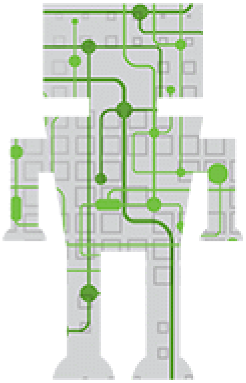
Kofax Kapow 10.3 Training and Certification

Module 11 – Working with Excel

Migrating Data from One Source to Another



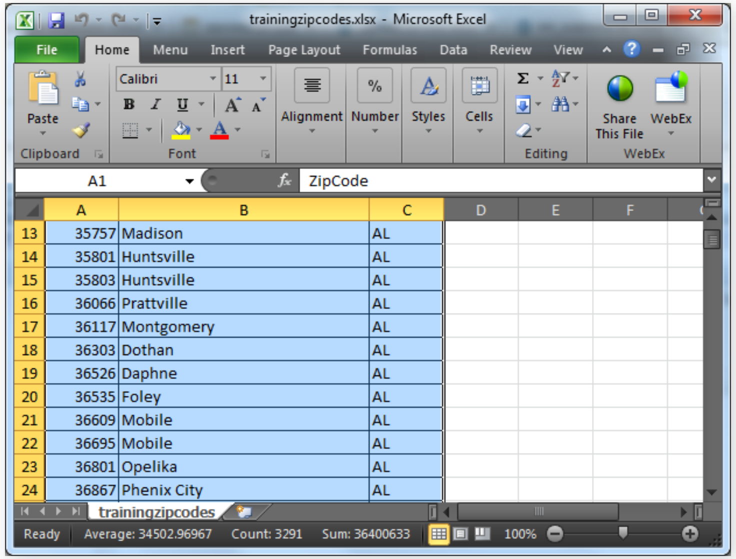
Training Module Overview

In this module, we'll create a Robot that takes data that exists in an Excel file, loops through it, and extracts the data which will then be stored in our Development Database for use in later training modules.

- Load Excel Spreadsheet
- View in Excel
- Loop in Excel
- Extract Data
- Store in Database

The Data We Need Already Exists in Excel

- HH store addresses already exist on our HardyHardware web site. *We will use that information in the next training module.*
- Correct city, state and zip code information already exists in an Excel spreadsheet.
- We want to store that correct information in our database for later use. *The purpose of this exercise is to learn how to move data from one source, like Excel to another source, like a database.*



trainingzipcodes.xlsx - Microsoft Excel

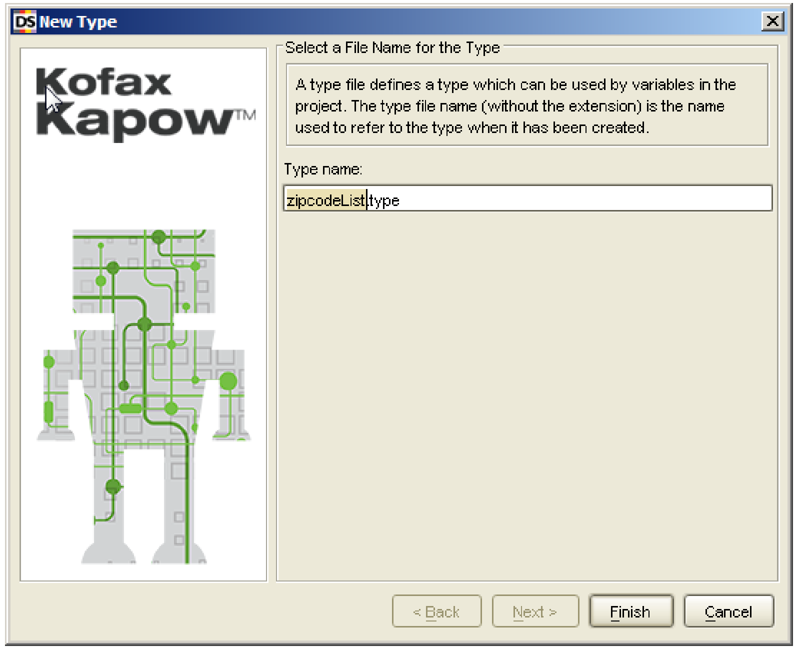
A1 ZipCode

	A	B	C
13	35757	Madison	AL
14	35801	Huntsville	AL
15	35803	Huntsville	AL
16	36066	Prattville	AL
17	36117	Montgomery	AL
18	36303	Dothan	AL
19	36526	Daphne	AL
20	36535	Foley	AL
21	36609	Mobile	AL
22	36695	Mobile	AL
23	36801	Opelika	AL
24	36867	Phenix City	AL

