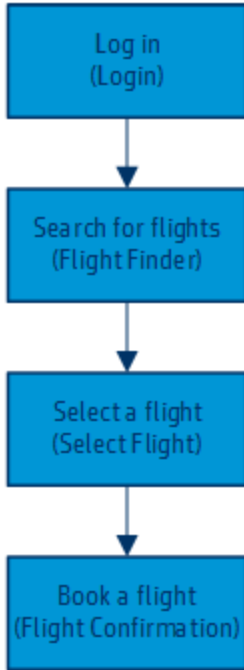
서비스 설명 및 해당 메서드를 UFT로 가져왔으므로 이제 메서드를 사용하여 테스트를 만들 준비가 되었습니다. "[Exercise 9c: Edit the Web Application Service Methods](#)" on page 1으로 이동하여 UFT에서 서비스 설명 메서드를 편집하는 방법을 알아보십시오.

연습 8d: 웹 응용 프로그램 서비스 메서드로 테스트 빌드

"[연습 8b: 웹 응용 프로그램 서비스 모델 가져오기](#)"(203페이지)에서는 웹 응용 프로그램 서비스에 사용된 메서드 설명이 들어 있는 WADL 파일을 가져왔으며, "[Exercise 9c: Edit the Web Application Service](#)"

Methods" on page 1에서는 WADL 파일에 정의되지 않은 추가 속성 정보를 이러한 메서드에 추가했습니다.

"3단원: 테스트에 단계 추가"(44페이지)에서 만든 Book Flights GUI 테스트에서 응용 프로그램 창 순서는 다음과 같습니다.



같은 응용 프로그램의 API 테스트를 만들 때 단계가 가능한 한 응용 프로그램 흐름과 일치하도록 만들 수 있습니다. WADL 파일에서 가져온 메서드 목록에는 다음 중 일부가 포함됩니다.

- **GetFlights**
- **GetFlightOrders**
- **UpdateFlightOrder**
- **DeleteFlightOrder**
- **DeleteAllFlightOrders**
- **ReserveOrder**

사용자 인터페이스 흐름과 일치시키려면 항공편을 찾는 API 테스트 단계를 만든 다음, 고객 입력을 기반으로 한 항공편 주문을 만들어야 합니다.

이 연습에서는 두 가지 테스트 단계 즉, **GetFlights** 및 **ReserveOrder**를 만듭니다.

1. UFT를 시작하고 항공편 예약 응용 프로그램 솔루션을 엽니다.




- a. 필요한 경우 "모든 테스트에 대한 솔루션 만들기"(25페이지)의 설명대로 UFT를 엽니다.
- b. 시작 페이지의 최근에 사용한 솔루션 영역에서 **Flight Reservation Application**을 클릭합니다.

"연습 8a: 웹 응용 프로그램 서비스에 대한 테스트 만들기"(203페이지)에서 만든 Flights WADL 테스트가 포함된 Flight Reservation Application 솔루션이 솔루션 탐색기에서 열립니다.

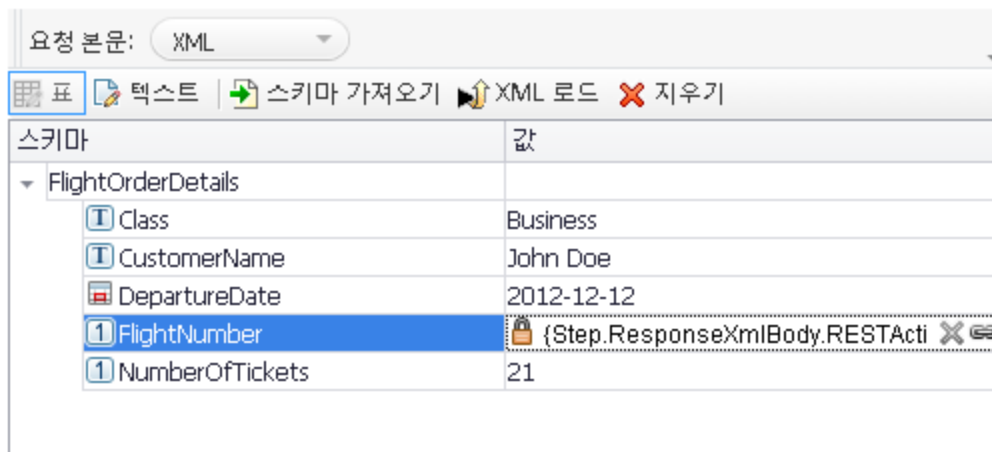
2. 테스트 단계를 만듭니다.

- a. 도구 상자 창의 로컬 작업 섹션에서 **Flights** 노드를 확장합니다(**Flight Service** 및 **HPFlights_REST** 노드 아래에 있음).
- b. Flights 노드에서 **GetFlights** 메서드를 캔버스로 끕니다. UFT에서 **GetFlights**라는 블록을 테스트 흐름에 추가합니다.
- c. FlightOrders 노드에서 **ReserveOrder** 노드를 캔버스로 끕니다. UFT에서 **ReserveOrder**라는 또 다른 블록을 테스트 흐름의 GetFlights 블록 아래에 추가합니다.

