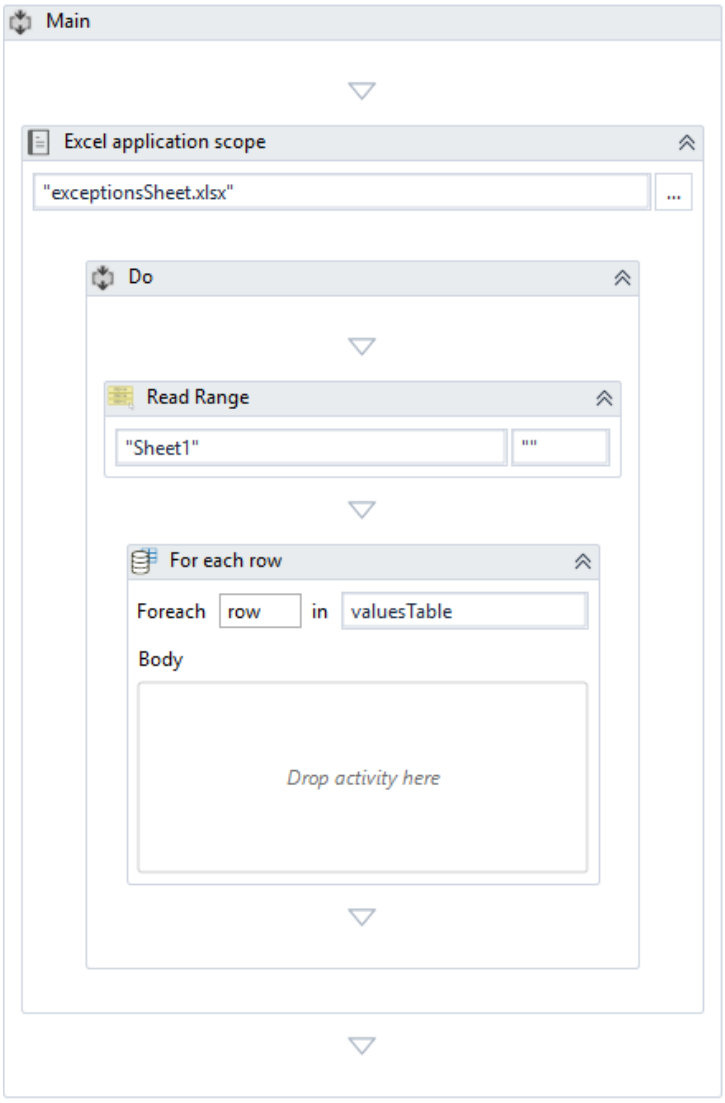
Practical Exercise - Walkthrough

Since we are going to be working with Excel, the first step will be adding an **Excel Application Scope** and reading data from the sheet.

- Find and add an **Excel Application Scope** activity to the **Main** panel.
    - Insert the path of the downloaded .xlsx file in the **WorkbookPath** property field.
- Find and add a **Read Range** activity inside the **Do** container of the **Excel Application Scope**.
    - Insert a new DataTable variable, called *valuesTable*, in the DataTable property field.
    - Make sure that the **AddHeaders** check box is selected.

Since the data is organized in a spreadsheet, a loop will be needed to iterate through each row.

- Find and add a **For Each Row** activity below the **Read Range** activity.
    - Set the **DataTable** to be looped through as *valuesTable***.**This is how the workflow should look so far:



We're going to use two Try Catch activities to check Cash In and Cash Out, and assign a variable result which holds the value we will write on that row.

- Create two Int32 variables called *cashIn* and *cashOut.*
- Create a GenericValue variable called result. Since it's a GenericValue type, it can hold either the result of subtracting two integers, or it can hold a string alerting us that one of the values was incorrect.
- Find and add a **Try Catch** activity in the **Body** section of the **For each row** activity.
    - This will catch exceptions if the Cash In is not a valid number.
- Find and add an **Assign** activity into the **Try** portion of the **Try Catch**.
    - It should assign *cint(row("Cash In"))* to *cashIn*.
        - This converts the cell value in the Cash In column of the current row to an integer and assigns it to *cashIn*.

        - If there is an exception (if the value is not a number), it will be caught in the **Catch** section.