Ready Average: 34502.96967 Count: 3291 Sum: 36400633

Create Type

- The first thing we'll want to do is to create a new Type file that has three short text attributes as shown below:

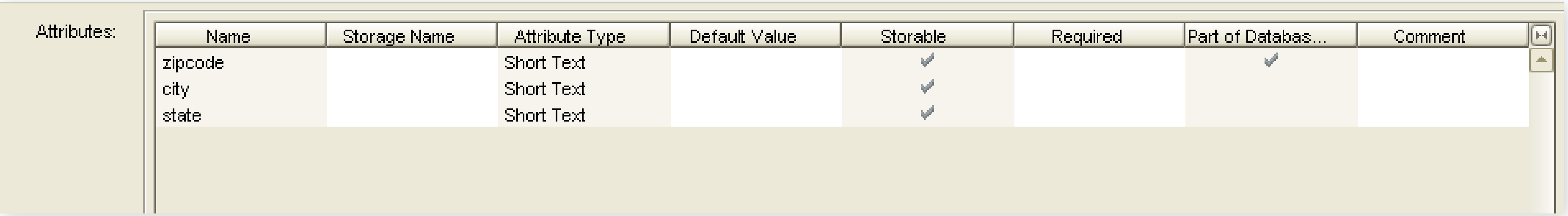


Select a File Name for the Type

A type file defines a type which can be used by variables in the project. The type file name (without the extension) is the name used to refer to the type when it has been created.

Type name:

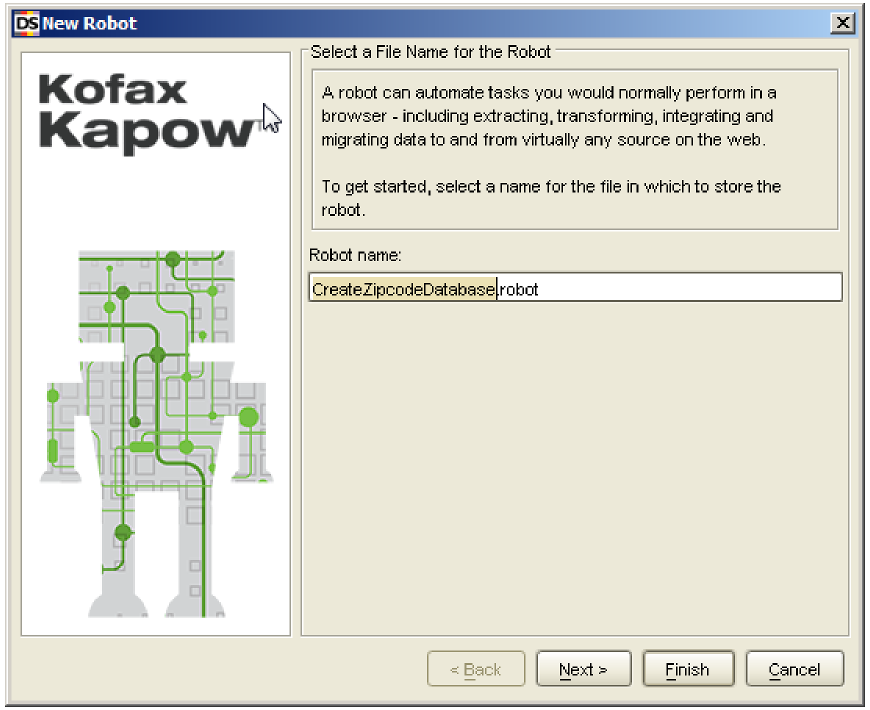
zipcodeList.type



Attributes:

Name	Storage Name	Attribute Type	Default Value	Storable	Required	Part of Databas...	Comment
zipcode		Short Text		✓		✓	
city		Short Text		✓			
state		Short Text		✓			