3. ReserveOrder 단계의 FlightNumber 속성을 GetFlights 단계의 출력에 링크합니다.


- a. 캔버스에서 **ReserveOrder** 단계를 선택합니다. **입력/검사점** 탭  이 속성 창에서 열립니다.
- b. 속성 창에서 **HTTP** 탭  을 선택합니다.
- c. HTTP 탭의 **요청 본문** 섹션에 있는 **FlightNumber** 속성의 값 셀에서 **데이터 원본으로 링크** 버튼을 클릭합니다. 링크 원본 선택 대화 상자가 열립니다.
- d. 링크 원본 선택 대화 상자에서 **사용 가능한 단계** 옵션을 선택합니다. **단계 선택:** 창(왼쪽 창)이 업데이트되고 사용 가능한 단계 목록이 표시됩니다.
- e. 단계 선택: 창에서 **GetFlights** 단계를 선택합니다. **속성 선택:** 창(오른쪽 창)이 업데이트되고 사용 가능한 속성 목록이 표시됩니다.
- f. 속성 선택: 창에서 **HTTP** 탭  을 선택합니다. HTTP 속성 목록이 표시됩니다.
- g. 속성 목록에서 **FlightNumber** 속성을 선택하고 **확인**을 클릭합니다.

UFT에서 ReserveOrder 단계의 FlightNumber 속성 값을 업데이트하여 GetFlights 단계 출력의 링크를 반영합니다.












4. 테스트에서 사용할 데이터 원본을 추가합니다.

- a. 필요한 경우 **보기 > 데이터**를 선택하여 데이터 창을 표시합니다.




- b. 데이터 창에서 **새 데이터 원본** 버튼  을 클릭하고 **Excel**을 선택합니다. 새 Excel 데이터 원본 대화 상자가 열립니다.
- c. 새 데이터 원본 대화 상자의 **Excel 파일 경로** 필드에서 **찾아보기** 버튼을 클릭합니다.
- d. 열기 대화 상자에서 **<UFT 설치 디렉터리>\samples\Flights Application**에 저장된 응용 프로그램 Excel 파일로 이동하고 **확인**을 클릭합니다.
- e. 새 Excel 데이터 원본 대화 상자에서 파일 이름을 **WADL_Flights**로 지정합니다.
- f. **원래 위치의 Excel 파일에 링크** 옵션을 선택합니다.
- g. **확인**을 클릭하여 데이터 원본 정보를 저장하고 Excel 데이터를 테스트에 추가합니다.

5. GetFlights 단계의 입력 속성을 데이터 원본에 링크합니다.

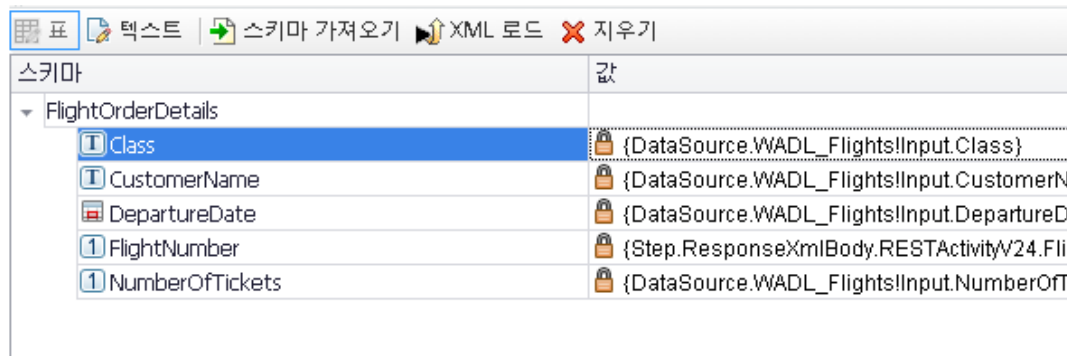
- a. 캔버스에서 **GetFlights** 단계를 선택합니다. **입력/검사점** 탭  이 속성 창에서 열립니다.
- b. 입력/검사점 탭에 있는 **DepartureCity** 속성의 **값** 셀에서 **데이터 원본으로 링크** 버튼  을 클릭합니다. 링크 원본 선택 대화 상자가 열립니다.
- c. 링크 원본 선택 대화 상자에서 **데이터 원본 열** 옵션을 선택합니다. **데이터 원본 선택**: 창(왼쪽 창)에 사용 가능한 모든 데이터 원본 목록이 표시됩니다.
- d. 데이터 원본 선택: 창에서 **WADL_Flights!Input** 노드를 선택합니다. **데이터 선택**: 창(오른쪽 창)에 사용 가능한 모든 데이터 매개 변수(열) 목록이 표시됩니다.
- e. 데이터 선택: 창에서 **DepartureCity** 매개 변수를 선택하고 **확인**을 클릭합니다. UFT에서 속성 창의 **DepartureCity** 속성에 대한 값 열을 업데이트하여 데이터 원본 링크를 반영합니다.
- f. 같은 프로세스를 반복하여 **ArrivalCity** 및 **Date** 속성을 데이터 원본에 링크합니다. UFT에서 **GetFlights** 속성 값을 표시하여 데이터 원본 링크를 보여 줍니다.

