

# **Desktop, Web and Mobile Testing Tutorials**





STA는 SMARTBEAR사와 비즈니스 파트너 관계를 맺고 있습니다.



## 시작하기 전에,

TestComplete에서는 세 가지 타입의 모듈을 테스트를 할 수 있습니다.

- 데스크탑(Desktop) 모듈 Window 시스템을 기반으로 PC에서 구동되는 어플리케이션을 테스트
- 웹(Web) 모듈 PC에 설치되어 있는 모든 브라우저(예, IE, Chrome, FireFox 등)를 테스트
- 모바일(Mobile) 모듈 Android와 iOS 단말기를 테스트

본 문서는 TestComplete의 사용자에게 TestComplete 사용에 필요한 전반적인 기능들을 설명하고, 주로 사용되는 프로젝트의 기능에 대해 샘플 페이지로 구성된 가이드를 제공합니다. 그리고 각 모듈 (desktop, web, mobile) 별로 프로젝트의 생성 및 수정, 실행에 대한 설명도 포함되어 있습니다.





자동화 테스트와 TestComplete	5
1. 자동화 테스트	5
2. 테스트 타입	5
3. TestComplete 프로젝트와 프로젝트 항목	6
4. TestComplete의 사용자 인터페이스	8
5. TestComplete의 테스트 오브젝트(Object)	9
6. 체크포인트(Checkpoint)와 저장소(Stores)	12

# 데스크탑(Desktop) 어플리케이션 테스트 13 1. 테스트 프로젝트 생성 14 2. 테스트 어플리케이션 설정 15 3. 테스트 프로젝트 설정 18

4. 테스트케이스 생성	 20
5. 테스트 레코딩 분석	32
6. 레코딩 된 테스트케이스 실행	 37
7. 테스트 결과 분석	 39

# 웹(Web) 어플리케이션 테스트 42

1.	테스트 프로젝트 생성	- 42
2.	테스트 모듈 선택	- 44
3.	테스트 프로젝트 설정	- 45
4.	웹 브라우저 설정	47
5.	테스트케이스 생성	- 47
6.	테스트 레코딩 결과 분석	57
7.	레코딩 된 테스트케이스 실행	60
8.	테스트 결과 분석	62
9.	멀티 브라우저 테스트	- 65



# 목 차

<u>안</u> .	드로이드(Android) 어플리케이션 테스트	69
	1. 테스트 환경 준비	70
	2. 테스트 프로젝트 생성	- 74
	3. 테스트케이스 생성	- 75
	4. 레코딩된 테스트케이스 분석	- 89
	5. 테스트케이스 실행	- 92
	6. 테스트 결과 분석	94
	7. 멀티 디바이스 테스트	- 98

iOS 어플리케이션 테스트		100
1. iOS 단말기 준비		101
2. iOS 어플리케이션 준비		103
3. 테스트 프로젝트 생성		104
4. 테스트케이스 생성	~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~	106
5. 레코딩 된 테스트케이스 분석		119
6. 테스트케이스 실행	······	123
7. 테스트 결과 분석		124
8. 멀티 디바이스 설정		126
9. 멀티 디바이스 테스트		128
온라인 지원 정보		130



## 자동화 테스트와 TestComplete

#### 1. 자동화 테스트

소프트웨어 테스팅은 대상 어플리케이션에 어떤 문제점을 가지고 있는지 확인하고 찾아내는 활동 입니다. 테스팅 활동 중에서 우리가 말하는 테스팅과 단순하게 테스트 하는 것과는 차이가 있습니 다. 테스팅이라 함은 우리가 테스트 한 결과에 대해 그것이 기대값과 일치 하는지 여부를 확인하는 데 있습니다. 다시 말하면, 사용자는 테스트 대상에 대해 결과 값들이 단순히 정상적으로 표시되는 것만 확인하고 그치는 것이 아니라 해당 값들이 적절한지에 대해 확인하는 것까지 포함됩니다.

반복 테스트에 포함되는 사항은 다음과 같습니다.

- 기대값에 대한 정의
- 적절한 입력값을 통한 테스트 수행
- 테스트 대상에 대한 기대값과 실행값의 비교 자료
- 테스트 결과(Fail)에 대한 개발자 혹은 매니저에게 보고

자동화 테스트는 인력 자원의 소모 없이 소프트웨어 테스트를 테스팅 도구를 통해 진행하는 것을 말합니다. 그리고 자주 반복되는 테스트 항목에 대해 간혹 테스트를 생략하고 지나침으로 인해 오는 불안감을 해소 시켜 주고, 그 만큼 품질에 대한 확신을 얻을 수 있게 됩니다.

TestCompelete은 테스트 실행에 대한 레코딩 후 테스트케이스 생성, 체크포인트(Checkpoint)에 대 한 설정 및 테스트 자동 실행과 테스트 결과에 대한 로그를 제공합니다. 테스트 레코딩 기능은 TestComplete에서 제공되는 'Recording' 기능을 통해 쉽게 사용할 수 있고, 레코딩 시 진행되는 모 든 시나리오들은 자동으로 레코딩 되어 사용할 수 있습니다. 또한 체크포인트 설정 마법사를 통해 체크포인트(검증포인트) 설정 시 각각의 설정 화면을 통해 쉽게 설정할 수 있습니다.

#### 2. 테스트 타입

TestComplete은 여러 가지 테스트 방법(단위 테스트, 기능 테스트, 이미지 테스트, 회귀(regression) 테스트 등)을 지원합니다. 이와 관련된 자세한 사항은 Help 메뉴에서 'Different Ways of Testing' 섹 션을 참조하시기 바랍니다. 본 문서에서는 가장 많이 사용되는 기능테스트를 설명합니다. 기능 테스 트는 사용자와 어플리케이션 간의 인터페이스 및 각각의 모든 기능들을 말합니다. 그래서 그 기능들 이 예상되는 행위와 맞게 운용되는지를 검증합니다.

일반적으로 기능테스트란 테스트 수행 시나리오, 예를 들어 버튼을 클릭하거나 키보드를 통해 문 자를 입력 하거나 특정 문자열을 반복으로 수행하는 등의 테스트를 실행해서 그 결과 값이 정확한 지 검증하는 행위입니다.

TestComplete에서는 키워드 테스트와 스크립트를 통해 기능 테스트케이스를 생성할 수 있습니다.

#### SMARTBEAR

키워드 테스트를 통해 테스트케이스 생성 방식은 개발 지식이 없이도 쉽게 생성할 수 있습니다. 스 크립트를 통한 테스트케이스 생성 방식은 개발 지식이 어느 정도는 필요합니다. 하지만 약간의 개발 지식이 있다면 좀 더 효율적이고 테스트케이스를 작업하기 어려운 환경에서도 용이하게 테스트케이 스를 생성할 수 있습니다. TestComplete은 VBScript, JScript, DelphiScript, C++Script, C#Script를 지 원하기 때문에 사용자가 원하는 혹은 편한 언어(Language)를 선택해서 사용할 수 있습니다.

#### 3. TestComplete 프로젝트와 프로젝트 항목

TestComplete은 프로젝트 스윗(suit)과 테스트 프로젝트(project)로 구성할 수 있습니다. 테스트케이 스는 프로젝트(project) 하위에 구성되어 있고, 프로젝트에는 테스트케이스를 비롯해서 체크포인트 (checkpoint)의 데이터, 테스트 어플리케이션에 대한 정보 그리고 테스트 수행에 필요한 모든 정보를 포함하고 있습니다. 또한 반복 테스트 및 멀티 테스트를 포함한 모든 테스트 로그도 저장되어 있습 니다.

하나의 프로젝트에는 테스트 어플리케이션에 대한 모든 테스트케이스를 생성할 수 있습니다. 다만 어플리케이션이 복잡하게 이루어졌거나 모듈 별로 구성되어 있다면 사용자 편의에 따라서 각각의 프로젝트로 분리해서 구성할 수 있습니다.

프로젝트 스윗(suit)에서는 하나 혹은 여러 개의 테스트 프로젝트를 관리할 수 있습니다. TestComplete에서는 테스트 프로젝트를 생성하면 자동으로 테스트 스윗(suit)을 생성합니다. 만약 새 로운 프로젝트 스윗(suit)을 생성하면 TestComplete에서는 자동으로 프로젝트 스윗에 따른 테스트 프트젝트를 생성할 수 있는 창이 발생합니다.

프로젝트 워크스페이스(Project Workspace)에는 테스트를 수행하거나 도움을 줄 수 있는 여러 가 지 항목들로 구성되어 있습니다. 그래서 사용자는 TestComplete에서 제공되는 'Project Explorer' 패 널을 통해 프로젝트 혹은 프로젝트 스윗을 관리하거나 프로젝트를 구성하는 항목들을 볼 수 있습니 다.





- 7 -

관련해서 더 자세한 내용은 TestComplete에서 제공되는 'Help' 메뉴의 'About Project Items' 항목을 참조하시기 바랍니다.







### 4. TestComplete 사용자 인터페이스

TestComplete의 메인 화면은 아래와 같습니다.

New + 😕 🍠 🕼 🚬 🔟 🖓 🏷	💧 🛯 🗙 . 🔡 📾 ն	• 🛃 🗐 🚬 🧶 • 🖹 • 🔳	Stop 😺 🗸	
roject Workspace Object Browser				
이 Project Explorer 2 후 [23]	Workspace			70
8	KeywordTests_DrawString	X		
🛛 🔞 Project Suite 'Hello_KeywordTests_Win	Operations 주	👂 Run Test 🛷 Add Variable	Add Parameter	+ + + -
Hello_KeywordTests_Win7	Search 🔎 👻	Item	Operation	Value
KeywordTests_DrawString     KeywordTests_Main     KeywordTests_SelectColor     NameMapping     TestedApps     mspaint     Wind String     Project Suite Logs	Test Actions       *         On-Screen Action       *         Run TestedApp       *         Menu Action       *         Process Action       *         Run Keyword Test       *         Nun Script Routine       *         Run Test       *         Run Code Snippet       *         Call Object Met       *	Comparison of the second	Click Click Click Click wText [Set] Keys Click Click Click wText [Set]	 "Arial" "[Enter]"  "72"
	Find Object If Object Data-Driven Loop Logging Web	RICHEDITSOW     KICHEDITSOW     RICHEDITSOW     RICHEDITSOW     btn_Transparent     SAfx	Keys Click Keys ClickButton Click	"[Enter]" strDraw
- m - +	🛃 Test Steps 🛄 Variables	C Parameters		
roject Explorer Code Explorer	📑 Test Visualizer			6

위와 같이 TestComplete은 사용자 인터페이스가 몇 개의 패널로 이루어져 있습니다. 화면의 왼쪽 에 'Project Explorer' 패널에는 프로젝트와 프로젝트 스윗에 대한 정보가 표시됩니다. 해당 패널에서 프로젝트나 테스트케이스 등을 클릭하면 관련 페이지로 이동됩니다.

화면 우측의 'Workspace' 패널에서는 생성된 테스트케이스에 대해 수정을 하거나 테스트 결과를 확인할 수 있습니다. 위의 'Workspace' 패널에서 보이는 항목들은 'Keyword Test' 방식을 통해 레코 딩 된 화면입니다. 'Workspace' 패널 아래에 'Test Visualizer' 패널에서는 테스트 레코딩 시 캡쳐된 이미지를 표시해 줍니다. 해당 이미지들은 각각의 스크립트들이 어떤 행위를 하는지에 대해 이미지 로 쉽게 인지할 수 있도록 도와줍니다.

TestComplete은 'Project Explorer', 'Workspace', 'Test Visualizer' 패널 외에도 또 다른 패널들이 있 습니다. 예를 들어, 'Watch List', 'Locals', 'Breakpoints', 'Call Stack' 패널들은 디버깅 시에 사용되는 패널들입니다. 'To Do' 패널은 실행되어야 할 항목들을 관리하고, 'Code Explorer' 패널은 스크립트들 이 어떤 방향으로 진행될지에 대한 정보를 보여줍니다.

'Object Browser' 패널은 윈도우(OS)에서 사용 가능한 모든 어플리케이션에 대한 정보가 나타납니 다. 모바일 테스트 시에도 모바일 기기가 연결되어 있는 경우 모바일 테스트의 대상이 되는 어플리 케이션에 대해 표시됩니다. 각각의 프로세스(Process)나 윈도우(Window)에 대해 TestComplete을 통 해 어떻게 접근해서 사용해야 하는 지에 대한 속성(Properties)과 메소드(Methods)가 표시됩니다. 다 시 말하면 'Object Browser'는 테스트 가능한 오브젝트(Objects)와 메소드(Methods), 그리고 속성 (Properties)을 보여준다고 할 수 있습니다. 이와 관련해서 자세한 설명은 Help 메뉴의 'Exploring Application Properties' 항목을 참조하시기 바랍니다.

각 패널에 대한 도움말 참조를 위해서는 도움말을 얻고자 하는 패널을 선택한 뒤 'F1' 키를 누르 면 선택된 패널에 대한 도움말 페이지가 자동으로 검색되어 나타납니다.

TestComplete에서 제공되는 툴 바는 Microsoft Visual Studio 혹은 다른 많이 알려진 프로그램과 비슷하게 구성되어 있습니다. 툴 바는 사용 중에 임의로 위치의 이동이 가능하고 자주 사용하지 않 는 버튼은 감추거나 감춰졌던 기능은 다시 표시되게 할 수 있습니다. 이에 대해 자세한 사항은 TestComplete의 Help 메뉴에서 'Working With TestComplete Toolbars and Menus' 섹션을 참조하 시기 바랍니다.

#### 5. TestComplete의 테스트 오브젝트(Object)

'Object Browser' 패널에서는 아래와 같이 오브젝트의 구조를 볼 수 있습니다.



Note: 위의 이미지는 Desktop 모듈을 사용하는 화면을 캡처한 이미지입니다. 웹이나 모바일 모듈에서의 이미지도 위와 비슷합니다.

TestComplete에서는 각 오브젝트들이 트리 구조로 표시됩니다. 트리 구조에서 처음 시작은 'Sys'로 표시되고, 모바일의 경우에는 'Mobile'로 표시됩니다.

'Process' 오브젝트들은 윈도우(OS)에서 실행되는 어플리케이션들이 표시됩니다. 그러므로 'Process' 는 어플리케이션으로 이해하셔도 됩니다.



'Process' 오브젝트에는 괄호 안에 해당 어플리케이션을 실행 할 때 사용되는 이름과 인덱스(Index) 가 표시됩니다.

Project Workspace Object Browser	
The executable name	O Name
Objects (minus extension)	Properties
Sys Process name Process name	Advance
Process("notepad"	🗄 Standar
Window("Note instance id", 1)	_NewE
Window("msctis_statusbar32", "", 1)	ChildCo
	Comma
Window("MSCTFIME UI", "MSCTFIME UI", 1)	CPUUsa
Window("IME", "Default IME", 1)	Exists
Process("Orders")	FileVers
WinEermeObject("MainEerm")	FullNam
	Handlo

각각의 프로세스(Process)들을 하위 오브젝트들을 포함합니다. 하위에 표시되는 오브젝트들은 사용 자의 필요에 따라 사용될 수 있습니다. 'Window'와 각각의 이름들은 어플리케이션 테스트 시 'Methods'나 'Properties'에 따라 구분되어 집니다. TestComplete에서는 두 가지 타입의 어플리케이 션에 대해 모두 사용이 가능합니다.

#### • 블랙박스 어플리케이션(Black-box applications)

내부적으로 메소드(methods)나 속성(properties) 접근이 제한된 어플리케이션에 대해서 블랙박 스 어플리케이션(black-box application)이라고 합니다. 'window'와 함께 표시되는 것은 어플리케 이션의 클래스 이름 혹은 타이틀(title)이거나 인덱스(index)입니다. 해당 클래스들은 표시되는 이 름(name)을 통해서 제어가 됩니다.

Project Workspace Object Browser	
i - T 🚱 🍓 🔂 🥃 🎸 🗟	
Objects V	Properti
E A Sys	Advar
Process("notepad", 2)	
Process("notepad")	🗆 Star
🕞 📃 Window("Notepad", "Untitled - Notepad", 1)	
Window class Sctatuchad 32" 1) Window text or title Index	
Window("MSCTFIME UI", "MSCTFIME UI", 1) Window("IME", "Default IME", 1)	C E
Process("Orders")	Fi







#### • 화이트 박스 어플리케이션(White-Box Applications)

내부적으로 사용되는 오브젝트(object)나 메소드(methods) 그리고 속성(properties)들을 TestComplete에서 가져다 사용할 수 있는 어플리케이션을 화이트박스 어플리케이션 혹은 오픈 (open) 어플리케이션이라고 합니다. 그것들은 'Object Browser'에 아이콘(�)으로 표시됩니다. 아 래 이미지를 참조하세요.

TestComplete에서는 Object 패널에 표시되는 이름을 통해서 해당 오브젝트 혹은 메소드를 규 정짓고 사용합니다. 예를 들어, 만약 MS WinForms Library를 통해서 C#으로 생성된 폼(form)의 이름이 'MainForm'이라면 TestComplete에서는 그 폼을 'WinFormsObject("MainForm"):'으로 사용 합니다.



좀 더 자세한 사항에 대해서는 TestComplete의 Help 메뉴에서 'Naming Objects' 섹션을 참조하 시기 바랍니다.

Note: 몇몇의 어플리케이션 - .NET, WPF, Visual Basic, Java, Web 등은 TestComplete을 통해 서 속성이나 메소드 등을 가져다 사용할 수 있는 오픈(open) 어플리케이션입니다. 더 자 세한 사항은 TestComplete의 Help 메뉴에서 'Open Applications' 항목을 참조하시기 바 랍니다.





#### 6. 체크포인트(CheckPoints)와 저장소(Stores)

일반적으로 테스트를 실행하기 위해서는 여러 값들을 비교해서 확인하게 됩니다. 예를 들어, 사용 자의 제어를 통해 데이터 파일을 전송하게 되면, 사용자는 전송된 데이터가 유효한지 확인을 해야 합니다. 이 때 사용자는 테스트 결과 확인을 위해 정상적인 파일과 테스트 하면서 전송 받은 파일을 비교를 하게 됩니다. 비록 하나의 예를 든 것이긴 하지만 테스트를 하면서 대부분이 이와 같이 비교 해서 확인하는 경우일 것입니다. 이와 같이 모든 테스트 유형(단위/기능/회귀 테스트 등)에서 테스트 의 자동화를 구현하기 위해서는 성공(Pass)과 실패(Fail)를 구분할 수 있는 참조 값이 필요합니다.

TestComplete에서는 사용자가 참조 값의 설정을 쉽게 할 수 있도록 구현되어 있습니다. 사용자는 레코딩(recording) 중 혹은 스크립트에서 직접 체크포인트(checkpoints)를 설정할 수 있습니다. TestComplete에서는 체크포인트를 설정 방법을 여러 가지 형태(이미지/파일/오브젝트(text)/속성 /XML문서/데이터베이스테이블 등)로 제공합니다. 그리고 저장소(Stores)에 해당 자료들이 구분되어 있어 체크포인트로 편리하게 가져다 사용할 수 있습니다. 또한 다른 프로젝트에서 사용된 값(이미지, 파일 등)도 저장소를 통해 사용이 가능합니다. 예외적으로 프로젝트에서 사용된 오브젝트(objects)의 속성(properties)은 저장소를 이용할 수 없습니다.

좀 더 자세한 사항은 TestComplete의 Help 메뉴에서 'About Checkpoints' 항목을 참조하시기 바 랍니다.





## 데스크탑(Desktop) 어플리케이션 테스트

본 섹션에서는 데스크탑 어플리케이션 - PC에서 구동되는 모든 어플리케이션 - 테스트 시 필요한 기본적인 사항들에 대해서 설명합니다. TestComplete에서 테스트 프로젝트를 생성하고 간단한 테스 트 레코딩 및 실행과 결과를 분석하는 방법을 설명합니다. 테스트 하는 방식은 사용자가 테스트케이 스에 따른 시나리오를 레코딩하고, 레코딩 중간에 필요한 검증항목을 TestComplete은 Checkpoint로 설정해서 비교하고 확인 합니다.

#### 테스트 어플리케이션(Orders)

TestComplete에서는 기본으로 저장되어 있는 'Orders'이라는 어플리케이션을 본 예제에 사용합니다. 'Orders' 어플리케이션은 물품을 추가, 수정, 삭제 및 주문하는 기능을 가지고 있습니다.

Eile Orders Repo	ort View				
Customer Name	Product	Quantity	Date	Street	City
John Smith Jr Clare Jefferson Susan McLaren Charles Dodgeson Steve Johns Samuel Clemens Bob Feather Mark Smith	MyMoney FamilyAlbum MyMoney ScreenSaver MyMoney ScreenSaver MyMoney FamilyAlbum FamilyAlbum	1 1 1 1 2 1 1	7/5/2009 5/4/2009 12/5/2008 12/5/2009 4/4/2008 12/12/2009 3/12/2010 2/2/2010	12, Orange Blvd 23, Owk Street 7, Flower Street 45, Stone st. 17, Park Avenue 3, Garden st. 14, North av. 9, Maple Valley	Grovetown, CA Greentown, CA Earlcastle Bringtone, TX Salmon Island Hillsberry, UT Miltown, WI Whitestone, Britis
4	m				•

Orders 어플리케이션의 위치는 아래와 같습니다.

- Win7, Vista, Win Server 2008 혹은 그 이후 버전 <u>C:/Users/Public/Public Documents/TestComplete 10 Samples/Open Applications</u>
- Win XP, Win Server 2003 <u>C:/Documents and Settings/All Users/Shared Documents/TestComplete 10 Samples/Open Application</u>

Note: 몇몇 파일들은 'Documents' 폴더의 'Shared Documents'와 'Public Documents' 폴더에서 관리됩니다.

본 문서에서는 'Orders' 어플리케이션을 C#을 이용한 방법으로 설명합니다.



#### 1. 테스트 프로젝트 생성

이제 테스트 프로젝트를 생성해 보겠습니다.

- ① 먼저 TestComplete에 이미 프로젝트나 프로젝트 스윗(suit)이 생성되어 있다면 종료 하세요. TestComplete의 메인 메뉴에서 'File | Close'를 선택하시면 됩니다.
- ② 그리고 다시 메인 메뉴에서 'File | New | New Project'를 선택하세요. 그러면 아래와 같이 새 로운 프로젝트를 설정하는 마법사가 나타납니다.

0	Create New P	roject	
	Specify the na	ame, location and other attributes of a new projec	t.
	Project name:	Orders	
	Location:	C:\Users\JohnSmith\Documents\TestComplete 10 Projects\	
			More
	Q Learn more a	about these settings	
		Next Fi	nish Cancel

- ③ 마법사의 첫 번째 페이지에서는 프로젝트의 이름과 저장 위치를 설정할 수 있습니다. 프로젝 트 이름에는 'Orders'를 입력합니다. 그리고 TestComplete에서는 자동으로 프로젝트의 경로를 지정해 주기 때문에 굳이 수정하지 않고 사용해도 됩니다. 저장되는 위치의 폴더에는 프로젝 트에 대한 모든 정보(Keyword tests, script, test logs, stores 등)가 저장됩니다. 다른 폴더에 저 장을 원하는 경우 'More' 버튼을 통해 변경할 수 있습니다.
- ④ 'Project name'와 'Location'의 설정을 마치면 'Next' 버튼을 클릭 하세요.

본 문서에서 언급되는 마법사 창 및 테스트 어플리케이션과 구체적인 설정 등은 'Orders' 어플리 케이션을 통해서 설명합니다.

#### 2. 테스트 어플리케이션 설정

TestComplete에 하나의 프로젝트에는 하나 이상의 테스트 어플리케이션이 있습니다. 이는 한 프로 젝트에 테스트 대상이 되는 어플리케이션이 어떻게 테스트가 되는지에 대한 구성도 나타난다고 할 수 있습니다. 또한 테스트 메뉴를 통해 프로젝트에 구성되어 있는 어플리케이션을 실행 시키는 것도 가능합니다. 물론 각각의 프로젝트들은 개별적으로 구성되고, 모든 어플리케이션 또한 하나의 프로 젝트에 제한되지 않고 사용될 수 있습니다

몇 가지 방법으로 통해 테스트 어플리케이션을 리스트에 추가할 수 있습니다.

- 프로젝트를 생성할 때 'Create New Project' 마법사를 통해서 가능합니다.
- 'Project Explorer' 패널에서 context 메뉴를 통해서 언제든지 추가할 수 있습니다.
- 테스트 레코딩 중에 'Run Tested Application' 메뉴를 통해서 추가할 수 있습니다. 어플리케이
   션을 추가하게 되면 레코딩이 끝난 후에 해당 어플리케이션은 어플리케이션 리스트에 추가됩
   니다.

본 문서에서는 프로젝트 생성 과정에서 어플리케이션을 'Create New Project' 마법사를 통해서 추 가해 보겠습니다.

프로젝트 이름과 저장 위치를 설정한 후에 아래와 같이 테스트 대상이 되는 어플리케이션
 의 타입을 설정하는 창이 나타납니다.

Specify	the type of the application under test.
	Generic Windows Application Native, .NET, WPF, Java or other application that can run as a standalone executable.
	Functional testing of web pages Simulate mouse actions and keystrokes on web pages, Flash, Flex or Silverlight applications.
	Android applications Applications (activities) running on physical and virtual Android devices.
	iOS applications Applications running on physical iOS devices.
9	Web service testing Call web services' methods and check their responses.
4	Adobe AIR Out-of-browser Adobe AIR applications.

위 페이지는 테스트 대상이 되는 어플리케이션의 타입을 선택하는 창입니다. 본 문서에서는 C#으로 만들어진 'Orders' 어플리케이션을 테스트하므로 'Generic Windows Application'을 선 택합니다. 만약 WinXP를 사용한다면 'Next' 버튼을 클릭하세요. Vista 이후의 운영체제인 경우 에는 WinXP에서 설정해야할 페이지는 생략되어 넘어갑니다.

#### SMARTBEAR



2) 다음 페이지에서는 테스트 프로젝트에 필요한 어플리케이션을 추가할 수 있습니다.

Autorun Name	Application file or entry point	Add
		Remove

- 'Add' 버튼을 클릭하면 어플리케이션을 테스트하기 위한 파라메타를 설정할 수 있는 'Parameters' 창이 아래와 같이 나타납니다.

🐢 Parameters	X
Specify parameters for Simple mode	
Application:	
Command-line parameters:	
Working folder:	
	OK Cancel Help
	OK Cancel Help

- 위 창의 'Application' 필드에서 폴더 버튼을 클릭하고 'Orders' 파일을 선택합니다.
- 'Orders' 파일의 위치는 아래와 같습니다.
- Win7, Vista, Win Server 2008 혹은 그 이후 버전인 경우, <u>C:/Users/Public/Public Documents/TestComplete 10 Samples/Open Applications</u>



- Win XP, Win Server 2003 일 경우, <u>C:/Documents and Settings/All Users/Shared Documents/TestComplete 10</u> Samples/OpenApplication
- Note: 몇몇 파일들은 'Documents' 폴더의 'Shared Documents'와 'Public Documents' 폴더에 있습니다.
- 'Orders.exe'를 선택하면 'Orders.exe'의 전체 경로와 함께 선택된 파일명이 표시됩니다. 그리 고 나서 파라메타 창에서 'OK' 버튼을 클릭하세요.
- 마법사 창에는 추가된 Order 어플리케이션의 이름과 경로가 리스트에 표시됩니다.
  - 3) 리스트에서 'Autorun' 항목을 체크하세요. 이 항목을 체크 설정하면 TestComplete에서 테스 트를 레코딩 할 때 'Orders' 어플리케이션이 자동으로 실행됩니다. 체크 설정되어 있지 않으 면 레코딩 할 때 어플리케이션을 직접 실행 시켜 주어야 합니다.

Add yo	our tested	application to the project.	
Autorun	Name	Application file or entry point	A <u>d</u> d
V	Orders	C:\Users\Public\Documents\TestComplete 10 Samples\Open Applix	[
			Add Java
			Demove

4) 어플리케이션을 추가하고 'Autorun' 항목을 체크하고 나서 'Next' 버튼을 클릭하세요.

#### 3. 테스트 프로젝트 설정

이전 단계에서 'Orders'라는 샘플 테스트 어플리케이션을 추가 했습니다. 이후에 남은 설정 단계를 진행하겠습니다.

 이플리케이션을 추가하고 나면 아래와 같이 레코딩 혹은 테스트 실행할 때 이미지를 저장 할지를 설정하는 페이지가 나타납니다.

Test Visualizer o	captures images for test actions during test recording and
phybacka	
-	Collect Test Visualizer data during recording
● >>>??	© off
	Capture images only
	Capture images and test object info
<b></b>	Collect Test Visualizer data during test run
	>>> © <u>O</u> ff
	Capture images only
	Capture images and test object info
	You can enable or disable the Visualizer now or at any time later in the test project options.
Learn more	

테스트 이미지 정보는 테스트 시나리오를 레코딩 하거나 실행할 때의 이미지가 저장됩니 다. 초기 설정은 테스트 이미지를 프레임 별로 스크린샷 해서 오브젝트의 이미지를 저장하는 것입니다.

테스트 시나리오가 레코딩 될 때 캡쳐 된 이미지는 나중에 해당 테스트의 시나리오가 어 떻게 구성되는지 이해하는데 도움을 줍니다. 테스트가 실행 될 때 캡쳐된 이미지는 테스트가 실행 당시 테스트 어플리케이션이나 다른 시스템들이 어떤 문제가 있었는지를 확인하는데 도움을 줍니다. 이런 정보들은 나중에 디버깅 자료로 아주 유용합니다. 그렇지만 이런 이미 지들로 인해 프로젝트의 용량이 더 커질 수 있습니다. 그래서 만약 프로젝트의 용량을 작게 하고 싶다면 해당 옵션을 'Off'로 설정하면 됩니다. 이 설정은 나중에 프로젝트 설정에서도 변경할 수 있습니다.

본 문서에서는 설정 옵션을 'Caputre images only'로 진행합니다. 그리고 'Next' 버튼을 클 릭하세요.



2) 아래 마법사 창에서는 프로젝트에 사용될 스크립트 언어(Language)를 선택할 수 있습니다.

0	Create New Project
	Specify the scripting language of your project.
	VB VBScript Visual Basic-like scripting syntax.
	JScript Java-like scripting syntax (similar to the C++ and C# syntax).
	DelphiScript Pascal-like scripting syntax.
	ConnectedApps Script Select this option if you are going to create Connected or Self-Testing Applications.
	Help me choose
	Next Finish Cancel
	18 59.1

3) TestComplete에서 생성되는 모든 프로젝트는 하나의 스크립트 언어(VBScript, JScript, DelphiScript, C++/C#Script)를 사용합니다. 스크립트 언어는 프로젝트에서 사용을 하지 않더 라도 중요합니다. 사용자가 스크립트 언어가 아닌 키워드 테스트만 사용한다고 하더라도 오 브젝트의 파라메타 정보를 사용하기 위해 코드나 스크립트에서 해당 값을 어떤 방식으로 호 출할지를 위해서라도 선택해야 합니다.

또한 스크립트 언어는 테스트에서 사용될 오브젝트의 이름에 대한 포맷을 스크립트나 키 워드 테스트에서 사용되기 위해서도 중요합니다. 왜냐하면 이름에 대한 포맷은 스크립트 언 어에서 사용되는 형태와 맞아야 하기 때문입니다. 예를 들면, VBScript와 JScript에서 '메모장' 의 이름은 'Process("Notepad")'로 사용 됩니다. DelphiScript의 경우에는 'Process('Notepad')' 입니다. C++/C#Script에서는 괄호가 추가 되서 '["Process"]("Notepad")' 형태로 됩니다.

좀 더 자세한 사항은 TestComplete의 Help 메뉴에서 'Selecting the Scripting Language' 항목을 참조하시기 바랍니다.

본 문서에서는 VBScript를 사용할 것이므로 해당 창에서 VBScript를 선택하시기 바랍니다. OS가 Vista 이후 버전이면 해당 설정 창에서 마법사가 종료될 것입니다. OS가 XP 버전이라 면 'Finish' 버튼을 클릭해서 설정을 마치시기 바랍니다.

