WinActorEye

Scenario Creation Manual

NTT ADVANCED TECHNOLOGY CORPORATION

Contents

WinActorEye　Scenario Creation Manual

1. Introduction

1.1. About this document

This document is a manual for creating scenarios that work in cooperation with WinActorEye.

WinActorEye is a tool to search for a target image with high accuracy by combining image matching under relatively simple conditions while checking the result after performing a selected filter processing. By combining with scenarios of WinActor, you can operate the mouse for the coordinate information of the target image.

In Chapter 2, you will create a scenario to click the "submit" button on the page where the WinActor sample file "Tutorial.html" is opened in Microsoft Edge (hereinafter, Tutorial page). Through the tutorial in this manual, you can learn the functions of WinActorEye.

For how to use WinActorEye, see the material No.4 in Table 4.1-1 in "4.1Reference materials."

1.2. Trademarks

The names described below and other names of companies and products in this document are trademarks or registered trademarks of their respective companies. The TM, ®, and © marks are omitted in this document.

- Windows, Microsoft .NET Framework, and Microsoft Edge are trademarks or registered trademarks of Microsoft Corporation in the United States and other countries.
  * The official name of Windows is Microsoft Windows Operating System.

1.3. Notes on this manual

- The copyright notice "Copyright © 2013-2025 NTT, Inc. & NTT ADVANCED TECHNOLOGY CORPORATION" attached to this manual and the provided software cannot be changed or deleted.
  The copyright of this manual belongs to NTT, Inc. and NTT ADVANCED TECHNOLOGY CORPORATION.

- The descriptions in this manual assume that users understand Windows operations and functions. For information that is not described in this manual, see the documents provided by Microsoft.

2. Scenario creation tutorial

2.1. Structure and operation of a scenario

The structure and operation of a scenario to be created are shown below.

You will create a scenario to click the "submit" button on the Tutorial page (inside the red frame in Figure 2.1-1).

By changing a name of a subroutine called in "Call preprocessing subroutine" in the scenario, you can switch between a histogram search and a search using the OCR tool.

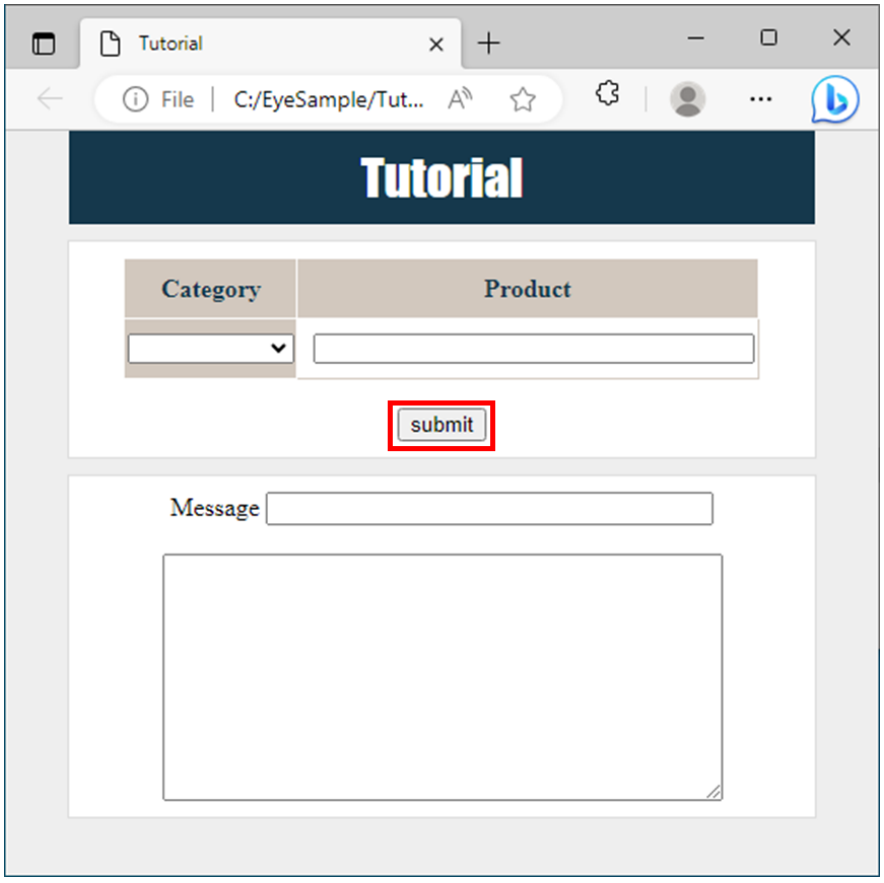


Figure 2.1-1. Tutorial page

[Files for the scenario to be created]

Create the files shown in Table 2.1-1 in the same folder on the computer where WinActor runs.

In this tutorial, the folder that stores the scenario is "C:\EyeSample."

Table 2.1-1. Description of each file

No.	Filename	Description
①	Eye_Hist_OCR_Sample.ums7	This is a WinActor scenario to be created in this tutorial. Here, the scenario with WinActor Ver.7 is used as an example.
②	Histogram_Search.json	This is a WinActorEye macro to be created in this tutorial. It runs a histogram search to search for the "submit" button on the Tutorial page.
③	Run_MSOCR.json	This is a WinActorEye macro to be created in this tutorial. It uses the OCR tool to get the string and coordinate information in the Tutorial page.
④	Search_submit.json	This is a WinActor Note macro to be created in this tutorial. It searches whether "submit" on the Tutorial page exists in the processing result of OCR tool.
⑤	Submit_Reference_Image.png	This is an image file to be created in this tutorial. In the macro of No.②, the position of the "submit" button will be searched from the targeted Tutorial page based on this image.
⑥	Tutorial.html	Copy this file from WinActor_Documents in advance.

In the scenario to be created in this tutorial, you will use the nodes and user libraries of WinActor as well as WinActor Note. For details of the nodes and user libraries in the scenario, see the materials No.2 and 3 in Table 4.1-1 in "4.1Reference materials." For details of WinActor Note, see the materials No.5 and 6 in Table 4.1-1 in "4.1Reference materials."

2.2. Scenario overview

The overview of how to create the scenario is as follows.

1. Creating the WinActorEye macro for the histogram search (see Section 2.5)
   A) Capturing the Tutorial page
   B) Creating and calling the "submit" reference image
   C) Calling the histogram search
   D) Converting the rectangle resource to the coordinate resource
2. Creating the WinActor scenario for the histogram search (see Section 2.6)
   A) Creating the outline of processing
   B) Creating the histogram search preprocessing subroutine
   C) Creating the subroutine for getting coordinate information and operating the mouse
3. Creating the WinActorEye macro for the OCR tool (see Section 2.7 and 2.8)
   A) Capturing the Tutorial page
   B) Calling the OCR tool
   C) Converting the rectangle resource to the coordinate resource
   D) Clearing the window and pasting from the clipboard
   E) Removing spaces from characters
   F) Searching for "submit"
4. Creating the WinActor scenario for the OCR tool (see section 2.9)
   A) Creating the OCR tool preprocessing subroutine

2.3. Note

You may not be able to find the targeted image even by following the steps in this manual. In that case, adjust the size and display font of the target Tutorial page or the parameters of each WinActorEye macro.

2.4. Preparation

Open the sample file "Tutorial.html" in Microsoft Edge before creating the scenario.
For the procedure, see the material No.2 in Table 4.1-1 in "4.1 Reference materials."

2.5. Creating the WinActorEye macro for the histogram search

For the operation of WinActorEye, see the material No.4 in Table 4.1-1 in "4.1 Reference materials."

The steps with "*" in the subsection title are subject to the macro recording.

Launch WinActorEye and select [View] > [WinActorEye edit macro] to display the "Edit macro" pane.

Click the "Record" icon on the Edit macro pane to enable the macro recording. (The state of the red frame ① in Figure 2.5-1 is the recording state. If it is not in the recording state as shown in the red frame ②, click the icon in the red frame.)

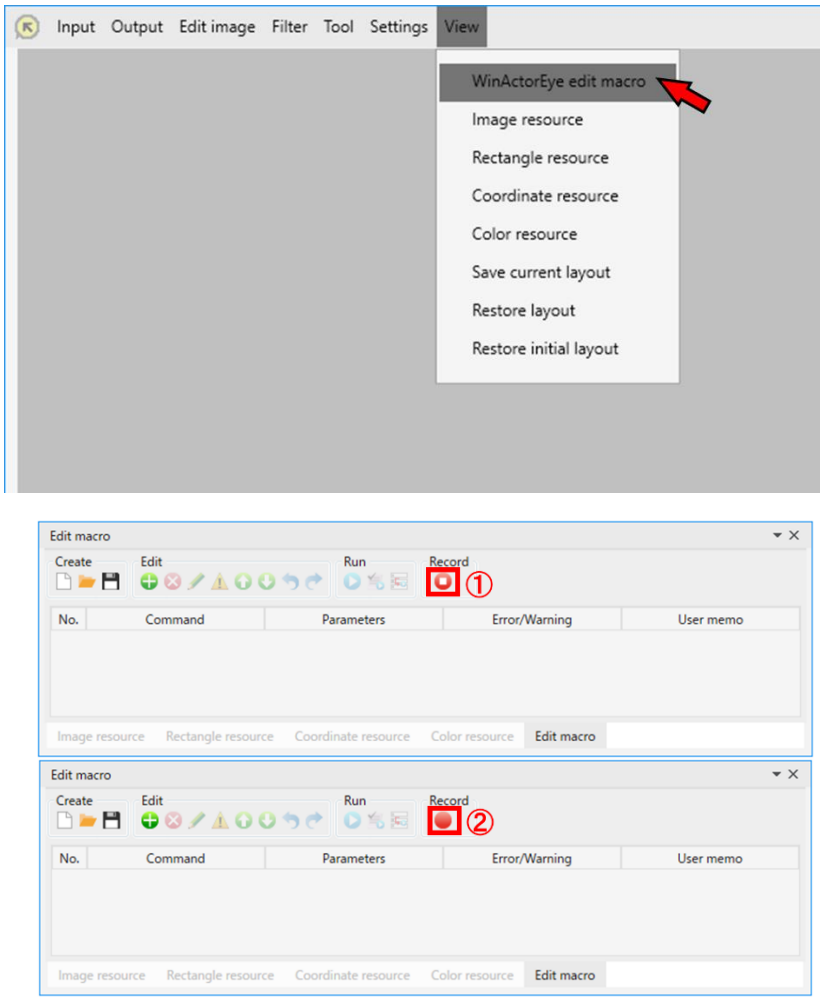


Figure 2.5-1. WinActorEye at the start of macro creation ①

Or, select the "Edit macro" tab in the resource area to display the "Edit macro" pane, and click the "Record" icon on the Edit macro pane. (Figure 2.5-2)

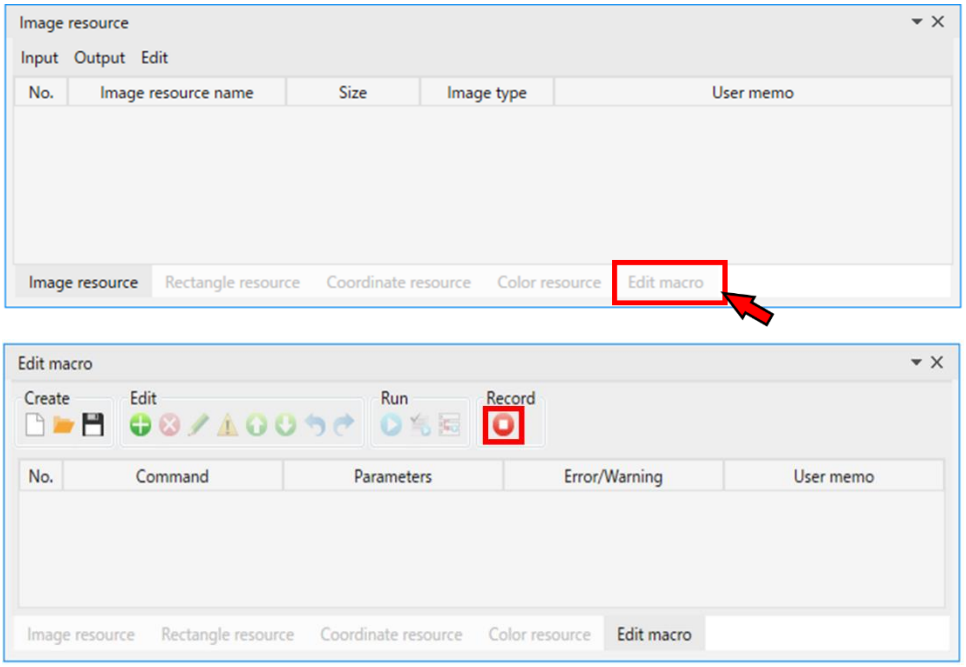


Figure 2.5-2. WinActorEye at the start of macro creation ②

2.5.1. Capturing the Tutorial page*

Select [Input] > [Snapshot] to display the "Snapshot" property window.

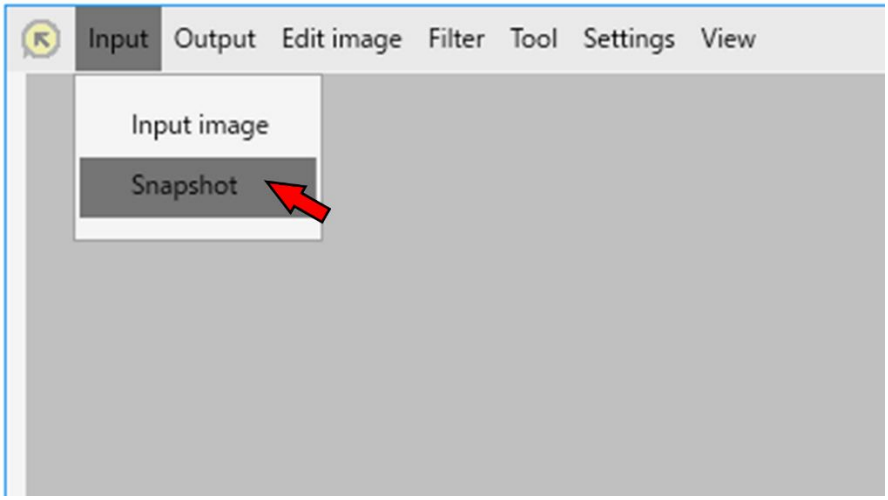


Figure 2.5-3. Selecting "Snapshot" from the menu

Click ① of Figure 2.5-4 and then click Microsoft Edge with the Tutorial page displayed. The window title is set as shown in ②. Then, click the button ③.

After this step, click the "Record" icon in the "Edit macro" pane in Figure 2.5-1. WinActorEye at the start of macro creation ① to disable the macro recording.

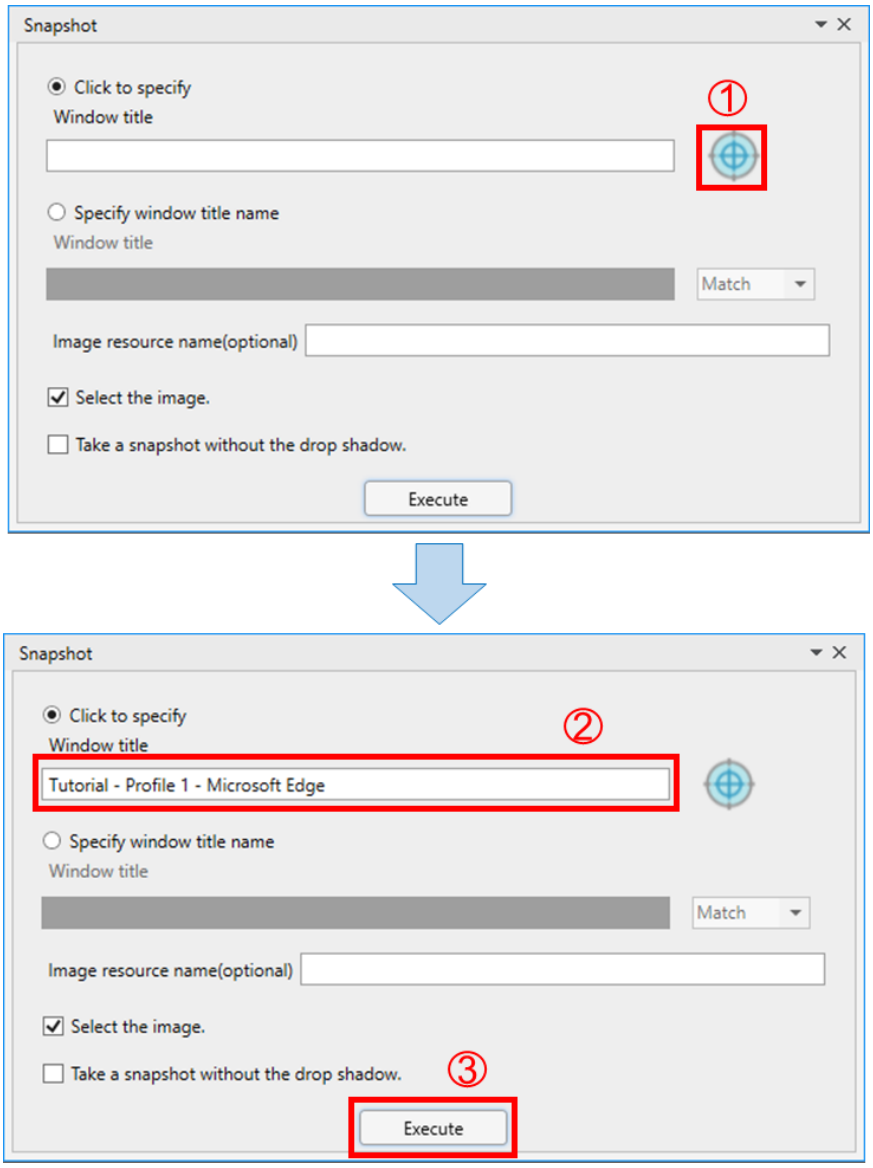


Figure 2.5-4. Selecting the "Tutorial" page window

The window of "Tutorial" page will be displayed on WinActorEye as shown in Figure 2.5-5.

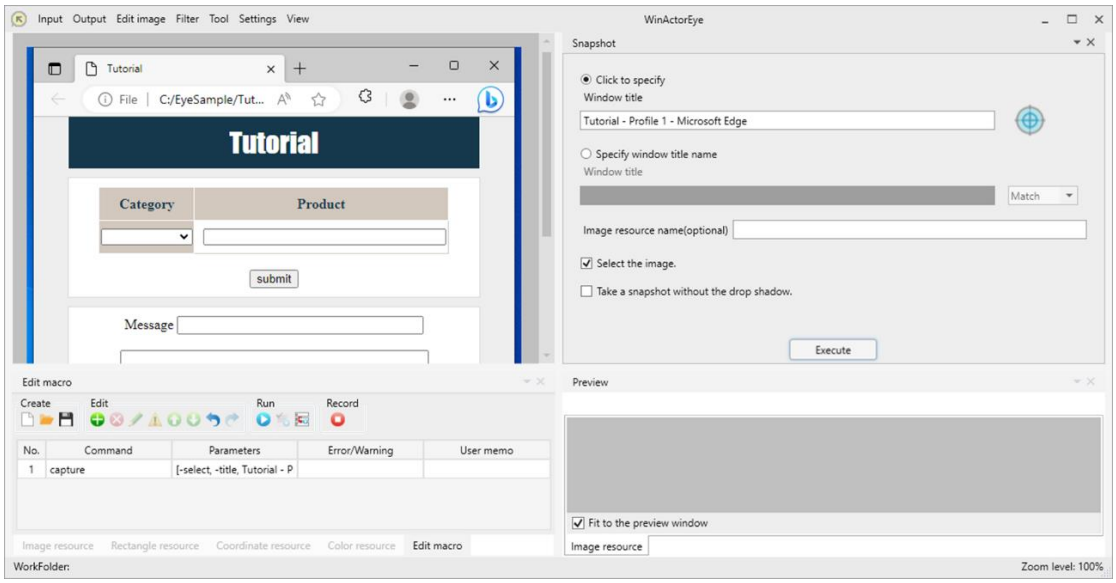


Figure 2.5-5. WinActorEye window

2.5.2. Creating the "submit" button reference image

To make the submit button as a target for the search, create a reference image for the submit button.

On the WinActorEye window, click and hold the left mouse button on the upper left of the submit button, move the mouse to the lower right of the submit button, and then release the left mouse button. You can select a range with this operation.

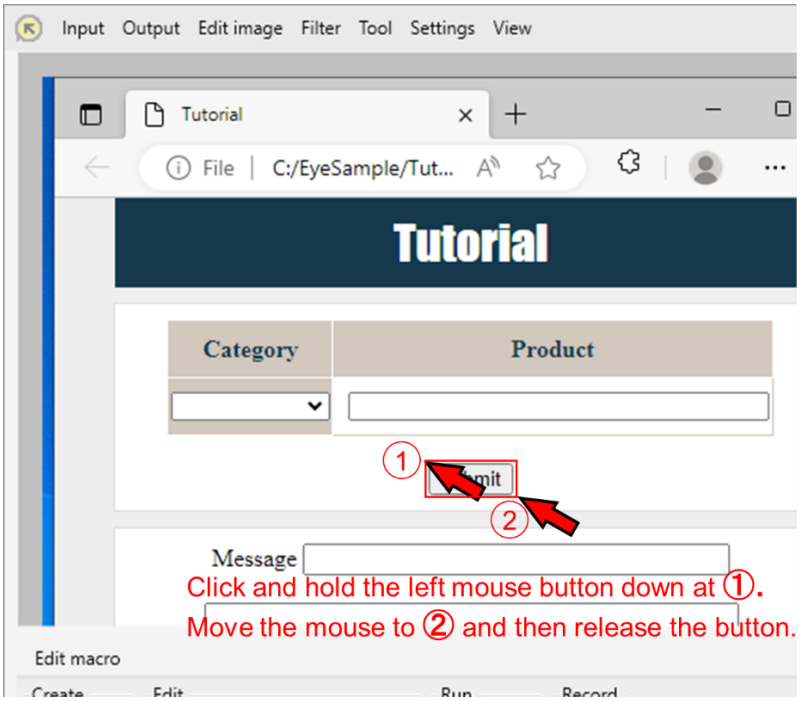


Figure 2.5-6. Selecting the range of the submit button

Select [Edit image] > [Trim] to display the "Trim" property window.

Click the ▽ button on the right edge of ① and select "Tutorial - Profile 1 - Microsoft Edge." Select "MAIN" in ②, enter "Submit" in ③, and click the Execute button of ④.

