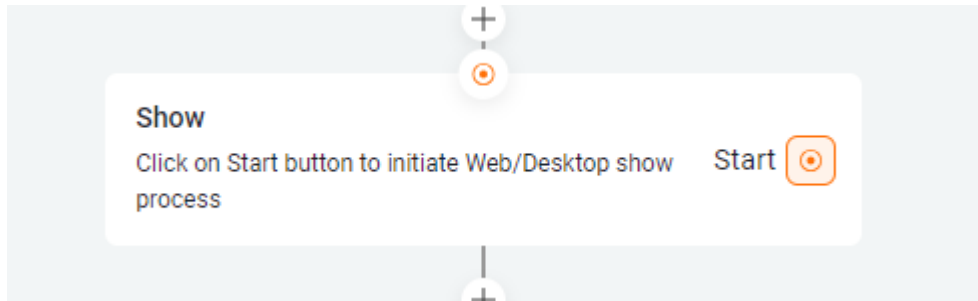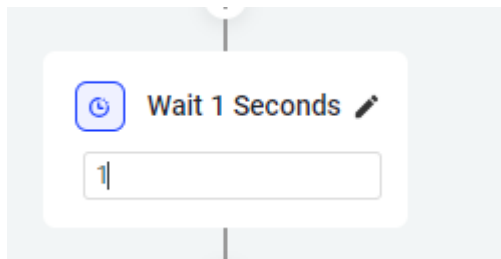# Action Level Documentation for Supervity

This document covers each of the actions in Supervity skill creation journey. These are especially useful while creating your own skill templates using the blank template option.
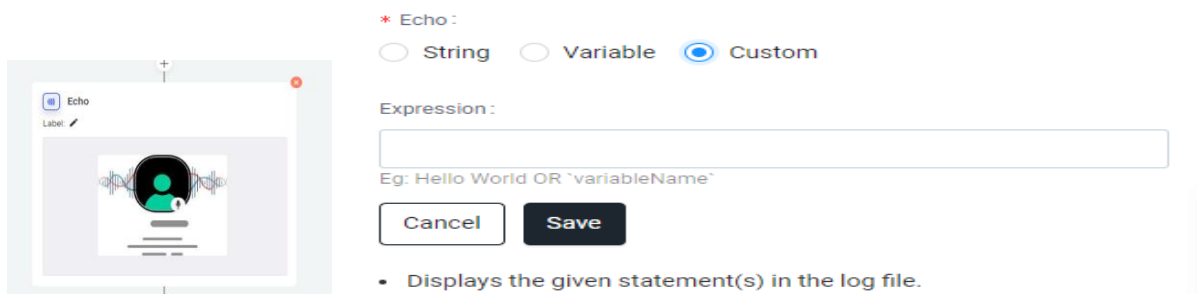
Show

Wait

Waits for a specified amount of time before continuing the workflow.
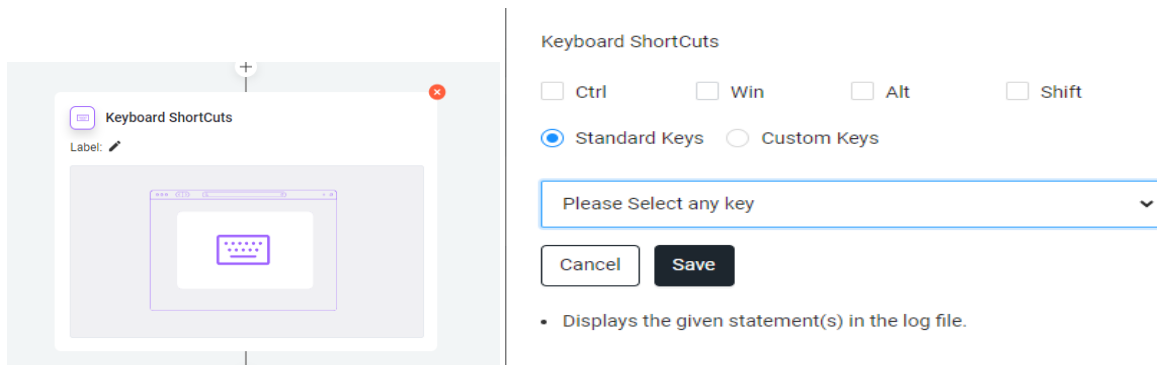Properties
1.Wait <n> Seconds: The amount of time (in seconds) for the following activity to be delayed.

Echo

1. String: The content to be written to the Output panel.
2. Variable: Type the variable name in the Output panel. Variables have to be placed between quotation marks.
3. Custom: The content to be written to the Output panel. This field accepts both texts and variables. Variables must be placed between quotation marks.

Keyboard shortcuts

This action is used to send keyboard shortcuts to active applications in which functions like CTRL, WIN, ALT, and SHIFT are assorted in supervity.
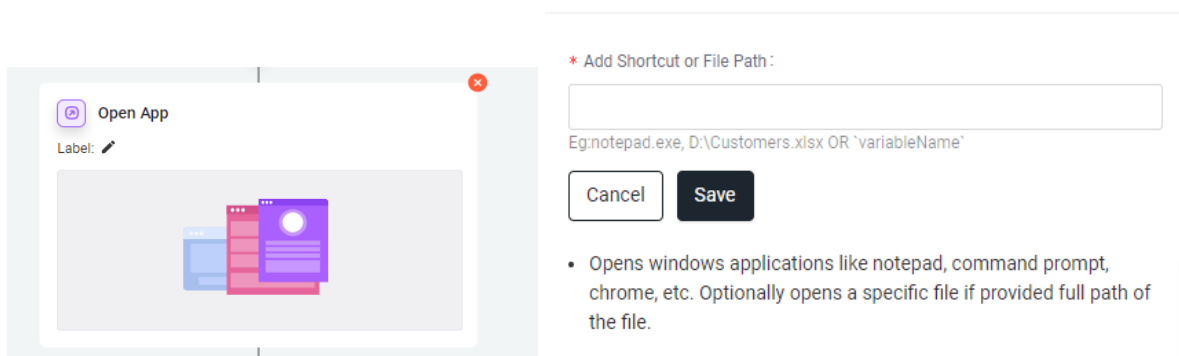
• Properties

1. Custom Keys: Provides a text box in which the user needs to type the function keys to be passed to the active application.

2. Standard Keys: Gives out a list of function keys and all the functions available on the keyboard. Click on keys and select the required function to be used on the active application. Check in the Standard keys and click on the drop-down box to find a drop-down list of the entire keyboard functions; choose ENTER and click on it.

Example

1. Requirement: Need to send (Control +Copy) and enter key using custom keys and Standard keys.

2. Control+ Copy: Check in the Ctrl box and click on Custom keys and type "c" in the text box.

Open App

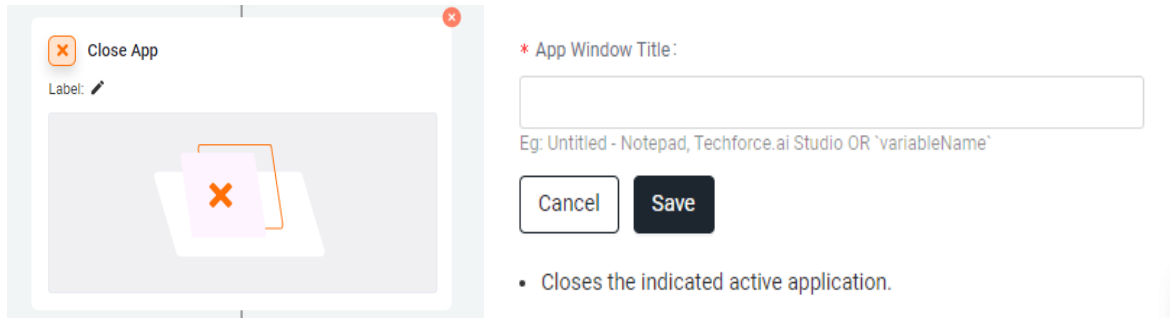

Opens Windows applications.

Properties

1. App Shortcut or File Path: It is used to open Windows applications by giving the file path of the application with an extension like notepad.exe. Copy the path and paste it in the Open App properties window.

Example

Requirement: Need to open Google by using Open App. Copy the full path of Google and paste it in the properties window.