여기까지 마쳤다면 새로운 프로젝트가 프로젝트 스윗(suit)과 함께 생성되었을 것입니다. 'Project Explorer' 패널을 통해 프로젝트와 프로젝트 스윗(suit)을 확인하실 수 있습니다.



#### 4. 테스트케이스 생성

#### Orders 어플리케이션 테스트

'Orders' 어플리케이션은 'Project Workspace' 패널의 'TestedApps' 카테고리에 있습니다. 아래와 같은 경우, 사용자는 때에 따라서 'Orders' 어플리케이션의 설정 상태가 정확한지 혹은 수정이 되어 야 하는지 등을 확인해야 할 필요가 있습니다.

- 테스트 목적: 테스트 어플리케이션(Orders) 실행 후 'Edit Order' 창에서 기존 항목에 대해 수 정이 되는지, 수정된 사항에 대해 정상적으로 변경이 되었는지 확인합니다.
- 테스트 시나리오: 주문 항목의 수정을 통해 주문 리스트가 변경되었는지 검증하고, 사용자는 시뮬레이션을 통해 어플리케이션에 테스트 시나리오에 따라 진행하고 나서 어떤 항목이 변경 되었는지 확인합니다.
- 테스트 결과 확인: 테스트 시나리오에 따라 변경된 사항이 정상적으로 적용 되었다면 Order 리스트에 해당 사항이 반영될 것입니다. 이 부분의 검증을 위해 기대 값과 실제 값을 비교할 것입니다. 그렇게 하기 위해선 두 개의 값을 비교하는 오퍼레이션을 삽입할 것입니다. 해당 오 퍼레이션은 테스트 로그에서 결과를 비교하고 우리는 그 값이 Pass인지 Fail인지 확인할 수 있 습니다.

이와 관련해서 좀 더 자세한 사항은 TestComplete의 Help 메뉴에서 'Planning Tests' 항목을 참조 하시기 바랍니다.

#### TestComplete를 통한 테스트케이스 생성

TestComplete에서는 두 가지 방법으로 테스트케이스를 생성할 수 있습니다.

- 직접 테스트케이스를 생성
- 테스트 레코딩 방식을 이용해서 생성

직접 테스트케이스를 생성하는 경우 사용자는 필요한 오퍼레이션이나 테스트 시나리오를 스크립 트나 키워드 테스트에 맞춰서 생성해야 합니다. 이 방법은 매우 강력하면서도 테스트케이스를 유동 적으로 사용할 수 있다는 장점이 있습니다. 하지만 그 만큼 테스트케이스를 생성하는데 많은 경험이 필요합니다.

직접 테스트케이스를 생성하는 방법의 단점은 많은 시간을 필요로 하고 많은 다른 문제들이 발생 할 수 있습니다. 예를 들면, 직접 테스트케이스를 생성하기 위해서는 테스트 어플리케이션의 클래스 네임(name)이나 오브젝트를 알아야 합니다. 이런 불편함을 해소하기 위해 TestComplete에서 테스트 레코딩만으로 편리하게 테스트케이스를 생성할 수 있는 기능을 제공합니다. 테스트시나리오를 한번 만 레코딩 하면 TestComplete에서 테스트 시나리오 뿐 만 아니라 스크립트 라인이나 테스트 오브젝 트들도 자동으로서 인지해서 키워드 테스트 스크립트로 자동으로 변환되어 생성됩니다. 이렇게 진행 되는 형태를 '테스트 레코딩(recording a test)'이라고 합니다. 이런 방식은 사용자의 역량이 많지 않 아도 편리하게 사용할 수 있기 때문에 매우 유용한 방식입니다. 그래서 본 문서에서는 테스트 레코 딩 방식으로 가이드를 진행할 것입니다. 다음 섹션에서 좀 더 자세하게 설명 드리겠습니다.

#### TestComplete을 통한 테스트 레코딩

테스트 레코딩을 위해서는 아래와 같이 세 가지 단계가 필요합니다.

1) 사용자는 TestComplete의 메뉴에서 제공하는 Record Keyword Test 혹은 Record Script를 통해해서 레코딩을 시작할 수 있습니다. 또한 시작 페이지에서 'Record a New Test' 버튼을 통해 서도 할 수 있습니다.

TestComplete에서는 여러 가지 방법(keyword test, scripts, low-level procedures, HTTP load testing tasks)으로 테스트 레코딩을 할 수 있습니다. 메인 메뉴에서 제공하는 레코딩 방식은 'keyword test' 방식과 'script' 방식 두 가지가 있습니다. 다른 레코딩 방식은 레코딩이 시작된 후에도 변경할 수 있습니다. 메인으로 사용되는 두 가지 테스트 레코딩 방식으로 레코딩이 시 작된 후에도 몇 개의 추가적인 기능이 제공됩니다.

레코딩이 시작되면 아래와 같이 레코딩 툴 바가 나타납니다.



레코딩 툴 바는 레코딩 중 일시정지와 종료 기능 뿐 만 아니라 레코딩 중에 필요한 기능들 을 제공합니다. 그리고 레코딩 되는 타입(Keyword test, script dode, low-level procedure)을 변경할 수 있습니다.

- 레코딩을 시작한 후에 매뉴얼로 테스트 시나리오를 수행합니다. 예를 들어 테스트 어플리케이 션을 실행하고, 시나리오에 따른 버튼을 클릭하거나 메뉴를 선택하고 텍스트를 입력하는 등의 행위를 합니다.
- 레스트 시나리오 수행을 완료한 후에 툴 바에 있는 종료 버튼()을 클릭해서 레코딩을 종료 합니다.

레코딩과 관련된 자세한 사항은 TestComplete의 Help 메뉴에서 'Recording in TestComplete' 항목을 참조하시기 바랍니다.

#### 'Orders' 어플리케이션을 이용한 테스트 레코딩

키워드 테스트 방식으로 'Orders' 어플리케이션을 레코딩 해 보겠습니다. 테스트 시나리오는 어플 리케이션을 실행하고 데이터를 로딩한 뒤 화면을 통해 버튼을 클릭한 뒤 어플리케이션에서 나타나 는 값을 검증하도록 하겠습니다. Note: 테스트 레코딩 중에 본 문서를 클릭해서 보는 등의 행위를 하지 마세요. 레코딩 엔진은 사용자의 모든 행위를 다 추적하기 때문에 테스트 시나리오 외에 다른 행위가 레코딩 될 수 있습니다.

레코딩에 대한 가이드는 가능하면 레코딩 전에 프린트해서 보시기를 권해드립니다. 혹은 두 개의 모니터가 있다면 다른 모니터에는 레코딩에 대한 설명 자료를 펼쳐 놓고 참조하시기 바랍니다. 레코 딩이 시작되게 되면 TestComplete의 메인 화면은 자동으로 최소화 되고 레코딩이 끝나기 전까지는 화면을 다시 띄울 수 없습니다. 만약 사용자가 레코딩 중에 TestComplete 메인 화면을 원복 시키려 고 하면 'TestComplete is in recording mode and cannot be activated."라는 메시지가 나타날 것입 니다. 이 때 레코딩을 계속 하고자 하면 'Continue' 버튼을 클릭하면 됩니다. 해당 메시지가 나타나 는 시점에는 자동으로 레코딩은 일시정지 됩니다. 그래서 당시에 어떤 수행을 해도 해당 시나리오에 대해서는 레코딩이 되지 않습니다.

이제 레코딩을 시작해 보겠습니다.

 1) 새로운 프로젝트를 생성하면 TestComplete에서는 Keyword 테스트 방식의 테스트케이스가 자 동으로 생성됩니다. 이 테스트케이스에서 레코딩하고자 하면 툴 바에서 'Append to Test' 버튼 (♥)을 클릭하면 됩니다.



레코딩을 시작하면 레코딩 툴 바가 나타날 것입니다. 'Interactive Help' 패널을 보면 해당 메 뉴를 통해 레코딩과 관련된 추가적인 메뉴들을 확인할 수 있습니다.





기본 화면은 아래와 같습니다.



버튼())을 클릭해서 추가 버튼이 보여 지도록 할 수 있습니다.



 레코딩을 시작하게 되면 어플리케이션 리스트에 있는 'Orders'가 자동으로 실행됩니다. 어플 리케이션 설정에서 'Autorun' 항목을 설정했기 때문입니다. 관련해서 자세한 사항은 TestComplete의 Help 메뉴에서 'Defining Applications' 항목을 참조하시기 바랍니다.

만약 'Autorun'의 설정이 해제되어 있어서 'Orders' 어플리케이션을 직접 실행해야 한다면 아 래와 같이 나타나는 버튼을 통해서 실행할 수 있습니다.



또한 프로젝트에 등록되지 않은 어플리케이션이라고 하더라도 파일 매니저를 통해서 등록할 수 있습니다.

TestComplete은 어플리케이션 실행부터 해서 모든 명령이나 행위들이 레코딩 됩니다. 사용 자는 레코딩을 완료한 후에 저장된 스크립트들을 확인할 수 있습니다.





 레코딩을 시작한 후에 테스트 어플리케이션이 실행되기 까지 기다리면 아래와 같은 메인 화 면이 나타납니다.

Orders - Untitled				
<u>File O</u> rders <u>R</u> epo	ort <u>V</u> iew			
) 🚄 🖪 🖻	7 X 🔤 🔭	8-8- 6-6- 6-6-		
Customer Name	Product	Quantity Date	Street	City
	m			
				A 65.7

만약 본 문서가 'Orders' 어플리케이션과 겹쳐진다면 본 문서의 사이즈를 줄이거나 위치를 이동시켜주세요. 그렇지 않으면 레코딩 중에 부득이하게 '일시 정지' 하고 본 문서를 다른 곳으로 이동시켜야 하는 불편함을 초래할 수 있습니다.

- 4) 'Orders' 어플리케이션에서 'File | Open' 메뉴를 선택하세요. 그러면 일반적으로 보여 지는 파일 검색 창이 나타납니다.
- 5) 파일 검색 창에서 'MyTable.tbl' 파일을 선택하세요. 파일 위치는 아래와 같습니다. <u>C:#Users#Public#Documents#TestComplete 10 Samples#Desktop#Orders</u>
  - Note: 몇몇 파일 매니저에서는 공유문서(Shared Documents)나 공용문서(Public Documents) 폴더가 문서(Documents) 폴더로 나타나는 경우도 있습니다.
  - Note: 파일 검색 창에서 마우스를 통해서 파일을 찾는 것보다 경로 입력 필드에 경로를 입력 해서 파일을 검색하는 방법을 추천 해 드립니다. 그렇게 해야 하는 이유는 파일을 한번 검색하고 나서 다시 검색할 때 시작 위치가 다르기 때문에 단순히 마우스를 통해서 선 택하는 경우에는 추후에 테스트 실행 시 문제가 생길 수 있습니다.





6) 'MyTable.tbl' 파일을 선택한 뒤 'Open'을 선택하세요. 그러면 아래와 같이 어플리케이션에 데 이터가 로딩 될 것입니다.

🖳 Orders - C:\Users	Public\Document	s\TestComp	lete 10 Samples\(	Open Applications\M	
<u>File</u> Orders <u>R</u> epo	rt <u>V</u> iew				
Customer Name	Product	Quantity	Date	Street	City
John Smith Jr Clare Jefferson Susan McLaren Charles Dodgeson Steve Johns Samuel Clemens Bob Feather Mark Smith	MyMoney FamilyAlbum MyMoney ScreenSaver MyMoney ScreenSaver MyMoney FamilyAlbum FamilyAlbum	1 1 1 1 2 1 1	7/5/2009 5/4/2009 12/5/2008 12/5/2009 4/4/2008 12/12/2009 3/12/2010 2/2/2010	12, Orange Blvd 23, Owk Street 7, Flower Street 45, Stone st. 17, Park Avenue 3, Garden st. 14, North av. 9, Maple Valley	Grovetown, CA Greentown, CA Earlcastle Bringtone, TX Salmon Island Hillsberry, UT Milltown, WI Whitestone, Britis
•	m				•

- 7) 화면의 리스트에서 'Samuel Clemens'를 클릭하세요.
- 8) 툴 바에서 'Edit order' 버튼(☞)을 클릭하면 아래와 같은 창이 나타납니다.

Product MyMor	iey			<u> </u>
Quantity: 2	÷ , price per unit	\$100	, discount	0%
			Total:	200
Date:	12/12/2009	3		
Customer Name	Samuel Clemens			
Street	3, Garden st.	City:	Hillsberry	, UT
State:	US	Zip:		
Card:	C Visa			
	MasterCard			
	C American Express			
Card No	123456789012			
-	02/02/2012	3		

9) 'Customer Name' 필드를 선택한 뒤 마우스 우측 버튼을 눌러서 '모두 선택(A)' 메뉴를 클릭 하고 'Mark Twain'이라고 입력하세요.

10) 'OK' 버튼을 클릭하세요. 'Order' 어플리케이션 리스트에 수정된 이름으로 업데이트 될 것입





니다.

11) 이제 값을 비교하는 기능을 삽입 합니다. 이는 어플리케이션 리스트에서 'Mark Twain'으로 수정된 이름이 정상적으로 실행 됐는지 검증할 것입니다.

우리는 이것을 '체크포인트(Checkpoint)'라고 부릅니다. TestComplete에서는 여러 가지 타입 의 형태로 설정할 수 있는 체크포인트 기능을 제공합니다. 이와 관련된 자세한 사항은 TestComplete의 Help 메뉴의 'Checkpoints' 섹션을 참조하세요.

대부분의 테스트에서는 'Property checkpoint' 타입을 사용할 것입니다. 이 타입은 어플리케 이션을 제어하는 속성을 가지고 있습니다. 그래서 본 문서에서는 'Property checkpoint' 방식을 이용해서 진행하겠습니다.

- 레코딩 툴 바에서 체크포인트 버튼의 드롭다운 버튼을 눌러서 나타나는 메뉴 중에 속성을 검증할 수 있는 'Create Property Checkpoint' 버튼(☜)을 클릭합니다.





- 아래 캡쳐 사진은 'Property Checkpoint(속성 검증)' 마법사 창입니다. 체크포인트 설정 마법 사는 체크포인트를 설정하기 위해 단계별로 구성되어 있습니다.

Choo	se an object whose property will be compared.
Æ	Drag the target to point to the object
3	Press the left mouse button over the target icon and drag the icon to the desired window or control. Releas the button to select this window or control. Show me how it works.
	Point and fix (Shift+Ctrl+A)
-3	Press this button and then move the mouse pointer to the desired window, control or any other object on screen. Press Shift+Ctrl+A to select the object. Show me how it works.
<u>O</u> bjec	ti
	Highlight
	<b>**</b> :

- 마법사의 첫 번째 페이지에서 타겟처럼 보이는 버튼()을 클릭 하고 드래그 하세요.

마법사 창이 자동으로 최소화 될 때까지 잠시 기다렸다가 고객 리스트(customer list)가 있는 창으로 마우스를 드래그 하세요. 마우스를 드래그 하는 중에 TestComplete에서는 마우스 커서 가 지나가는 곳의 오브젝트를 인식해서 붉은 색 박스로 표시됩니다. 원하는 오브젝트에 마우 스 커서를 위치한 뒤 붉은 박스가 발생한 상태에서 드래그 마우스 버튼을 뗍니다.

		# <b>[m</b> ]			
Customer Name	Product	Quantity	Date	Street	City
ohn Smith Jr Jare Jefferson	MyMoney FamilyAlbum	1	7/5/2009	12, Orange Blvd 23, Owk Street	Grovetown, CA Greentown, CA
usan McLaren	MyMoney	1	12/5/2008	7, Flower Street	Earlcastle
harles Dodgeson	MyMoney	1	12/5/2009	45, Stone st.	Bringtone, TX
teve Johns	ScreenSaver	1	4/4/2008	17, Park Avenue	Salmon Island
ob Feather	FamilyAlbum	1	3/12/2010	14. North av.	Milltown, WI
lark Smith	FamilyAlbum	1	2/2/2010	9, Maple Valley	Whitestone, Br





- 마우스 버튼을 떼고 나면 최소화 됐던 마법사 창이 다시 나타나고, 아래와 같이 선택한 오브
 젝트의 이름과 이미지가 표시됩니다.

	se an object	whose pr	roperty	MIII De	e comp	ared.		
<b>(</b>	Drag the targ	et to point	to the obj	ect				
	the button to se	ouse button o elect this wind	over the tar low or cont	rol. <u>Shor</u>	w me how	the icon to the it works.	e desired win	dow or control. Rele
[]	Point and fix (	(Shift+Ctrl+	A)					
( N)	Press this butto screen. Press S	n and then m hift+Ctrl+A t	ove the mo o select the	use poin object.	ter to the Show me	desired windo how it works.	w, control or	any other object or
Object		99099774458	0.000000000000000000000000000000000000	1997				
Aliases	s.Orders.MainFor	m.OrdersViev	N					Highlight
Previev	N:							
Previev	N:	Customer Name	Product	Quetty	Date	Street	Cay	
Previev	N:	Customer Name John Sridh Jr Class Jafferson	Product MyMoney Family Abum	Quertty	Des. 7/5/209 5/4/209	Street 12. Orange Bird 23. Outs Enset	Oty Grovetson, CA Generation, CA Redentit	
Previev	<b>*:</b>	Customer Name John Sreih, Jr Care Jaffetton Ssaan Miclaren Osafes Dodgeson Sevre Johns	Product My/Money Family Houmey My/Money Science/Saver	Quertity 2 1 1	Dee 7.5-2009 5-4/2009 12.5-2008 12.5-2008 4-4/200	Street 12, Orange Bind 23, Ouk Street 7, Fouce Sheat 45, Storp st. 17, Park Avenue	Chy Grovetown, CA Cherestown, CA Enforcette Bingtone, TX Selhon Rand	
Previev	N:	Castoner Name John Stein, Jr Care Jeffson Suise McLaren Castes Dodgeson Steve Johns Bob Feather Mark Soth	Product My/Koney Famby/Abum My/Koney My/Koney Science/Sparet Famby/Abum	Guertty 1 2 1 1 1	Dete: 7-5-2009 5-4-2009 12-5-2008 4-4-2009 3-4-22009 3-4-22009 3-4-22009 3-4-22010	Street 12. Ortenge Bind 23. Over Street 7. Flower Street 62. Park Avenue 14. Nach av 14. Nach av 14. Nach av	Oty Groetson, CA Greetson, CA Estraatie Brygton, TX Selino, Idano Mitaam, Wi Witeetore, Bite	
Previev	N:	Customer Neme John Softh Jr. Care Jefferson Suine McLeven Onletes Dodgeson Steve Johns Bob Feature Mark Shido	Product My/Money Family/Roum My/Maney Scores/Source Family/Roum Family/Roum	Duartity 2 1 1 1 1	Date: 7:5:2009 5:4:2009 12:5:2008 12:5:2008 14:4:2008 14:3:2009 2:4:2010 2:4:2010	Street 12: Onego Brot 23: One Rated 5: Score al. 17: Path. Average 3: Commun. 14: North an. 8: Maple Volkey	On Gruetsen, CA Gruetsen, CA Besolave, TX Selato, Idad Mazer, W Whitestore, Bite	
Previev	M:	Customer Nerse John Stein, Jr Care Jahrson Sales McLarum Deres Johns Deres Johns Deres Johns Deres Johns Mark Stein	Product My/Money Family/Rumey My/Money Scoreor/Saver Family/Rume Family/Rume Family/Rume	Querty 2 1 1 1 1	Den 7-5-200 5-4-205 12-4-200 12-4-200 2-7-200 2-7-200 2-7-200 2-7-200 2-7-200 2-7-200 2-7-200	Street 12: Onerge Bird 23: Ouk Street 7: Rinner Street 65: Store at 17: Tok Aretra 4: Note Aret 4: Note Aret 8: Mape Valley	. Or Growtown, CA Extraatie Brington, TX Selon Iden Mazen, Wi Wheedone, Bite	
Previev	N:	Contorner Nerne John Stellt & Dates Todgeson Base McLerer Oblies Dodgeson Bere Johns Bab Frather Nerk Stellt	Product My/Money Family/Abum Hy/Money Scheer Claurer Family/Abum Family/Abum	Duartity 1 2 1 1 1 1 1	Den 5-4-2005 12-5-2005 12-5-2005 4-2200 3-12-2005 3-12-2006 3-12-2006 3-12-2006 3-12-2006	Street 12: Orange Bird 23: Owk Street 7: Rome Balat 40: Pask Arenge 3: Ontword 14: North are 8: Maple Volky	Otr Groetson, CA Genetion, CA Editate South of the South of the Masse, W Wheestone, Bite	
Previev	N:	Centerner Nerne John Stefft J. Saan McLaver Ooders Dodgeson Steve shree Mark Sauth Mark Sauth	Product Michiony Michiony Land Michiony Land Michiony Science Generation Michiony Mi	Overtty 1 2 3 1 1 1 1 1 1	Det: 7/5/2009 5/4/2000 3/12/2000 3/12/2000 3/12/2010 3/12/2010	Street 12: Orange Bird 20: Ook Street 7: Finance Batt 19: Josh Annual 10: Josh Annual 14: Jack	. Or Grovetowe, CA Grovetowe, CA Berozale, TX Berozale, T	
Previev	N:	Contarner Name John Solm-Jr. Care, Jahnson Diatas Dodgeson Berre John Berre John Mark Solare Mark Solare	Product Multimary Family Maxim Multimary Science Gener Family Abusit Family Abusit	Querty 2 1 1 1	0m 7x5.2005 12.4.2005 12.4.2005 12.4.2005 12.4.2005 12.4.2005 12.4.2005 12.4.2005 2.4.2005 2.4.2010	Street 12: Orange Bird 20: Ook Street 7: Risses Balt 40: Pask Arenge 3: Online 3: Online 3: All Noth Are 3: Magin Volky	Cry Controll, CA Greeton, CA Entranti Entranti Battone (Internet Malane, W Whitesciele, Bitto	

'Next' 버튼을 클릭하세요.

마법사의 다음 페이지에서는 오브젝트의 속성이 표시됩니다. 이 단계에서는 TestComplete에 서 규정하는 어플리케이션의 속성들이 나타납니다. 예를 들어 테스트 대상 어플리케이션이 C# 으로 개발되어졌다면 속성들은 .NET 클래스로 나타날 것입니다. 지금 예를 들어 진행하고 있 는 것은 기본적인 속성들만 포함하고 있습니다. 모든 사용 가능한 속성들을 보고자 한다면 'View more members(Advanced view)' 링크를 이용하면 됩니다.

TestComplete에서는 속성을 두 개의 그룹으로 구분하고 있습니다. 하나는 윈도우에서 제어 되고 공통적으로 사용되는 속성들을 보여줍니다. 사용자는 그것들을 Standard 링크에서 볼 수 있습니다. 다른 하나는 속성들에 대해 좀 더 구체적인 트리 형태로 나타납니다. 속성들의 이름 은 'w.'로 시작됩니다. 사용자는 Extended 링크에서 확인할 수 있습니다. 데이터의 검증을 위해 서 사용자는 'wItem' 속성을 사용할 것입니다. 그것은 오브젝트에서 트리 형태로 좀 더 분할되 어 나타나는 속성입니다.

리스트에서 'wItem' 속성(Extended 항목에 있음)을 찾은 뒤 'Params' 버튼을 클릭 하세요. 그





럼 아래와 같이 팝업이 발생합니다.

Properties					
Basic view	w <u>view more</u>	e members (Advan	ceo view)		
Enable Enable	rd ad		True		
Height	:u		299		
Name	• 2		WinFormsObject(*	'OrdersView")	
Parent	t		(Object)		
Visible	8		True		
Width			592		
Extende	ed	مى رىمى مەرىپىدىن			
witem		Par	ams J		
- NET	Item	5	Туре		
CIrFL	SubItem	0	C String	rms.ListView	
			Integer		

- 위 창에서 'Mark Twain'에 대해서 아래와 같이 정의 합니다.
  - ① 'Type' 리스트에서 'Integer' 라디오 버튼을 선택합니다.
  - ② 'Item' 필드에 '5'를 입력합니다.(트리 보기에서 '5'는 'Mark Twain'의 인덱스입니다. 인덱 스는 시작이 '0'입니다.)
  - ③ 'OK' 버튼을 클릭합니다.



'OK' 버튼 클릭 까지 진행하면 마법사 창에서는 해당 인덱스의 값을 아래와 같이 수집해서 보여줍니다.

Properties			
Basic view <u>View more membe</u>	rs (Advanced view)		
E Standard	True		
Haight	299		
Name	WinFormsQbject("Orde	rsView")	
Parent	(Object)		
Visible	True		
Width	592		
Extended			
wItem(5, 0)	Params Mark Twain		
wItemCount	8		
🖂 "NET			
ClrFullClassName	System.Windows.Form	s.ListView	

'Next' 버튼을 클릭하세요.



다음 마법사 페이지에서는 속성의 이름을 볼 수 있습니다. 그래서 'Value' 박스에 있는 데이 터를 통해서 값을 비교하고 검증할 수 있습니다.

Specify comparison settings	
specify comparison settings.	
Object:	
Aliases.Orders.MainForm.OrdersView	
Property:	
wItem(5, 0)	
Condition:	
Equals	
Learn more about these settings	

'Finish(마침)' 버튼을 클릭하게 되면 체크포인트 설정이 완료 됩니다. TestComplete은 레코딩 스크립트에 생성된 체크포인트 추가하게 됩니다.

- 12) 'X' 버튼을 눌러서 'Orders' 어플리케이션을 닫으세요. 그럼 변경 사항을 저장할건지 묻는 창 이 나타나는데 '아니요(N)'를 선택하고 닫습니다.
- 13) 레코딩 툴 바에서 Stop 버튼(■)을 눌러서 레코딩을 종료 합니다. 종료 버튼을 누르게 되면 TestComplete은 테스트가 진행된 부분을 저장합니다.

1





#### 5. 테스트 레코딩 분석

레코딩이 완료된 후에 TestComplete은 키워드 테스트 방식으로 레코딩 된 스크립트가 생성됩니다. 생성된 스크립트를 통해서 수정도 가능합니다.

	Item	Operation	Value	Description
Frequently Used	Run TestedApp     Orders     Ant MainForm	Orders		Runs the "Orders" t
Run TestedApp	MainMenu	Click OpenFile	"File Open" "C:\Users\Public\Documents	Moves the mouse cu Opens a file specifie
Process Action	Art OrderForm	ClickItem Click	"Samuel Clemens", "Orders Edit order"	Clicks the 0 subitem Moves the mouse cu
Run Test	Customer	Click ClickR Click	 "Select All"	Clicks at point (102, Clicks at point (102, Moves the mouse cu
Find Object	Customer     SuttonOK     Property Checkpoint	SetText ClickButton	"Mark Twain" Aliases.Orders.MainForm.Or	Enters the text 'Mar Clicks the 'ButtonOK' Checks whether the
Test Steps Variab	es 🚰 Parameters			
Update	Update	Up	date	Update

저장된 테스트 스크립트는 위의 이미지와 비슷하게 구성되어 보여 질 것입니다. 테스트 실행이 다 르게 진행되었다면 위의 스크립트와 조금 다를 수도 있습니다. 예를 들어 Visual C++이나 Delphi로 설정해서 진행되었다면 오브젝트의 이름이나 인덱스 값이 다를 수도 있습니다.

테스트 스크립트는 사용자가 'Orders' 어플리케이션에서 테스트 시나리오대로 수행한 이력이 저장 되어 보여 집니다. TestComplete에서는 저장된 항목들을 '오퍼레이션(operations)'이라고 부릅니다. 그리고 각각 수행된 시나리오들에 대해서 'Test Visualizer' 패널에서 이미지 형태로 저장되어 보여 집니다.



Call Object M	Orders     Orders     MainForm     Orders     digConfirmation     Š btnNo	Close	Clicks the 'btnNo' bu
Test Visualizer	ables 🚰 Parameters		

이 이미지들은 사용자가 테스트 시나리오에 따라 어떤 행위들을 했는지 도움을 줍니다. 위 이미지 들은 사용자의 행위에 대해서도 마우스를 클릭하거나 텍스트를 입력하는 등의 행위만 캡쳐를 합니 다.

만약 사용자가 어떤 레코딩 된 스크립트(오퍼레이션)를 선택하게 되면 그와 연관되는 이미지가 활 성화 돼서 이해하기 쉽게 나타납니다. 이와 관련된 자세한 사항은 TestComplete의 Help 메뉴의 'Test Visualizer' 섹션을 참조하세요.

이미지를 더블 클릭하게 되면 'Visualizer Frame' 패널이 활성화 되면서 좀 더 자세하게 확인할 수 있습니다. 'Visualizer Frame' 패널은 사용자가 수행한 내용을 좀 더 자세하게 확인할 수 있습니다. 예를 들어 줌(zoom) 인(in)과 아웃(out) 기능을 활용할 수 있고, 이미지를 파일로 저장할 수 있습니 다. 이와 관련된 자세한 사항은 'Visualizer Frame' 패널을 선택 후 'Help' 버튼(F1)을 눌러서 확인하 시기 바랍니다.

File Orders Rep	ort View	:: <b>- m</b>			
Customer Name John Smith Jr Clare Jefferson	Product MyMoney FamilyAlbum	Quantity 1 2	Date 7/5/2009 5/4/2009	Street 12, Orange Blvd 23, Owk Street	City Grovetown, CA Greentown, CA
Susan McLaren Charles Dodgeson Steve ohns Samu I Cham ns	MyMoney MyMoney ScreenSaver MyMoney	1 1 1 2	12/5/2008 12/5/2009 4/4/2008 12/12/2009	7, Flower Street 45, Stone st. 17, Park Avenue 3, Garden st.	Earlcastle Bringtone, TX Salmon Island Hillsberry, UT
Mark Smith	FamilyAlbum FamilyAlbum	1	2/2/2010	9. Maple Valley	Whitestone, Briti
۹ 📃	III.				•





테스트 스크립트에서 첫 번째 오퍼레이션은 'Run TestedApp'입니다. 'Run TestApp' 오퍼레이션은 테스트 어플리케이션을 실행하는 기능을 하는데 본 문서에서 레코딩 된 어플리케이션은 'Orders'이 므로 'Orders' 어플리케이션을 실행시키는 오퍼레이션입니다. TestComplete은 테스트 시 어플리케이 션을 자동으로 실행 시키거나 레코딩 툴 바 혹은 다른 UI를 통해서 실행 시키면 이를 'Run TestApp' 오퍼레이션으로 저장합니다.

Item	Operation	Value	Description	*
🕞 🛄 Run TestedApp	Orders		Runs the "Orders" t	
🖯 🗖 Orders				1
🖨 👧 MainForm				
🕂 🐏 MainMenu	Click	"File Open"	Moves the mouse cu	
- 🔂 dlgOpen	OpenFile	"C:\Users\Public\Documents	Opens a file specifie	
MainForm				III
🕆 🖒 OrdersView	ClickItem	"Samuel Clemens",	Clicks the 0 subitem	

다음 오퍼레이션은 'File | Open' 메뉴 항목입니다.

Item	Operation	Value	Description	*
Crimer Run TestedApp	Orders		Runs the "Orders" t	
MainMenu	Click	"File  Open"	Moves the mouse cu	1
→ B dlgOpen	OpenFile	"C:\Users\Public\Documents	Opens a file specifie	Е
TordersView	ClickItem	"Samuel Clemens",	Clicks the 0 subitem	

다음 오퍼레이션은 파일 오픈 팝업을 통해서 파일을 오픈하는 항목입니다.

Item	Operation	Value	Description	
Conternation Cont	Orders		Runs the "Orders" t	
MainMenu	Click	"File Open"	Moves the mouse cu	<u>.</u>
🛛 🕞 dlgOpen	OpenFile	"C: \Users\Public\Documents	Opens a file specifie	
MainForm	ClickItem	"Samuel Clemens",	Clicks the 0 subitem	III.

만약 PC의 운영체제가 Win Vista, Win7 이후 버전이라면 TestComplete은 파일을 오프닝 하는 창에서 수행되는 것들을 모두 레코딩 합니다.

Note: 파일 검색 창에서 마우스를 통해서 파일을 찾는 것보다 경로 입력 필드에 경로를 입력 해서 파일을 검색하는 방법을 추천 해 드립니다. 그렇게 하는 것이 좋은 이유는 파일을 한번 검색하고 나서 다시 검색할 때 보이는 위치가 다르기 때문에 단순히 마우스를 통 해서 선택하는 경우에는 추후에 테스트 실행 시 문제가 생길 수 있습니다.



