

Kofax Kapow 10.3 Training and Certification

Module 5 – Your First Robot

An introduction to robot building

**Kofax
Kapow™**



Module Overview

- ◆ Design Studio GUI
- ◆ Prerequisite Work
- ◆ Setting up Types and Attributes
- ◆ Creating a Robot
- ◆ Variables
- ◆ Load Page Step
- ◆ Extract Data Step
- ◆ Tag Finders
- ◆ Testing
- ◆ Debug Panel



Design Studio - Robot Editor

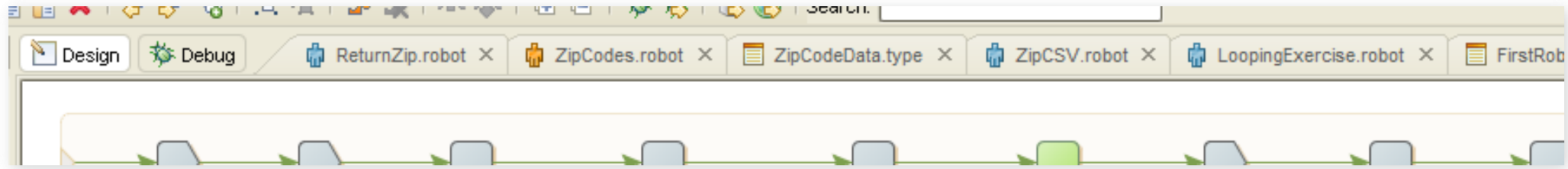
The screenshot displays the Design Studio - Robot Editor interface, which is used for creating and managing robots. The interface is divided into several panes and views, each highlighted with a blue callout box:

- Project View:** Located on the left side, it shows a tree structure of projects and components, including "HardyHardware", "Databases", "Device Mappings", "Robots", "zipWebService.robot", and "zipWebService2.robot".
- Robot View:** Located at the top center, it shows a sequence of steps in a robot workflow, including "Load Page", "Click Item", "Extract Name", "Extract Description", "Extract SKU", "Extract Aisle", "Extract Bay", and "Return Value".
- Browser View:** Located in the center, it displays a web browser window showing the "Hardy Hardware" website, specifically the "2 lb. Root Kill" product page.
- Source View:** Located at the bottom center, it shows the HTML source code of the web page being browsed, with elements like "hikashop_product_left_part" and "hikashop_product_right_part" visible.
- Variables View:** Located on the right side, it shows the variables used in the robot, including "HardyHardware" and "Return Value".
- Step View:** Located on the right side, it shows the details of a specific step in the robot workflow, including the "Return Value" and the "Variable" used.
- Database View:** Located on the left side, it shows the database structure and data, including "Management Console", "Development Database", "Design Studio Databases", and "Development Database".
- Shared Projects View:** Located on the left side, it shows the shared projects and their details, including "Management Console (localhost)" and "Default project".

Additional callouts include "Quick Access Tabs" at the top right and "Database View" at the bottom left.

Quick Access Tabs

- ◆ Just below the main toolbar, you have quick access to both the Design and Debug modes, as well as any currently open Robots and Types.
- ◆ Debug mode is used for testing your Robot.



Prerequisite Work



- ◆ Before you begin building a robot, you need to consider the following:
 - ◆ What do I need to extract?
 - ◆ Where is that data available?
 - ◆ How do I get to that data?
 - ◆ Is looping through multiple items or pages required?
 - ◆ Is the data I want embedded in data I don't need?
 - ◆ Does the data need to be formatted a certain way?
 - ◆ Are there conditions that determine whether I want the data or not?
 - ◆ And others
- ◆ One of the best ways to answer those questions is to go to the source and step through the extraction process as a human. Take notes as you go!

Your First Robot



- ◆ In this first simple robot, let's assume you want to get some basic information about the top featured item from a website called HardyHardware
- ◆ Specifically, you want to collect:
 - The item name
 - Its description
 - An overview of the product
 - Its current price
 - A small picture of the product
- ◆ Let's first step through the process as a human would...

Step through the detail as a human. Note what you did.

The screenshot shows a web browser window with the address bar containing the URL `http://class.kofax.com/hardyhardware/`. The website header features the "Hardy Hardware" logo and a cartoon character. The main content area is titled "Home" and displays a large image of a metal shelving unit. A blue callout box with the text "1. Load page by entering URL" points to the address bar. Another blue callout box with the text "2. Click on link for best rated product" points to a small thumbnail image of a light fixture in the bottom left corner of the main product image. On the right side of the page, there is a "Search" bar, a "Store Locator" section with an address input field, and a "Best rated" section featuring a product listing for a "2-Light Brushed Nickel Flush Mount (2-Pack)" with a price of \$22.97 each, a 5-star rating, and an "Add to cart" button.

`http://class.kofax.com/hardyhardware/`

1. Load page by entering URL

2. Click on link for best rated product

Note location of data you want to extract and record.

Clicking on the link takes you to a new page where you can see the data you want to extract.

Hardy Hardware

Home > Categories listing > 2-Light Brushed Nickel Flush Mount (2-Pack)

2-Light Brushed Nickel Flush Mount (2-Pack)

★★★★★ (10)

Common price: \$22.97 each

Our price: \$22.97 each

1 item in stock

1 + Add to cart

SKU 574296735 Aisle 37 Bay 006

Description

Commercial Electric 2-Light Brushed Nickel Flush Mount (2-Pack)

Product Overview

The Commercial Electric twin pack Flush Mount is a great look at a great value. From the professional to the do-it-yourselfer, the twin pack flush mount is an exceptional choice. Each Flush Mount is finished in Brushed Nickel with frosted white glass. Each Flush Mount uses two 60 watt medium-base light bulbs. From kitchen to bath or any room in between, the CE Brushed Nickel twin pack will not only enhance but will save cost to your home as well. * CE brushed

3. Name

4. Description

5. Overview

6. Price

7. Picture

Search

Search...

Store Locator

Address

Radius

25 Miles Search

Best rated

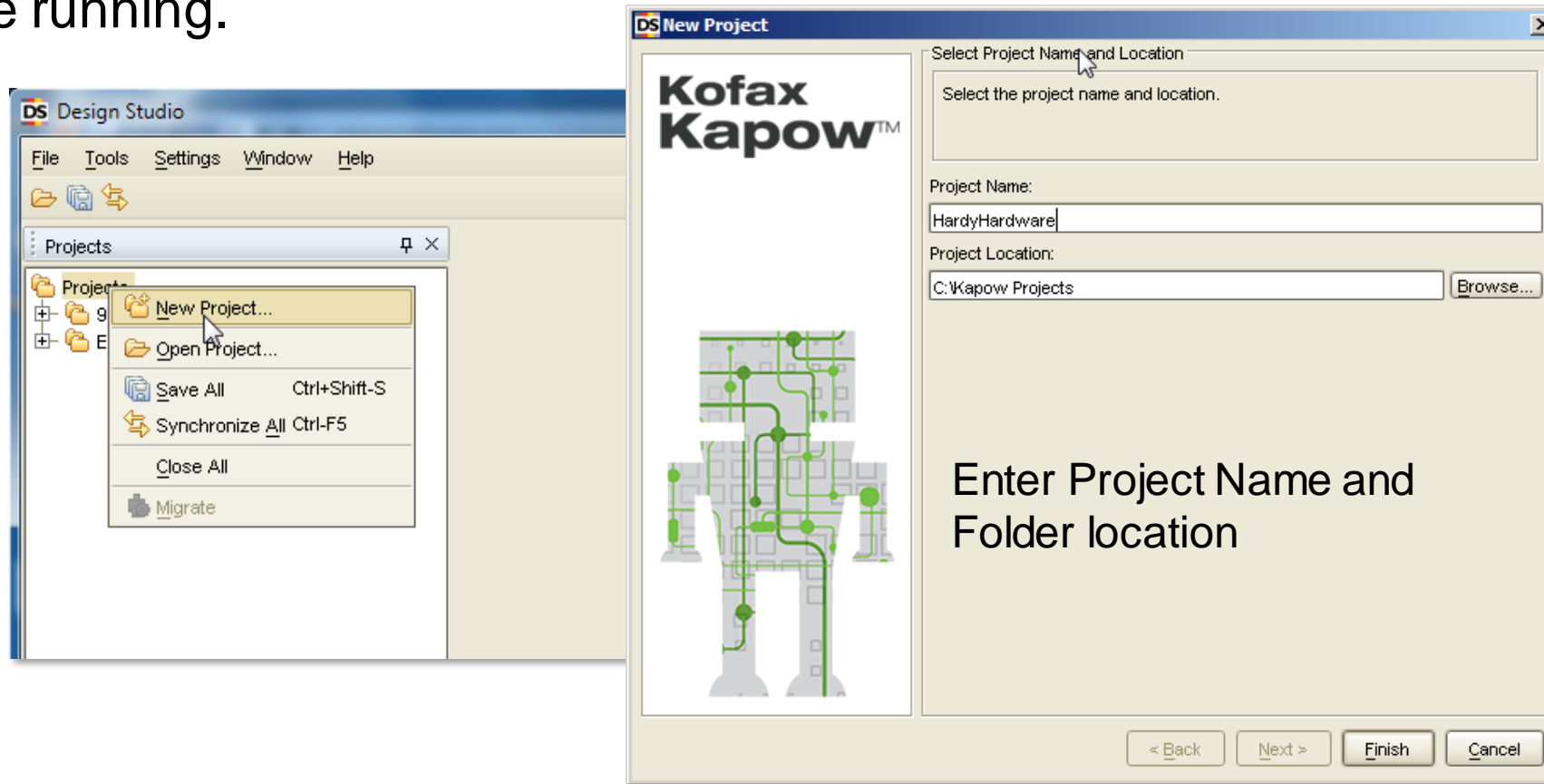
2-Light Brushed Nickel Flush Mount (2-Pack)

★★★★★ (10)

