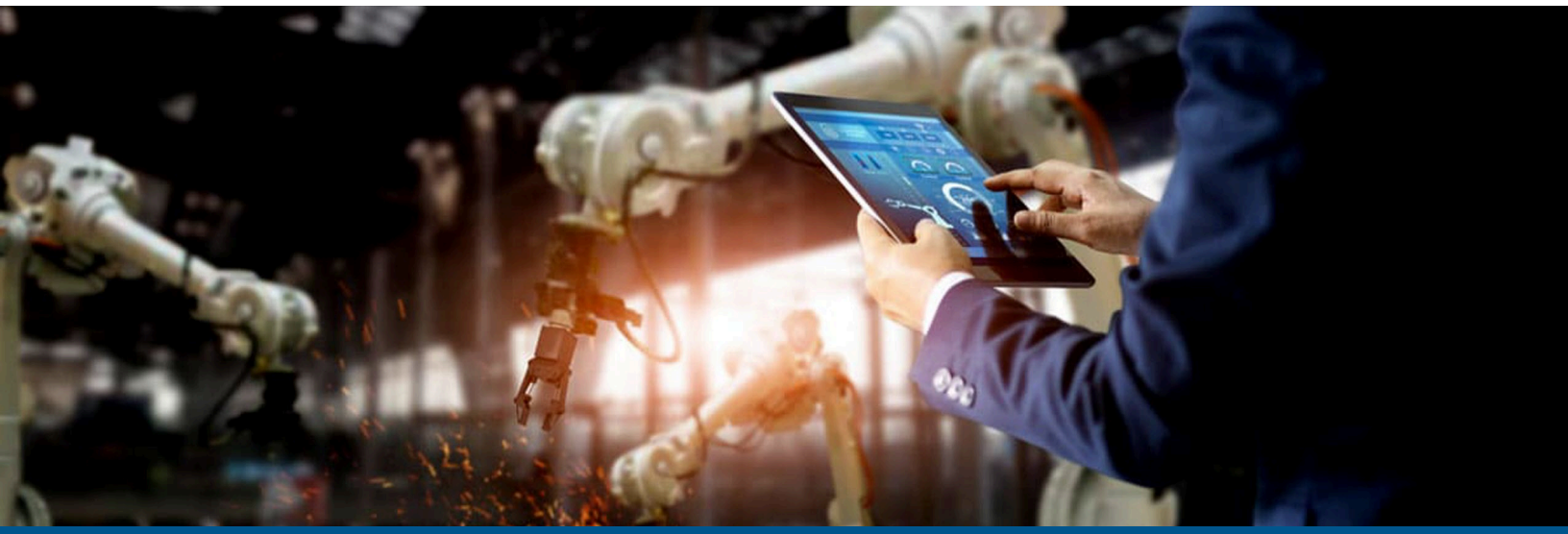


Kofax RPA

Getting Started with Robot Building

Version: 11.5.0

Date: 2023-10-02

KOFAX

Table of Contents

Preface

This guide provides a tutorial that walks you through the process of using Kofax RPA to build a robot.

The instructions provided in this document assume that you have downloaded and installed Kofax RPA on your computer. See the chapter "Quick Start Guide" in the *Kofax RPA Installation Guide* to start using the product.

Related Documentation

The documentation set for Kofax RPA is available here:[1]

https://docshield.kofax.com/Portal/Products/RPA/11.5.0-nlfihq5gwr/RPA.htm

The documentation set includes the following resources listed in alphabetical order:

Kofax RPA Administrator's Guide
Describes administrative and management tasks in Kofax RPA.

Kofax RPA Best Practices Guide
Offers recommended methods and techniques to help you optimize performance and ensure success while using Robot Lifecycle Management in your Kofax RPA environment.

Kofax RPA Desktop Automation Service Guide
Describes how to configure and manage the Desktop Automation Service required to use Desktop Automation on a remote computer.

Kofax RPA Developer's Guide
Contains programmer user guides for the Java and the .NET APIs used to execute robots on RoboServer. Also, includes information on the Management Console REST services provided with the product.

[1] You must be connected to the Internet to access the full documentation set online. For access without an Internet connection, see the *Installation Guide*.