Close App
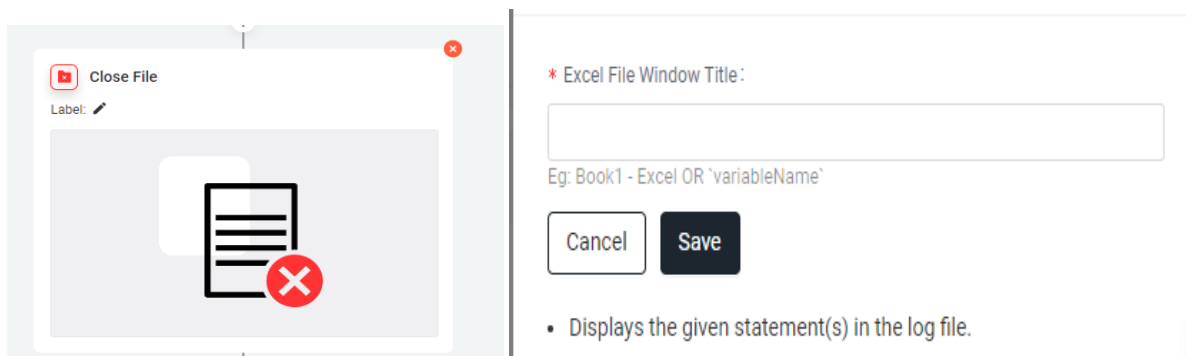

Closes the active application(s).

Properties

App Window Title: Similar to the Open App, Close App also works on the same formulae but instead of the file path, just give the header of the existing app.

Example

Requirement: Need to close the Amazon website using the Close App. Copy the header of Amazon.in and paste it in the properties window.

Close File
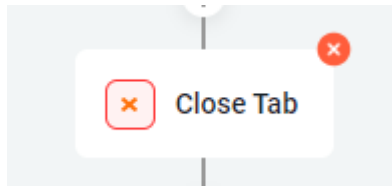

Closes the indicated active file.

Properties

Excel File Window Title : Similar to the Close App, Close File also works on the same formulae but closes the existing file.
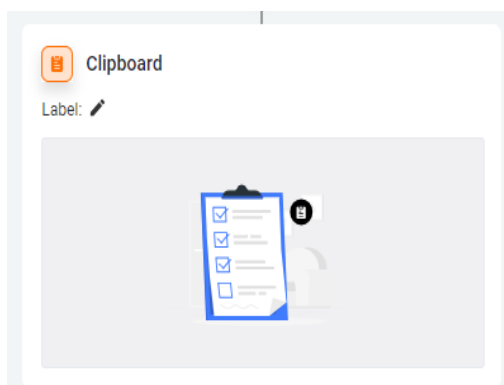
Example

Requirement: Need to close the Excel file using Close File. Copy the header of the Excel file and paste it in the properties window.

Close Tab

Closes the Active tab in the Browser.

Clipboard

To store the data in the clipboard memory into a variable.

1.Variable to capture the Clipboard Data: Create a variable in the variable panel and pass the variable to store the data from the clipboard memory into the variable.

Switch App
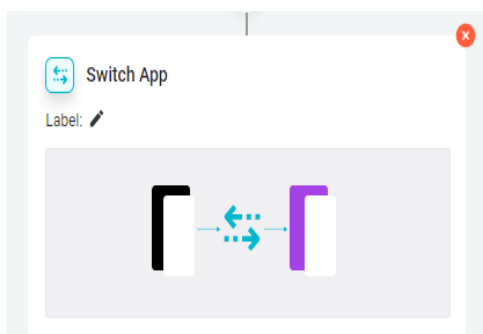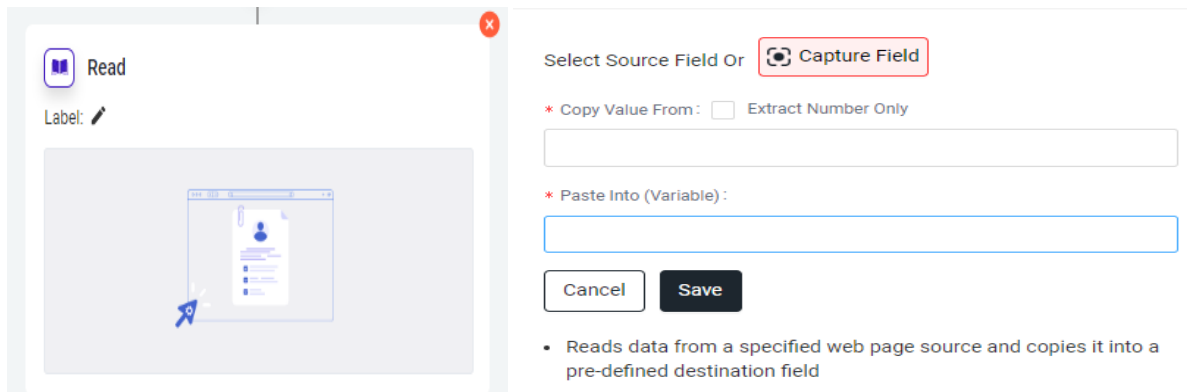

Closes the indicated active file.

Properties

App Window Title: Switch App is used to switch to a particular window or app when multiple windows are open. Just write down the header of the required app/window or variable and paste it in the properties window.

Example

Requirement: Need to switch apps when two apps/windows are active.

Input: Just write down the heading of the required app/window or variable and paste it in the properties window.
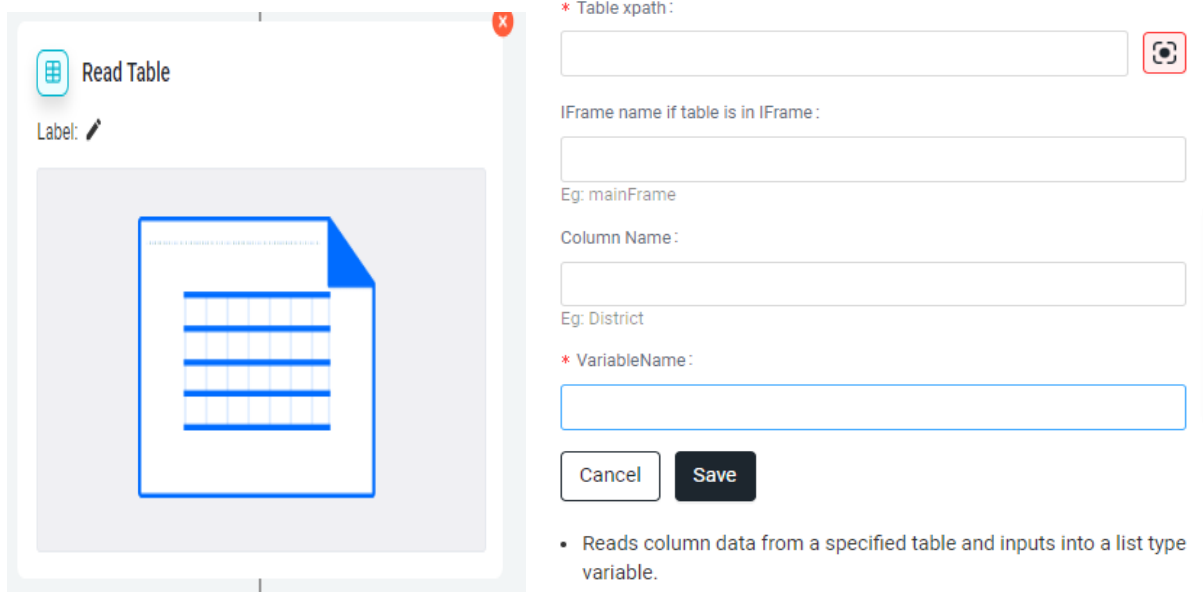
Read

Read data from the web.

Copy Value From: Copy and paste the XPath of the corresponding web element.

Paste Into: Create the variable in Variables Panel and mention the corresponding variable to store the value from the web.

Read Table

You can read the data from tables in web pages.

Properties

Table XPath: Copy the XPath from the corresponding web element.

IFrame name: Enter the iframe name if the table is in the IFrame.

Column Name: Pass the table column name from which you want to retrieve the data.

