**学习大纲**

* 界面布局
* 添加活动 (Activities)
* 使用变量
* 使用流程图 (Flowcharts) 或序列 (Sequences)

1. 界面布局

* Ribbon
* Activities
* Workflow Designer
* Properties

2. FlowChart和Sequence

* Sequence适用于需要执行几个连续动作的线性过程。
* FlowChart适用于为复杂的活动建立联系。
* FlowChart和Sequence可以互相嵌套。
* 为FlowChart和Sequence命名，使程序的框架看上去更直观。

3. 变量

变量用于存储多种类型的数据。变量的另一个关键方面是它们的值可以改变，以便您可以控制循环体执行的次数。

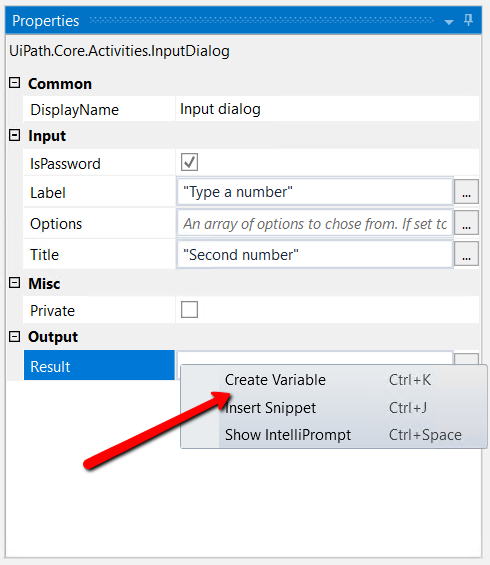
**Scope**表示变量的应用范围，比如变量仅在某一个特定的FlowChart中使用，或是能在整个项目中使用。即使在不同的Scope中使用，也需要使用不同的名称创建变量。

1) 常用变量：

* Integer：整数 (1, 2, 3, 45345)
* String：字符串，通常用引号括起
* Boolean：布尔型，True或False
* Generic：存储任何类型的数据
* Array of...：任何类型的同类数据的列，例如{1, 15, 36}; {strFirstName, StrMiddleName, strLastName}

2) 添加变量：

* 从“设计”功能区选项卡添加：
  + 在“**Design**” Ribbion的“**Variables**”组中，选择“**Create Variable**” > **[Type of Variable]**。将显示“Create Variable”窗口。
* 从右键菜单或使用快捷键Ctrl + K添加：
  + 在任何活动的“**Properties**”面板中，右键单击可以编辑的字段，然后在菜单中选择“**Create Variable**”，或按Ctrl + K。
  + 填写名称并按Enter键。这种方式创建的变量的范围是它所属的最小范围。通过“**Manage Variables**” > “**Promote to Global Scope**” 以将所有变量都提升为全局变量。



* 从变量面板添加：
  + 在Workflow Designer的下方可以调出变量面板。
  + 变量名 Name：必填；变量类型 Type：必填；变量范围 Scope：必填；变量默认值 Default：可选。
  + 在变量面板中修改变量名，将会应用到所有使用此变量的Activities或Workflow中。

 3) 删除变量：

直接在变量面板中选中并按Delete键。

* 删除所有在当前项目中**未使用的变量**：
  + 在“**Design**” Ribbion的“**Variables**”组中，选择“**Manage Variables**” **>**“**Remove Unreferenced**”。
  + 变量面板中仅包含自动化过程中使用过的变量。

4) 浏览.Net变量类型

搜索变量类型列表中不显示的类型：

* 在变量面板的变量类型下拉列表中，选择“**Browse for Types**”。
* 在“**Type Name**”字段中键入要查找的变量类型的关键字，例如Excel。