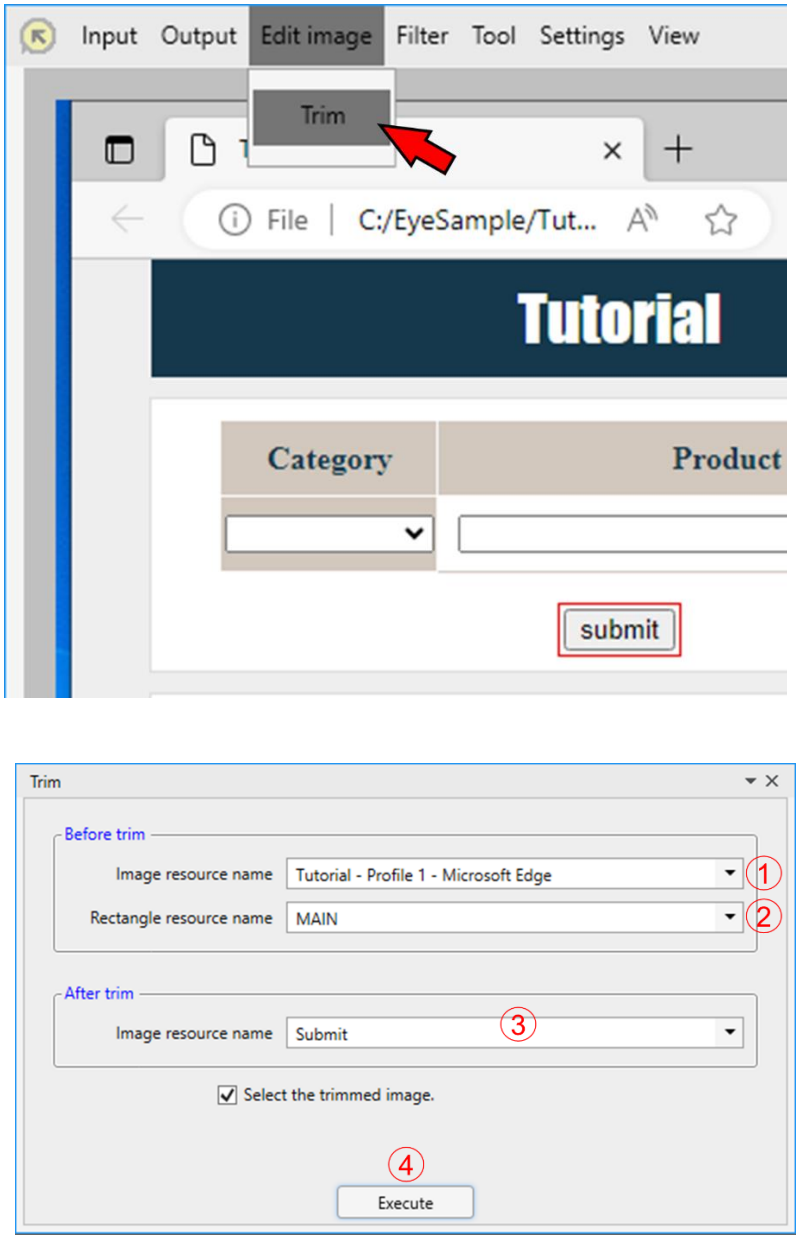
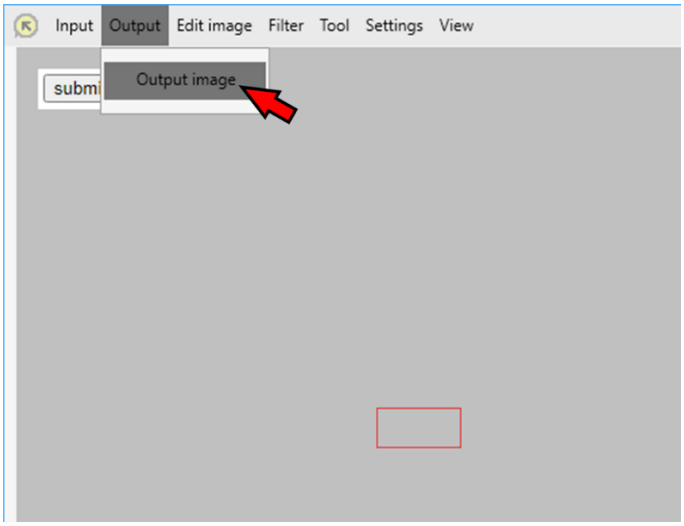
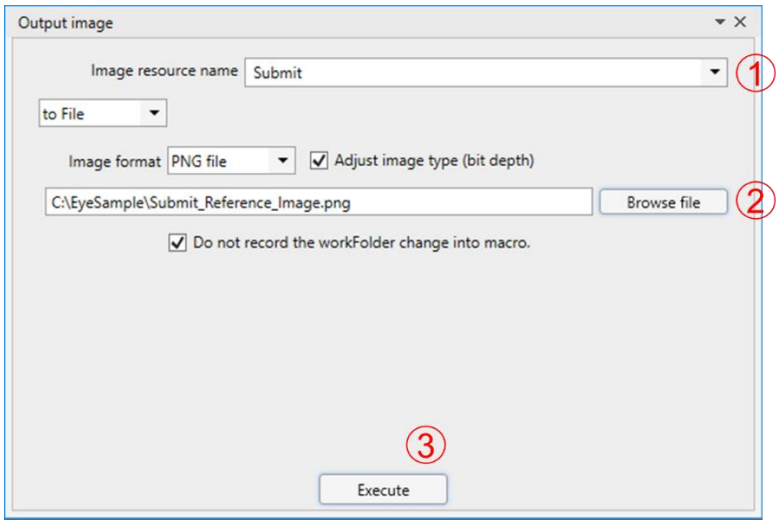


Figure 2.5-7. "Trim" property window

Next, select [Output] > [Output image] to display the "Output image" property window. Click the button of ① and select "Submit." Click "Browse file" of ②, and in the "Save" window, select "C:\EyeSample," enter "Submit_Reference_Image" for the filename, and click the Save button. Then, click the Execute button of ③.





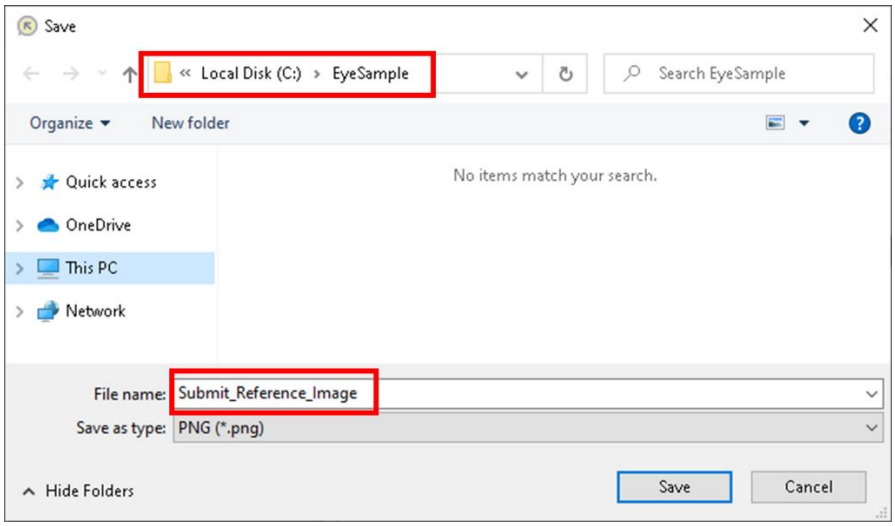


Figure 2.5-8. Outputting the image

2.5.3. Calling the "submit" button reference image*

Click the "Record" icon in the "Edit macro" pane in Figure 2.5-1. WinActorEye at the start of macro creation ① to enable the macro recording.

Select [Input] > [Input image] to display the "Input image" property window.

Click "Browse file" of ① and select "C:\EyeSample\Submit_Reference_Image.png." Check the box of "Do not record the workFolder change into macro." (②), and then click the Execute button of ③.

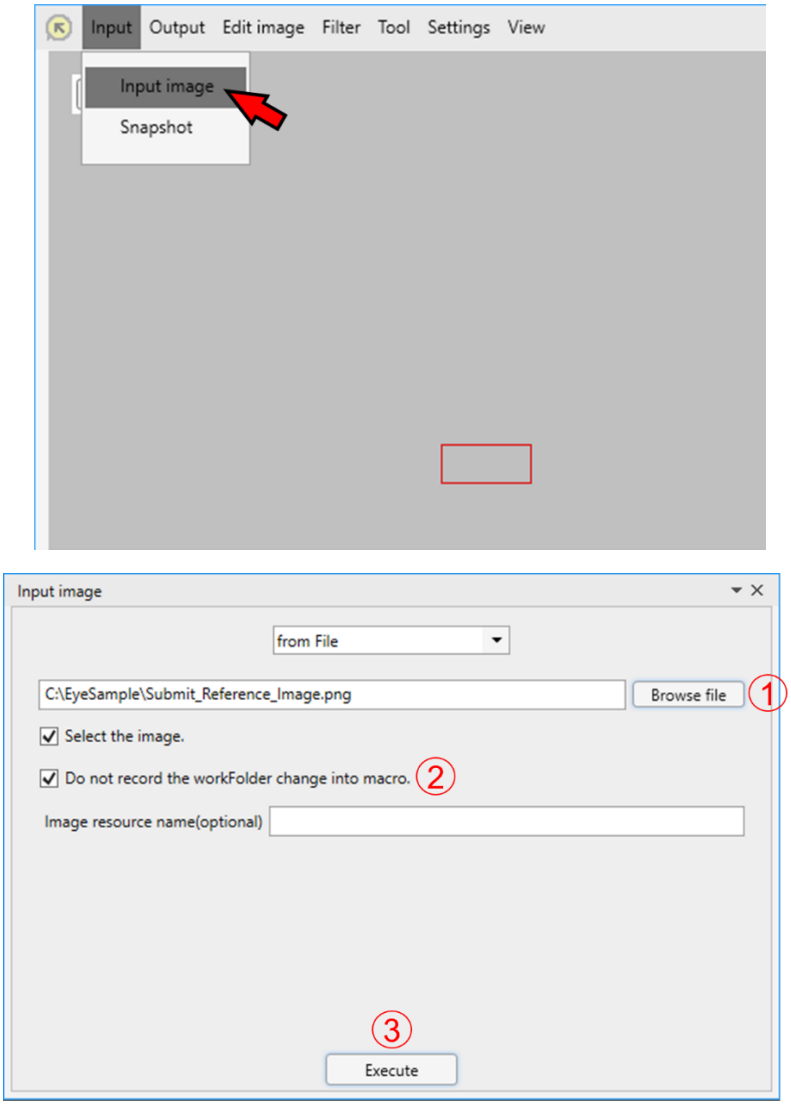


Figure 2.5-9. "Input image" property window

2.5.4. Calling the histogram search*

Select [Filter] > [Histogram search] to display the "Histogram search" property window. Click each button of ① and ②, and select "Tutorial - Profile 1 – Microsoft Edge" for ① and "Submit_Reference_Image.png" for ②. Enter "hist" for both ③ and ④, and then click the Execute button of ⑤. Based on the settings above, the location of the target image will be searched from the "Tutorial - Profile 1 - Microsoft Edge" image resource according to the information of "Submit_Reference_Image.png," and the information will be output to the rectangle resource specified in ④.

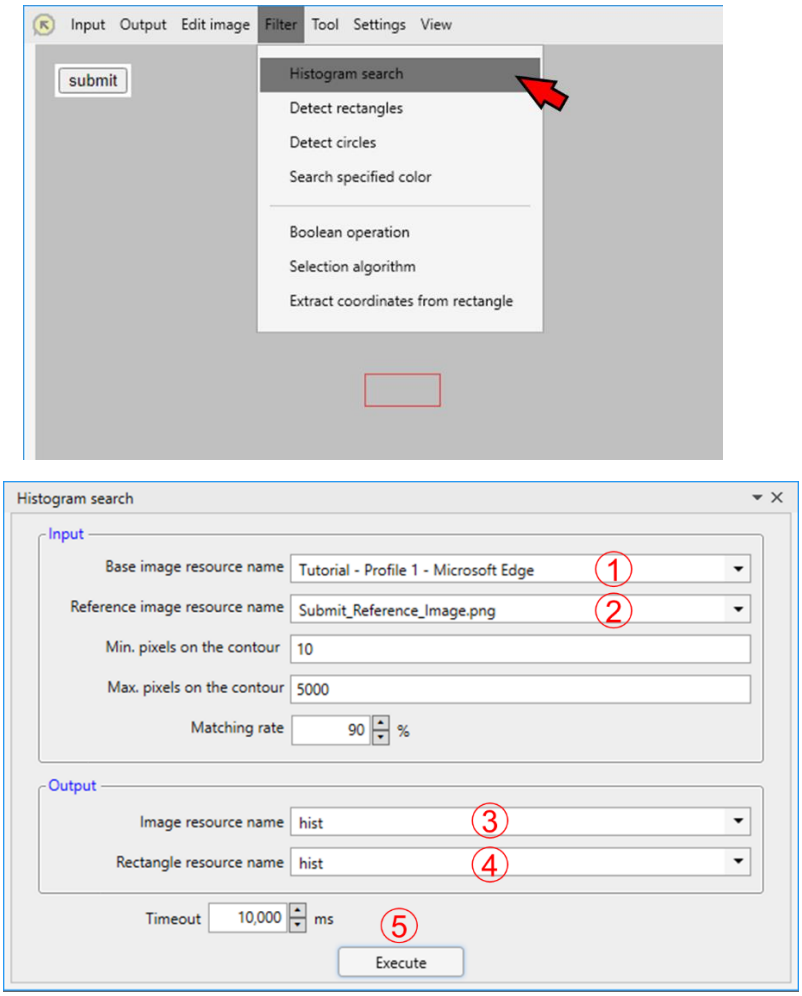


Figure 2.5-10. "Histogram search" property window

2.5.5. Converting the rectangle resource to the coordinate resource*

Lastly, convert the rectangle resource of the targeted image to the coordinate resource to get the coordinate information of the targeted image.

Select [Filter] > [Extract coordinates from rectangle] to display the "Extract coordinates from rectangle" property window. Click the button of ① and select "hist." Enter "Coordinate resource output" in ②, and then click the Execute button of ③.

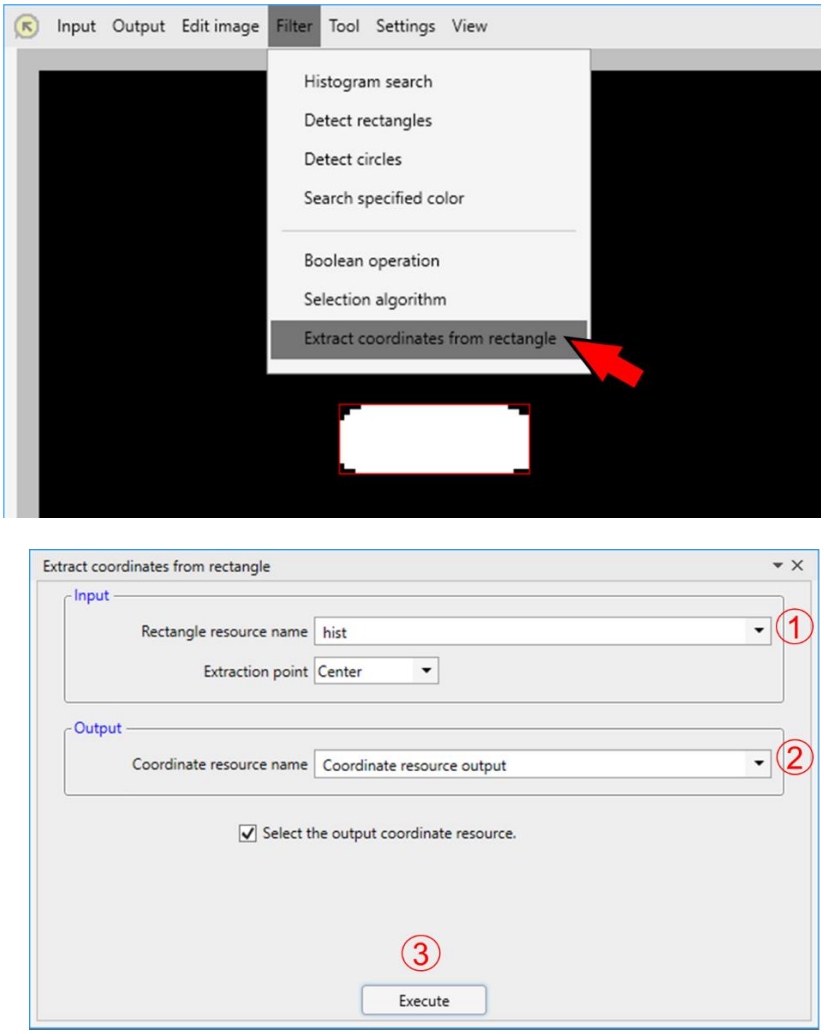


Figure 2.5-11. "Extract coordinates from rectangle" property window

Next, save the created macro.

In the "Edit macro" pane, click the "Record" icon ① and then click the Save icon ② to save the created macro to "C:\EyeSample\Histogram_Search.json."

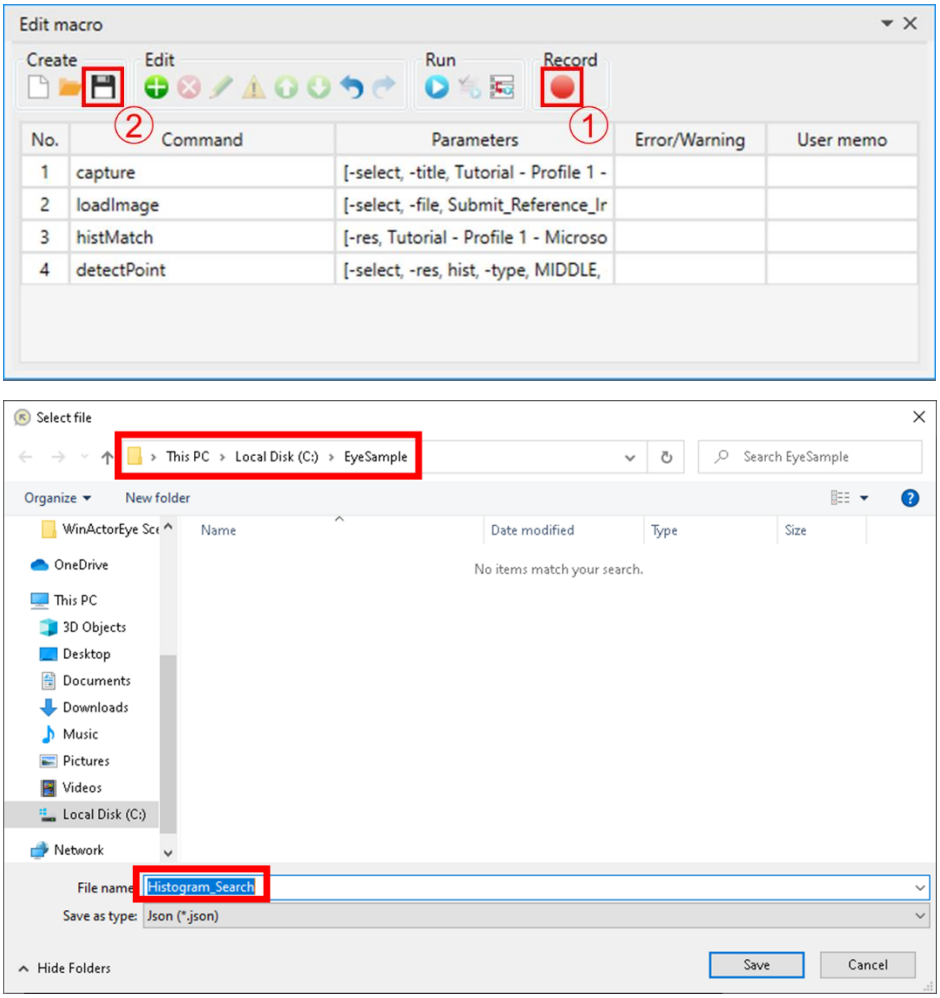


Figure 2.5-12. Saving the macro

2.5.6. Checking the operations

Check the operations of the macro created so far.

Click [Settings] > [Clear all resources] to initialize WinActorEye.

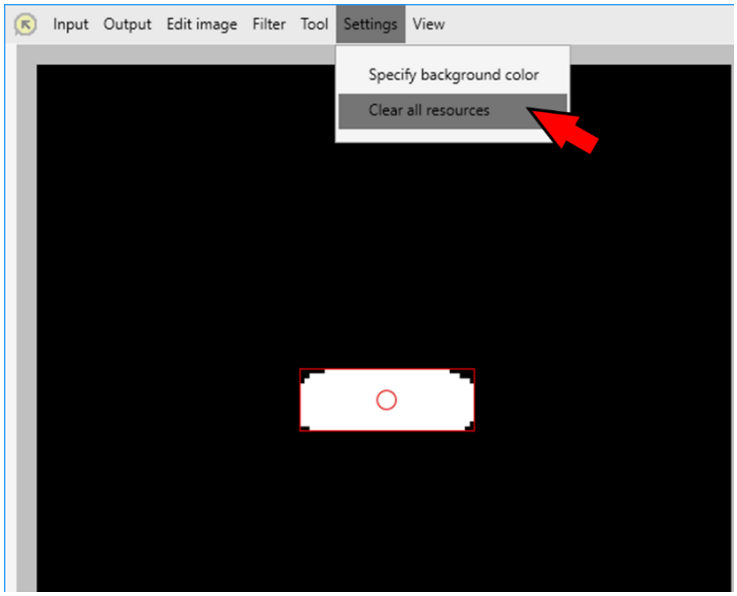


Figure 2.5-13. Clearing all resources

Next, open the saved macro. Click the "Open" icon ① and select "C:\EyeSample\Histogram_Search.json." Click the "Run all" icon ② to run the macro.

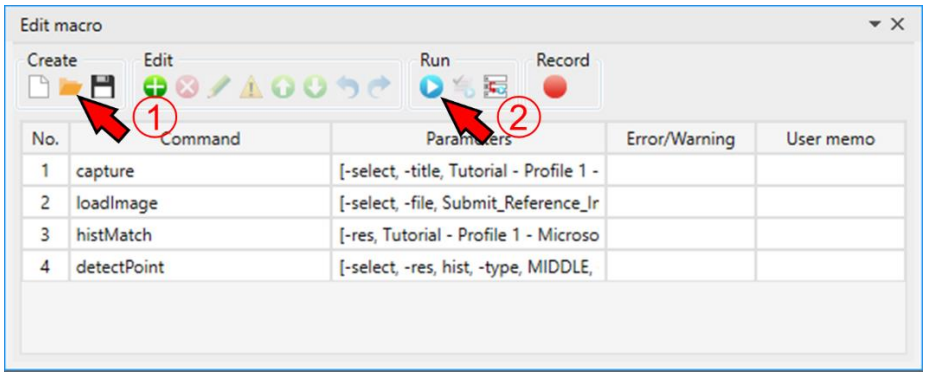
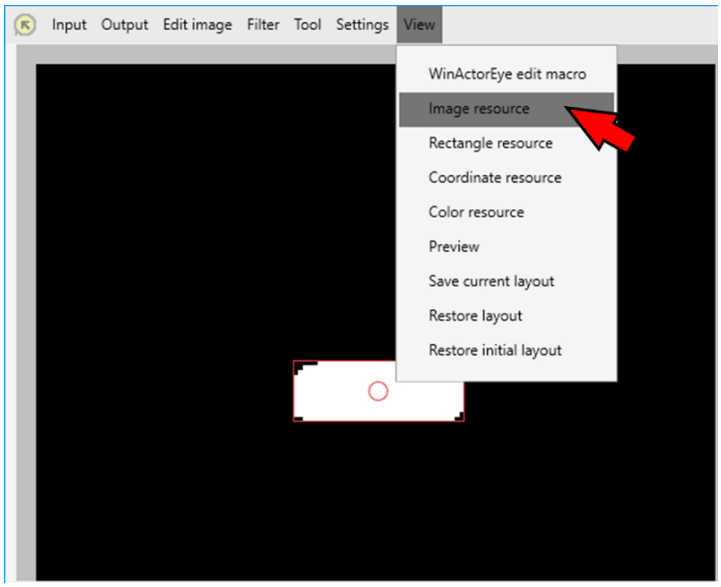


Figure 2.5-14. Opening the macro

Check the result of running the macro. Select [View] > [Image resource] to display the Image resource pane.
Click No.1 in the Image resource pane.