Select variable: Create the variable as a Data Table. Mention the variable name in the given field.
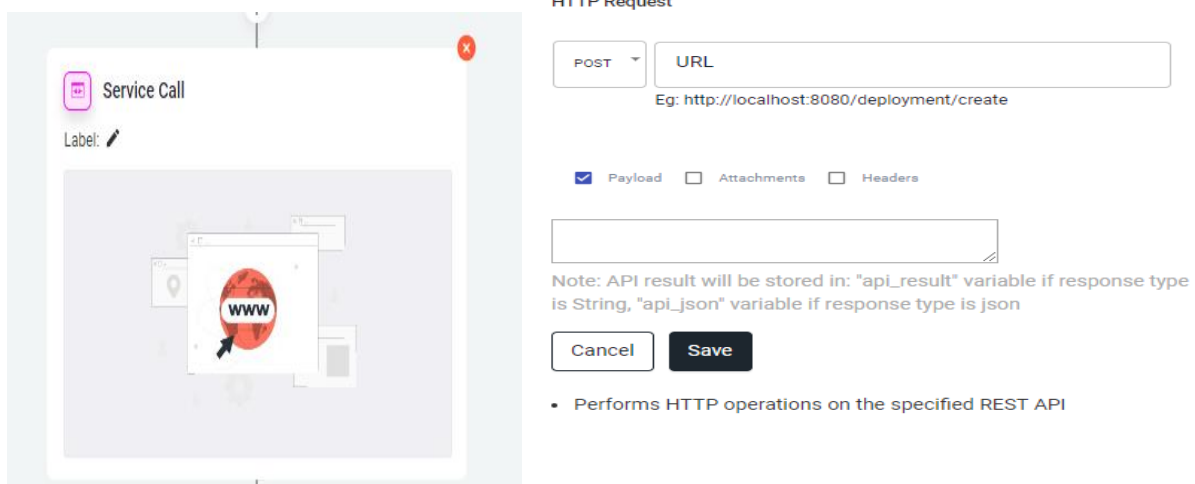
Service call

Enables the user to perform HTTP operations to a specified web API. When you first drag this activity to the Designer panel, the HTTP Request Wizard window appears, which provides an easier way to build requests and preview server responses.

Properties Configuration

URL: URL to hit the corresponding API.

Get: One of the methods to hit API. This will get a response from the corresponding API. Here only headers can be added.

POST: One of the methods to hit API. This sends the JSON object and gets a response from the corresponding API. Here add body(payload) along with the header.

PUT: This request creates a new resource or replaces a representation of the target resource with the requested payload.

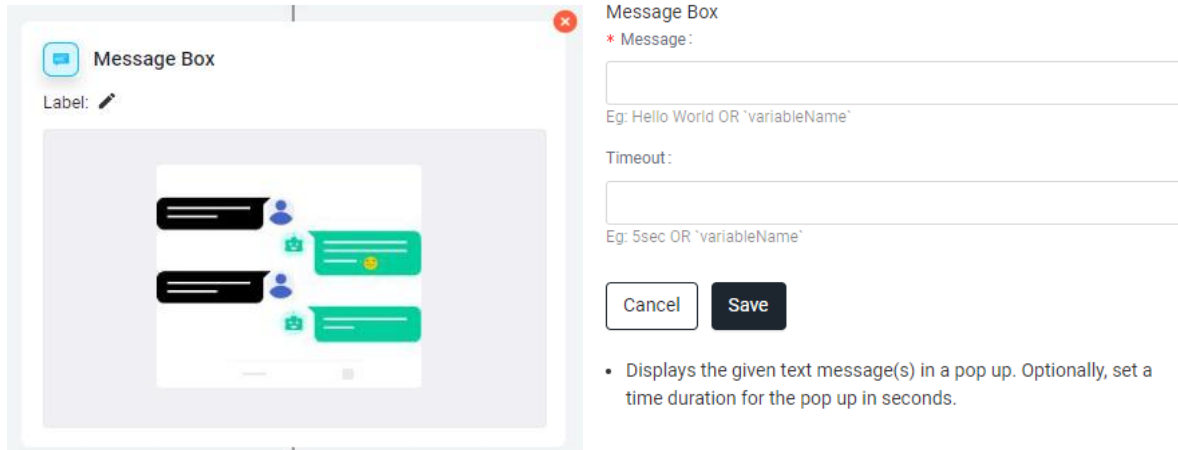DELETE: This request is used to delete the specified resource.

Assign

Allocates any values to a variable. It can be used to increment the value of a variable in a loop. For example, it sums up the values of two or more variables and assigns the result to a different variable.

Properties

1. Custom/Variable: The name of the variable to be assigned a value.

2. Custom/Variable (second field): The value to be assigned to the variable. It may be a string value or can pass the variable.

Message Box

Displays a message box with a given text with the button options. The User can also set the time to present the message box.

Properties:

Message: The text to be displayed in the message box. The User has to enter own text in this box or can pass the variable.

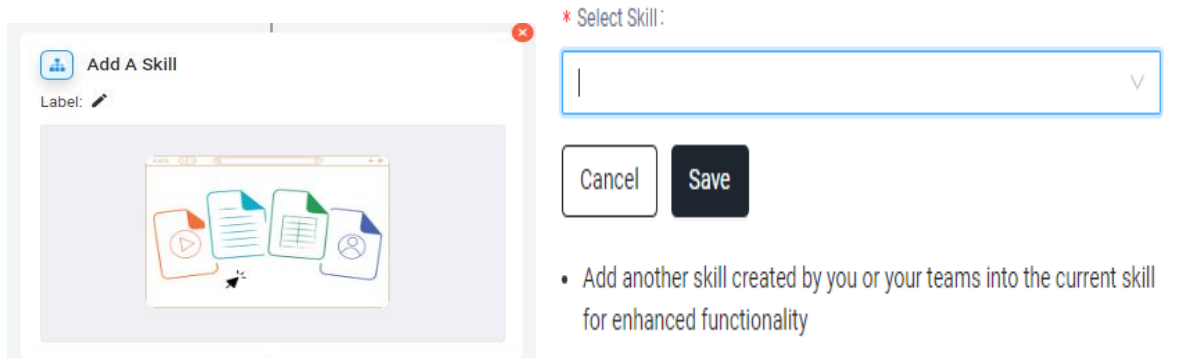Time out: The time to display the message box. Directly pass the variable.

NOTE

Pass the variables between the left quote commands. Ex.(` `).

Invoke Skill

Read Credentials

Add A ToDo

Add A Skill

This action helps to add sub-skills into the main skill.

Select Skill : select the sub-skill from the drop down.

Screenshot
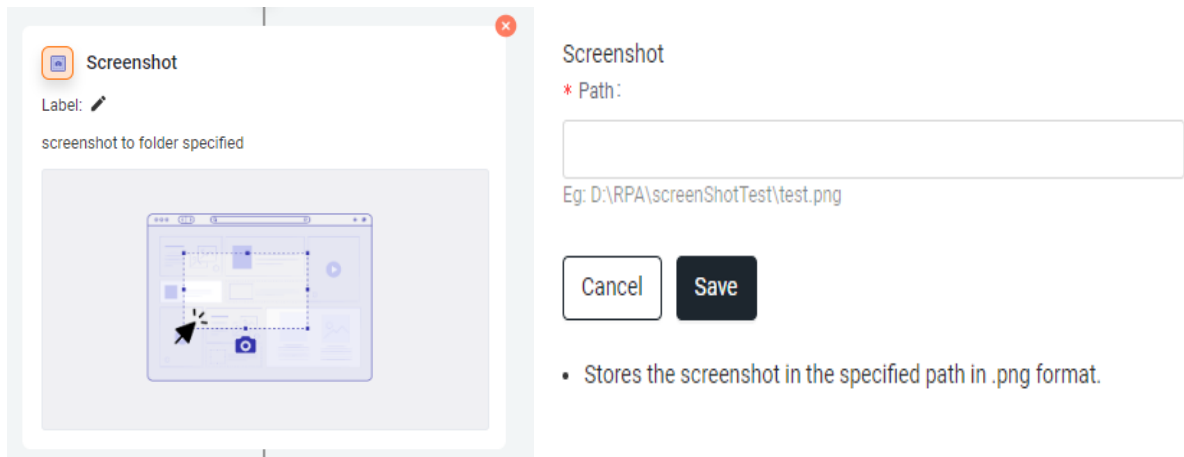
Captures the screenshot of the current screen and saves it in the specified location.

Properties

Path: The screenshot action captures the screenshot of the current screen and saves the image with the specified name and extension in the specified location.

Example

Requirement: To take the screenshot of the current screen.

Input: Just give the location path with name and extension to save the image.