Kofax RPA Getting Started with Robot Building Guide

Provides a tutorial that walks you through the process of using Kofax RPA to build a robot.

Kofax RPA Getting Started with Document Transformation Guide

Provides a tutorial that explains how to use Document Transformation functionality in a Kofax RPA environment, including OCR, extraction, field formatting, and validation.

Kofax RPA Help

Describes how to use Kofax RPA. The Help is also available in PDF format and known as *Kofax RPA User's Guide*.

Kofax RPA Installation Guide

Contains instructions on installing Kofax RPA and its components in a development environment.

Kofax RPA Java API documentation

Provides access to the Kofax RPA Java API packages and classes for developers to use with Kofax RPA.

The Kofax RPA APIs include extensive references to RoboSuite, the original product name. The RoboSuite name is preserved in the APIs to ensure backward compatibility. In the context of the API documentation, the term RoboSuite has the same meaning as Kofax RPA.

Kofax RPA Release Notes

Contains late-breaking details and other information that is not available in your other Kofax RPA documentation.

Kofax RPA Technical Specifications

Contains information on supported operating systems and other system requirements.

Kofax RPA Upgrade Guide

Contains instructions on upgrading Kofax RPA and its components to a newer version.

Kofax RPA User's Guide

Contains instructions for using Kofax RPA and its components. Includes the *Kofax RPA Help* topics, plus more in depth coverage not available in the *Help*.

Training

Kofax offers both classroom and computer-based training to help you make the most of your Kofax RPA solution. Visit the Kofax Education Portal at https://learn.kofax.com/ for details about the available training options and schedules.

Also, you can visit the Kofax Intelligent Automation SmartHub at https://smarthub.kofax.com/ to explore additional solutions, robots, connectors, and more.

Getting help with Kofax products

The Kofax Knowledge Portal repository contains articles that are updated on a regular basis to keep you informed about Kofax products. We encourage you to use the Knowledge Portal to obtain answers to your product questions.

To access the Kofax Knowledge Portal, go to https://knowledge.kofax.com.

The Kofax Knowledge Portal is optimized for use with Google Chrome, Mozilla Firefox, or Microsoft Edge.

The Kofax Knowledge Portal provides:

- Powerful search capabilities to help you quickly locate the information you need.
  Type your search terms or phrase into the **Search** box, and then click the search icon.
- Product information, configuration details and documentation, including release news.
  To locate articles, go to the Knowledge Portal home page and select the applicable Solution Family for your product, or click the View All Products button.

From the Knowledge Portal home page, you can:

- Access the Kofax Community (for all customers).
  On the Resources menu, click the **Community** link.
- Access the Kofax Customer Portal (for eligible customers).
  Go to the Support Portal Information page and click **Log in to the Customer Portal**.
- Access the Kofax Partner Portal (for eligible partners).
  Go to the Support Portal Information page and click **Log in to the Partner Portal**.
- Access Kofax support commitments, lifecycle policies, electronic fulfillment details, and self-service tools.
  Go to the Support Details page and select the appropriate article.

Chapter 1

Build a robot

Overview

With Kofax RPA, you can build robots that can automate work processes involving Windows and Java applications on your networked computers to have automated control of these applications.

The **robot workflow** is a sequence of steps executed one after the other. The steps model how a user would interact with the application that is being automated.

Steps are the basic building blocks of the robot workflow. Some steps are simple and perform one action such as moving a mouse or pressing a key. Others, called composite steps, may contain additional steps.

When editing the robot workflow, you are presented with a view of the robot and the applications being automated along with details on the robot state and buttons to control the robot manually.

For more information, see "Robot Building" in *Kofax RPA Help*.

Desktop Automation Service

To automate applications on a remote computer, install and configure the Desktop Automation Service and connect the service to Design Studio.

For details, see the *Kofax RPA Desktop Automation Service Guide*.

As the network environment and applications may vary with each computer, the tutorial "How to build a Robot" does not involve the use of applications on a remote computer. Therefore, you can build the tutorial robot in Design Studio without installing the Desktop Automation Service.