4. 判断

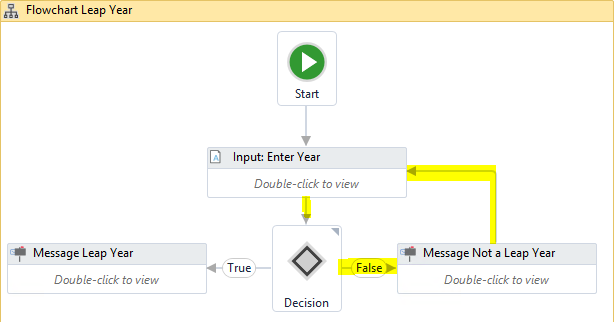
FlowChart：**Flow Decision**

Sequence：**If**

在Condition里添加判断条件，可以用And或Or连接。

5. 循环

FlowChart：直接将箭头指向之前的某个活动就可以构建循环。注意要给循环一个出口，否则就会死循环。

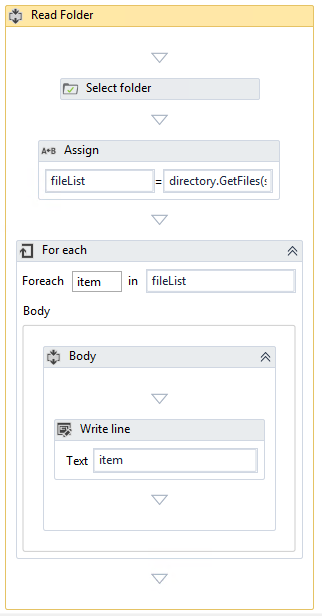


Sequence：**While，Do While，For Each**

* While：先判断条件，如果符合条件则执行循环（条件在行动前）。
* Do While：先执行循环，再判断条件，符合条件则继续下一轮循环（条件在行动后）。
* For Each：对一个集合里的每个项目执行循环。

For Each 实例：

1. Select Folder活动弹出窗口，要求用户选择一个文件夹。
2. 通过.NET代码Directory.GetFile()读取文件夹中的所有文件路径，并存入数组fileList中。
3. For Each循环将数组fileList中的每一个元素（此例中是文件路径），并显示在Output窗口中。

* 
  + For Each的属性TypeAgrument，用来选择每个项目(item)的数据类型。
  + Directory.GetFiles()是vb.NET语言，Uipath建立在.NET框架上的。

\*按下Ctrl键可以选择多个活动。

\***GenericValue的连接：**"123"+123=123123，123+"123"=246

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Generic Value 的方法** | | | | |
| 方法名称 | 数据类型 | | | |
| 字符串型(String) | 整数型(Integer) | 小数型(Float) | 布尔型(Boolean) |
| **Split** | √ | 自动转换为字符串型 | 自动转换为字符串型 | 自动转换为字符串：  "Ture"或"False" |
| **Replace** | √ |
| **Substring** | √ |
| **Length** | √ |
| **Contains** | √ |
| **Trim** | √ |
| **IndexOf** | √ |
| **ToUpper** | √ |
| **ToLower** | √ |
| **ToInt** | √ | √ | 取下限 | True=1, False=0 |
| **ToString** | √ | √ | √ | "Ture"或"False" |

6. 练习：猜数字

生成一个0-1000的随机数让用户猜测。如果猜对了则提示并停止程序；如果猜错了，提示用户猜测的数字比正确答案大了还是小了，再继续猜测。

**思路：**

新建一个Flowchart→Assign：将生成的随机数赋值给变量rNum→Input Dialog：输入猜测的数字并赋值给变量gNum，将Label属性设置为变量Hint用于显示提示→Flow Decision：判断gNum是否等于rNum→是则显示Message Box→否则添加第二个Flow Decision：gNum是否大于rNum→是则Assign：Hint赋值为“猜一个更小的数字”→否则Assign：Hint赋值为“猜一个更大的数字”→将两个Assign的箭头拖到Input Dialog处建立循环。

生成随机数：new Random().Next(0,1000)，可以直接谷歌关键词“.NET随机数”。

**\*本课使用过的活动、方法、函数等：**

* **Message Box**
* **Assign**
* **Input Dialog**
* **Write Line**
* **Excel Application Scope**：设置路径时，如果文件不存在，会自动创建该文件。
* **Write Cell**：设置工作表名称时，如果工作表不存在，会自动创建工作表。
* **Write Text File**
* **Select Folder**
* **Delay**：Duration属性设置延迟的时间，如00:00:20表示延迟20秒钟。
* **Switch**：多条件判断，类似于VBA的Select Case。默认的Case值为整数型，可以在属性TypeArgument里修改。
* **Break**：只用于For Each循环。
* **Open Application**：可以用Indicate window on screen来抓取已经打开的程序，也可以在FileName属性里填写程序的路径。
* **Attach Window**：连接到已经打开的窗口并执行一系列活动。
* **Type Into**
* **Close Application**
* Now：返回项目执行时的日期和时间 (dd/MM/yyyy hh:mm:ss)
* .Substring(0,1)：返回字符串的第一个字母。
* .ToString：将数据转化为字符串类型。
* .Subtract()：减去括号内的值。
* new Random().Next(0,1000)：生成一个0到1000的随机整数。