Then Create a New Robot and Load the Excel File



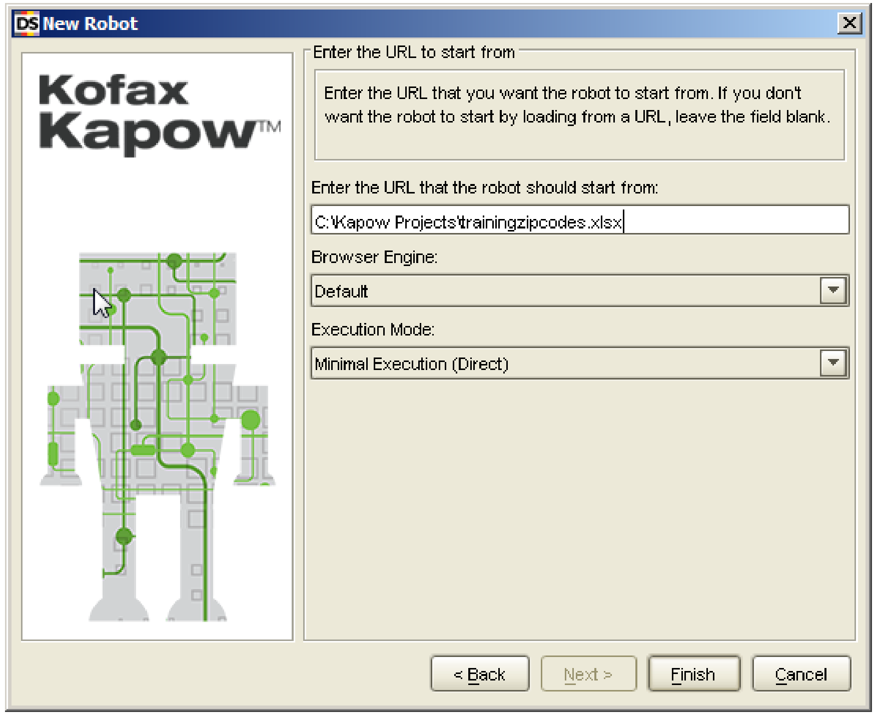
Select a File Name for the Robot

A robot can automate tasks you would normally perform in a browser - including extracting, transforming, integrating and migrating data to and from virtually any source on the web.

To get started, select a name for the file in which to store the robot.

Robot name: CreateZipcodeDatabase.robot

1. Create the Robot.



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: C:\Kapow Projects\trainingzipcodes.xlsx

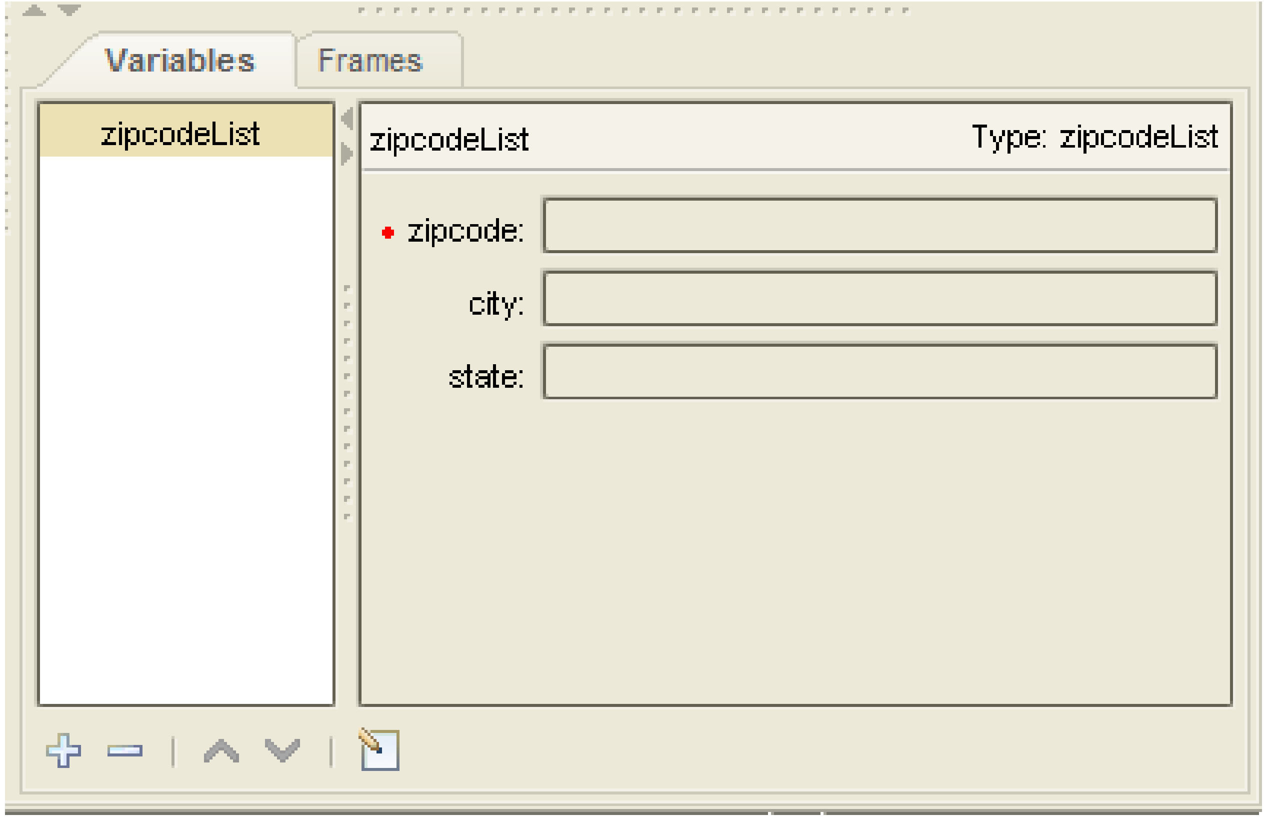
Browser Engine: Default

Execution Mode: Minimal Execution (Direct)