How to build a Robot

This step-by-step tutorial shows how to create, edit and use a Robot . The tutorial covers some of the most commonly used functions such as Loop steps, Extract Value steps, Conditional step, Assign step, Enter Text step, and others.

The tutorial consists of four main parts:

- Using the built-in browser, extract information from the Education page on the Kofax website (https://learn.kofax.com/index.php/jem-categories/category) about the first three training courses available in the schedule
- Using built-in Excel driver, write the extracted information to a spreadsheet
- Save the Excel file to a local folder
- Close the browser and Excel

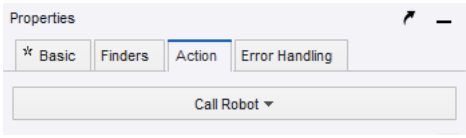
Preliminary steps

Before proceeding to the main sections of the tutorial, complete the following preliminary steps. First, you need to create a Basic Engine Robot in Design Studio, create a Robot, and then call it from the Basic Engine Robot. You also need to create a type to store the extracted data. As opposed to Basic Engine Robots that are identified by a blue icon , Robots are identified by a green icon .

1. **Create a Basic Engine Robot**
   a. Start **Design Studio**.
   b. Click **File** > **New Basic Engine Robot**.
   c. Name the robot **TrainingSchedule**, select a project, and then click **Finish**.
      The new robot appears on a new tab in the editor window. By default, the Smart Re-execution (Full) execution mode is selected and the End step is selected in the created robot.
   d. To start editing and executing your robot, you need to prepare it for execution by clicking **Prepare Execution** in the Applications view or on the toolbar. By clicking this action, you put the robot into execution mode, which enables you to execute it while editing. You can execute action steps right after you insert them in the robot workflow and immediately see the result. When a Basic Engine Robot is not prepared for execution, you can still perform some basic editing, such as add steps, but you will not be able to execute the steps and see the result.

      ⚠ Only one Basic Engine Robot at a time can have the execution privilege, so to take the execution privilege from one robot to another, open the tab with the required robot and click **Prepare Execution**.

   e. Insert an **Action Step** in the new robot. To do so, right-click the robot workflow and click **Insert Step** > **Action Step**.
   f. Save the changes.
2. **Create a Robot**
   a. Click **File** > **New Robot**.
   b. Name robot **TrainingScheduleDA**, select a project, and then click **Finish**.
      The new robot appears on a new tab in the editor window.
3. **Call a Robot from a Basic Engine Robot**
   a. Open the tab with the **TrainingSchedule** Basic Engine Robot.
   b. In the inserted step, click **Select an Action** on the **Action** tab and select **Call Robot**.



c. In the **Robot** drop-down list, select the **TrainingScheduleDA** robot.

d. Save the changes.

e. When the execution is allowed, open the Robot workflow. To do so, click **Step Into Robot** on the toolbar.

The tab with your **TrainingScheduleDA** robot is opened.

When a Basic Engine Robot has the execution privilege, the editor tab of this robot is highlighted. When a Basic Engine Robot is calling a Robot, the tabs of both robots are highlighted for convenience as shown below. The robot where execution is currently located is marked with a red dot.

4. **Create a type**

a. Click **File** > **New Type**.

b. Name the type **TrainingScheduleType**, select a project, and then click **Finish**.

c. Click the plus sign to add new attributes to the type. Add the following attributes and specify their types:

Name	Attribute Type
Date	Short Text
Course	Short Text
Location	Short Text
LocationInfo	Short Text

Short Text is a simple type that can contain text, not exceeding one line.

d. Save the changes.

You are now ready to start designing the automation workflow. Proceed to the next section.

Open website and Excel

1. **Open the Kofax Education website with the built-in browser**

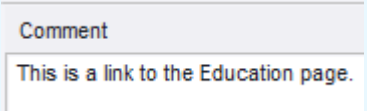
a. First, open the tab with the **TrainingScheduleDA** robot. To open the Education page on the Kofax website in the built-in browser, right-click the first flow point (small circle) and click **Applications** > **Browse**. Rename the step to **Open Website**.

b. Expand the inserted Browse step, in the **Browser** list, select **Chromium**, in the **Action** list, select **Load Page**, enter the application name (such as "app"), and then paste the following URL to the **URL** property:

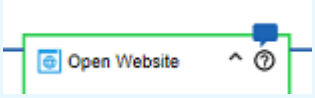
`https://learn.kofax.com/index.php/jem-categories/category`

c. Click **Step Over** to execute this step. In the **Recorder View**, the website is opened in a new tab.