1 item in stock Add to cart

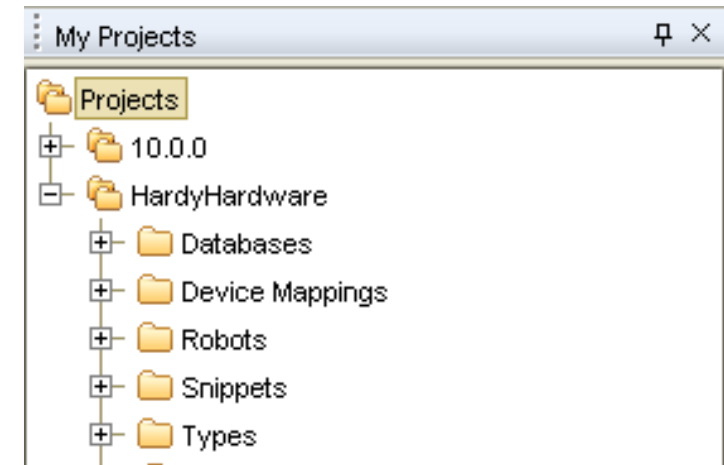
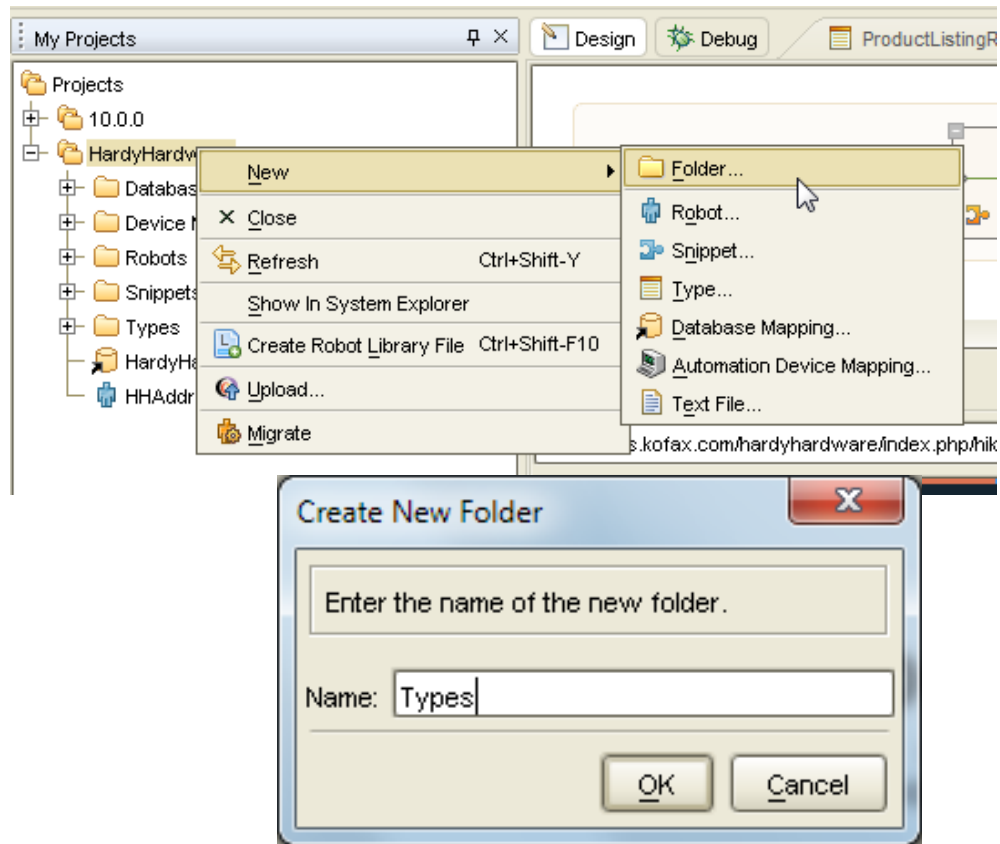
Open Design Studio and Create a New Project

- Remember, to open Design Studio, RoboServer and the Management Console must first be running.



Create Library Folders

- ◆ For your new project, create the folders that will contain things like Types, Robots, Snippets and Databases and Device Mappings

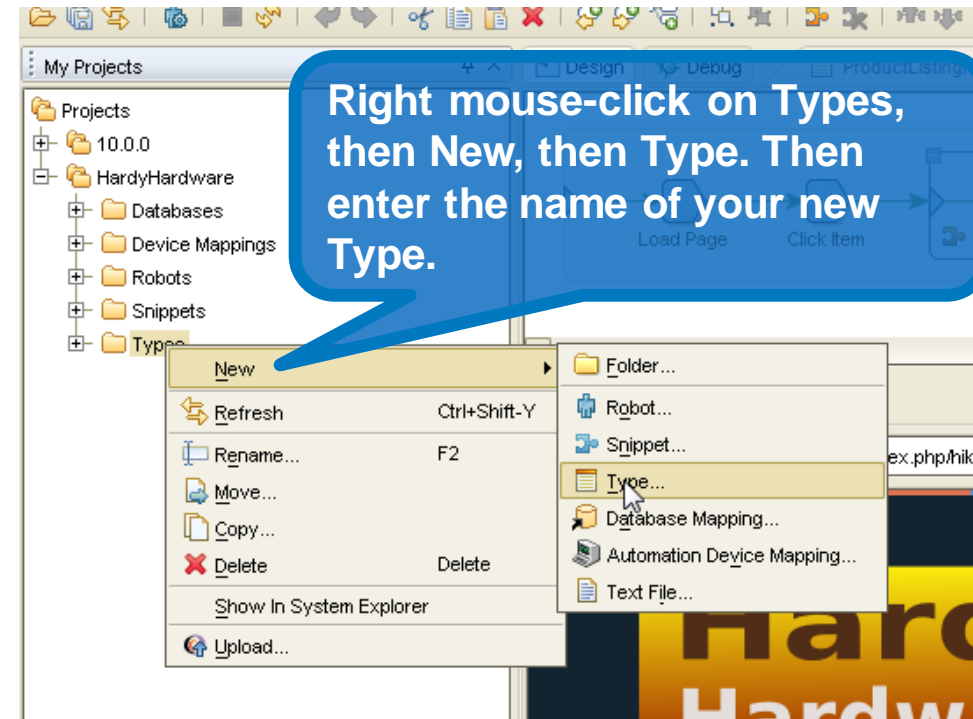


Creating project-specific folders for databases, robots, snippets, types and device mappings for each project will help you keep better organized.

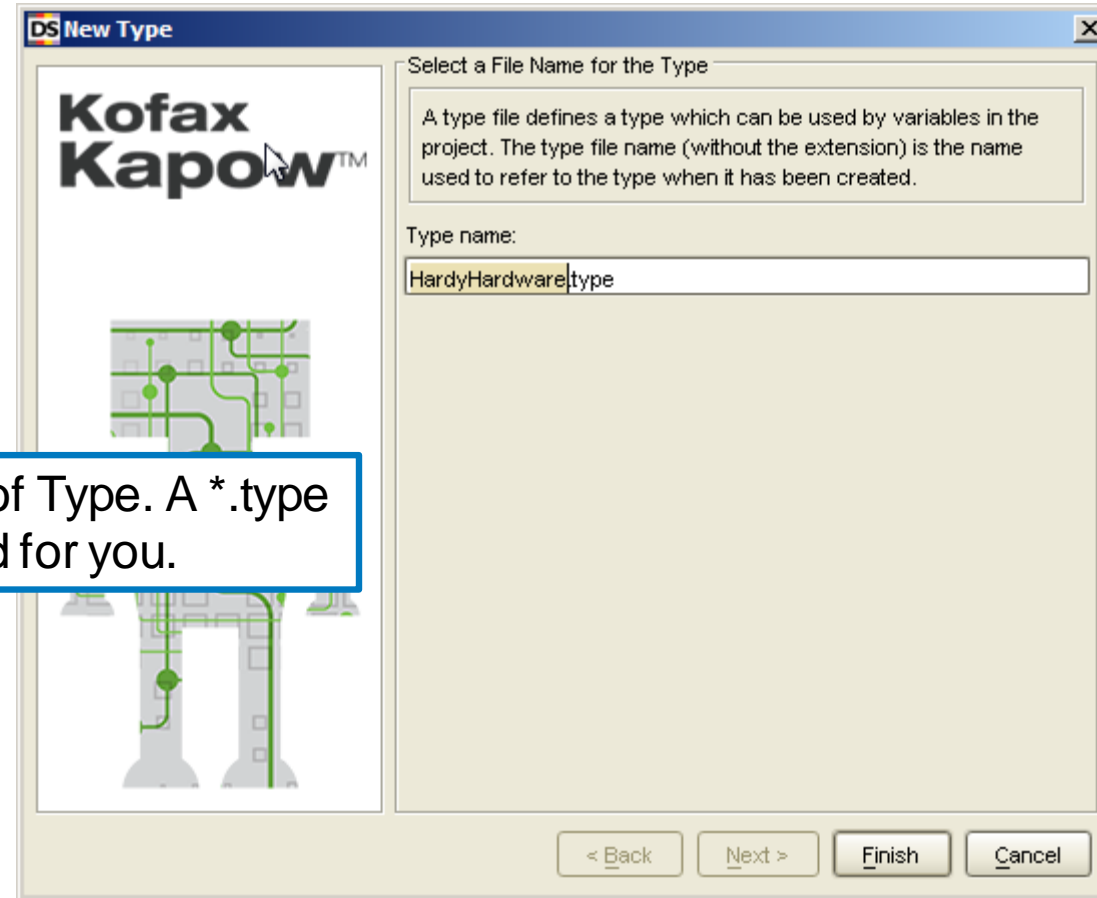
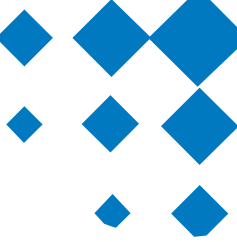
Creating Types

- Two important concepts in Design Studio are those of **Variables** and **Types**. When creating a Variable, it must be chosen which Type it should have. There are two kinds of Types: complex types and simple types. A **complex type** defines a set of **Attributes**. This expresses that each variable of a complex type denotes several (named) values.

- Before we begin building our Robot, we should build one complex Type to contain the data we want to handle. We will use it to create a Variable in our Robot. This will become more clear in a moment...



Creating Types (cont.)



Enter name of Type. A *.type file is created for you.

Attributes

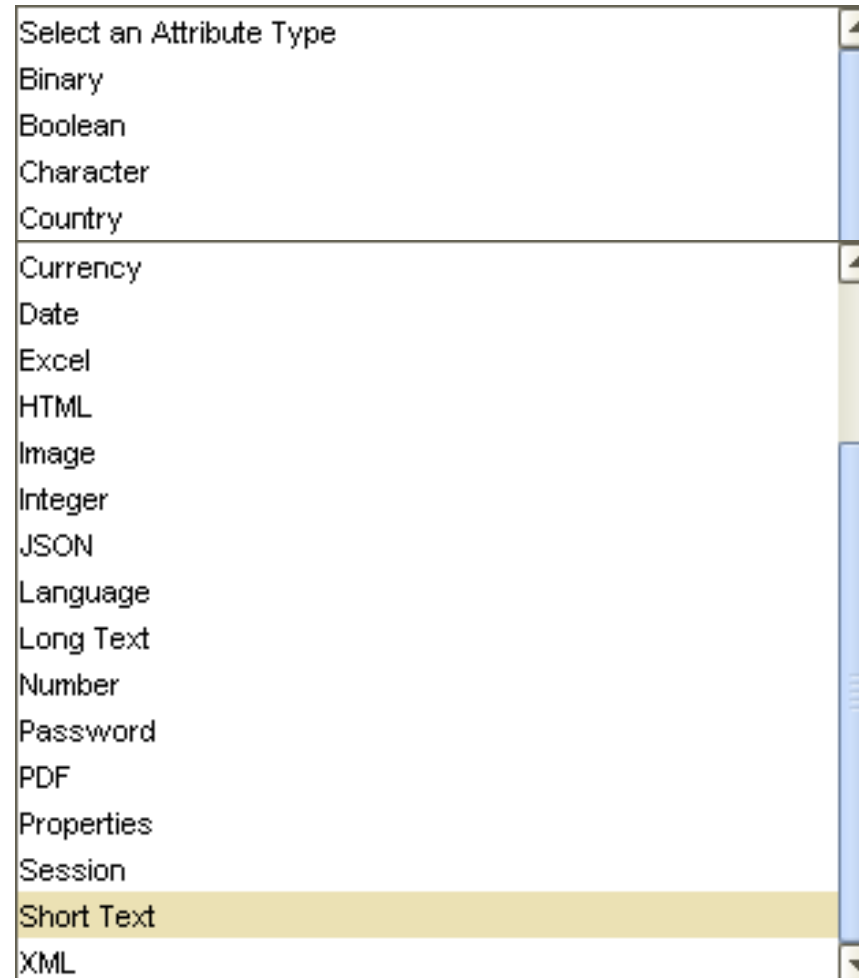


Click on the plus (+) symbol at the bottom of the panel to add a new Attribute. Select the type and optional default value. Decide whether the field is required and if it will be part of the database key created by Kapow.