입력	값
속성	
DepartureCity	  {DataSource.WADL_Flights!Input.
ArrivalCity	 {DataSource.WADL_Flights!   
Date	 {DataSource.WADL_Flights!Input.De

6. ReserveOrder 단계의 HTTP 요청 속성을 데이터 원본에 링크합니다.

- a. 캔버스에서 ReserveOrder 단계를 선택합니다. **입력/검사점** 탭  이 속성 창에서 열립니다.
- b. 속성 창에서 **HTTP** 탭  을 선택합니다. HTTP 요청 및 응답 속성 목록이 표시됩니다.
- c. **클래스** 속성의 **값** 셀에서 **데이터 원본으로 링크** 버튼  을 클릭합니다. 링크 원본 선택 대화 상자가 열립니다.
- d. 링크 원본 선택 대화 상자에서 **데이터 원본 열** 옵션을 선택합니다. **데이터 원본 선택**:(왼쪽 창)에 사용 가능한 모든 데이터 원본 목록이 표시됩니다.
- e. 데이터 원본 선택: 창에서 **WS_Flights!Input** 노드를 선택합니다. **데이터 선택**: 창(오른쪽 창)에 사용 가능한 모든 데이터 매개 변수(열) 목록이 표시됩니다.
- f. 데이터 선택: 창에서 클래스 열을 선택하고 **확인**을 클릭합니다. UFT에서 클래스 속성의 값을 업데이트하여 데이터 원본 링크를 반영합니다.
- g. 다른 HTTP 요청 속성에 대해 같은 프로세스를 반복합니다.
 - o **CustomerName**
 - o **DepartureDate**
 - o **NumberOfTickets**

모든 속성(FlightNumber 속성 제외)을 데이터 원본에 링크하면 속성 창에 값이 업데이트되어 링크가 반영됩니다.



7. 테스트를 저장합니다.

도구 모음에서 **저장**  을 클릭합니다.

WADL 파일에서 가져온 메서드를 사용하여 테스트를 만들었으므로 이제 테스트를 실행하고 실행 결과를 볼 준비가 되었습니다. "[연습 8e: 웹 응용 프로그램 서비스 테스트 실행\(209페이지\)](#)"으로 이동하여 테스트를 실행하십시오.

연습 8e: 웹 응용 프로그램 서비스 테스트 실행


"[연습 8d: 웹 응용 프로그램 서비스 메서드로 테스트 빌드\(205페이지\)](#)"에서는 이전 연습에서 가져오고 편집한 웹 응용 프로그램 서비스 모델에 대한 메서드를 사용하여 테스트를 만들었습니다.

이 연습에서는 만든 테스트를 실행하여 결과를 확인합니다.



1. **UFT를 시작하고 항공편 예약 응용 프로그램 솔루션을 엽니다.**

- a. 필요한 경우 "모든 테스트에 대한 솔루션 만들기"(25페이지)의 설명대로 UFT를 엽니다.
- b. 시작 페이지의 최근에 사용한 솔루션 영역에서 **Flight Reservation Application**을 클릭합니다.
"연습 8a: 웹 응용 프로그램 서비스에 대한 테스트 만들기"(203페이지)에서 만든 Flights WADL 테스트가 포함된 Flight Reservation Application 솔루션이 솔루션 탐색기에서 열립니다.

2. **테스트 반복 수를 설정합니다.**


- a. 캔버스에서 테스트 흐름을 선택합니다. 입력 탭  이 속성 창에서 열립니다.
- b. 입력 창에서 'For' 루프 옵션을 선택합니다.
- c. 반복 수 필드에서 8을 입력합니다.

3. **데이터 원본에 대한 데이터 탐색 속성을 설정합니다.**

- a. 캔버스에서 테스트 흐름을 선택합니다. 입력/검사점 탭  이 속성 창에서 열립니다.
- b. 속성 창에서 데이터 원본 탭  을 선택합니다. 테스트 흐름과 연결된 모든 데이터 원본 목록이 표시됩니다.
- c. 데이터 원본 목록에서 **WADL_Flights!Input** 데이터 원본을 선택하고 편집을 클릭합니다. 데이터 탐색 대화 상자가 열립니다.
- d. 데이터 탐색 대화 상자에서 다음 데이터 탐색 속성을 구성합니다.