Add Dataset
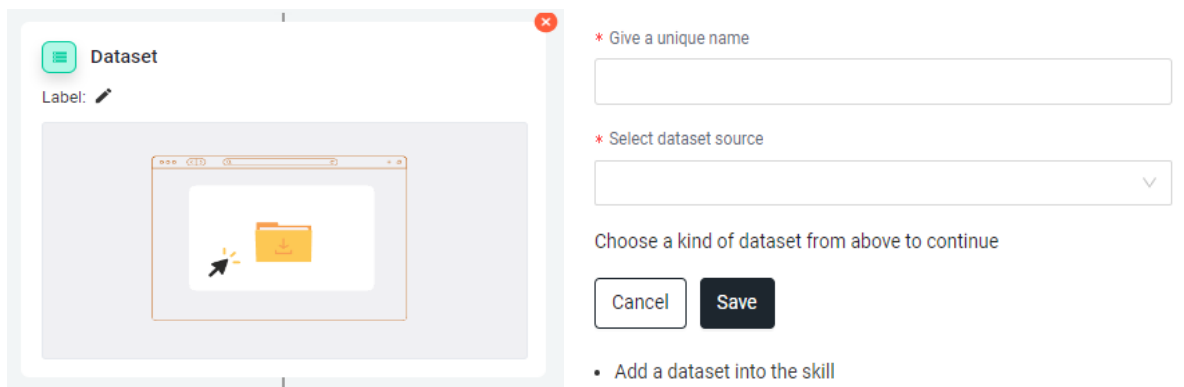
Outlook
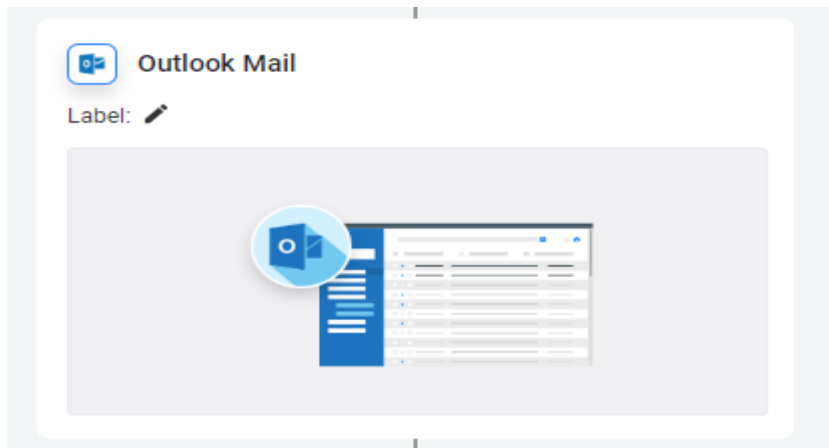
Send Outlook mail – Outlook

Sends emails from the configured outlook account to the specified mail ids.

1. To: The receivers mail id.

2. CC: Mail id if need to add any.

3. BCC: Mail id if need to add any.

4. Subject: The subject of the mail to be sent.

5. Body: The body of the mail to be sent.

6. Attachments: Path of the attachment that is to be sent in the mail.

Note

You can send variables instead of direct text into To, Cc, Bcc, subject, body, and Attachments.

Get Outlook mail – Outlook

Reads emails from the configured outlook account.

Properties

Inbox

1. Folder: Check the Folders to get the Inbox, Sent Items, Custom, and Delete items options.

    1. Inbox: To read the mail data of the specified email from the inbox.
    2. Sent Items: To read the mail data from the specified number of sent emails.
    3. Customs: To read the mail data from any specified folder.
    4. Delete Items: To read the mail data from the specified number of deleted mail items.

2. Unread: Filters only unread emails.

3. Top: Filters a particular number of emails that are present on the top.

4. Enter Variable Name: Stores the emails collected from the corresponding email account.

Note

Here array variable is used to store emails.

1. To access the subject of the mail, use variableName[index].Subject

2. To access the body of the mail, use variableName[index].Body.

3. To access the From address of the mail, use variableName[index].From.

4. To access the To address of the mail, use variableName[index].Received.

Note

We can send variable instead of direct text in to, subject, body

Save Attachment – Outlook

- Stores an outlook email attachment to the given folder.

Saves attachments in the local path, which are downloaded from the corresponding mail account.

Folder Path to Save Attachment : Mention the path where you want to save the attachments.

Mail Object : Mention the index of the Mail object.

SMTP Send Mail
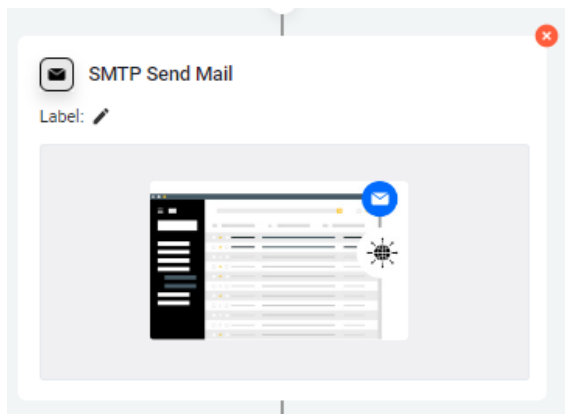
Send SMTP Mail

* Username :

* Password :

* Hostname or IP Address :

* Port :

* To :

CC :

BCC :

* Subject :

* Body :

Attachments :

Eg: D:\myFolder\test.csv

Conditions

Branch
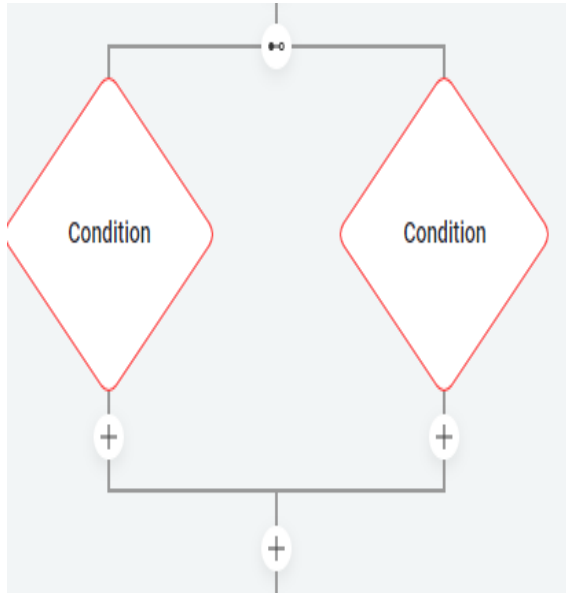
- HTML Selector ◯ Image Selector

- Select from Dropdown ◯ Capture Field

* Source:

* Condition:

* Target Value:

Cancel Save

- Add a branched IF/ELSE condition to the skill

Exception Handling

Try catch.
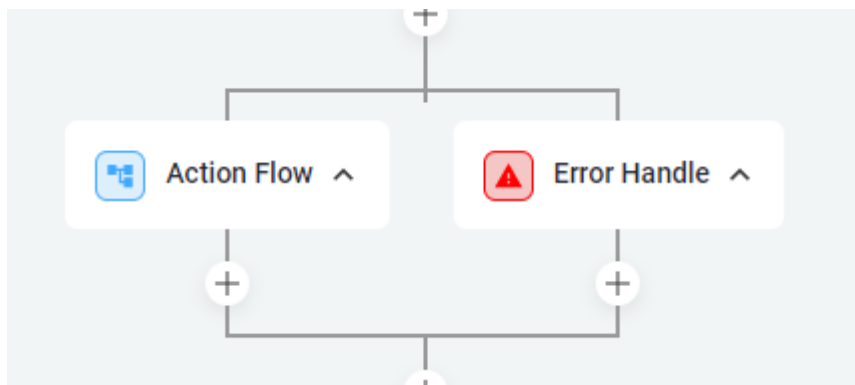
Whatever be the actions involved in the flow, add each step or action in the Try block. After running the flow, if any exceptions are found, those exceptions will be handled by the Catch block