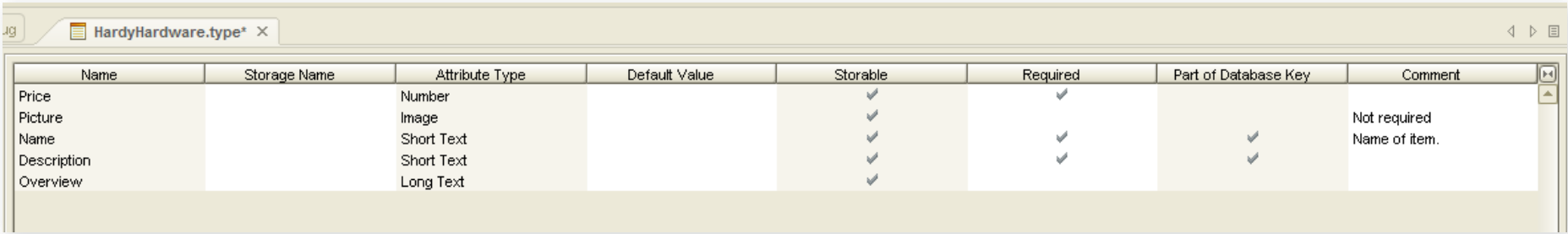
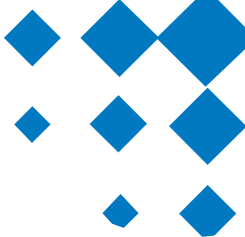
The screenshot shows the 'DS Design Studio' window for 'HardyHardware.type'. The 'Attributes' panel is open, displaying a table with columns: Name, Storage Name, Attribute Type, and Default Value. The 'Add Attribute' dialog box is open, showing the 'Basic' tab. The 'Name' field is 'Name', 'Type and Default Value' is 'Short Text', 'Required' is checked, 'Part of Database Key' is checked, and the 'Comment' is 'Name of item.'. The 'Add Attribute' dialog is overlaid on the 'Attributes' panel. A blue box highlights the plus (+) symbol at the bottom of the 'Attributes' panel, with a tooltip that reads 'Add and Configure New Attribute Ctrl-Insert'.

Attributes (cont.)

- ◆ The following attributes are available in Kapow.



Attributes added to Types



Name	Storage Name	Attribute Type	Default Value	Storable	Required	Part of Database Key	Comment
Price		Number		✓	✓		
Picture		Image		✓			
Name		Short Text		✓	✓	✓	Not required
Description		Short Text		✓	✓	✓	Name of item.
Overview		Long Text		✓			

All the attributes we are interested in have been added and set. We will use these when we create our Robot to add a Variable.

Reorder these as necessary using the up/down arrows at the bottom of the screen.



+

-

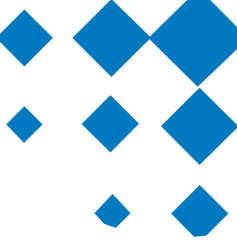
↶

↷

✎

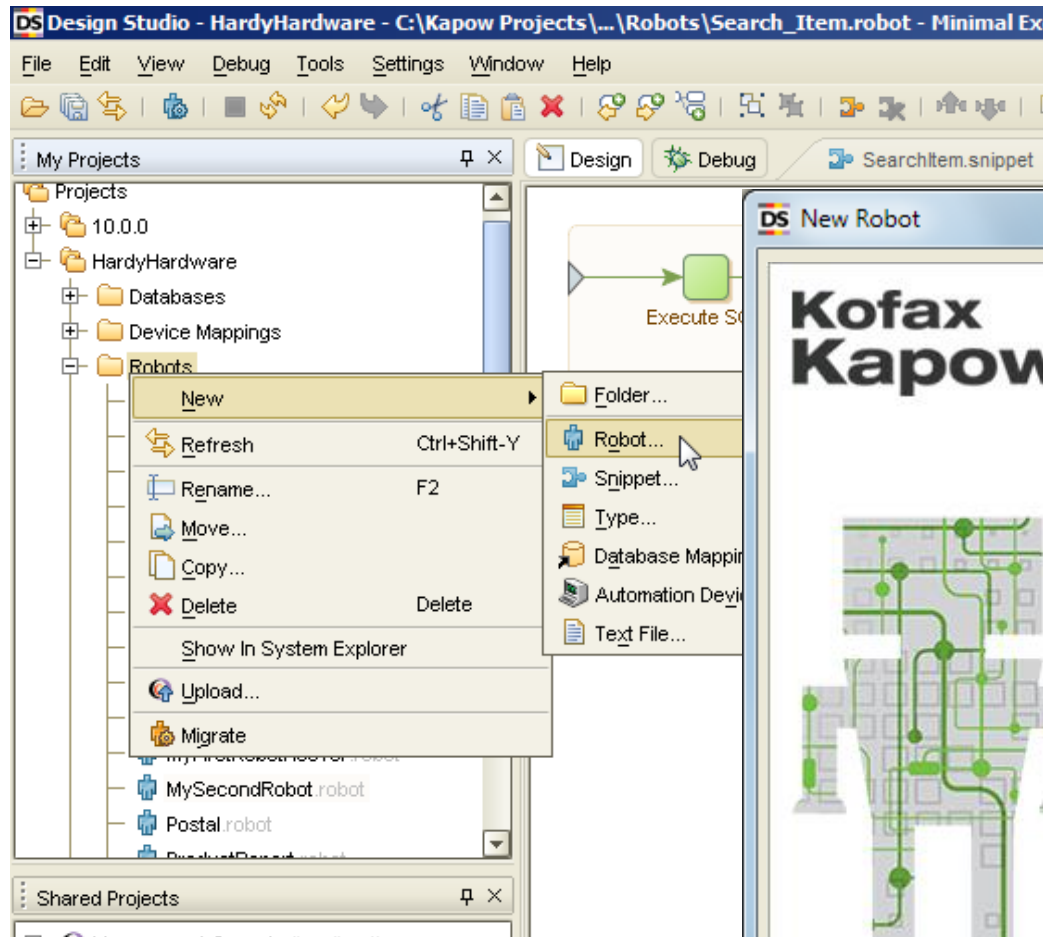
Name	Storage Name	Attribute Type	Defa
Name		Short Text	
Description		Short Text	
Overview		Long Text	
Picture		Image	
Price		Number	

Types Explained

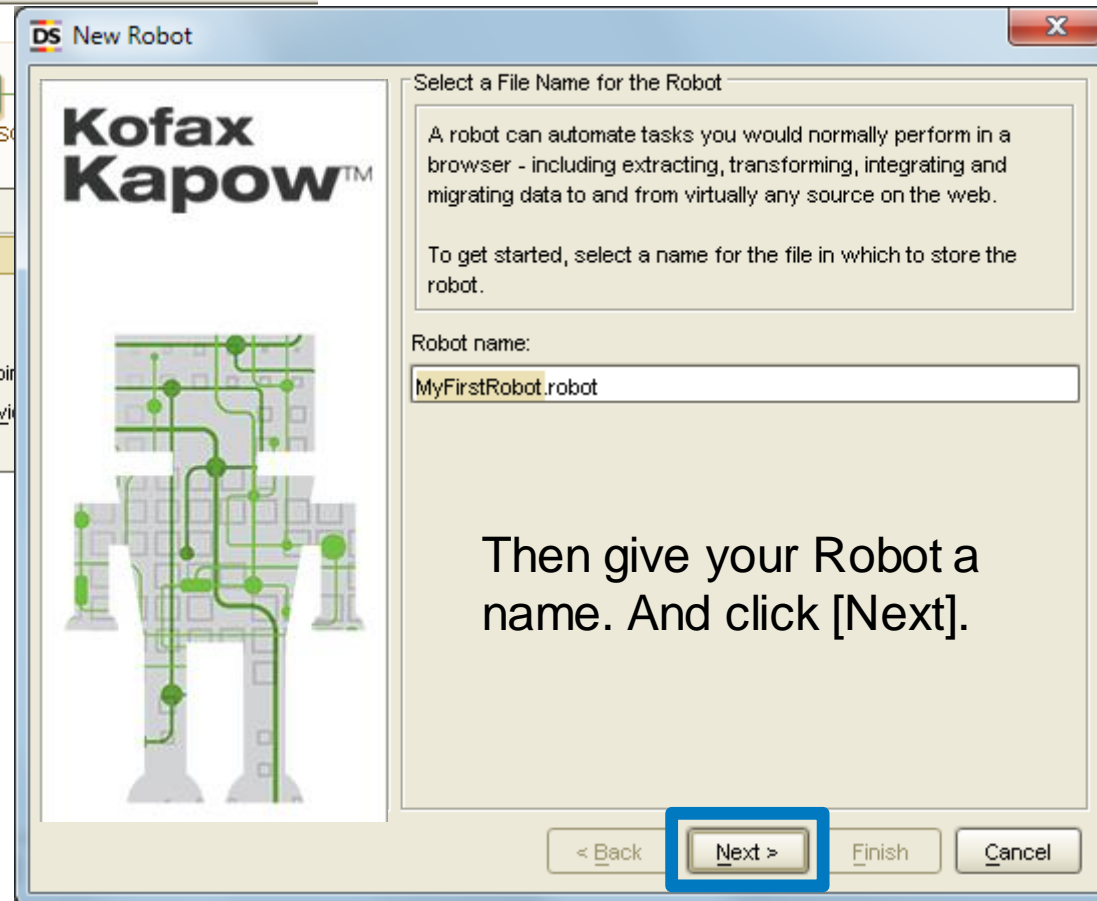


- ◆ In Kofax Kapow, variables are used by Robots as containers for data. Robots use variables as input, output or to store temporary values during execution. These variables are categorized by the kind of data they will contain. Examples might be text, image, PDF, number and so on. These examples are all what we call Simple Types. They are predefined in Design Studio and set up in the main interface on the Variables panel. These are typically used during execution of a Robot and cannot be used for input or output.
- ◆ Alternately we can create our own Complex Types. Think of a Complex Type as a bucket of simple types. They are set up as we just demonstrated in the preceding slides in special Types window in Design Studio and are stored as a file with a .type extension. Complex Types may be used as input and output variables.