2. Enter the location of the file in the "Enter the URL…" box.

We'll Add a Variable to Our New Robot to Contain the Data

- The Variable will use the zipcodeList Type file we just created.



zipcodeList Type: zipcodeList

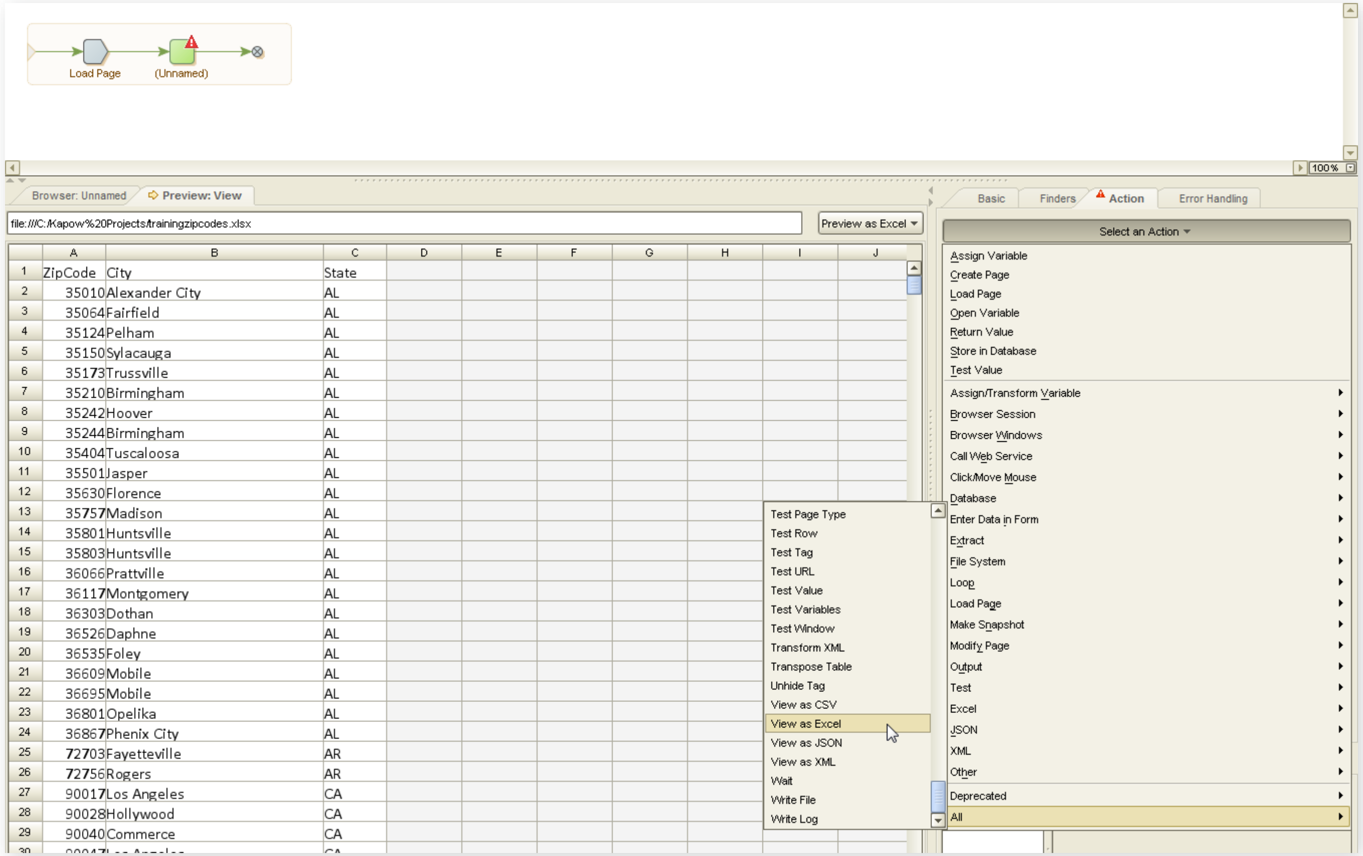
zipcode:

city:

state:

Then Add a View in Excel Step

- Next, we need to set up a "View in Excel" Action Step.
- This action opens downloaded Excel content in an Excel view. *The step action only works on downloaded Excel content.*