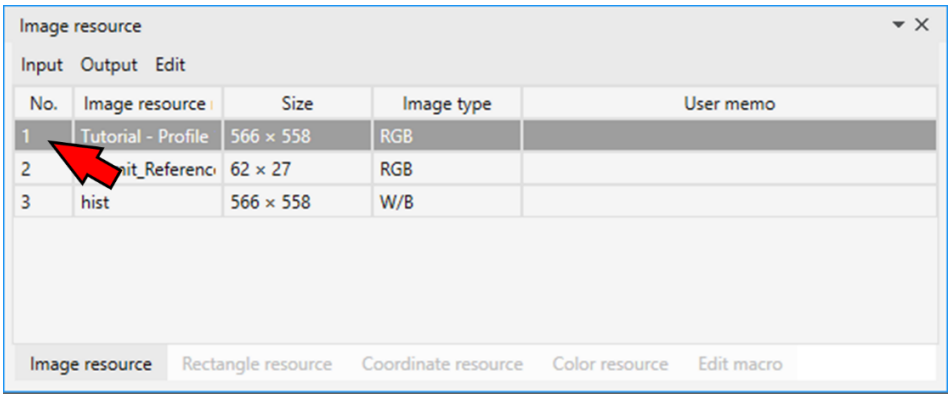


Figure 2.5-15. Selecting the image resource to be displayed ①

Or, select the "Image resource" tab in the resource area to display the Image resource pane, and then click No.1 in the Image resource pane. (Figure 2.5-16).

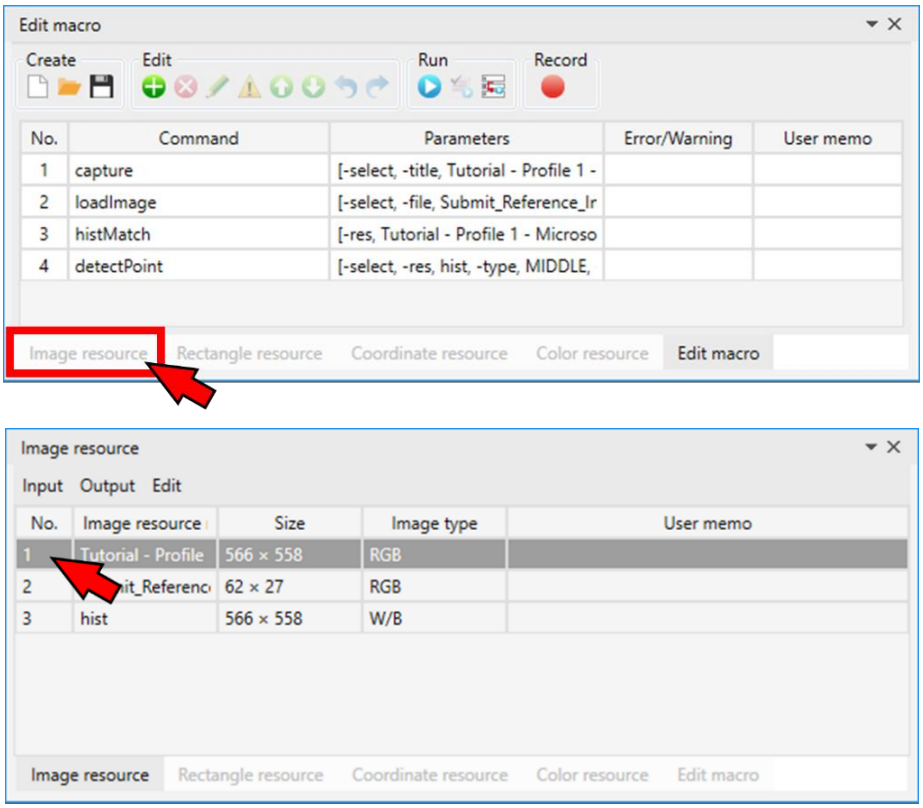


Figure 2.5-16. Selecting the image resource to be displayed ②

The operations are successful if a red circle is displayed on the submit button in the WinActorEye window as shown in Figure 2.5-17.

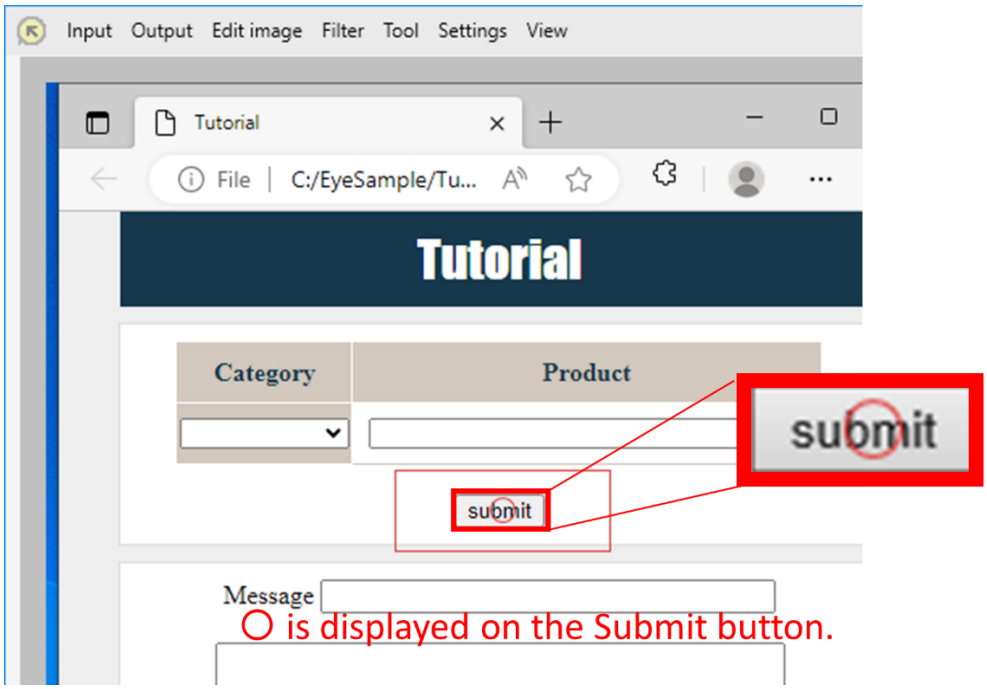


Figure 2.5-17. Expected result

2.6. Creating the WinActor scenario for the histogram search

In WinActor, create the scenario for clicking the submit button on the Tutorial page by using the library that runs the macro created in Section 2.5.

2.6.1. Creating the outline of processing

Create the outline of the processing to call the individual processing performed by the WinActorEye macro as a preprocessing subroutine and the processing to call the common processing of mouse clicking based on the coordinate information acquired from WinActorEye as a post-processing subroutine.

First, fix the position and size of the window of the Tutorial page so that the operation will be stable.

Drag and drop "Window_SetWindowRectangle" from the Library tab.

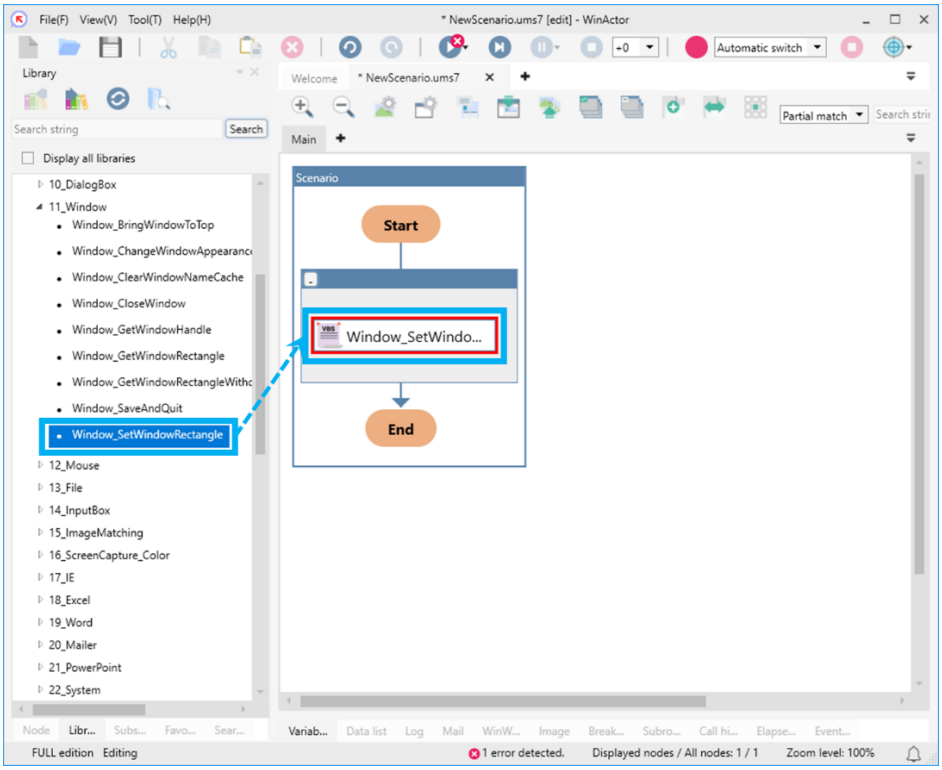


Figure 2.6-1. Placing the "Window_SetWindowRectangle" library

Set the property values according to the table below.

Table 2.6-1. Property settings

Item	Setting value	Remarks
WinID name	Tutorial - Profile 1 - Microsoft Edge	Click the icon in the blue frame in Figure 2.6-2 and select the window of the Tutorial page.
Upper_left(x)	Value=>0	Select "Value=>" and enter the value to the right of "Value=>."
Upper_left(y)	Value=>0	
Horizontal_width	Value=>566	
Vertical_height	Value=>558	

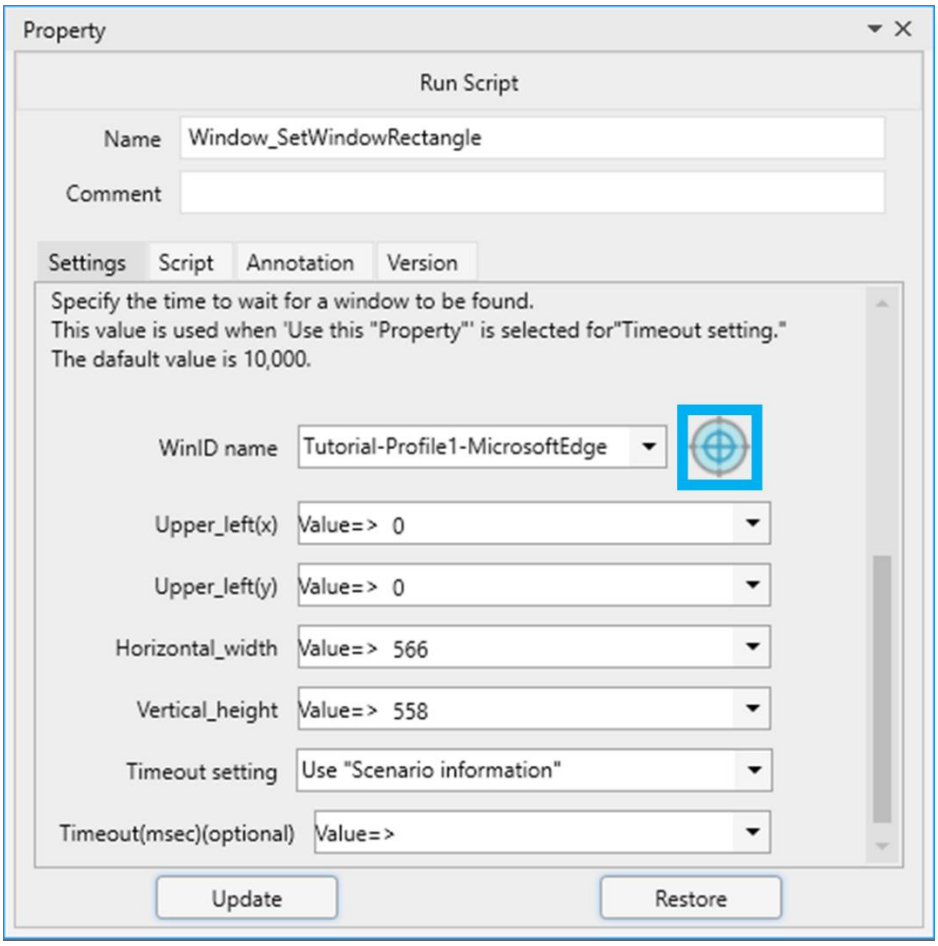


Figure 2.6-2. Setting the WinID name

Save the scenario once at this point.

To save the scenario, select [File] > [Save as], set "C:\EyeSample" for the destination and "Eye_Hist_OCR_Sample" for the filename.

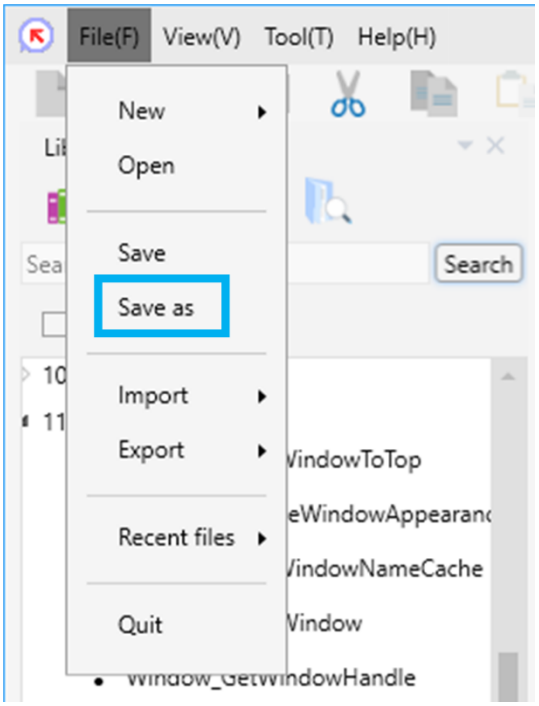


Figure 2.6-3. Saving the scenario

Next, bring the window of the Tutorial page to the front so that it can be captured in WinActorEye.

Drag and drop "Window_BringWindowToTop" from the Library tab.

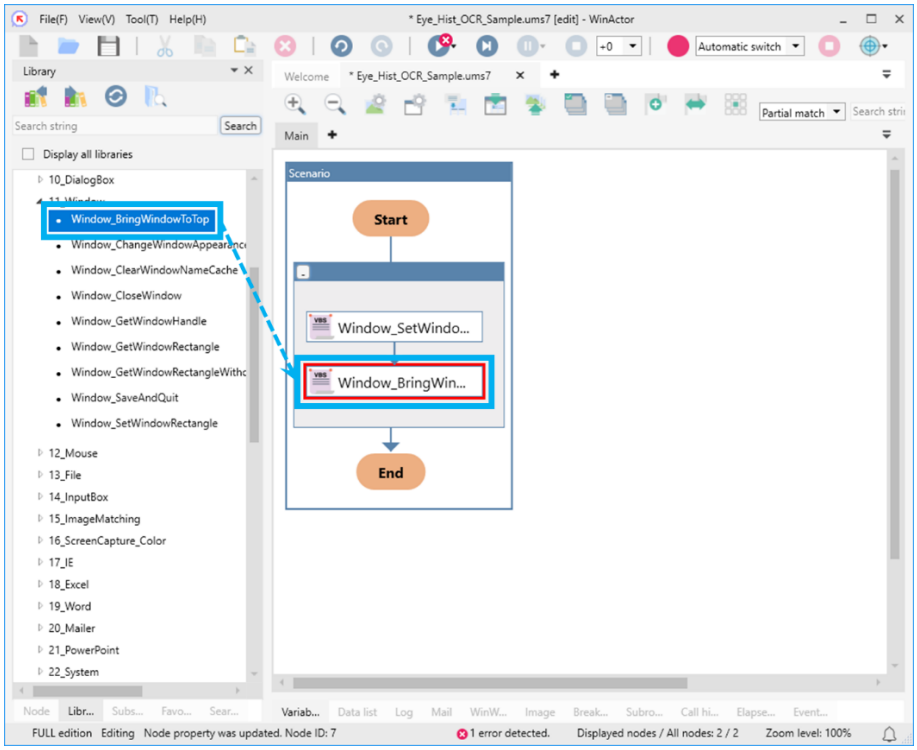


Figure 2.6-4. Placing the "Window_BringWindowToTop" library

Set the property value according to the table below.

Table 2.6-2. Property setting

Item	Setting value	Remarks
WinID name	Tutorial - Profile 1 - Microsoft Edge	Select from the drop-down list.

Next, call the subroutine for preprocessing.

Drag and drop "Call Subroutine" from the Node tab.

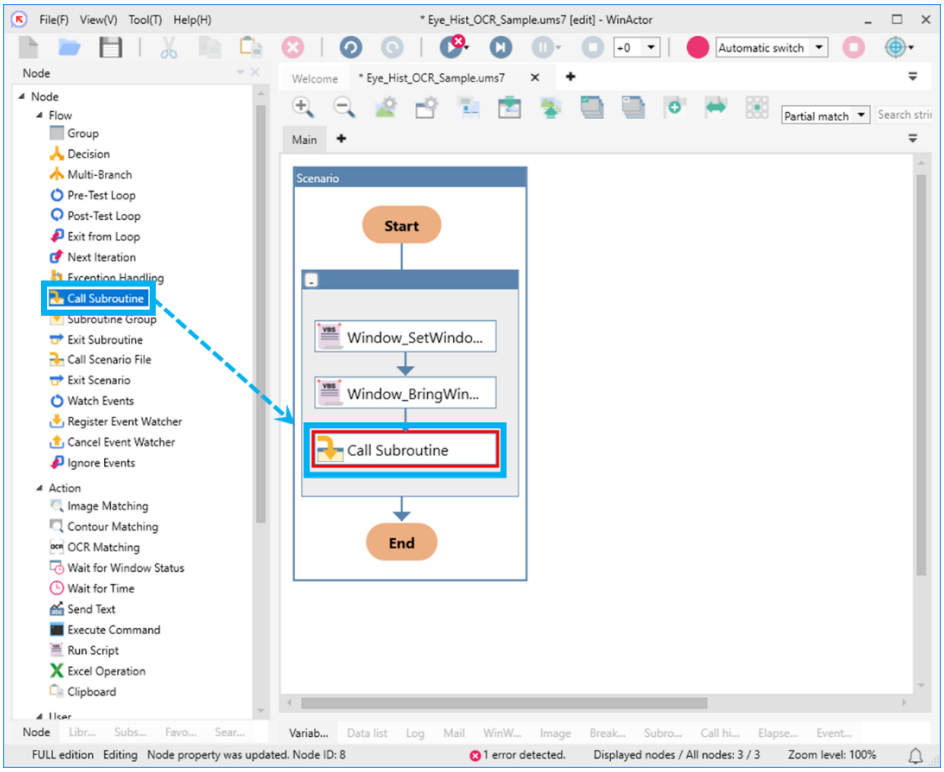


Figure 2.6-5. Placing the "Call Subroutine" node

Add a decision group so that the processing result of the subroutine for preprocessing can be judged.

Drag and drop "Decision" from the Node tab.

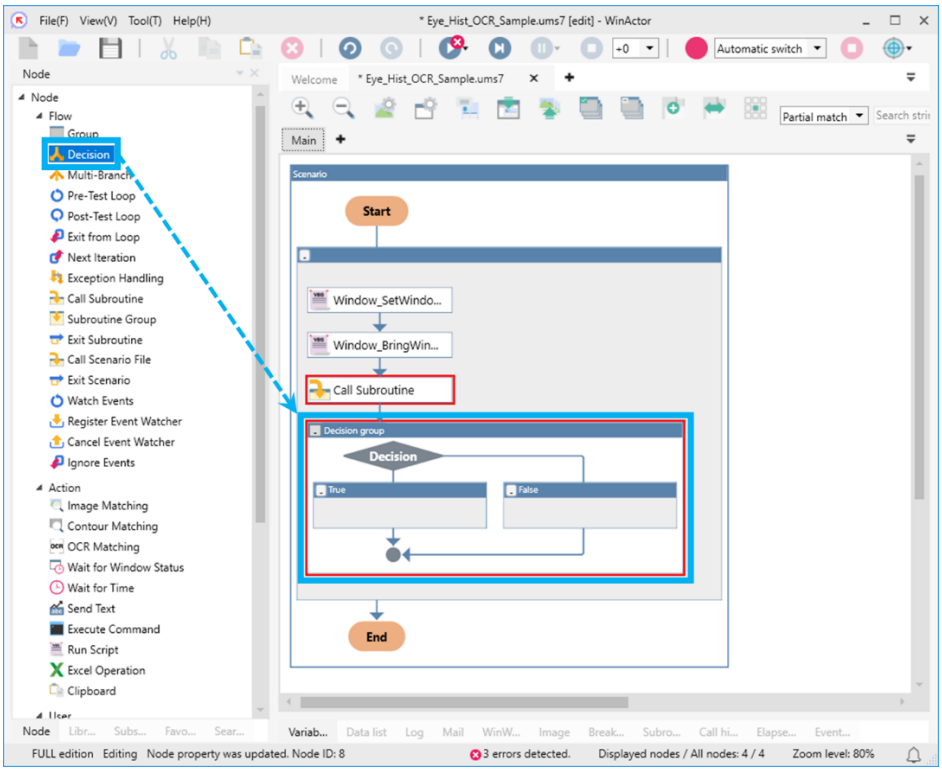


Figure 2.6-6. Placing the "Decision" node

Click the ‘Edit’ button of the property window and set the conditional expression according to the table below.

Table 2.6-3. Conditional expression settings

Item	Left-hand side	Comparison operator	Right-hand side
Setting value	Processing_result	is equal to	Value=>OK
Remarks	-	-	Select "Value=>" and enter the value to the right of "Value=>."

Call the subroutine for post-processing.

Drag and drop "Call Subroutine" from the Node tab.