Creating a Robot



Right mouse-click on Robots in the Project Panel and select New | Robot from the context menus.



Enter URL of Start Page and Select Browser Engine



Enter the URL to start from

Enter the URL that you want the robot to start from. If you don't want the robot to start by loading from a URL, leave the field blank.

Enter the URL that the robot should start from:

Browser Engine:

Default

Execution Mode:

Minimal Execution (Direct)

Back Next

The Default browser engine is the newer of the two and has some performance advantages. But sometimes, you might find that it doesn't work as well as the older "Classic" engine. You can always select one or the other here, or migrate between the two as you're testing to see which provides you with the best results.

When you click a step in Minimal Execution mode in the robot graph, Design Studio takes the shortest direct path to that step, skipping any previous branches and iterations that are not on the direct path.

In Smart Re-execution mode, the way that the robot is executed in Design mode is similar to the way it is executed at runtime or in Debug mode.

Load Page Step and End Step Automatically Created

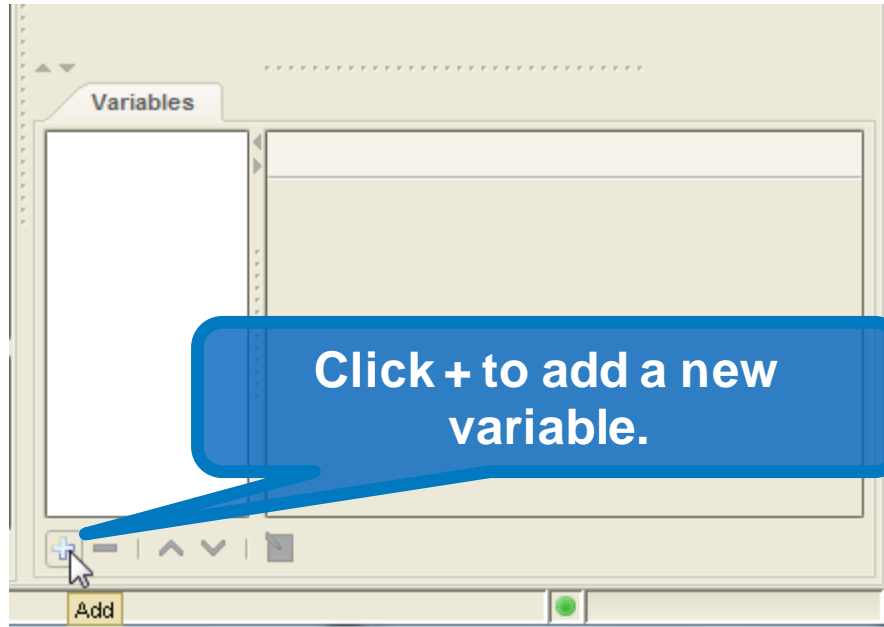
The screenshot displays the Kofax Design Studio interface. On the left, the 'Projects' pane shows a tree structure with 'MyFirstRobot.robot' selected. A blue callout bubble points to it with the text 'Robot created'. The main workspace shows a 'Load Page' step in the design flow, with a blue callout bubble stating 'NOTE: End Step is automatically selected, which means Load Step has already occurred.' Below this, a browser window displays a web page titled 'Hardy Hardware' with a cartoon character. A blue callout bubble points to the page with the text 'Page is loaded.' The bottom pane shows the HTML code for the page, starting with '

Robot created

NOTE: End Step is automatically selected, which means Load Step has already occurred.

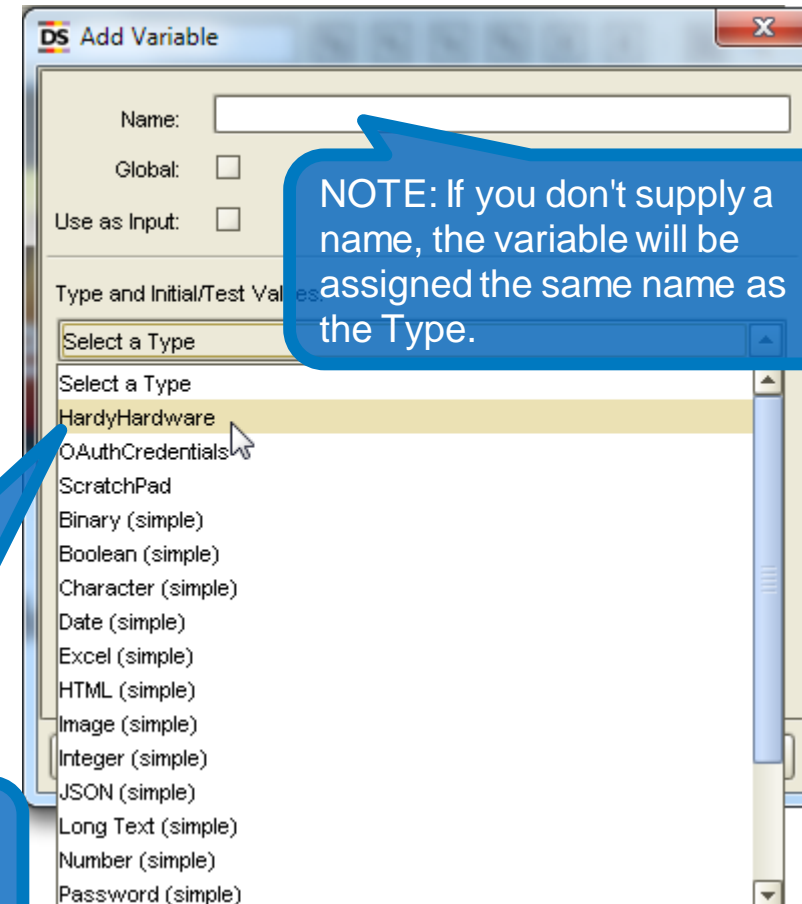
Page is loaded.

Add Variables



Click + to add a new variable.

Go to the Variables panel located in the bottom right window of Design Studio.



NOTE: If you don't supply a name, the variable will be assigned the same name as the Type.

Select a Type. Here, we're selecting the HardyHardware type we just created.

Variable has been added

Red dots before attribute name mean that a value for that attribute is required.

The screenshot shows the Kofax Design Studio interface for a project named 'MyFirstRobot.robot'. The left sidebar contains a 'Projects' tree with folders for 'Exercises', 'HardyHardware', 'Databases', 'Robots', 'Snippets', and 'Types'. The 'HardyHardware' folder is expanded, showing 'MyFirstRobot.robot' and 'HardyHardware.type'. The central design canvas displays a web page layout for 'Hardy Hardware' with a header, navigation links ('Home', 'Products listing', 'Categories listing'), and a main content area. The right-hand properties panel shows the 'Variables' section for the 'HardyHardware' type, with attributes like 'Price', 'Name', 'Description', and 'Overview'. Red dots are visible next to 'Price', 'Name', and 'Description', indicating that values are required for these attributes. A blue callout box points to these red dots with the text: 'Red dots before attribute name mean that a value for that attribute is required.'

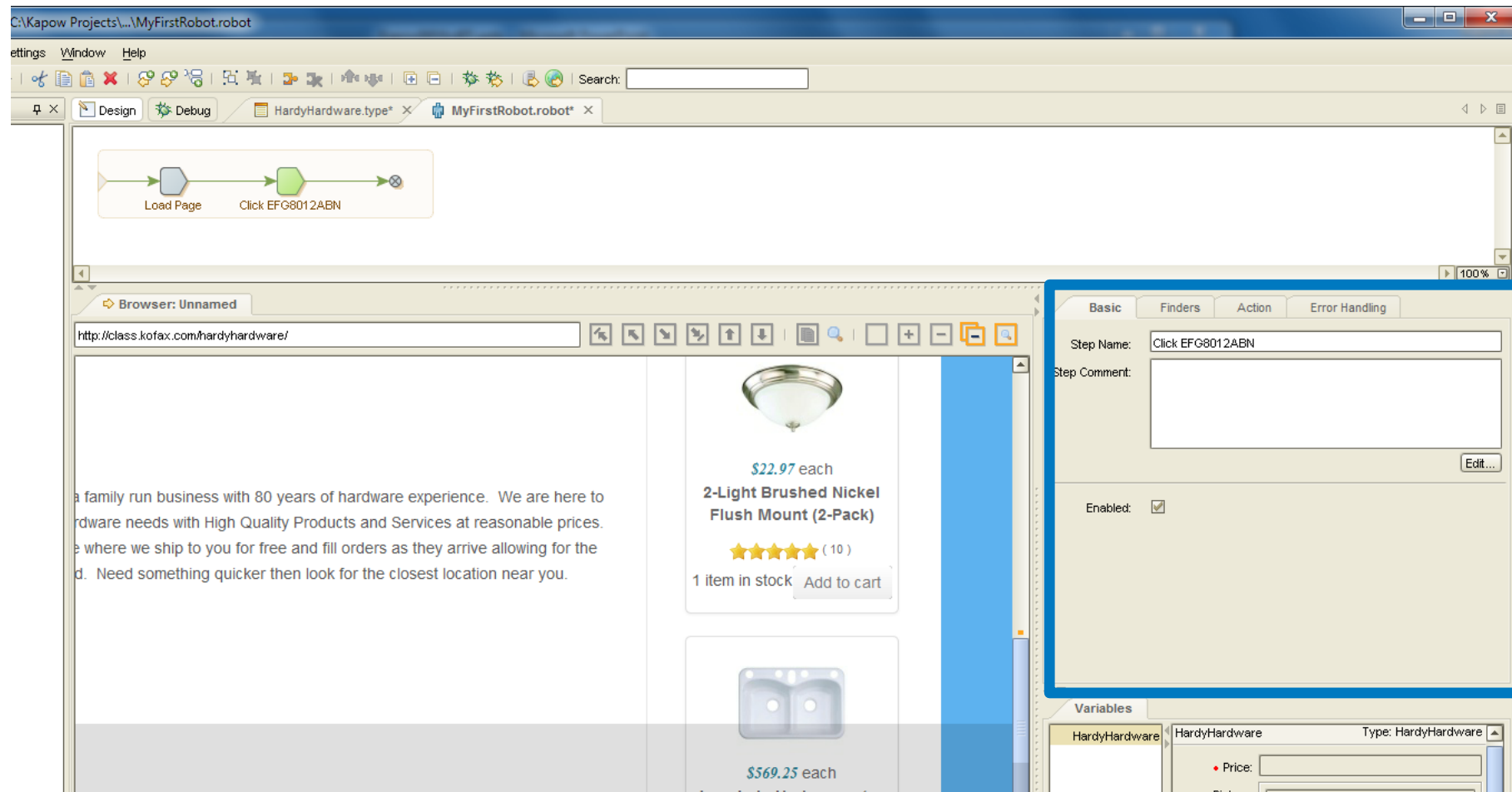
Click Step

End step automatically selected

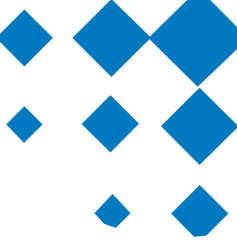
Our first step after loading the page is to click on the item. Select the item with your left mouse button. Then right mouse click and select "Click" from the context menu. The step will be added before the currently selected step in the robot view panel above.

