VARIABLES, DATA TYPES AND CONTROL FLOW

# CONTROL FLOW

CONTROL FLOW OVERVIEW:

* It is the order in which individual statements, instructions and function calls are executed or evaluated in a software project.
* 4 predefined types of workflows:
* Sequence
* Flowchart
* State Machine
* Global Exception
* Two types of concepts through which control flow is enacted:
* Sequence: the process steps flow in a clear succession. Activities in sequence are easier to read and maintain, therefore are highly recommended for simple, linear workflows.
* Flowcharts: Use flowcharts when decision points and branching are needed in order to accommodate complex scenarios, workarounds and decision mechanisms. Flowcharts can be used in variety of settings, from large jobs to small projects that you can reuse in other projects. The most important aspect of flowchart is that, unlike sequences, they present multiple branching logical operators, that enable you to create complex business processes and connect activities in multiple ways.
* The most common flow statement’s:
* If/else decision
* Loops
* Switch

## THE IF STATEMENT:

* The if statement is:
* The condition that is verified (with two potential outcomes – true or false).
* The set of actions to be executed when the condition is true (the THEN branch).
* The set of actions to be executed when the condition is false (the ELSE branch).
* Business scenarios in which the IF statement can be used:
* Checking the status of a payment (done/not done) and performing a set of operations in each case.
* Making sure that the outcome of the previous operation in the sequence is successful.
* Checking the balance of an account to endure that there is enough money to pay an invoice.
* Checking of something has happened in a system, like if an element or an image exists and performing an action based on that.
* Example: (saved it as IF Statement)

Business Scenario: If the reporting year is leap year or not?

A year is a leap year if it meets the following conditions:

* The year is evenly divisible by 4
* If the year can be evenly divided by 100, it is not a leap year, unless the year is also evenly divisible by 400.

Recap:

* If statement in a Sequence:

We used an ‘Input Dialog’ activity to get an input value from the user and store it in an Int32 variable.

We added an ‘If Sequence’ activity and defined the condition using the mod operator to check the remainder in a division: (year mod 4 = 0 and year mod 100 <> 0) or (year mod 400 = 0). If the condition is true, the value is a leap year.

* If statement in a Flowchart:

We used an ‘Input Dialog’ activity to get an input value from the user and store it in a Int32 variable.

We added ‘Flow Decision’ activity with the same condition as in the ‘If Statement’ activity above: (year mod 4 = 0 and year mod 100 <> 0) or (year mod 400 = 0).

* VB. Net Operator:

We defined the project as a sequence and used an ‘Input Dialog’ activity to get the input value from the user and store it in an Int32 variable.

We defined a String variable (message) and used it as a output of an ‘Assign’ activity, In the value field of the ‘Assign’ activity, we used the same expression from the previous example, followed by 2 pieces of text between quotation marks – the first to be assigned when the condition is true: If (year mod 4 = 0 and year mod 100 <> 100) or (year mod 400 = 0), “Leap Year”, “Not a Leap Year”).

## LOOPS:

* Loops are repetitions of a set of operations based on a given condition.
* The most important loops are:
* Do While: It executed a specific sequence while a condition is met. The condition is evaluated after each execution of the statements.
* While: It executed a specific sequence while a condition is met. The condition is evaluated after each execution of the statements. In many cases, it is interchangeable with Do While, the only difference being when the condition verification is made. But in some cases, one is preferable over the other.

Do While and While – the While and the Do While works by repeating a given sets of actions, while the specified condition is true. The only difference is the order in which the two elements are executed. The While loop checks the condition first and if it is true, it executed the set of actions in the body. Whereas the Do While loop executes the action, then checks the condition and if it is true, the it executes the activities again.

Example: In the same sequence where the If Statement had been applying the leap year condition, we have added a ‘Do While’ activity and moved all the previous activities (‘Input Dialog’ and ‘If Statement’) in its body.

As the condition of the Do While, we have specified the opposite of the condition is the ‘If Statement’ activity, as we wanted the loop to be repeated as long as the value wasn’t a leap year: not ( (year mod 4 = 0 and year mod 100 < > 0 ) or ( year mod 400 = 0) ).

In the Flowchart, it was much easier, as we simply dragged a arrow from the ‘Message Box’ activity in the false branch to the ‘Input Dialog’ activity above.

* For Each: It performs an activity or a series of activities on each element of a collection. This is very useful in data processing.

Example: We have started the project as a sequence and added the ‘Select Folder’ activity, that prompts the user to indicate a folder on the machine. The output of the activity is the full path stored in a String variable that we have created.

We have created an Array of Strings variable and used in an ‘Assign’ activity in order to get the file names using ‘Directory.GetFiles’ method:

We have added a ‘For Each’ activity to go through each object (file name) in the Array and perform the following activities:

1. Generate the new name of the file by using the ‘Replace’ and ‘Now.ToString’ methods and store it in a local String variable: filename.Replace(“.pdf”, Now.ToString(“\_yyyy\_MM\_dd”) + “.pdf”)
2. Use a ‘Log Message’ activity to print the new name.
3. Use a ‘Move File’ activity to change the names of the files in the folder.

## SWITCH:

* It is a type of control flow statement that executes a set of statements out of multiple, based on the value of a specific expression. In other words, we use it instead of an IF Statement when we need at least 3 potential courses of action. This is done through the condition, which is not Boolean like in this case of IF Statement, but multiple.