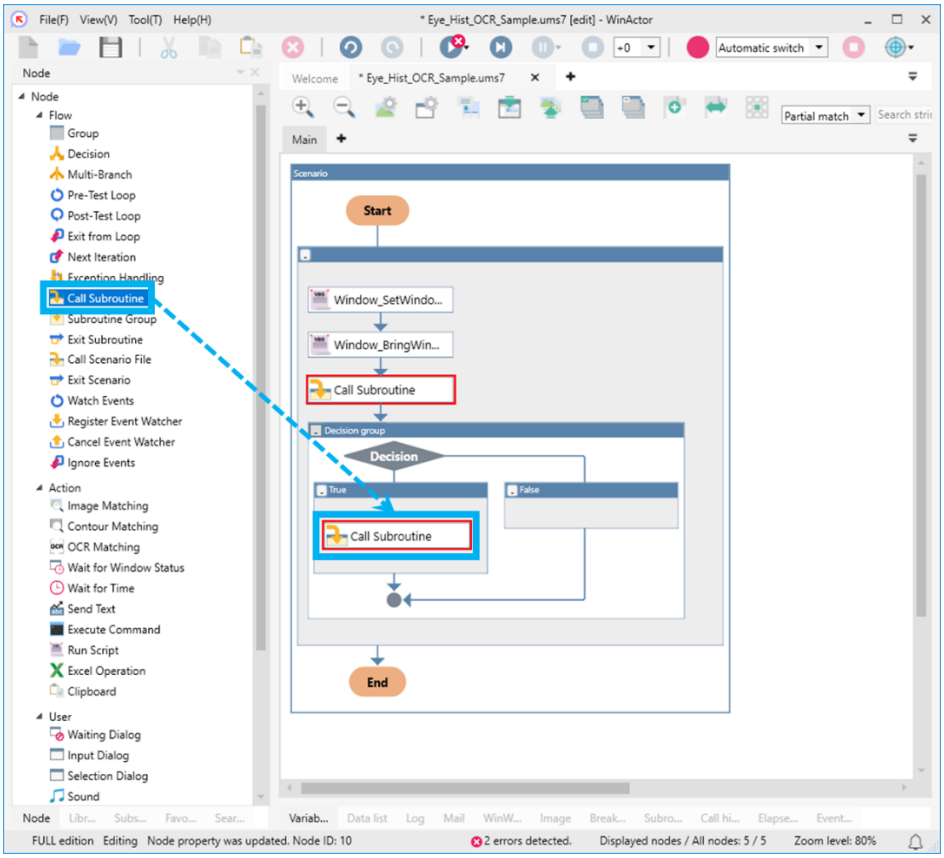


Figure 2.6-7. Placing the "Call Subroutine" node

2.6.2. Creating the histogram search preprocessing subroutine

Create the processing specific to the histogram search as a preprocessing subroutine.

Drag and drop "Subroutine Group" from the Node tab.

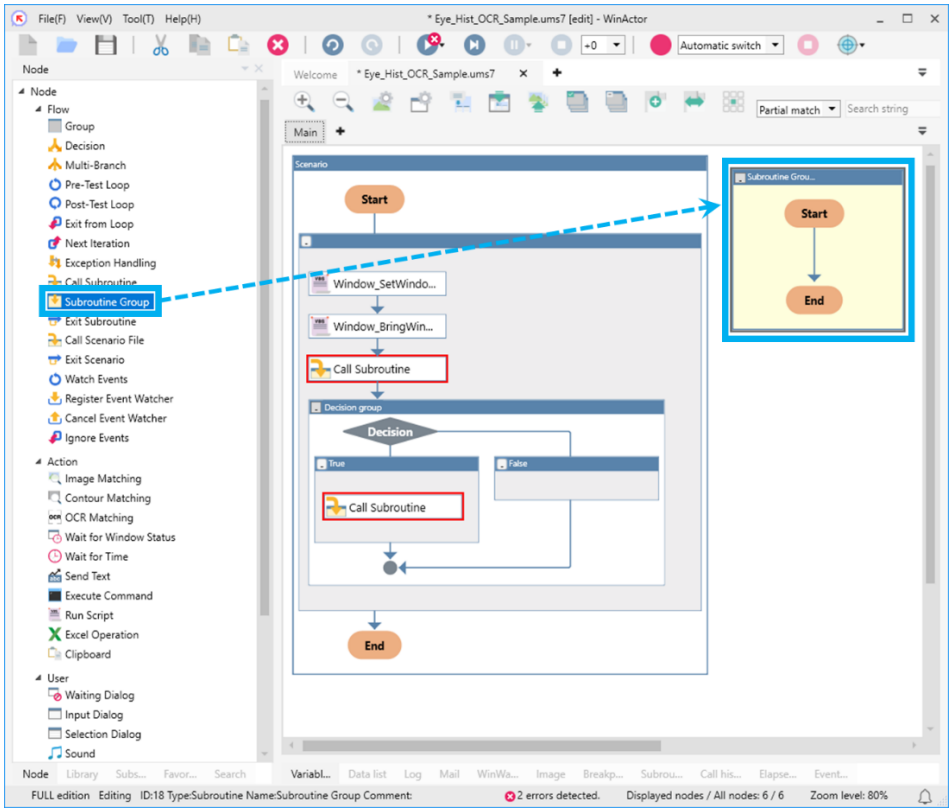


Figure 2.6-8. Placing the "Subroutine Group" node

Set the property value according to the table below.

Table 2.6-4. Property setting

Item	Setting value	Remarks
Name	Histogram search preprocessing	-

Add the processing for reading and running the macro for the histogram search.

Drag and drop "Eye_ReadAndRunMacro" from the Library tab.

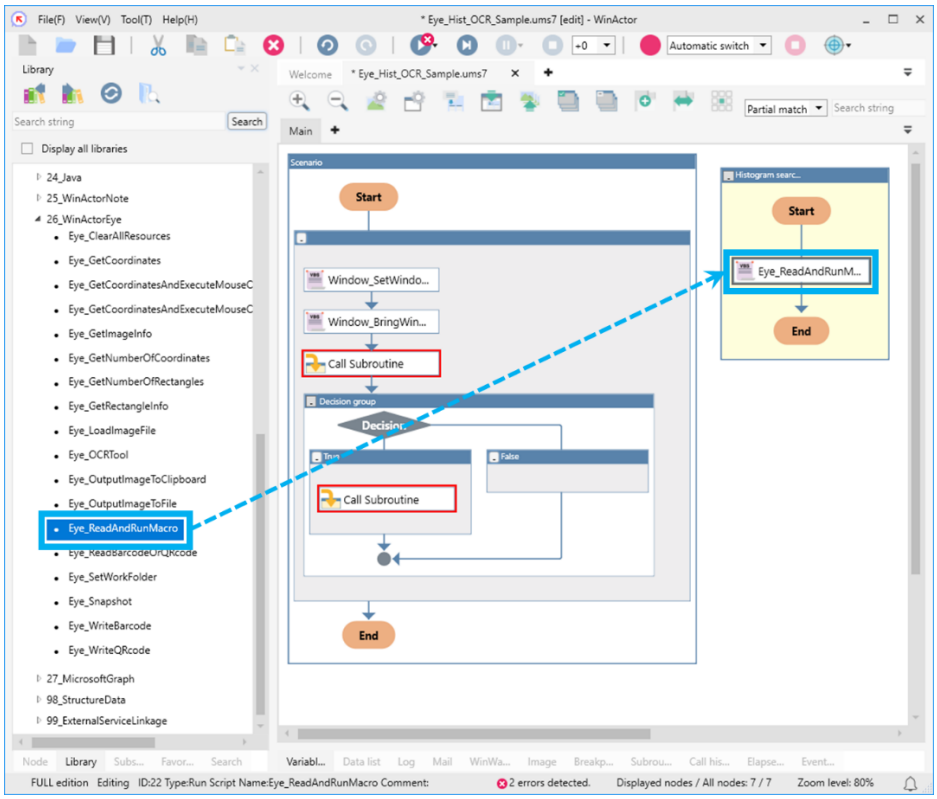


Figure 2.6-9. Placing the "Eye_ReadAndRunMacro" library

Set the property value according to the table below.

Table 2.6-5. Property setting

Item	Setting value	Remarks
Macro_filename	Value=>Histogram_Search.json	Select "Value=>" and enter the value to the right of "Value=>."

2.6.3. Creating the subroutine for getting coordinate information and operating the mouse

As a post-processing subroutine, create the processing for operating the mouse based on the coordinate information acquired from WinActorEye.

Drag and drop "Subroutine Group" from the Node tab.

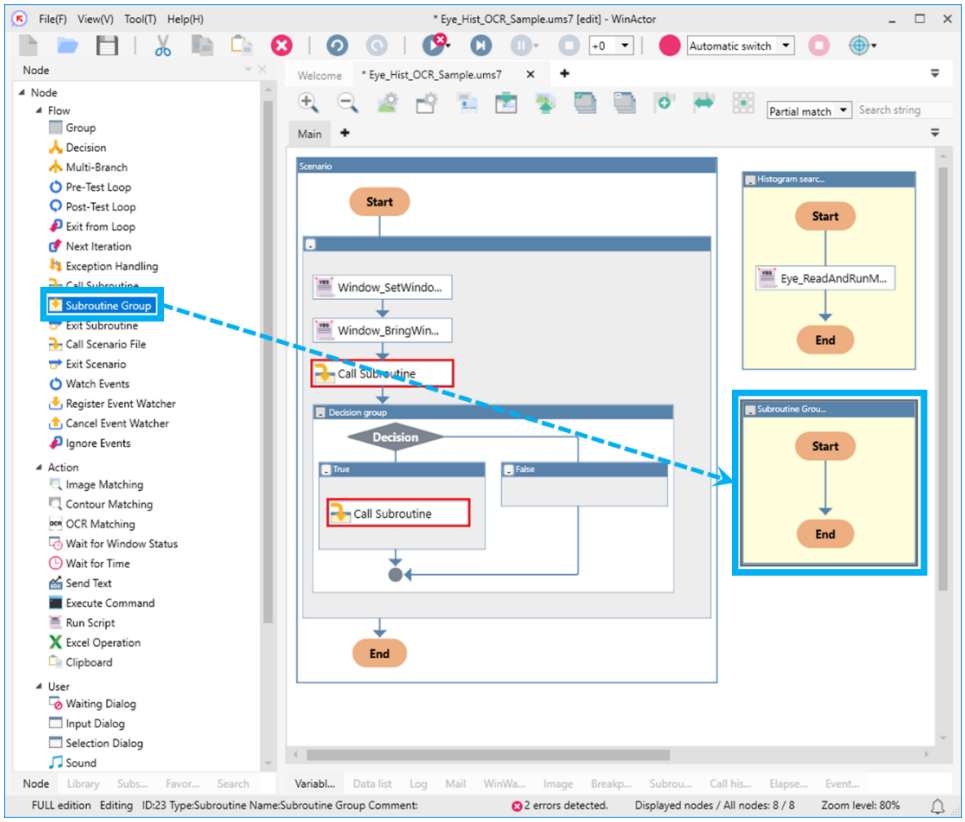


Figure 2.6-10. Placing the "Subroutine Group" node

Set the property value according to the table below.

Table 2.6-6. Property setting

Item	Setting value	Remarks
Name	Get coordinate info and operate mouse	-

Add the processing for getting the coordinate information after running the macro for the histogram search.

Drag and drop "Eye_GetCoordinates" from the Library tab.

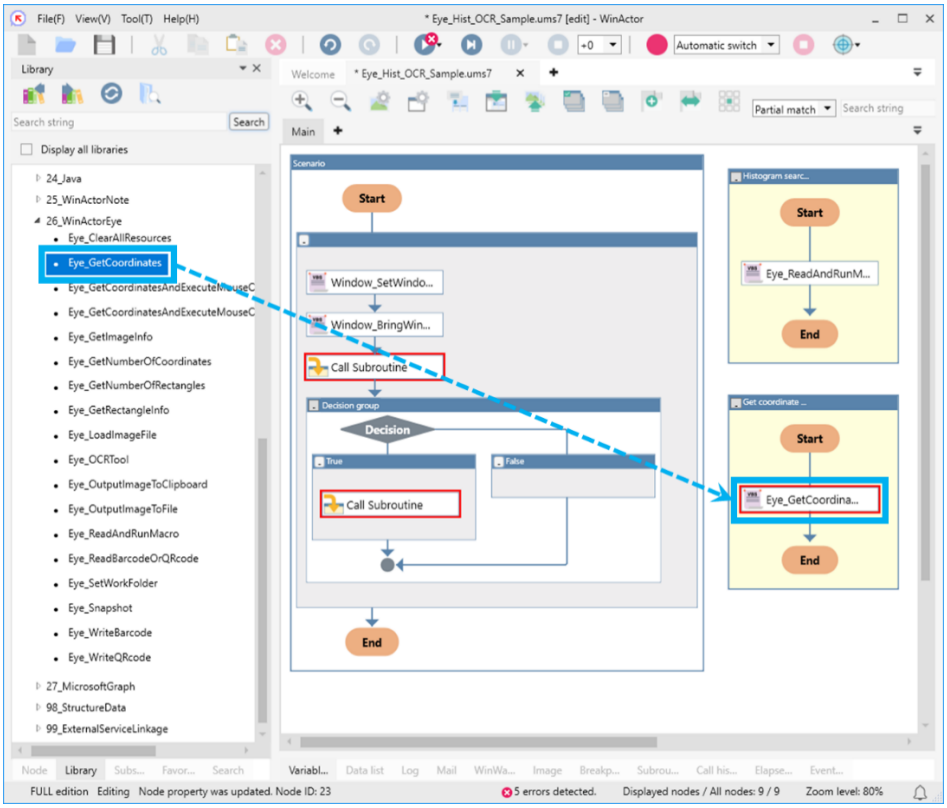


Figure 2.6-11. Placing the "Eye_GetCoordinates" library

Set the property values according to the table below.

Table 2.6-7. Property settings

Item	Setting value	Remarks
Coordinate_resource_name	Coordinate_resource_name	-
Index	Index	-
Variable_for_x_coordinate	x-coordinate	-
Variable_for_y_coordinate	y-coordinate	-

Add the processing for getting the position information of the window of the Tutorial page.

Drag and drop "Window_GetWindowRectangle" from the Library tab.

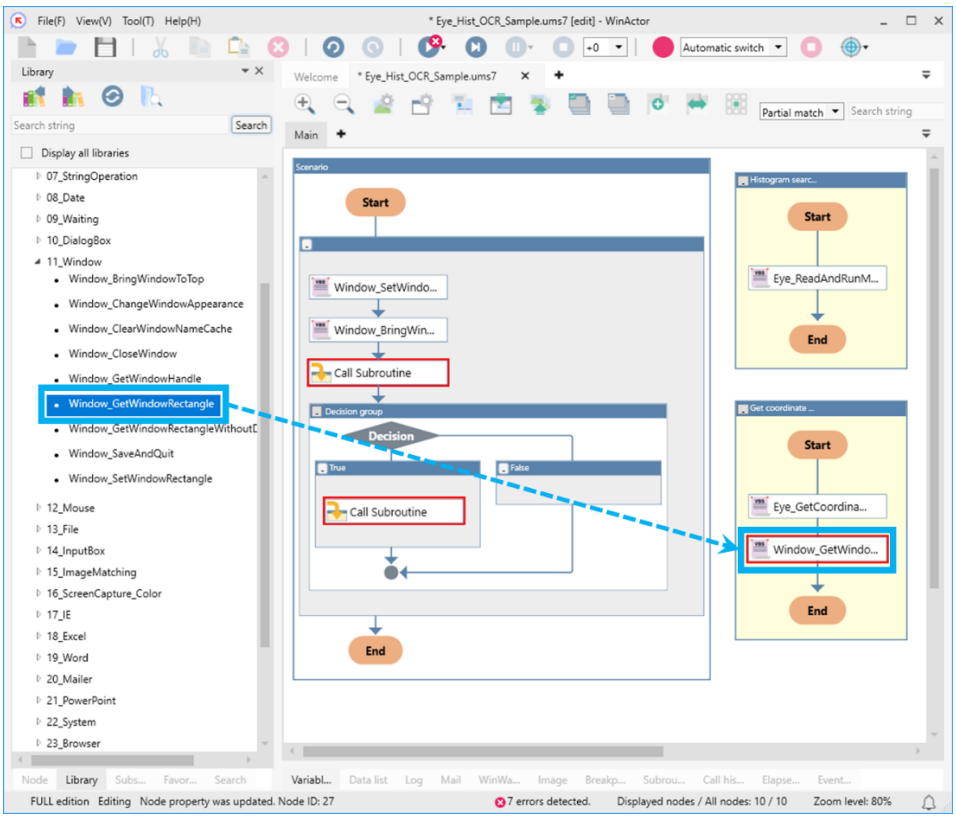


Figure 2.6-12. Placing the "Window_GetWindowRectangle" library

Set the property values according to the table below.

Table 2.6-8. Property settings

Item	Setting value	Remarks
WinID name	Tutorial - Profile 1 - Microsoft Edge	Select from the drop-down list.
Upper_left(x)	Window_position_x	-
Upper_left(y)	Window_position_y	-
Horizontal_width	Unused_1	-
Vertical_height	Unused_2	-

For the coordinate information after running the macro for the histogram search processing and the position information of the window of the Tutorial page, add each of the x-coordinate and y-coordinate and calculate the coordinates for operating the mouse.

Drag and drop two "Four Arithmetic Operations" from the Node tab.

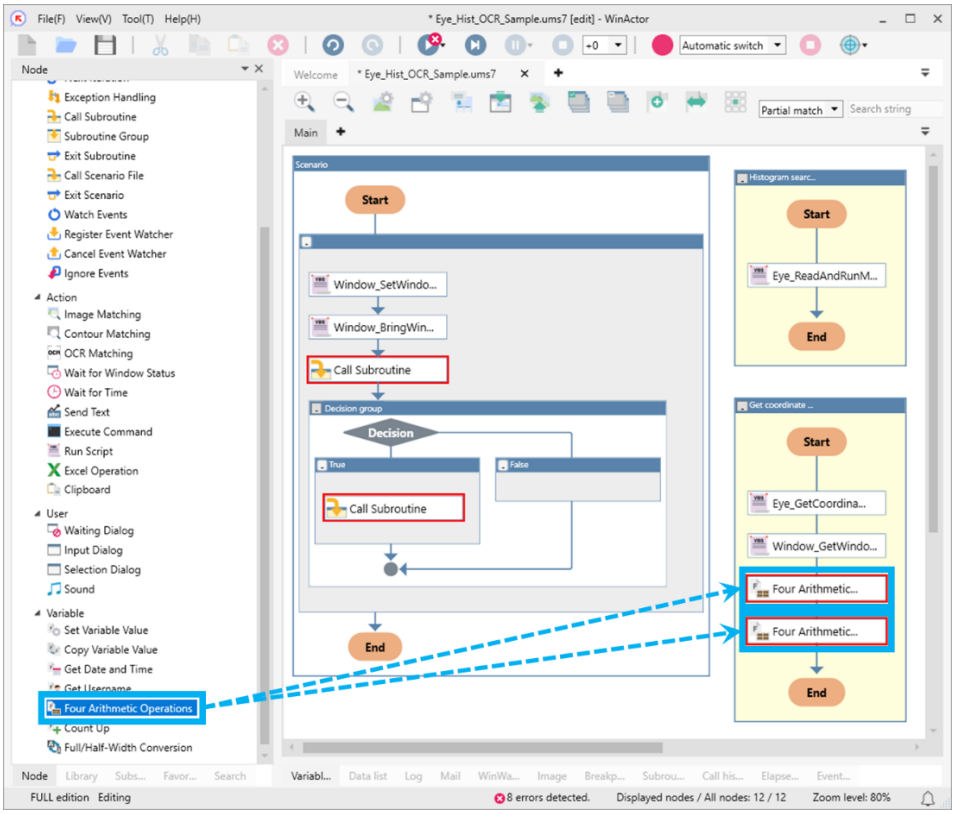


Figure 2.6-13. Placing two "Four Arithmetic Operations" nodes

Set the property values according to the tables below.

Table 2.6-9. Property settings (The first "Four Arithmetic Operations")

Item	Setting value	Remarks
Calculation result	x-coordinate	Select from the drop-down list.
Left side of +	x-coordinate	Select from the drop-down list.
Operator	+	Select from the drop-down list.
Right side of +	Window_position_x	Select from the drop-down list.

Table 2.6-10. Property settings (The second "Four Arithmetic Operations")

Item	Setting value	Remarks
Calculation result	y-coordinate	Select from the drop-down list.
Left side of +	y-coordinate	Select from the drop-down list.
Operator	+	Select from the drop-down list.
Right side of +	Window_position_y	Select from the drop-down list.

Add the processing of moving the mouse pointer to the acquired coordinates.

Drag and drop "Mouse_MovePointer" from the Library tab.

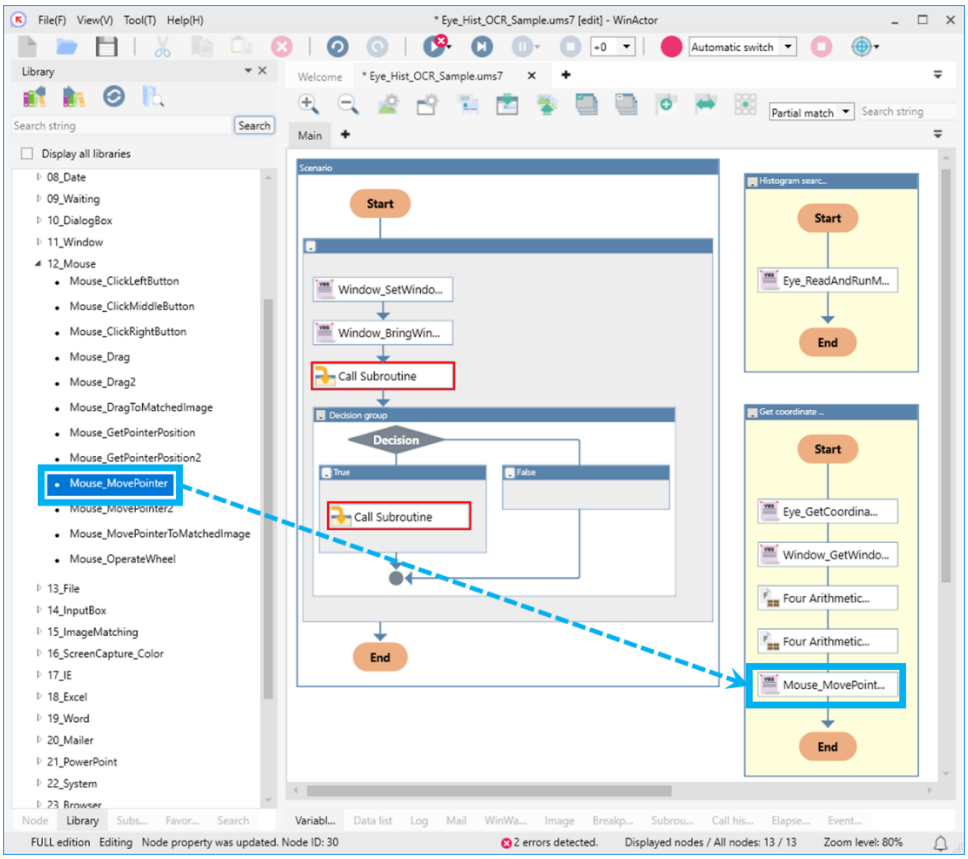


Figure 2.6-14. Placing the "Mouse_MovePointer" library

Set the property values according to the table below.

Table 2.6-11. Property settings

Item	Setting value	Remarks
Coordinate_specification	Absolute_coordinate	Select from the drop-down list.
X-coordinate	x-coordinate	Select from the drop-down list.
Y-coordinate	y-coordinate	Select from the drop-down list.

