The document explains how to split a single PDF into multiple invoices.

Author: Ankit Parwarkar
Date: 14-09-2023

Description:

To implement this, you need to identify keywords that indicate the start and end of each invoice.
Additionally, there can be keywords to indicate middle pages within an invoice if required. Condition to check for middle keywords can be added but not considered in this document.

1. Obtain the total page count of the PDF and iterate through each page using a 'For Loop.'
2. Inside the loop, extract the text of the current page using the Read PDF with OCR activity.

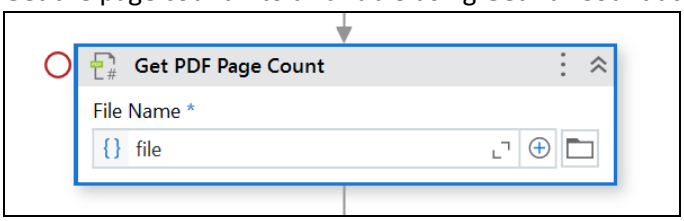
   A. Scenario: a single-page invoice:
   - Check if both the Start and End keywords are present on the page.
   - If they are, extract this page as an individual invoice.

   B. Scenario: a multiple-page invoice:
   - Check if both the Start and End keywords are present on the page. And this time, the above condition returns False.
   - Check if the Start keywords are present on the current page.
   - If they are, add the page index value to a variable.
   - In the next iteration, directly check for the End keywords.
   - If the End keywords are not present, add the current page index to the variable.
   - If the End keywords are present, add the current page index to the variable, and extract all the pages indicated by the variable as an individual invoice and reset the variable for the next iteration.

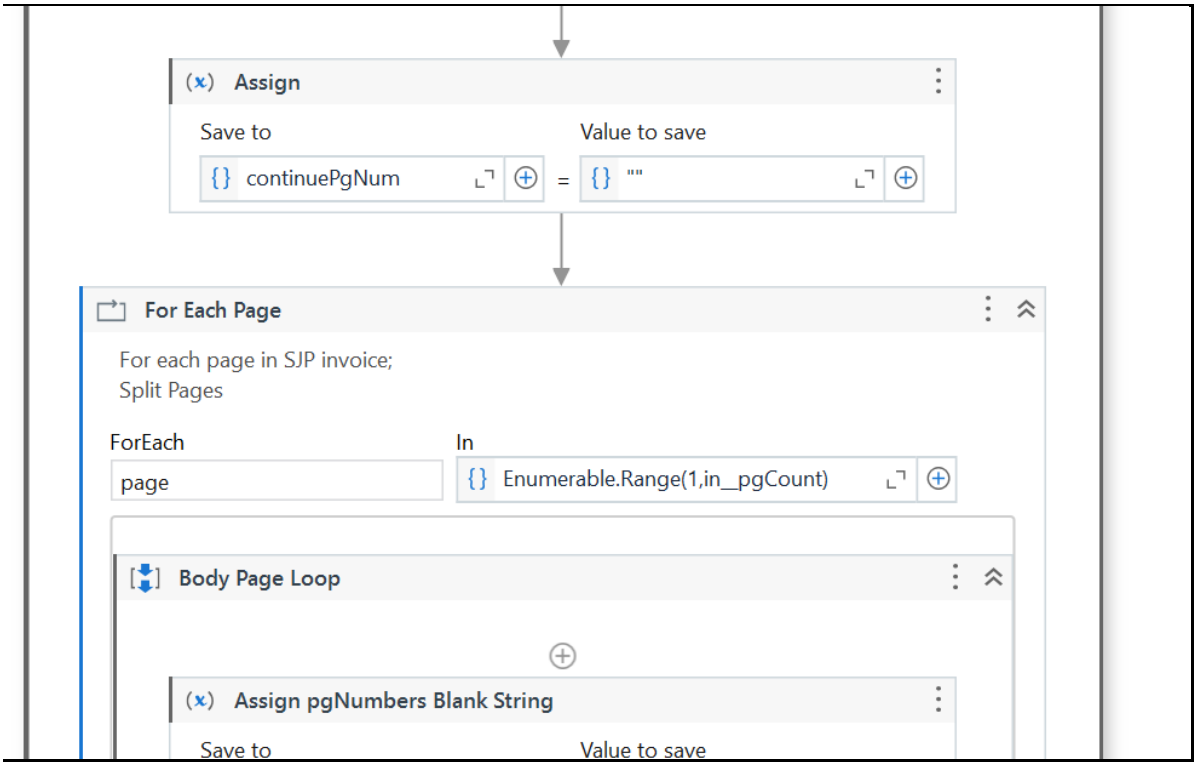
   Repeat the process for each new page.