시작 위치:	First row
위로 이동:	1 rows Forward
종료 위치:	Last row
마지막 행에 도달할 경우	Wrap around

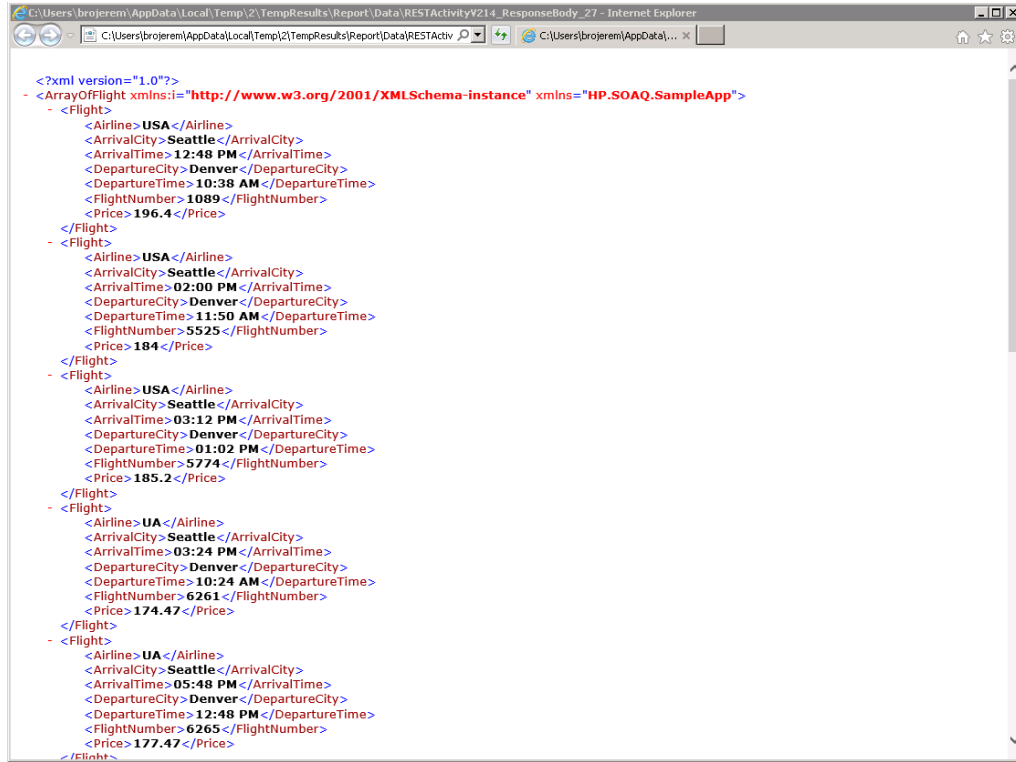
4. **테스트를 실행합니다.**

- a. HP Flights Service API 응용 프로그램이 열려 있는지 확인합니다.
- b. 도구 모음에서 실행 버튼  을 클릭합니다.
UFT에서 테스트 단계를 실행하여 데이터 원본의 속성 값을 제공합니다. 테스트 실행 로그가 출력 창에 표시됩니다.
테스트 실행이 완료되면 실행 결과가 열립니다.

5. **실행 결과를 분석합니다.**

- a. 테스트 흐름에서 **ReserveOrder** 단계를 찾습니다. 실행 결과에 단계 요약이 표시됩니다.

- b. 캡처된 데이터에서 요청 및 응답 정보를 확인합니다. 요청 본문 및 응답 본문 셀에서 링크를 클릭하여 XML 응답 및 요청 정보를 브라우저 창에서 엽니다.



5부: 단일 테스트에서 GUI 및 API 테스트 만들기 및 실행

참고: 자습서의 이 부분을 수행하기 전에 "[자동화 GUI 테스트 만들기 및 실행](#)"(27페이지) 및 "[자동화 API 테스트 만들기 및 실행](#)"(137페이지)에 포함된 자습서 단원을 수행해야 합니다.

응용 프로그램을 테스트할 때 포괄적인 테스트를 수행하려면 사용자 인터페이스(GUI) 및 서비스 레이어(API)를 둘 다 테스트해야 합니다. 이 작업을 수행할 때 해결할 과제 중 하나는 응용 프로그램의 각 부분에 대한 개별 테스트를 유지 관리하고 실행하는 것입니다.

그러나 UFT에서는 응용 프로그램의 GUI 및 API 테스트를 둘 다 만들고 유지 관리해야 하더라도 응용 프로그램의 GUI 및 API를 둘 다 하나의 통합 테스트 실행에서 테스트하는 통합 테스트를 실행할 수 있습니다. GUI 테스트에서 API 테스트를 호출하기만 하면 UFT가 단일 테스트 실행 내에서 응용 프로그램의 두 레이어를 모두 실행합니다. 그리고 나서 테스트 실행이 완료되면 실행 결과에는 단일 보고서로 GUI 및 API 레이어의 수행을 보고하는 통합 보기가 표시됩니다.

이 부분에서는 GUI 및 API 테스트를 둘 다 단일 테스트 실행에 포함하는 테스트를 만들고 실행하는 방법을 알아봅니다.

이 섹션의 내용은 다음과 같습니다.

- 1단원: GUI 및 API 테스트를 함께 실행할 테스트 만들기213
- 2단원: GUI 테스트에서 API 테스트 호출 214
- 3단원: API 테스트를 호출하는 GUI 테스트 실행218

1단원: GUI 및 API 테스트를 함께 실행할 테스트 만들기

이 단원에서는 GUI 및 API 테스트가 함께 포함된 통합 테스트를 실행하기 위해 별도의 테스트를 만듭니다.

1. UFT를 시작하고 Book Flights 테스트를 엽니다.

- a. "모든 테스트에 대한 솔루션 만들기"(25페이지)의 설명대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- b. 시작 페이지의 최근에 사용한 솔루션 영역에서 **Flight Reservation Application**을 클릭합니다. "자동화 GUI 테스트 만들기 및 실행"(27페이지)에서 만든 **Book Flights** GUI 테스트와 "자동화 API 테스트 만들기 및 실행"(137페이지)에서 만든 API 테스트가 포함된 Flight Reservation Application 솔루션이 열립니다.

2. Book Flights 테스트를 Flight Reservation Application으로 저장합니다.

- a. 솔루션 탐색기에서 **Book Flights** 노드를 마우스 오른쪽 버튼으로 클릭하고 **다른 이름으로 저장**을 선택합니다. 다른 이름으로 저장 대화 상자가 열립니다.
- b. 다른 이름으로 테스트 저장 대화 상자에서 **C:\%HOMEPATH%\My Documents\Unified Functional Testing**으로 이동하고 테스트를 Flight Reservation Application으로 저장합니다. 솔루션 탐색기에서 Book Flights 테스트가 새 Flight Reservation Application 테스트로 대체됩니다. Book Flights 테스트는 파일 시스템에 별도로 저장되어 있습니다.

3. Book Flights 테스트를 다시 솔루션에 추가합니다.

- a. **파일 > 추가 > 기존 테스트**를 선택합니다. 솔루션에 테스트 추가 대화 상자가 열립니다.
- b. 테스트를 솔루션에 추가 대화 상자에서 **C:\%HOMEPATH%\My Documents\Unified Functional Testing** 디렉터리로 이동하고 **Book Flights** 테스트를 선택합니다.
- c. **추가**를 클릭하면 Book Flights 테스트가 솔루션으로 돌아갑니다. Book Flights 테스트가 다시 솔루션 탐색기에 표시됩니다.