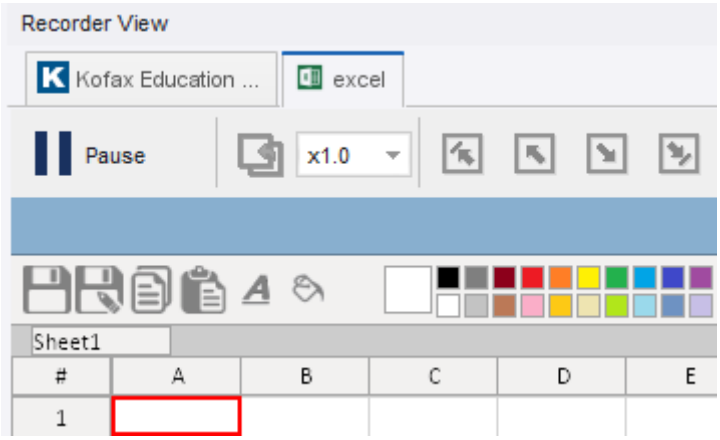
If you have some information to note, you can leave comments to the steps in your automation workflow.



To write or change a comment, click a step or a Group step and add/change your notes in the Comment window. You can use the Undo and Redo buttons here. The comment is automatically saved when you click outside the window. A step that contains a comment is marked with a comment sign.



2. **Open the built-in Excel driver**
   a. Right-click the next flow point in the workflow and click **Applications** > **Excel**.
   b. Expand the inserted step and in the **Action** list, select **Create File**.
   c. Click **Step Over** to execute this step. In the **Recorder View**, Excel is opened in a new tab.



When finished, proceed to the next section.

Extract information from website

In this procedure, you extract information on the first three scheduled courses, which includes the date that the course will be held, course title, course location, and URL to the location page containing additional information.

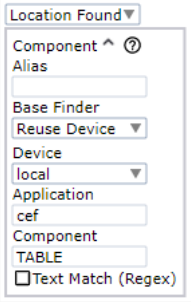
1. **Locate the website area to extract information and add a loop**

   a. In the built-in browser, right-click the first cell of the first row in the table and click **Loop** > **Each Table Row** > **Exclude first row**.

   This action inserts in your robot a For Each Loop step that iterates over all table rows, except for the header row.

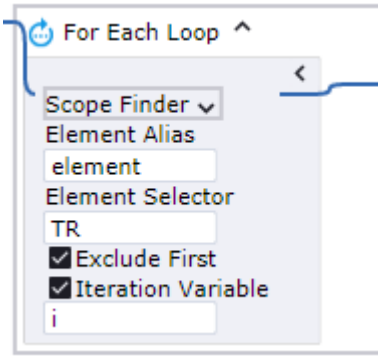
   b. In the workflow, expand the "For Exclude first row" step and do the following:

   - Expand the **Component** box and verify that the properties match the following screen.

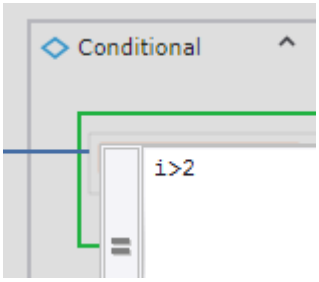


   The Component field contains the specific upper-level table element found by the selector: `TABLE`. This component is used to find the correct elements when looping over the table rows.

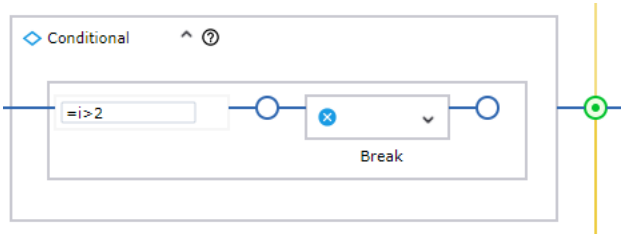
   - Expand the **For Each Loop** box, then click **>** to open the loop property panel. Select **Iteration Variable** to store the iteration value, and enter **i** as the variable name. Also, verify that **Exclude First** is selected.