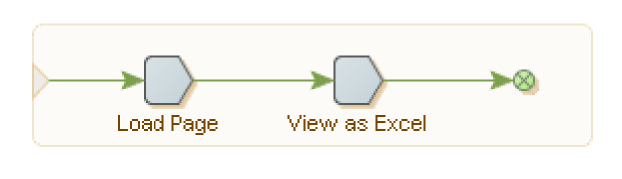


file:///C:/Kapow%20Projects/trainingzipcodes.xlsx

	A	B	C	D	E	F	G	H	I	J
1	ZipCode	City	State							
2	35010	Alexander City	AL							
3	35064	Fairfield	AL							
4	35124	Pelham	AL							
5	35150	Sylacauga	AL							
6	35173	Trussville	AL							
7	35210	Birmingham	AL							
8	35242	Hoover	AL							
9	35244	Birmingham	AL							
10	35404	Tuscaloosa	AL							
11	35501	Jasper	AL							
12	35630	Florence	AL							
13	35757	Madison	AL							
14	35801	Huntsville	AL							
15	35803	Huntsville	AL							
16	36066	Prattville	AL							
17	36117	Montgomery	AL							
18	36303	Dothan	AL							
19	36526	Daphne	AL							
20	36535	Foley	AL							
21	36609	Mobile	AL							
22	36695	Mobile	AL							
23	36801	Opelika	AL							
24	36867	Phenix City	AL							
25	72703	Fayetteville	AR							
26	72756	Rogers	AR							
27	90017	Los Angeles	CA							
28	90028	Hollywood	CA							
29	90040	Commerce	CA							
30	90047	Los Angeles	CA							

Set Up a Loop through Rows Step

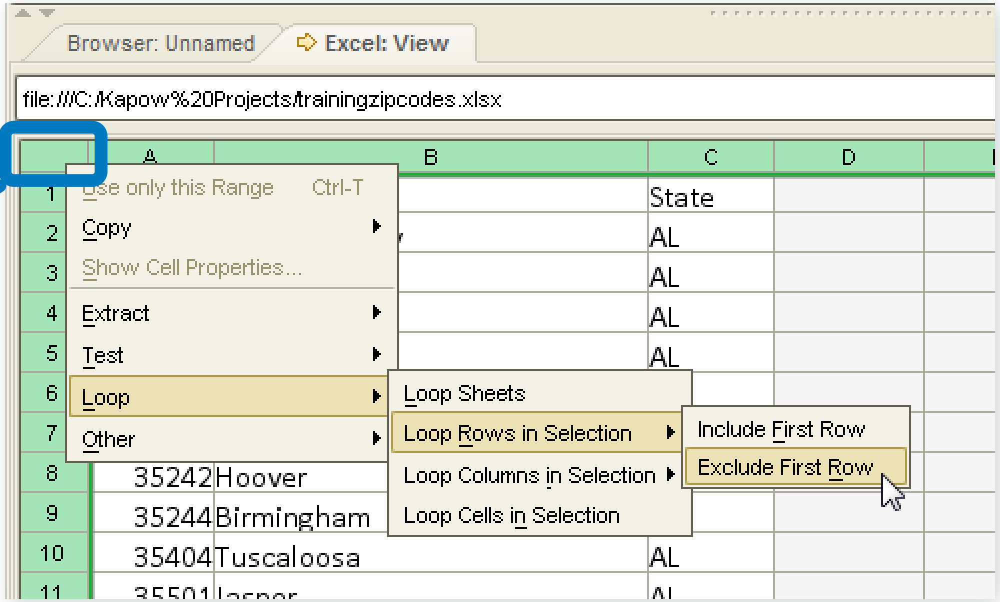
- We want to loop through ALL rows in the spreadsheet, so we need to select the entire contents. To do that, click on the cell at the top-most left corner of the sheet. As you can see, the entire sheet is selected as indicated by the green box.



Click here to select entire sheet

Then right mouse-click and select Loop | Loop Rows in Selection | Exclude First Row

(The first row contains the column headings)



file:///C:/Kapow%20Projects/trainingzipcodes.xlsx

	A	B	C	D
1			State	
2			AL	
3			AL	
4			AL	
5			AL	
6				
7				
8	35242	Hoover		
9	35244	Birmingham		
10	35404	Tuscaloosa	AL	
11	35501	Jasper	AL	

Loop in Excel

The Loop in Excel action can be configured using the following properties:

- *Loop Over:*This determines what kind of element the action will loop over. There are 4 possibilities for this:
  - *Sheets* -The action will loop over sheets in the spreadsheet document. No range finder is needed for this choice.
  - *Columns* -The action will loop over the columns in the range found by the range finder.
  - *Rows* -The action will loop over the rows in the range found by the range finder.
  - *Cells* -The action will loop over the cells in the range found by the range finder.
- *First Index:* The number of the first element to include in the loop. The number can be specified to count either forward from the first element, or backward from the last element. the new range (useful for stateful in-page looping)

Loop in Excel (cont.)