Detailed Development Steps:

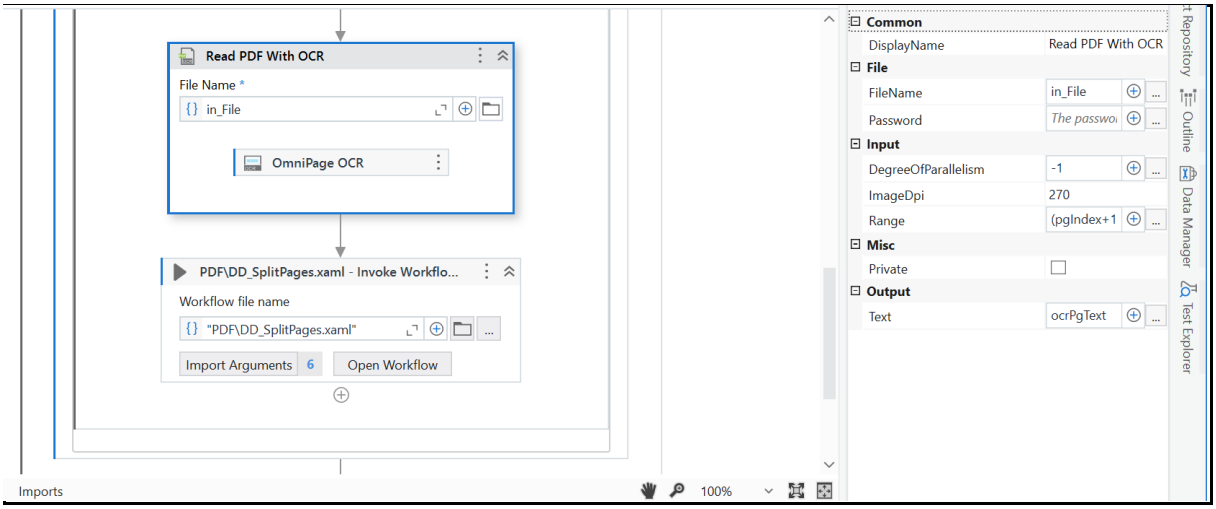
1. Use ‘Read pdf with OCR’ Activity to identify the invoice.
2. Get the page count into a variable using Get Pdf Count activity. Output Variable – ‘pgCount’.