Add the processing of clicking the mouse after moving the mouse pointer.

Drag and drop "Mouse_ClickLeftButton" from the Library tab.

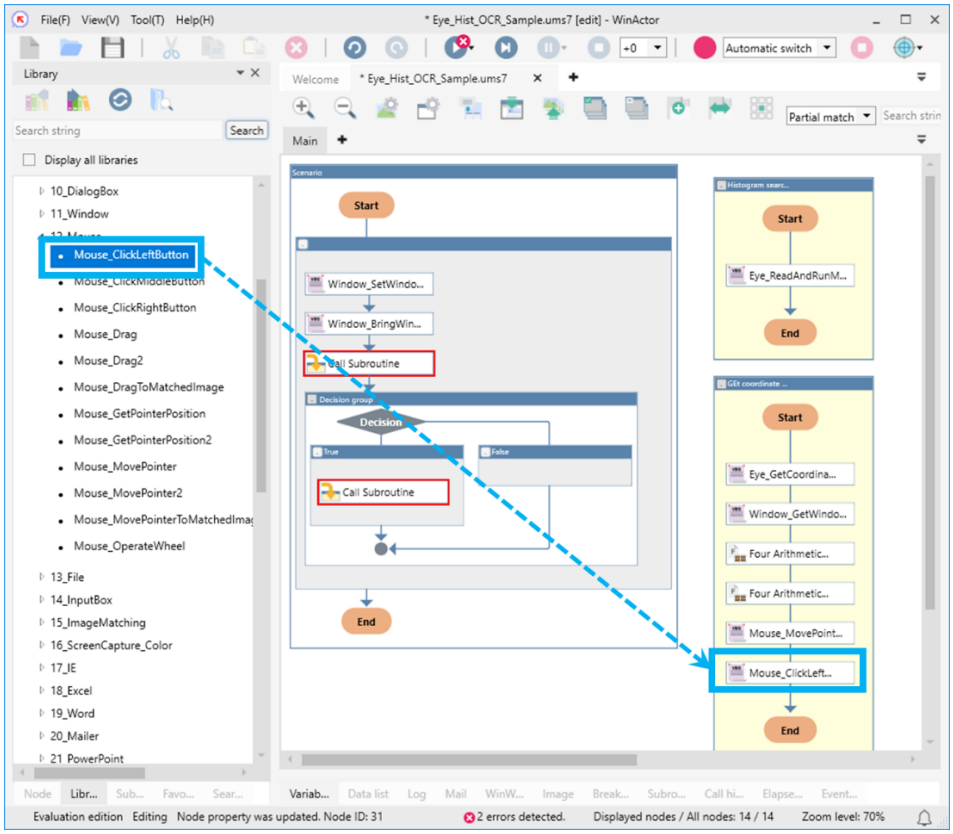


Figure 2.6-15. Placing the "Mouse_ClickLeftButton" library

Set the property value according to the table below.

Table 2.6-12. Property setting

Item	Setting value	Remarks
Mouse_operation	Left_click	Select from the drop-down list.

Set variables in such a way that the scenario is to be operated.

Select [View] > [Variable list] from the WinActor menu to display the "Variable list" pane.

Set the initial values according to the table below.

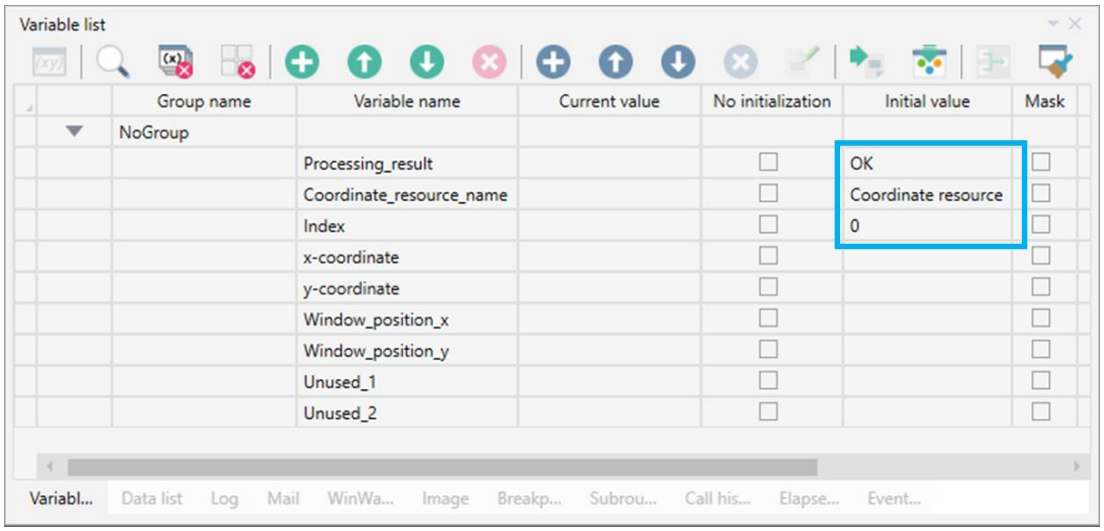


Figure 2.6-16. Setting the initial values

Table 2.6-13. Property settings

Variable name	Initial value	Remarks
Processing_result	OK	Enter in half-width characters
Coordinate_resource_name	Coordinate resource output	-
Index	0	Enter in half-width characters

For the "Call Subroutine" added in Figure 2.6-5, set the property values according to the table below.

Table 2.6-14. Property settings

Item	Setting value	Remarks
Name	Call preprocessing subroutine	-
Subroutine name	Histogram search preprocessing	Select from the drop-down list.

For the "Call Subroutine" added in Figure 2.6-7, set the property values according to the table below.

Table 2.6-15. Property settings

Item	Setting value	Remarks
Name	Call post-processing subroutine	-
Subroutine name	Get coordinate info and operate mouse	Select from the drop-down list.

The final scenario will be as shown in the figure below.

Overwrite the created scenario.

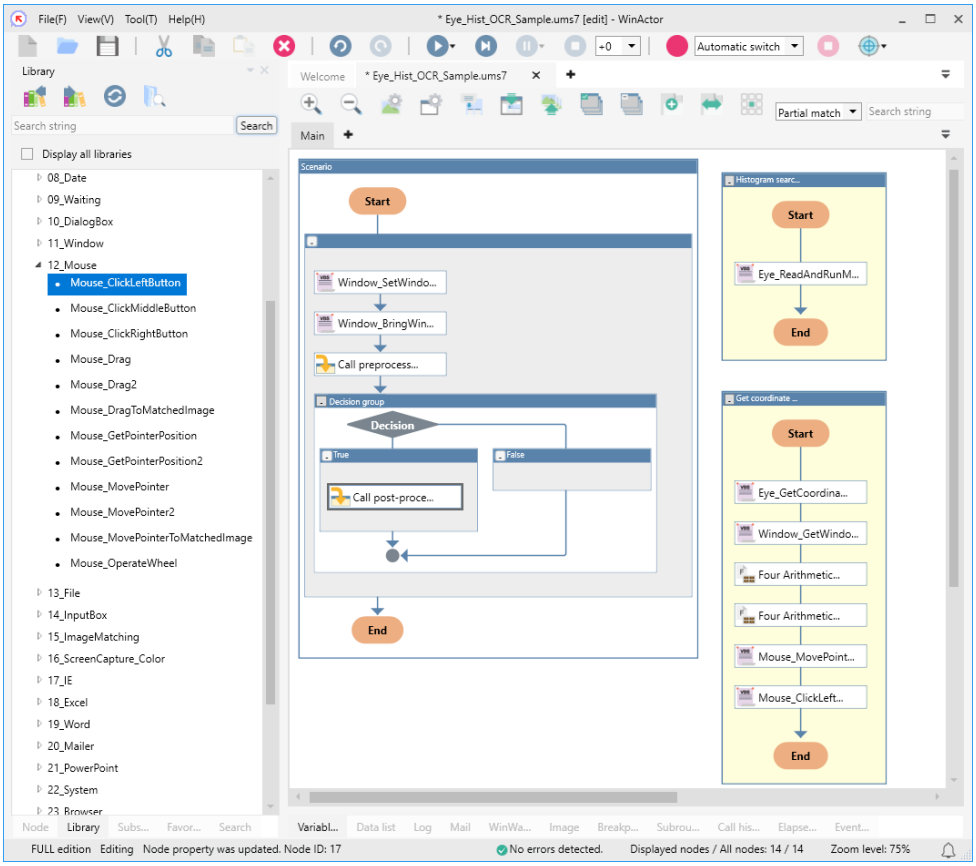


Figure 2.6-17. Created scenario

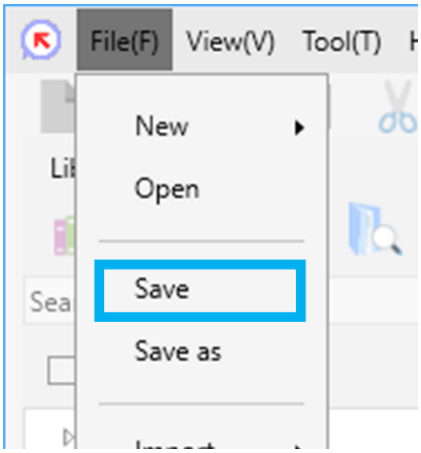


Figure 2.6-18. Overwriting the scenario

2.6.4. Checking the operations

Run the created scenario.

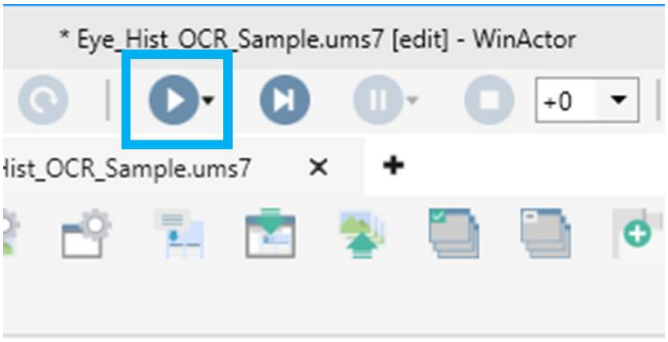


Figure 2.6-19. Running the scenario

If the operations are successful, the Tutorial page will be as shown in the figure below.
If it does not work, launch WinActorEye and check if it looks as shown in Figure 2.5-17.

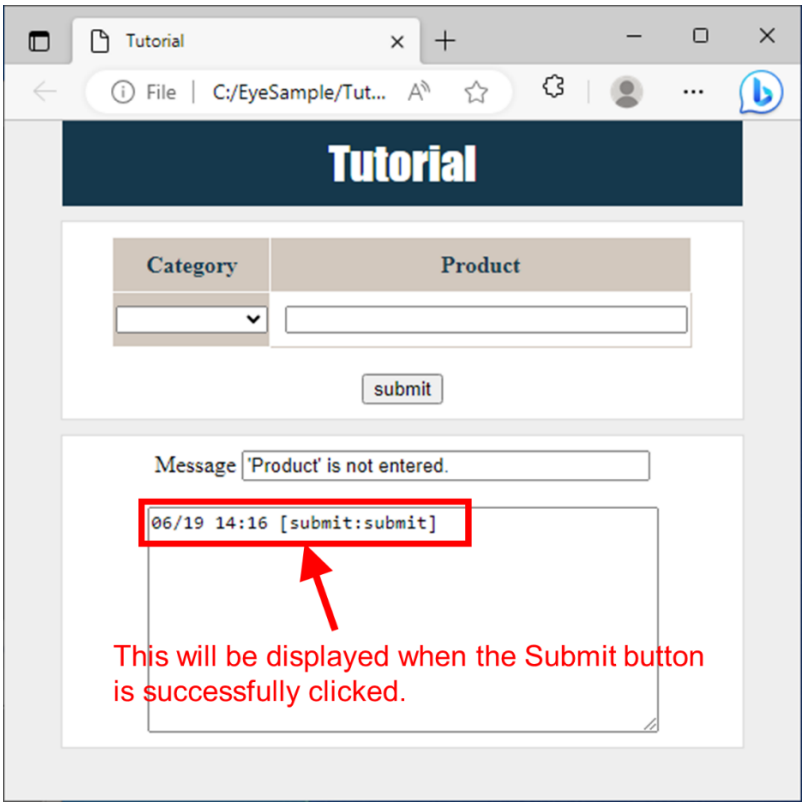


Figure 2.6-20. Successful example

2.7. Creating the WinActorEye macro for the OCR tool

In this section, you will get character information and coordinate information from the Tutorial page by using the OCR tool.

Launch WinActorEye and enable the macro recording as described in Section 2.5.

2.7.1. Capturing the Tutorial page

Capture the Tutorial page according to the steps in 2.5.1.

2.7.2. Calling the OCR tool

Select [Tool] > [OCR tool] to display the "OCR tool" property window.

Click the button of ① and select "Tutorial - Profile 1 - Microsoft Edge," enter "OCR" in ②, and then click the Execute button of ③. Based on these settings, the character information and the rectangle information will be output to the clipboard and to the rectangle resource specified in ② respectively from the "Tutorial - Profile 1 - Microsoft Edge" image resource.

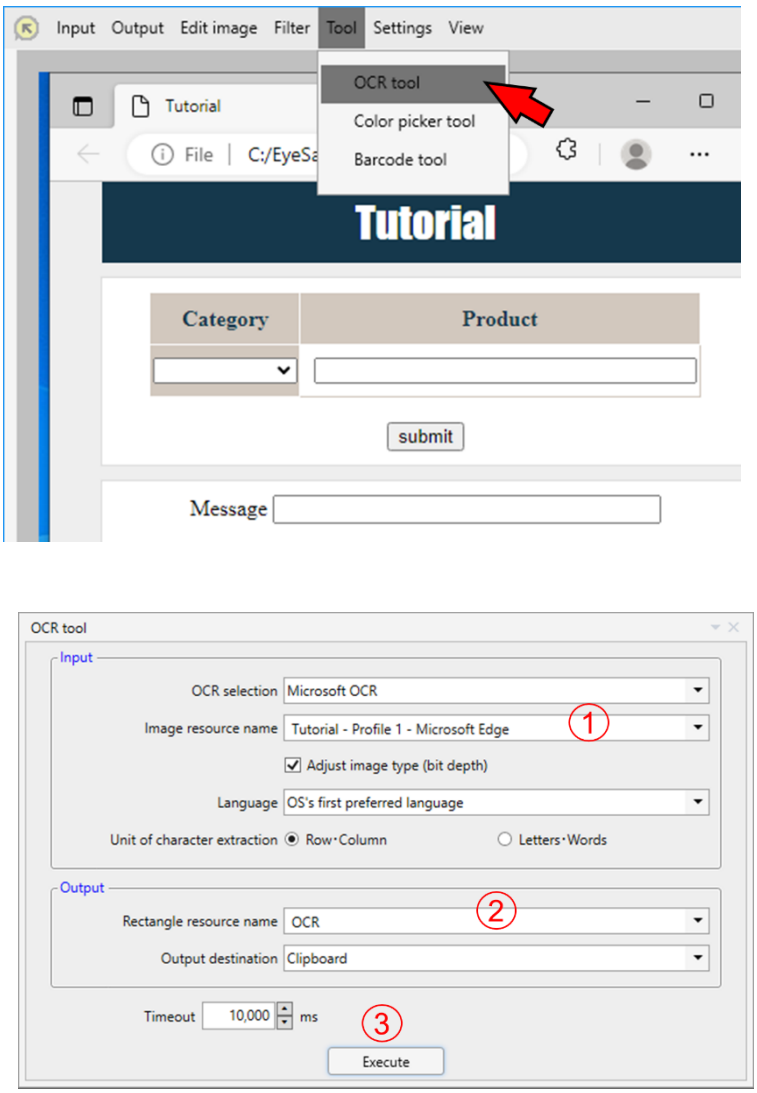


Figure 2.7-1. "OCR tool" property window

2.7.3. Converting the rectangle resource to the coordinate resource

To get the coordinate information of the targeted image, convert the rectangle resource of the targeted image to the coordinate resource.

Select [Filter] > [Extract coordinates from rectangle] to display the "Extract coordinates from rectangle" property window. Click the button of ① and select "OCR," enter "Coordinate resource output" in ②, and then click the Execute button of ③.

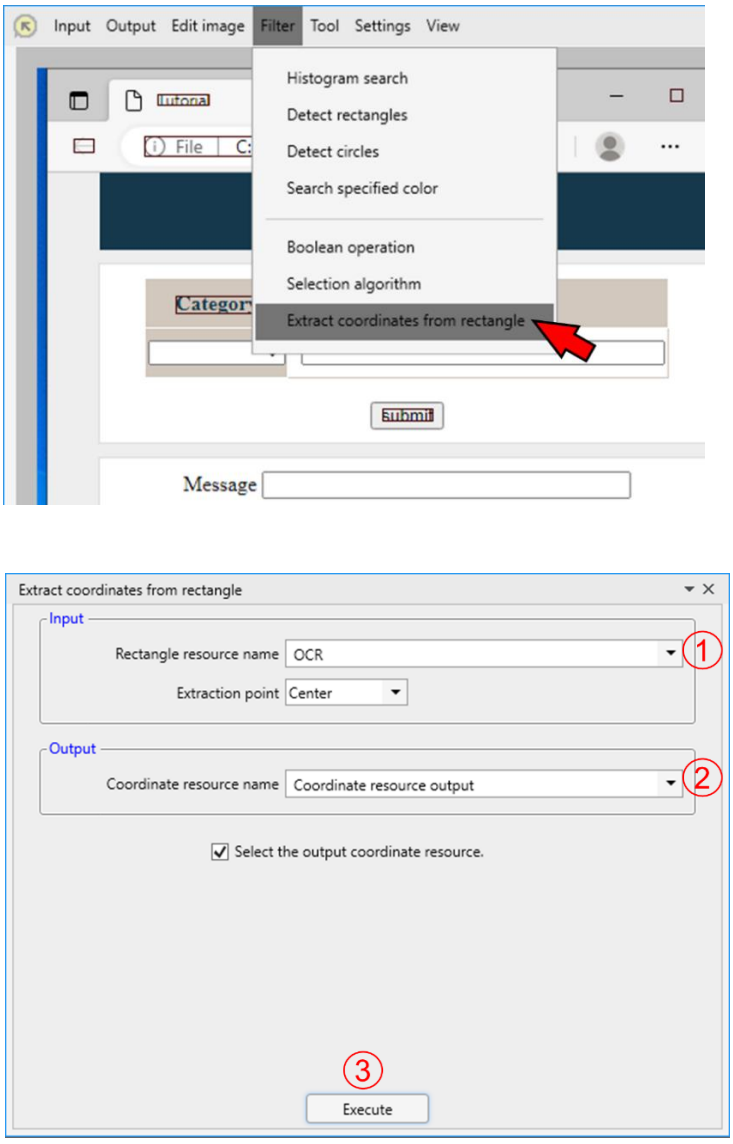


Figure 2.7-2. "Extract coordinates from rectangle" property window

Save the created macro.

Click the "Record" icon ① to disable the macro recording, and then click the Save icon ② to save the created macro to "C:\EyeSample\Run_MSOCR.json."

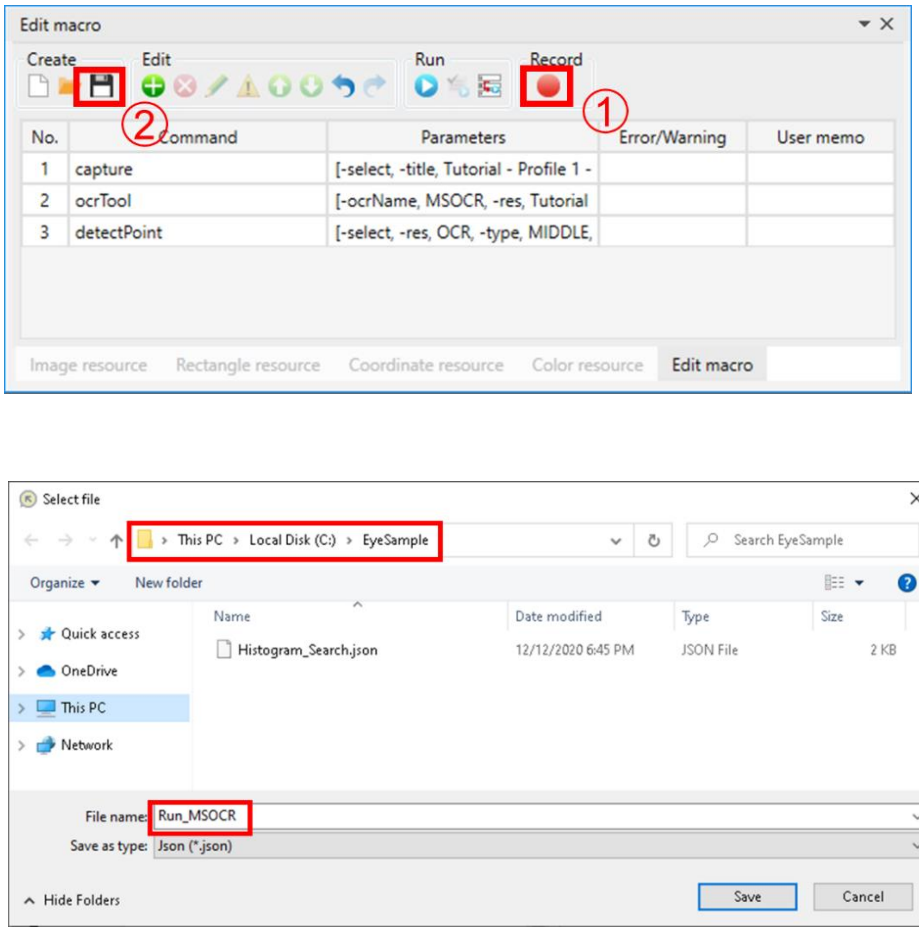


Figure 2.7-3. Saving the macro

2.8.　Creating the WinActor Note macro for the OCR tool

Since the character information is stored in the clipboard in Section 2.7, create the macro for searching the target characters from the clipboard using WinActor Note.

For how to use WinActor Note, see the material No.5 in Table 4.1-1 in "4.1Reference materials."

Launch WinActor Note and enable the macro recording. (The state in Figure 2.8-1 is the recording state. If it is not in the recording state, click the icon in the red frame.)

