

第14章 描述性编程

本章教学目标及重点



教学目标

- 了解描述性编程
- 掌握描述性编程的UFT应用
- 掌握ChildObject方法

重点

- 掌握描述性编程的UFT应用方法

本章安排



- 14.1 描述性编程概述
- 14.2 何时使用描述性编程
- 14.3 描述性编程的使用
- 14.3.1 直接描述性编程
- 14.3.2 Description描述性编程
- 14.4 ChildObject方法



- 描述性编程,通过把对象的属性及属性的值存放到脚本中,通过 UFT调用脚本中相应对象的属性及属性的值来识别被测应用程序中 的对象,而不必再依赖于UFT的对象库
- 描述性编程能提供更加灵活的测试脚本编写方法,并摆脱测试对象 库的限制,编写出更为复杂、适应力更强的测试脚本。
- 具体操作中只需要为UFT提供对象的一组属性和值,就可以识别相应的对象并对其进行操作
- 通过描述性编程的方式,可以让UFT识别具有相同属性的对象,并 对其操作



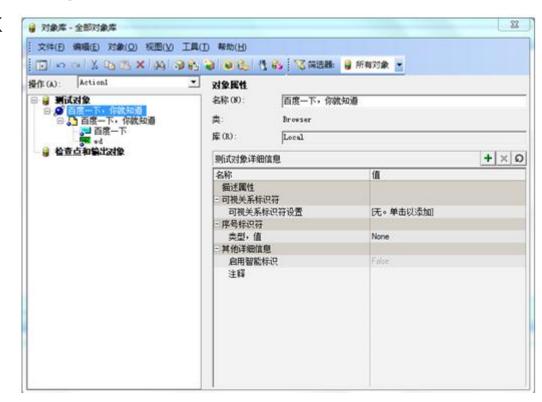
描述性编程可对未存储在对象库中的对象上执行操作,从增强脚本灵活性 方面考虑描述性编程是一种非常有用的方式。

Browser("百度一下,你就知道").Page("百度一下,你就知道")._

WebEdit("wd").Set "UFT"

Browser("百度一下, 你就知道").Page("百度一下, 你就知道")._

WebButton("百度一下").Click





示例中的问题

- 当对象库中没有添加这些对象,需要使用UFT提供的描述性编程 解决建立测试对象的问题

'代码1

Browser("micClass:=Browser")._

Page("micClass:=Page")._

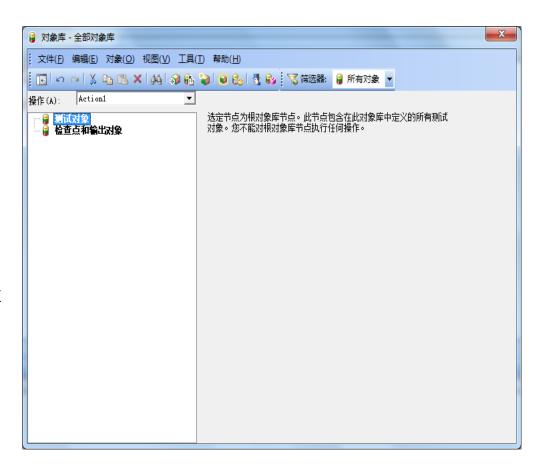
WebEdit("name:=wd").Set "UFT "

'代码2

Browser("micClass:=Browser")._

Page("micClass:=Page")._

WebButton("name:=百度一下").Click





- 描述性编程与对象库编程有以下几点区别
 - 描述性编程把需要识别的对象的属性及属性值从对象库中迁移到 了脚本里面,通过在脚本中的特殊语法格式来完成对象的识别。
 - 由于描述性编程不需要经过录制,因此描述性编程更加灵活。
 - 对于不能捕获的对象,可以采用描述性编程

14.2 何时使用描述性编程



- 那么何时使用描述性编程呢?
 - 程序的对象是动态的,如logout<<Username>>。
 - 对象库变得非常大,影响性能回放,仅限少数描述性编程代码。
 - 更改对象库的对象,但对象只可读,如公共对象库。
 - 对很对类似的对象或者相同的对象进行操作,如CheckBox。
- 测试web页面时,下面情况需要使用描述性编程
 - 在测试web页面时,对CheckBox控件进行全选或者取消全选
 - 设置所有TextBox的值为空
 - 对每一个WebRedioGroup都选择第一项