GUI 및 API 테스트를 함께 실행하기 위한 별도의 테스트가 만들었으므로 이제 두 테스트 유형을 모두 포함하도록 테스트를 빌드할 준비가 되었습니다. "2단원: GUI 테스트에서 API 테스트 호출"(214페이지)로 이동하여 두 테스트 유형을 모두 포함하도록 테스트를 빌드하는 방법을 알아보십시오.

2단원: GUI 테스트에서 API 테스트 호출


GUI 테스트에서 API 테스트를 실행하려면 먼저 API 테스트를 호출해야 합니다. 이 단원에서는 하나의 통합 테스트 실행에서 두 테스트를 모두 실행하기 위해 GUI 테스트에서 API 테스트 호출을 추가하는 방법을 알아봅니다.

1. UFT를 시작하고 Book Flights 테스트를 엽니다.

- a. 필요한 경우 "모든 테스트에 대한 솔루션 만들기"(25페이지)의 설명대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- b. 시작 페이지의 **최근에 사용한 솔루션** 영역에서 **Flight Reservation Application**을 클릭합니다. "1단원: GUI 및 API 테스트를 함께 실행할 테스트 만들기"(213페이지)에서 만든 Flight Reservation Application 테스트가 포함된 Flight Reservation Application 솔루션이 열립니다.


2. API 테스트 호출에 대한 새 수행을 만듭니다.

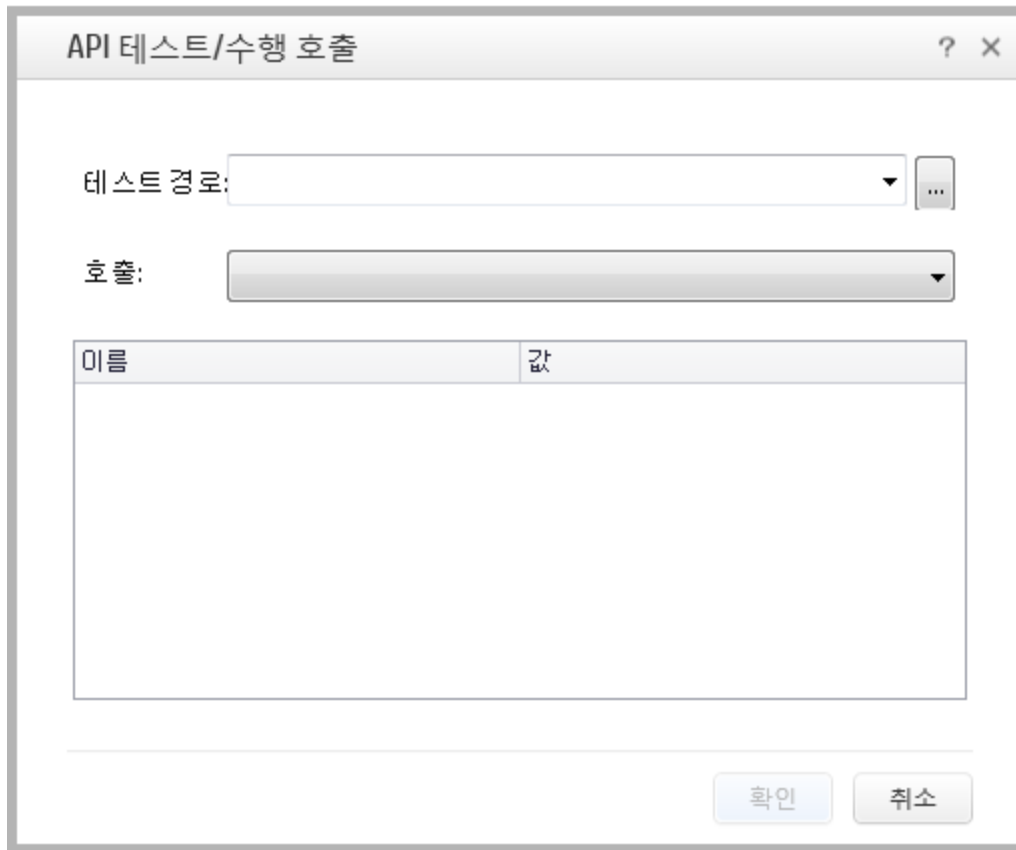
API 테스트를 호출하면 UFT에서는 호출된 API 테스트를 전체적으로 실행합니다. API 테스트 호출을 수행의 개별 단계로 삽입하거나 고유한 독립 수행으로 삽입할 수 있습니다.

- a. 도구 모음에서 **새 수행 호출** 드롭다운 화살표  를 클릭하고 **새 수행 호출**을 선택합니다. 새 수행 호출 삽입 대화 상자가 열립니다.
- b. 새 수행 호출 삽입 대화 상자에서 수행 이름을 API Test Call로 지정합니다.
- c. **테스트의 끝** 옵션이 선택되었는지 확인합니다. 기타 옵션은 기본값으로 유지합니다.
- d. **확인**을 클릭하여 수행을 테스트에 추가합니다.
- e. 캔버스에서 **API Test Call** 수행을 마우스 오른쪽 버튼으로 클릭하고 **위로 이동**을 선택합니다. 테스트 흐름에서 API Test Call 수행이 **Flight Confirmation** 수행 위로 이동합니다.