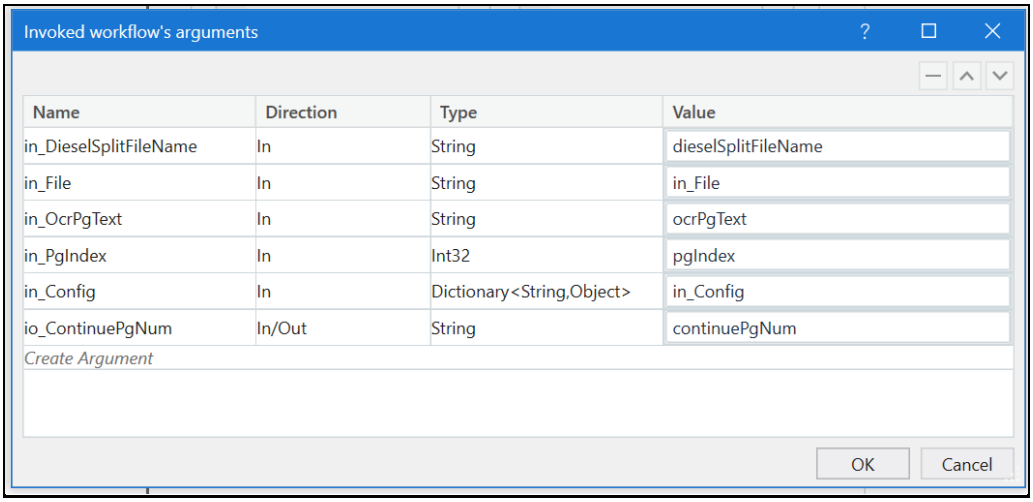
3. Use For Loop to iterate through all the pdf pages. Index variable – ‘pgIndex’.
   Before For Loop, Assign ‘continuePgNum’ variable an empty string. This variable is used to store the continuous page numbers in string.



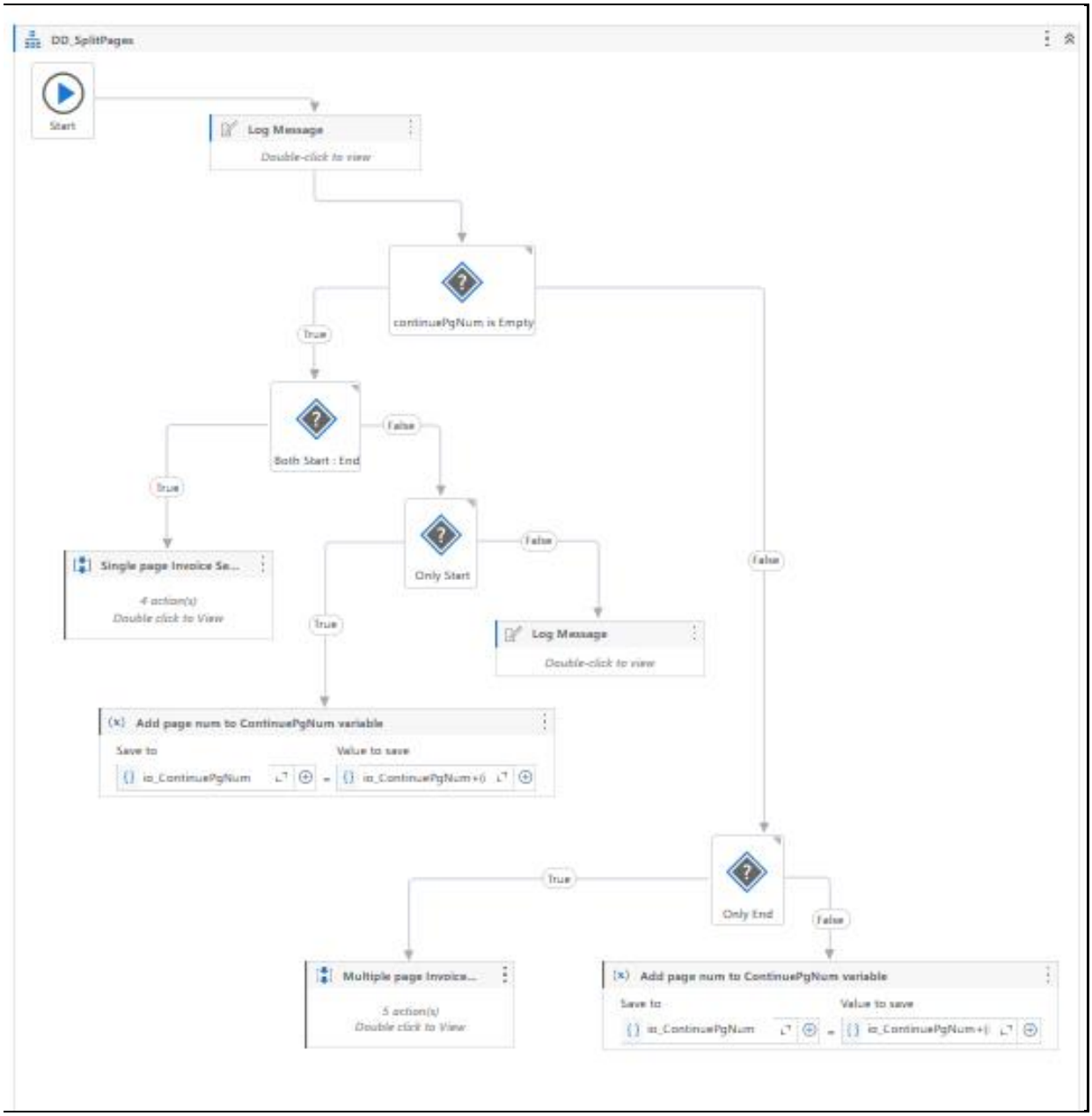
4. Inside the For Loop Body:
    a. Read pdf with OCR – Range will be ‘pgIndex+1’. Read only the current page. Output Variable – ‘ocrPgText’.
    b. Call the Split pages workflow.