14.3 描述性编程的使用



- 直接描述性编程是在测试语句中列出所描述对象的属性和值的集合
- 直接描述性编程是通过对象属性及属性值(property:=value)来表示对象,而不是指定对象名,因此可以直接在语句中描述对象

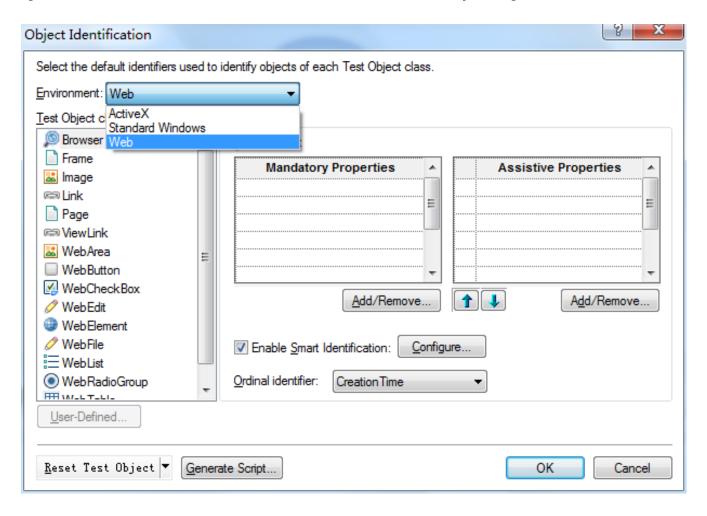
```
Eg:Browser ( "micClass:=Browser" ).page( "micClass:=Page" )._
WebEdit( "n ame:=wd" ).Set "UFT"
```

```
TestObject("PropertyName1 : =PropertyValue1" , "....." , 
"PropertyNameX : =PropertyValueX")
```

- (1) TestObject:测试对象所属类。
- (2) ropertyName: =PropertyValue:测试对象的属性及属性值。各个property: =value对之间应用逗号和引号分开。



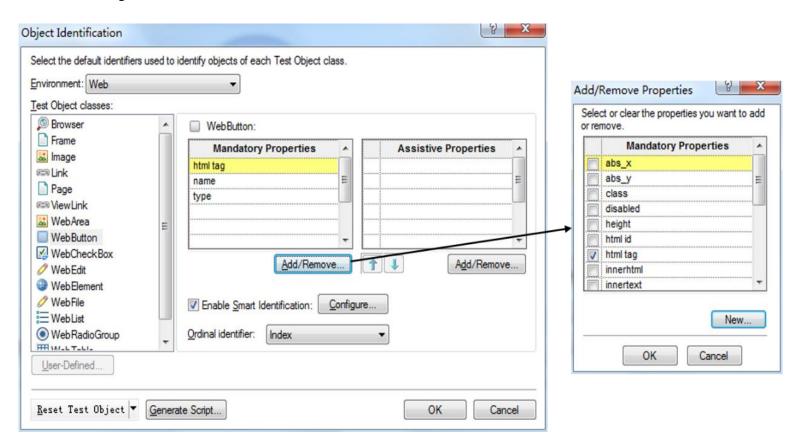
- 获取Property的方法
 - 通过Object Identification获取,选择Tools | Object Identification