3. API 테스트 호출을 추가합니다.

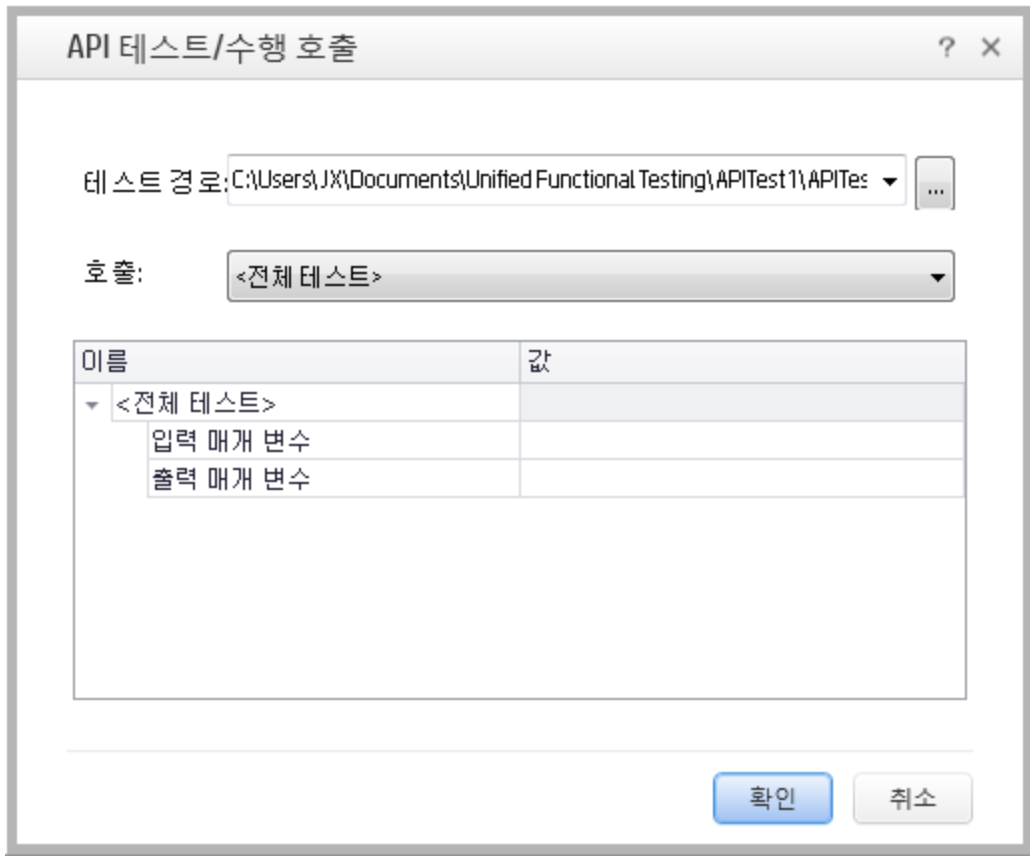
- a. 캔버스에서 **API Test Call** 수행 블록을 두 번 클릭합니다. 수행이 문서 창에서 개별 탭으로 열립니다.

- b. 도구 모음에서 새 수행 호출 드롭다운 화살표  를 클릭하고 기존 API 테스트/수행 호출을 선택합니다. API 테스트 수행 호출 대화 상자가 열립니다.



- c. API 테스트/수행 호출 대화 상자의 테스트 경로에서 찾아보기 버튼을 클릭합니다. 테스트 열기 대화 상자가 열립니다.
- d. 테스트 열기 대화 상자에서 **C:\%HOMEPATH%\Unified Functional Testing**에 저장된 **Flight Reservation Application** 솔루션 테스트가 있는 폴더로 이동합니다.

- e. 이 디렉터리에서 **Book Flights Web Service** 테스트를 선택하고 열기를 클릭합니다. UFT에서 테스트 이름 및 해당 매개 변수를 API 테스트/수행 호출 대화 상자에 추가합니다.



참고: 테스트에 출력 매개 변수가 있으면 이 대화 상자를 사용하여 API 테스트 출력 매개 변수를 저장할 위치를 지정할 수 있습니다. 이 경우 API 테스트에는 출력 매개 변수가 없으므로 이 단계를 수행하지 않습니다.

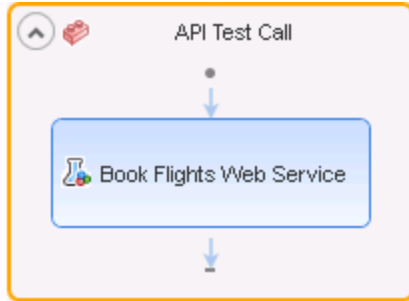
- f. **확인**을 클릭하여 테스트에서 호출을 단계로 추가합니다.
UFT에서 API Test Call 수행의 API 테스트 호출에 대한 단계를 추가합니다. 편집기에는 단계가 다음과 같이 표시됩니다.

RunAPITest "Book Flights Web Service"

키워드 보기에서는 단계가 다음과 같이 표시됩니다.

항목	작업	값	설명
함수 호출	RunAPITest	"Book Flights Web Service"	"Book Flights Web Service" API 테스트/수행을 실행합니다.
+ 새 단계			

또한 UFT에서 수행 호출에 대한 시각적 표시기를 캔버스의 API Test Call 수행에 추가합니다.



4. 테스트를 저장합니다.

파일 > 저장을 선택합니다.

API 테스트를 호출하는 테스트 단계를 만들었으므로 테스트를 실행할 준비가 되었습니다. ["3단원: API 테스트를 호출하는 GUI 테스트 실행"\(218페이지\)](#)으로 이동하여 테스트를 실행하고 실행 결과를 봅니다.

3단원: API 테스트를 호출하는 GUI 테스트 실행

"2단원: GUI 테스트에서 API 테스트 호출"(214페이지)에서 API에 대한 호출을 기존 GUI 테스트에 추가했으므로 항공편 예약 응용 프로그램의 통합 테스트를 하나의 테스트 실행으로 실행할 수 있습니다.