   - To extract information only on the first three course (from the first three rows), add a condition to the loop. Right-click the flow point to the right of the loop property panel and click **Conditions and Control** > **Conditional** . In the step, click the plus sign, click the text field, and then type the condition **i > 2** so that the expression is evaluated.



Then, right-click the flow point next to the text field and click **Loop** > **Break**. Double-click the flow point next to the Conditional step to execute to this point.

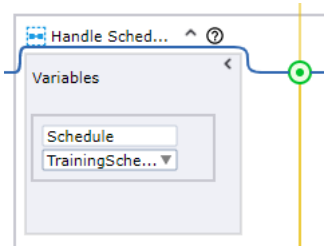


The loop now iterates over the first three rows and then stops.

2. **Add steps to extract the information**

   a. Before adding steps that extract information from the table rows and write it to an Excel spreadsheet, for convenience, add a Group step to the For Each Loop step. In the **For Each Loop** box, right-click the flow point to the right of the Conditional step and click **Conditions and Control** > **Group**. For example, you can name it "Handle Schedule Data."

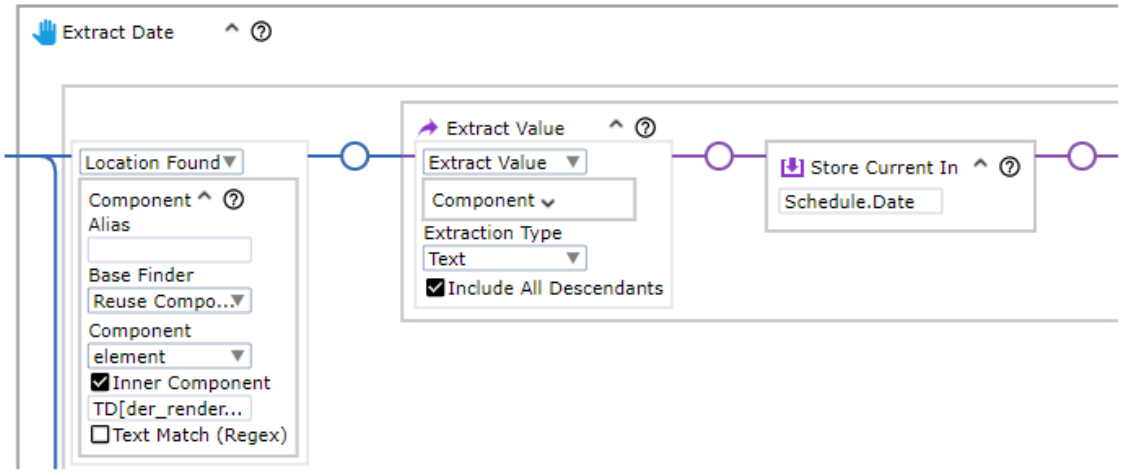
   b. In the Group step, expand the **Variables** box, specify a name for the variable to store extracted information, such as "Schedule," and from the drop-down list, select the type **TrainingScheduleType**. Double-click the flow point inside the created Group step to execute to this point.



   c. In the built-in browser, in the table with training courses, right-click the first cell of the second row that contains the date range and then click **Extract Value From** > **Text Into** > **Schedule: TrainingScheduleType** > **Date: Text**.

   Ensure that the entire cell is selected, not the date range.

   The Extract Value step is inserted in the workflow. For example, you can name it "Extract Date."