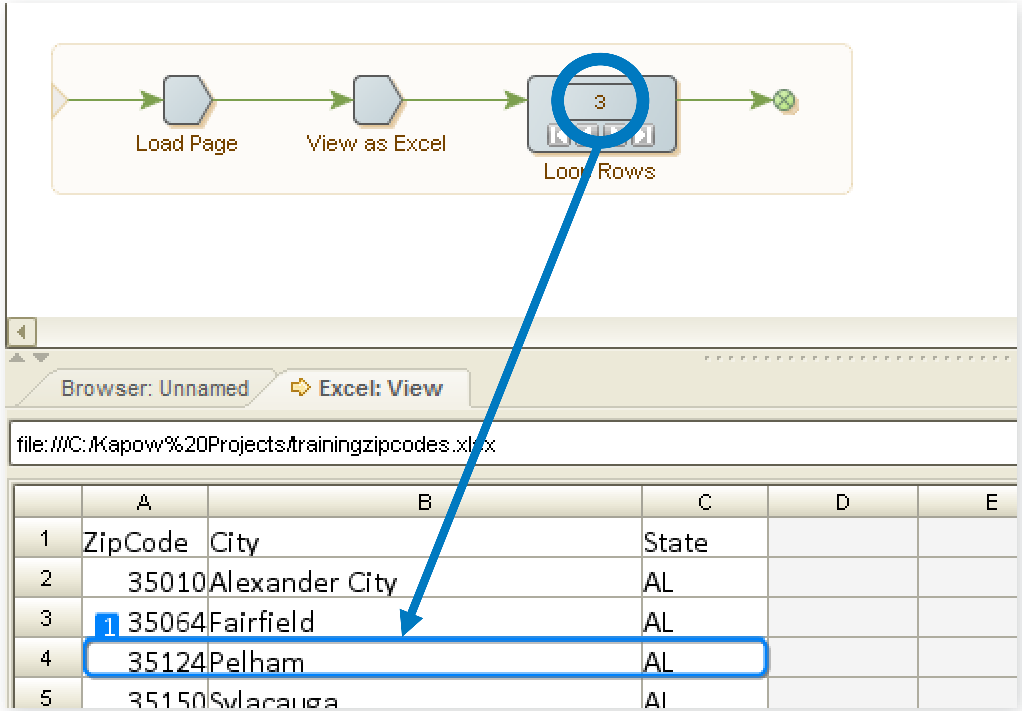
- *Last Index:* The number of the last element to include in the loop. The number can be specified to count either forward from the first element, or backward from the last element.
- *Increment:* Make the loop skip elements. For example, if an increment of 2 is specified, the loop will skip every second element.
- *Loop Backwards:* Select that the loop should loop through the matching elements in reverse order. Please note that the loop will go through exactly the same elements as if it were looping forward just in reversed order. This means that the *First Index* is referring to first element in the selection of elements to loop over and not the first element visited when looping (actually it will be the last when looping backwards).
- *Range Name:* Has two options, **Auto** or **Named**.

Auto gives the range a name which is number. The first Auto-numbered range will have number 1, the next number 2 etc. Note that the number may change if additional Auto-numbered ranges are inserted before this step (on the same page).

Named gives the range a fixed and explicitly stated name.

Test your Loop

- Remember, you can test a loop by going the step just past the loop step (in this case, the end step) and use the forward arrow to see the loop advance.