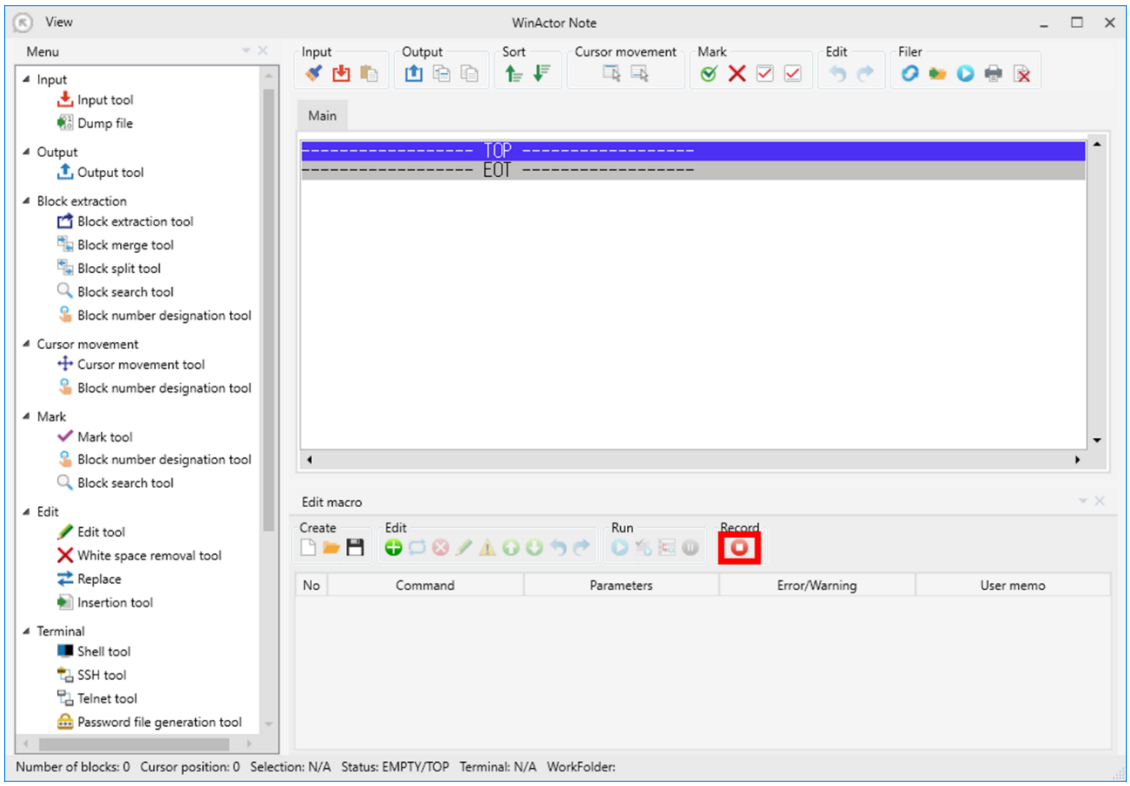


Figure 2.8-1. WinActor Note at the start of macro creation

2.8.1. Clearing the window and pasting from the clipboard

Click the [Clear] icon in the Input menu to clear the data in WinActor Note so that the previous execution logs do not remain in WinActor Note.

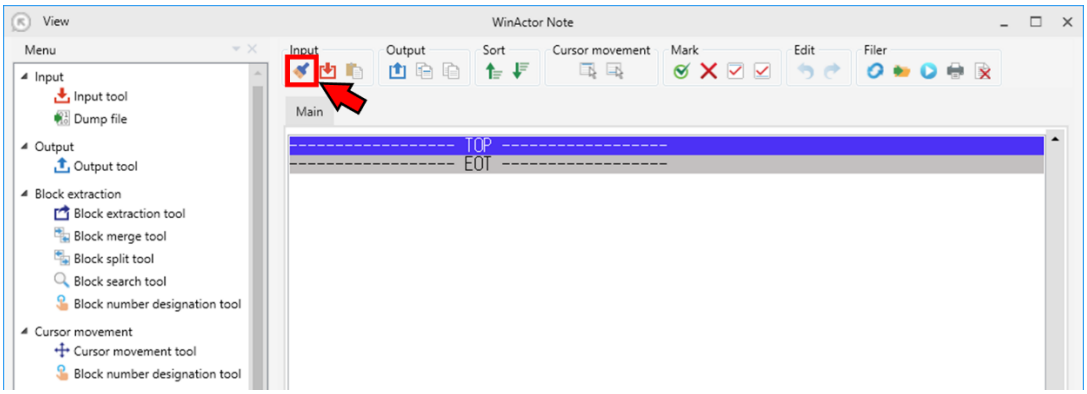


Figure 2.8-2. Clearing the data

In the "Edit macro" pane of WinActor**Eye**, click the "Run all" icon.

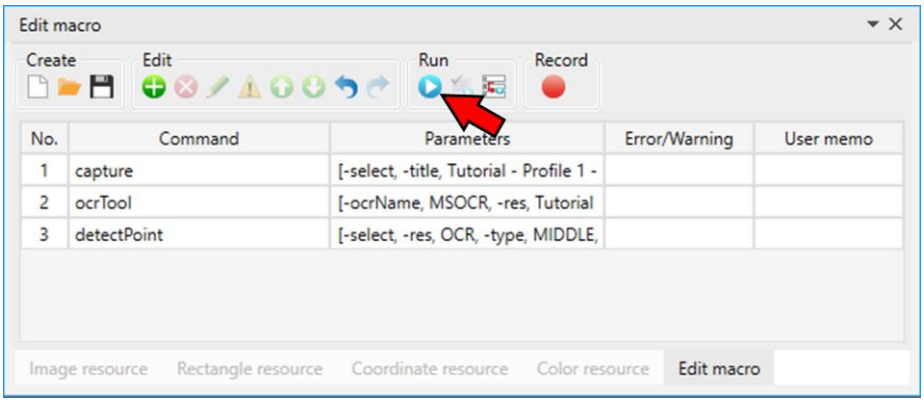


Figure 2.8-3. Running the macro of WinActorEye

Click the [Paste] icon in WinActor Note. The characters acquired by running the macro of WinActorEye will be pasted in WinActor Note.

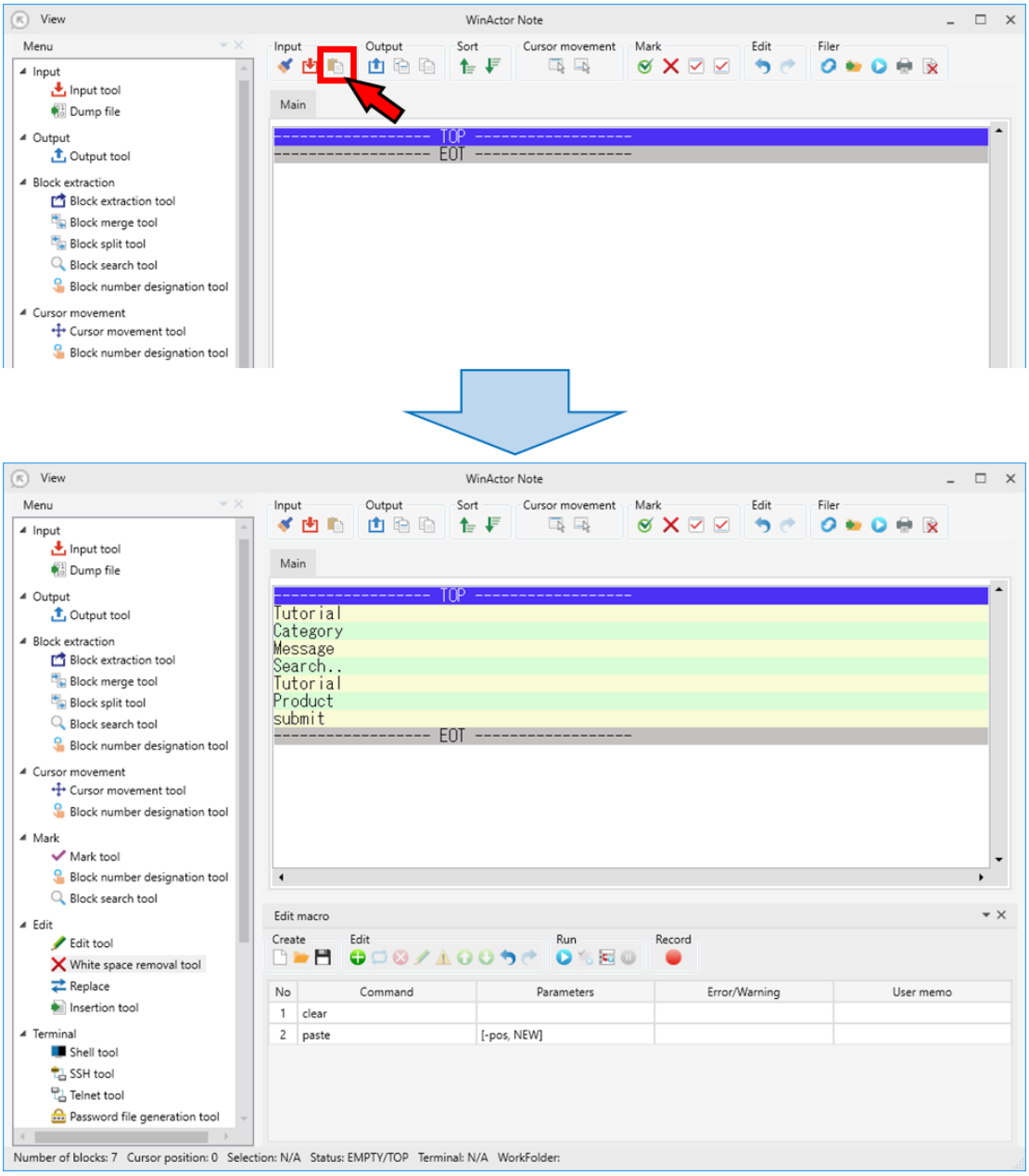


Figure 2.8-4. Pasting the characters from the clipboard

2.8.2. Removing spaces from the characters

Remove unnecessary spaces from the characters on WinActor Note.

Select [Edit] > [White space removal tool] to display the "White space removal tool" property window.

Select "All blocks" for the target ①, "Remove white spaces" for ②, and click the Execute button ③.

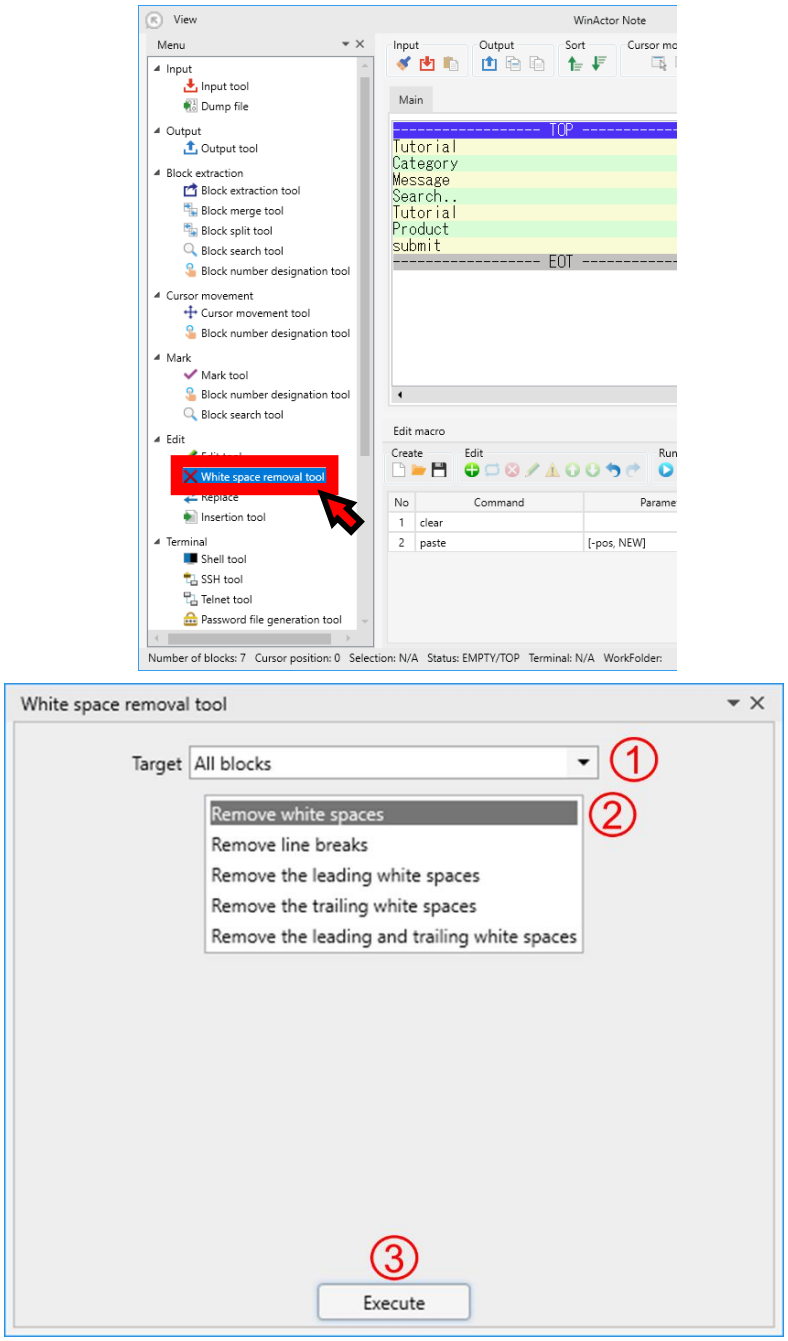


Figure 2.8-5. "White space removal tool" property window

2.8.3. Searching for "submit"

Search for the character string "submit" in WinActor Note.

Select [Block extraction] > [Block search tool] to display the "Block search tool" property window.

Enter "submit" for the keyword ①, select "Contain" for ② and "Forward search (cursor moves down)" for ③, and click the Execute button ④.

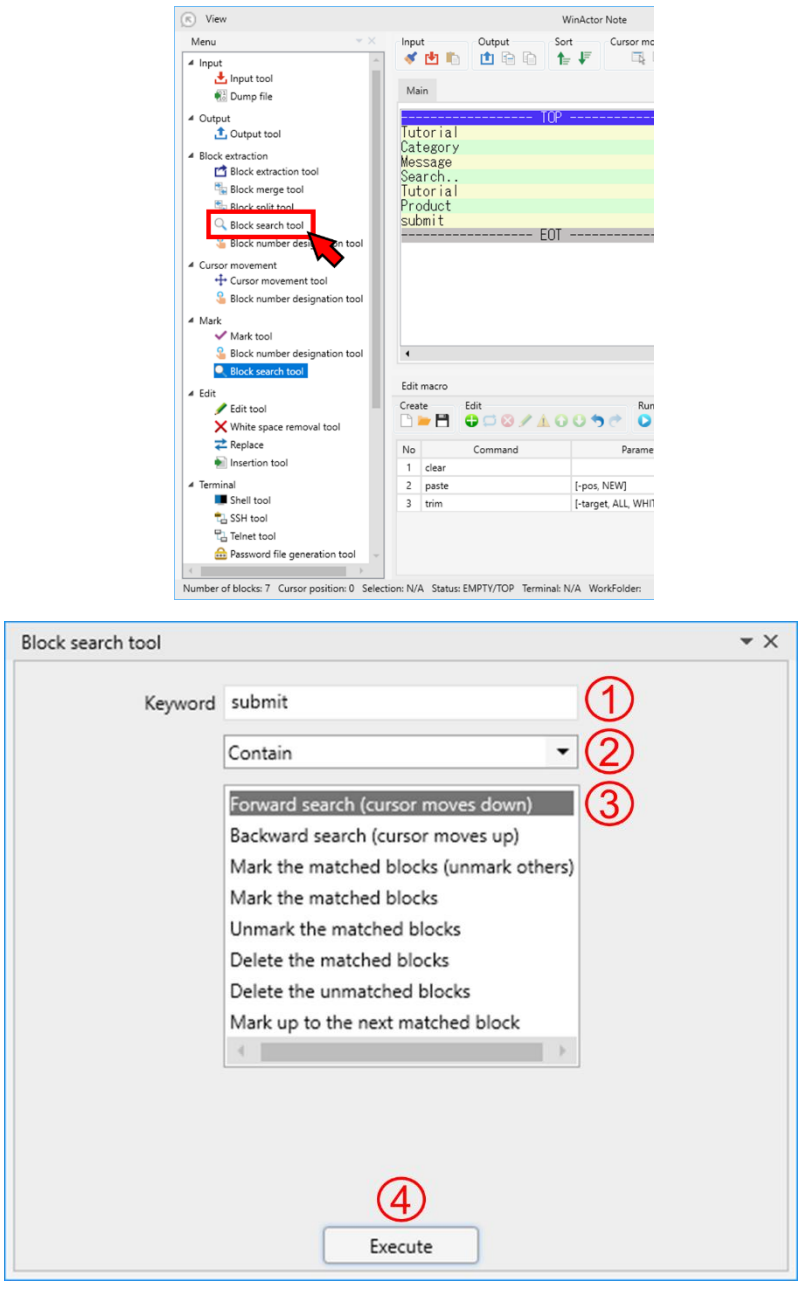


Figure 2.8-6. "Block search tool" property window

After clicking the Execute button, the cursor will move to the line containing "submit."

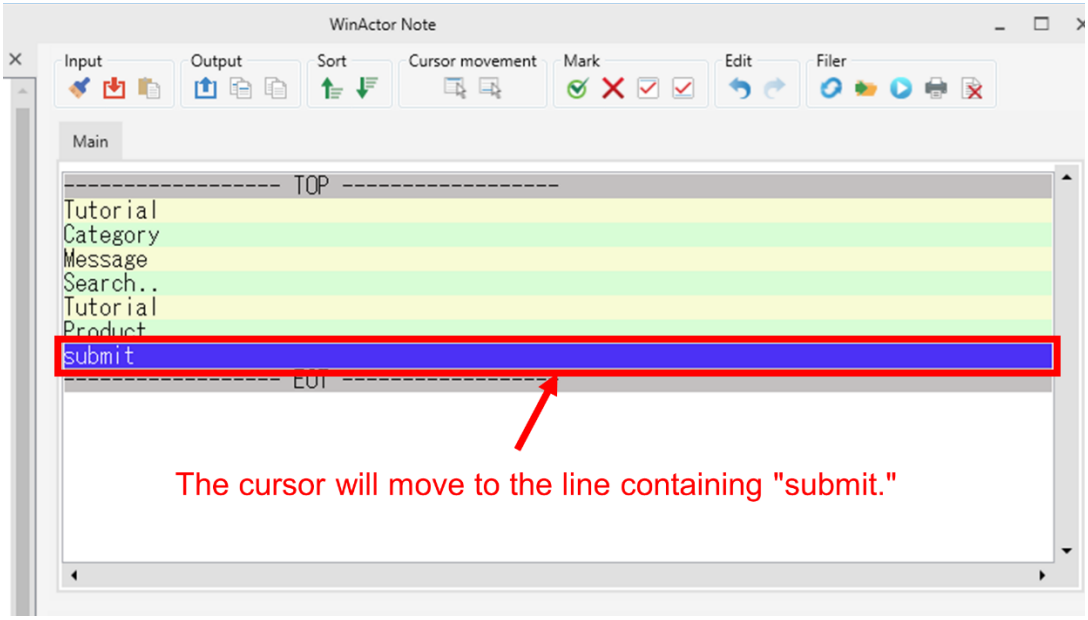


Figure 2.8-7. After searching for "submit"

Save the created macro.

Click the "Record" icon ① to disable the macro recording, and then click the "Save" icon ② to save the created macro to "C:\EyeSample\Search_submit.json."

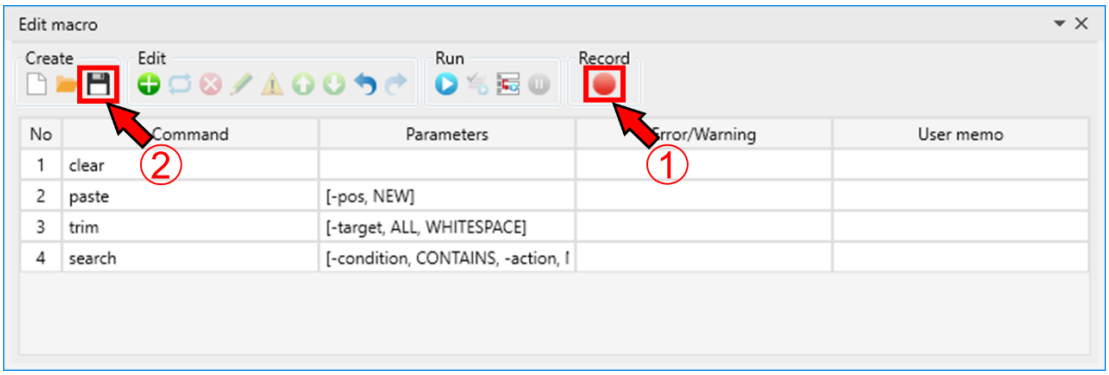


Figure 2.8-8. Saving the macro

2.9.　Creating the WinActor scenario for the OCR tool

In WinActor, create the scenario for clicking the submit button on the Tutorial page by using the libraries that run the macros created in Section 2.7 and 2.8.

2.9.1. Creating the OCR tool preprocessing subroutine

Create the processing specific to the OCR tool as a preprocessing subroutine.

Drag and drop "Subroutine Group" from the Node tab.

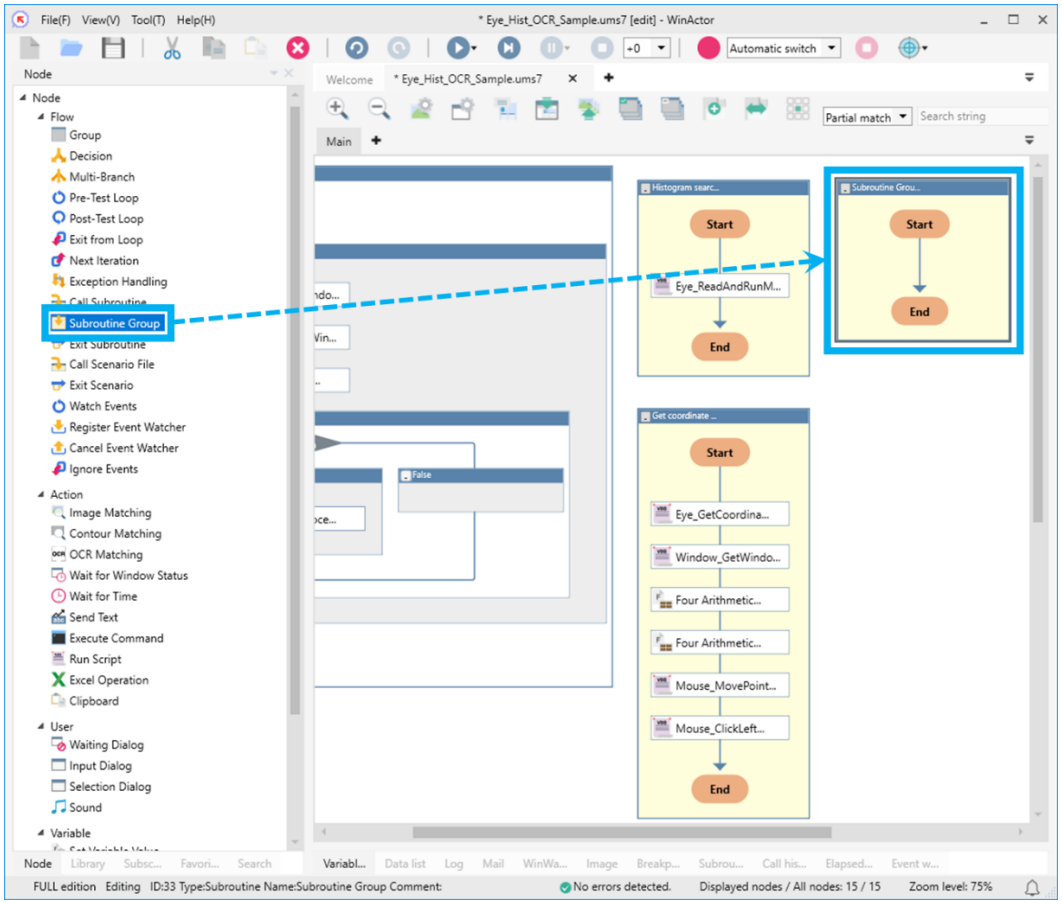


Figure 2.9-1. Placing the "Subroutine Group" node

Set the property value according to the table below.