    c. Arguments required in the SplitPages workflow:

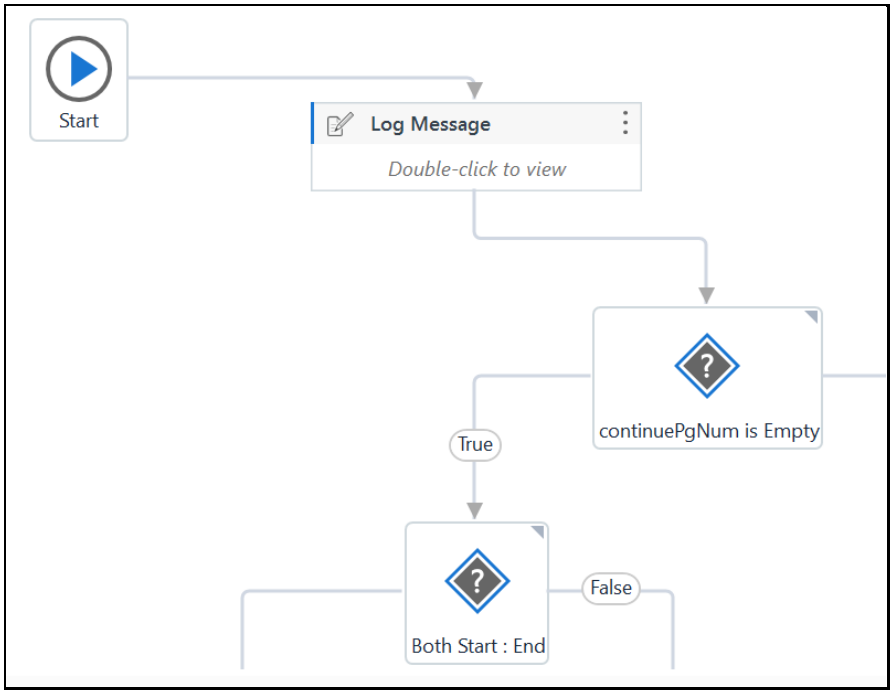


5. SplitPages workflow:



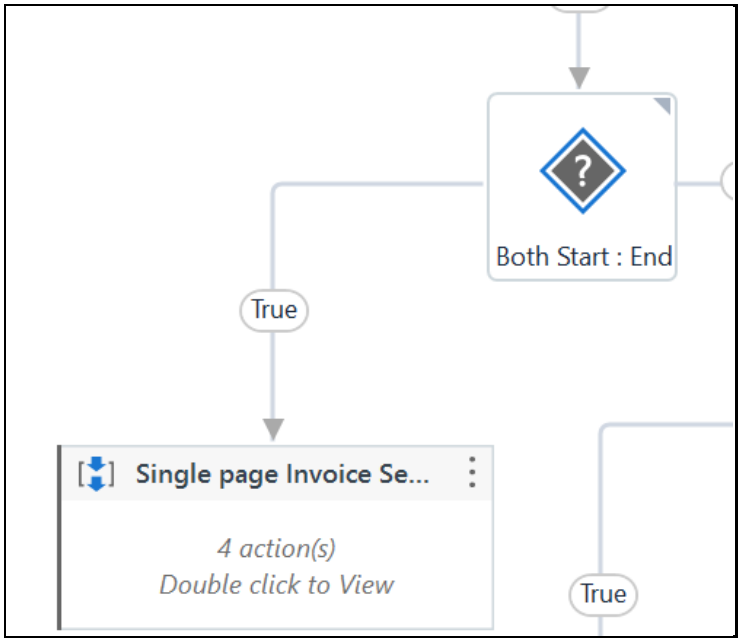
a. 1st Flow Decision – check if io_ContinuePgNum is empty.
Condition – String.IsNullOrEmpty(io_ContinuePgNum)