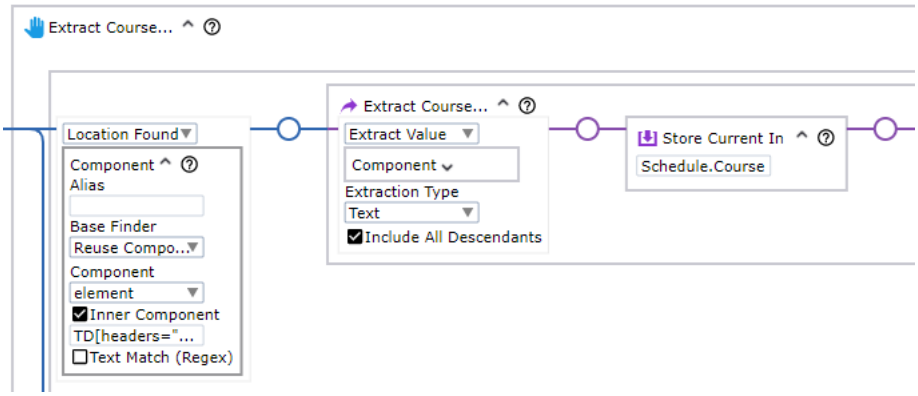
The Inner Component field is used to find components within the already found table component and contains the following lower-level table element: `TD[headers="jem_date"]`. `TD` denotes "table row."

d. Click **Step Over** to execute this step.

e. In the same row, right-click the cell with a course title and click **Extract Value From** > **Text Into** > **Schedule: TrainingScheduleType** > **Course: Text**.

Ensure that the entire cell is selected, not the title itself.

The Extract Value step is inserted in the workflow. For example, you can name it "Extract Course Title."



The Inner Component field contains the following lower-level table component: `TD[headers="jem_title"]`.

f. Click **Step Over** to execute this step.

g. Repeat the same action for the Location cell, selecting the **Location** variable this time.

The Inner Component field will contain the following lower-level table component: `TD[headers="jem_location"]`.

You can name the step "Extract Course Location."

h. Click **Step Over** to execute this step.

3. **Extract a relative URL to the web page for each location**

   a. Right-click in the Location cell, this time selecting the location name itself, not the entire cell. Then click **Extract Value From** > **Attribute** > **href Into** > **Schedule: TrainingScheduleType** > **LocationInfo: Text**.

   The Extract Value step is inserted in the workflow. The Inner Component field contains the following lower-level table component:

   ```
   TD[der_rendered="y"]:nth-of-type(3) > A[der_rendered="y"]
   ```

   You can name it "Extract Relative URL."

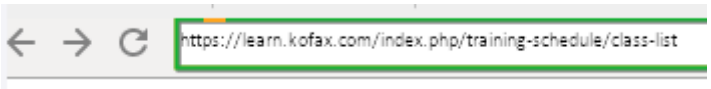
   b. Click **Step Over** to execute this step.

4. **Compose an absolute URL to the web page for each location**

   In the preceding step, you extracted the *relative* URL to the web page for a course location. For example, `/index.php/ilt-training-locations/venue/1-kofax-mechelen-belgium`. This URL cannot be used independently as it does not contain the base URL `https://learn.kofax.com`.

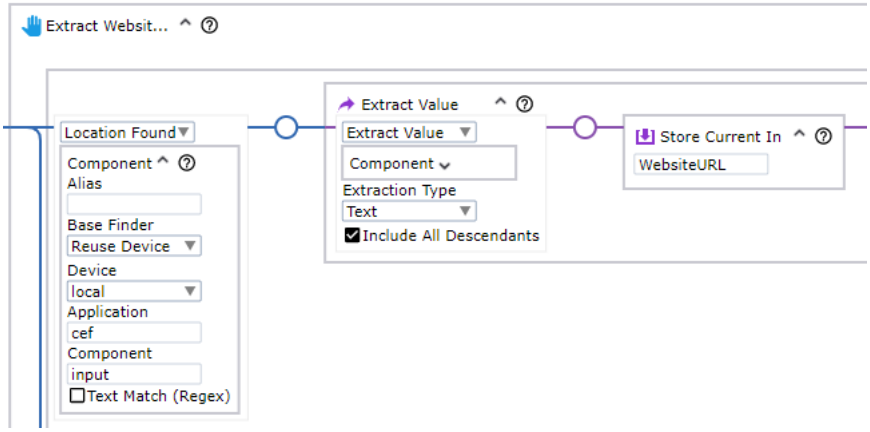
   To convert it to an *absolute* URL that can be used separately, you need to extract the Kofax Education page URL, trim it to the base URL as shown below, and then concatenate it (link) with the relative URL as shown in Write extracted information to Excel.

   a. Right-click the website address at the top of the page and click **Extract Value From** > **Text Into** > **Create Variable**.

   b. In the new dialog box, assign a name for the variable to store the extracted URL. For example, name it **WebsiteURL** and then click **OK**.