Table 2.9-1. Property setting

Item	Setting value	Remarks
Name	OCR tool preprocessing	-

Place the library that runs the WinActorEye macro created in Section 2.7.

Drag and drop "Eye_ReadAndRunMacro" from the Library tab.

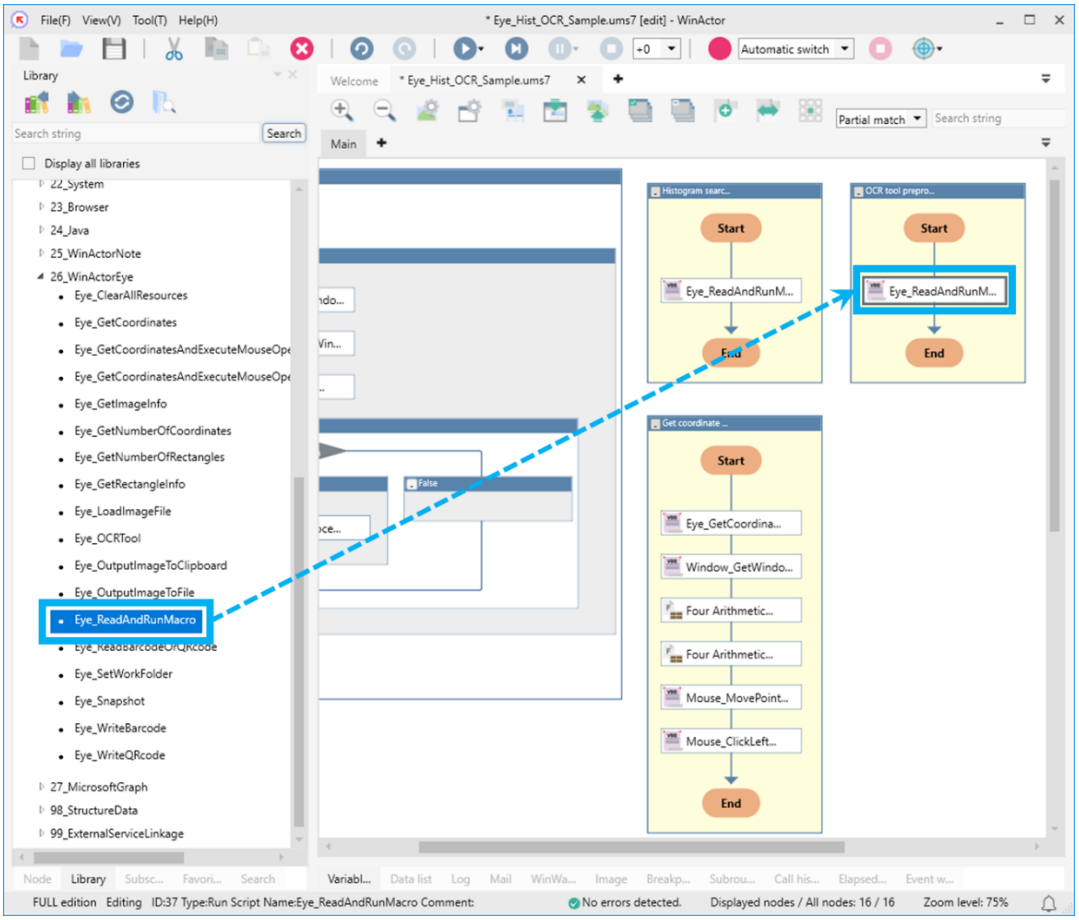


Figure 2.9-2. Placing the "Eye_ReadAndRunMacro" library

Set the property value according to the table below.

Table 2.9-2. Property setting

Item	Setting value	Remarks
Macro_filename	Value=>Run_MSOCR.json	Select "Value=>" and enter the value to the right of "Value=>."

To check the character acquired by MSOCR, open WinActor Note in the edit mode.
Drag and drop "Note_ChangeMode" from the Library tab.

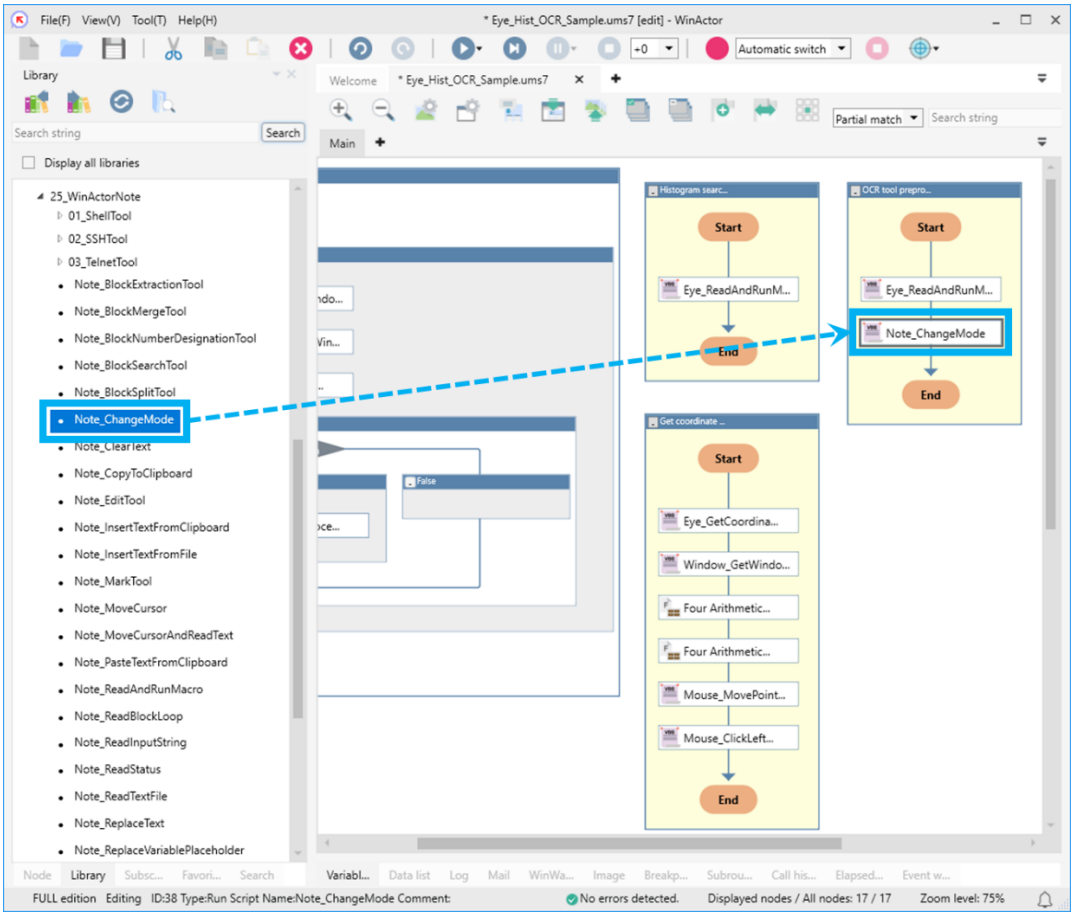


Figure 2.9-3. Placing the "Note_ChangeMode" library

Set the property value according to the table below.

Table 2.9-3. Property setting

Item	Setting value	Remarks
Mode	Edit	Select from the drop-down list.

Place the library that runs the WinActor Note macro created in Section 2.8.

Drag and drop "Note_ReadAndRunMacro" from the Library tab.

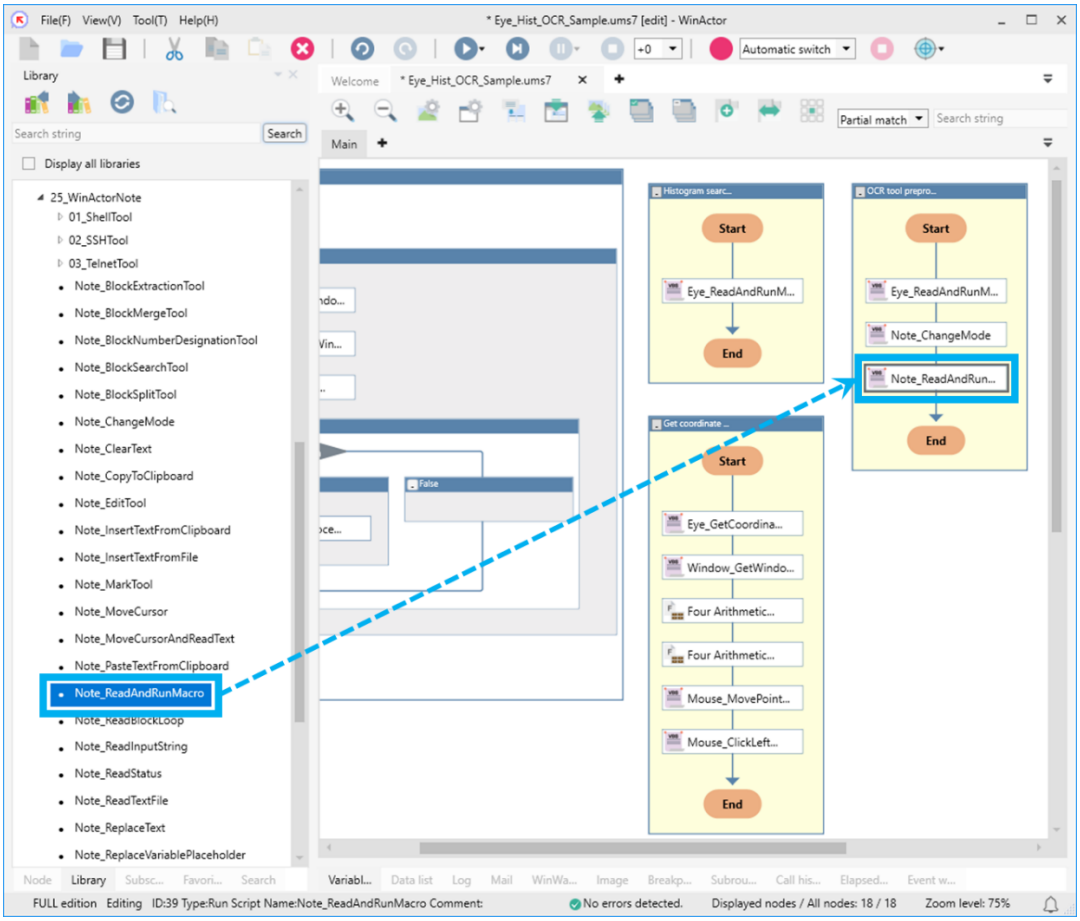


Figure 2.9-4. Placing the "Note_ReadAndRunMacro" library

Set the property value according to the table below.

Table 2.9-4. Property setting

Item	Setting value	Remarks
Macro_filename	Value=>Search_submit.json	Select "Value=>" and enter the value to the right of "Value=>."

Get the cursor position resulting from the search for the character string "submit" in WinActor Note.

Drag and drop "Note_ReadStatus" from the Library tab.

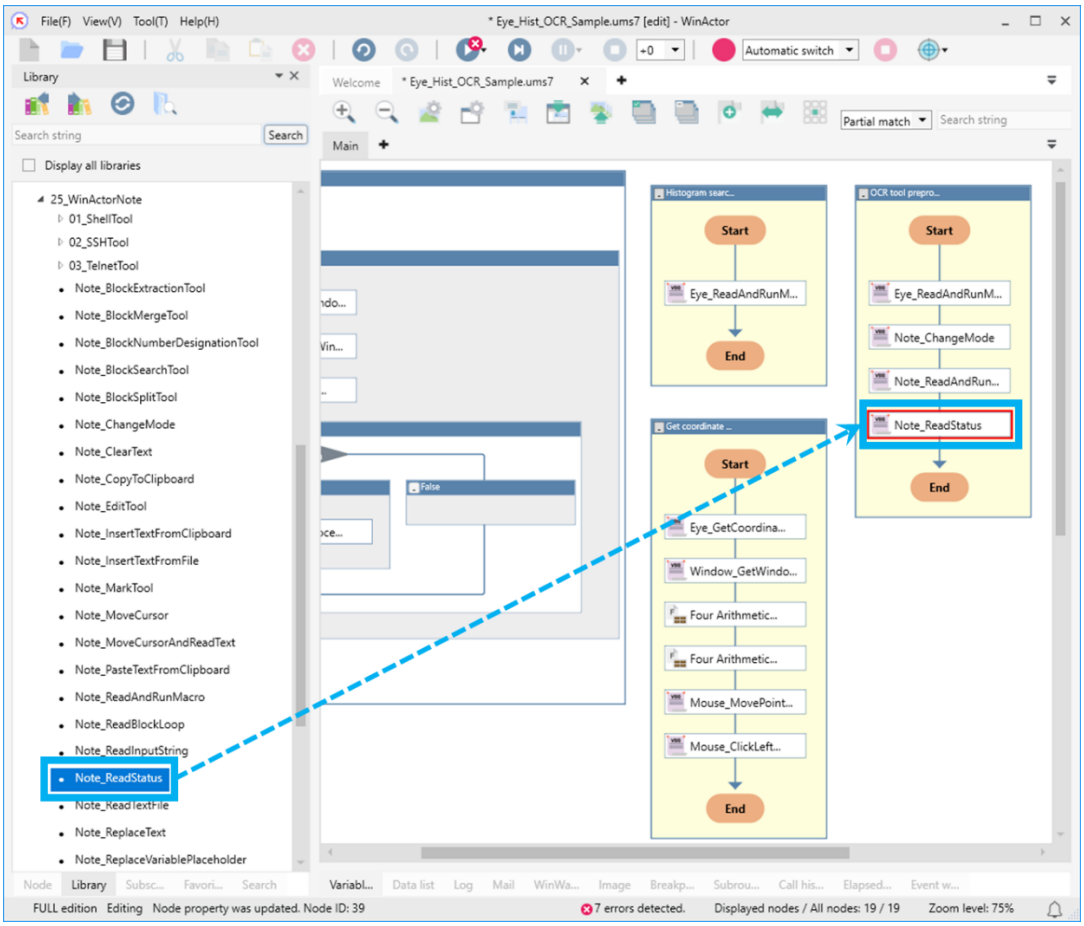


Figure 2.9-5. Placing the "Note_ReadStatus" library

Set the property values according to the table below.

Table 2.9-5. Property settings

Item	Setting value	Remarks
Selection	Unused_1	Select from the drop-down list.
Cursor_position	Cursor_position	-
Number_of_blocks	Unused_2	Select from the drop-down list.
MARK	Unused_3	-
EMPTY	Unused_4	-

TOP	Unused_5	-
EOT	EOT	-

Add the processing of determining whether the character string "submit" was found based on the cursor position acquired by running "Note_ReadStatus."

Drag and drop "Decision" from the Node tab.

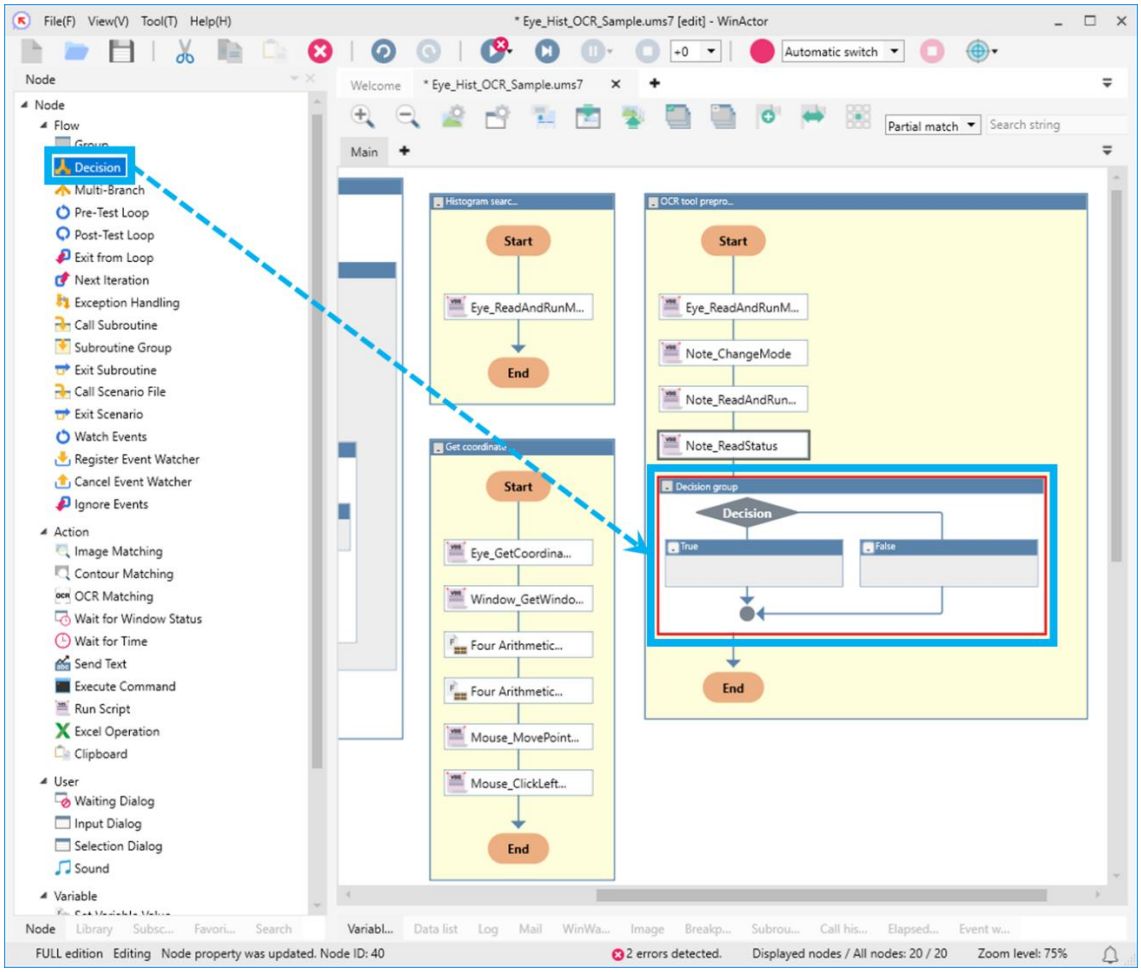


Figure 2.9-6. Placing the "Decision" node

Click the ‘Edit’ button of the property window and set the conditional expression according to the table below.

Table 2.9-6. Conditional expression settings

Item	Left-hand side	Comparison operator	Right-hand side
Setting value	EOT	is not equal to	Value=>true
Remarks	Select from the drop-down list.	-	Select "Value=>" and enter the value to the right of "Value=>."

In the normal flow of Decision, convert the information of the cursor position into the index number of the coordinate resource used in WinActorEye.

Drag and drop "Four Arithmetic Operations" from the Node tab.

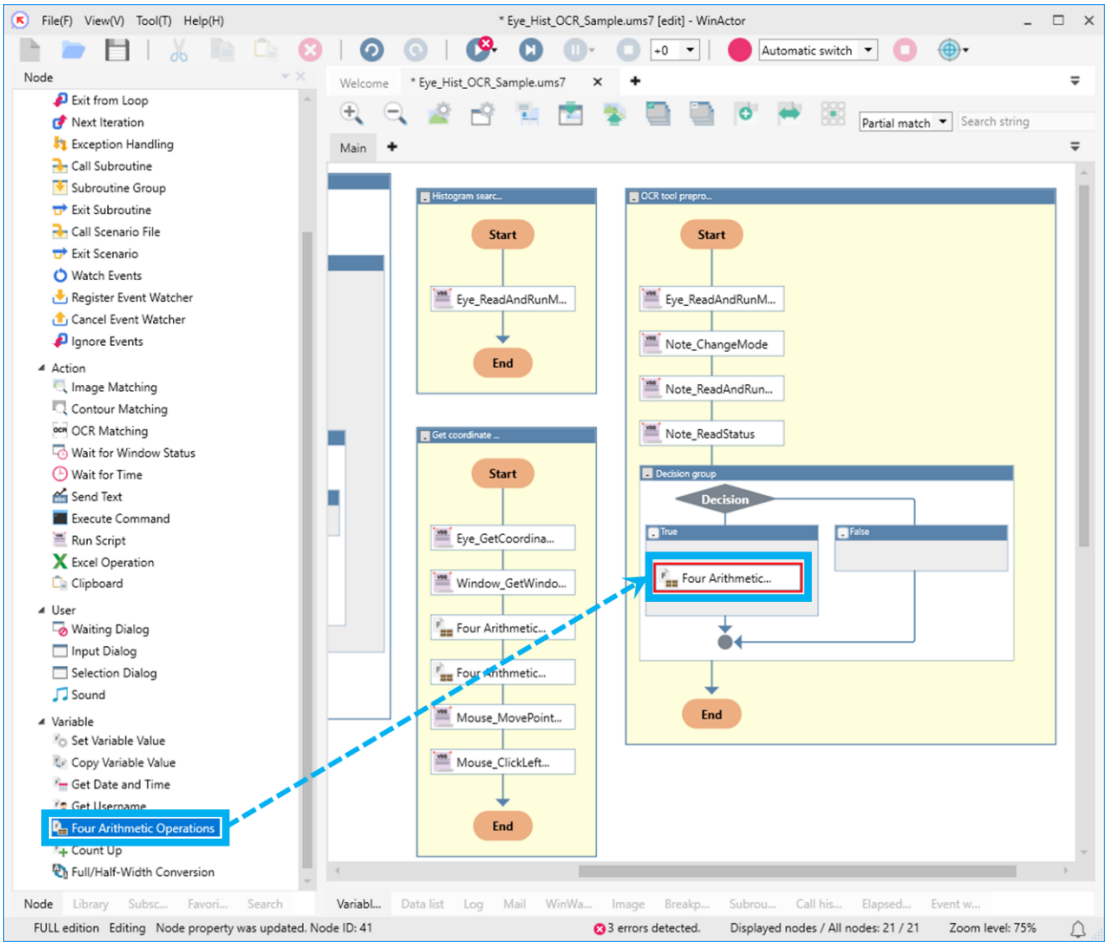


Figure 2.9-7. Placing the "Four Arithmetic Operations" node

Set the property values according to the table below.

Table 2.9-7. Property settings

Item	Setting value	Remarks
Calculation result	Index	Select from the drop-down list.
Operator	-	Select from the drop-down list.
Left side of -	Cursor_position	Select from the drop-down list.
Right side of -	Value=>1	Select "Value=>" and enter the value to the right of "Value=>."

In the exceptional flow of Decision, set the processing of displaying a message if the expected character string was not detected by the OCR tool.

Drag and drop "Waiting Dialog" from the Node tab.