Item selected.

Click Step has been Created



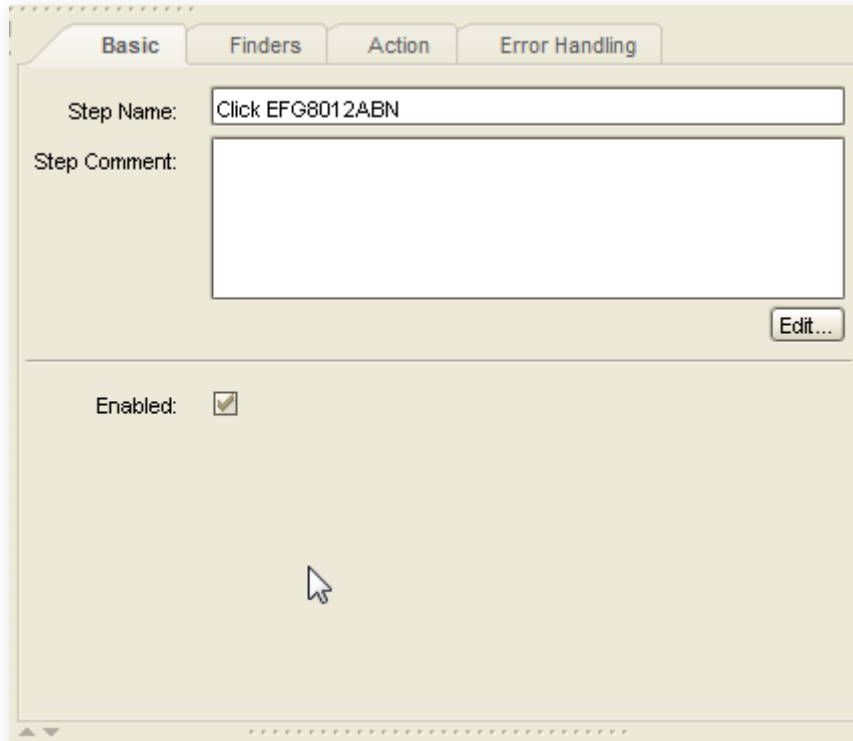
Steps - The Parts that Make up a Step



- ◆ Basic
 - ◆ Name and description of step. Steps are initially automatically named by system
- ◆ Tag finders
 - ◆ locate page content by finding HTML tags
- ◆ Action
 - ◆ action to perform
- ◆ Error handling
 - ◆ Manage errors
 - from both tag finders and actions

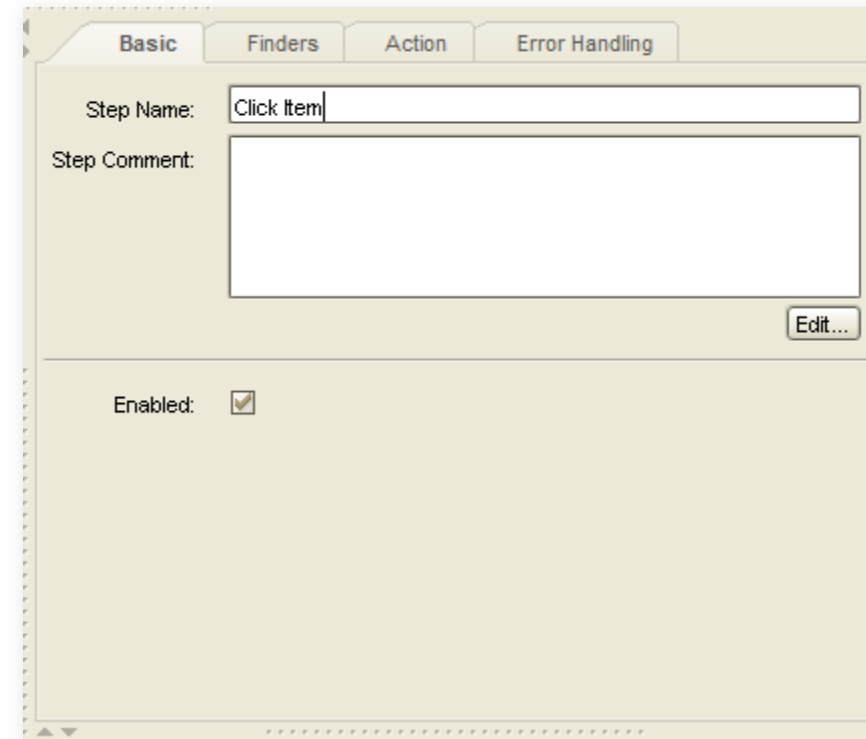
Basic Tab

Original



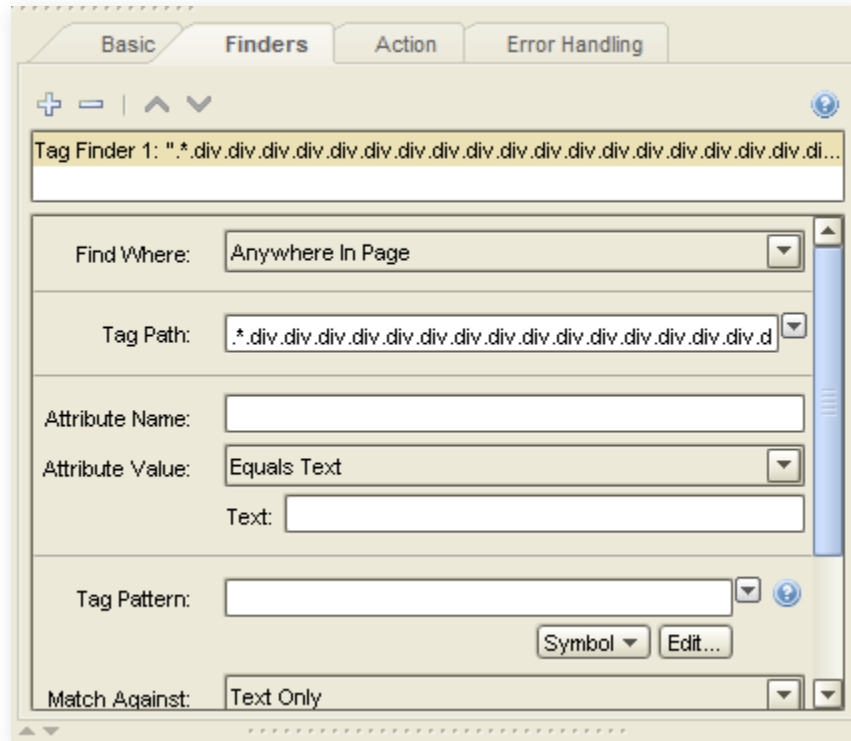
The image shows a software configuration window with four tabs: Basic, Finders, Action, and Error Handling. The Basic tab is selected. It contains a 'Step Name' field with the text 'Click EFG8012ABN', a 'Step Comment' text area, and an 'Edit...' button. Below these is an 'Enabled' checkbox which is checked. A mouse cursor is visible near the bottom of the window.

In this case, we have renamed this click step to be more generic.



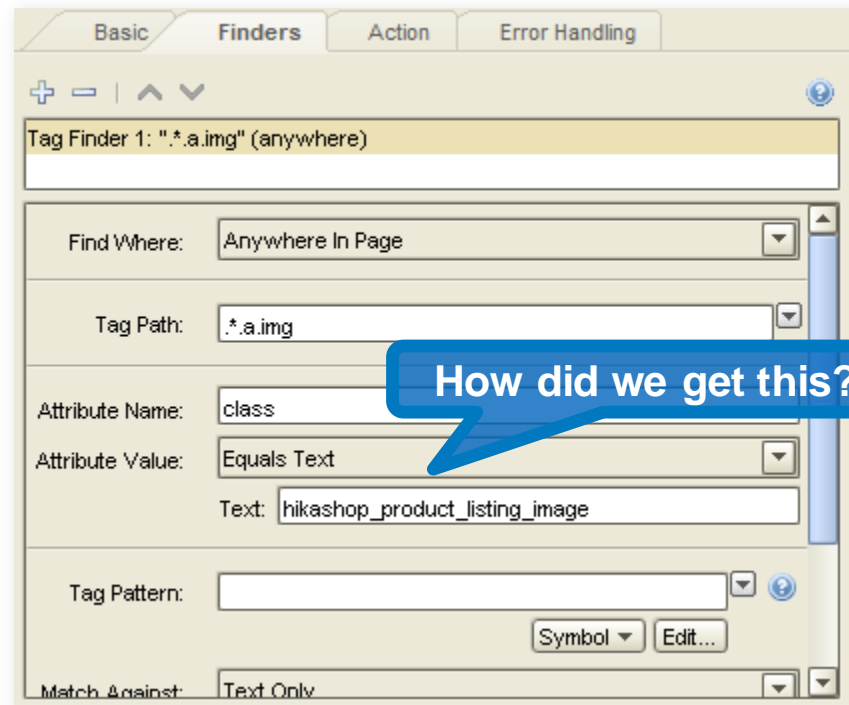
The image shows the same software configuration window, but the 'Step Name' field has been changed to 'Click Item'. The 'Step Comment' text area and the checked 'Enabled' checkbox remain the same. The 'Edit...' button is still present.

Tag Finders automatically populated...BUT...