CSV
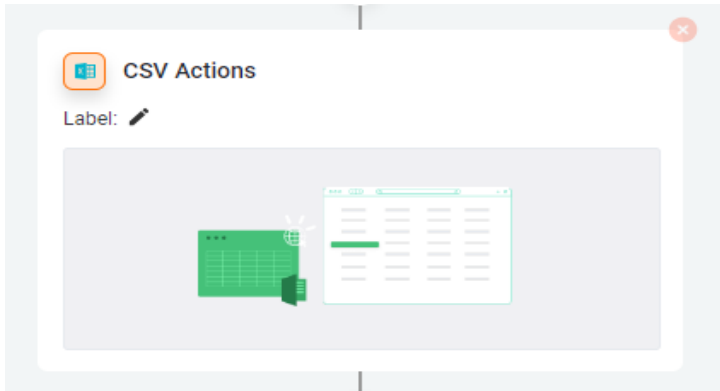
CSV Headers

Select action type

Csv Header ⌄

* CSV File Path (Include file name with extension) :

Eg: D:\\NewFile.csv

Value :

Cancel   Save

CSV Row

Select action type

Csv Row

* CSV File Path (Include file name with extension) :

Eg: D:\\NewFile.csv

* Variable Name :

Eg: variableName

Cancel   Save

Excel
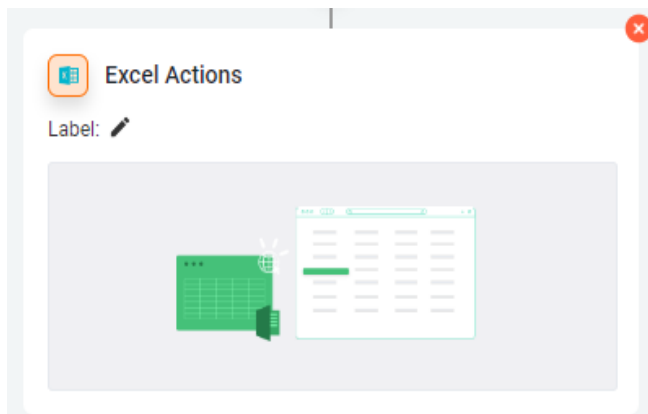
Get Sheet names

Select action type

Get SheetNames

* File Path (Include file name with extension) :

Eg: D:\\NewFile.xlsx or `variableName`

* Variable Name :

Eg: variableName

Cancel   Save

- Returns a list of all the sheet names in the given workbook, ordered by their index.

Returns a list of all the sheet names in a workbook as String variables, ordered by their index (index value starts from zero).

Properties

1. File Path: Need to give the full path of the Excel spreadsheet. This option is shown only while taking the Get Sheet Names action directly. If dragged into the open spreadsheet activity, it will not ask the path, and no need to mention the path.

2. Variable Name: Create one list variable and select that variable from the drop-down. Here only the list variable is allowed to select from the drop-down. The result value will be stored in the list variable.

Get Column

- Reads values from the specified column.

Reads the values from a column beginning with the cell specified in the Starting Cell property field, and stores them in a list variable.

Properties

1. File Path: Need to give the full path of the Excel spreadsheet.

2. Sheet Name: Enter the name of the sheet containing the column that needs to be read. Only String variables and strings are supported (Example: Sheet1).

3. Column Address or Number: The cell from which to start extracting the column data. Mention Column name A or B or number of the column.

4.Variable Name: Need to create the variable as a list. Stores the information from the specified spreadsheet column in a variable. Only list variables are supported.

Hide Column

- Hides the specified column from the spreadsheet.

Hides the column specified in the Column address field.

Properties

1. File Path: Need to give the full path of the Excel spreadsheet. This option is shown only while taking the Hide Column action directly.

2. Sheet Name: Enter the name of the sheet name in which the column that needs to hide. Only String variables and strings are supported (Example: Sheet1).

3. Column Address or Number: The cell which need to hide the column data. Mention Column name A or B or number of the column.

Delete Column

- Deletes the specified column from the spreadsheet.

Deletes a column from a spreadsheet-based on its name.

Properties

1. File Path: Enter the full path of the Excel spreadsheet to be used. This option is shown only while taking the Delete Column action directly.

2. Sheet Name: Enter the name of the sheet containing the column that needs to be deleted. Only String variables and strings are supported (Example: Sheet2).

3. Column Address or Number: The name or number of the column to be deleted. Only String variables and number values are supported.

Get Row

- Reads values from the specified row.

Reads the values from a row beginning with the cell specified in the Starting Cell field and stores it in a list variable.

Properties

1. File Path: Enter the full path of the Excel spreadsheet being used. This option is shown only when taking the Delete Column action directly.

2. Sheet Name: Enter the name of the sheet containing the column to be deleted. Only String variables and strings are supported (Example: Sheet1).

3. Row Number: Enter the exact row number to be read. Only String variables and number values are supported.

4. Variable Name: Need to create the variable as a list. Stores the information from the specified spreadsheet row in a variable. Only list variables are supported.

Delete Row

- Deletes the specified row from the spreadsheet.

Remove a specified row at a certain position.

Properties

1. File Path: Enter the full path of the Excel spreadsheet. This option is shown only when taking the Delete Row action directly.

2. Sheet Name: Enter the name of the sheet containing the Row that needs to be deleted. Only String variables and strings are supported (Example: Sheet1).

3. Row Number: Enter the exact number of the row to be deleted. Only String variables and number values are supported.

Get Cell

- Reads values from the specified cell.

Reads the values from a row beginning with the cell specified in the Starting Cell field and stores it in a string variable.

Properties

1. File Path: Enter the full path of the Excel spreadsheet. This option is shown only when taking the Get Cell action directly.

2. Sheet Name: Enter the name of the sheet containing the Cell to be read. Only String variables and strings are supported (Example: Sheet1).

3. Cell Address: The exact Cell address that needs to be read. Only String variables and number values are supported.

4. Variable Name: Create the variable as string Stores the information from the specified spreadsheet Cell in a variable. Only string variables are supported.

Write cell

Writes a value into a specified spreadsheet cell. If the sheet does not exist, a new one is created with the Sheet Name value. If a value exists, it is overwritten. Changes are immediately saved.

Properties

1. File Path: Enter the full path of the Excel spreadsheet being used. This option is shown only when taking the Write Cell action directly.

2. Sheet Name: Enter the name of the sheet containing the Cell to be written. Only String variables and strings are supported (Example: Sheet1).

3. Cell Address: Enter the exact Cell address to be written. Only String variables and number values are supported.

4. Cell Value: Enter the cell value to be written. It allows only the String and String variable. If passing the value through variable, mention the variable within the tilde symbol (Example: `VaribleName`).

Get Range

Reads the value of an Excel range and stores it in a DataTable variable. If the range isn't specified, the whole spreadsheet is read. If a particular range is specified, it reads the values from that specific range only.

Properties

1. File Path: Enter the full path of the Excel spreadsheet. This option is shown only when taking the Get Range action directly.

2. Sheet Name: Enter the name of the sheet containing the Cell that needs to be read. Only String variables and strings are supported (Example: Sheet1).

3. Range: Specify the range of cells to be read. If this value is not specified, the whole spreadsheet is read. Only String variables and strings are supported to mention that range (Example: B5: G15).

4.Variable Name: Create the variable as DataTable Stores the information from the specified spreadsheet range values in a variable. Only DataTable variables are supported select.

Write Range

Writes the data from a DataTable variable in a spreadsheet starting with the cell indicated in the Starting Cell field. If the starting cell is not specified, the data is written starting from the A1 cell. If the sheet does not exist, a new sheet is created with the value specified in the Sheet Name property. All cells within the specified range are overwritten. Changes are immediately saved.

Properties

1. File Path: Enter the full path of the Excel spreadsheet. This option is shown only when taking the Write Range action directly.

2. Sheet Name: Enter the name of the sheet containing the range that needs to be written. Only String variables (Example: `VaribleName`) and strings are supported (Example: Sheet1).

3. Range: Specify the cell from which to start writing the data. Only string variables and strings are supported (Example: A1 or B1).

4. Variable Name: Select the variable the data that needs to be written to the specified range, as a DataTable variable. Only DataTable variables are supported.

Delete Range

- Deletes the specified rannge from the spreadsheet.

Deletes a specified range in an Excel workbook.

Properties

1. File Path: Enter the full path of the Excel spreadsheet. This option is shown only when taking the Delete Range action directly.

2. Sheet Name: Enter the name of the sheet containing the range that needs to be deleted. Only String variables (Example: `VaribleName`) and strings are supported (Example: Sheet1).

3. Range: Specify the range of cells to be deleted. If this value is not specified, the whole spreadsheet is deleted. Only String variables and strings are supported to mention that range (Example: B5: G15).

Get cell Color

- Reads the colour of the specified excel cell.