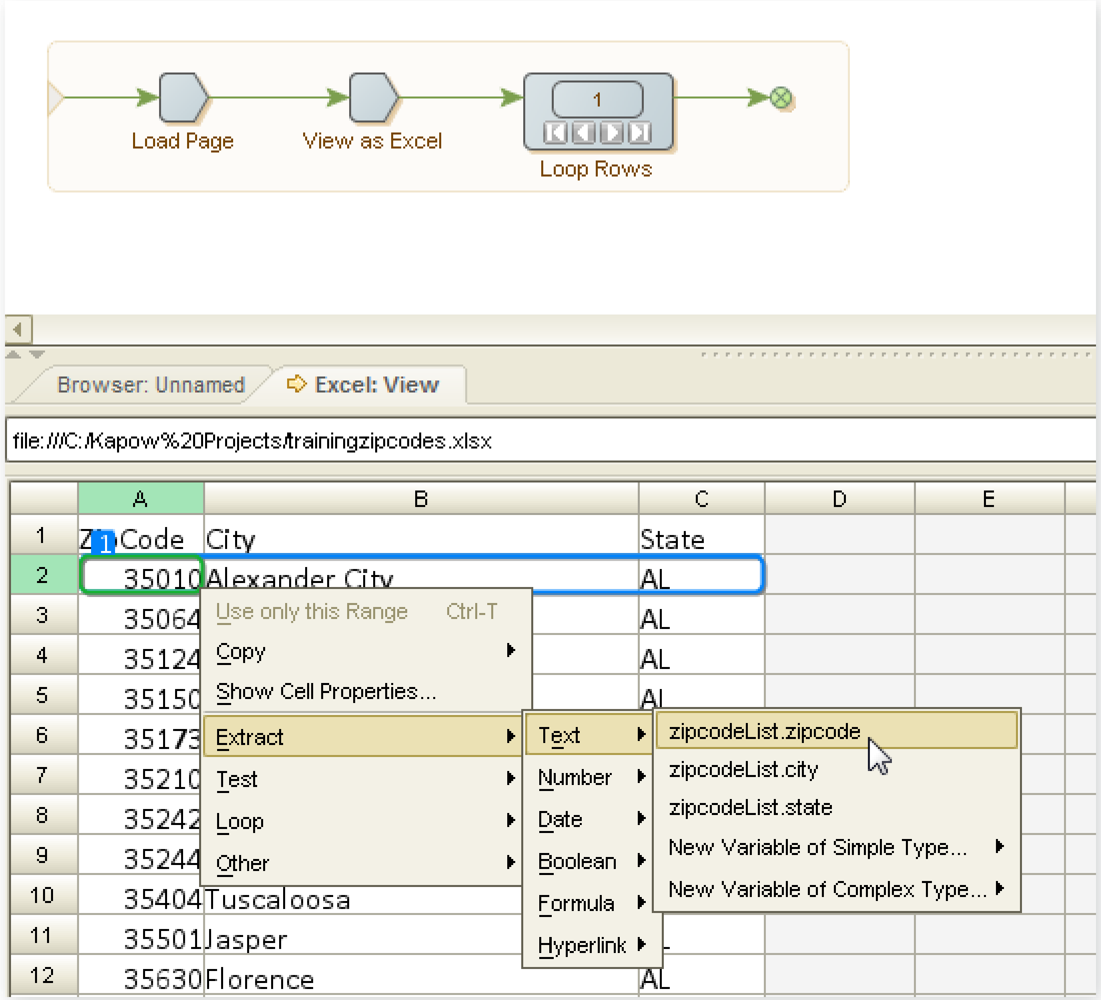


file:///C:/Kapow%20Projects/trainingzipcodes.xlsx

	A	B	C	D	E
1	ZipCode	City	State		
2	35010	Alexander City	AL		
3	35064	Fairfield	AL		
4	35124	Pelham	AL		
5	35150	Sylacauga	AL		

Then Extract Zipcode

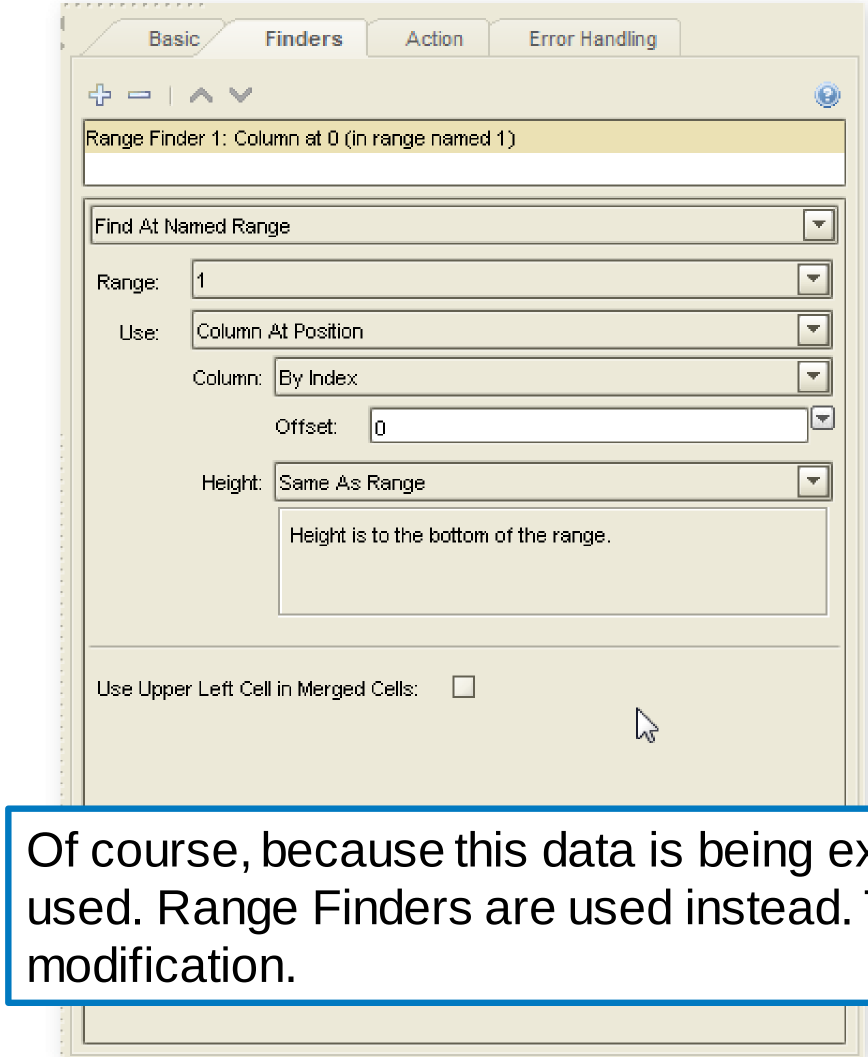
- Data is extracted in a similar fashion to how we extracted it on a web page: Click on the data to select it and then choose Extract | Text | zipcodeList.zipcode from the context menus.