6. If io_ContinuePgNum is empty:

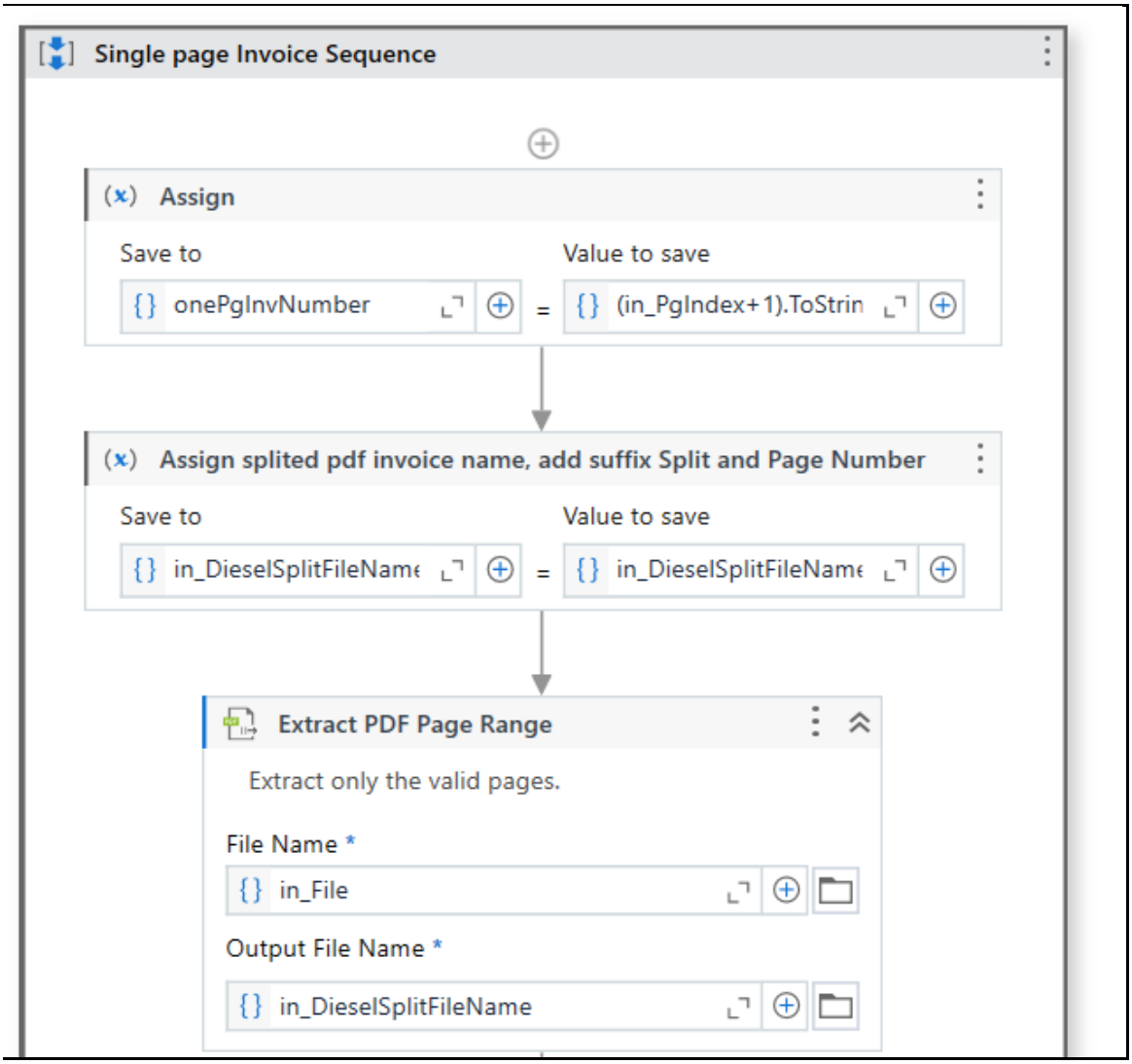


a. Execute 2nd Flow Decision – Both Start : End
Check if both the start keywords and end keywords are present. If present, then