다음에 나오는 오퍼레이션은 'Orders' 어플리케이션의 메인 창과 폼(form)입니다.

	Item	Operation	Value	Description
	Run TestedApp	Orders		Runs the "Orders" t
	MainForm	Click	"File IOpen"	Moves the mouse ou
		OpenFile	"C: \Users \Public \Documents	Opens a file specifie
	MainForm	ClickItem	"Samuel Clemens",	Clicks the 0 subitem
	A MainMenu	Click	"Orders Edit order"	Moves the mouse cu
	🖳 🏹 Customer	Click		Clicks at point (102,
	📆 Customer	ClickR	01	Clicks at point (102,
	PopupMenu	Click	"Select All"	Moves the mouse cu
-	- 🚯 Customer	SetText	"Mark Twain"	Enters the text 'Mar
	🔁 👘 🖏 ButtonOK	ClickButton		Clicks the 'ButtonOK'
	- Report Checkpoint		Aliases.Orders.MainForm.Or	Checks whether the

스크립트에 나타나는 마우스 이벤트들이나 키 입력 등에 대한 자세한 사항은 TestComplete의 Help 메뉴의 'Simulating User Actions' 섹션을 참조하세요.

그 다음은 레코딩 중에 추가시켰던 값을 검증하는 체크포인트 오퍼레이션입니다.

		and the second s		
- Customer	SetText	"Mark Twain"	Enters the text 'Mar	
🖞 ButtonOK	ClickButton		Clicks the 'ButtonOK'	н
Property Checkpoint		Aliases.Orders.MainForm.Or	Checks whether the	
 🖻 🛄 Orders				
- B MainForm	Close		Closes the 'MainFor	
🖻 📃 dlgConfirmation				
to btnNo	ClickButton		Clicks the 'btnNo' bu	

마지막으로 'Orders' 어플리케이션을 'X' 버튼을 통해 종료하고, 저장 여부를 묻는 팝업에서 'No' 버튼을 클릭한 오퍼레이션입니다.

	Customer	SetText ClickButton	"Mark Twain"	Enters the text 'Mar Clicks the 'ButtonOK'	
- Lamontal	Property Checkpoint		Aliases.Orders.MainForm.Or	Checks whether the	-
	Crders Crders Contern	Close		Closes the 'MainFor	
	ligConfirmation	ClickButton		Clicks the 'btnNo' bu	
					-

위에서 보신 바와 같이, TestComplete은 하나의 윈도우에서 수행된 오퍼레이션들에 대해서 그룹으 로 묶어서 관리합니다. 그룹 관리는 오퍼레이션들에 대한 이해를 좀 더 수월하게 해 주고, 어플리케 이션에 내재되어 있는 오브젝트의 계층에 대한 정보를 제공합니다.



본 문서에서는 하나의 테스트 시나리오에 대한 수행을 레코딩 했습니다. 그래서 스크립트에선 'process' 그룹만 나타나 있습니다. 수행한 시나리오에 대해 그룹화 된 것은 아래와 같습니다.

	Item	Operation	Value	Description	*
Grouping nodes	Run TestedApp	Orders	(111	Runs the "Orders" t	
	MainMenu	Click	"File  Open"	Moves the mouse cu	
	🙀 🖓 dlgOpen	OpenFile	"C:\Users\Public\Documents	Opens a file specifie	E
	C Inet MainForm	ClickItem	"Samuel Clemens",	Clicks the 0 subitem	
	MainMenu	Click	"Orders  Edit order"	Moves the mouse cu	
	Customer	Click		Clicks at point (102,	
	Customer	ClickR		Clicks at point (102,	4
	PopupMenu	Click	"Select All"	Moves the mouse cu	
	Customer	SetText	"Mark Twain"	Enters the text 'Mar	
	ButtonOK	ClickButton		Clicks the 'ButtonOK'	
	Roperty Checkpoint		Aliases.Orders.MainForm.Or	Checks whether the	+

간혹 사용자는 'Object Browser' 패널에서 봤던 이름과 스크립트에 나타나 있는 이름이 달라서 혼 란스러울 수 있습니다. 예를 들면 'Object Browser'에서는 이름이 'Process("Orders")'이고 'Orders'라 고 명했는데, 메인화면에서는 'WinFormsObject("MainForm")이고 'MainForm'이라고 명한 부분이 그 것입니다.

그것은 TestComplete에서는 자동으로 레코딩 중에 오브젝트의 'custom names'을 수집하고 사용하 기 때문입니다. TestComplete은 기본으로 제공되는 이름을 이해하는데 어려울 수 있기 때문에 레코 딩 시 수집된 이름들을 매핑 시킵니다. 매핑된 이름을 사용하는 것은 테스트를 보다 쉽고 이해하기 편하게 만들어 줍니다. 이에 대한 자세한 사항은 TestComplete의 Help 메뉴의 'Name Mapping' 섹 션을 참조하세요.




# 6. 레코딩 된 테스트케이스 실행

이제 사용자 시나리오에 따라 레코딩 된 테스트케이스를 실행해 보겠습니다.

테스트 스크립트를 실행하기 전에 테스트 환경을 레코딩 당시의 환경과 동일하게 설정해 주어야 합니다. 예를 들면 대부분의 테스트는 어플리케이션이 실행 중인 상태에서 시작합니다. 그러므로 시 나리오에 따른 레코딩 전에 사용자는 어플리케이션을 실행한 상태가 되어야 합니다. 본 문서의 경우 에서도 테스트 어플리케이션은 실행 시킨 후에 테스트를 시작 했습니다. 그렇지 않다면 사용자는 테 스트 어플리케이션을 직접 실행 시켜야 합니다.

테스트 실행을 위한 가장 간단한 방법은 툴 바에서 테스트 실행 버튼(▶Run Test)을 클릭하는 것입 니다.

est1 >	Append to Test 🕨 Run Te	est 🛷 Add Va	riable 🔗 Add Parameter	1
I	tem	Operation	Value	C
	Orders	Click	"File Open"	м
	🕂 🕞 dlgOpen 🖨 👧 MainForm	OpenFile	"C:\Users\Public\Documents	s C
	🛱 OrdersView	ClickItem	"Samuel Clemens",	C

그럼 모든 화면 자동으로 사라지고 짧은 시간 내에 테스트가 진행 될 것입니다.

Note: 테스트가 실행되는 동안 마우스를 움직이거나 키를 입력하는 등의 행위를 해서는 안 됩 니다. 그런 행위로 인해 테스트 실행에 오류가 생길 수 있습니다.

테스트 실행 후에 TestComplete은 테스트 결과를 자동으로 수집에서 보여줍니다. 테스트 결과 분 석에 대해서는 다음 섹션에서 해 드리겠습니다.

테스트 실행과 관련해서 몇 가지 더 말씀 드리면,

 테스트케이스가 생성 됐다고 해서 테스트 자체가 완료되었다는 것은 아닙니다. 사용자는 TestComplete을 통해서 테스트를 실행할 수 있습니다. PC에 TestComplete이 설치되어 있지 않다 면 테스트케이스 실행을 위해서는 'TestExecute'라는 프로그램이 필요합니다. 또한 만약 사용자가 스크립트 코드를 외부로 출력해서 사용하고 싶다면 그것은 가능합니다. 더 자세한 사항은 TestComplete의 Help 메뉴의 'Connected and Self-Testing Applications' 섹션을 참조하세요.





- 테스트케이스를 실행하는 동안 TestComplete은 화면 오른쪽 상단에 인디케이터만 표시합니다.

Workspace	h & - Q = . ( <b>A</b> - )	📕 Stop 😡	Show Mobile Screen	iting for the Customer obje
Start Page 🗙 🔒 Te	esti X	F		
Search	Kun T	est   🌾 Add Va Operation	riable / Add Parameter   1	Description
Frequently Used 🔺 Test Actions	Orders			
On-Screen A	MainMenu	Click	"File  Open" "C: \Users\Public\Documents	Moves the mouse cu
Menu Action	Addition MainForm	ClickItem	"Samuel Clemens", "Orders Edit order"	Clicks the 0 subitem
Run Keyword	Customer	Click ClickR		Clicks at point (102, Clicks at point (102,
🐌 Run Test	PopupMenu	Click SetText	"Select All" "Mark Twain"	Moves the mouse cu Enters the text 'Mar
Find Object	Image: Constraint of the second se	ClickButton	Aliases.Orders.MainForm.Or	Clicks the ButtonOK'
Test Steps 🖾 Variab	les ੋ Parameters			16

- 인디케이터는 테스트 실행에 대한 상황을 메시지로 표시해 줍니다.
- TestComplete은 테스트케이스가 모두 완료될 때까지 실행됩니다. 만약 사용자가 실행을 멈추고자 한다면 인디케이터에서 'Stop' 버튼(•)을 클릭하거나 TestComplete 메인 화면에서 'Test | Stop'를 선택하면 됩니다.

데스트 실행 중에 일시정지 하고 싶다면 정지 버튼()을 클릭하세요. 일시정지 되어 있는 동안 에는 사용자가 테스트 외에 다른 행위를 해도 상관없습니다. 예를 들면 사용자는 일시정지 시키고 나서 'Watch List' 패널이나 'Locals' 패널 혹은 'Evaluate' 창에서 테스트 로그를 체크 하거나 사용 된 오브젝트를 확인하는 등의 행위를 할 수 있습니다. 관련해서 자세한 사항은 TestComplete의 Help 메뉴의 'Debugging Tests' 섹션을 참조하세요.

- 테스트 실행을 위해 툴 바에서 'Run Test' 버튼을 누릅니다. 이 것은 테스트 실행을 위한 여러 가 지 방법 중에 하나입니다. 사용자는 'Project Explorer' 패널을 통하거나 그 외에 다른 방법을 통해 서도 테스트를 실행 시킬 수 있습니다. 또한 테스트 스크립트 패널에서도 테스트를 실행시킬 수 있습니다.

TestComplete에서 테스트 실행과 관련된 정보는 TestComplete의 Help 메뉴의 'Running Tests' 섹 션을 참조하세요.

# 7. 테스트 결과 분석

TestComplete은 수행된 모든 테스트에 대해서 테스트 로그를 보관합니다. 테스트 로그를 확인할 수 있는 링크는 'Project Explorer' 패널의 아랫부분에 'Project Suite Logs | Orders Log' 카테고리에 서 확인할 수 있습니다. 이 카테고리는 프로젝트나 프로젝트 스윗(suit)에서 기본으로 제공되는 결과 보관 장소 입니다. 하나의 결과(한 항목에 대해 노드(node)라고도 합니다.)는 테스트를 실행하는 횟 수와 부합 됩니다. 노드의 왼쪽에 있는 이미지는 테스트가 성공했는지 아닌지를 표시해 줍니다.



TestComplete은 테스트 실행이 완료 되면 자동으로 테스트 로그를 생성합니다. 그렇다는 것은 테 스트 결과는 테스트가 끝나기 전까지는 표시되지 않는다고 말씀드릴 수 있습니다. 만약 테스트 중간 에 테스트 결과를 보고 싶다면 테스트 중에 일시정지(pause)를 하고 볼 수는 있습니다.

본 문서에서는 이전에 한 개의 테스트에 대해 한 번만 실행을 했기 때문에 위에 보시는 바와 같이 'Project Explorer' 패널에 한 개의 결과 로그만 나타납니다. 그리고 기본적으로 TestComplete은 테스 트 실행이 완료되면 테스트 결과 페이지도 'Workspace' 패널에 자동으로 나타납니다. 만약 다른 테 스트 로그를 확인하고 싶다면 해당 로그를 더블 클릭하거나, 마우스 우클릭 후 'Open' 메뉴를 선택 하면 됩니다.



og Items	Test Log		0	9 🔀
Keyword Test Log [Test1]	🔇 🖉 Error 🛕 🕼 Warning 🜵 🕼 Message 🥝 [ Search			P
	Type Message	Time 💧	Pri	H L
	→ → The application "C:\Users\Public\Documents\TestComplete 10 Samp	9:13:13	Nor	
	- 📀 The menu item 'File Open' was dicked.	9:13:14	Nor	
	The file 'MyTable.tbl' was selected in the Open dialog.	9:13:15	Nor	
	The list view item ('Samuel Clemens', 0) was clicked with the left mo	9:13:15	Nor	
	- O The menu item 'Orders Edit order' was dicked.	9:13:16	Nor	
	- 📀 The window was clicked with the left mouse button.	9:13:16	Nor	
	Or The window was clicked with the right mouse button.	9:13:17	Nor	
	The menu item 'Select All' was clicked.	9:13:17	Nor	
	The text 'Mark Twain' was entered in the text editor.	9:13:18	Nor	
	The button was dicked with the left mouse button.	9:13:18	Nor	
	The property checkpoint passed (the wItem(5, 0) equals "Mark Twa	9:13:19	Nor	
	The button was clicked with the left mouse button.	9:13:21	Nor	
Information	Picture Additional Info Call Stack Performance Counters			
Warnings: 0	Expected Image Actual	Image		
itart Time: 9:13 AM 12/7/2013 ind Time: 9:13 AM 12/7/2013 iun Time: 0:00:09 ile Name: C:\Users\JohnSmith\C				<b>``</b>

우리가 이전에 했던 샘플 테스트에 대한 결과 로그는 다음과 같습니다.

테스트 로그 화면은 하나의 테스트 실행 결과에 대해서 보여줍니다. 화면에서 왼쪽 부분은 테스트 결과를 트리 구조로 보여 지는데 각 노드들을 선택하면 해당되는 결과를 볼 수 있습니다. 예를 들면 본 문서에서는 하나의 테스트만 실행을 했기 때문에 트리 구조에는 오직 하나의 테스트 결과만 나 타납니다. 그 결과를 통해 테스트가 성공 했는지 아니면 실패 했는지 알 수 있습니다.

18 . 1. 1

테스트 로그는 결과에 따라 'Error', 'Warning' 등의 항목들로 필터링 해서 확인할 수 있습니다. 메 시지 타입은 왼쪽 아이콘을 보면 알 수 있습니다. 화면 상단에 있는 체크 박스를 통해서 사용자는 원하는 메시지들을 필터링 할 수 있습니다. 각각의 메시지들은 실행 시간 순서에 따라 나타납니다. 사용자는 그것을 타임 칼럼을 통해 확인할 수 있습니다.

TestComplete은 테스트 메시지를 통해서 이미지나 관련된 텍스트를 보여주기도 합니다. 테스트 메 시지의 상세 정보는 해당 메시지를 클릭하기만 하고 화면 하단의 'Additional Information'과 'Picture' 탭을 선택 하면 됩니다. 예를 들어 위의 'Picture' 탭에서 나타나는 스크린 샷 이미지를 통해 서 'Orders' 어플리케이션을 테스트 레코딩 할 때 'Orders | Edit order' 메뉴를 클릭 했던 것임을 알 수 있습니다.

'Picture' 패널은 'Expected Image'와 'Actual Image'로 나뉘어 져서 나타납니다. 'Expected Image' 는 레코딩 시에 저장된 이미지이고, 'Actual Image'는 테스트가 실행되면서 저장된 이미지입니다. 테 스트 로그에는 위와 같이 두 개의 이미지를 비교해서 다른 점을 찾아낼 수 있지만, 'Picture' 패널에 서 제공되는 'View Comparison Result' 버튼을 통해서도 쉽게 비교할 수 있습니다. 좀 더 자세한 사 항은 TestComplete의 Help 메뉴의 'Test Visualizer' 섹션을 참조하세요.

'Call Stack' 탭에서는 테스트가 실행된 로그에 대해서 계층별로 보여줍니다.

'Performance Counters' 탭에서는 테스트 실행 중에 모니터링 된 성능 카운터에 대한 값이 그래프



형태로 표시됩니다.

테스트 스크립트의 각 오퍼레이션들을 자세히 보기 위해서는 해당 오퍼레이션을 더블클릭 하면 됩니다. 그럼 TestComplete은 선택된 오퍼레이션과 관련된 창이 발생합니다. 예를 들면 만약 로그의 메뉴 항목의 'Orders | Edit order' 메시지를 클릭하면 관련된 정보가 활성화 됩니다.

16	Record New Test   Record New Test	to Test 🕨 Run Tes	t 💞 Add Variable 💸 Add Parame	ter 🛉 🛉 🎍 ≑ 🔿 🎙
	Item	Operation	Value	Description
	Run TestedApp	Orders		Runs the "Orders" teste
	🗇 🗖 Orders			
	MainForm			
	MainMenu	Click	"File Open"	Moves the mouse cursor
	- 🔁 dlgOpen	OpenFile	"C: \Users\Public\Documents\TestC	Opens a file specified by
	MainForm			
	🕆 🕅 OrdersView	ClickItem	"Samuel Clemens",	Clicks the 0 subitem of th
	- 👻 MainMenu	Click	"Orders  Edit order"	Moves the mouse cursor
	I Int OrderForm			
	- 🖏 Customer	Click		Clicks at point (102, 9) o
	- 🖏 Customer	ClickR		Clicks at point (102, 9) o
	PopupMenu	Click	"Select All"	Moves the mouse cursor
	- 🕞 Customer	SetText	"Mark Twain"	Enters the text 'Mark Tw
	ButtonOK	ClickButton		Clicks the 'ButtonOK' but
	- Property Checkpoint		Aliases Orders MainForm OrdersVi	Checks whether the 'wIt

테스트 로그 패널에 대한 자세한 사항은 TestComplete의 Help 메뉴의 'Test Results' 섹션을 참조하세요.

Note: 본 문서에서 설명하고 있는 로그들을 대표적으로 사용되는 키워드 테스트 스크립트입니 다. 다른 타입의 테스트는 로그 형태나 구조가 조금 다를 수 있습니다. 다른 형태의 로 그들에 대해서 알고 싶다면 다른 항목의 문서를 참조하거나 가이드를 보고자 하는 패널 에서 'F1' 버튼을 이용하시기 바랍니다.

## 에러의 해결

어쩌면 테스트를 실행 했을 때 문제가 발생할 수도 있습니다. 이번 섹션에서는 문제가 발생할 수 있는 몇 가지 가능성에 대해서 말씀 드립니다. 예를 들어, 개발자가 테스트 어플리케이션을 업데이 트 해서 몇몇 오브젝트의 이름이 변경되었다면 오브젝트의 인식에 문제가 발생하는 경우가 있을 수 있고, 테스트 목적과 다른 어플리케이션이 테스트 대상이 되는 어플리케이션과 겹쳐져서 그로 인해 테스트 실행에 문제가 생기는 등의 경우가 발생할 수 있습니다.

대표적인 원인 중에 하나는 레코딩 할 때의 환경과 테스트를 실행할 때의 환경이 달라서 발생 합 니다. 이런 문제를 피하기 위해서는 테스트를 실행하는 초기 환경과 레코딩 할 때의 환경을 확실히 동일하게 만들어 주어야 합니다. 예를 들면, 만약 레코딩 시에 테스트 어플리케이션이 실행되고 있 다면 테스트를 실행할 때도 테스트 어플리케이션은 실행되고 있어야 합니다. 만약 레코딩 할 때 테 스트 대상이 되는 웹 페이지의 두 번째 탭이 선택되어 있다면, 테스트를 실행할 때 역시 해당 웹 페 이지의 두 번째 탭이 선택되어 있어야 합니다.

이와 관련된 좀 더 자세한 사항은 TestComplete의 Help 메뉴의 'Handling Playback Errors' 섹션을 참조하세요.

# 웹(Web) 어플리케이션 테스트

이번 섹션에서는 TestComplete에서 프로젝트를 생성하고, 웹 페이지를 대상으로 테스트 시나리오 를 레코딩 한 뒤 테스트 실행하고 결과를 분석하는 것까지 단계별로 설명해 드리겠습니다. 테스트는 특정 웹 페이지를 통해서 테스트 시나리오를 레코딩하고 몇 개의 데이터를 검증하는 방식으로 진행 해 보겠습니다.

# 'Web Orders' 웹 페이지

본 문서에서 사용될 샘플 웹 페이지는 'Web Orders'입니다. 'Web Orders' 사이트에 대한 정보는 '<u>http://support.smartbear.com/samples/testcomplete10/weborders/</u>'를 통해서 확인하실 수 있습니다. 이 웹 페이지는 특정 항목을 추가하거나 수정 혹은 삭제한 뒤 주문을 할 수 있는 형태로 구성되어 있습니다.

	_										Welcome, Te	esteri
Web Orders												
	Lis	t of All (	Orders									
<ul> <li>View all orders</li> </ul>	a	ieck All	Uncheck All									
> View all products		Name	Product	#	Date	Street	City	State	Zip	Card	Card Number	Ex
> Order		Paul Brown	ScreenSaver	2	03/12/2010	5, Ringer Streat	Las Vegas, NV	US	748	MasterCard	123456789012	02/
	Ð	Mark Smith	FamilyAlbum	1	03/07/2010	9, Maple Valley	Whitestone, British	Canada	76743	VISA	770077007700	01/
	E	Steve Johns	ScreenSaver	1	02/26/2010	17, Park Avenue	Salmon Island	Canada	21233	MasterCard	444555444555	03/
	5	Charles Dodgeson	MyHoney	1	02/15/2010	45, Stone st.	Bringtone, TX	us	23233	American Express	333222333222	07/
	Ð	Susan McLaren	МуМолеу	1	01/05/2010	7, Flower Street	Earlcastle	Great Britain	21444	MasterCard	000858777658	04/
	8	Bob Feather	FamilyAlbum	1	12/31/2009	14, North av.	Militown, Wi	US	81734	VISA	111222111222	ocy
	Ð	Samuel Clemens	MyHoney	2	12/21/2009	3, Garden st.	Hillsberry, UT	us	53665	MasterCard	398743242342	03/
	-	Clare	Exercited		12/04/2008	23, Curk	Greentown,	105	63338	Martin Card		

# 1. 테스트 프로젝트 생성

이제 새로운 프로젝트를 생성해 보겠습니다.

1) 만약 TestComplete에 새로운 프로젝트나 프로젝트 스윗(suit)이 있다면 그것들은 닫으시기 바 랍니다. 메인 메뉴에서 'File | Close'를 선택하시면 됩니다.



2) 다시 메인 메뉴에서 'File | New | New Project'를 선택합니다. 그러면 'Create New Project' 마법사 창이 발생할 것입니다.

Create Nev	v Project
Specify the	name, location and other attributes of a new project.
Project name:	WebOrders
Location:	C:\Users\johnsmith\Documents\TestComplete 10 Projects
	More
Q Learn mor	re about these settings
	Next Finish Cancel

3) 첫 번째 페이지에서 프로젝트 명과 저장 위치를 지정할 수 있습니다. 'Project name' 입력 필 드에는 'WebOrders'를 입력합니다. TestComplete은 저장 위치의 경우는 기본으로 제공되는 경 로가 있습니다. 프로젝트 폴더는 프로젝트와 관련된 모든 정보(keyword tests, scripts, test logs, stores 등)가 저장됩니다. 사용자는 저장되는 위치를 필요에 따라 원하는 위치로 변경할 수 있습니다. 본 문서에서는 그대로 사용 하겠습니다.

또한 'More' 버튼을 통해서 프로젝트 스윗(suit)의 이름과 저장 위치를 변경할 수 있습니다. 본 문서에서는 변경하지 않고 그대로 진행 하겠습니다.

4) 프로젝트 이름과 저장 위치를 확인한 후에 'Next' 버튼을 클릭합니다.

다음 섹션에서도 위의 내용과 이어지면서 다음 설정에 대한 설명을 진행하겠습니다.



## 2. 테스트 모듈 선택

TestComplete에서는 추가되는 프로젝트들의 테스트 대상이 각각 다를 수 있습니다. 그런 이유로 프로젝트 생성 시 테스트 모듈을 설정해 주어야 합니다.

프로젝트에서 모듈을 설정하는 방법은 다음과 같습니다.

프로젝트를 생성할 때 'Create New Project' 마법사 창에서 설정 합니다. 이 경우에 TestComplete
 은 프로젝트 생성이 끝나면 사용자가 선택한 모듈에 맞게 프로젝트의 구조가 보여 질 것입니다.
 다른 방법으로 'Project Explorer' 패널에서 사용자가 언제든지 원하는 모듈로 설정할 수 있습니다.

본 문서에서는 프로젝트를 생성할 때 모듈을 선택하는 것으로 진행하겠습니다.

프로젝트 생성 마법사에서 프로젝트의 이름과 저장 위치를 설정한 후에 다음 페이지에서 테
 스트 대상 어플리케이션의 타입에 대한 모듈을 선택할 수 있습니다.

Specif	fy the type of the application under test.
M	Generic Windows Application Native, .NET, WPF, Java or other application that can run as a standalone executable.
6	Functional testing of web pages Simulate mouse actions and keystrokes on web pages, Flash, Flex or Silverlight applications.
	Android applications Applications (activities) running on physical and virtual Android devices.
	iOS applications Applications running on physical iOS devices.
Q	Web service testing Call web services' methods and check their responses.
4	Adobe AIR Out-of-browser Adobe AIR applications.
2 11-	

앞 서 'Web Orders' 웹 페이지를 테스트 한다고 말씀 드렸습니다. 그러므로 위 마법사 창 에서 웹 페이지를 테스트 할 수 있는 모듈을 선택하겠습니다. 위 보시는 메뉴 중에 'Functional Testing of web pages'를 클릭합니다. 그러면 자동으로 다음 페이지로 이동될 것 입니다.

다음 섹션에서 프로젝트 생성을 위한 나머지 설정을 끝마치도록 하겠습니다.



### 3. 테스트 프로젝트 설정

이전 단계에서 우리는 테스트 모듈을 설정 했습니다. 나머지 마법사 설정을 마치고 프로젝트를 생성 해 보겠습니다.

1) 테스트 대상에 대한 타입을 설정한 뒤, 다음 마법사 창에서는 TestComplete에서 제공되는 이 미지 캡쳐 기능을 설정할 것인지 아닌지를 물어보는 단계입니다.



이 기능은 테스트 레코딩 및 실행하는 동안 진행되는 과정들에 대한 이미지를 캡쳐 합니다. 이와 관련된 선택사항에는 오직 이미지만 캡쳐할 것인지 아니면 이미지와 함께 오브젝트에 대 한 정보도 함께 캡쳐 할 것인지 아무것도 캡쳐하지 않을 것인지 선택이 가능합니다.

테스트 레코딩 중에 캡쳐된 이미지들은 사용자가 어떤 시나리오에 따라 레코딩 됐는지, 또 테스트 대상이 어떻게 동작하는지 등을 이해하는데 도와줍니다. 테스트 실행 중에 캡쳐된 이 미지들은 테스트 어플리케이션이나 시스템이 테스트 실행 시 어떻게 구동이 됐는지를 이해하 기 쉽게 도와줍니다. 이런 이미지들은 나중에 디버깅 할 경우에도 참조자료로 유용합니다.

그런 장점들이 있는 반면에 이미지와 오브젝트 정보들을 모두 저장하면 나중에 프로젝트를 생성하거나 테스트 수행에 대한 결과 파일을 공유할 때 데이터 크기가 너무 커서 문제가 생길 수 있습니다. 그래서 프로젝트 성격 상 테스트 레코딩 시에나 실행 시 이미지 캡쳐가 필요하 거나 오브젝트 정보가 필요한 경우만 선별적으로 위 메뉴를 설정하시기 바랍니다. 이 설정은



나중에 프로젝트 설정에서 변경할 수 있습니다.

본 문서에서는 레코딩 및 실행 시 이미지만 캡쳐하는 'Capture images only'를 선택합니다. 그리고 'Next'를 클릭합니다.

2) 다음 마법사 창에서는 스크립트 언어(Language)를 선택할 수 있습니다.

0	Create New Project
S	pecify the scripting language of your project.
	VB VBScript Visual Basic-like scripting syntax.
	J JScript Java-like scripting syntax (similar to the C++ and C# syntax).
	DelphiScript Pascal-like scripting syntax.
	ConnectedApps Script Select this option if you are going to create Connected or Self-Testing Applications.
K	Help me choose
	Next Einish Cancel

TestComplete의 모든 프로젝트들은 하나의 지원되는 언어(VBScript, JScript, DelphiScript, C++Script or C#Script)를 선택해야 합니다. 스크립트 언어는 프로젝트에서 사용을 안 한다고 해도 중요한 설정사항 입니다. 사용자가 만약 키워드 테스트 방식만 사용한다고 해도 프로젝트에서는 코드에서 호출하는 경우가 발생하거나 특정 오퍼레이션 파라메타에서 스크립트를 사용할 수도 있습니다.

또한 스크립트 언어의 설정은 오브젝트 이름의 포맷을 정의하기 위해서도 중요한 사항입니 다. 오브젝트의 이름에 대한 포맷은 설정된 언어의 문법과 연관이 됩니다. 예를 들면, VBScript 와 JScript에서 IE의 이름은 'Browser("iexplorer")이고, DelphiScript에서는 'Browser('iexplore'), 그리고 C++Script와 C#Script에서는 "[Browser"]("iexplore")로 사용됩니다.

관련해서 더 자세한 내용은 TestComplete에서 제공되는 'Help' 메뉴의 'Selecting the Scripting Language' 항목을 참조하시기 바랍니다.

본 문서에서는 VBScript를 선택해서 사용 하겠습니다. 그러므로 'VBScript'를 선택하시기 바랍 니다. Window Vista 이 후 운영체제인 경우에는 스크립트 언어를 선택하면 마법사 창은 자동 으로 종료 됩니다. 만약 Window XP 이전 버전의 경우에는 스크립트 언어를 선택한 뒤 'Finish' 버튼을 클릭하세요.

여기까지 진행되면 TestComplete에는 새로운 프로젝트(WebOrders.mds)가 생성될 것입니다. 그리고 프로젝트 스윗(suit) 또한 생성되어 있을 것입니다. 생성 결과는 'Project Explorer' 패널 에서 확인하실 수 있습니다.

# 4. 웹 브라우저 설정

웹 테스트 위해서는 TestComplete에 웹 테스트를 레코딩하고 실행할 수 있는 설정을 적용해 줘야 합니다. 설정이 적용되면 여러 타입의 브라우저를 동일한 스크립트로 실행할 수 있게 됩니다. 이와 관련된 자세한 내용은 TestComplete에서 제공되는 'Help' 메뉴의 'Preparing Web Browsers' 항목을 참조하시기 바랍니다.

# 5. 테스트케이스 생성

### 웹 페이지 테스트 계획하기

샘플 Orders 웹페이지는 주문 리스트 기능이 있습니다. 사용자는 다음과 같이 주문을 하거나 주문 된 내용을 수정하는 기능을 사용할 수 있습니다.

- 테스트 목적: 기존의 주문 내역을 수정한 뒤 정상적으로 수정되었는지 혹은 변경되었는지 주 문 리스트에서 확인합니다.
- 테스트 단계: 본 문서에서 테스트 하는 과정은 주문 내역에서 수정하거나 리스트의 데이터를 검증하는 것입니다. 그래서 웹 페이지를 통해 사용자 시나리오를 만들고 레코딩 할 것입니다. 그리고 주문 리스트 중 하나의 속성을 변경해 보겠습니다.
- 테스트 결과의 확인: 만약 변경된 내역이 정상적으로 확인 됐다면 주문 리스트에 반영되어 나 타날 것입니다. 이것을 체크하기 위해 테스트케이스에서는 기대값을 데이터의 리스트와 비교 할 것입니다. 그리고 비교하기 위한 오퍼레이션(Checkpoint)을 삽입할 것입니다. 이 오퍼레이션 은 결과 로그에서도 확인이 가능합니다. 그래서 사용자는 결과 로그를 통해 Pass/Fail 여부를 확인할 수 있습니다.

관련해서 더 자세한 내용은 TestComplete에서 제공되는 'Help' 메뉴의 'Planning Tests' 항목 을 참조하시기 바랍니다.



### 테스트케이스 생성

TestComplete에서는 두 가지 방법으로 테스트케이스를 생성할 수 있게 지원합니다.