   The Extract Value step is inserted in the workflow. You can name it "Extract Website URL."

c. Click **Step Over** to execute this step.

This step extracts the URL to the Education page on the Kofax website and stores it in a variable.

d. Now you need to trim the Education page URL so it only contains the base URL.

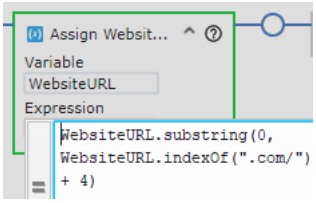
1. Right-click the flow point after the **Extract Website URL** step and click **Assign and Convert** > **Assign**.

The Assign step is inserted in your workflow. Expand the step.

2. In the **Variables** field, enter **WebsiteURL**. In the **Expression** field, enter the following expression: **WebsiteURL.substring(0, WebsiteURL.indexOf(".com/") + 4)**

With this expression, the string contained in the WebsiteURL variable is trimmed to a substring. In other words, the Rankings page URL is shortened to only contain the base part ending with ".com".

3. Click the gray bar on the left so the equal sign appears, and the expression can be evaluated.



4. Click **Step Over** to execute this step.

5. In the end, you have five Extract Value steps and one Assign step. Group the steps. You can name the group "Extract Information."

Double-click the flow point next to this group to execute to this point.

After the group is executed, you can check the extracted values. In the **State** pane on the right, expand the **Variables** branch.

When finished, proceed to the next section.

Write extracted information to Excel

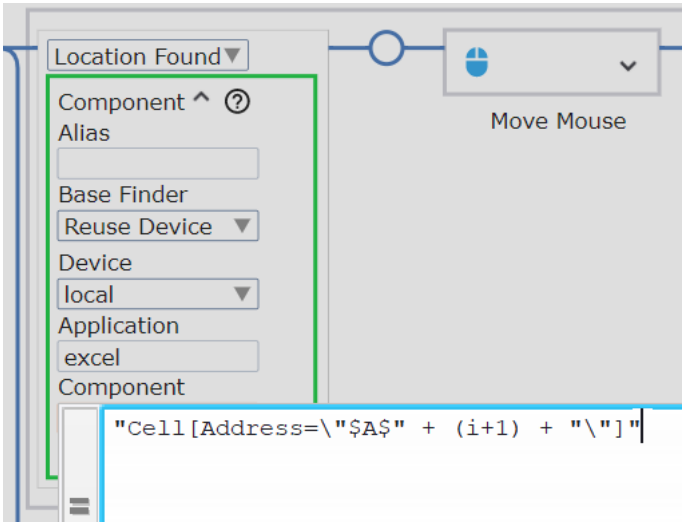
1. **Add steps that write the extracted information to an Excel spreadsheet**

a. In Excel, right-click the A1 cell, and then click **Replace Text** > **From variable** > **Schedule: TrainingScheduleType** > **Date: Text**.

The "Input text from Schedule.Date" step is inserted in the workflow.

b. Expand the step, expand the **Component** box, and then add **+ (i+1) + "\"** to the expression in the **Component** field to move to the next line in the spreadsheet after the current line. The final expression must look like the following: **"Cell[Address=\"A" + (i+1) + "\"]"**.

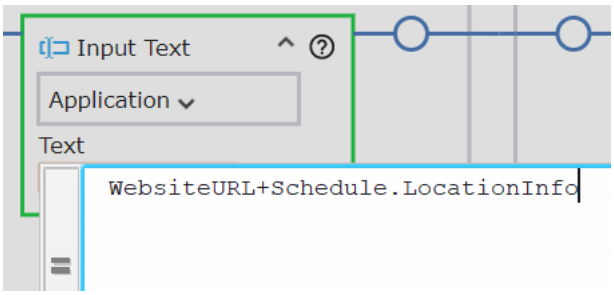
Click the gray bar on the left so the equal sign appears.



c. Click **Step Over** to execute this step.