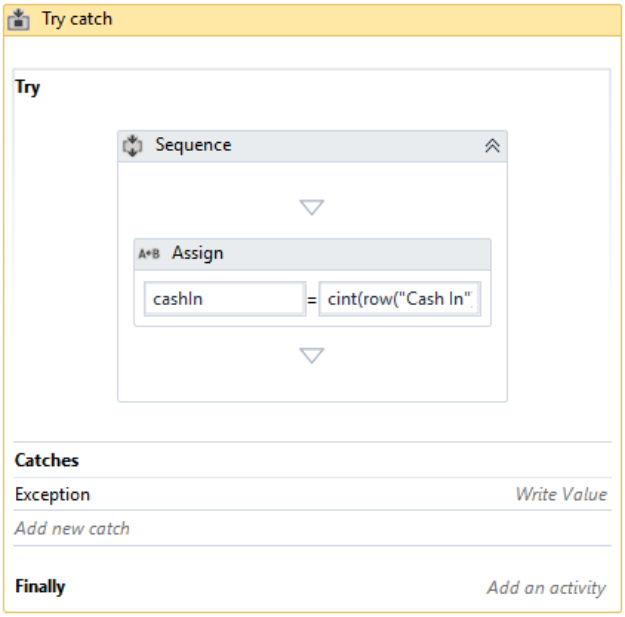
    - Click on **Add new catch** at the bottom of the activity.

        - Select **System.Exception** in the drop-down (search for it if it is not there).

        - Add an Assign activity into the exception area

        - Set the **To** to *result.*

        - Set the **Value** to “Cash In wrong”.



- Click back on the **Try** section: find and add another **Try Catch** below the **Assign** activity.

    - This one will *try* the *cash out* values and catch them differently.

- Add an **Assign** activity inside the **Try** section.

    - It should assign *cint(row("Cash Out"))* to *cashOut*.

- Find and add another **Assign** activity

    - If the robot hasn’t thrown an error by this point, both Cash In and Cash Out are valid numbers.

    - Set the **To** to *result.*

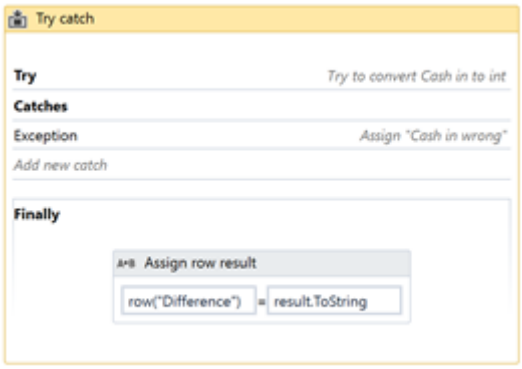
    - Set the **Value** as "*cashIn - cashOut*", the difference between the two numbers.

- For the **Catch** section of this **Try Catch**, add another **System.Exception**.
    - Add an **Assign** in the **Exception** field.
        - Set the **To** to *result.*
        - Set the **Value** as "*Cash out wrong"*.

In the **Finally** block of a **Try-Catch** activity are the steps that the robot will take regardless of whether the **Try** block or the **Catch** one was executed. This workflow is where the robot updates the DataTable with the result, which was calculated by the **Try-Catch** activities.

- Click on the **Finally** section of the outer **Try Catch** activity.

- Drag an **Assign** into the **Finally** block
    - Set the **To** field to *row("Difference").*
    - Set the **Value** to *result.ToString*.



For the finishing touch, write the processed DataTable back to the Excel file.

- Drag a **Write Range** activity after the entire **For-Each** loop
  - Set the **Data table** to *valuesTable*

Your finished workflow should look like this:


Main
Excel application scope
"Lesson 12 - practice 1 - exceptionSheet.xlsx"
Do
Read Range
"Sheet1"
For each row
Foreach row in valuesTable
Body
Body
Try catch
Try
Try to convert Cash in to int
Assign cash in
cashin
cint(row("Cash In"))
Try catch
Try
Try to convert Cash out to int
Assign cash out
cashOut
cint(row("Cash out
Assign result
result
cashin- cashOut
Catches
Exception
Assign "Cash out wrong"
Add new catch
Finally
Add an activity
Catches
Exception
Assign "Cash in wrong"
Add new catch
Finally
Assign row result
Write Range back to sheet
"Sheet1"
"A1"
valuesTable