- 직접 테스트케이스를 생성
- 테스트 레코딩 방식을 이용해서 생성

직접 테스트케이스를 생성하는 경우 사용자는 필요한 오퍼레이션이나 테스트 시나리오를 스크립 트나 키워드 테스트에 맞춰서 생성해야 합니다. 이 방법은 매우 강력하면서도 테스트케이스를 유동 적으로 사용할 수 있다는 장점이 있습니다. 하지만 그 만큼 테스트케이스를 생성하는데 많은 경험이 필요합니다.

직접 테스트케이스를 생성하는 방법의 단점은 많은 시간을 필요로 하고 많은 다른 문제들이 발생 할 수 있습니다. 예를 들면 직접 테스트케이스를 생성하기 위해서는 테스트 어플리케이션의 클래스 네임(name)이나 오브젝트를 알아야 합니다. 이런 불편함을 해소하기 위해 TestComplete에서 테스트 레코딩만으로 편리하게 테스트케이스를 생성할 수 있는 기능을 제공합니다. 테스트시나리오를 한번 만 레코딩 하면 TestComplete에서 테스트 시나리오 뿐 만 아니라 스크립트 라인이나 테스트 오브젝 트들도 자동으로서 인지해서 키워드 테스트 스크립트로 자동 변환되어 생성됩니다. 이렇게 진행되는 형태를 '테스트 레코딩(recording a test)'이라고 합니다. 이런 방식은 사용자의 역량이 많지 않아도 편리하게 사용할 수 있기 때문에 매우 유용한 방식입니다. 그래서 본 문서에서는 테스트 레코딩 방 식으로 가이드를 진행할 것입니다. 다음 섹션에서 좀 더 자세하게 설명 드리겠습니다.

#### 테스트 레코딩

테스트 레코딩을 위해서는 아래와 같이 세 가지 단계가 필요합니다.

① 사용자는 TestComplete의 메뉴에서 제공하는 Record Keyword Test 혹은 Record Script를 통해 해서 레코딩을 시작할 수 있습니다. 또한 시작 페이지에서 'Record a New Test' 버튼을 통해 서도 할 수 있습니다.

TestComplete에서는 여러 가지 방법(keyword test, scripts, low-level procedures, HTTP load testing tasks)으로 테스트 레코딩을 할 수 있습니다. 메인 메뉴에서 제공하는 레코딩 방식은 keyword test와 script입니다. 다른 테스트는 레코딩이 시작된 후에 적용할 수 있습니다. 메인으로 사용되는 두 가지 테스트 레코딩 방식은 레코딩이 시작된 후에 몇 개의 조작 기능이 제 공됩니다.

레코딩이 시작되면 아래와 같이 레코딩 툴 바가 나타납니다.



레코딩 툴 바는 레코딩 중 일시정지와 종료 기능 뿐 만 아니라 레코딩 중에 필요한 기능들 을 제공합니다. 그리고 레코딩 되는 타입(Keyword test, script code, low-level procedure)을 변 경할 수 있습니다.



- ② 레코딩을 시작한 후에 매뉴얼로 테스트 시나리오를 수행합니다. 예를 들어 테스트 어플리케 이션을 실행하고, 시나리오에 따른 버튼을 클릭하거나 메뉴를 선택하고 텍스트를 입력하는 등 의 행위를 합니다.
- ③ 테스트 시나리오 수행을 완료한 후에 툴 바에 있는 종료 버튼()을 클릭해서 레코딩을 종료 합니다.

레코딩과 관련된 자세한 사항은 TestComplete의 Help 메뉴에서 'Recording in TestComplete' 항목을 참조하시기 바랍니다.

### 'Orders' 웹 페이지의 레코딩

키워드 테스트 방식으로 'Orders' 웹 페이지를 레코딩 해 보겠습니다. 먼저 브라우저를 실행하고 특정 페이지가 로딩 되면 테스트 시나리오에 따라 버튼을 클릭하거나 키를 입력해서 데이터를 검증 할 것입니다.

Note: 테스트 레코딩 중에 TestComplete의 Help 메뉴를 보는 등의 행위를 하지 마세요. 레코 딩 엔진은 사용자의 모든 행위를 추적하기 때문에 테스트 시나리오 외에 다른 행위가 레코딩 될 수 있습니다.

### 레코딩 시작

 1) 새로운 프로젝트를 생성하면 TestComplete에서는 Keyword 테스트 방식의 테스트케이스가 자 동으로 생성됩니다. 이 테스트케이스에서 레코딩하고자 하면 툴 바에서 'Append to Test' 버튼 (♥)을 클릭하면 됩니다.





레코딩을 시작하면 레코딩 툴 바가 나타날 것입니다. 만약 'Interactive Help' 패널이 보인다 면 해당 메뉴를 통해 레코딩과 관련된 추가적인 메뉴들을 확인할 수 있습니다.

기본으로 제공되는 메뉴는 아래와 같습니다.



버튼())을 클릭해서 모든 기능의 버튼이 보이도록 할 수 있습니다.

Recordi	ing					٤
			2. *	• 🗗	* ሴ *	
Rec.	Stop	Pause	1	2 *		SHE .

- 2) 테스트 웹 사이트의 주소는 <u>Http://support.smartbear.com/samples/testcomplete10/weborders/</u> 입니 다. TestComplete은 사용자에 의한 테스트 시나리오를 레코딩 하게 되고, 레코딩이 완료되면 브라우저 시작부터 레코딩 된 것을 스크립트를 통해 확인할 수 있습니다.
- 3) 인터넷 브라우저를 시작한 뒤 아래 페이지가 로딩될 때까지 기다리세요.

					08
Inttp://supports	smartbean.com/Samples/TestCompletel	n/WebOrdesn P = E d	🔆 🗶 🔷 Web Orders Login	×	
le Edit Yiew Fgrorites	Inch Help				
• 回 • □ · □ · • •	age * Satety * Tools * 🐏 *				
		Osemanie			
		Paranoso.			
		LOG	n		
	In order	to log in Orders sample -	se the following information	0	
		Passi	me - Texter vord - best		
					\$ 100%

만약 본 문서의 화면이 테스트 해야 할 웹 페이지와 겹쳐진다면 본 문서나 테스트 웹 페이 지의 크기를 교정하시기 바랍니다. 레코딩 시 본 문서화면이 방해가 되지 않도록 미리 준비가 되어야 합니다.





- 4) 로그인에 대한 정보는 아래와 같습니다.
  - Username: Tester
  - Password: test

로그인 정보를 입력 후에 'Login' 버튼을 클릭하세요.

5) 로그인 하면 기본 페이지는 보여 집니다. 기본 페이지에는 모든 주문과 관련된 기능들이 포함 되어 있습니다. 이제 하나의 주문을 수정 하겠습니다. 리스트에서 'Samuel Clemens'의 수정 버 튼(☑)을 클릭하세요. 그러면 아래와 같은 주문 수정 페이지가 나타날 것입니다.

fdit Yew fyrontes Look Bei • 🔯 + 🖾 🎰 + Boge + Sefety	* Tgols * 😥 *		
Web Orders		Welcome, T	estert (
	Edit Order		
> View all orders	Product Inform	ation	
> View all products	Product:**	MyMoney 💌	
> Order	Price per unit: Discount: Totali	100 0 % 200 Carcult	
	Address Inform	ation	
	Street: *	3. Garden st.	
	Eky:*	Hillsberry, UT	
	Zip:"	53666	
	Payment Inform	nation	
	Card:*	C visa # HasterCard C American Express	
	Card Nr:	396743242342	
	Expres data (mmyyy):"	Update	
			₹ 100%

- 6) 위 페이지에서 'Customer name' 텍스트 박스를 클릭한 뒤 'Mark Twain'으로 수정하세요.
- 7) 'Update' 버튼을 클릭하세요. 페이지는 다시 메인 화면으로 돌아오고, 수정된 사항은 반영되서 보여 질 것입니다.
- 8) 이제 값을 검증하는 오퍼레이션(Checkpoint)을 삽입할 것입니다. 이 오퍼레이션은 리스트화면 에서 'Mark Twain'으로 변경되었는지 검증합니다.

우리는 이렇게 검증하는 기능을 '체크포인트'(Checkpoint)라고 합니다. TestComplete은 여러 가지 타입의 체크포인트를 제공합니다. 그 중에 가장 많이 쓰이는 타입은 'Property checkpoint'입니다. 이 타입은 어플리케이션을 제어하는 데이터들을 검증하는데 사용됩니다. 우 리는 이 체크포인트를 사용해서 진행할 것입니다.



아래와 같이 체크포인트 드롭버튼을 클릭해서 'Create Property Checkpoint( \$\vdots)' 버튼을 클릭 하세요.



위 과정을 통해서 속성을 검증하는 체크포인트(Property Checkpoint)가 삽입될 것입니다. 그 리고 체크포인트 설정을 진행하기 위한 마법사 창이 발생합니다.

Create Pro	perty Checkpoint			
Choose an	object whose pro	operty will be compar	red,	
Press the b	the target to point to the left mouse button ov itton to select this windo	• the object /er the target icon and drag th w or control. Show me how it	e icon to the desired windoworks.	ow or control. Releas
Point Press scree	and fix (Shift+Ctrl+A this button and then mov h. Press Shift+Ctrl+A to	<ol> <li>ve the mouse pointer to the deselect the object. <u>Show me ha</u></li> </ol>	esired window, control or a w it works.	ny other object on
Object:				Highlight
ł				
Preview:	e about these settings			



첫 번째 마법사 페이지에서 타겟 버튼 🛞을 마우스로 클릭한 채로 드래그 합니다.

마법사 창이 자동으로 최소화 되고난 뒤 'Mark Twain' 리스트로 마우스를 드래그 합니다. 마 우스를 드래그 하는 동안 화면에 존재하는 여러 오브젝트들이 인식되어 붉은 박스가 발생할 수도 있습니다.

검증하고자 하는 주문 리스트에서 'Mark Twain'으로 마우스 포인트를 이동시키면 아래와 같 이 붉은 박스가 나타날 것입니다. 그 후에 마우스를 떼면 됩니다.

đ	eck All	Uncheck All										
	Name	Product	#	Date	Street	City	State	Zip	Card	Card Number	Exp	
۳	Paul Brown	ScreenSaver	2	03/12/2010	S, Ringer Street	Las Vegas, NV	US	748	MasterCard	123456789012	02/07	
2	Mark Smith	FamilyAlbum	1	03/07/2010	9, Maple Valley	Whitestone, British	Canada	76743	VISA	770077007700	01/09	đ
	Steve Johns	ScreenSaver	1	02/26/2010	17, Park Avenue	Salmon Island	Canada	21233	MasterCard	444555444555	03/09	
٥	Charles Dodgeson	MyMoney	1	02/15/2010	45, Stone st.	Bringtone, TX	US	23233	American Express	333222333222	07/10	ø
e	Susan McLaren	MyMoney	1	01/05/2010	7, Flower Street	Earlcastle	Great Britain	21444	MasterCard	999888777888	04/10	Z
Ð	Bob Feather	FamilyAlbum	1	12/31/2009	14, North av.	Milltown, WI	US	81734	VISA	111222111222	06/08	ø
F	Mark 🏵 Twain	MyMoney	2	12/21/2009	3, Garden st.	Hillsberry, UT	US	53665	MasterCard	398743242342	03/09	
E	Clare Jefferson	FamilyAlbum	2	12/04/2009	23, Owk Street	Greentown, CA	US	63325	MasterCard	770000770000	03/08	



- 드래그 하던 마우스 버튼을 검증하고자 하는 오브젝트 위에서 떼고 나면 마법사 창은 다시
 나타나고 선택된 오브젝트의 이미지가 아래와 같이 나타납니다.

Choose an object whose property will be compared.
Orag the target to point to the object         Press the left mouse button over the target icon and drag the icon to the desired window or control. Release the button to select this window or control. Show me how it works.         Point and fix (Shift+Ctrl+A)         Press this button and then move the mouse pointer to the desired window, control or any other object on screen. Press Shift+Ctrl+A to select the object. Show me how it works.         Object:         Viases.browser.pageWebOrders2.formAspnetform.tableCtl00MaincontentOrdergrid.cellMarkTwain
Preview: Mark Twain
Learn more about these settings







다음 마법사 페이지에서는 오브젝트의 속성들이 표시됩니다. 이 리스트들은 TestComplete에 서 규정하는 오브젝트들로 구성되어 보여 집니다. 'View more members(Advanced View)' 링크 를 클릭하면 모든 사용 가능한 속성들을 볼 수 있습니다.

TestComplete은 선택된 오브젝트의 속성에 대해서 세 가지 그룹으로 구성되어 나타납니다. 첫 번째 그룹은 테스트 환경에서 공통으로 사용되는 속성들입니다. 그것은 'Standard'에서 확 인할 수 있습니다. 두 번째 그룹은 그리드(grid) -*그리드 오브젝트를 선택한 경우*- 를 제어하는 속 성 그룹이고, 'Extended'에서 확인할 수 있습니다. 세 번째 그룹은 인터넷 브라우저에서 제공되 는 속성들입니다. 사용자는 사용된 브라우저 이름의 하위 노드에서 그 속성들을 확인할 수 있 습니다. 데이터를 검증하기 위해서는 브라우저의 'contentText' 속성을 사용할 것입니다. 그것 은 브라우저에서 나타나는 텍스트를 보여주는 속성을 가지고 있고, 그 텍스트에 대한 문제가 있는지를 확인할 수 있습니다.

2. formAspnetform.tableCtl00N		
Properties	laincontentOrdergrid.cellMarkTwain.contentText Search	٩
Advanced view View basic me	nbers (Basic view)	
Height	61	
Id	154	-
Left	310	
Name	Cell(7, 1)	
NativeWebObject	[object HTMLTableDataCellElement]	
Parent	(Object)	
ScreenLeft	1635	
ScreenTop	746	
Тор	571	
Visible	True	
VisibleOnScreen	True	
Width	70	
Extended		
ColumnIndex	1	
contentText	Mark Twain	
idStr		
namePropStr		

리스트에서 'contentText' 속성을 찾으세요. 그리고 그 속성을 선택한 뒤 'Next' 버튼을 클릭 하세요.



다음 마법사 페이지에서는 속성의 이름을 확인할 수 있습니다. 아래 나타나는 'Value' 박스를 통해서 값을 비교, 검증할 수 있습니다.

Specify comparison settin	ngs.
Object:	
Aliases.browser.pageWebOrders2.f	formAspnetform.tableCtl00MaincontentOrdergrid.cellMarkTwain
Property:	
contentText	
Condition:	
Equals	
Value:	
Mark Twain	
Learn more about these setting	ngs

체크포인트 생성 완료를 위해 'Finish' 버튼을 클릭하세요. TestComplete은 레코딩 된 스크립 트에 체크포인트를 삽입할 것입니다.

9) 테스트 웹 페이지의 상단 오른쪽에 있는 'Logout' 버튼을 클릭하세요.

10) 'X' 버튼을 클릭해서 테스트 웹 페이지를 닫으세요.

11) 레코딩 종료 버튼(■)을 눌러서 레코딩을 종료하세요. TestComplete은 레코딩 된 스크립트를 테스트케이스로 저장할 것입니다.





# 6. 테스트 레코딩 결과 분석

레코딩을 마친 후에 TestComplete은 수정이 가능한 키워드 타입의 테스트 스크립트가 자동으로 아래와 같이 생성됩니다.

Operations 🔶	📒 👧 Record New Test   Kappend to Test 🕨 Run	Test 🛛 🤣 Add Va	riable 🔗 Add Parameter 👎	🔓 (* * 🕹 🕹
Search 🔎 💌	Item	Operation	Value	Description
Frequently Used	Run Browser	Internet Explorer	"http://support.smartbear.co	Launches the specified
Test Actions	browser	11 <sup>1</sup>		
Logging	pageWebOrdersLogin	Wait	(A.A.F.)	Waits until the browser.
Web	🖃 🛄 pageWebOrdersLogin			
읅 Run Browser	T textboxCtl00MaincontentUsername	SetText	"Tester"	Sets the text 'Tester' i
🜍 Navigate	passwordboxCtl00MaincontentPassw	SetText	"test"	Sets the text 'test' in t
If Browser		ClickButton		Clicks the 'submitbutto
	pageWebOrders	Wait	344C	Waits until the browser.
Browser Loop	🖨 🛄 pageWebOrders			
🗊 Run Virtual Browser	📑 👘 🙀 imagebutton	Click		Clicks at point (15, 7) o.
Virtual Browser Loop	💦 pageWebOrders2	Wait		Waits until the browser.
	😑 🛄 pageWebOrders2			
	textboxCtl00MaincontentFmworderT	SetText	"Mark Twain"	Sets the text 'Mark Tw.
	📑 👘 🖞 linkCtl00MaincontentFmworderUpda	Click		Clicks the 'linkCtl00Mai
	pageWebOrders	Wait		Waits until the browser.
	- Report Checkpoint		Aliases.browser.pageWebOr	Checks whether the 'c
	🖃 🌀 browser			
	pageWebOrders2			
	🔄 👘 🖞 linkCtl00Logout	Click		Clicks the 'linkCtl00Log
Checkpoints	- D pageWebOrdersLogin	Wait		Waits until the browser.
Statements	BrowserWindow	Close	••••	Closes the 'BrowserWi

레코딩 된 테스트케이스는 위의 이미지와 비슷할 것입니다. 만약 본 문서에서 언급된 시나리오와 조금 다르게 진행되었다면 스크립트 역시 조금 다를 수 있습니다. 예를 들면 웹 페이지의 다른 여백 을 그냥 클릭 했다고 하더라도 그 행위 역시 위의 스크립트에 저장되어 나타납니다.

레코딩 된 스크립트들은 레코딩 시작 후 사용자가 테스트 웹 페이지에서 수행된 모든 행위들에 대 해 보여줍니다. 우리는 그것들을 오퍼레이션(operations)이라고 합니다.

아래의 'Test Visualizer' 패널에는 레코딩 중에 캡쳐된 이미지들이 표시됩니다.

		Wait	
Checkpoints Statements Miscellaneous	imagebutton	Click Wait	
Test Steps * 🖾 Variables	S Parameters		
Update	Update	Updat	6 6
	=2-		
10.1			



위의 이미지들은 테스트 시나리오가 어떻게 수행되었는지 이해하는데 도움을 줍니다. TestComplete은 사용자가 진행한 테스트 수행(마우스 클릭, 텍스트 입력 등)과 동일하게 이미지가 캡쳐 됩니다.

사용자가 임의의 오퍼레이션을 선택한다면 그와 매칭되는 이미지(이미지가 있는 경우)가 자동으로 선택되어 표시되기 때문에 이미지가 어떤 스크립트와 연관이 되어 있는지 쉽게 확인할 수 있습니다. 이와 관련된 자세한 사항은 TestComplete의 Help 메뉴의 'Test Visualizer' 섹션을 참조하세요.

테스트 스크립트에서 첫 번째 오퍼레이션은 'Run Browser'입니다. 이것은 키워드 테스트에서 인터 넷 브라우저를 실행하고 어떤 페이지로 이동할 것인지에 대한 정보를 담고 있는 오퍼레이션입니다. TestComplete에서는 브라우저를 실행할 때 이를 자동으로 레코딩 합니다.



두 번째 오퍼레이션은 웹 브라우저가 로딩 되는 것을 기다리는 기능을 가지고 있습니다. 아래 나 타난 오퍼레이션은 테스트 웹 페이지(Orders)의 로그인 화면이 완전히 로딩될 때 까지 테스트 진행 을 기다리게 합니다.

Item		Operation	Value
	Run Browser	Internet Explorer	"http://support.smartbear.com/Sa
📄 🖨 🥹	browser		
	🕞 pageWebOrdersLogin	Wait	***
e	pageWebOrdersLogin		

아래의 오퍼레이션은 로그인 창에서 수행되는 작업에 대한 것입니다.

🖨 📀 browser			
	Wait		
🖃 🔲 pageWebOrdersLogin			
textboxCtl00MaincontentUsername	SetText	"Tester"	
passwordboxCtl00MaincontentPassw	SetText	"test"	
🖞 submitbuttonLogin	ClickButton		
pageWebOrders	Wait	1.1.1.X	

다음에 나타나는 오퍼레이션은 테스트 웹 페이지에서 주문 내역 페이지와 그 중에 'Customer Name' 필드를 수정하는 것과 관련된 것입니다.

	Wait	
🖯 🗖 pageWebOrders		
timagebutton	Click	
	Wait	
🖻 🛅 pageWebOrders2		
- 🕞 textboxCtl00MaincontentFmworderT	SetText	"Mark Twain"
🛄 🖏 linkCtl00MaincontentFmworderUpda	Click	
pageWebOrders	Wait	

이와 관련된 마우스 이벤트 및 키보드 입력 등의 기타 행위들에 대한 스크립트 정보는 TestComplete의 Help 메뉴의 'Simulating User Actions' 섹션을 참조하세요.



아래의 오퍼레이션은 레코딩하면서 추가했던 값을 검증하는 체크포인트입니다.

		Wait	
	Roperty Checkpoint		Aliases.browser.pageWebOrders2
E	🗄 🥥 browser		
	🖃 🔲 pageWebOrders2		
	hinkCtl00Logout	Click	
	🔂 pageWebOrdersLogin	Wait	
	BrowserWindow	Close	

마지막으로 웹 페이지에서 로그아웃 한 뒤 종료하는 오퍼레이션입니다.

	bageWebOrders	Wait	
	Roperty Checkpoint		Aliases.browser.pageWebOrders2
E	💿 browser		
	🖃 📃 pageWebOrders2		
	🐪 抗 linkCtl00Logout	Click	
	🗠 🕞 pageWebOrdersLogin	Wait	
	BrowserWindow	Close	

위에서 보셨듯이, TestComplete은 사용자가 수행된 것과 동일하게 오퍼레이션을 생성하고 그룹화 해서 보여 줍니다. 오퍼레이션들의 그룹화는 테스트 시나리오의 구조를 이해가기 쉽게 도와주고 웹 에 대한 계층 구조의 정보도 제공해 줍니다.

본 문서에서는 하나의 브라우저에서 테스트 시나리오를 가지고 레코딩 했습니다. 그래서 브라우저 에 대한 하나의 그룹만 생성되었습니다. 그리고 브라우저에서 이루어진 모든 수행 내역에 대해서도 저장이 되었습니다. 만약 다른 브라우저에서도 테스트가 수행되었다면 'page'라는 그룹으로 달리 구 성되었을 것입니다.

	Item	Operation	Value
	Run Browser	Internet Explorer	"http://support.smartbear.com/Sa
	•🖯 🥥 browser		
		Wait	
	🗕 🗖 pageWebOrdersLogin		
	textboxCtl00MaincontentUsername	SetText	"Tester"
	passwordboxCtl00MaincontentPassw	SetText	"test"
Grouping node	S submitbuttonLogin	ClickButton	
	- B pageWebOrders	Wait	
	🛶 🔁 pageWebOrders		
	🔤 🖏 imagebutton	Click	
		Wait	
	textboxCtl00MaincontentFmworderT	SetText	"Mark Twain"
	🔓 🕅 linkCtl00MaincontentFmworderUpda	Click	***





간혹 사용자는 인터넷 브라우저의 이름을 'Object Browser' 패널에서 확인하기 위해 기억해야 할 수도 있습니다. 예를 들면 'Object Browser'에서 인터넷 브라우저의 이름은 'Browser("iexplore")', 'Browser(firefox")', 'Browser("chrome"), 'Browser("opera") 등으로 사용될 수 있지만, 테스트 중에는 단순하게 'browser'로만 불릴 수 있습니다. 또 다른 예로 'Object Browser'에서 로그인 페이지 (http://support.smartbear.com/Samples/TestComplete10/WebOrders/Login.aspx)는 테스트 하는 동안 'pageWebOrdersLogin'으로 사용됩니다.

그런 이유는 기본적으로 TestComplete은 테스트 레코딩 하는 동안 오브젝트의 커스텀네임 (customer name)을 수집하고 필요에 따라 사용합니다. 커스텀네임을 수집하고 분류하는 것을 맵핑 (mapping) 한다고 합니다. TestComplete은 기본으로 사용되는 이름이 사용자가 이해하기 어려울 수 있기 때문에 이름들을 맵핑 하는 작업을 합니다. 맵핑된 이름을 사용하는 것은 테스트를 좀 더 쉽게 이해하도록 도움을 줍니다. 이와 관련된 자세한 사항은 TestComplete의 Help 메뉴의 'Name Mapping' 섹션을 참조하세요.

## 7. 레코딩 된 테스트케이스 실행

이전까지 간단한 테스트케이스를 실행하기 위한 준비를 모두 마쳤습니다.

레코딩 된 테스트케이스를 실행하기 전에 유념해야 할 부분은, 테스트를 실행하기 위한 환경이 레 코딩 당시의 환경과 동일하게 구성해 주어야 한다는 것입니다. 예를 들면, 웹 테스트는 거의 대부분 이 인터넷 브라우저 페이지를 오픈 하면서 시작되는 것이 일반적입니다. 본 문서에서도 'Run Browser' 오퍼레이션을 통해서 브라우저를 열고 테스트 웹 페이지로 이동하면서 테스트가 시작됩니 다. 또한 테스트 종료 시점에 로그아웃을 한 뒤 웹 페이지를 종료하는 것도 기억하시기 바랍니다.

Item	Operation	Value
Run Browser	Internet Explorer	"http:/
🖨 🥥 browser		
pageWebOrdersLogin	Wait	
😑 🛄 pageWebOrdersLogin		
textboxCtl00MaincontentUsername	SetText	"Tester
passwordboxCtl00MaincontentPassw	SetText	"test"
📳 🕺 🖞 submitbuttonLogin	ClickButton	

테스트 실행을 위한 가장 간단한 방법은 툴 바에서 'Run Test' 버튼()을 클릭하는 것입니다.

실행 버튼을 클릭하면 TestComplete의 화면은 자동으로 최소화 되고, 테스트가 실행될 것입니다.

Note: 테스트가 실행되는 동안 마우스를 움직이거나 키를 입력하는 등의 행위를 해서는 안 됩 니다. 그런 행위로 인해 테스트 실행에 오류가 생길 수 있습니다. 테스트 실행이 완료된 후에 TestComplete은 테스트 결과를 수집한 뒤 보여줄 것입니다. 다음 섹션 에서 테스트 결과에 대한 분석을 설명 드리겠습니다.

테스트 실행과 관련해서 몇 가지 더 말씀 드리면,

테스트케이스가 생성 됐다고 해서 테스트 자체가 완료되었다는 것은 아닙니다. 사용자는 TestComplete을 통해서 테스트를 실행할 수 있습니다. PC에 TestComplete이 설치되어 있지 않다 면 테스트케이스 실행을 위해서는 'TestExecute'라는 프로그램이 필요합니다. 또한 만약 사용자가 스크립트 코드를 외부로 출력해서 사용하고 싶다면 그것은 가능합니다. 더 자세한 사항은 TestComplete의 Help 메뉴의 'Connected and Self-Testing Applications' 섹션을 참조하세요.

- 테스트 실행 중에 TestComplete은 화면 상단에 아래와 같은 인디케이터를 표시합니다.



위의 인디케이터는 테스트 수행이 어떻게 진행되고 있는지에 대한 정보를 보여줍니다.

- TestComplete은 테스트케이스가 모두 완료될 때까지 실행됩니다. 만약 사용자가 실행을 멈추고자 한다면 인디케이터에서 'Stop' 버튼(■)을 클릭하거나, TestComplete 메인 화면에서 'Test | Stop'를 선택하면 됩니다.

테스트 실행 중에 일시정지 하고 싶다면 정지 버튼())을 클릭하세요. 일시정지 되어 있는 동안 에는 사용자가 테스트 외에 다른 행위를 해도 상관없습니다. 예를 들면, 사용자는 일시정지 시키고 나서 'Watch List' 패널이나 'Locals' 패널 혹은 'Evaluate' 창에서 테스트 로그를 체크 하거나 사용 된 오브젝트를 확인하는 등의 행위를 할 수 있습니다. 관련해서 자세한 사항은 TestComplete의 Help 메뉴의 'Debugging Tests' 섹션을 참조하세요.

테스트 실행이나 테스트 실행과 관련된 설정에 대한 정보는 TestComplete의 Help 메뉴의 'Running Tests' 섹션을 참조하세요.

# 8. 테스트 결과 분석

TestComplete은 테스트 실행 시 모든 오퍼레이션에 대한 수행 결과를 보관하고 있습니다. 테스트 결과에 대한 리스트는 'Project Explorer' 패널의 'Project Suite Logs | WebOrders Log'에 표시됩니 다. 결과 노드의 개수는 테스트 실행 횟수와 일치합니다. 아래에 나타나는 이미지는 테스트가 한번 실행 됐고, 그 결과는 패스(pass) 했다는 것을 의미합니다.

Project Workspace Object Browser	
C Project Explorer ? 주 🔀	Workspace
12 🗈 🕘 🖻 🕩 🌮 💶	😧 Start Page 🗙 🛃 Test 1
Project Suite 'WebOrders' (1 project WebOrders Advanced KeywordTests Test1 NameMapping	Go to Next: S Error ▼ Log Items Keyword Test Log [Test1]
Stores	

TestComplete에서는 테스트를 실행하고 나면 마지막 부분에 그 결과를 자동으로 추가 합니다. 테 스트 결과는 테스트 실행 중에는 표시되지 않습니다. 만약 테스트 실행 중에 결과를 확인하고 싶다 면 테스트 중지 기능(pause)을 통해서 확인할 수 있습니다.

본 문서에서는 한 번의 테스트만 실행했기 때문에 'Project Explorer' 패널에는 하나의 결과만 확인 할 수 있습니다. 그리고 테스트가 종료되면 'Workspace' 패널에 그 결과가 자동으로 나타납니다. 직 접 결과 페이지를 열과 싶다면 결과 리스트에서 마우스 우클릭 후 'Open' 메뉴를 클릭하면 됩니다.



보	문서에서	진행되	테스티	격과	로ㄱ릌	보며	다음과	간습니다	
	신지에지	L' O T'		2-1			니머피	ㅌㅂㅋㅋ	

🖉 Workspace			? I 🛛 📈
🛃 Test1 🗙 🔏 Keyword Tes	tlog [Test1] 4/17/20 🗴		
Ge to Next: 😢 Error 🔻 🧃	🖥 🛪 🔺 🗇 🛸		
Log Items	TestLog		
🧾 Keyword Test Log [Te	🔕 🖉 Error 🔺 🕼 Warning 🦊 🖉 Message 🥥 🖉 Event - Se	earch	P -
	Type         Message           Image: State in the internet in the internet in the internet internet in the internet	Time         △         Priority           15:29:51         Normal           15:29:55         Normal           15:29:56         Normal           15:29:56         Normal           15:29:58         Normal           15:29:58         Normal           15:29:58         Normal           15:29:58         Normal           15:30:00         Normal           15:30:03         Normal           15:30:04         Normal	Has Link
	Picture Additional Info Call Stack Performance Counters		
	Expected Image     A       Image     Image       Image		×
Information Errors: 0 Warnings: 0 Start Time: 15:29 4/17/2014 End Time: 15:30 4/17/2014 Run Time: 0:00:13 File Name: C:\Users\johnsmith\		- 197 - 197 - 192 - 192	

테스트 로그 화면은 하나의 테스트 실행 결과에 대해서 보여줍니다. 화면에서 왼쪽 부분은 테스트 결과를 트리 구조로 보여 지는데 각 노드들을 선택하면 해당되는 결과를 볼 수 있습니다. 예를 들면 본 문서에서는 하나의 테스트만 실행을 했기 때문에 트리 구조에는 오직 하나의 테스트 결과(노드) 만 나타납니다. 그 노드를 통해 테스트 결과가 성공 했는지 아니면 실패 했는지 알 수 있습니다.

테스트 로그는 결과에 따라 'Error', 'Warning' 등의 항목들로 필터링 해서 확인할 수 있습니다. 메 시지 타입은 왼쪽 아이콘을 보면 알 수 있습니다. 화면 상단에 있는 체크 박스를 통해서 사용자는 원하는 메시지들을 필터링 할 수 있습니다. 각각의 메시지들은 실행 시간 순서에 따라 나타납니다. 사용자는 그것을 'Time' 칼럼을 통해 확인할 수 있습니다.

TestComplete은 테스트 메시지를 통해서 이미지나 관련된 텍스트를 보여주기도 합니다. 테스트 메 시지의 상세 정보는 해당 메시지를 클릭하고 화면 하단의 'Additional Information'과 'Picture' 탭을 확인하면 됩니다.