이 단원에서는 테스트를 실행하고 결과를 확인합니다.

1. UFT를 시작하고 Book Flights 테스트를 엽니다.

- 필요한 경우 "모든 테스트에 대한 솔루션 만들기"(25페이지)의 설명대로 UFT를 엽니다. WPF Add-in이 로드되었는지 확인합니다.
- 시작 페이지의 **최근에 사용한 솔루션** 영역에서 **Flight Reservation Application**을 클릭합니다. "1단원: GUI 및 API 테스트를 함께 실행할 테스트 만들기"(213페이지)에서 만든 Flight Reservation Application 테스트가 포함된 Flight Reservation Application 솔루션이 열립니다.

2. Book Flights Web Service 테스트에 대한 반복 수를 업데이트합니다.

Book Flights Web Service 테스트를 만들 때에는 여러 반복이 포함된 테스트를 실행했습니다. 그러나 Book Flights Web Service 테스트를 호출하는 Flight Reservation Application 테스트에는 하나의 테스트 반복만 있습니다. 따라서 같은 수의 반복을 실행하도록 API 테스트를 수정해야 합니다.


- 솔루션 탐색기의 **Book Flights Web Service** 노드 아래에서 노드를 확장합니다.
- Book Flights Web Service 노드에서 **흐름** 노드를 두 번 클릭합니다. Book Flights Web Service 테스트 흐름이 문서 창에서 개별 탭으로 열립니다.
- 캔버스에서 **테스트 흐름**을 선택합니다. **입력 탭**  이 속성 창에서 열립니다.
- 입력 탭의 **반복 수** 필드에서 숫자를 **1**로 변경합니다.
- 파일 > 저장**을 선택하여 수정된 설정을 저장합니다.

3. Flight Reservation Application 테스트에 대한 실행 설정을 지정합니다.

- 문서 창에서 **Flight Reservation Application** 탭을 선택합니다. Flight Reservation Application 탭이 닫히면 솔루션 탐색기에서 **Flight Reservation Application** 노드를 두 번 클릭하여 엽니다.
- 기록 > 기록 및 실행 설정**을 선택합니다. 기록 및 실행 설정 대화 상자가 열립니다.
- Windows 응용 프로그램** 탭에서 **기록 및 실행 장소**: 및 **아래 지정된 응용 프로그램** 옵션이 선택되었는지 확인합니다. 응용 프로그램 세부 정보는 "4단원: GUI 테스트 실행 및 분석"(69페이지)에서 원래 Book Flights 테스트를 실행한 때부터 저장해야 합니다.
- 확인**을 클릭하여 대화 상자를 닫습니다.

4. Flight Reservation Application 테스트를 실행합니다.

- 테스트를 실행하기 전에 **HP MyFlight Sample Application**(항공편 예약 응용 프로그램 사용자 인터페이스) 창이 닫혔는지 확인합니다.

- b. **HP Flights Service APIs** 창이 열려 있는지 확인합니다.
- c. 도구 모음에서 실행 버튼  을 클릭합니다. 실행 대화 상자가 열립니다.
- d. 실행 대화 상자의 **결과 위치** 탭에서 **새 실행 결과 폴더** 옵션을 선택합니다. 기본 폴더 이름을 유지합니다.
- e. **실행**을 클릭하여 테스트 실행을 시작합니다.

UFT에서는 HP MyFlight Sample Application 창을 열고 GUI 테스트에서 만든 대로 응용 프로그램 개체에서 단계를 수행합니다. 테스트가 API 테스트 호출까지 진행되면 GUI 테스트가 일시 중지되고 API 테스트에 포커스가 설정됩니다.

API 테스트가 실행되는 동안 출력 창에서 API 테스트 실행의 진행 상황을 확인할 수 있습니다.

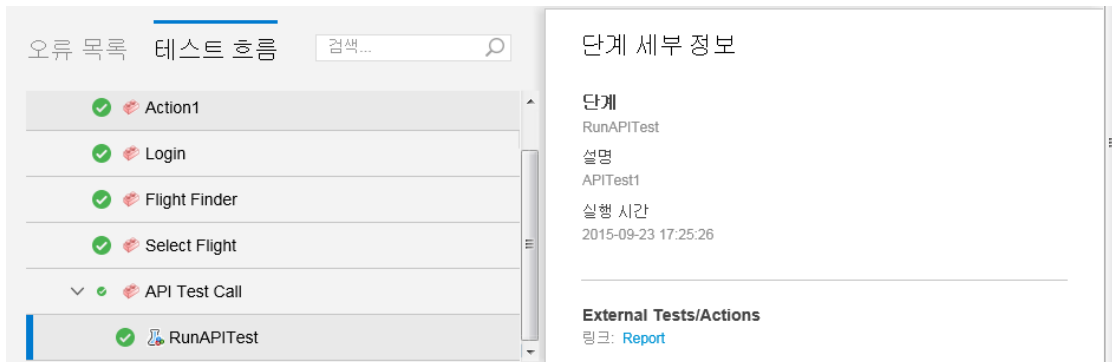
API 테스트 실행이 완료되면 나머지 GUI 테스트가 실행됩니다.

전체 테스트 실행이 완료되면 실행 결과가 열리고 테스트 결과가 표시됩니다.

5. 실행 결과를 봅니다.

실행 결과에 **테스트 흐름**을 표시합니다.

GUI 테스트의 각 수행에 대해 개별 노드가 있는지 확인합니다. 그러나 API 테스트 실행 결과를 실행 결과 일부로 확인할 수도 있습니다.