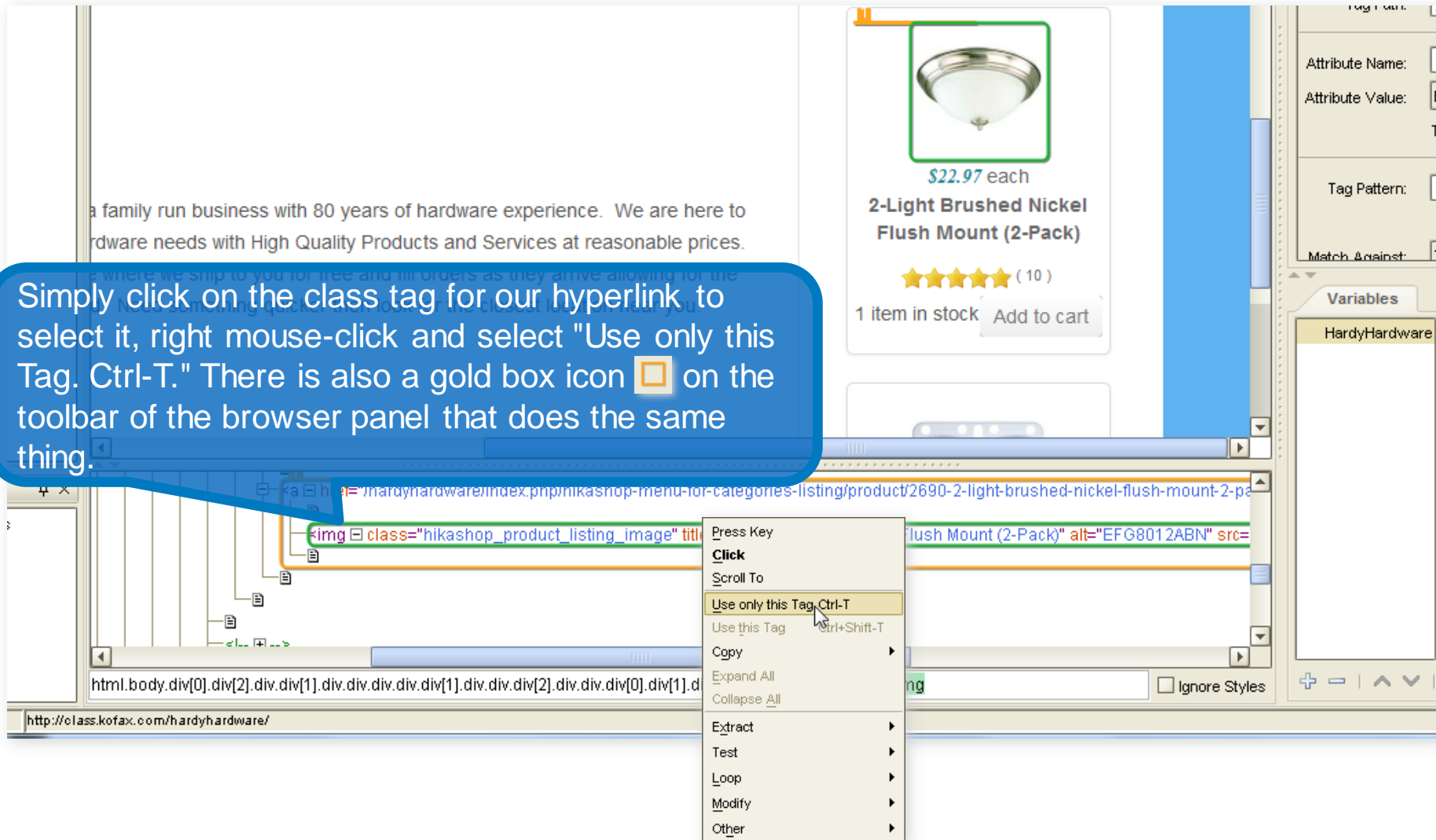
This way is not only much simpler, but much more robust. It's looking for the *.a.img tag only...and only if the class tag is equal to hikashop_product_listing_image.

...notice all the <div> tags returned. Not only is this unnecessarily complicated, but if the website changes, you stand a good chance of not getting what you want.

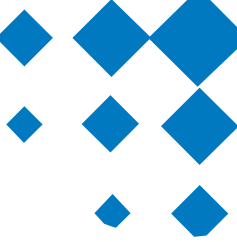


Look in the HTML Code for a Unique Tag

Simply click on the class tag for our hyperlink to select it, right mouse-click and select "Use only this Tag. Ctrl-T." There is also a gold box icon  on the toolbar of the browser panel that does the same thing.



A Little Bit about Tag Finders



- ◆ A tag finder is used to find **one** tag
- ◆ Tag finder usage:
 - ◆ Tag finders are only used on step actions for which a tag needs to be found on a page, such as for data extraction, setting a current tag, mimicking a mouse action or defining a selection in which to loop
 - ◆ Examples:
 - **Load Page:** URL inserted by developer - no tag finder
 - **Click:** The button to click - one tag finder, finds one tag
- ◆ A tag finder will by default always select the **first** occurrence that meet all the criteria

Tag Finder: Tag path



◆ Nodes

- ◆ * (star) means any tag(s)
- ◆ tag name *that* specific tag
- ◆ tr[3] (brackets) refers to the occurrence of a specific tag when multiple in same parent tag
- ◆ .text select only text inside tag
- ◆ .comment select comment tag

◆ Other

- ◆ | (or) multiple nodes
- ◆ In context of named tags, the path is relative

Tag Path Example

```
<html>
  <body>
    <p>
      <a id="gokapow" href="http://kapowsoftware.com">Go</a>
    </p>
  </body>
</html>
```

- Different ways to find the <a> tag
 - "html.body.p.a" Absolute path
 - "html.*.a" First <a> inside <html>
 - "*.p.a" First <a> inside a <p>
 - "*.a" First <a>
 - "a" Short hand for "*.a"

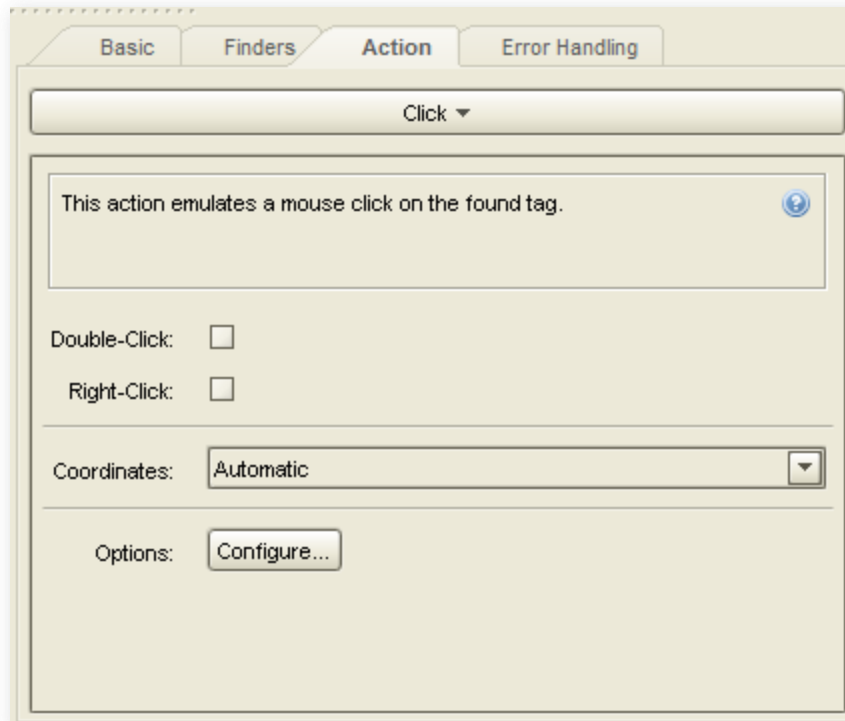
Colored Boxes and Tags

- ♦ Green box marks the **current** selection
- ♦ Gold box marks a **found** tag (for the action)
- ♦ Blue box marks a **named** tag (for the tag finder)

1	539 S MISSOURI AVENUE CLEARWATER FL 33756 - (727)442-2887	1.85 mi. FREESTANDING	✓	✓	✓	✓
2	1934 N HERCULES AVE CLEARWATER FL 33763 - (727)449-0220	2.32 mi. FREESTANDING	✓	✓	✓	✓
3	1860 GULF-TO-BAY BLVD. CLEARWATER FL 33765 - (727)442-1286	2.73 mi. FREESTANDING	✓	✓	✓	✓
4	1645 MAIN STREET DUNEDIN FL 34698 - (727)734-3166	4.09 mi. FREESTANDING	✓	✓	✓	✓
5	347 S GULFVIEW BLVD CLEARWATER BEACH FL 33767 - (727)461-7054	5.16 mi. FREESTANDING		✓	✓	