'Picture' 패널은 'Expected Image'와 'Actual Image'로 나뉘어 져서 나타납니다. 'Expected Image' 는 레코딩 시에 저장된 이미지이고, 'Actual Image'는 테스트가 실행되면서 저장된 이미지입니다. 테 스트 로그에는 위와 같이 두 개의 이미지를 비교해서 다른 점을 찾아낼 수 있지만, 'Picture' 패널에 서 제공되는 'View Comparison Result' 버튼을 통해서도 쉽게 비교할 수 있습니다. 이와 관련된 좀



더 자세한 사항은 TestComplete의 Help 메뉴의 'Test Visualizer' 섹션을 참조하세요.

오퍼레이션의 메시지에 대한 자세한 정보를 보기 위해서는 해당 로그를 마우스로 더블클릭 하면 됩니다. 그러면 TestComplete은 선택된 오퍼레이션에 대한 창을 보여줄 것입니다. 예를 들면, 로그 메시지 중에서 'The button was clicked with the left mouse button'이라는 메시지를 더블클릭 하 면, 이 때 수행됐던 키워드 테스트 오퍼레이션이 활성화 될 것입니다.

Item	Operation	Value	*
Run Browser	Internet Explorer	"http://support.smartbear.com/Sa	
pageWebOrdersLogin	Wait		
bugeweborderscogin     bugeweborderscogin     bugeweborderscogin     bugeweborderscogin	SetText	"Tester"	
passwordboxCtl00MaincontentPassw	SetText	"test"	
submitbuttonLogin	ClickButton		Е
pageWebOrders	Wait		
🛱 🛄 pageWebOrders			
- 🖒 imagebutton	Click	3444	
	Wait	***	
🖻 🔲 pageWebOrders2			
textboxCtl00MaincontentFmworderT	SetText	"Mark Twain"	
🔄 🖔 linkCtl00MaincontentFmworderUpda	Click		
b pageWebOrders	Wait		
Property Checkpoint		Aliases.browser.pageWebOrders2.	
📄 🥯 browser			-

이와 관련된 테스트 로그 패널이나 로그의 메시지 혹은 결과에 대한 자세한 정보는 TestComplete 의 Help 메뉴의 'About Test Log' 섹션을 참조하세요.

Note: 본 문서에서 설명하고 있는 로그들은 기능 테스트 자동화 도구에서 대표적으로 사용되 는 키워드 테스트 스크립트입니다. 다른 타입의 테스트는 로그 형태나 구조가 조금 다를 수 있습니다. 다른 형태의 로그들에 대해서 좀 더 자세히 알고 싶다면 다른 섹션을 참조 하시거나 궁금하신 패널을 선택한 후 'F1' 버튼을 이용하시기 바랍니다.

#### 에러의 해결

어쩌면 테스트를 실행 했을 때 문제가 발생할 수도 있습니다. 이번 섹션에서는 문제가 발생할 수 있는 몇 가지 가능성에 대해서 말씀 드립니다. 예를 들어, 개발자가 테스트 어플리케이션을 업데이 트 해서 몇몇 오브젝트의 이름이 변경되었다면 오브젝트의 인식에 문제가 발생하는 경우가 있을 수 있고, 테스트 목적과 다른 어플리케이션이 테스트 대상이 되는 어플리케이션과 겹쳐져서 그로 인해 테스트 실행에 문제가 생기는 등의 경우가 발생할 수 있습니다.

대표적인 원인 중에 하나는 레코딩 할 때의 환경과 테스트를 실행할 때의 환경이 달라서 발생 합 니다. 이런 문제를 피하기 위해서는 테스트를 실행하는 초기 환경과 레코딩 할 때의 환경을 확실히 동일하게 만들어 줘야 합니다. 예를 들면, 만약 레코딩 시에 테스트 어플리케이션이 실행되고 있다 면 테스트를 실행할 때도 테스트 어플리케이션은 실행되고 있어야 합니다. 만약 레코딩 할 때 테스 트 대상이 되는 웹 페이지의 두 번째 탭이 선택되어 있다면, 테스트를 실행할 때 역시 해당 웹 페이 지의 두 번째 탭이 선택되어 있어야 합니다.

이와 관련된 좀 더 자세한 사항은 TestComplete의 Help 메뉴의 'Handling Playback Errors' 섹션을 참조하세요.

## 9. 멀티 브라우저 테스트

브라우저 테스트 시나리오를 레코딩 한 후 테스트를 실행 했을 때 성공적으로 수행이 되었다면 그 테스트케이스에 간단한 수정만으로 다른 브라우저 역시 테스트를 수행할 수 있습니다. 이것을 크로 스 브라우저 테스팅(cross-browser testing) 혹은 멀티 브라우저 테스팅이라고 합니다. 멀티 브라우저 테스팅은 여러 종류의 브라우저를 하나의 테스트케이스로 테스트가 가능하게 도와줍니다.

이제부터 간단한 수정을 통해서 다른 브라우저와 같이 테스트가 진행 되도록 하겠습니다.

- 키워드 테스트 오퍼레이션 화면으로 이동 합니다.

- 'Run Browser' 오퍼레이션에서 마우스 우클릭 후 'Make Browser Loop' 메뉴를 선택합니다.

Item	Operation	Value	
Run Browser pageWebC pag	<u>Run Test</u> Run Selected Operation         Run from Selected Operation <u>Cut</u> Copy         Paste         Delete         Insert New Operation         Toggle Breakpoint         Enable Operations         Disable Operations         Set Auto-Wait Timeout for Operations         Make Browser Loop         Make Browser Loop         Show Operations Panel         Auto Width         Field Chooser         Panel Options	Ctrl+X Ctrl+C Ctrl+V F9	er.pageWebOrders2





- 브라우저 루프(Browser Loop) 설정창이 발생하면 'Iterate Through All Browser'을 선택한 뒤 'Finish' 버튼을 클릭합니다.

Bro	wser Loop
5	Select Loop Mode Specify the browser loop mode.
Ite	erate Through All Browsers
Ite	rates through all supported browsers that are available on the computer.
It	erate Through Specific Browsers
Se	lect some of the supported browsers for iteration.
	< Back Next > Finish Cancel Help
	NT 8 16 14
	18 291

이 기능은 'Run Browser' 오퍼레이션을 'Browser Loop' 오퍼레이션으로 바꿔 줍니다.

- 'Browser Loop' 오퍼레이션을 제외한 나머지 모든 오퍼레이션을 선택 한 뒤 하위로 이동 시키 는 버튼(→)을 클릭하세요. 그럼 모든 오퍼레이션들은 각각의 브라우저를 반복적으로 테스트 실행할 것입니다.





다음에	보이는	화면이	멀티브라우저	테스팅	오퍼레이션을	적용한	것입니다.
		피나히	27-917	-11	~ [ - ] - ] - ] - ] - ] - ] - ] - ] - ] -	701	것 ㅂ ㅋ ㅋ •

tem	Operation	Value
🖃 🎯 Browser Loop	All Installed br	"http://support.smartbear.com/Sa
🖻 🥥 browser	_	
	Wait	
🖨 🗔 pageWebOrdersLogin		
- 🙀 textboxCtl00Mainconter	itUsern SetText	"Tester"
🚽 🙀 passwordboxCtl00Mainc	ontentP SetText	"test"
🖞 submitbuttonLogin	ClickButton	
	Wait	
🖃 📃 pageWebOrders		
🖏 imagebutton	Click	
	Wait	
🖨 📃 pageWebOrders2		
	itFmwor SetText	"Mark Twain"
🕺 🕺 linkCtl00MaincontentFmv	vorderU Click	
💀 🙀 pageWebOrders	Wait	
🗝 😪 Property Checkpoint		Aliases.browser.pageWebOrders2
🖻 🥥 browser		
pageWebOrders2		
hinkCtl00Logout	Click	
🕞 pageWebOrdersLogin	Wait	111
and the second se	Close	60

- 여기까지 진행된 내용을 'File | Save'를 선택해서 저장합니다.

멀티 브라우저 테스팅에 대한 자세한 사항은 TestComplete의 Help 메뉴의 Preparing Web Browsers' 섹션을 참조하세요.





이제 테스트케이스를 실행 해 보시기 바랍니다.

TestComplete은 테스트 스크립트를 몇 번 반복해서 실행 할 것입니다. 그리고 테스트 수행 중에 각각의 브라우저도 실행 되서 테스트가 진행될 것입니다.

테스트 결과 로그에도 어떤 브라우저에 대한 테스트가 수행되었는지 확인할 수 있습니다.

e	Message	Time 🛆	Priority	Has	Link
i	Nozilla Firefox was launched.	10:01:44	Normal		
0	Navigating to the http://support.smartbear.com/Samples/TestComplete10/WebOr	10:01:59	Normal		
0	The text 'Tester' was entered in the text editor.	10:02:00	Normal		
0	The text 'test' was entered in the text editor.	10-02-00	Normal		
-	he then was clicked with the left neuron he the	10.02.04	1901100	-	
	The property crecipion opposed to recontent Fext equals "Mark Twain").	10:02:06	Normal		
0	The window was clicked with the left mouse button.	10:02:07	Normal		
0	The 'Web Orders Login - Mozilla Firefox' window was dosed.	10:02:08	Normal		
1	Microsoft Internet Explorer was launched.	10:02:09	Normal		
0	Navigating to the http://support.smartbear.com/Samples/TestComplete10/WebOr	10:02:12	Normal	1	
õ	The text 'Tester' was entered in the text editor.	10:02:14	Normal		
	The tout 'tec' was entered in the text editor.	10.02.13	PRESS	-	-
	The property uncoponreposed (one content lext equals "Mark Twain").	10:02:20	Normal		
0	The window was cloked with the left mouse button.	10:02:20	Normal		
۲	The 'Web Orders Login - Windows Internet Explorer' window was closed.	10:02:22	Normal		
4	Google Chrome was launched.	10:02:23	Normal		
0	Navigating to the http://support.smartbear.com/Samples/TestComplete10/WebOr	10:02:25	Normal		
0	The text 'Tester' was entered in the text editor.	10.02.28	Normal		
-	The trites' was entered in the text of the	10.02.37	1900 Hitter		
	The property crecipion possed (one content Text equals "Mark Twain").	10:02:38	Normal		
0	The window was clicked with the left mouse button.	10:02:38	Normal		
0	The 'Web Orders Login - Google Chrome' window was closed.	10:02:41	Normal		
4	Opera was launched.	10:02:43	Normal		1
0	Navigating to the http://support.smartbear.com/Samples/TestComplete10/WebOr	10:02:46	Normal		
0	The text 'Tester' was entered in the text editor.	10.02.49	Normal	-	
-	I we that the text of a	20.03.06	ROTINE		
	The property crecipion passed the contentText equals "Mark Twain").	10:03:19	Normal		
0	The window was cloked with the left mouse button.	10:03:20	Normal	2	
0	The 'Web Orders Login - Opera' window was closed.	10:03:27	Normal		1
4	Apple Safari was launched.	10:03:32	Normal		
0	Navigating to the http://support.smartbear.com/Samples/TestComplete10/WebOr	10:03:35	Normal		
۲	The text 'Tester' was entered in the text editor.	10:03:41	Normal		
٢	The text 'test' was entered in the text editor.	10:03:41	Normal		
0	The button was clicked with the left mouse button.	10:03:42	Normal		
0	The window was clicked with the left mouse button.	10:03:45	Normal		
0	The text 'Mark Twain' was entered in the text editor.	10:03:48	Normal		
0	The window was clicked with the left mouse button.	10:03:48	Normal		-
	The property checkpoint passed (the contentText equals "Mark Twain").	10:03:51	Normal		
0	The window was clicked with the left mouse button.	10:03:52	Normal		
-				-	-

이와 관련된 자세한 사항은 TestComplete의 Help 메뉴의 'Cross-Browser Testing-Overview' 섹션 을 참조하세요.





# 안드로이드(Android) 어플리케이션 테스트

본 문서에서는 TestComplete에서 안드로이드 어플리케이션을 어떻게 테스트 할 수 있는지 설명합니다. 설명하는 과정은 먼저 모바일 테스팅을 위한 환경을 준비하고 모바일을 PC에 연결한 뒤 테스트를 위한 어플리케이션을 설치할 것입니다. 그 후에 간단한 테스트케이스를 생성한 뒤 실행해 보고 그 결과를 분석하겠습니다. 테스트는 가상 스크린을 통해서 모바일 어플리케이션을 실행하고 몇 개 의 체크포인트를 사용해서 데이터를 검증 할 것입니다.

### 테스트 어플리케이션

본 문서에서는 TestComplete에 이미 저장되어 있는 'Orders'라는 어플리케이션을 사용할 것입니다. 그 어플리케이션은 주문 리스트와 주문 추가, 수정, 삭제 기능이 있습니다.



테스트 어플리케이션은 모바일 단말기에 설치되어 사용될 것입니다.



## 1. 테스트 환경 준비

이번 섹션에서는 TestComplete에서 안드로이드 어플리케이션을 테스트 하기 위한 환경을 준비해 야 할 것들을 설명합니다.

Note: 사용자는 모바일 어플리케이션을 테스트 할 때 실제 단말기와 가상 모바일 스크린, 그리 고 Android-x86 가상 기기를 통해서 테스트를 실행할 수 있습니다. 본 문서에서는 실제 단말기를 테스트하기 위한 준비를 설명할 것입니다. 가상 스크린과 Android-x86 가상 기기를 테스트하기 위한 준비과정은 본 섹션과 좀 다릅니다. 이와 관련된 자세한 사항은 TestComplete의 Help 메뉴의 'Preparing Devices, Emulators and Virtual Machines' 섹션 을 참조하세요. 이 부분들을 제외한 나머지 테스트 방식은 본 문서에서 설명하는 것과 대부분 흡사합니다.

### 필수 소프트웨어 설치

JDK와 Android SDK 설치

안드로이드 어플리케이션을 테스트하기 위해서는 PC에 JDK(Java Development Kit) 혹은 JRE(Java Runtime Environment)와 Android SDK의 설치가 필요합니다.

- Note: 만약 위에서 언급된 소프트웨어가 이미 설치되어 있다면 아래 5번 항목부터 진행하면 됩니다.
- 1) JDK 다운로드: <u>http://www.oracle.com/technetwork/java/javase/downloads/index.html</u>
- 2) 다운로드를 마치면 설치 마법사에 따라 설치를 완료하시기 바랍니다.
- 3) SDK 다운로드: <u>https://developer.android.com/sdk/index.html</u>
   기본적으로 같이 다운로드 되는 ADT는 필요하지 않고, 오직 SDK만 필요합니다. SDK 다운로
   드를 위해 'Download for Other Platforms' 섹션에서 'SDK Tools Only'으로 가서 'installer\_rXX.X.-windows.exe'를 다운로드 하시면 됩니다.
- 4) 다운로드를 마치면 설치 마법사에 따라 설치를 완료하시기 바랍니다.
- 5) SDK의 설치가 완료되면 Android SDK Manager을 실행하세요. '시작' 메뉴에서 '모든 프로그 램 | Android SDK Tools | SDK Manager'을 선택하시면 됩니다.



- Android SDK Platform-tools
- Android SDK Build-tools
- Google USB Driver

Tools	
C:\Android SDK	
R	
ame API Rev. Statu	us
Tools	
Android SDK Tools 22.3 🔯 In	stalled
🖉 🦨 Android SDK Platform-tools 🛛 19 🗍 N	lot installed
🛛 🥜 Android SDK Build-tools 19 🗌 N	lot installed
Andraid SDK Build-tools 18.1.1 🗌 N	lot installed
🛛 🥕 Andraid SDK Build-tools 18.1 🗍 N	lot installed
Android SDK Build-tools 18.0.1 🗋 N	lot installed
🗌 🥕 Android SDK Build-tools 17 🗌 N	lot installed
🛓 Android 4.4 (API 19)	
- Android 4.3 (ΔΡ[18)	
Android 1.5 (API 3)	
Extras	
Android Support Repository 3 N	lot installed
Android Support Library 19 N	lot installed
Google Ad Mob Ads SUK	lot installed
Google Analytics App Tracking SDK 3	lot installed
IDeprecated] Google Cloud Messaging for Android Library     3     N	lot installed
Google Play services for Froyo 12 N	lot installed
Google Play services 13 N	lot installed
Google Repository 4	lot installed
Google Play APK Expansion Library 3	lot installed
Google Play Billing Library 5	lot installed
Google Play Licensing Library 2	lot installed
🖉 🔂 Google USB Driver 8 🗌 N	lot installed
🗌 🔯 Google Web Driver 🔰 2 🛄 N	lot installed
Intel x86 Emulator Accelerator (HAXM)     3      N	lot installed
🗸 Updates/New 🖉 Installed 📄 Obsolete Select <u>New</u> or <u>Updates</u> 🚺 Install 3	packages
API level © Repository Deselect All Defete	packages _
API level      Repository     Deselect All     Deselect All	ete

7) 'Install packages'를 클릭한 뒤 라이선스에 동의하시고 설치 마법사에 따라 진행하세요.

8) 'SDK Manager'을 종료하세요.

# USB 드라이버 설치

위에서 언급한 것과 같이 본 문서에서는 실제 단말기를 대상으로 테스트를 진행 할 것입니다. 그 래서 TestComplete가 설치된 컴퓨터에 USDB 드라이버도 설치가 필요합니다. Android 문서를 보면 'OEM USB Drivers'에 관련된 정보를 확인하실 수 있습니다.

http://developer.android.com/tools/extras/oem-usb.html

## TestComplete의 Plug-ins 설치 및 구성

Android 어플리케이션을 테스트 하기 위해서는 TestComplete에 모바일 모듈을 설치해야 합니다. 이 모바일 모듈은 플러그 인을 통해서 자동으로 구성할 수 있습니다. 만약 모바일 모듈의 설치가 되 어 있지 않다면 TestComplete를 재설치를 통해 설치 마법사에서 모바일 모듈을 같이 설치 해주어야 합니다.

#### TestComplete 구성

플러그 인의 활성화를 위해 TestComplete에서 JDK와 Android SDK의 설치 경로를 따로 설정해 줘 야 할 수도 있습니다. TestComplete은 설치 시 설치 마법사를 통해 PC에서 경로를 찾으려고 시도 합니다. 하지만 마법사를 통해 경로를 찾지 못 하는 경우 사용자가 TestComplete의 메뉴를 통해 직 접 설정 해 줘야 합니다.

경로 설정이 올바른지 확인을 위해 아래와 같이 진행하시기 바랍니다.

- ① TestComplete을 실행합니다.
- ② 'Tool | Options' 메뉴를 선택합니다. 그럼 'Options' 창이 발생할 것입니다.
- ③ 'Options' 창 왼쪽 메뉴에서 'Engines | Mobile | Android'를 선택합니다.
- ④ 'Path to Android SDK'의 입력 필드에서 Android ADK가 설치된 경로를 지정해 줍니다.
- ⑤ 'Path to Java Runtime'의 입력 필드에서 Java가 설치된 경로를 지정해 줍니다.

) General Engines	Android		
General Custom Controls	Auto-wait timeout for Ar	ndroid devices, s: 60 👘	
Events	Path to Android SDK:	C: \Android \sdk	
Java Bridge Log	Path to Java Runtime:	C:\Program Files\Java\jre7	
Ceneral	Terminate ADB if it do	es not respond	

⑥ 다시 'Engines | Mobile | General' 메뉴를 선택합니다.

⑦ 'the Automatically display Mobile Screen on OnScreen operation' 메뉴를 체크 설정합니다.

⑧ 'the Automatically display Mobile Screen on recording' 메뉴를 체크 설정 합니다.

⑨ 'OK' 버튼을 클릭해서 설정을 저장하고 종료합니다.

### 모바일 단말기 준비

PC에 모든 소프트웨어 준비사항을 마친 뒤에 모바일 단말기의 메뉴설정을 합니다.

### 단말기에서 개발자 옵션 설정

실제 안드로이드 단말기를 테스트 하기 위해서는 단말기에서 개발자 옵션을 활성화 해주어야 합니다.

1) 안드로이드 설정 메뉴에서 '설정 | 개발자 옵션'을 선택합니다.
Note: 개발자 옵션 메뉴는 안드로이드 단말기와 버전에 따라 다를 수 있습니다. 테스트가 진행 되는 단말기의 버전에 따라 안드로이드에서 제공되는 정보를 참조하시기 바랍니다.

- 73 -

2) 'USB 디버깅' 메뉴와 '켜진 상태로 유지' 메뉴를 체크 설정합니다.

- USB 디버깅: 이 메뉴를 체크 설정하면 단말기와 ADB(Android Debug Bridge)간에 통신을 할 수 있습니다. 다시 말하면 단말기를 ADB를 통해서 TestComplete와 통신할 수 있다는 것입 니다.

- 켜진 상태로 유지: 디버깅하는 동안 단말기가 슬립(sleep) 모드로 빠지는 것을 방지합니다.

3) 변경 상태를 저장합니다.

### 단말기를 PC에 연결

단말기를 켠 후에 PC에 USB 케이블을 통해서 연결합니다. ADB는 연결된 단말기를 자동으로 인식 합니다.

### 단말기 연결 확인

단말기가 PC에 연결이 정상적으로 이루어졌는지 확인합니다.

TestComplete의 툴 바에 'Show Mobile Screen' 버튼을 클릭합니다. 만약 단말기가 정상적으로 연 결이 이루어졌다면 가상 스크린이 나타날 것입니다. 하나 이상의 단말기가 PC에 연결이 되어 있다 면 'Show Mobile Screen' 메뉴에서 'Select Current Device' 창을 통해서 원하는 단말기를 선택해서 스크린을 나타나게 할 수 있습니다. 단말기 리스트에 원하는 단말기가 표시되는지 확인하시고, 단말 기가 확인되면 선택하고 'OK' 버튼을 클릭하면 됩니다.

### 안드로이드 에이전트(Agent) 설치

안드로이드 에이전트는 테스트를 위해 TestComplete에 설치된 안드로이드 어플리케이션의 데이터 를 설정 하는데 도움을 줍니다. 에이전트는 모바일 단말기에 설치되어야 합니다. 에이전트를 설치하기 위해서는 아래와 같이 진행합니다.

- ① 'Mobile Screen' 창을 열기 합니다.
- ② 툴 바에서 'Install Android Agent' 버튼()을 클릭합니다.

#### 어플리케이션 테스트를 위한 준비

안드로이드 어플리케이션의 오브젝트를 TestComplete에서 인식하게 하기 위해서는 추가적인 설정 이 필요합니다. 이 방법을 자세히 알고 싶으시다면 TestComplete의 Help 메뉴의 'Instrumenting Android Applications' 섹션을 참조하세요.

TestComplete에 설치된 'Orders' 어플리케이션은 화이트 박스 테스팅을 위한 준비가 되어 있습니

다. 그러므로 이와 관련된 설정은 본 문서에서는 생략하도록 하겠습니다.

이제 TestComplete에는 테스트를 위한 모바일 단말기와 어플리케이션이 준비되어 있습니다. 다음 섹션에서는 어떻게 테스트 프로젝트를 생성하고 간단한 테스트를 레코딩 한 뒤 실행하는 것을 설명 해 드리겠습니다.

## 2. 테스트 프로젝트 생성

이제 새로운 테스트 프로젝트를 생성해 보겠습니다.

- 1) 만약 이미 생성된 테스트 프로젝트나 스윗(suit)이 있다면 닫으시기 바랍니다. TestComplete의 메인 메뉴에서 'File | Close'를 선택하시면 됩니다.
- 2) 그리고 TestComplete 메인 메뉴에서 'File | New | New Project' 메뉴를 클릭합니다. 그럼 새 로운 프로젝트를 생성하기 위한 마법사가 나타납니다.
- 3) 마법사 창에서 프로젝트 이름과 프로젝트 파일이 저장될 위치를 선택합니다. 본 문서에서는 프로젝트 이름을 'AndroidOrders'로 하고 저장 위치는 기본으로 제공되는 위치로 설정하겠습니 다.

Create New F	Project
Specify the r	name, location and other attributes of a new project.
Project name:	AndroidOrders
Location:	C:\Users\johnsmith\Documents\TestComplete 10 Projects\
	More
@ Learn more	about these settings
	Next Finish Cancel

4) 다음 페이지에서 'Android applications'를 선택합니다.

- 5) 다음 페이지에서 프로젝트에 테스트 어플리케이션을 추가할 수 있습니다.
  - 'Add' 버튼을 클릭하면 'Add Android Tested Application' 마법사가 발생할 것입니다.
  - 마법사 창에서 폴더 버튼()을 클릭하고 폴더 검색 창에서 안드로이드 'Orders' 패키지 파



일을 선택합니다. 파일 경로는,

- Win7이나 Vista 이후 버전일 경우, <u>C: #Users #Public WPublic Documents #TestComplete 10 Samples #Mobile #Android #Orders</u> <u>#Orders Application #bin #Orders.apk</u>
- Win XP이나 Win Server 2003 이전 버전일 경우, <u>C: \U00cuments and Settings \U004All Users \U004BlackBared Documents \U0047TestComplete 10 Samples \U0044Mobile</u> <u>\U004Android \U004Fest \U004Fest Application \U0044bin \U004Fest.apk</u>
- Note: 때로는 '공유 문서(Shared Documents)'나 '공용 문서(Public Documents)' 폴더에 있는 경우도 있습니다.
- 어플리케이션을 설치할 때 'Deploy to the device on start' 메뉴를 체크 설정하세요. 그럼 TestComplete가 실행될 때 해당 어플리케이션이 같이 실행됩니다.
- 'Keep the data and cache directories on removal' 메뉴는 설정 해제 하세요. 그래야 테스트 어플리케이션이 매번 초기 상태에서 실행될 수 있습니다.
- 그 다음 'Next' 버튼을 클릭한 뒤 'Finish' 버튼을 눌러서 마법사를 종료하세요.
- 'Autorun' 메뉴는 체크 설정하세요. 그래야 레코딩이 시작될 때 자동으로 'Orders' 어플리케 이션이 실행될 것입니다. 만약 설정이 되어 있지 않으면 레코딩할 때 사용자가 실행 메뉴를 통해서 어플리케이션을 실행시켜 주어야 합니다.

'Next' 버튼을 클릭하세요.

- 6) 다음 페이지에서 레코딩 및 테스트 실행 할 때 이미지와 관계되는 'Test Visualizer' 설정을 할 수 있습니다. 본 문서에서는 체크 설정하고 'Next' 버튼을 클릭하겠습니다.
- 7) 마지막으로 스크립트 언어를 선택해서 설정을 종료합니다.

설정이 종료되면 TestComplete의 'Project Explorer' 패널에 새롭게 생성된 프로젝트가 보일 것입니 다.

## 3. 테스트케이스 생성

## TestComplete에서 테스트케이스 생성

TestComplete에서는 두 가지 방법으로 테스트케이스를 생성할 수 있습니다.

- 직접 테스트케이스를 생성
- 테스트 레코딩 방식을 이용해서 생성

직접 테스트케이스를 생성하는 경우 사용자는 필요한 오퍼레이션이나 테스트 시나리오를 스크립 트나 키워드 테스트에 맞춰서 생성해야 합니다. 이 방법은 매우 강력하면서도 테스트케이스를 유동 적으로 사용할 수 있다는 장점이 있습니다. 하지만 그 만큼 테스트케이스를 생성하는데 많은 경험이 필요합니다.

직접 테스트케이스를 생성하는 방법의 단점은 많은 시간을 필요로 하고 많은 다른 문제들이 발생 할 수 있습니다. 예를 들면 직접 테스트케이스를 생성하기 위해서는 테스트 어플리케이션의 클래스 네임(name)이나 오브젝트를 알아야 합니다. 이런 불편함을 해소하기 위해 TestComplete에서 테스트 레코딩만으로 편리하게 테스트케이스를 생성할 수 있는 기능을 제공합니다. 테스트시나리오를 한번 만 레코딩 하면 TestComplete에서 테스트 시나리오 뿐 만 아니라 스크립트 라인이나 테스트 오브젝 트들도 자동으로서 인지해서 키워드 테스트 스크립트로 자동으로 변환되어 생성됩니다. 이렇게 진행 되는 형태를 '테스트 레코딩(recording a test)'이라고 합니다. 이런 방식은 사용자의 역량이 많지 않 아도 편리하게 사용할 수 있기 때문에 매우 유용한 방식입니다. 그래서 본 문서에서는 테스트 레코 딩 방식으로 가이드를 진행할 것입니다.

## 테스트케이스 레코딩

테스트 레코딩을 위해서는 아래와 같이 세 가지 단계가 필요합니다.

1) 사용자는 TestComplete의 메인 메뉴나 툴 바에서 'Test | Record | Record Keyword Test' 혹 은 'Test | Record | Record Script' 메뉴를 통해서 레코딩을 시작할 수 있습니다. 아니면 시작 페이지에서 'Record a New Test' 버튼만 클릭해도 레코딩을 시작할 수 있습니다.

레코딩을 시작하게 되면 TestComplete 화면은 레코딩 모드로 전환이 됩니다. 레코딩 모드에 서는 TestComplete의 메인 화면은 자동으로 축소 되고 레코딩 툴 바와 모바일 스크린 화면이 나타납니다.

사용자는 레코딩 툴 바를 통해서 레코딩과 관련된 기능들을 제어할 수 있습니다. 일시정지 (Pause)나 정지(Stop) 그리고 레코딩 타입을 변경할 수 있습니다.



모바일 스크린 창에는 연결된 단말기의 화면이 나타납니다. 이 창은 모바일 어플리케이션을 테스트 하기 위해 사용됩니다. 이 창은 'Tools | Options | Mobile | General' 메뉴를 통해서 설 정할 수 있습니다.

 레코딩을 시작한 후에 매뉴얼로 테스트 시나리오를 수행합니다. 예를 들어 테스트 어플리케이 션을 실행하고, 시나리오에 따른 버튼을 클릭하거나 메뉴를 선택하고 텍스트를 입력하는 등의 행위를 합니다.  3) 테스트 시나리오 수행을 완료한 후에 툴 바에 있는 종료 버튼(■)을 클릭해서 레코딩을 종료 합니다.

레코딩과 관련된 자세한 사항은 TestComplete의 Help 메뉴에서 'Recording in TestComplete' 항목 을 참조하시기 바랍니다.

### 'Orders' 어플리케이션의 테스트 계획

- 테스트 목적: 테스트 어플리케이션(Orders) 실행 후 'Edit Order' 창에서 기존 항목에 대해 수 정이 되는지, 수정된 사항에 대해 정상적으로 변경이 되었는지 확인합니다.
- 테스트 시나리오: 주문 항목의 수정을 통해 주문 리스트가 변경되었는지 검증하고, 사용자는 시뮬레이션을 통해 어플리케이션에 테스트 시나리오에 따라 진행하고 나서 어떤 항목이 변경 되었는지 확인합니다.
- 테스트 결과 확인: 테스트 시나리오에 따라 변경된 사항이 정상적으로 적용 되었다면 Order 리스트에 해당 사항이 반영될 것입니다. 이 부분의 검증을 위해 기대 값과 실제 값을 비교할 것입니다. 그렇게 하기 위해선 두 개의 값을 비교하는 오퍼레이션을 삽입할 것입니다. 해당 오 퍼레이션은 테스트 로그에서 결과를 비교하고 우리는 그 값이 Pass인지 Fail인지 확인할 수 있 습니다.

이와 관련해서 좀 더 자세한 사항은 TestComplete의 Help 메뉴에서 'Planning Tests' 항목을 참조하시기 바랍니다.

## 'Orders' 어플리케이션의 테스트 레코딩

이제 'Orders' 어플리케이션을 키워드 테스트 방식으로 레코딩 해 보겠습니다.

- 모바일 스크린 창을 띄우세요. TestComplete은 오직 모바일 스크린에서만 수행되는 것들만 레코딩이 되고 다른 창이나 단말기에서 직접 실행되는 것들에 대해서는 레코딩이 되지 않습니 다.
- 툴 바에서 'Show Mobile Screen' 버튼을 클릭 하세요
- 2) TestComplete 화면을 보면, 새로운 프로젝트가 생성될 때 자동으로 비어있는 키워드 테스트 항목(Test1)이 생성됩니다. 그래서 처음 레코디을 할 때에는 비어있는 테스트케이스에 레코딩 을 시작합니다.