it means that the invoice is a single page invoice.

7. If Both Start : End condition is True:



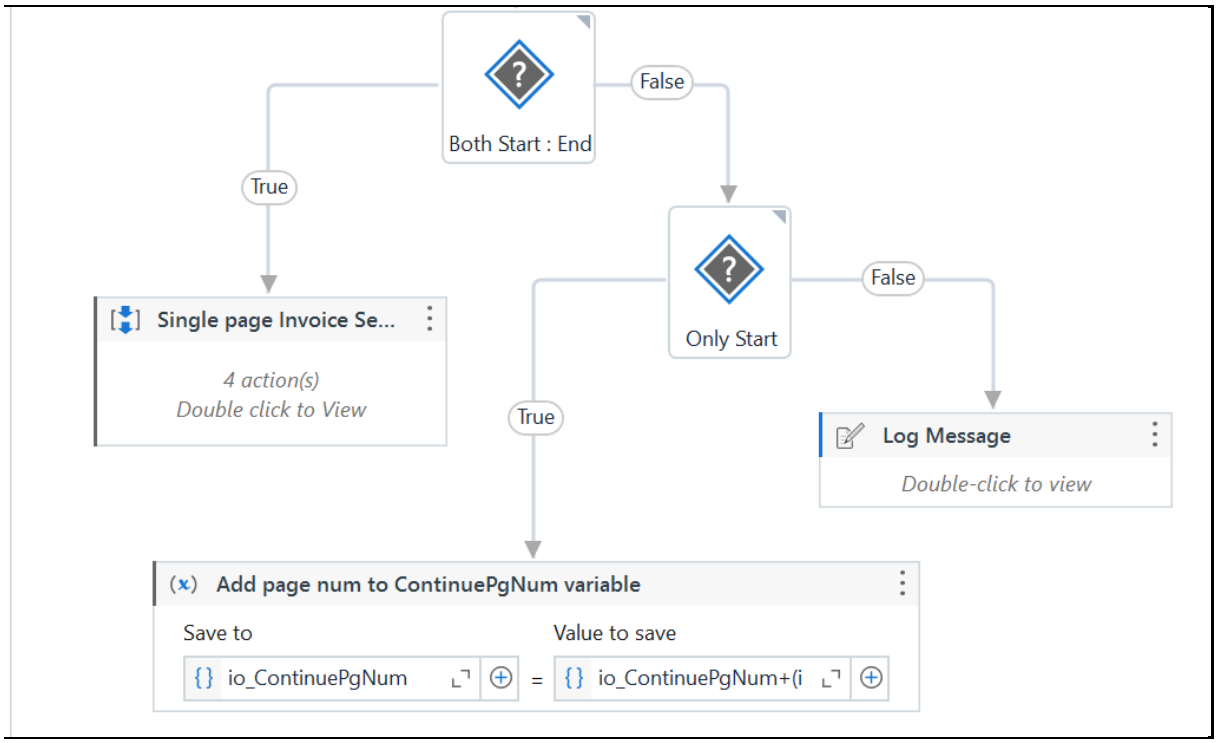
a. Extract invoice page. In Range use only current page index.