Extracts the background Color of a cell and saves it as a Color variable. It can be used within the Open Spread Sheet Action and outside the Open Spread Sheet Action.

Properties

1. File Path: The full path of the Excel spreadsheet that you want to use. This option is shown only if you are taking the Get Cell Color action directly.

2. Sheet Name: The name of the sheet in which the cell color name that you want to read. Only String variables (Eg: `VaribleName`) and strings are supported (Eg: Sheet1).

3. Cell Address: The exact Cell address that you want to get the color. Only String variables and strings are supported (Eg: B1).

4. Variable Name: Create and select the variable as a string. Stores the name of the color from the specified spreadsheet Cell in a variable. Only string variables are supported.

Set range Color

- Sets the given colour to the specified range.

Changes the color of a specified cell or cell range using a Color variable. It Can only be used within the Open Spread Sheet action and outside the Open Spread Sheet Action.

Properties

1. File Path: The full path of the Excel spreadsheet that you want to use. This option is shown only if you are taking the Set Range Color action directly.

2. Sheet Name: The name of the sheet in which the cell color name that you want to read. Only String variables (Eg: `VaribleName`) and strings are supported (Eg: Sheet1).

3. Range Address: The range address that you want to set the color. Only String variables and strings are supported (Eg: A1: B5 or A3).

4. Color: Select the color from the drop-down to set the range color. (Eg: green or black etc).

Get Cell Formula

- Reads the formula of the specified excel cell.

Extracts the formula used in the specified Excel cell.

Properties

1. File Path: The full path of the Excel spreadsheet that you want to use. This option is shown only if you are taking the Get Cell Formula action directly.

2. Sheet Name: The name of the sheet in which the cell formula that you want to read. Only String variables (Eg: `VaribleName`) and strings are supported (Eg: Sheet1).

3. Cell Address: The exact Cell address that you want to get the cell formula. Only String variables and strings are supported (Eg: B1).

4. Variable Name: Create and select the variable as a string. Stores the formula from the specified spreadsheet Cell in a variable. Only string variables are supported.

Filter Columns

- Filters the excel data based on the column conditions.

Filter column action is used to Filter the specified column data from an excel sheet, based on the specified column values.

Properties

1. File Path: Enter the full path of the Excel spreadsheet. This option is shown only while taking the filter column action directly.

2. Sheet Name: Enter the name of the sheet containing the column that needs to be filtered.

3. Include headers: Click the checkbox to get the data from the headers too.

4. Column name: Here you need to provide the column name which you want to extract data.

5. Values: The data which you want to filter.

6. Variable Name: Need to create the variable as a data table. Stores the information from the specified spreadsheet filter column in a variable. Only data table variables are supported.

Convert to xlsx

- Converts the given csv or xls file formats into xlsx files format.

Converts the Excel workbook from XLS to XLSX format.

Properties

1. File Path: Give the full path of the Workbook (.xls file) which you want to convert from XLS to XLSX format.

2. Destination File path: File name is optional. If a file name is not mentioned, then the Workbook is converted from xls to xlsx format with the same name in the same location. If a file name is mentioned for that file, the new file will be created in xlsx format with that filename. If in conversion already a file exists with the same name in that location in xlsx format, then it will overwrite the file with the same name. If passing the file name through a variable, you need to mention that variable within the tilde symbol (Eg: `varibleName`).

Insert column

- Inserts a column in the specified position.

The Insert Column action is used to create a new column into an Excel file.

Properties

1.File Path: Need to give the full path of the excel spreadsheet. This option is shown only while taking the insert column action directly.

2. Sheet Name: Enter the name of the sheet containing the insert column that needs to be inserted.

3. Number of columns: Need to give the number of columns to be added in the excel.

4. Positions: The position from where the column that is to be inserted, the number should be given.

Create Excel

Select action type

Create Excel ▾

* File Path (Include file name with extension):

[   ]

Eg: D:\\NewFile.xlsx or `variableName`

* File Name:

[   ]

Eg: D:\text.xlsx or variableName

[Cancel] [**Save**]

- Performs various actions on the excel file located at the given path.

Refresh Exel File

Select action type

Refresh Excel ▾

* File Path (Include file name with extension):

[   ]

Eg: D:\\NewFile.xlsx or `variableName`

[Cancel] [**Save**]

- Refresh the excel file at the given path.

Sort Excel Data

Sort Excel Data ⌄

* File Path (Include file name with extension) :

Eg: D:\\NewFile.xlsx or `variableName`

* Sheet Name :

Eg:Book or `variableName`

* Column Name :

Eg: Name or Age or `variableName`

○ Ascending   ○ Descending

* Variable Name :

Eg: variableName

Cancel   Save

- Performs various actions on the excel file located at the given path.

Vlook Up

Vlook Up ⌄

* File Path (Include file name with extension) :

Eg: D:\\NewFile.xlsx or `variableName`

* Enter Sheet Name :

* Column Address or Number :

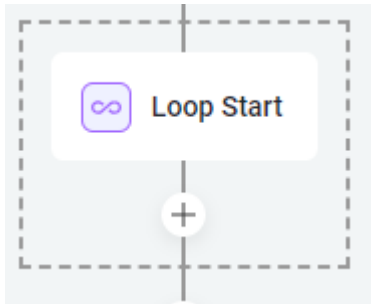Eg: A or B or 1 or 2 or `variableName`

* Column Index :

* Variable Name :

Cancel   Save

- Performs various actions on the excel file located at the given path.
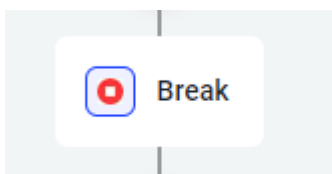
Loops

For

Loops are an efficient way to complete work through multiple iterations until the targeted result is achieved.

Properties

1. Enter Index Variable: The variable that the user wants to iterate.

2. Enter Start Index: The index value of the element to start with. In general, the index starts with zero.
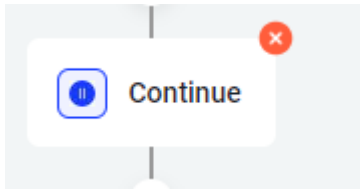
3. Enter FOR_LOOP Range:

Jump statement.

Break

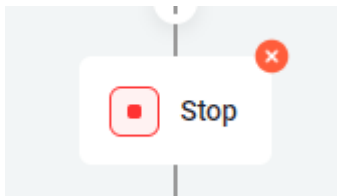Use this statement only for a loop. If the iteration needs to stop after a certain number of executions, then use this Break statement.
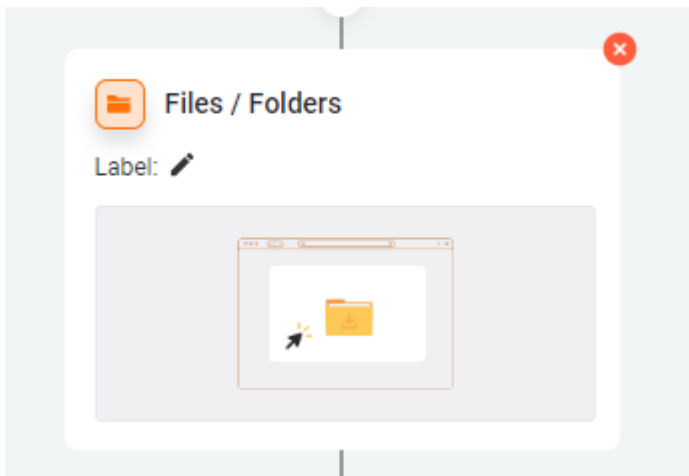
Continue

Use the Continue action in a loop to execute several iterations continuously

Stop

To stop the execution process at the specified step.

Files and Folders

write to file

- Appends or overwrites the given value to the file at specified file path.

Appends/Overwrites the specified string or the data assigned to a variable to the specified file.

Properties

File Path: Give the full path of the file. The path can be given by assigning it to a variable using double slashes. For example: path = C:\\Desktop\\filename.Extension. The variable should be included within `` while passing to file path.

Content to Write: Give the string/variable to be written to the file specified. If it is a string, it can be given directly. If it is a variable, it should be included in ``.

Overwrite: The file data will be overwritten with the string/variable data if any data is existing in the file.

Append: The file data will be appended with the string/variable data if any data existing in the file.

Copy folder /file

Copy/Move a file from one location to another location as specified.