미리 생성되어 있는 테스트케이스(Test1)를 확인하기 위해서는,

- 'Project Explorer' 패널로 이동
- 'KeywordTests' 노드를 확장(+) 선택
- 'Test1' 노드를 더블 클릭



3) 툴 바의 'Append To Test' 버튼(🛸)을 눌러서 테스트를 시작합니다.

그러면 자동으로 레코딩 툴 바가 나타날 것입니다. 만약 본 문서와 테스트 해야 하는 화면이 겹쳐지면 본 문서를 축소하거나 이동 시켜서 화면이 겹치지 않도록 하시기 바랍니다.

레코딩 툴 바의 기본 화면은 아래와 같습니다.



확장 버튼(\*\*\*)을 클릭해서 아래와 같이 추가 메뉴들을 볼 수 있습니다.







4) 레코딩을 시작하게 되면 자동으로 'Orders' 어플리케이션도 실행될 것입니다. 만약 자동으로 실행되는 기능이 해제되었다면 아래에 나타나 있는 메뉴를 통해 직접 실행할 수도 있습니다.



TestComplete은 테스트 어플리케이션이 시작되는 것도 레코딩 합니다. 이는 나중에 레코딩 이 종료되고 스크립트가 자동 생성되면 확인할 수 있습니다.

5) 모바일 단말기에 'Orders' 어플리케이션이 설치되고 실행될 때 까지 기다립니다. 그리고 나면 모바일 스크린에 어플리케이션의 초기 화면이 나타날 것입니다.

	MERUS	A .   Bour	mpps • 🔟	20 日 日	18
📬 Orders					1
New	Ø	Edit	Ë	Delete	
John Smith Jr 1. MyMoney, 07.05.2009					
Clare Jefferson 2. FamilyAlburn, 05.04.2009					
Susan McLaren 1. MyMoney, 12.05.2008					
Charles Dodgeson 1. MyMoney, 12 05 2009					
Steve Johns 1. ScreenSaver, 04.04.2008					
Samuel Clemens 2. Mythoney, 01.03.2012					
Bob Feather 1, PamilyAlburn, 03.12,2010					
-		$\Box$			

만약 본 문서와 모바일 스크린이 겹쳐진다면 본 문서의 사이즈를 변경하거나 다른 곳으로 이동 시켜서 테스트 화면이 겹치지 않도록 하시기 바랍니다.



```
6) 모바일 스크린 화면에서 'Samuel Clemens's'를 클릭합니다.
```







7) 'Orders' 어플리케이션에서 Edit 버튼을 클릭하면 아래와 같은 수정 화면이 나타날 것입니다.

🛐 📴 📲 📕 Nexus 7 🔹 Bun Apps	
) 🕸 📾	15:25
👘 Orders	- Community
Product:	
MyMoney	2
Quantity:	
n Price our unit (S):	1
100	
Discount (%)	
0	
Total (S):	
200.0	
Customer Name:	
Samuel Clemens	
Street	
3, Garden st.	
City:	
Hillsberry, UT	
State	
US	
O Vian	
Master Card	
American Express	
Card Number	
123456789012	
Expire date (mm/yy):	
	Transmission in which the
J D	

8) 상세화면에서 고객 이름을 변경하세요.

'Samuel Clemens' 텍스트를 삭제하고 'Mark Twain'으로 입력하세요. 모바일 스크린에서 PC 의 키보드로 텍스트를 입력할 수 있습니다.



9) 스크롤을 내려서 하단에 있는 'OK' 버튼을 클릭하세요. 그럼 변경된 값이 저장되고 다시 'Orders' 리스트로 돌아올 것입니다.



10) 그리고 나서 값을 비교할 수 있는 기능을 삽입하겠습니다. 이것은 'Mark Twain'으로 변경된 고객 이름이 정상적으로 수정되었는지 검증할 것입니다.

본 문서에서는 그것을 체크포인트(Checkpoint)라고 합니다. TestComplete에서는 여러 가지 타입의 체크포인트를 제공하는데, 그것들 중에 검증 상황에 맞춰서 사용하시면 됩니다. 체크포 인트와 관련된 정보는 TestComplete의 Help 메뉴에서 'Checkpoints' 섹션을 참고하시면 됩니 다. 여러 가지 체크포인트 타입 중에 가장 많이 사용하는 타입은 'Property checkpoint'(속성검 증)입니다. 이 체크포인트 타입은 어플리케이션을 제어하는 데이터의 값을 검증하는 기능을 수 행합니다. 그러므로 본 문서에서는 이 체크포인트 타입을 사용해 보겠습니다.



- 체크포인트의 드롭다운 버튼을 클릭하고, 리스트 중에 'Create Property Checkpoint' 메뉴(%) 를 클릭하세요.



이 메뉴를 선택하면 'Property Checkpoint'와 관련된 마법사가 발생할 것입니다. 이 마법사 는 속성을 검증할 수 있는 체크포인트를 설정하는데 가이드를 할 것입니다.

reate Property Checkpoint	
Select object Choose an object whose property will be compared.	
<ul> <li>Drag the target to point to the object</li> <li>Press the left mouse button over the target icon and drag the button to select this window or control. Show me how it works</li> <li>Point and fix (Shift+Ctrl+A)</li> <li>Press this button and then move the mouse pointer to the des Shift+Ctrl+A to select the object. Show me how it works.</li> </ul>	e icon to the desired window or control. Release the <u>5.</u> sired window, control or any other object on screen. Pre
<u>O</u> bject:	Highlight
Yreview:	

- 마법사의 첫 번째 페이지에서 타겟처럼 보이는 버튼()을 클릭 하고 드래그 하세요.



마법사 창이 자동으로 최소화 될 때까지 잠시 기다렸다가 주문 리스트(orders list)가 있는 창으로 마우스를 드래그 하세요. 마우스를 드래그 하는 중에 TestComplete에서는 마우스 커서가 지나가는 곳의 오브젝트를 인식해서 붉은 색 박스로 표시됩니다.

'Mark Twain' 리스트에 마우스 커서를 위치한 뒤 붉은 박스가 발생한 상태에서 드래그 하던 마우스 버튼을 뗍니다.





- 드래그 하던 마우스 버튼을 떼면, 마법사 창이 다시 발생하고 선택된 오브젝트의 이미지와
 이름이 마법사 창에 아래와 같이 표시될 것입니다.

Crea	ate Property Checkpoint
Choos	se an object whose property will be compared.
	Drag the target to point to the object
1999 1997	Press the left mouse button over the target icon and drag the icon to the desired window or control. Relea the button to select this window or control. <u>Show me how it works.</u>
	Point and fix (Shift+Ctrl+A)
-13	Press this button and then move the mouse pointer to the desired window, control or any other object on screen. Press Shift+Ctrl+A to select the object. Show me how it works.
	Select object in Visualizer
PO	Press this button. This will open a dialog box with a Visualizer image in it. Click a window or control in the image to select it. This works only if there is a selected image in the Test Visualizer panel.
<u>O</u> bject	
M	ark Twain
2. M	lyMoney, 02.03.2012
@ Lea	Im more about these settings

'Next' 버튼을 클릭하세요.

다음 마법사 페이지에서는 오브젝트의 속성과 필드를 선택할 수 있는 리스트가 나타날 것입
 니다. 이것은 TestComplete에서 오브젝트에 포함된 속성을 제공하는 것으로 테스트 어플리
 케이션에 대한 속성과 각 필드가 정의되어 있습니다.

데이터를 검증하기 위해서 어플리케이션에 정의된 몇 개의 필드를 사용할 것입니다. 오브 젝트의 필드를 보기 위해서는 'View more members (Advanced View)' 링크를 클릭하세요.





- 'Search' 박스에서 'mText1'을 입력하세요. 그러면 동일한 텍스트가 필터링 돼서 보여 질 것 입니다.

¢	reate Property Checkpoint
Sele	ect an object property to compare.
Proc	rocess_example_orders.RootLayout.ListView_listView1.TwoLineListItem_NO_ID mText1
Ad	Vanced view View basic members (Basic view)
6	B Private
	mText1 (Object)
	Protected
Filtered	
01	Learn more about these settings
	Next Cancel

- 'mText1' 필드를 선택하세요. 이 필드는 오브젝트에 포함된 것입니다. 오브젝트의 다른 항목 들도 보길 원하시면 메뉴를 확대 버튼(+)을 클릭하세요.





- 검색어 입력 필드에 'mText'을 입력하세요.

Prop	perties Fields		
Ad	vanced view <u>View basic memb</u>	ers (Basic view)	
	E Private		
	mText	Mark Twain	[
	mTextColor	(Object)	(
	mTextDir	(Object)	[
	mTextEditSuggestionItemLayo	u 17367257	
	mTextPaint	(Object)	(
	mTextSelectHandleLeftRes	17302986	
	mTextSelectHandleRes	17302987	
red	mTextSelectHandleRightRes	17302988	
ilte	Protected		
-	Public		

- 리스트에서 'mText' 속성을 찾은 뒤 이 속성을 선택하고 'Next' 버튼을 클릭하세요.

N Y



- 다음 마법사 페이지에서는 'value' 박스가 나타나는데 이 데이터를 통해 값을 검증하고 비교 할 수 있습니다.

Specify comparison set	tings.		
Object:			
Aliases.Device.Process_example	_orders.RootLayout.ListView_listVie	ew1.TwoLineListItem_NO_ID.mText	1
Property:			
mText			
Condition:			
Equals			
Learn more about these s	rttings		
Learn more about these s	rttings		

'Finish 버튼을 클릭해서 체크포인트 생성 마법사를 종료하세요.

11) 레코딩 툴 바에서 종료 버튼(■)을 클릭해서 레코딩을 종료하세요. 그럼 레코딩 된 테스트들이 자동으로 테스트케이스로 저장될 것입니다.



# 4. 레코딩된 테스트케이스 분석

레코딩 완료되고 나면 키워드 기반 테스트 스크립트가 표시될 것입니다.



레코딩 된 결과를 보면 위의 사진과 비슷하겠지만, 약간은 다를 수 있습니다. 예를 들어 불필요한 클릭을 했다면 그 부분이 스크립트에 추가되어 있을 수도 있습니다. 레코딩 하면서 'Orders' 어플리 케이션에 수행된 사항에 대해 테스트 스크립트에 모두 반영되어 있을 것입니다. 본 문서에서는 그것 을 오퍼레이션(operations)이라고 합니다.

아래에 'Test Visualizer' 패널에는 레코딩 할 때 같이 저장된 이미지가 표시됩니다.

Call Object M Call Object M Find Object	Button_Ok	TouchButton Aliases.Device.Pro	Checks whether the '
🗿 Test Visualizer			
Update	🐻 🗌 Update	Update	Update
Conception of the second	The second secon	E particular de la construir	Sectors The sector of the sec

이미지들은 각각의 오브젝트들에 대한 테스트 수행에 대해 이해를 돕기 위해 이미지 형태로 보여 줍니다. 그 이미지들은 사용자가 터치, 드래그, 텍스트 입력 등의 시나리오에 따라 수행한 내용과 일 치해서 보여 줍니다.

만약 어떤 오퍼레이션 하나라 선택하면 'Test Visualizer' 패널에는 그와 매칭되는 이미지가 하이라 이트 되어 테스트 시나리오에 대한 이해를 도와줍니다. 이와 관련된 자세항 사항은 TestComplete의



Help 메뉴에서 'Test Visualizer' 섹션을 참조하시기 바랍니다.

첫 번째 오퍼레이션은 'Select Device'입니다. 이것은 어떤 모바일 단말기가 실행되는지를 선택하는 기능을 합니다. 모든 단말기가 이 오퍼레이션을 통해 실행될 수 있습니다.

Item	Operation	Value	Description
Select Device		"Nexus 7",	Specifies the mobile dev
💀 🚯 Run TestedApp	Orders		Runs the "Orders" teste
🗇 📫 Device			
Process_example_orders			
- 💫 ListView_listView1	TouchItem	5	Simulates a single touch
	TouchButton		Performs a single touch
🔤 🔂 EditText_cust_name	SetText	"Mark Twain"	Enters text in the edit c
Button_Ok	TouchButton		Performs a single touch
Roperty Checkpoint		Aliases.Device.Process	Checks whether the 'mT
		1.8° ~ 1/1	

두 번째 오퍼레이션은 'Run TestedApp' 입니다. 이 오퍼레이션은 테스트 할 어플리케이션을 구동 시키는 기능을 합니다. TestComplete은 레코딩 툴 바를 통해서 어플리케이션을 구동시키거나 직접 어플리케이션을 구동 시키는 경우에 자동으로 이 오퍼레이션을 통해 레코딩 합니다.

Item	Operation	Value	Description
Select Device		"Nexus 7",	Specifies the mobile dev
Run TestedApp	Orders		Runs the "Orders" teste
Device     Process_example_orders     Device     Process_example_orders     ListView_listView1     Button_Edit     G EditText_cust_name     Button_Ok     Property Checkpoint	TouchItem TouchButton SetText TouchButton	5 "Mark Twain" Aliases.Device.Process	Simulates a single touch Performs a single touch Enters text in the edit c Performs a single touch Checks whether the 'mT

그 다음 오퍼레이션들은 사용자가 어플리케이션에서 진행된 시나리오에 따라 레코딩 된 것들입니 다. 주문 리스트에서 항목을 선택하고, 수정 버튼을 누르고, 값을 변경한 뒤 'OK' 버튼을 누르는 것 까지 나타납니다.

Item	Operation	Value	Description
Select Device		"Nexus 7",	Specifies the mobile dev
🖶 🖶 Run TestedApp	Orders		Runs the "Orders" teste
Device           Image: second system           Image: second			
🖳 🔂 ListView_listView1	TouchItem	5	Simulates a single touch
Button_Edit	TouchButton		Performs a single touch
- 🔂 EditText_cust_name	SetText	"Mark Twain"	Enters text in the edit c
Button_Ok	TouchButton		Performs a single touch
Roperty Checkpoint		Aliases.Device.Process	Checks whether the 'mT
		IFX 3	

화면을 터치 하거나 텍스트를 입력하는 등의 테스트 수행에 대한 자세한 정보는 TestComplete의 Help 메뉴의 'Simulating User Actions Over Android Devices' 섹션을 참조하시기 바랍니다.

마지막으로, 레코딩 중에 추가했던 값을 비교하는 체크포인트 오퍼레이션입니다.

 Item	Operation	Value	Description
Select Device		"Nexus 7",	Specifies the mobile dev
🗝 🚯 Run TestedApp	Orders		Runs the "Orders" teste
🖶 👘 Device			
Process_example_orders			
	TouchItem	5	Simulates a single touch
Button_Edit	TouchButton		Performs a single touch
- 🔂 EditText_cust_name	SetText	"Mark Twain"	Enters text in the edit c
Button_Ok	TouchButton		Performs a single touch
Property Checkpoint		Aliases.Device.Process	Checks whether the 'mT

위에서 보신 바와 같이, TestComplete은 하나의 모바일 단말기에서 수행된 오퍼레이션들에 대해서 그룹으로 묶어서 관리합니다. 그룹 관리는 오퍼레이션들에 대한 이해를 좀 더 수월하게 해 주고, 어 플리케이션에 내재되어 있는 오브젝트의 계층에 대한 정보를 제공합니다.



본 문서에서는 하나의 모바일 단말기로 하나의 시나리오를 수행 했습니다. 그래서 스크립트에는 두 개의 그룹 노드로 구성되어 있습니다. 'Device' 노드는 테스트 실행된 단말기를 나타내고, 'Process' 노드는 모든 시나리오 수행에 대한 사항들이 나타납니다.

	Item	Operation	Value	Description
	Select Device		"Nexus 7",	Specifies the mobile dev
Grouping nodes	🗝 🌗 Run TestedApp	Orders		Runs the "Orders" teste
	🖶 📫 Device			
	- 💫 ListView_listView1	TouchItem	5	Simulates a single touch
		TouchButton		Performs a single touch
	- 🕞 EditText_cust_name	SetText	"Mark Twain"	Enters text in the edit c
	Button_Ok	TouchButton		Performs a single touch
	Property Checkpoint		Aliases.Device.Process	Checks whether the 'mT

'Object Browser' 패널에 보면 테스트 프로세스에 따라 진행된 오브젝트 이름들이 다르다는 것을 볼 수도 있습니다. 예를 들면, 테스트 당시 'Process\_example\_orders;'가 'Object Browser'에서 프로세 스 이름이 'Process("smartbear.example.orders")'로 쓰여졌다는 것을 알 수 있습니다. 수정(Edit) 버튼 을 테스트 할 때는 'Button\_Edit'이었는데, 확인해 보면 'Button("editButton")'으로 되어 있습니다.

그것은 TestComplete 내에서 자동으로 레코딩 중에 오브젝트의 'custom names'을 수집하고 사용 하기 때문입니다. TestComplete은 기본으로 제공되는 이름을 이해하는데 어려울 수 있기 때문에 레 코딩 시 수집된 이름들을 매핑 시킵니다. 매핑된 이름을 사용하는 것은 테스트를 보다 쉽고 이해하 기 편하게 만들어 줍니다. 이에 대한 자세한 사항은 TestComplete의 Help 메뉴의 'Name Mapping' 섹션을 참조하세요.

# 5. 테스트케이스 실행

이제 레코딩 된 테스트케이스를 실행해 보겠습니다.

테스트케이스를 실행하기 전에 확인해야 할 부분은 테스트케이스를 생성할 당시와 똑같은 환경을 구성해 주어야 한다는 것입니다. 예를 들어, 테스트가 되어야 할 어플리케이션을 실행하기 위해 이 어플리케이션을 가지고 있는 모바일 단말기가 연결이 되어 있어야 합니다. 그리고 'Select Device' 오 퍼레이션을 통해 해당 단말기를 실행할 수 있습니다. 대부분의 경우 테스트 단말기에 어플리케이션 을 미리 설치해서 실행되도록 합니다. 본 문서의 경우 테스트 어플리케이션의 실행을 위해 'Run TestedApp' 키워드 오퍼레이션이 사용되었습니다.



Item	Operation	Value	Description
Select Device		"Nexus 7",	Specifies the mobile dev
	Orders		Runs the "Orders" teste
🖶 👾 Device			
Process_example_orders			
🙀 ListView_listView1	TouchItem	5	Simulates a single touch
	TouchButton		Performs a single touch
EditText_cust_name	SetText	"Mar <mark>k T</mark> wain"	Enters text in the edit c
Button_Ok	TouchButton		Performs a single touch
Report Checkpoint		Aliases.Device.Process	Checks whether the 'mT

테스트 실행을 위한 가장 간단한 방법은 툴 바에서 'Run Test' 버튼())을 클릭하는 것입니다.

TestComplete은 테스트를 실행할 것입니다. 테스트 스크립트를 통해서 테스트 어플리케이션이 실 행되고 고객 이름을 변경한 뒤 변경된 값을 검증하는 순서로 진행될 것입니다.

Note: 테스트가 실행되는 동안 마우스를 움직이거나 키를 입력하는 등의 행위를 해서는 안 됩 니다. 그런 행위로 인해 테스트 실행에 오류가 생길 수 있습니다.

테스트 실행이 완료가 되면 TestComplete은 자동으로 결과를 표시합니다.

테스트 실행과 관련해서 몇 가지 더 말씀드리면,

• 테스트가 실행되는 동안 TestComplete은 화면 상단 코너 부분에 인디케이터를 표시하게 됩니다.



인디케이터는 테스트가 수행되는 상황에 대한 메시지를 보여줍니다.

• TestComplete은 테스트케이스가 모두 완료될 때까지 실행됩니다. 만약 사용자가 실행을 멈 추고자 한다면 인디케이터에서 'Stop' 버튼()을 클릭하거나 TestComplete 메인 화면에서 'Test | Stop'를 선택하면 됩니다.

테스트 실행 중에 일시정지 하고 싶다면 정지 버튼(**…**)을 클릭하세요. 일시정지 되어 있는 동안 에는 사용자가 테스트 외에 다른 행위를 해도 상관없습니다. 예를 들면, 사용자는 일시 정지 시키고 나서 'Watch List' 패널이나 'Locals' 패널 혹은 'Evaluate' 창에서 테스트 로그를 체크 하거나 사용된 오브젝트를 확인하는 등의 행위를 할 수 있습니다. 관련해서 자세한 사 항은 TestComplete의 Help 메뉴의 'Debugging Tests' 섹션을 참조하세요.

테스트 실행과 관련된 더 자세한 정보는 TestComplete의 Help 메뉴의 'Running Tests' 섹션을 참 조하세요.

# 6. 테스트 결과 분석

TestComplete은 수행된 모든 테스트에 대해서 테스트 로그를 보관합니다. 테스트 로그를 확인할 수 있는 링크는 'Project Explorer' 패널의 아랫부분에 'Project Suite Logs | Orders Log' 카테고리에 서 확인할 수 있습니다. 이 로그는 프로젝트나 프로젝트 스윗(suit)에서 기본으로 제공되는 공간 입 니다. 하나의 결과 노드(node)는 테스트 실행하는 횟수와 부합 됩니다. 노드의 왼쪽에 있는 이미지는 테스트가 성공했는지 아닌지를 표시해 줍니다.



TestComplete은 테스트 실행이 완료 되면 자동으로 테스트 로그를 생성합니다. 그렇다는 것은 테 스트 결과는 테스트가 끝나기 전까지는 표시되지 않는다고 말씀드릴 수 있습니다. 만약 테스트 중간 에 테스트 결과를 보고 싶다면 테스트 중에 일시정지(pause)를 하고 볼 수는 있습니다.

우리는 사전에 한 개의 테스트에 대해 한 번 실행을 했기 때문에 위에 보시는 바와 같이 'Project Explorer' 패널에 한 개의 결과만 나타납니다. 그리고 기본적으로 TestComplete은 테스트 실행이 완 료되면 테스트 결과 페이지도 'Workspace' 패널에 자동으로 나타납니다. 만약 다른 테스트 로그를 확인하고 싶다면 해당 로그를 더블 클릭하거나, 마우스 우클릭 후 'Open' 메뉴를 선택하면 됩니다.



이전에	진행했던	테스트에	대한	결과	로그는	다음과	같습니다.
-----	------	------	----	----	-----	-----	-------

og Items	TestLog				X
Keyword Test Log [Test1]	😆 🖉 Error 🛕 🕼 Warning 🜵 🖉 Message 🛛 🖉 Event	Search		10	0
	Type Message	Time 🛆	Priority	Has	Link
	The device with the name "Nexus 7" and index 1 has been made curren	t. 12:54:58	Normal		
	<ul> <li>In the Android application "Orders" has started.</li> </ul>	12:55:05	Normal		
	The listview item 5 was touched.	12:55:09	Normal		
	The button was touched.	12:55:14	Normal		
	The text 'Mark Twain' was entered in the text editor.	12:55:18	Normal		
	🛛 💿 The button was touched.	12:55:26	Normal		
	The property checkpoint passed (the mText equals "Mark Twain").	12:55:28	Normal		
	Expected Image	Actual Image		Zoom:	-6
	Expected Image	Actual Image		Zoom:	-0
	Expected Image	Actual Image		Zoom:	
	Expected Image	Actual Image	2 💽 :	200m: *	
nformation	Expected Image	Actual Image		Zoom:	-0
nformation 3 Errors: 0	Expected Image	Actual Image	e in the	Zoom: *	
nformation Terrors: 0 Warnings: 0	Expected Image	Actual Image	Deer	Zoom:	
information Errors: 0 Warnings: 0 tart Time: 12:54 4/9/2014	Expected Image	Actual Image	Dee	zoom:	
nformation 3 Errors: 0 1 Warnings: 0 tart Time: 12:54 4/9/2014 nd Time: 12:55 4/9/2014	Expected Image	Actual Image	Deer	zoom: *	
Information Errors: 0 Warnings: 0 tart Time: 12:54 4/9/2014 ind Time: 12:55 4/9/2014 un Time: 0:00:29	Expected Image	Actual Image	Deer	zoom: *	

테스트 로그 화면은 하나의 테스트 실행 결과에 대해서 보여줍니다. 화면에서 왼쪽 부분은 테스트 결과를 트리 구조로 보여 지는데 각 노드들을 선택하면 해당되는 결과를 볼 수 있습니다. 예를 들면 본 문서에서는 하나의 테스트만 실행을 했기 때문에 트리 구조에는 오직 하나의 테스트 결과(노드) 만 나타납니다. 그 노드를 통해 테스트 결과가 성공 했는지 아니면 실패 했는지 알 수 있습니다.

테스트 로그는 결과에 따라 'Error', 'Warning' 등의 항목들로 필터링 해서 확인할 수 있습니다. 메 시지 타입은 왼쪽 아이콘을 보면 알 수 있습니다. 화면 상단에 있는 체크 박스를 통해서 사용자는 원하는 메시지들을 필터링 할 수 있습니다. 각각의 메시지들은 실행 시간 순서에 따라 나타납니다. 사용자는 그것을 타임 칼럼을 통해 확인할 수 있습니다.

TestComplete은 테스트 메시지를 통해서 이미지나 관련된 텍스트를 보여주기도 합니다. 테스트 메 시지의 상세 정보는 해당 메시지를 클릭하기만 하고 화면 하단의 'Additional Information'과 'Picture' 탭을 선택 하면 됩니다.

'Picture' 패널은 'Expected Image'와 'Actual Image'로 나뉘어 져서 나타납니다. 'Expected Image' 는 레코딩 시에 저장된 이미지이고, 'Actual Image'는 테스트가 실행되면서 저장된 이미지입니다. 테 스트 로그에는 위와 같이 두 개의 이미지를 비교해서 다른 점을 찾아낼 수 있지만, 'Picture' 패널에 서 제공되는 'View Comparison Result' 버튼을 통해서도 쉽게 비교할 수 있습니다. 좀 더 자세한 사 항은 TestComplete의 Help 메뉴의 'Test Visualizer' 섹션을 참조하세요.

'Call Stack' 탭에서는 테스트가 실행된 로그에 대해서 계층별로 보여줍니다.

'Performance Counters' 탭에서는 테스트 실행 중에 모니터링 된 성능 카운터에 대한 값이 그래프 형태로 표시됩니다.

테스트 스크립트의 각 오퍼레이션들을 자세히 보기 위해서는 해당 오퍼레이션을 더블클릭 하면 됩니다. 그럼 TestComplete은 선택된 오퍼레이션과 관련된 창이 발생합니다. 예를 들면 만약 로그의 메뉴 항목의 'Orders | Edit order' 메시지를 클릭하면 관련된 정보가 활성화 될 것입니다.



테스트 로그 및 결과 패널에 대한 자세한 정보는 TestComplete의 Help 메뉴의 'About Test Log' 섹션을 참조하시기 바랍니다.

Note: 본 문서에서 설명하고 있는 로그들을 대표적으로 사용되는 키워드 테스트 스크립트입니 다. 다른 타입의 테스트는 로그 형태나 구조가 조금 다를 수 있습니다. 다른 형태의 로 그들에 대해서 좀 더 자세히 알고 싶다면 다른 섹션을 참조하시거나 궁금하신 패널을 선택한 후 'F1' 버튼을 이용하시기 바랍니다.



#### 에러의 해결

어쩌면 테스트를 실행 했을 때 문제가 발생할 수도 있습니다. 이번 섹션에서는 문제가 발생할 수 있는 몇 가지 가능성에 대해서 말씀 드리겠습니다. 예를 들어, 개발자가 테스트 어플리케이션을 업 데이트 해서 몇몇 오브젝트의 이름이 변경되었다면 오브젝트의 인식에 문제가 발생하는 경우가 있 을 수 있고, 테스트 목적과 다른 어플리케이션이 테스트 대상이 되는 어플리케이션과 겹쳐져서 그로 인해 테스트 실행에 문제가 생기는 등의 경우가 발생할 수 있습니다.

대표적인 원인 중에 하나는 레코딩 할 때의 환경과 테스트를 실행할 때의 환경이 달라서 발생 합 니다. 이런 문제를 피하기 위해서는 테스트를 실행하는 초기 환경과 레코딩 할 때의 환경을 확실히 동일하게 만들어 줘야 합니다. 예를 들면, 만약 레코딩 시에 테스트 어플리케이션이 실행되고 있다 면 테스트를 실행할 때도 테스트 어플리케이션은 실행되고 있어야 합니다. 만약 레코딩 할 때 테스 트 대상이 되는 어플리케이션의 두 번째 탭이 선택되어 있다면, 테스트를 실행할 때 역시 해당 어플 리케이션의 두 번째 탭이 선택되어 있어야 합니다.

테스트 실행 중에 문제가 발생해서 결과가 Fail 처리가 되었더라도 그것이 환경 설정과 관련된 문 제라면 이를 수정한 뒤 다시 테스트를 실행 했을 때 그 결과가 패스로 바뀔 수도 있습니다.

이와 관련된 좀 더 자세한 사항은 TestComplete의 Help 메뉴의 'Handling Playback Errors' 섹션을 참조하세요.

## 7. 멀티 디바이스의 테스트

하나의 모바일 단말기를 테스트 해서 성공적인 결과를 얻었다면, 사용자는 테스트 스크립트를 약 간 수정해서 여러 대의 단말기를 테스트 할 수 있습니다. 이 방법은 하나의 어플리케이션이 여러 종 류의 단말기(스마트폰, 테블릿PC 등)에서 정상적으로 구동이 되는지 확인할 수 있는 테스트입니다.

여러 대의 단말기를 테스트 방법은 하나의 오퍼레이션만 수정하면 됩니다.

① 키워드 테스트 스크립트를 수정할 수 있는 화면을 엽니다.

② 화면 좌측의 'Mobile' 카테고리에서 'Device Loop' 오퍼레이션을 드래그 해서 테스트 스크립 트 영역의 최상단인 테스트 시작 위치에 놓습니다. 그럼 'Device Loop' 오퍼레이션의 상세 설정을 위한 마법사 창이 발생하는데, 아래와 같은 화면에 서 'Iterate Through All Connected Devices'를 선택하고 'Finish' 버튼을 눌러서 설정을 종료합니다.

C Device Loop	X
Select Loop Mode Specify the mobile device loop mode.	
Iterate Through All Connected Devices	
Iterates through all devices that are connected to the computer.	
Iterate Through Specific Devices	
Select some of the connected devices for iteration.	
h	
< Back Next > Finish	Cancel Help

- ③ 그리고 나서 테스트 스크립트에 있는 'Select Device' 오퍼레이션은 삭제하거나 'disable' 처리 합니다. 왜냐하면 'Select Device' 오퍼레이션의 역할을 'Device Loop' 오퍼레이션이 대신 하기 때문에 더 이상 필요가 없어져서 삭제해야 합니다.
- ④ 다음엔 나머지 모든 오퍼레이션을 선택한 뒤 하위로 이동 버튼(→)을 클릭합니다. 그럼 모든 테스트가 각각의 루프를 통해서 이루어지게 됩니다.
- 아래 캡쳐 사진이 'Device Loop' 오퍼레이션을 적용한 화면입니다.

Item	Operation	Value	Description
🖃 🍯 Device Loop	All connected devices		Iterates through t
- 🚯 Run TestedApp	Orders		Runs the "Orders"
🖻 👘 Device			
🗄 衝 Process_example_orders			
ListView_listView1	TouchItem	5	Simulates a single
Button_Edit	TouchButton		Performs a single t
- B EditText_cust_name	SetText	"Mark Twain"	Enters text in the
Button_Ok	TouchButton		Performs a single t
Roperty Checkpoint		Aliases, Device, Pro	Checks whether t

⑤ 위 단계까지 수정된 부분을 메뉴 'File | Save'를 통해서 저장합니다.





다른 단말기도 테스트 PC와 연결이 되고 테스트를 하기 위한 준비를 하시기 바랍니다. 준비가 다 되었다면 테스트를 실행하고 결과를 확인해 보도록 하겠습니다. TestComplete은 몇 번에 걸쳐서 테 스트를 반복할 것입니다. 테스트가 반복해서 진행되는 동안 각각의 단말기들이 번갈아 가며 테스트 가 될 것입니다. 테스트 결과를 보면 어떤 단말기가 테스트 수행 됐고 결과가 어땠는지를 확인할 수 있습니다.