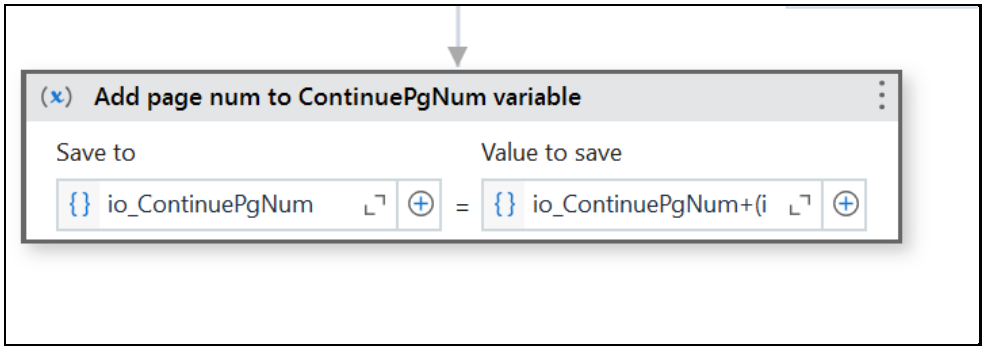
8. Single page invoice is extracted.

---

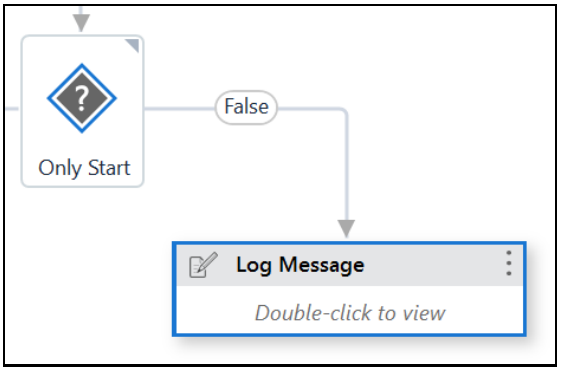
9. In the next For Loop iteration, io_ContinuePgNum is still empty.
10. If Both Start : End condition is False: Means not a single page invoice.
    a. Execute 3rd Flow Decision – Only Start.
       Check if only Start keywords are present.



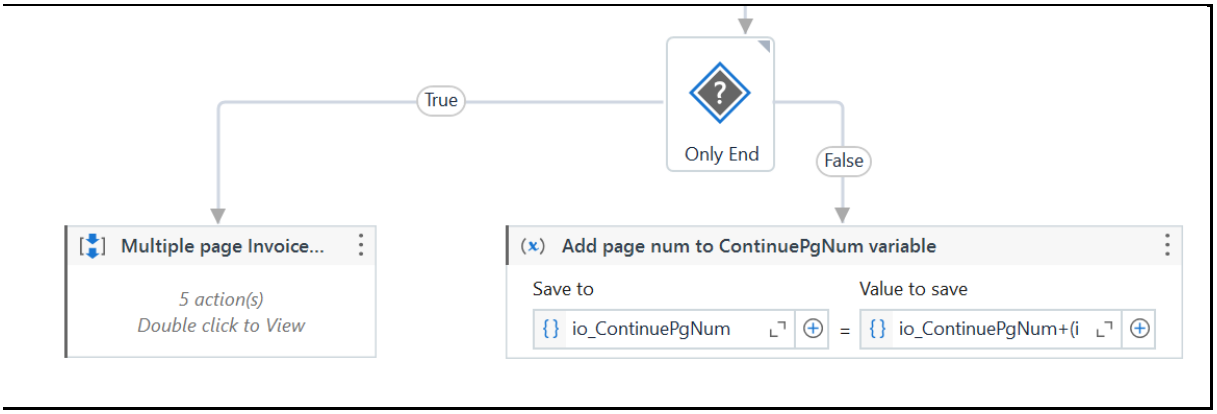
11. If Only Start keywords are present,
    a. then append the current page number to io_ContinuePgNum. Now for next For Loop iteration, io_ContinuePgNum will not be empty.
    Assign: io_ContinuePgNum = io_ContinuePgNum+(in_PgIndex+1).ToString+","



    b. If the Start keywords are not present, then add a log message. If Invoice is correct, this log message will never be executed.