Properties

File

Source Path: Give the full path of the file to be copied/moved including the extension. The path can be given by assigning it to a variable using double slashes. For example: path = C:\\Desktop\\filename.Extension. The variable should be included within `` while passing to file path.

Destination Path: Give the full path of the location where the file needs to be copied/moved including the extension. The path can be given by assigning it to a variable using double slashes. For example: path = C:\\Desktop\\filename.Extension. The variable should be included within `` while passing to file path.

Copy: Makes a copy of the file specified from the source path to the destination path.

Move: Moves the file specified in the source path to the destination path.

Overwrite: Overwrites the file while moving/copying the file specified if there is a file existing with the same name in the destination.

Skip: Skips the copying/moving of the file specified, if there is a file existing with the same name in the destination.

Folder

Source Path: Give the full path of the folder to be copied/moved. The path can be given by assigning it to a variable using double slashes. For example: path = C:\\Desktop\\foldername. The variable should be included within `` while passing to file path.

Destination Path: Give the full path of the location where the folder needs to be copied/moved. The path can be given by assigning it to a variable using double slashes. For example: path = C:\\Desktop\\foldername. The variable should be included within `` while passing to file path.

Copy: Makes a copy of the folder specified in the source path to the destination path.

Move: Moves the folder specified in the source path to the destination path.

Overwrite: Overwrites the folder while moving/copying the folder specified if there is a folder existing with the same name in the destination.

Skip: Skips the copying/moving of the folder specified if there is a folder existing with the same name in the destination.

Get folder content

- Lists the folder contents from the specified folder path.

Lists the contents of the folder based on the filter criteria.

Properties

File

Folder Path: The full path of the folder from where the contents to be retrieved. The path should be given using the forward-slash (/). Example: C:/Desktop/FolderName. If it is given by variable, it should be included in ``. for ex: `variable name`.

Variable: Variable name to list the contents of the folder. The result will be stored in a list.

Which Contents to List

Folders: Lists the folders in the folder specified in the Folder Path.

Files: Lists the files in the folder specified in the Folder Path.

Filter: Filters the files based on the given extension like .png, .xls, .xlsx etc.

Filter Criteria

Date Modified: Filters the contents of the folder based on the date modified selected in the From Date and To Date fields.

Delete file /folder

Deletes the file/folder specified.

Properties File

File Path: The full path of the file to be deleted including the extension. The path can be given by assigning it to a variable using double slashes. Example: path = C:\\Desktop\\filename.Extension. The variable should be included within `` while passing to file path.

Folder

Folder Path: The full path of the folder to be deleted. The path should be given using the forward-slash (/). Example: C:/Desktop/FolderName. If it is given by variable, it should be included in ``. For example: `variable name`.

Read from file

Reads all the content from the file specified and stores it into a variable specified.

Properties

File Path: The full path of the file to be read including the extension. The path can be given by assigning it to a variable using double slashes. Example: path = C:\\Desktop\\filename.Extension. The variable should be included within `` while passing to file path.

Variable to capture File content: Variable name to store the contents of the file specified.

Unzip files

- Extracts the contents from the zipped source file to the destination folder specified.

Technologies

JavaScript

- Encapsulates java script code blocks.

The user can add a JavaScript code snippet using this action. It will support all ES5 methods and functions.

Assign
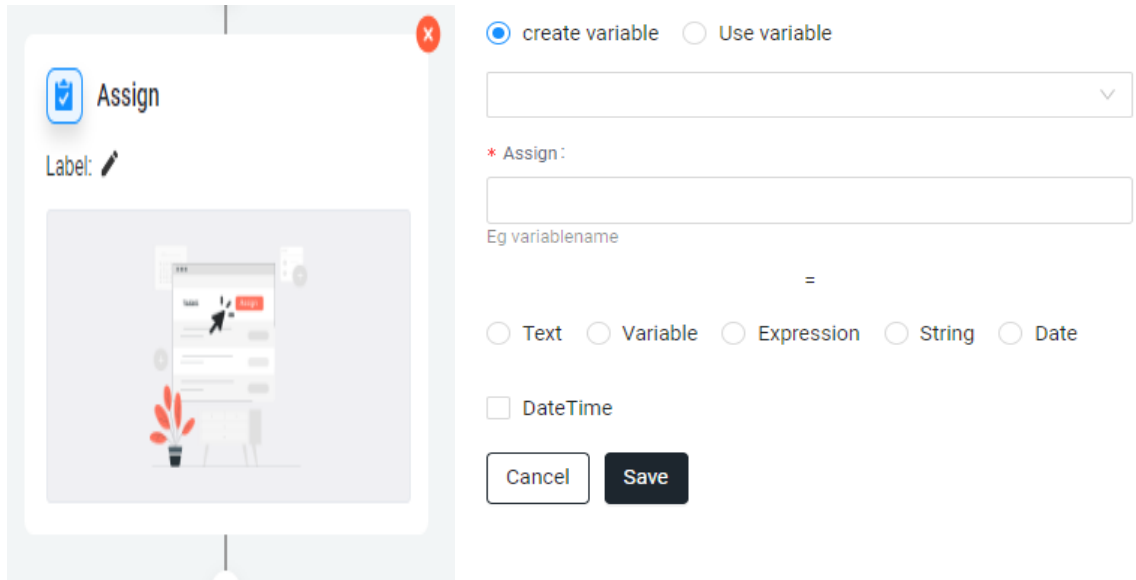
string format
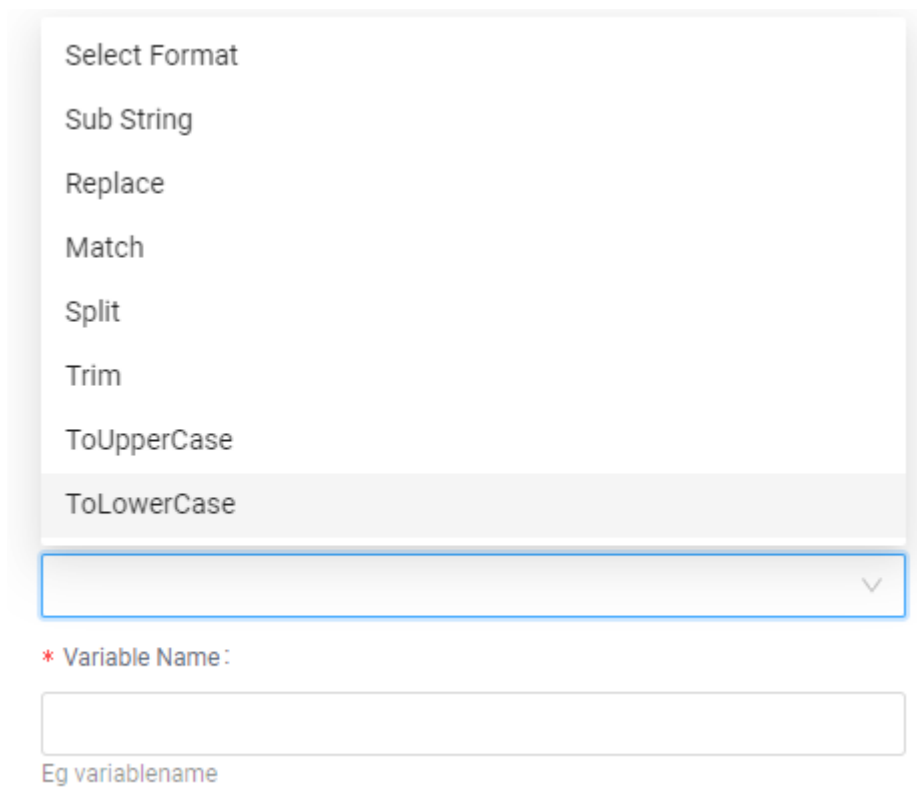
The string format action is used to manipulate the strings according to the required format. The string format contains "Substring", "Replace", "Match", "Split", "Trim", "ToUpperCase", "ToLowerCase" methods. Either string values or RegExp can be used in the methods.