8	🕼 Error 🛕 🕼 Warning 🤣 🕼 Message 🥝 🕼 Event 💆 3	Search		<b>ب</b> م
Туре	Message	Time 🛆	Priority	Has Link
-4	The device with the name "VirtualBox" and index 1 has been made current.	14:41:29	Normal	
- 0	The Android application "Orders" has started.	14:41:35	Normal	
-	The listview item 5 was touched.	14:41:39	Normal	
- 6	The button was touched.	14:41:41	Normal	
- 6	The text 'Mark Twain' was entered in the text editor.	14:41:43	Normal	<b>1</b>
- 6	The button was touched.	14:41:46	Normal	<b></b>
- 8	The property checkpoint passed (the mText equals "Mark Twain").	14:41:47	Normal	
- 4	The device with the name "Nexus 7" and index 1 has been made current.	14:41:47	Normal	
-0	The Android application "Orders" has started.	14:41:54	Normal	
- 6	The listview item 5 was touched.	14:41:59	Normal	
- 6	The button was touched.	14:42:03	Normal	<b></b>
- 6	The text 'Mark Twain' was entered in the text editor.	14:42:08	Normal	
- 6	The button was touched.	14:42:15	Normal	
	The property checkpoint passed (the mText equals "Mark Twain").	14:47:16	Normal	

# iOS 어플리케이션 테스트

이번 섹션에서는 TestComplete에서 iOS 어플리케이션을 어떻게 테스트 하는 지를 설명해 드리겠 습니다. 본 문서에서는 모바일 테스트를 위해 환경을 설정하고, 단말기와 PC를 연결하고, 테스트 어 플리케이션을 어떻게 구성해서 테스트 실행하고 결과를 확인하는 것까지 진행 했습니다. iOS 테스트 에서도 마찬가지로 가상화를 통해서 테스트를 진행하고 몇 개의 데이터를 검증할 것입니다.

## 사전 준비

- TestComplete를 설치할 때 모바일 모듈과 'Apple iTunes'가 설치되어 있어야 합니다. 'iTunes' 가 필요한 이유는 TestComplete와 iOS 단말기를 연결시켜 주기 위한 USB 드라이버가 'ITunes' 에 포함되어 있기 때문입니다.
- iOS 6.x 혹은 7.0~7.1.1.이 탑재되어 있는 iPhone, iPad 혹은 iPod를 준비 합니다.
- 샘플 어플리케이션(Orders)을 컴파일하기 위한 Xcode의 Mac 컴퓨터, iOS SDK와 iOS 개발 라 이선스가 필요합니다.

# 테스트 어플리케이션

본 문서에서는 TestComplete에 탑재되어 있는 iOS 버전의 'Orders' 어플리케이션을 사용해서 설명 할 것입니다. 테스트 어플리케이션을 통해 구매 주문을 하고, 주문 내역을 수정하거나 삭제하는 형 태로 테스트가 진행될 것입니다.

2 9	🗐 🗍 John Smith	n's iPad •	Bun Apps * 📠
end 🌩			17/36 Mol Charges
Edit	Orders	+	Detail
Bob Feather, t, FarriyAbarr,	5 <b>F</b> 2010 12-00	ä	PRODUCTINFO
Samuel Cla	mons 99-12-12		Flamma
Charles Do 5. MyNamy, 20	dgeson 02-05-12	. (j.	Pice
John Smith 1, MyNoniy, 20	i dir. 09-05-07	35	Discount
Clare Juffer 2. FamilyAbam	2009-04-25		Quantity
Susan McL 1, MyNoney, 20	aren 05-03-12	7	Date
Steve John 1, SerenSever.	15 2009-04-04	25	Totol
			CUSTOMER INFO
			Nama
			Street
			Gáy
			Stara
			Zo
			CARD REO
			Date
		Orebra	Satings About

테스트 어플리케이션은 모바일 단말기에 본 문서에 따라 설치가 되어야 합니다.



# 1. iOS 단말기 준비

iOS에서 어플리케이션을 테스트 하기 전에 준비해야 할 부분이 있습니다.

iOS 개발자 프로그램을 통해서 iOS 단말기에 어플리케이션 권한설정 프로파일을 추가합니다.
 단말기를 USB 케이블을 통해서 TestComplete과 연결합니다.

## 테스트 단말기에 어플리케이션의 권한설정 프로파일 추가

이 단계는 'Xcode 5'에서만 필요합니다. 만약 'Xcode 6'이라면 자동으로 단말기를 검색해서 연결이 됩니다.

만약 iOS 개발자 프로그램('iOS Developer Enterprise Program'이 아님)을 가지고 있다면, 권한설정 프로파일을 추가해야 합니다. 이 권한설정을 하게 되면 단말기에 어플리케이션의 설치가 가능해 집 니다. 가장 쉬운 방법은 Mac PC에서 'Xcode'를 사용하는 것입니다.

- ① 'iOS' 단말기를 Mac에 연결하세요.
- ② 'Xcode' 메뉴에서 'Window | Organizer'으로 진입한 뒤 'Devices'를 선택하세요.
- ③ 테스트 할 단말기를 선택한 후에 'Use for Development'나 'Add to Member Center' 버튼을 클릭하세요.

000	Orga	nizer - Devices		H <sub>M</sub>
	Devices	Projects Archives		
LIBRARY  Device Logs  Screenshots  DEVICES  My Mac 10.9.1 (13842)  John Smith's IPad 7.0.4 (118554a)  Provisioning Profiles Applications Console Device Logs Screenshots		Capacity Model Serial Number ECID Identifier Software Version	John Smith's iPad 13,79 G8 iPad Air (Model A1474) 	
	Add to Member Center	Remove		

④ 개발자 프로그램 계정 옆에 있는 체크 박스를 선택한 뒤 'Choose'를 클릭합니다.



### 단말기와 TestComplete 연결

USB 케이블을 통해 iOS 단말기와 TestComplete를 연결하세요. 만약 단말기에서 PC를 신뢰를 묻는 창이 발생하면 'Trust'를 선택하세요.

또한 단말기에서 아래와 같이 설정도 변경하시기 바랍니다.

- ① 'Settings | General' 메뉴로 진입하세요.
- ② 'Auto-Lock'를 선택하세요.
- ③ 'Never'로 선택하세요.
- ④ 'Settings | Wallpapers&Brightness' 메뉴로 진입하세요.
- ⑤ 밝기 설정을 최소 상태로 설정하세요.

이 설정은 테스트 중에 스크린이 잠금으로 전환 되거나 장시간 사용으로 인해 배터리가 소모되는 것을 막기 위함입니다.

TestComplete의 오브젝트 화면을 통해 연결된 단말기가 보이는지 확인 하세요. 연결된 단말기는 'Mobile' 카테고리 밑에 'Device("your\_device\_name") 형태로 나타나야 합니다.



다음 단계에서는 샘플 어플리케이션(Orders)를 테스트 하기 위한 준비를 설명 드리겠습니다.

## 2. iOS 어플리케이션 준비

TestComplete에서 iOS 어플리케이션을 테스트 하기 위해서는 TestComplete 라이브러리를 컴파일 하고 빌드 설정을 해야 합니다.

본 문서에서는 'Xcode'에서 iOS 어플리케이션을 어떻게 설정해야 하는지를 설명합니다. 만약 어플 리케이션을 이미 컴파일 한 상태라면 TestComplete에서 바로 사용할 수 있습니다. 이와 관련된 자 세한 내용은 TestComplete의 Help 메뉴에서 'Instrumenting iOS Applications' 섹션을 참조하세요.

테스트 어플리케이션을 통한 설명을 간소화 하기 위해 테스트에 영향을 받는 부분에 대해서만 설 명하는 방식으로 진행할 것입니다. 하지만 실제 프로젝트에서는 테스트를 위한 빌드와 애플의 앱 스 토어에 배포하는 빌드를 따로 생성해야 합니다. 왜냐하면 TestComplete 라이브러리는 private APIs 를 사용하기 때문에 테스트 어플리케이션은 앱 스토어에 제출할 수가 없기 때문입니다. 본 문서에서는 'Xcode 5'에서 iOS 어플리케이션을 어떻게 준비해야 하는지를 설명 드리겠습니다. 다음 단계를 진행하기 위해 테스트 어플리케이션을 컴파일하고 아카이브(archive) 한 '.ipa' 파일이 필 요합니다. 그 파일을 통해 TestComplete은 테스트 시 어플리케이션을 단말기에 자동으로 설치하고 실행될 수 있게 합니다.

1) Mac 컴퓨터로 파일을 복사

다음의 경로로 파일을 복사하고 Mac 컴퓨터에 TestComplete를 설치합니다.

- TestComplete Samples>₩Mobile₩iOS₩Orders₩ 'Orders' 어플리케이션의 파일 위치입니다.
- TestComplete>₩Bin₩Extensions₩iOS₩libTestComplete-agent-static.a TestComplete의 라이 브러리 에이전트입니다.

만약 'Xcode 6'을 사용하고 있다면 're-sign.sh' 파일도 '<u>TestComplete>#Bin#Extensions#iOS</u> 폴더에 복사 하셔야 합니다. 이와 관련된 자세한 사항은 TestComplete의 Help 메뉴의 'Instrumenting iOS Applications in Xcode 6' 섹션을 참조하세요.

- 2) Xcode에 샘플 프로젝트 설정
  - ① 'Xcode'에서 'Orders' 프로젝트(Orders.xcodeproj)를 열기 합니다.
  - 2 Xcode의 Navigator 영역에서 'libTestComplete-agent-static.a' 라이브러리를 프로젝트 노드 (node)로 드래그 합니다.
  - ③ 타겟을 선택한 뒤 빌드 설정 페이지로 전환 시킵니다.
  - ④ 아래 사진과 같이 'Linking'에서 'Other Linker Flags'를 '-all\_load -ObjC'로 설정합니다.



3) 'IPA' 파일 생성하기

'IPA' 파일을 생성하기 위해서는 다음과 같이 진행합니다.

- ① 'Xcode' 메뉴에서 'Product | Archive' 메뉴를 선택합니다. 'Xcode'는 생성된 내역을 'Archives organizer'에 보여줄 것입니다.
- ② 'archive'를 선택한 뒤 'Distribute'를 클릭하세요.
- ③ 마법사 창에서 'Save for Ad Hoc or Enterprise Distribution'을 선택한 뒤 'Next'를 클릭하 세요.



④ 어플리케이션에서 'provisioning profile'를 선택한 뒤 'Export'를 클릭하세요.

'Xcode 5' 어플리케이션에서는 'iOS App Development' 타입의 프로파일을 선택하시기를 권고해 드립니다. 본 문서에서 진행되는 경우에는 테스트 어플리케이션을 자동으로 설치하고 실행시킬 수 있습니다. 이와 관련된 자세한 사항은 온라인 help의 'Preparing iOS application'에서 'Choosing a Provisioning Profile for Your Application' 섹션을 참조하시기 바랍니다.

만약 iOS 개발자 프로그램(iOS 개발자 엔터프라이즈 프로그램이 아님)을 가지고 있다면 이 전에 언급했던 프로파일의 구성을 통해 테스트 단말기에 포함된 권한 설정된 프로파일을 선 택해서 확인해 보시기 바랍니다.

⑤ '.ipa' 파일을 저장하세요. '.ipa' 파일을 생성한 후에 TestComplete으로 복사하시기 바랍니다.

이제 테스트 자동화를 위한 어플리케이션과 단말기가 준비 되었습니다. 다음 단계에서는 테스트케 이스를 자동화 하고 실행하는 것을 설명 드리겠습니다.

## 3. 테스트 프로젝트 생성

이제 새로운 테스트 프로젝트를 생성해 보겠습니다.

- 만약 이미 생성된 테스트 프로젝트나 스윗(suit)이 있다면 닫으시기 바랍니다. TestComplete
   의 메인 메뉴에서 'File | Close'를 선택하시면 됩니다.
- ② 그리고 TestComplete 메인 메뉴에서 'File | New | New Project' 메뉴를 클릭합니다. 그럼 새 로운 프로젝트를 생성하기 위한 마법사가 나타납니다.

③ 마법사 창에서 프로젝트 이름과 프로젝트 파일이 저장될 위치를 선택합니다. 본 문서에서는 프로젝트 이름을 'iOSOrders'로 하고 저장 위치는 기본으로 제공되는 위치로 설정하겠습니다.

Project name:	iOSOrders	
Logation:	C:\Users\johnsmith\Documents\TestComplete 10 Projects	<u>1</u> ore

④ 다음 페이지에서는 'iOS applications'를 선택합니다.

⑤ 그리고 'Add' 버튼을 클릭한 뒤 'Orders.ipa' 파일을 선택합니다. 이 항목은 Mac PC에서 TestComplete가 설치된 PC로 파일을 카피하는 단계입니다.

⑥ 'Deploy to the device on start'를 선택한 뒤 'OK'를 클릭하세요.

iOS Application Parameters		x
Specify parameters of the iOS t	tested application	
OS application archive file:		
C:\Work\jOSApps\Orders.ipa		
☑ Deploy to the de <u>v</u> ice on start		
	OK Cancel Help	



- ⑦ 'Autorun' 항목이 체크 설정되어 있는지 확인하세요. 만약 설정되어 있다면 TestComplete에 서는 레코딩이 시작되면 자동으로 모바일 단말기의 'Orders' 어플리케이션을 실행시킬 것입니 다. 만약 체크가 되어 있지 않다면 레코딩 할 때 사용자가 직접 메뉴를 통해 실행시킬 수 있 습니다.
- ⑧ 다음 페이지에서는 'Test Visualizer'을 설정할 수 있습니다. 라디오 버튼을 통해서 선택한 뒤 'Next' 버튼을 클릭합니다.
- ⑨ 마지막으로 스크립트 언어를 선택한 뒤 'Finish' 버튼을 클릭합니다.

TestComplete은 프로젝트를 생성한 뒤 이를 'Project Explorer'에 표시할 것입니다.

이제 iOS 단말기를 연결한 뒤 테스트 어플리케이션의 설치가 완료되었습니다. 'Process Explorer'의 어플리케이션 리스트에서 마우스 우클릭을 하고 'Run'을 선택하면 어플리케이션이 실행될 것입니다.

# 4. 테스트케이스 생성

## TestComplete에서 테스트케이스 생성

TestComplete에서는 두 가지 방법으로 테스트케이스를 생성할 수 있습니다.

- 직접 테스트케이스를 생성
- 테스트 레코딩 방식을 이용해서 생성

직접 테스트케이스를 생성하는 경우 사용자는 필요한 오퍼레이션이나 테스트 시나리오를 스크립 트나 키워드 테스트에 맞춰서 생성해야 합니다. 이 방법은 매우 강력하면서도 테스트케이스를 유동 적으로 사용할 수 있다는 장점이 있습니다. 하지만 그 만큼 테스트케이스를 생성하는데 많은 경험이 필요합니다.

직접 테스트케이스를 생성하는 방법의 단점은 많은 시간을 필요로 하고 많은 다른 문제들이 발생 할 수 있습니다. 예를 들면, 직접 테스트케이스를 생성하기 위해서는 테스트 어플리케이션의 클래스 네임(name)이나 오브젝트를 알아야 합니다. 이런 불편함을 해소하기 위해 TestComplete에서 테스트 레코딩만으로 편리하게 테스트케이스를 생성할 수 있는 기능을 제공합니다. 테스트시나리오를 한번 만 레코딩 하면 TestComplete에서 테스트 시나리오 뿐 만 아니라 스크립트 라인이나 테스트 오브젝 트들도 자동으로서 인지해서 키워드 테스트 스크립트로 자동으로 변환되어 생성됩니다. 이렇게 진행 되는 형태를 '테스트 레코딩(recording a test)'이라고 합니다. 이런 방식은 사용자의 역량이 많지 않 아도 편리하게 사용할 수 있기 때문에 매우 유용한 방식입니다. 그래서 본 문서에서는 테스트 레코 딩 방식으로 가이드를 진행할 것입니다.

## 테스트케이스 레코딩

테스트 레코딩을 위해서는 아래와 같이 세 가지 단계가 필요합니다.

① 사용자는 TestComplete의 메인 메뉴나 툴 바에서 'Test | Record | Record Keyword Test' 혹

은 'Test | Record | Record Script' 메뉴를 통해서 레코딩을 시작할 수 있습니다. 아니면 시작 페이지에서 'Record a New Test' 버튼만 클릭해도 레코딩을 시작할 수 있습니다.

레코딩을 시작하게 되면 TestComplete 화면은 레코딩 모드로 전환이 됩니다. 레코딩 모드에 서는 TestComplete의 메인 화면은 자동으로 비활성화 되고 레코딩 툴 바와 모바일 스크린 화 면이 나타납니다.

사용자는 레코딩 툴 바를 통해서 레코딩과 관련된 기능들을 제어할 수 있습니다. 일시정지 (Pause)나 정지(Stop) 그리고 레코딩 타입을 변경할 수 있습니다.



모바일 스크린 창에는 연결된 단말기의 화면이 나타납니다. 이 창은 모바일 어플리케이션을 테스트 하기 위해 사용됩니다. 이 창은 'Tools | Options | Mobile | General' 메뉴를 통해서 설 정할 수 있습니다.

- ② 레코딩을 시작한 후에 매뉴얼로 테스트 시나리오를 수행합니다. 예를 들어 테스트 어플리케 이션을 실행하고, 시나리오에 따른 버튼을 클릭하거나 메뉴를 선택하고 텍스트를 입력하는 등 의 행위를 합니다.
- ③ 테스트 시나리오 수행을 완료한 후에 툴 바에 있는 종료 버튼(■)을 클릭해서 레코딩을 종료 합니다.

레코딩과 관련된 자세한 사항은 TestComplete의 Help 메뉴에서 'Recording in TestComplete' 항목 을 참조하시기 바랍니다.

## iOS 'Orders' 어플리케이션의 테스트 계획

'Orders' 샘플 어플리케이션은 주문을 할 수 있는 기능이 있습니다. 그러므로 테스트를 하기 위해 본 문서에서는 다음의 사항들에 맞게 주문 리스트를 수정하고 이에 대한 검증을 진행하려고 합니다.

- 테스트 목적: 테스트 어플리케이션(Orders) 실행 후 'Edit Order' 창에서 기존 항목에 대해 수 정이 되는지, 수정된 사항에 대해 정상적으로 변경이 되었는지 확인합니다.
- 테스트 시나리오: 주문 항목의 수정을 통해 Order 리스트가 변경되었는지 검증합니다. 본 문서 에서는 간단한 진행을 위해 하나의 항목에 대해서 하나의 속성이 변경되었는지 확인 합니다.
- 테스트 결과 확인: 테스트 시나리오에 따라 변경된 사항이 정상적으로 적용 되었다면 Order 리스트에 해당 사항이 반영될 것입니다. 이 부분의 검증을 위해 기대 값과 실제 값을 비교할 것입니다. 그렇게 하기 위해선 두 개의 값을 비교하는 오퍼레이션을 삽입할 것입니다. 해당 오 퍼레이션은 테스트 로그에서 결과를 비교하고 우리는 그 값이 성공인지 실패인지 확인할 수 있습니다.





이와 관련해서 좀 더 자세한 사항은 TestComplete의 Help 메뉴에서 'Planning Tests' 항목을 참조 하시기 바랍니다.

## 'Orders' 어플리케이션의 테스트 레코딩

시작하기 전에 두 가지를 먼저 확인 하겠습니다.

- 테스트 레코딩 중에 본 문서를 확인하는 등의 행위를 하지 마세요. 레코딩 엔진은 사용자 행 위를 모두 레코딩 하기 때문에 문서를 확인하는 등의 행위까지도 레코딩에 포함됩니다. 본 문 서를 보고자 하면 프린트를 해서 확인 하거나, 다중 모니터라면 다른 화면에 본 문서를 미리 열어놓고 확인하시기 바랍니다.
- 테스트 레코딩은 오직 테스트 어플리케이션을 위해서만 진행이 되어야 합니다. 만약 다른 어 플리케이션과 전환되는 상황이 발생하면 나중에 레코딩 된 오퍼레이션이 오동작할 수도 있습 니다. 그래서 레코딩을 하는 중에는 오직 테스트 어플리케이션만 띄어 놓으시기 바랍니다. 레 코딩 중에는 다른 단말기로 전환되지 않게, 테스트 어플리케이션이 최소화 되지 않게, 다른 어 플리케이션이 실행되지 않게, 'Home' 버튼을 눌러서 어플리케이션이 닫쳐지지 않게 해야 합니 다.

이제 'Orders' 어플리케이션을 키워드 테스트 레코딩을 시작해 보겠습니다.

- 레코딩을 시작하기 전에 모바일 스크린 창을 띄우세요. TestComplete은 오직 모바일 스크린 에서만 수행되는 것들만 레코딩이 되고 다른 창이나 단말기에서 직접 실행되는 것들에 대해서 는 레코딩이 되지 않습니다.
  - 툴 바에서 'Show Mobile Screen' 버튼을 클릭 하세요
- 2) TestComplete 화면을 보면, 새로운 프로젝트가 생성될 때 자동으로 비어있는 키워드 테스트 케이스(Test1)가 생성됩니다. 키워드를 기반으로 한 테스트케이스는 이 항목에 생성될 것입니 다.


미리 생성되어 있는 테스트케이스(Test1)를 확인하기 위해서는,

- 'Project Explorer' 패널로 이동
- 'KeywordTests' 노드를 확장(+) 선택
- 'Test1' 노드를 더블 클릭



3) 툴 바의 'Append To Test' 버튼(≪)을 눌러서 테스트를 시작합니다. 그러면 자동으로 레코딩 툴 바가 나타날 것입니다. 만약 본 문서와 테스트 해야 하는 화면이 겹쳐지면 본 문서를 축소 하거나 이동 시켜서 화면이 겹치지 않도록 하시기 바랍니다.

레코딩 툴 바의 기본 화면은 아래와 같습니다.



확장 버튼(\*\*\*)을 클릭해서 아래와 같이 추가 메뉴들을 볼 수 있습니다.





4) 레코딩을 시작하게 되면 모바일 단말기에 자동으로 'Orders' 어플리케이션도 실행될 것입니 다. 이 기능은 프로젝트를 생성하는 단계에서 'Deploy to the device on start' 메뉴를 설정하 거나 'Autorun' 옵션을 체크 설정했기 때문입니다. 만약 자동으로 실행되는 기능이 해제되었다 면 아래에 나타나 있는 메뉴를 통해 직접 실행할 수도 있습니다.



TestComplete은 테스트 어플리케이션이 시작되는 것도 레코딩 합니다. 이는 나중에 레코딩 이 종료되고 스크립트가 자동 생성되면 확인할 수 있습니다.

5) 모바일 단말기에 'Orders' 어플리케이션이 설치되고 실행될 때 까지 기다립니다. 그리고 나면 모바일 스크린에 어플리케이션의 초기 화면이 나타날 것입니다.

		1			247
	John Smith's iPad *	Run Ap	ps * 📠		
Fed ?	Orders	4	17:25	Detail	Not Charging
Bob Feat	ther 1010-12-03	×	constitute state in the same	1.1070300	
Samuel C	Clemens 2009-12-12	- 25	Name		
Charles E	Dodgeson 2009-05-12	2	Price		
John Smi 1, MyMoney,	ith Jr 2009-05-07	2	Discount		
Clare Jef	ferson m. 2009-04-05	$\rightarrow$	Guantity		
Susan Mo 1. MyMoney.	cLaren 2008-05-12	- 10	Date		
Steve Jol 1, ScreenSav	hns w, 2008-04-01	ξ	Total		
			CUSTOMER INFO		
			Name		
			Street		
			Cay		
			State		
			Zip		
			CARD INFO		
			Date		
		1	£ 1	0	

만약 본 문서와 모바일 스크린이 겹쳐진다면 본 문서의 사이즈를 변경하거나 다른 곳으로 이동 시켜서 테스트 화면이 겹치지 않도록 하시기 바랍니다. 본 문서 파일 때문에 레코딩이 정상적으로 진행되지 않을 수 있습니다.



6) 모바일 스크린에서 'Edit' 버튼을 클릭하세요. 그럼 어플리케이션 화면이 수정 모드로 전환될 것입니다.

	John Smith's Pac	* Kun Ap	ips * 💼		
d 🕈 Done	Orders	+	17:19	Not Charg	ing I
) B	ob Feather FamilyAlburn, 2010-12-03	5	PROCECT INFO		
S 2	amuel Clemens MyMoney, 2009-12-12	>	Name		
0	harles Dodgeson MyMonex 2009-05-12	0	Price		
) Ji	ohn Smith Jr MyMoney: 2009-05-07	2	Discount		
) C	lare Jefferson FamilyAlburn, 2009-04-05	2	Quantity		
) S	usan McLaren MyMoney, 2008-05-12	>	Date		
) S	teve Johns ScreenSaver, 2008-04-04	2	Total		
			CUSTOMER INFO		
			Name		
			Street		
			City		
			State		
			Zip		
			CARD INFO		
			Date		
		1	6 E		
		Ordern	Settings Ab	P	







7) 'Samuel Clemens' 리스트를 클릭하세요. 그럼 수정 가능한 Order' 패널이 발생할 것입니다.

9 6 6	John Smith's iPad	· Run Ac	ips * 🛃		
nd 🖓			17:26		Not Charging
Dona	Ordere	2012		Order	Case
DOUE	orders			Under	Olive
Bob 1, Farr	Feather HyAlburn, 2010-12-03	8	PRODUCT INFO		
Sam	uel Clemens Joney, 2009-12-12		MyMoney		Product ()
Char	fles Dodgeson Anney, 2009-05-12	>	Price		1065
John	n Smith Jr Joney, 2009-05-07	×	Discount		094
Clare	e Jefferson ilyAlbum, 2009-04-05	8	PRIME		
Susa	an McLaren Anney, 2008-05-12	3			2
Stev	e Johns enSave: 2008-04-04	<b>2</b> 2			
			ORDER DATE		
			2009-12-12		Date 3
			CUSTOMER INFO		
			Samuel Clemens		
			3, Garden st.		
			Hillsberry, UT		
			US		
			CARD INFO		
			123456789012		
			2012-03-02		Date :
			Visa	MC	AExpress

8) 고객 이름을 변경하세요.

'CUSTOMET INFO' 섹션에서 기존의 이름을 삭제하고, 'Mark Twain'으로 입력한 뒤 'Enter' 버튼을 클릭하세요. 모바일 스크린에서 테스트 PC의 키보드를 통해 입력하실 수 있습니다.

9) 'Save' 버튼을 클릭하세요. 아래와 같이 팝업 창이 발생할 것입니다.

10) 변경 팝업 창에서 'Yes'를 클릭하세요.

11) 만약 iPhone나 iPod 터치 단말기를 통해 레코딩 했다면 네비게이션 바에서 'Orders' 버튼을 터치 해야 합니다. 그럼 다시 주문 리스트 패널로 이동될 것입니다. iPad 단말기는 두 개의 패 널이 표시되기 때문에 이렇게 할 필요가 없습니다.

12) 'Done' 버튼을 클릭하세요.



and the second s	n Apps * 🔝	
ę	18:50 Not Chargin	g 🗖
orders -	- Order S	ave
b Feather amiyAlbum, 2010-12-03	> PRODUCT INFO	
ark <b>Twain</b> AyMoney, 2009-12-12	> MyMoney Produc	a >
harles Dodgeson AyMoney, 2009-05-12	> Price 1	005
hn Smith Jr JyMoney, 2009-05-07	Discount	0%
are Jefferson amityAlbum, 2009-04-05	2 COLORED	
Isan McLaren JyMoney, 2008-05-12	5	2
eve Johns criterisavor, 2008-04-04	*	
	0009-12-12 Dat	
		8.5
	CUSTOMER INFO	
	Mark Twain	
	3, Garden st.	
	Hillsberry, UT	
	US	
	Ze	
	CARD INFO	
	123456789012	
	2012-03-02 Dat	į,
	Vise WO AExpress	٦
		_

주문 리스트는 다시 기본 화면으로 전환될 것입니다.

13) 이제 값을 비교하는 오퍼레이션을 추가하겠습니다. 이 오퍼레이션을 통해 어플리케이션의 고객 이름 중에 수정한 값(Mark Twain)을 확인 하겠습니다.

본 문서에서는 그것을 체크포인트라고 합니다. TestComplete에서는 여러 가지 타입의 체크 포인트를 제공하는데, 그것들 중에 검증 상황에 맞춰서 사용하시면 됩니다. 체크포인트와 관련 된 정보는 TestComplete의 Help 메뉴에서 'Checkpoints' 섹션을 참고하시면 됩니다. 여러 가지 체크포인트 타입 중에 가장 많이 사용하는 타입은 'Property checkpoint'(속성검증)입니다. 이 체크포인트 타입은 어플리케이션을 제어하는 데이터의 값을 검증하는 기능을 수행합니다. 그 러므로 본 문서에서는 이 체크포인트 타입을 사용해 보겠습니다.



• 체크포인트의 드롭다운 버튼을 클릭하고, 리스트 중에 'Create Property Checkpoint' 메뉴(%) 를 클릭하세요.



이 메뉴를 선택하면 'Property Checkpoint'와 관련된 마법사가 발생할 것입니다. 이 마법사는 속성을 검증할 수 있는 체크포인트를 설정하는데 가이드를 할 것입니다.

reate Prope	erty Checkpoint	
Select objec Choose an c	:t object whose property will be compared.	
Drag Press buttor Press Shift+	the target to point to the object the left mouse button over the target icon and drag the icon to the desired window or control. Relea n to select this window or control. <u>Show me how it works</u> . : and fix (Shift+Ctrl+A) this button and then move the mouse pointer to the desired window, control or any other object on : +Ctrl+A to select the object. <u>Show me how it works</u> .	ise the screen. Pres
<u>O</u> bject:		Highlight
Preview:		
	<back next=""> Cancel</back>	Help

 마법사의 첫 번째 페이지에서 타겟처럼 보이는 버튼()을 클릭 하고 드래그 하세요.
 마법사 창이 자동으로 최소화 될 때까지 잠시 기다렸다가 주문 리스트(orders list)가 있는 창으로 마우스를 드래그 하세요. 마우스를 드래그 하는 중에 TestComplete에서는 마우스 커 서가 지나가는 곳의 오브젝트를 인식해서 붉은 색 박스로 표시됩니다.



'Mark Twain' 리스트에 마우스 커서를 위치한 뒤 붉은 박스가 발생한 상태에서 드래그 하 던 마우스 버튼을 뗍니다.

1 22 1	John Smith S iPa	d · Kun A	pb? . TU		
96 P			15:58		Not Charging
Edit	Orders	+		Detail	
Bob Feat	ther um. 2010-12-03	5	READ AT MEA		
Mark Twa 2. MyMoney.	ain 2009-12-12	- 3E	Name		
Charles [ . MyMoney.	Dodgeson 2009-05-12	5	Price		
John Sm I. MyMonay,	ith Jr 2009-05-07	2	Discount		
Clare Jef 2. FamilyAlbu	ferson um. 2009-04-05	2	Quantity		
Susan M	cLaren 2008-05-12	30	Date		
Steve Jo I. ScreenSan	hns wr. 2008-04-04	>	Total		
			CUSTOMER INFO		
			Name		
			Street		
			City		
	æ		State		
	<b>6</b>		Zip		
			CARD INFO		
			Date		
		1	6	0	

드래그 하던 마우스 버튼을 떼면, 마법사 창이 다시 발생하고 선택된 오브젝트의 이미지와
 이름이 마법사 창에 아래와 같이 표시될 것입니다.



Cr	reate Property Checkpoint	
Choo	ose an object whose property will be cor	mpared.
	Press the left mouse button over the target icon and i	drag the icon to the desired window or control. Release
	the button to select this window or control. Show me	how it works.
-6	Point and fix (Shift+Ctrl+A)	
9	<ul> <li>Press this outton and then move the mouse pointer to screen. Press Shift+Ctrl+A to select the object. Show</li> </ul>	me desired window, control or any other object on me how it works.
<u>O</u> bje	ect:	
Alias	ses.Device.processOrders.window0.tableview1	Highlight
Previ	iew:	1
Previ	iew:	

'Next' 버튼을 클릭하세요.

다음 마법사 페이지에서는 오브젝트의 속성과 필드를 선택할 수 있는 리스트가 나타날 것입
 니다. 이것은 TestComplete에서 오브젝트에 포함된 속성을 제공하는 것으로 테스트 어플리
 케이션에 대한 속성과 각 필드가 정의되어 있습니다.