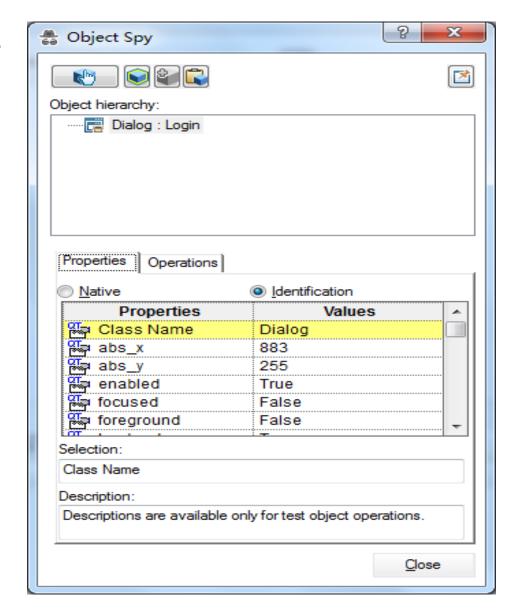


- 获取Property的方法
 - 通过Object Identification获取



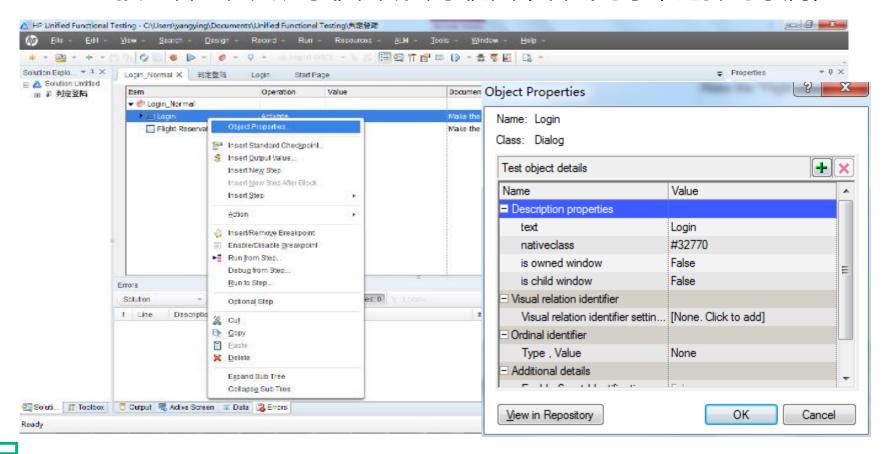


- 获取Property的方法
 - 使用Spy获取





- 获取Property的方法
 - 直接查看:在关键字视图或专家视图中右击对象,选择对象属性





后续强迫症

如果在测试对象车次结果的某一点开始使用了描述性编程,则该测试对象层次结果下的后续测试对象都需要使用描述性编程方式来描述对象。

描述性编程方式来描述对象。

"正确 — 父对象如果描述了,子对象必须描述
Browser ("micClass:=Browser").Page ("micClass:=Page")._

WebEdit ("name:=wd").Set "UFT"

"错误 — 父对象描述了,子对象没有描述

Browser ("micClass:=Browser").Page ("百度一下,你就知道")._

WebButton ("百度一下").Click

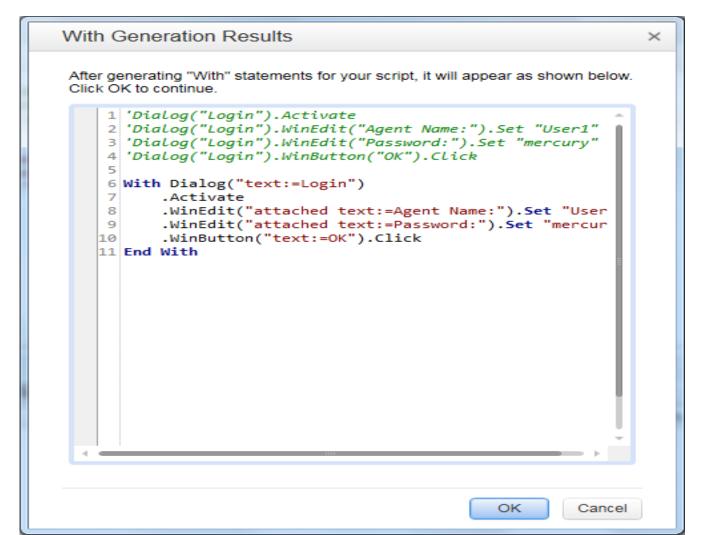






With语句:提高脚本的层次性、加快描述性编程的速度

选择: Edit | Format | Apply "With" to Script...显示如下对话框





14.3.2 Description描述性编程



- 测试对象的属性和描述都包装在Description中,用Description的实 例来代表一个测试对象的属性描述
- 使用 Description 对象可以返回包含一组 Property 对象的 Properties 集合对象
- Property 对象由属性名和值组成
- 可以在语句中指定用返回的Properties 集合代替对象名