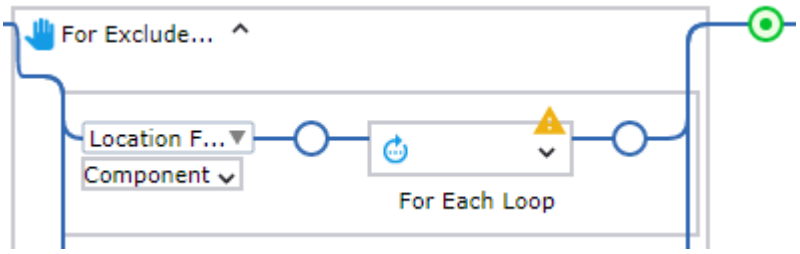
d. Repeat these actions for the B1, C1 and D1 cells in the spreadsheet, but select the respective variables each time: **Course**, **Location**, and **WebsiteURL**.

2. Expand the **Input text from WebsiteURL** step and then expand the **Input Text** box.
In the **Text** field, enter the following expression: **WebsiteURL+Schedule.LocationInfo**
Click the gray bar on the left so the equal sign appears.



With this expression, the two URLs that you extracted in the previous section are concatenated to become an absolute URL to each course location web page on the Kofax website.

3. In the end, you have four "Input text from" steps. Group the steps. For example, you can name the group "Insert to Excel."
Double-click the flow point next to the "For Exclude first row" step to execute the entire step.



Observe how the information is being extracted from the web page and written into Excel.

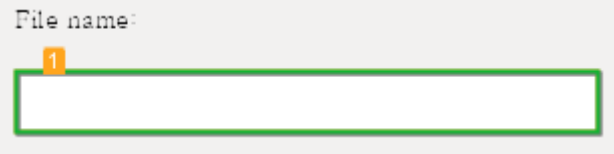
When finished, proceed to the next section.

Save Excel file locally and close applications

1. **Save the Excel spreadsheet**

   a. In Excel, right-click the Save button and click **Click** > **Left**.
   The Left Click step is inserted in the workflow. Click **Step Over** to execute this step.

   b. In the Save As dialog box, select the "File name" text field.

   Then, right-click the field and click **Replace Text** > **Fixed Value**.
   Specify the location where to save the file and the file name, such as `C:/Documents/KofaxRPAScheduledCourses.xlsx`, and click **OK**. Make sure the specified path exists.
   The Input step is inserted in the workflow. Click **Step Over** to execute this step.

   c. Select and right-click the **Save** button and then click **Click** > **Left**.
   The Left Click step is inserted in the workflow. Click **Step Over** to execute this step.

   For convenience, you can rename the steps.

2. **Close built-in Excel driver and the built-in browser**
   To ensure that open windows are not duplicated when re-starting the robot, which may lead to an error, add steps that close the Excel window and the web page at the end of the run.

   a. Select the Excel tab, right-click the Close button in the upper right corner and click **Click** > **Left**.
   Click **Step Over** to execute this step. The Excel tab is now closed.

   b. Select the browser tab and perform the same action on the Close button.
   Click **Step Over** to execute this step. The browser tab is now also closed.

   For convenience, you can rename the steps.

> Recorder View
> **Recorder View**, click **Unrecorded Instant Click**, and select the **Left** mouse click. The Unrecorded Instant Click action is also useful when you need to see available options in context menus and drop-down lists without recording these actions in the workflow.
> To close built-in application tabs, such as Browser, Excel, or others while editing a robot, you can just click the Close button in the top right corner of the tab. This is not a step and the tab closes as any other window.

Your Robot is now ready for use. Save the robot. After you save the created workflow, refresh it, and then click **Start Execution** to execute the workflow from the beginning. When the Robot finishes executing, navigate to the selected location and review the results in your Excel file.

- To step out of the Robot and switch to the Basic Engine Robot, click **Step Out** on the toolbar after the entire workflow is executed. In the Basic Engine Robot, the Call Robot step is now shown as executed.
- To close the robot without executing it to the end or returning a result, click **Leave Robot** on the toolbar. The tab with the Basic Engine Robot is now opened. The Call Robot step is now shown as *not* executed.