데이터를 검증하기 위해서 어플리케이션에 정의된 몇 개의 필드를 사용할 것입니다. 오브 젝트의 필드를 보기 위해서는 'View more members (Advanced View)' 링크를 클릭하세요.

TestComplete에는 선택된 오브젝트에 대해 세 가지 속성의 그룹으로 구성되어 있습니다. 첫 번째 그룹은 공통적으로 사용되는 속성들이 포함되어 있는 그룹입니다. 그 그룹은 'Standard' 링크에서 확인할 수 있습니다. 또 다른 그룹은 모바일을 제어할 수 있는 좀 더 구 체적인 속성들이 포함되어 있습니다. 이 그룹은 'Extended' 링크에서 확인할 수 있습니다. 마 지막 그룹은 선택된 오브젝트의 모든 속성들을 볼 수 있는 'iOS' 링크에서 확인할 수 있습니 다. 데이터를 검증하기 위해서는 'wItem' 속성을 사용할 것입니다. 해당 속성의 테이블을 통해 서 구체적인 정보를 확인할 수 있습니다.

Alarra Davias ananao	dana uta dau O table	the second se	George	0
Allases.Device.processo	rders.windowu.table	eview1.witem( , 1)	Search	20
Properties				
Basic view View more me	embers (Advanced v			
🖃 Standard				
Enabled		True		
Height		1024		
Name		TableView(1)		
Parent		(Object)		(e)
Visible		True		
Width		320		
□ Extended				
ObjectGroupIndex		1		
ObjectType	_	TableView		
wItem("", 1)	Params	Mark Twain		
wItemCount	Params	)		
wItemDetail	Params	1		
wItemSelected	Params	)		
<ul> <li>wPositionX</li> </ul>		0		
<ul> <li>wPositionY</li> </ul>		0		
wSection	Params			
wSectionCount		1		

• 리스트에서 'wItem' 속성을 찾으세요. 'Params' 버튼을 클릭하고 'Index' 필드에서 '1'을 입력 하세요. 'OK'를 클릭하고 'Next'를 선택하세요.





• 다음 마법사 페이지에서는 속성의 값을 확인할 수 있습니다. Value 박스에 나타나는 속성 값을 통해 데이터를 비교해서 검증하는 것입니다.

Specify comparison settings.	
Object:	
Aliases.Device.processOrders.window0.tableview1	
Propert <u>y</u> :	
wItem("", 1)	
Condition:	
Equals	
Value:	
Mark I wain	
U Learn more about these settings	

- 'Finish' 버튼을 클릭해서 체크포인트 설정을 마무리 합니다. 그럼 레코딩이 완료되었을 때 테스트 스크립트에 해당 체크포인트가 삽입되어 나타날 것입니다.
- 14) Stop 버튼(■)을 눌러서 레코딩을 종료하세요. TestComplete은 레코딩 된 테스트 시나리오를 키워드 기반 테스트 스크립트도 저장할 것입니다.





## 5. 레코딩 된 테스트케이스 분석

레코딩을 마친 후에 TestComplete은 키워드 테스트 스크립트를 확인하고 수정할 수 있는 창을 자 동으로 보여줍니다.

🛃 Test1 🗙			
👯 Append to Test 🕨 Run Te	est 🛛 🤣 Add V	ariable 🛭 🗇 Add Parameter	↑ ↓ ← → 🖓 👘
Item	Operation	Value	Description
Select Device		"John Smith\'s iPad",	Specifies the mobile device as
Run TestedApp	Orders		Runs the "Orders" tested appli
🖨 💰 Device			
😑 💰 processOrders			
📑 👘 🖓 navigationbar 1	TouchItem	"Edit"	Performs a touch on the 'Edit' i
tableview1	TouchItemXY	0, "Samuel Clemens", 102, 12	Simulates a touch at point (10
textfield0	SetText	"Mark Twain"	Enters the text 'Mark Twain' in
📑 🛛 🖓 navigationbar0	TouchItem	"Save"	Performs a touch on the 'Save
alertview0	TouchButton	"Yes"	Performs a touch on the 'Yes'
📑 👘 🖓 navigationbar 1	TouchItem	"Done"	Performs a touch on the 'Done
Roperty Checkpoint		Aliases.Device.processOrde	Checks whether the 'wItem(""

레코딩 된 스크립트의 위 이미지에 나타난 것과 비슷할 것입니다. 만약 어떤 다른 수행을 했다면 조금 달라졌을 수도 있겠지만, 테스트를 진행하기 위한 클릭 같은 행위들에 대해서는 모두 동일할 것입니다.

위의 테스트 스크립트에는 'Order' 어플리케이션을 레코딩하면서 수행했던 것들과 동일하게 저장 된 항목들이 있는데, 이것을 '오퍼레이션'(operations)이라고 합니다.

아래 사진에 보여지는 것은 'Test Visualizer' 패널로 레코딩 중에 수행된 행위를 TestComplete에서 캡쳐한 이미지입니다.

IV Run Test	Revenue of the concentration of the concentrat	Aliases.Device.process	Checks whether the 'wI
🛃 Test Steps 🔛 Variabl	es 🔀 Parameters		
🗎 Test Visualizer			0
Update	Update	Update	Update
			7 9 90 90 90 90 90 90 90 90 90 90 90 90 9
		and the second s	

위의 이미지들은 테스트 시나리오가 어떻게 수행되었는지 이해하는데 도움을 줍니다. TestComplete은 사용자가 진행한 테스트 수행(마우스 클릭, 텍스트 입력 등)과 동일하게 이미지가 캡쳐 됩니다. 사용자가 임의의 오퍼레이션을 선택한다면 그와 매칭되는 이미지(이미지가 있는 경우) 가 자동으로 선택되어 표시되기 때문에 쉽게 확인할 수 있습니다. 이와 관련된 자세한 사항은 TestComplete의 Help 메뉴의 'Test Visualizer' 섹션을 참조하세요.

#### SMARTBEAR

첫 번째 오퍼레이션은 'Select Device'입니다. 이 오퍼레이션은 테스트 진행 시 어떤 단말기가 수행 될지를 표시해 줍니다. 다른 오퍼레이션들은 'Select Device'를 통해 연결된 단말기에서 수행되는 것 입니다.

ect Device TestedApp ice processOrders Navigationbar1	Orders	"John Smith\'s iPad",	Specifies the mobile device as . Runs the "Orders" tested appli.
TestedApp ice processOrders Navigationbar1	Orders		Runs the "Orders" tested appli.
navigationbar 1	T		
tableview1	TouchItem TouchItemXY SetText	"Edit" 0, "Samuel Clemens", 102, 12 "Mark Twain"	Performs a touch on the 'Edit' i. Simulates a touch at point (10 Enters the text 'Mark Twain' in.
<ul> <li>navigationbar0</li> <li>alertview0</li> <li>pavigationbar1</li> </ul>	TouchItem TouchButton	"Save" "Yes" "Done"	Performs a touch on the 'Save. Performs a touch on the 'Yes' Performs a touch on the 'Done
	navigationbar0	Note     Note       Not     Note <tr< td=""><td>navigationbar0 TouchItem "Save" Salertview0 TouchButton "Yes" Navigationbar1 TouchItem "Done" perty Checkpoint Aliases.Device.processOrde</td></tr<>	navigationbar0 TouchItem "Save" Salertview0 TouchButton "Yes" Navigationbar1 TouchItem "Done" perty Checkpoint Aliases.Device.processOrde

두 번째 오퍼레이션은 'Run TestedApp'입니다. 이것은 키워드 테스트에서 어떤 어플리케이션이 실 행될지를 나타내는 오퍼레이션입니다. TestComplete에서는 레코딩 중에 툴 바를 통해서 매뉴얼로 어플리케이션을 실행하거나 혹은 레코딩 시 자동으로 실행된 어플리케이션에 대해 이 오퍼레이션으 로 기록을 해서 어떤 어플리케이션이 실행되어야 하는지를 설정합니다.

🛃 Test1 🗙			
😤 Append to Test 🕨 Run Tes	st 🛛 🛷 Add V	ariable 🛭 🗇 Add Parameter	♠↓♦♦ ₽
Item	Operation	Value	Description
Select Device		"John Smith\'s iPad",	Specifies the mobile device as
Run TestedApp	Orders		Runs the "Orders" tested appli
Device			
📑 👘 🖓 navigationbar 1	TouchItem	"Edit"	Performs a touch on the 'Edit' i
ableview1	TouchItemXY	0, "Samuel Clemens", 102, 12	Simulates a touch at point (10
📑 👘 🕞 textfield0	SetText	"Mark Twain"	Enters the text 'Mark Twain' in
📑 👘 🖓 navigationbar0	TouchItem	"Save"	Performs a touch on the 'Save
alertview0	TouchButton	"Yes"	Performs a touch on the 'Yes'
📑 👘 🖓 navigationbar 1	TouchItem	"Done"	Performs a touch on the 'Done
Property Checkpoint		Aliases.Device.processOrde	Checks whether the 'wItem(""



다음의 오퍼레이션들은 테스트 어플리케이션에서 실제로 어떤 테스트 수행을 했는 지와 관련된 것들입니다. 수정 버튼을 클릭하거나, 주문리스트를 선택하거나, 값을 변경하거나 저장하는 등의 행 위와 관련된 항목들입니다.

ē.	Test1 X			
6	Append to Test 🕨 Run Te	st   🛷 Add V	ariable 🛭 🗇 Add Parameter	╋╋╋
	Item	Operation	Value	Description
	Select Device		"John Smith\'s iPad",	Specifies the mobile device as
	Run TestedApp	Orders		Runs the "Orders" tested appli
	🖻 💰 Device			
	🖻 💰 processOrders			
	🖳 🔂 navigationbar 1	TouchItem	"Edit"	Performs a touch on the 'Edit' i
	tableview 1	TouchItemXY	0, "Samuel Clemens", 102, 12	Simulates a touch at point (10
	📲 🦷 🙀 textfield0 SetTe		"Mark Twain"	Enters the text 'Mark Twain' in
	🖓 navigationbar0	TouchItem	"Save"	Performs a touch on the 'Save
	🖓 alertview0	TouchButton	"Yes"	Performs a touch on the 'Yes'
	🖓 navigationbar 1	TouchItem	"Done"	Performs a touch on the 'Done
	Property Checkpoint		Aliases.Device.processOrde	Checks whether the 'wItem(""

이와 관련된 자세한 사항은 TestComplete의 Help 메뉴의 'Simulating User Actions on iOS Applications'와 'Working With iOS Controls'를 참조하세요.

37,63

마지막으로 테스트 레코딩 중에 삽입한 값을 비교해서 검증하는 오퍼레이션입니다.

ē.	Test1 X				
6	Append to Test 🕨 Run Te	st 🛛 🛷 Add V	ariable 🛭 🗇 Add Parameter	↑ ↓ ← ⇒ 📳 🔅	
	Item	Operation	Value	Description	
	Select Device		"John Smith\'s iPad",	Specifies the mobile device as	
	Run TestedApp	Orders		Runs the "Orders" tested appli	
	🖻 💰 Device				
	🖻 💰 processOrders				
	🔤 🔂 navigationbar 1	TouchItem	"Edit"	Performs a touch on the 'Edit' i	
	tableview1 TouchItemXY		0, "Samuel Clemens", 102, 12	Simulates a touch at point (10	
	textfield0 SetText		"Mark Twain"	Enters the text 'Mark Twain' in	
	🔤 🔂 navigationbar0	TouchItem	"Save"	Performs a touch on the 'Save	
		TouchButton	"Yes"	Performs a touch on the 'Yes'	
	🔤 🔂 navigationbar 1	TouchItem	"Done"	Performs a touch on the 'Done	
	Property Checkpoint		Aliases.Device.processOrde	Checks whether the 'wItem(""]	

위에서 보신 바와 같이, TestComplete은 하나의 단말기에서 수행된 오퍼레이션들에 대해서 그룹으 로 묶어서 관리합니다. 그룹 관리는 오퍼레이션들에 대한 이해를 좀 더 수월하게 해 주고, 어플리케 이션에 내재되어 있는 오브젝트의 계층에 대한 정보를 제공합니다. 테스트는 하나의 단말기에서 하나의 프로세스 시나리오에 대한 수행을 레코딩 했습니다. 그래서 두 개의 그룹 노드로 구성되어 있습니다. 'device' 노드 그룹은 테스트가 수행된 단말기에 대한 결과 물이고, 'process' 노드는 테스트 시나리오를 통해 진행된 모든 시나리오에 대한 수행 내역입니다.

Iter	n	Operation	Value	Description		
	Select Device	1	"John Smith\'s iPad",	Specifies the mobile device as .		
uping nodes	Run TestedApp	Orders		Runs the "Orders" tested appli.		
	Cevice					
	🖻 💰 processOrders					
	🖓 navigationbar 1	TouchItem	"Edit"	Performs a touch on the 'Edit' i.		
	tableview1	TouchItemXY	0, "Samuel Clemens", 102, 12	Simulates a touch at point (10.		
	textfield0	SetText	"Mark Twain"	Enters the text 'Mark Twain' in.		
	🕞 navigationbar0	TouchItem	"Save"	Performs a touch on the 'Save.		
	- 🕞 alertview0	TouchButton	"Yes"	Performs a touch on the 'Yes'		
	🕞 navigationbar 1	TouchItem	"Done"	Performs a touch on the 'Done.		
	Roperty Checkpoint		Aliases.Device.processOrde	Checks whether the 'wItem(""		

'Object Browser' 패널에 보면 테스트 프로세스에 따라 진행된 오브젝트 이름들이 다르다는 것을 볼 수도 있습니다. 예를 들면 'Object Browser'에서 보면 테스트 당시에는 'ProcessOrders;'로 명했던 것이 'Process("Orders")로 이름 지어졌다는 것을 볼 수 있습니다. 네비게이션 바 역시 테스트 시에는 'navigationbar1'이었는데, 'NavigationBar(1)로 이름 지어졌습니다.

이것은 TestComplete에서는 자동으로 레코딩 중에 오브젝트의 'custom names'을 수집하고 사용하 기 때문입니다. TestComplete은 기본으로 제공되는 이름을 이해하는데 어려울 수 있기 때문에 레코 딩 시 수집된 이름들을 매핑 시킵니다. 매핑된 이름을 사용하는 것은 테스트를 보다 쉽고 이해하기 편하게 만들어 줍니다. 이에 대한 자세한 사항은 TestComplete의 Help 메뉴의 'Name Mapping' 섹 션을 참조하세요.

# 6. 테스트케이스 실행

이제 레코딩 된 테스트케이스를 실행 해 보겠습니다.

### 테스트케이스 실행

테스트케이스를 실행하기 위해서는 'Run Test' 버튼())을 클릭합니다.

Test1 X				
Append to Test 🕨 Run Te	st 🔗 Add V	ariable 🗳 Add Parameter	<b>↑ ↓ ← →   </b> <sup>2</sup> ,	
Item	Operation	Value	Description	
Select Device		"John Smith\'s iPad",	Specifies the mobile device as	
🔂 Run TestedApp	Orders		Runs the "Orders" tested appli	
🖻 💰 Device				
🖃 💰 processOrders				
🕞 🕞 navigationbar 1	TouchItem	"Edit"	Performs a touch on the 'Edit' i	
tableview 1	TouchItemXY	0, "Samuel Clemens", 102, 12	Simulates a touch at point (10	
textfield0 SetText		"Mark Twain"	Enters the text 'Mark Twain' in	
🖳 🔂 navigationbar0	TouchItem	"Save"	Performs a touch on the 'Save	
	TouchButton	"Yes"	Performs a touch on the 'Yes'	
🕞 navigationbar 1	TouchItem	"Done"	Performs a touch on the 'Done	
Property Checkpoint		Aliases.Device.processOrde	Checks whether the 'wItem(""	

TestComplete은 'Orders' 어플리케이션을 단말기에서 실행 시키고 테스트를 수행할 것입니다.

#### 테스트 결과

테스트가 완료된 후에 TestComplete은 테스트 결과 로그를 보여 줍니다. 만약 테스트 결과가 성공 했다면 그것을 바로 확인할 수 있습니다. 다음 단계에서 테스트 결과를 좀 더 상세하게 설명 드리겠 습니다.

#### 테스트케이스 실행

- Note: 테스트가 실행되는 동안 마우스를 움직이거나 키를 입력하는 등의 행위를 해서는 안 됩 니다. 그런 행위로 인해 테스트 실행에 오류가 생길 수 있습니다.
- 테스트를 실행하면 TestComplete은 PC 화면의 구석 부근에 인디케이터를 띄울 것입니다.



인디케이터를 통해서 테스트케이스 실행과 관련된 정보를 확인할 수 있습니다.



 테스트 실행 중에 실행을 멈추고 싶으면 툴 바에 있는 Stop 버튼()을 누르거나 TestComplete의 메뉴에서 'Test | Stop' 메뉴를 선택하면 됩니다. 만약 테스트 실행 중에 일시 정지를 시키고 싶다면 Pause 버튼())을 누르면 됩니다. 일시 정지 중에는 어떤 행위를 해도 상관 없습니다. 예를 들어 일시 정지 상태에서는 테스트 로그나 변수 혹은 어떤 오브젝트가 사용되는지 등을 여러 패널들을 통해서 확인할 수 있습니다. 이와 관련된 자세한 정보는 TestComplete의 Help 메뉴의 'Debugging Tests' 섹션을 참조하세요.

테스트케이스 실행과 관련된 정보나, 프로젝트 설정 중에 테스트 실행에 영향을 주는 설정에 대해 알고 싶다면, TestComplete의 Help 메뉴의 'Running Tests' 섹션을 참조하세요.

# 7. 테스트 결과 분석

테스트가 완료된 후에 TestComplete은 모든 오퍼레이션에 대한 테스트 결과 로그를 자동으로 보 여줍니다. 이전에 레코딩 된 테스트케이스의 실행 결과는 다음과 같습니다.



테스트 결과 로그는 여러 타입(actions, events, checkpoints 등)의 메시지를 보여줍니다. 사용자는 각각의 메시지들을 툴 바에서 필터링 해서 원하는 타입만 볼 수 있습니다. 로그 메시지를 더블 클릭 하면 이와 관련된 오퍼레이션으로 자동 연결되어 보여 집니다. 이 기능은 어떤 문제가 발생해서 연 관된 오퍼레이션을 찾고자 할 때 매우 유용하게 사용됩니다. 'Picture' 패널은 'Expected Image'와 'Actual Image'로 나뉘어 져서 나타납니다. 'Expected Image' 는 레코딩 시에 저장된 이미지이고, 'Actual Image'는 테스트가 실행되면서 저장된 이미지입니다. 테 스트 로그에는 위와 같이 두 개의 이미지를 비교해서 다른 점을 찾아낼 수 있지만, 'Picture' 패널에 서 제공되는 'View Comparison Result' 버튼을 통해서도 쉽게 비교할 수 있습니다. 좀 더 자세한 사 항은 TestComplete의 Help 메뉴의 'Test Visualizer' 섹션을 참조하세요.

화면 아래의 두 개의 탭에 대해서 추가로 설명을 드리면,

- Call Stack 탭에 있는 정보는 각각의 테스트에 대한 디버깅을 할 때 매우 유용합니다. 이 화면 은 오퍼레이션을 반복적으로 호출하거나 에러를 발생시키는 것들을 보여줍니다.
- Performance Counters 탭에서는 테스트 실행 중에 모니터링 된 성능 카운터에 대한 값이 그 래프 형태로 표시됩니다. 본 문서에서는 'Performance Counters' 기능을 사용하지 않았지만 만 약 client-server 간 iOS 어플리케이션을 가지고 있다면 이 기능을 통해서 서버 매트릭을 추적 할 수 있습니다.

모든 테스트 로그는 'Project Explorer' 패널의 'Project Suite Logs | ProjectName Logs' 아래 보관 되어 언제든지 확인할 수 있습니다.



때때로 테스트 결과가 실패할 수도 있습니다. 이런 경우는 어플리케이션의 오브젝트 속성이 변경 되거나 'Name Mapping'의 속성과 매칭이 되지 않아 발생할 수 있습니다. 그래서 사용자는 테스트 가 실패하면 어떤 이유로 인한 실패인지 확인해야 합니다. 이와 관련된 자세한 사항은 TestComplete의 Help 메뉴의 'Handling Playback Error'이나 'Debugging Test' 섹션을 참조하시기 바 랍니다.

#### 8. 멀티 디바이스 설정

하나의 단말기에서 테스트가 성공적으로 진행되었다면 이 테스트케이스에서 약간의 수정만으로 여러 개의 단말기를 테스트 할 수 있습니다.

iOS 단말기를 멀티로 테스트 하기 위해서는 다음과 같은 작업이 필요합니다.

1) 테블릿 PC와 스마트폰 간의 차이가 다른 어플리케이션 레이아웃

어플리케이션의 사용자 화면 구성은 작은 화면(iPhones 혹은 iPods)을 사용하는 지, 큰 화면 (iPads)을 사용하는 지에 따라 달라질 수 있습니다. 작은 화면에서 사용되는 어플리케이션은 UI 구성 요소(패널, 페이지, 화면, 탭 등)를 레이아웃 별로 나눌 수 있습니다. 그래서 테블릿 PC와 스마트폰 간의 수행되는 것들이 달라질 수 있습니다. 그런 이유로 iPad에서 작성된 테스트케이 스는 다른 iPads에서도 사용할 수 있지만, iPhone이나 iPod touch에서는 사용할 수 없습니다. 반대로 iPhone에서 생성된 테스트케이스는 'iPod touch'에서는 사용 가능 하지만, 'iPad'에서는 사용할 수 없습니다.

그런 몇몇의 이유로 가장 쉬운 방법은 두 가지 형태(테블릿PC, 스마트폰)로 테스트케이스를 생성하는 것입니다. 좀 까다롭지만 다른 해결 방법으로는 하나의 테스트케이스가 두 가지 타 입에 적용되도록 하는 것입니다. 이 방법은 테스트 구조를 제어 가능하고 오브젝트 인식에 대 한 기준을 설정할 수 있어야 합니다.

 2) iOS의 각각 다른 버전에 따른 오브젝트 계층, 각각의 iOS 버전은 다른 각각의 오브젝트 계층
 을 가지고 있을 수 있습니다. 그래서 테스트는 다른 버전에 대한 오브젝트를 찾아낼 수가 없 습니다. 이 문제를 해결하기 위해서는 사용자는 프로젝트의 'Name Mapping'에서 매칭되지 않 는 오브젝트에 대한 속성을 확인하고 고쳐주어야 합니다.

본 문서에서 사용한 'Orders' 어플리케이션 또한 테블릿 PC이나 스마트폰 등 사용 대상에 따 라 변경되어야 합니다. 간단한 설명만으로는 단말기의 두 가지 타입에 대한 가이드를 제공하 기가 어렵습니다. 그래서 본 문서를 통한 진행을 위해서는 같은 타입의 단말기를 사용하시기 바랍니다.

또한 'Orders' 어플리케이션은 iOS6와 그 이후 버전(iOS7~8)에서 실행될 때는 오브젝트의 계 층 구조가 약간 다릅니다. iOS6 이후 버전에서는 오브젝트의 풀 네임(full name)에 'textfield0' 이 추가되었습니다. 이 오브젝트는 'Edit Order' 패널의 'Customer name' 에디터(editor)와 일치 합니다. 그래서 오브젝트의 매핑된 이름은 iOS6와 iOS7~8이 다릅니다.

• iOS6의 경우,

NameMapping.Mobile.Device.processOrders.window0.tableview0.tableviewcell6.textfield0

• iOS7~8의 경우,

NameMapping.Mobile.Device.processOrders.window0.tableview0.tableviewcell6.scrollview0.textfield0



위에서 볼 수 있듯이 'scrollview0' 오브젝트가 다릅니다. 테스트가 문제없이 실행될 수 있게 하기 위해서는 'textfield0' 오브젝트의 검색 기준을 확장시켜야 합니다.

첫 번째로 TestComplete에서 이 오브젝트를 찾을 수 있어야 합니다. 모바일 스크린을 통해 서 다음과 같이 설정 하세요.

- ① 'Orders' 어플리케이션에서 'Edit' 버튼을 누르세요.
- ② 임의의 주문 리스트를 선택하면 'Customer name' 수정창의 'Edit Order' 패널이 나타날 것입니다.
- ③ 'Name Mapping' 수정 창을 열기 위해 'Project Explorer'에서 'NameMapping' 항목을 더 블 클릭합니다. 그러면 매핑된 오브젝트들의 속성을 확인하고 수정할 수 있습니다.
- ④ 'Mapped Objects' 탭을 선택하세요.



⑤ 만약 확인 창이 발생하면 'Yes'를 클릭하세요.

⑥ iOS7~8의 트리 메뉴에서 'textfield0' 오브젝트를 'tableviewcell6'으로 드래그 하세요.

⑦ iOS7~8의 트리 메뉴의 맵핑된 오브젝트에서 'scrollview0'을 삭제하세요. 마우스 우클릭

후 메뉴에서 'Delete'를 선택하면 됩니다.

⑧ 이제 오브젝트 네임을 업데이트 하면 됩니다.

- 'Test1' 테스트 케이스를 열기 합니다.
- 고객 이름을 수정하기 위해 사용된 오퍼레이션을 더블클릭 합니다.
- 아래의 오브젝트 이름을 A에서 B로 수정합니다.
- A. Aliases.Device.processOrders.window0.tableview0.tableviewcell6.scrollview0.textfield0
- B. Aliases.Device.processOrders.window0.tableview0.tableviewcell6.textfield0

⑨ 'Finish'를 클릭합니다.



⑩ 모바일 스크린에서 'Orders' 어플리케이션은 초기 상태로 돌아갑니다. 주문과 관련된 변 경사항은 저장하지 않고 'Edit Order' 패널은 닫은 뒤 'Done' 버튼을 클릭합니다.

이제 멀티 디바이스 테스트를 하기 위한 준비를 마쳤습니다.

## 9. 멀티 디바이스 테스트

이제 하나의 테스트케이스를 가지고 여러 단말기에 테스트를 실행 해 보겠습니다.

먼저, 'Mobile' 카테고리에 'Device Loop' 오퍼레이션을 추가합니다. 추가하는 위치는 테스트를 시 작 하는 맨 위에 위치해야 합니다.

여러 대의 단말기에서 테스트를 하기 위해 'Iterate Through All Connected Devices'를 선택한 뒤 'Finish'를 클릭합니다.

Device Loon		×
Select Loop Mode		
Specify the mobile device loop mode.		
Iterate Through All Connected Devices		
Iterates through all devices that are connected to the computer.		
Iterate Through Specific Devices		
Select some of the connected devices for iteration.		
	( )	1

- 'Select Device' 오퍼레이션의 경우는 '사용 안 함'으로 설정하거나 삭제하세요. 'Select Device' 오퍼레이션은 'Device Loop' 오퍼레이션이 추가 됐기 때문에 더 이상 필요가 없습니다.
- 'Device Loop'의 아래에 있는 모든 오퍼레이션을 선택 한 뒤 하위로 이동 버튼(→)을 눌러서 'Device Loop' 오퍼레이션 밑으로 이동 시킵니다. 이제 모든 오퍼레이션들이 단말기에 따라 반복적으로 수행될 것입니다.





Item	Operation	Value	Description		
🖃 🔮 Device Loop	All connected devices		Iterates through the specified		
Run TestedApp	Orders		Runs the "Orders" tested appli		
🚊 💰 processOrders					
🔤 🔂 navigationbar 1	TouchItem	"Edit"	Performs a touch on the 'Edit' it		
🔂 tableview 1	TouchItemXY	0, "Samuel Clemens", 102, 12	Simulates a touch at point (102		
	SetText	"Mark Twain"	Enters the text 'Mark Twain' in		
🖓 🕞 🕞 🕞 🕞	TouchItem	"Save"	Performs a touch on the 'Save'		
🔤 🔂 alertview0	TouchButton	"Yes"	Performs a touch on the 'Yes' b		
🖓 navigationbar 1	TouchItem	"Done"	Performs a touch on the 'Done'		
Property Checkpoint		Aliases.Device.processOrder	Checks whether the 'wItem("",		

여기까지 진행된 테스트 스크립트는 아래의 사진과 같습니다.

• TestComplete의 메인 메뉴에서 'File | Save'를 선택해서 저장 하세요.

'iOS 단말기의 준비' 섹션을 참조해서 다른 단말기도 같이 준비하세요. 단말기의 타입은 동일 해야 하고 생성된 테스트케이스를 통해 실행될 수 있어야 합니다.

이제 테스트를 실행하고 결과를 확인 해 보겠습니다. TestComplete은 오퍼레이션들을 반복적으로 실행할 것입니다. 각각 반복 될 때 테스트 단말기들은 번갈아 가며 진행될 것입니다. 테스트 결과 역시 각각의 단말기 별로 진행 되서 나타날 것입니다.

Test Lo	9			[	
8	🛛 Error 🛕 🗹 Warning 🔄 🖉 Message 🛛 🖉 Event 🔽 🗹 Checkpo	oint Search			,₽ 🖵
Туре	Message	Time 🛆	Priority	Has P	Link
- 4	The device with the name "John Smith's iPad" and index 1 has been made current.	15:19:55	Normal		
- 4	The iOS application "Orders" has started.	15:20:01	Normal		
📀	The toolbar item 'Edit' was touched.	15:20:05	Normal		
	The 'Samuel Clemens' item in the 0 section was touched.	15:20:09	Normal		
	The text 'Mark Twain' was entered in the text editor.	15:20:13	Normal		
	The toolbar item 'Save' was touched.	15:20:15	Normal		
	The 'Yes' button was touched.	15:20:19	Normal		
	The toolbar item 'Done' was touched.	15:20:20	Normal		
	The property checkpoint passed (the wItem("", 1) equals "Mark Twain").	15:20:23	Normal		
- 4	The device with the name "Andrew's iPad" and index 1 has been made current.	15:21:00	Normal		
- 4	The iOS application "Orders" has started.	15:21:05	Normal		
	The toolbar item 'Edit' was touched.	15:21:09	Normal		
📀	The 'Samuel Clemens' item in the 0 section was touched.	15:21:13	Normal		
📀	The text 'Mark Twain' was entered in the text editor.	15:21:17	Normal		
	The toolbar item 'Save' was touched.	15:21:19	Normal		
	The 'Yes' button was touched.	15:21:23	Normal		
	The toolbar item 'Done' was touched.	15:21:24	Normal		
· · · ·	The property checkpoint passed (the wItem("", 1) equals "Mark Twain").	15:21:26	Normal		





# 온라인 지원 정보

만약 TestComplete과 관련된 질문이 있거나 사용상에 문제 혹은 도움을 필요로 한다면 지원 팀 (support team, 미국)에 연락하시면 됩니다.

http://support.smartbear.com/message/?prod=TestComplete

TestComplete 지원 팀은 e-mail을 통해서 질문에 대한 답변을 드릴 것입니다. 만약 대화식의 질문 을 원하시면 'Contact Support' 포럼(forums)을 이용하시기 바랍니다.

포럼(forums)을 통해서 질문을 할 수 있고, 답변들을 검색하거나 다른 사람들의 코멘트를 통해서 도움을 받을 수 있습니다. 자주 묻는 질문은 동영상이나 스크린 샷을 통한 설명을 볼 수 있습니다.

더 자세한 정보는 아래 웹 사이트를 참조하시기 바랍니다.

http://support.smartbear.com/

국내에서도 TestComplete에 대한 기술 지원을 받을 수 있습니다.

STA 사는 Smartbear 사와 비즈니스 파트너 관계를 맺고 있기 때문에, STA 사 내의 자동화솔루션 사업팀으로 부터 TestComplete 자동화 도구에 대한 구매 문의 및 기술지원을 받을 수 있습니다.

TestComplete의 소개자료(<u>http://www.sten.or.kr/bbs/board.php?bo\_table=testing\_tools&sca=Test+Complete</u>)와 카 페(<u>http://www.sten.or.kr/club/club\_main.php?cb\_id=cb\_autotest</u>)에서도 정보를 얻을 수 있고, e-mail을 통해 온 라인 문의(<u>stas@sta.co.kr</u>)도 하실 수 있습니다.