Set ObjDescription = Description.Create ()

ObjDescripton ("PropertyName1"). Value="PropertyValue1"

ObjDescripton ("PropertyName2"). Value="PropertyValue2"



14.3.2 Description描述性编程



・ 实例:

```
Set ObjDialog = Description.Create()
          ObjDialog("text").Value="Login"
Set ObjName = Description.Create()
         ObjDialog("nativeclass").Value = "Edit"
          ObjDialog("attached text"). Value = "Agent Name"
Set ObjPwd = Description.Create()
          ObjPwd("nativeclass").Value = "Edit"
          ObjPwd("attached text"). Value = "Password"
Set ObjButton = Description.Create()
          ObjButton("nativeclass"). Value = "Button"
          ObjButton("text").Value = "OK"
With Dialog(ObjDialog)
          .Activate
          .WinEdit(ObjName).Set "Catherine"
          .WinEdit(ObjPwd).Set "Mercury"
          .WinButton(ObjButton).Click
End With
Set ObjDialog = Nothing
Set ObjName = Nothing
Set ObjPwd = Nothing
Set ObjButton = Nothing
```



14.3.2 Description描述性编程



· 直接描述性编程

更适合应用于普通脚本中,即对象库编程无法完成任务的时候,描述性编程临时增加一句,这样显得比较直观,代码数量也少。但是缺陷是无法复用

· Description描述性编程

更适合应用于基本框架的脚本中,代码虽然多,但可以复用性强于直接描述性编程

14.4 ChildObjects方法



- 使用ChildObjects方法可以检索指定父对象中的所有子对象,或者只检索符合特定描述性编程的子对象
- 要检索子对象的自己,首先需要创建一个Description对象,然后使用Description对象添加一组属性及属性值,一边子对象集合能与之进行匹配
- 使用Description对象来为ChildObjects描述参数创建编程性描述, 而不能使用property:=value语法直接将编程性描述添加到参数中
- 当完成在Description对象中"built"了描述,就可以使用下面的语法来获取与描述匹配的子对象

Set MySubSet = TestObject.ChildObjects(MyDescription)

14.4 ChildObjects方法



- 而当需要选中某个页面上所有复选框时,就需要使用ChildObjects方法来实现
 - 首先,建立一个Description对象

'创建 Description 对象

Set MyDescription = Description.Create ()

- 设置Description对象的属性

'设置 Description 对象的属性

MyDescription ("html tag"). Value = "InPUT"

MyDescription ("type"). Value = "Checkbox"

- 把Description对象作为ChildObjects的输入参数

'调用 ChildObjects 方法实现选中所有复选框

Dim counter

Set Checkboxes =Browser ("title:=学习教育").Page ("title:=学习教育").

ChildObjects (MyDescription)

Num = Checkboxes.Count

For Counter=0 to Num-1

Checkboxes (Counter).Set "ON"

Next

课堂练习



1. 何时使用描述性编程?

答:

- ✓ 程序的对象是动态的,如logout<<Username>>。
- ✓ 对象库变得非常大,影响性能回放,仅限少数描述性编程代码。
- ✓ 更改对象库的对象,但对象只可读,如公共对象库。
- ✓ 对很对类似的对象或者相同的对象进行操作,如CheckBox。

2. 通过那些方法可以查看对象属性?

答:

- ✓ 通过Object Identification获取
- ✓ 使用Spy获取
- ✓ 直接查看



课堂练习



3. 用自己的话解释什么是描述性编程的"后续强迫性"问题

答:

✓ 如果在测试对象车次结果的某一点开始使用了描述性编程,则该测试对象层次结果下的后续测试对象都需要使用描述性编程方式来描述对象。

4. ChildObjects方法的作用是什么?

答:

✓ 检索指定父对象中的所有子对象,或者只检索符合特定描述性编程的子对象。

总结



 本章主要介绍了UFT中描述性编程的内容,使得UFT能在不添加对象 库的情况下识别对象并运行脚本。描述性编程包括直接描述性编程和 Description描述性编程,并在此基础上介绍了Childobjects方法, 方便检索子对象。