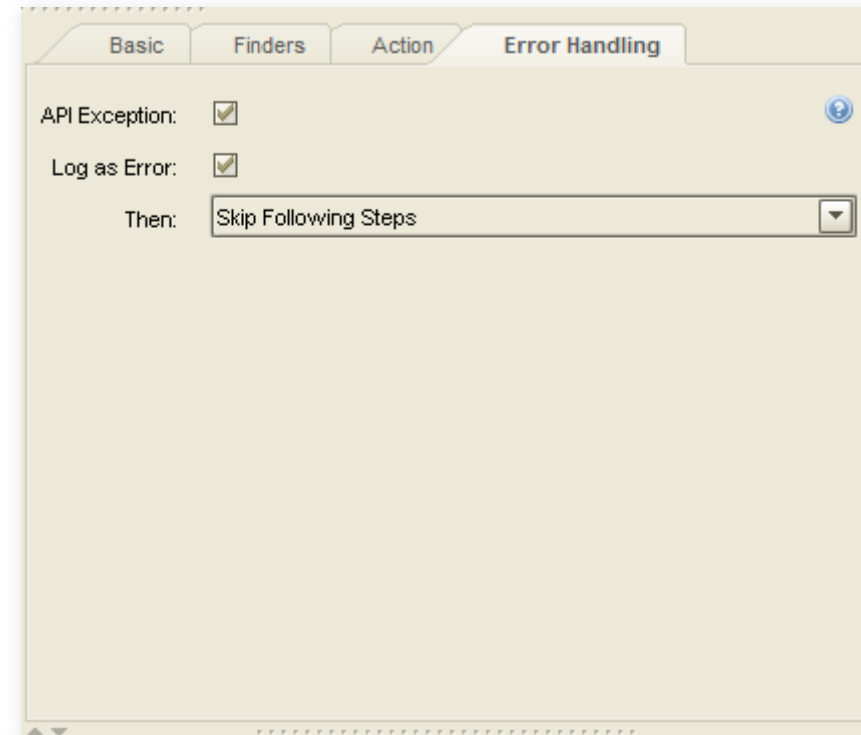
1/7A

Actions and Error Handling Tabs

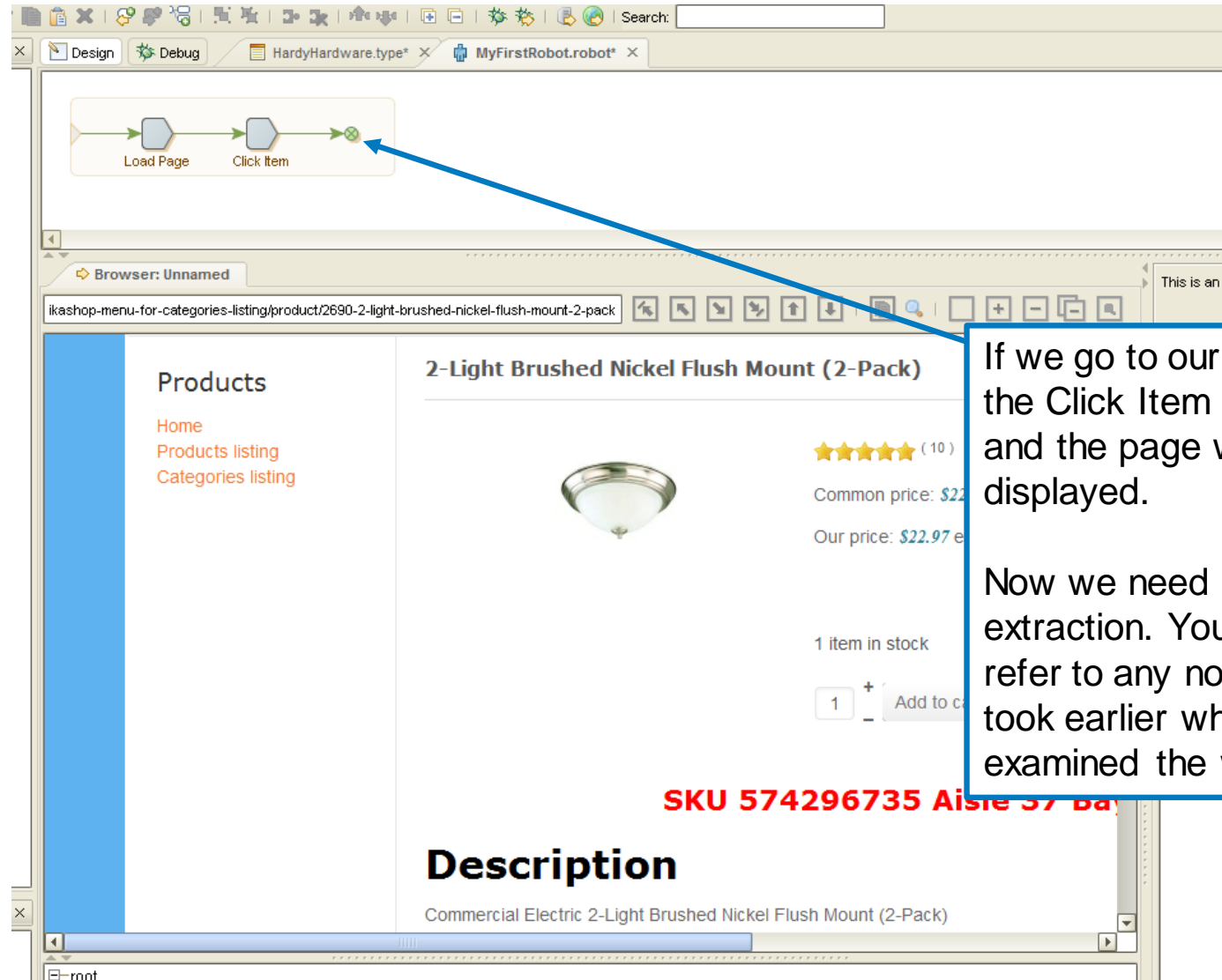


A simple click is all we need to do. The defaults are fine as is.

Likewise, the default error handling is fine as is. If the robot failed at this point, it would skip any steps that follow and return an error.



Extract Steps



If we go to our end step, the Click Item step is run and the page we want is displayed.

Now we need to set up extraction. You should refer to any notes you took earlier when you examined the website.

Extract Name

Browser: Unnamed

ikashop-menu-for-categories-listing/product/2690-2-light-brushed-nickel-flush-mount-2-pack

Products

Home
Products listing
Categories listing

2-Light Brushed Nickel Flush Mount (2-Pack)

Press Key
Scroll To
Use only this Tag Ctrl-T
Use this Tag Ctrl+Shift-T
Copy
Extract
Test
Loop
Modify
Other

With XML Data Mapper...
Text
Structured Text
HTML
Image
Screenshot
Number
Date
Attribute
URL
Target

HardyHardware.Name
HardyHardware.Description
HardyHardware.Overview
New Variable of Simple Type...
New Variable of Complex Type...

Common price: \$22.97

SKU 574296735 AISIE 37 Bay

Description

Commercial Electric 2-Light Brushed Nickel Flush Mount (2-Pack)

This is an End step

Variables

HardyHardware

As with the Click step, left mouse-click on the item you want to extract. From the context menus select "Extract" | "Text" | "HardyHardware.Name" (the attribute belonging to the Variable you are extracting to).

Repeat for the other Items you wish to Extract

- ◆ Description
- ◆ Overview
- ◆ Price
- ◆ Picture

The screenshot displays a web browser window showing a product page for '2-Light Brushed Nickel Flush Mount (2-Pack)'. The page includes a navigation menu, product details, a price of \$22.97, and an 'Add to cart' button. A red text overlay reads 'SKU 574296735 Aisle 37 Bay'. Below the browser window, a workflow diagram shows a sequence of steps: Load Page, Click Item, Extract Name, Extract Description, Extract Overview, Extract Price, and Extract Image. The 'Variables' panel on the right shows the extracted data for the product, including Name, Description, Overview, Price, and Picture.

Workflow steps:

- Load Page
- Click Item
- Extract Name
- Extract Description
- Extract Overview
- Extract Price
- Extract Image

Variables:

Variable	Type	Value
HardyHardware	HardyH	
Name		2-Light Brushed Nickel Flush Mount
Description		Commercial Electric 2-Light Brushed
Overview		Product Overview The Commercial Electric twin pack F Mount is a great look at a great value. From the professional to the do-it-yourselfer, the twin pack flush
Price		22.97
Picture		No Image.

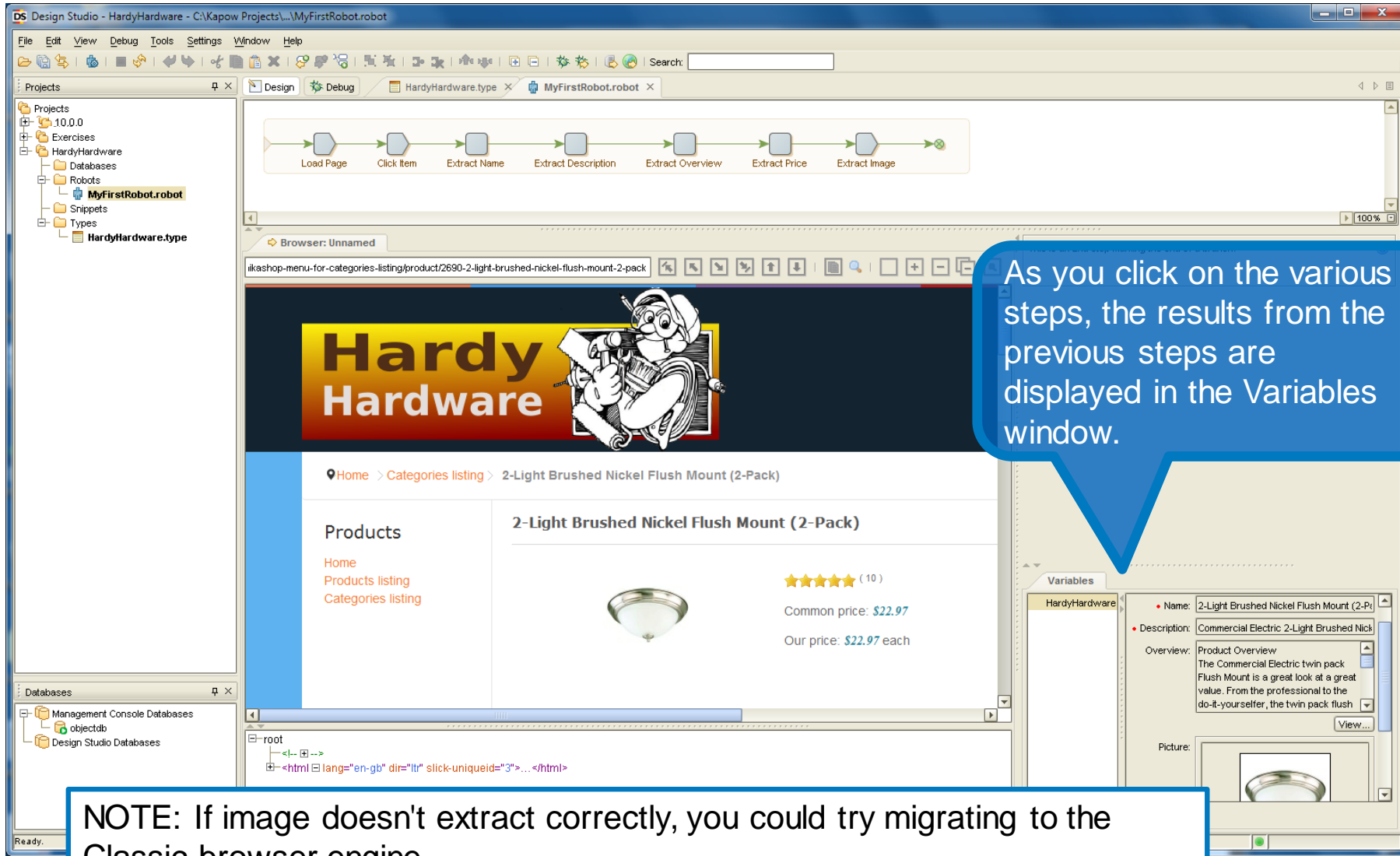
Tags and Tag Finders

- Remember to simplify your tag finders if necessary. You don't need a gazillion <div> tags. The .* symbol is a wildcard for anything that precedes the last tag in the Tag Path. Look for a unique attribute to help if possible. This is much easier to read and more resistant to changes in the website as well.

The screenshot shows a software interface with four tabs: 'Basic', 'Finders', 'Action', and 'Error Handling'. The 'Finders' tab is active. It contains a list of tag finders, with 'Tag Finder 1: ".*.span" (anywhere)' selected. Below the list, there are several configuration fields:

- Find Where:** A dropdown menu set to 'Anywhere In Page'.
- Tag Path:** A text input field containing '.*.span'.
- Attribute Name:** A text input field containing 'class'.
- Attribute Value:** A dropdown menu set to 'Equals Text'.
- Text:** A text input field containing 'hikashop_product_price hikashop_product_price_0'.

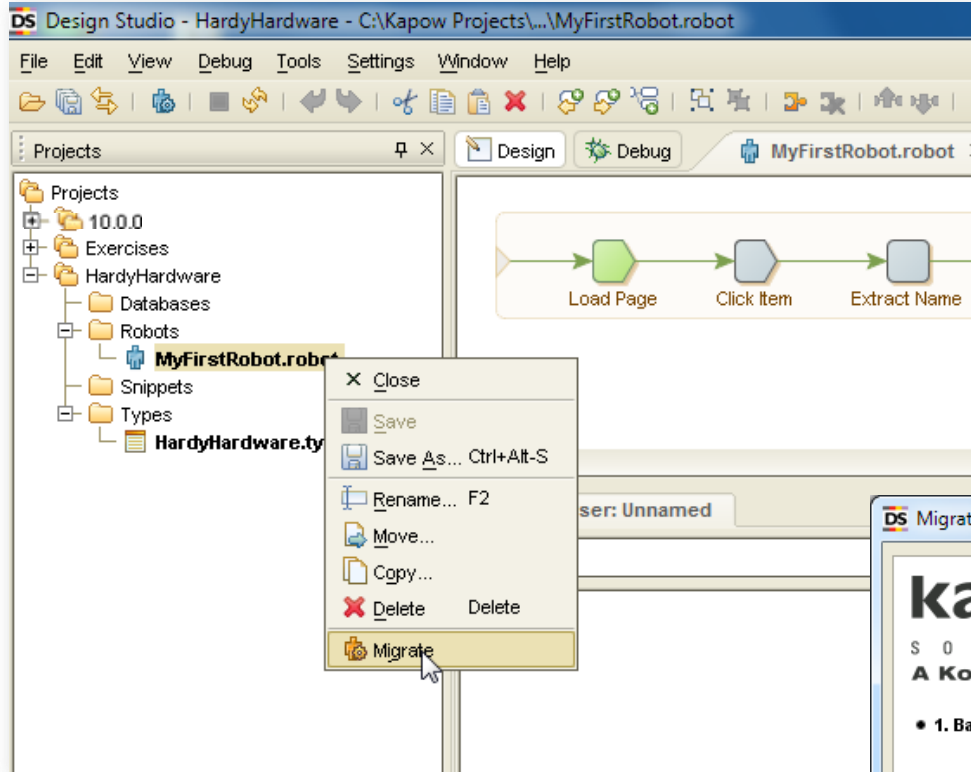
Testing



As you click on the various steps, the results from the previous steps are displayed in the Variables window.

