

A close-up photograph of a green leaf with a ladybug perched on its edge. In the background, there are small yellow flowers. The image is slightly blurred, creating a soft, naturalistic feel.

DEBUGGING

UIPATH ACADEMY

What to do before you delete everything, change your name, and run away to
the Philippines

DEBUGGING

DEBUGGING ACTIONS



- The process of detecting and resolving errors in a project that would cause it to behave unexpectedly or crash
- It takes a lot of patience and tenacity
- Usually done in dev

BREAKPOINTS

- During development, you can add breakpoints to certain activities in your automation
- When you do a debug run, the program pauses its execution when it hits a breakpoint
- Upon hitting a breakpoint, you can:
 - Step over to the next activity
 - Step into a workflow that is called
 - Step out completes the execution of activities in the current container, before pausing the debugging
 - Works well for nested sequences



PAUSING IN EXCEPTIONS

- When your automation pauses due to an exception you can:
 - **Retry**
 - Re-executes the previous activity, and throws the exception if it's encountered again
 - **Ignore**
 - Used to ignore an encountered exception and continue the execution from the next activity
 - **Restart**
 - Restarts debugging activity from the first activity of the project
 - When the restart option is clicked after using the run from this activity action, the debugging is restarted from the previously indicated activity



DEBUGGING ACTIVITIES

TEST ACTIVITIES

- **Run to this activity**
 - Starts the debugging process and pauses before the selected activity is executed
- **Run from this activity**
 - Enters the debugging in a pause state, allowing you to make changes to the values of the variables and arguments from the locals panel
 - Begins execution at selected activity
- **Test Activity**
 - Used for running a test on the currently selected activity
 - Only executes the selected activity

DEBUGGING PANELS

LOCALS PANEL
BREAKPOINTS PANEL
WATCH PANEL
IMMEDIATE PANEL
CALL STACK PANEL



- **Locals**
Shows: Exceptions, Arguments, Variables, Properties of previously executed activity, properties of current activity
- **Immediate**
Used for evaluating variables, arguments, or statements and inspecting data available at a certain point during debugging
Simply type the variable or argument name in the immediate window and press enter
- **Breakpoints**
Displays all breakpoints in current message
- **Watch**
Displays values of variables or arguments you want to monitor
- **Call Stack**
Displays the next activity to be executed and its parent containers
- **Output**
Display output of log message or writeline activities



QUESTIONS?

Comments? Concerns? Violent Reactions?

A background image of three damselflies perched on a thin brown branch against a blurred green background. The damselflies have large, prominent eyes and segmented bodies. The text 'DISCUSSION QUESTIONS' is overlaid on the left side of the image.

DISCUSSION QUESTIONS

- What is debugging?
- What is a breakpoint?
- Difference between run from activity vs run to activity vs test activity?
- Difference between retry and restart?
- What are the different debugging panels?