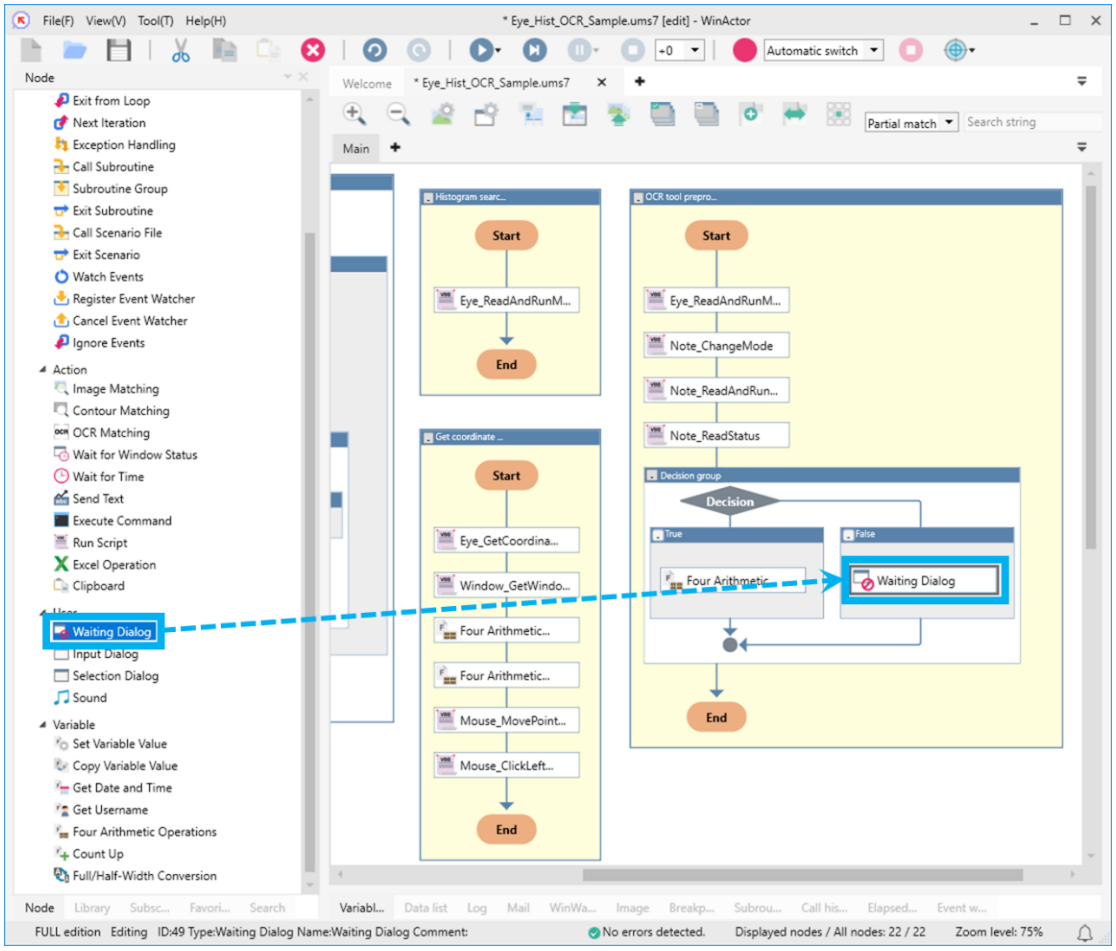


Figure 2.9-8. Placing the "Waiting Dialog" node

Set the property values according to the table below.

Table 2.9-8. Property settings

Item	Setting value	Remarks
First option	Confirmation dialog (OK button only)	-
Prompt message	Message	-
Message contents	The targeted character string was not detected.	-

In the exceptional flow of Decision, set the processing of not calling the post-processing subroutine.

Drag and drop "Set Variable Value" from the Node tab.

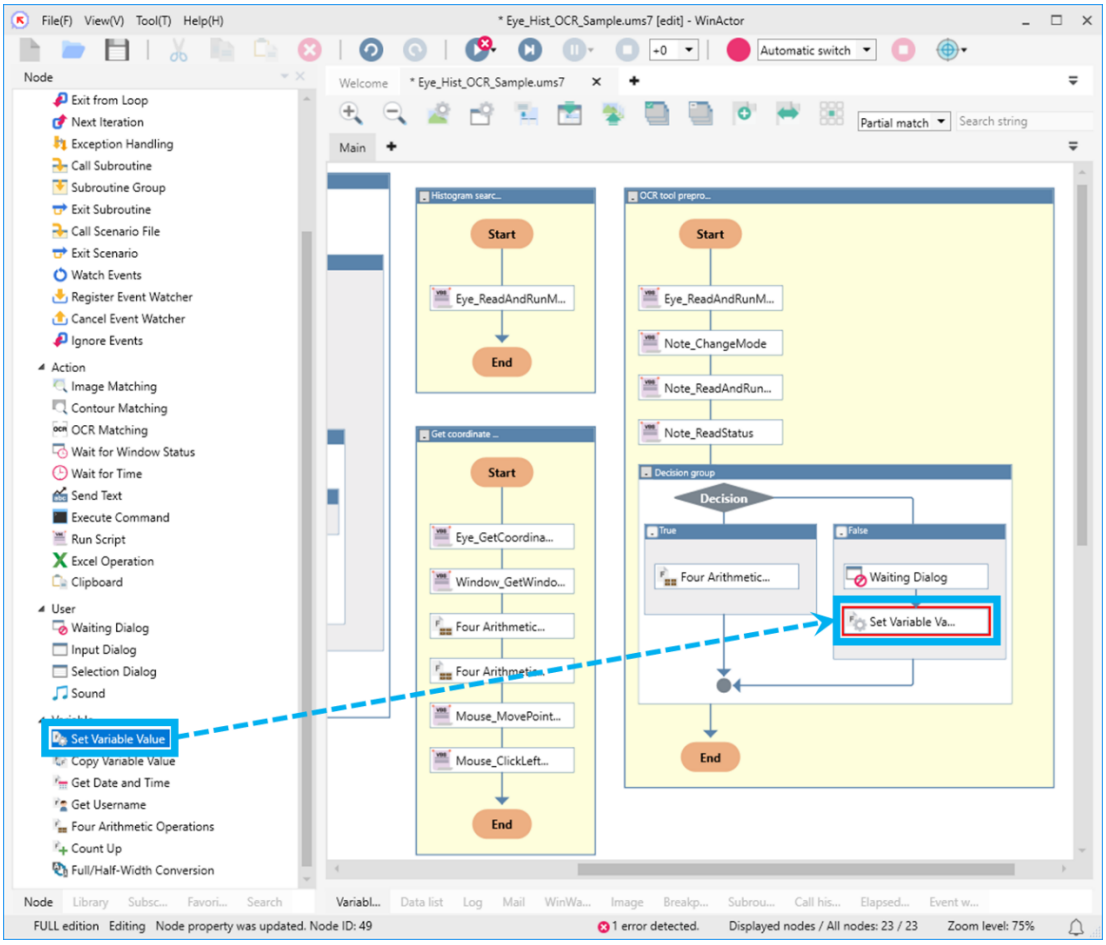


Figure 2.9-9. Placing the "Set Variable Value" node

Set the property values according to the table below.

Table 2.9-9. Property settings

Item	Setting value	Remarks
Variable name	Processing_result	Select from the drop-down list.
Value	NG	Enter in half-width characters

Set "OCR tool preprocessing" to be called in "Call preprocessing subroutine."

For the "Call Subroutine" added in Figure 2.6-5, set the property value according to the table below.

Table 2.9-10. Property setting

Item	Setting value	Remarks
Subroutine name	OCR tool preprocessing	Select from the drop-down list.

The final scenario will be as shown in the figure below.

Overwrite the created scenario.

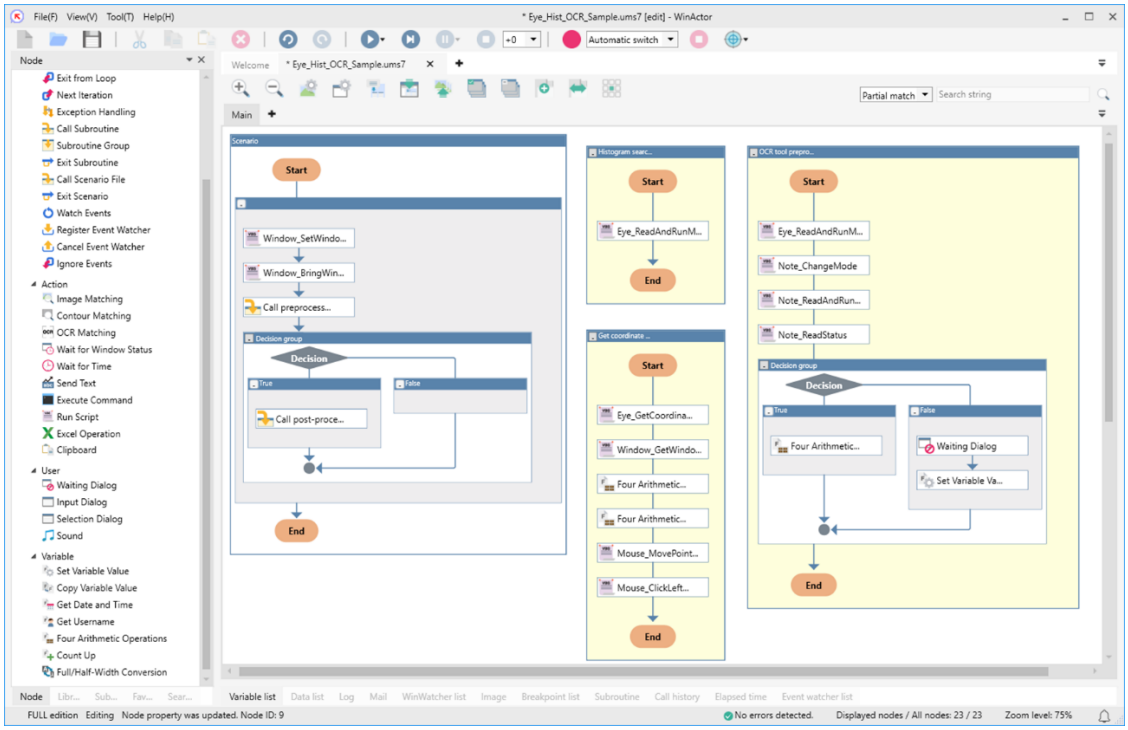


Figure 2.9-10. Created scenario

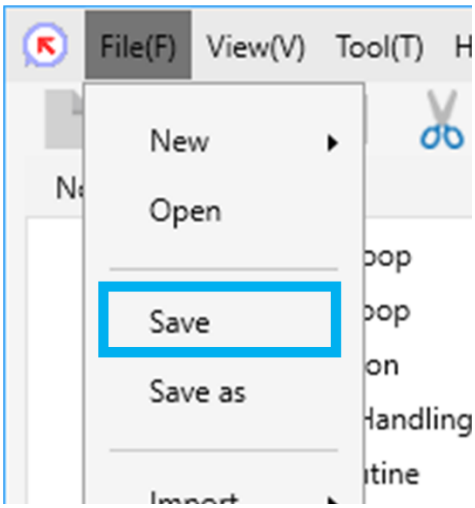


Figure 2.9-11. Overwriting the scenario

2.9.2. Checking the operations

Run the created scenario.

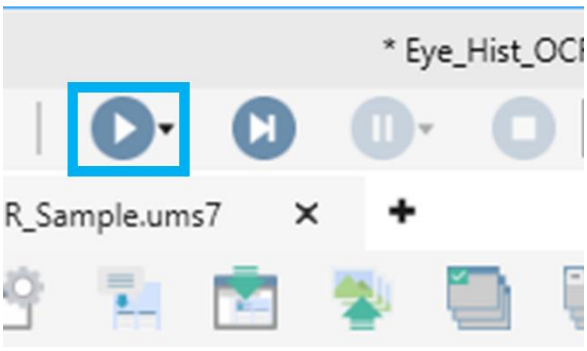


Figure 2.9-12. Running the scenario

If the operations are successful, the Tutorial page will be as shown in the figure below.

If it does not work, launch WinActorEye and check if the "submit" button is surrounded by a red frame as shown in Figure 2.7-2.

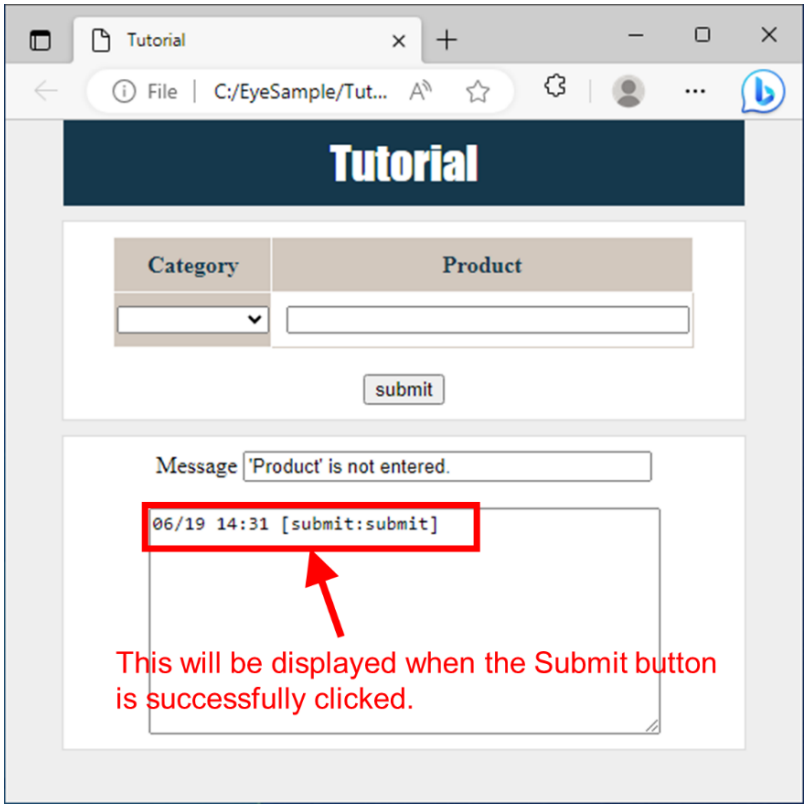


Figure 2.9-13. Successful example

3.Library and property list

This chapter introduces the user libraries related to WinActorEye.

3.1. Eye_ReadAndRunMacro

This library is to read a macro into WinActorEye and run the macro.

Property item	Description
Macro_filename	Specify a name of a macro file to be read with an absolute or relative path. If a relative path is specified, it will be a relative path from the following folder. Work folder is set: Work folder. Work folder is not set: The folder where the scenario currently in progress is saved.

3.2. Eye_GetCoordinates

This library is to get coordinate information for a specified coordinate resource name in the coordinate resource of WinActorEye.

Property item	Description
Coordinate_resource_name	Specify a coordinate resource name for which you want to get coordinate information.
Index	Specify which x-coordinate and y-coordinate you want to get from the acquired coordinate information. If not entered, it will be the first x-coordinate and y-coordinate. The index of the first x-coordinate and y-coordinate is 0.
Variable_for_x_coordinate	Specify a variable to store the acquired x-coordinate.
Variable_for_y_coordinate	Specify a variable to store the acquired y-coordinate.

3.3. Eye_GetNumberOfCoordinates

This library is to get the number of coordinates for a specified coordinate resource name in the coordinate resource of WinActorEye.

Property item	Description
Coordinate_resource_name	Specify a coordinate resource name for which you want to get the number of coordinates.

Variable_for_number_of_coordinates	Specify a variable to store the acquired number of coordinates.

3.4. Eye_GetCoordinatesAndExecuteMouseOperation

This library is to execute a specified mouse operation for a specified window based on the coordinate information of WinActorEye.

Property item	Description
WinID name	Specify a window where you want to operate the mouse.
Coordinate_resource_name	Specify a coordinate resource name for which you want to get coordinate information.
Index	Specify which coordinate information in the specified coordinate resource you want to get. If not entered, it will be the first x-coordinate and y-coordinate. The index of the first x-coordinate and y-coordinate is 0.
Variable_for_x_coordinate	Specify a variable to store the acquired x-coordinate.
Variable_for_y_coordinate	Specify a variable to store the acquired y-coordinate.
Mouse_operation	Specify a mouse operation.
Click_interval(ms)	Specify a wait time between clicks. If not set, or if it is set with non-numeric characters or outside the valid range, it will be executed without waiting time. (Range: 0 to int type maximum (2147483647))

3.5. Eye_GetCoordinatesAndExecuteMouseOperationWithoutDropShadow

This library executes a mouse operation on the specified window with the coordinates taken from WinActorEye.

* The mouse cursor is operated on the coordinates of the specified window without drop shadow, which is the visual effect “Show shadows under windows” of Windows.

Property item	Description
WinID name	Specify the window on which the mouse operation is performed.
Coordinate_resource_name	Specify the coordinate resource name.
Index	Specify the index of the coordinates. When omitted, the first coordinates are taken. The index of the first coordinates is 0.
Variable_for_x_coordinate	Specify a variable to store the x-coordinate.
Variable_for_y_coordinate	Specify a variable to store the y-coordinate.
Mouse_operation	Select a mouse operation to be performed.
Click_interval(ms)	Enter an interval between clicks in milliseconds. When it is empty or not a valid number, no interval time is applied. (valid number: 0 - maximum integer (2147483647))

3.6. Eye_GetImageInfo

This library is to get information about the width, height, and image type of a specified image resource.

Property item	Description
Image_resource_name	Specify an image resource name for which you want to get image information.
Variable_for_width	Specify a variable to store the acquired width.
Variable_for_height	Specify a variable to store the acquired height.
Variable_for_imgae_type	Specify a variable to store the acquired image type.

3.7. Eye_GetRectangleInfo

This library is to get information about the x-coordinate, y-coordinate, width, height, and angle of a specified rectangle resource.

Property item	Description
Rectangle_resource_name	Specify a rectangle resource name for which you want to get rectangle information.
Index	Specify which rectangle information in the specified rectangle resource you want to get. If not entered, it will be the first x-coordinate, y-coordinate, width, height, and angle. The index of the first rectangle information is 0.
Variable_for_x_coordinate	Specify a variable to store the acquired x-coordinate.
Variable_for_y_coordinate	Specify a variable to store the acquired y-coordinate.
Variable_for_width	Specify a variable to store the acquired width.
Variable_for_height	Specify a variable to store the acquired height.
Variable_for_angle	Specify a variable to store the acquired angle.

3.8. Eye_GetNumberOfRectangles

This library is to get the number of rectangles for a specified rectangle resource.

Property item	Description
Rectangle_resource_name	Specify a rectangle resource name for which you want to get the number of rectangles.
Variable_for_number_of_rectangles	Specify a variable to store the acquired number of rectangles.

3.9. Eye_SetWorkFolder

This library is to set a work folder of WinActorEye.

Property item	Description
Specify_the_folder	Select a folder to be set as a work folder from "Folder_name" (any folder) and "Scenario_folder." If "Scenario_folder" is selected, the folder (scenario folder) where the scenario currently in progress is saved will be specified.
Folder_name	Specify a folder to be set as a work folder with an absolute or

	relative path. If a relative path is specified, the scenario folder will be the starting point. If "Scenario_folder" is selected in Specify_the_folder, the input contents will be replaced with the scenario folder.

3.10. Eye_ClearAllResources

This library is to clear all WinActorEye resources (image resource, rectangle resource, coordinate resource, color resource).

3.11. Eye_LoadImageFile

This library is to load an image file into WinActorEye.

Property item	Description
Image_filename	Specify a name of an image file to be loaded with an absolute or relative path. If a relative path is specified, it will be a relative path from the following folder. Work folder is set: Work folder. Work folder is not set: The folder where the scenario currently in progress is saved.
Selection	Specify whether to select the loaded image. When "Select" is specified, the image resource specified in the property item "Image_resource_name" will be selected on WinActorEye.
Image_resource_name	Specify an image resource name for registering image information. If omitted, the image file name specified in "Image_filename" will be the image resource name.

3.12. Eye_OCRTool

This library reads an image with the ‘OCR tool’ and extracts character data.

Property item	Description
OCR selection	Select an external OCR tool to use.
Image resource name	Specify an input image resource name.
Adjust image type	Select an adjustment of the image type (bit depth).
Language	Select the language of characters to be acquired by OCR.
Unit of character extraction	Select an OCR. Character extraction unit.
Output rectangle resource name	Specify a rectangle resource name for output. If omitted, the selected rectangle resource in the “Rectangle resource” pane is overwritten.
Output destination	Specify an output destination of the extracted characters.
Timeout (ms)	Specify a timeout value. The default value is 10,000 milliseconds.

3.13. Eye_ReadBarcodeOrQRcode

This library reads an image with the ‘Barcode tool’ and extracts character data.

Property item	Description
Image resource name	Specify an image resource name for input.
Output destination	Specify an output destination of the extracted character string.
Timeout (ms)	Specify a timeout value. The default value is 10,000 milliseconds.

3.14. Eye_WriteBarcode

This library creates a barcode from character data with the ‘Barcode tool.’

Property item	Description
Type	Specify a type of barcode.
Calculate check digits automatically	Select whether to calculate and replace check digits or not. This selection is used when EAN-6 or EAN-13 is specified for the type.
Data	Specify data from which to create a barcode.

Image resource name	Specify an image resource name for the output destination.
Timeout (ms)	Specify a timeout value. The default value is 10,000 milliseconds.

3.15. Eye_WriteQRcode

This library creates a QR code from character data with the ‘Barcode tool.’

Property item	Description
Level of error correction	Select a level of error correction.
Character encoding	Select a character encoding for the QR code.
Data	Specify data to create the QR code.
Image resource name	Specify an image resource name for the output destination.
Timeout (ms)	Specify a timeout value. The default value is 10,000 milliseconds.

3.16. Eye_OutputImageToFile

This library outputs an image resource of WinActorEye into a file.

Property item	Description
Image resource name	Specify an image resource name to output.
Image format	Select an output file format.
Filename	Specify an image filename for the output in an absolute or relative path. * The origin of the relative path is as follows: 1. When the work folder is set: the work folder. 2. When the work folder is not set: the folder in which the currently operated scenario file exists. 3. When the currently operated scenario has not saved: the folder in which currently running ‘WinActor.exe’ exists.
Adjust image type	Select whether to adjust the image type (bit depth) or not.

3.17. Eye_OutputImageToClipboard

This library outputs an image resource of WinActorEye into the clipboard.

Property item	Description
Image resource name	Specify an image resource name to copy.

Adjust image type	Select whether to adjust the image type (bit depth) or not.

3.18. Eye_Snapshot

This library inputs an image specified by clicking a window into WinActorEye.

Property item	Description
WinID name	Specify an application window for the input image by a click.
Image resource name	Specify an image resource name for the input destination.
Select the image	Specify whether to set the input image selected state or not.
Include the drop shadow	Select whether to include the drop shadow in the input image or not.

4. Appendix

4.1. Reference materials

Table 4.1-1 shows the materials referenced in this manual.

Table 4.1-1. Reference materials

No.	Material name
1	Tutorial.html
2	WinActor Operation Manual
3	WinActor User Library Sample Manual
4	WinActorEye Operation Manual
5	WinActor Note Operation Manual
6	WinActor Note Text Processing Scenario Creation Manual

WinActor®

WinActorEye
Scenario Creation Manual

NTT ADVANCED TECHNOLOGY CORPORATION

WA7-S-20250603