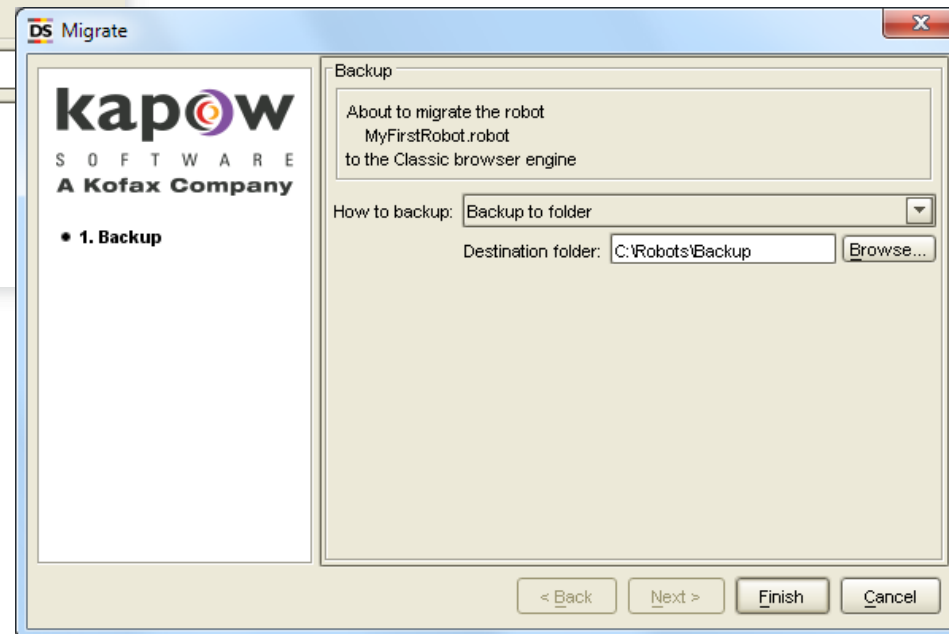
NOTE: If image doesn't extract correctly, you could try migrating to the Classic browser engine

Migration to another Browser Engine

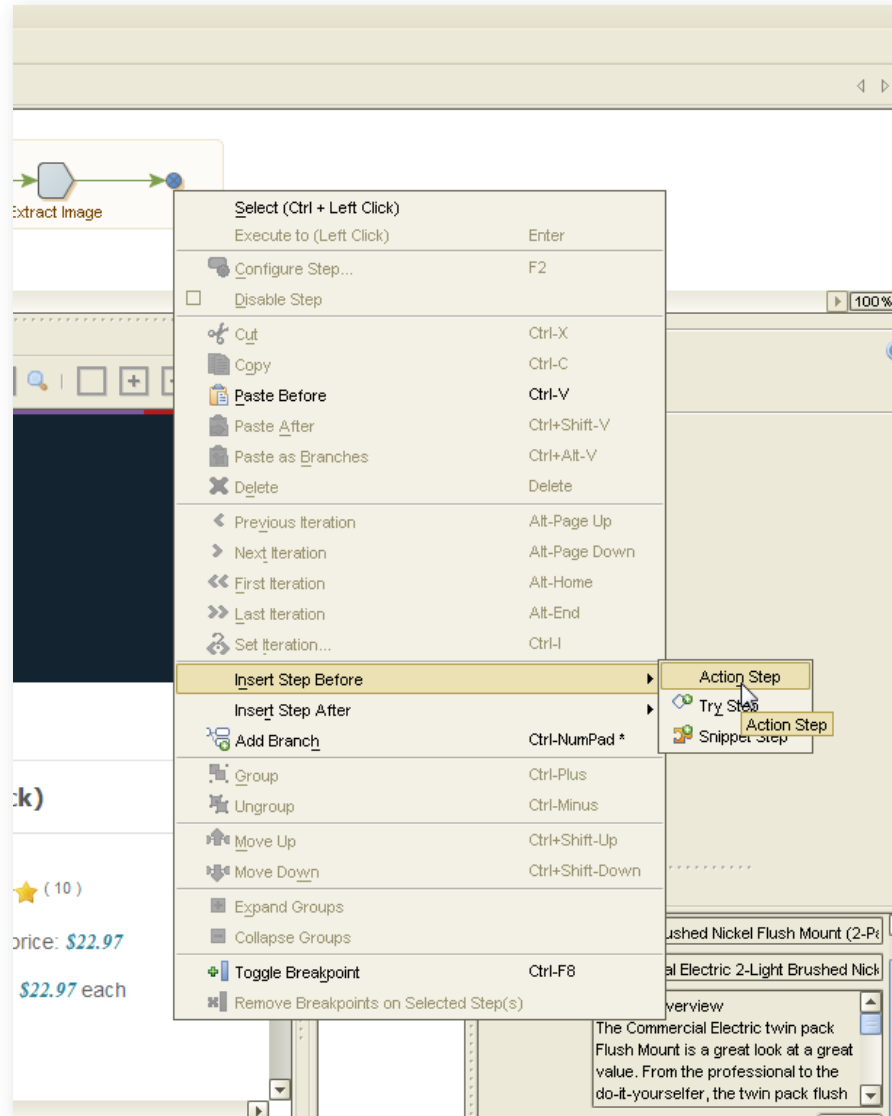


"Migrate" will use the engine not currently selected. You may migrate back and forth. Make sure to test your Robot after migrating. **NOTE:** Before migrating, make sure you've saved your Robot.

You may optionally backup your current Robot to any folder you specify.

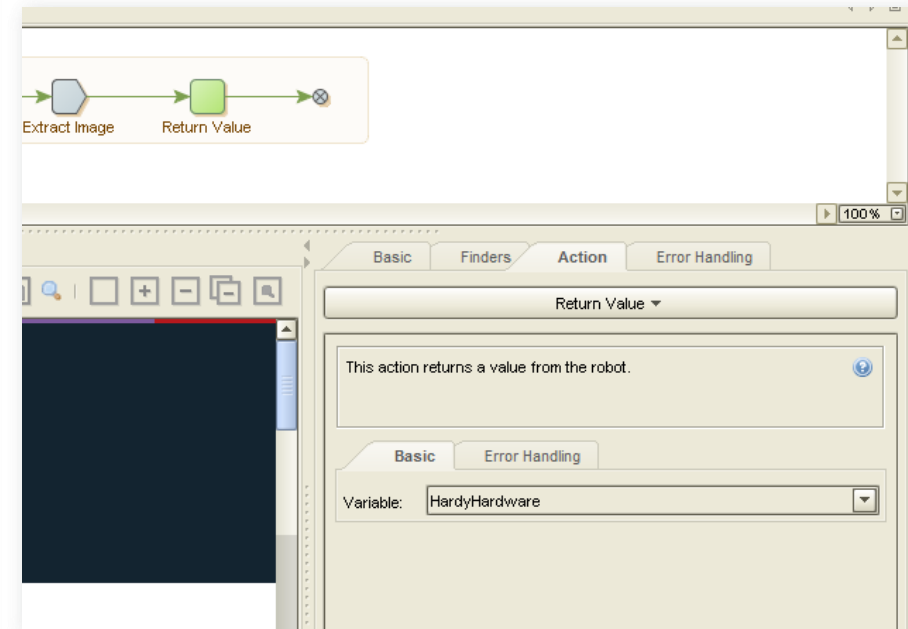


Debugging



To run our Robot in Debug mode and be able to view the values returned, we'll add an action step just before the end step.


For the Action, we'll select "Return Value."



Running in Debug Mode

Click on [Debug], then "Run."

The screenshot displays the Design Studio interface for a project named "MyFirstRobot.robot". The top menu bar includes File, Debug, View, Breakpoints, Tools, Settings, Window, and Help. The toolbar contains icons for Run (a green play button) and Debug (a green bug icon). The Run icon is circled in blue, and the Debug icon is highlighted with a blue box. A text box with the instruction "Click on [Debug], then 'Run.'" points to these icons. The main workspace shows a workflow diagram with steps: Load Page, Click Item, Extract Name, Extract Description, Extract Overview, Extract Price, Extract Image, and Return Value. Below the workflow, the "Input/Output" tab is active, showing the "Output (Returned Values)" for the "HardyHardware.type". The output is a table with columns: #, Name, Description, Overview, Picture, and Price. The first row shows a "2-Light Brushed Nickel Flush Mount (2-Pack)" with a price of 22.97. Below the table, the details for the selected item are displayed, including the Name, Description, Overview (a paragraph of text), and a Picture of the flush mount. The "Summary" tab on the right shows various statistics: Returned Values: 1, Error Reports: 0, HTTP Requests: 34, Received (KB): 963, Sent (KB): 17, KCU-Point Usage: 20006, and Execution Time (s): 3.74. The "Stop When" section has checkboxes for "Values are Returned or Stored" (unchecked), "API Exceptions are Reported" (checked), and "Breakpoints are Reached" (checked). The "Steps to Skip" section has checkboxes for "Store in Database" (unchecked), "Delete from Database" (unchecked), "Execute SQL" (unchecked), "Execute Command Line" (unchecked), and "Send Email" (unchecked). The status bar at the bottom indicates "Execution completed successfully."

#	Name	Description	Overview	Picture	Price
1	2-Light Brushed Nickel Flush Mount (2-Pack)	Commercial Electric 2-Light Brushed Nickel Flush Mount (2-Pack)	Product Overview The Commercial Electric twin pack Flush Mount is a great look at a great value. From the professional to the do-it-yourselfer, the twin pack flush mount is an exceptional choice. Each Flush Mount is finished in Brushed Nickel with frosted white glass. Each Flush Mount uses two 60-watt medium-base light bulbs. From kitchen to bath or any room in between, the CE Brushed Nickel twin pack will not only enhance but will save cost to your home as well. * CE brushed nickel twin pack 2-light flush-mount * UL certified * Each flush mount uses two light bulbs (60 watt maximum each) (not included) * Assembled dimensions - 5.625 in. H x 13 in. W		22.97

Demonstration and Lab

Your first robot