API 테스트 결과에서 노드를 선택하면 단계 요약에서 단계 세부 정보를 볼 수 있습니다.

6. 실행 결과를 닫습니다.

실행 결과를 본 후 실행 결과가 포함된 탭을 닫습니다.

다음 단계

자동화 GUI 테스트 만들기, 자동화 API 테스트 만들기, GUI 및 API 테스트를 실행하는 테스트 만들기를 포함하여 UFT를 사용하는 방법을 알아보았으므로 UFT를 사용하여 고유한 응용 프로그램을 테스트할 준비가 되었습니다.

고유한 응용 프로그램을 테스트할 때 다음 절차를 사용하는 것이 좋습니다.

1. 응용 프로그램을 분석합니다.

- 개발 환경을 확인합니다. 그러면 관련 UFT Add-in을 로드하고 응용 프로그램 개체에 대한 지원을 제공할 수 있습니다.
- 사용자가 수행할 비즈니스 프로세스를 확인합니다. 이에 따라 테스트와 수행을 계획합니다.
- 테스트 구성 방법 및 포함할 작업을 결정합니다. 테스트 목표를 고려하고 응용 프로그램과 UFT가 테스트 요구 사항에 맞게 설정되었는지 확인합니다.

이 단계에서 응용 프로그램을 테스트할 때 사용할 골격 테스트 및 수행을 만들 수 있습니다.

2. 테스트 인프라를 준비합니다.

테스트의 개체를 저장하는 방법을 결정합니다. 각 수행에 대한 개체를 해당하는 로컬 개체 저장소에 저장하거나, 각 수행에 대한 개체를 하나 이상의 공통(공유) 개체 저장소에 저장할 수 있습니다. 여러 수행에 동일한 공유 개체 저장소를 사용할 수도 있습니다.

- **테스트가 처음인 경우** 개별 수행에 로컬 개체 저장소를 사용할 수 있습니다. 이 설정이 기본 설정이고 모든 개체가 각 수행의 로컬 저장소에 자동으로 추가됩니다.
- **테스트가 친숙한 경우** 하나 이상의 수행에 사용할 수 있는 공유 개체 저장소를 사용하는 것이 가장 효율적일 수 있습니다. 개체 정보가 하나의 중앙 위치에 보관되므로 응용 프로그램의 개체가 변경될 때 이 단일 위치에서 여러 테스트의 여러 수행에 대해 개체를 업데이트할 수 있습니다.

이 자습서에서는 설명되지 않지만 로컬 개체 저장소에서 공유 개체 저장소로 테스트 개체를 내보내고 개체 저장소를 병합할 수도 있습니다.

함수 라이브러리를 만들어 UFT 기능을 향상할 수도 있습니다.

자세한 내용은 *HP Unified Functional Testing User Guide*를 참조하십시오.

3. 테스트를 빌드합니다.

테스트 단계를 만드는 동안 사용자가 응용 프로그램을 탐색할 때 수행해야 하는 단계를 추적합니다.

4. 테스트를 향상합니다.

- 검사점을 추가하여 페이지, 개체, 텍스트 문자열 또는 테이블 셀의 특정 값을 검색합니다.
- 테스트의 고정 값을 매개 변수로 바꿔서 응용 프로그램에서 여러 데이터 집합을 사용하여 동일한 작업을 어떻게 수행하는지 확인합니다.

테스트에 논리를 추가하는 프로그래밍 문, 조건 문 및 루프 문을 사용하여 테스트를 더욱 향상할 수 있습니다. 자세한 내용은 *HP Unified Functional Testing User Guide*를 참조하십시오.

5. 테스트를 디버깅합니다.

테스트를 디버깅하여 테스트가 원활하게 중단 없이 작동하는지 확인합니다. 자세한 내용은 *HP Unified Functional Testing User Guide*를 참조하십시오.

6. 테스트를 실행합니다.

응용 프로그램에서 테스트를 실행하여 응용 프로그램이 예상대로 작동하는지 확인합니다.

7. 실행 결과를 분석합니다.

테스트 결과를 검토하여 응용 프로그램에서 결함을 파악합니다. 검사점 또는 매개 변수에 대한 실행 결과에서 찾아야 할 내용을 알아보려면 이 자습서의 해당하는 섹션을 참조하십시오.

Windows 8.X 이상 운영 체제에서의 UFT 액세스

참고: 기본적으로 Windows 8.x 이상의 시작 및 앱 화면은 Internet Explorer를 메트로 모드로 열도록 설정되어 있습니다. 하지만 컴퓨터에서 사용자 계정 컨트롤이 꺼져 있는 경우에는 Windows 8에서 Internet Explorer가 메트로 모드로 열리지 않습니다. 따라서 시작 또는 앱 화면에서 UFT 도움말이나 추가 정보 파일과 같은 HTML 바로 가기를 열려고 하면 오류가 표시됩니다.

이 문제를 해결하려면 Internet Explorer의 기본 동작을 변경하여 메트로 모드에서 열리지 않도록 하면 됩니다. **인터넷 옵션 대화 상자 > 프로그램 탭에서 웹 페이지 링크를 여는 방법을 선택하십시오** 옵션으로 **항상 데스크톱 Internet Explorer로 열기**를 선택합니다. 자세한 내용은

<http://support.microsoft.com/kb/2736601> 및

<http://blogs.msdn.com/b/ie/archive/2012/03/26/launch-options-for-internet-explorer-10-on-windows-8.aspx>를 참조하십시오.

피드백 보내기



이 자습서를 개선할 수 있는 방법을 제안하시겠습니까?

sw-doc@hp.com으로 의견을 보내주시기 바랍니다.