Properties

Method:

1. SubString: The substring method is used to retrieve the part of the main string as per the given index values.

2. Replace: The replace method is used to replace the part of the main string as per the given values.
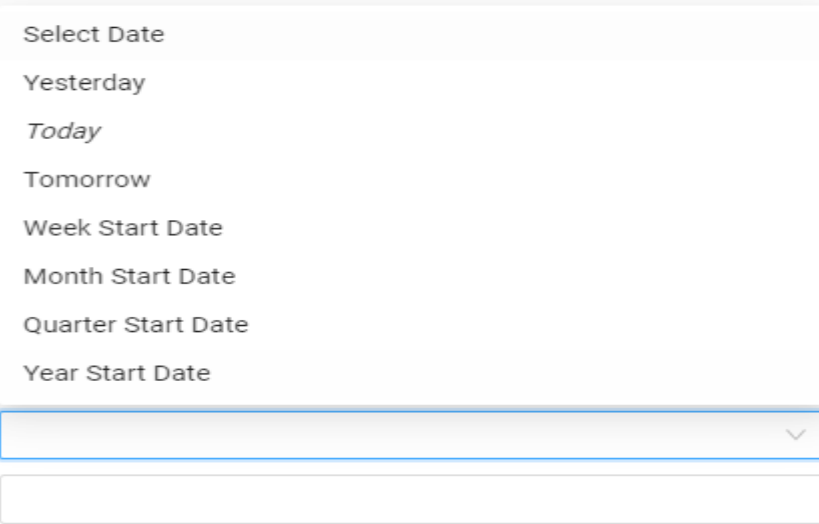
3. Match: The match method is used to match the part of the main string as per the given pattern.

4. Split: The split method is used to split the main string as per the given pattern.

5. Trim: The trim method is used to remove the white spaces from both ends of the main string.

6. To Upper Case: The to upper case method is used to convert all the characters of the main string into Capital case letters.

7. To Lower Case: The to lower case method is used to convert all the characters of the main string into Lower case letters.

8. Variable Name: The result of the Matches action will be stored in a list variable provided

Date format

The Date format action is used to provide a customized date and time according to required formats.

Properties

1. Select: The required date can be selected from the available drop-down list, "Yesterday", "Today", "Tomorrow", "Week start date", "Month start date", "Quarter start date" and "Year Start date".

2. Date Format: The date and time format can be in which the result is to be displayed is passed.

OCR

Supervity OCR

HTTP Request

URL

POST https://ocrtf-apimgmt.azure-api.net/ocr-techforc
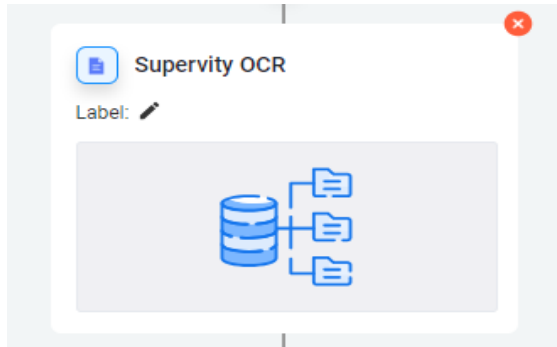
Eg: http://localhost:8080/deployment/create

☑ Payload ☐ Attachments ☑ Headers

Key	Value		
Content-Type	application/json	-	+

Key	Value		
Ocp-Apim-Subsc	Value	-	+

```
{"name":"Your <Filename> goes here"}
```

Valid Json !!

Table Data Prefix

For Eg: if Table Data Prefix is myList, then the table data gets captured in a variable named myList_tableData

Approval

Review

Review

Label: ✏️

---

* Add approvers
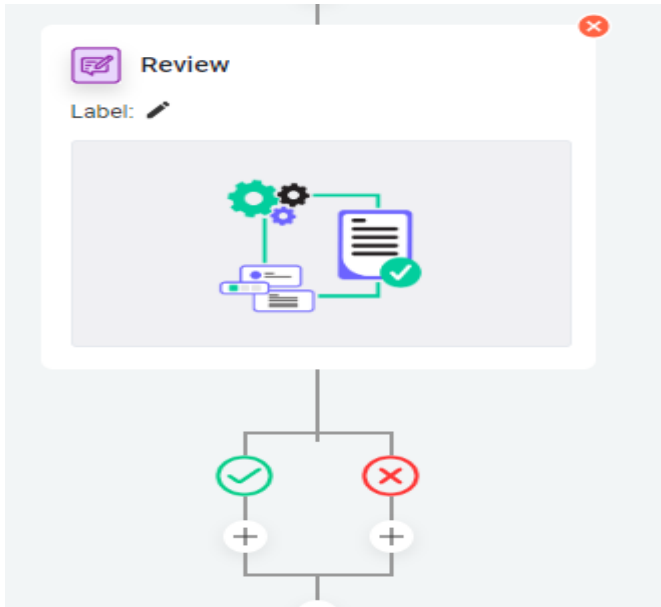
Add approvers email

Approval Rule

⌄

Choose actions to visualize while approval

GET_RANGE
excel-9cc1a37a-3f59-409d-b43d-68b9aad9e516

WRITE_RANGE
excel-bd57fd28-7952-4731-b7fa-dbda1f23e150

GET_CLIPBOARD
GET_CLIPBOARD-f4f9b955-000a-42a6-a4ae-c68934892796

SWITCHAPP-78720f9d-28a3-4083-8b1a-78ec809dcabd

READ_TABLE
READ_TABLE-4d88fd02-3034-4a7b-9a5f-19c60da72665

read
read-ad3df284-9872-44f9-9638-848ea60cb15e

closetab
closetab-3c8ffbd3-ae70-4e8a-9cf3-92daf954123c

echo-fe33a4fe-3d39-4ece-8b26-91e824dd3e67

OPENAPP-47f182cd-b9d4-4b67-9d03-71d96c58b4e0

CLOSEAPP-47c0f917-6746-447c-9466-61c3dee586f4

CLOSE
CLOSE-39ceffe1-4b1a-4562-a62c-0ab3fc72017d

HTTP
httpapi-ba543543-b4d1-405a-91f0-78a33df65ff4

KEY_STROKES
KEY_STROKES-d044b859-2451-413a-af6c-a5a03aad1bd8

wait
Wait 1 Seconds
wait-f6ea1ef6-f3fa-41ca-9d09-d160fbbfab08

record-62261cc0-58f6-4dcb-90bc-2dfea04573f2

read
read-52200b9a-958c-4af8-bac0-22452c2ec414

messagebox-4dd438e4-5110-4988-85eb-450aab3d1460

invokeSkill
addSkillSet-ec49e3d2-e69f-4e16-9aa0-ef1a4d6592dc

SCREENSHOT
screenshot to folder specified
SCREENSHOT-21915a50-2e9d-4096-948e-724f8acbe72f

data-set-d942d102-5c39-439d-81d7-fcf3f2e6932f

sendOutlookMail
sendMail-b0dfa099-4fac-4f23-a537-ccb6eb1023a4

SEND_SMTP_MAIL
SMTP-9258caab-897c-49af-983f-75e913d6becd

break
Break
break-4c945859-5c74-422a-a6eb-9da7ea507dd0

continue
Continue
continue-68118fab-56ed-4ed2-a83a-2348639fc5ac

filesandfolders-56dd811f-05c7-4ee1-97e8-d418b9569646

TECHNOLOGY-1f876ac3-8473-4898-ba54-ed6a9db6103b

ASSIGN
assign-050bf0a1-9389-4d75-b0dd-911db7f77d43

HTTP
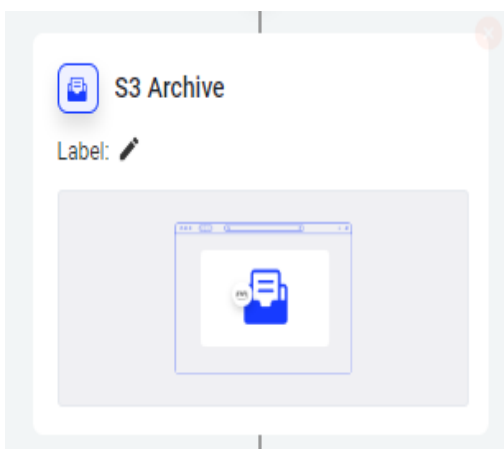supervityOcr-ac8baa95-4287-49b1-8efb-3ac3ab02216c

Cancel   Save

S3

S3 Connect

Access Key :

Secret Key :

Cancel　Save

- Connects to AWS S3 bucket.

S3 File Archive

Region :

Source Bucket Name :

Source KeyName :

Destination BucketName :

Destination KeyName :

Cancel　Save

- Moves the processed files to another S3 bucket.

S3 File Download

Region :

Bucket Name :

Folder Path :

Key Name :

☐ Archive

Cancel　Save

- Downloads the file from S3 bucket to local system.