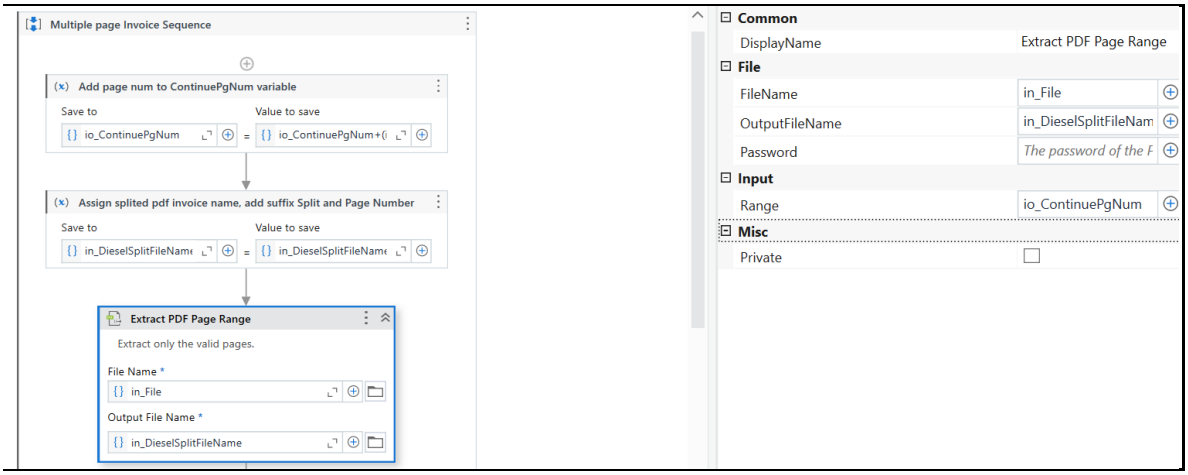


12. In the next For Loop iteration, ‘io_ContinuePgNum’ will not be empty.
    a. Execute 4h Flow Decision: Only End. To check if End keywords are present.

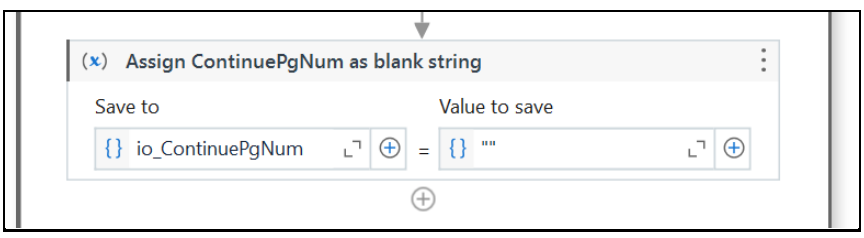


13. If End keywords are not present, then it means that this page is between start and end pages.
    a. Add the current page index to the io_ContinuePgNum.
    Assign: io_ContinuePgNum = io_ContinuePgNum+(in_PgIndex+1).ToString+","

14. If End keywords are present, then it means that this page is the last/end page of the invoice.
    a. Add last page index to the io_ContinuePgNum.
    Assign: io_ContinuePgNum = io_ContinuePgNum+(in_PgIndex+1).ToString
    b. Extract invoice page. In Range use io_ContinuePgNum.



    c. After Extracting, empty io_ContinuePgNum. Assign: io_ContinuePgNum = “” for the next For Loop iteration.



15. Multiple page invoice is extracted.