file:///C:/Kapow%20Projects/trainingzipcodes.xlsx

	A	B	C	D	E
1	ZipCode	City	State		
2	35010	Alexander City	AL		
3	35064		AL		
4	35124		AL		
5	35150		AL		
6	35173				
7	35210				
8	35242				
9	35244				
10	35404	Tuscaloosa			
11	35501	Jasper			
12	35630	Florence	AL		

Properties are Set for the Extract Step



Range Finder 1: Column at 0 (in range named 1)

Find At Named Range

Range: 1

Use: Column At Position

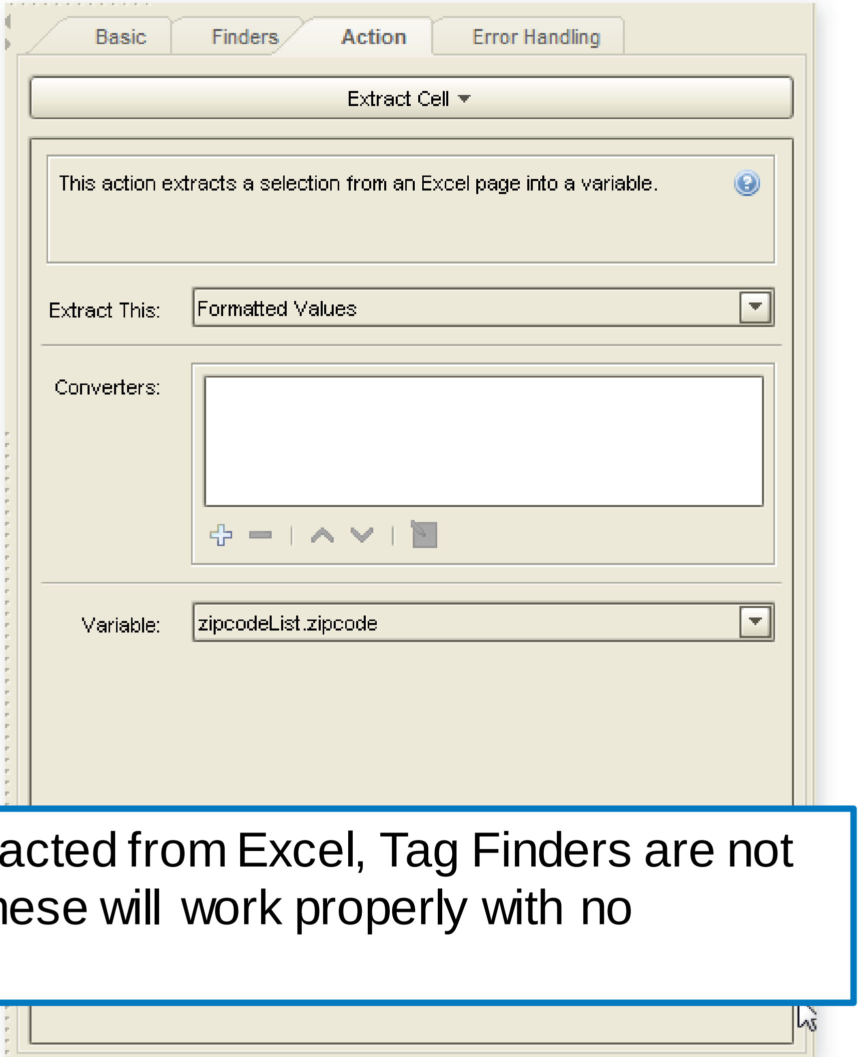
Column: By Index

Offset: 0

Height: Same As Range

Height is to the bottom of the range.

Use Upper Left Cell in Merged Cells: □



This action extracts a selection from an Excel page into a variable.

Extract This: Formatted Values

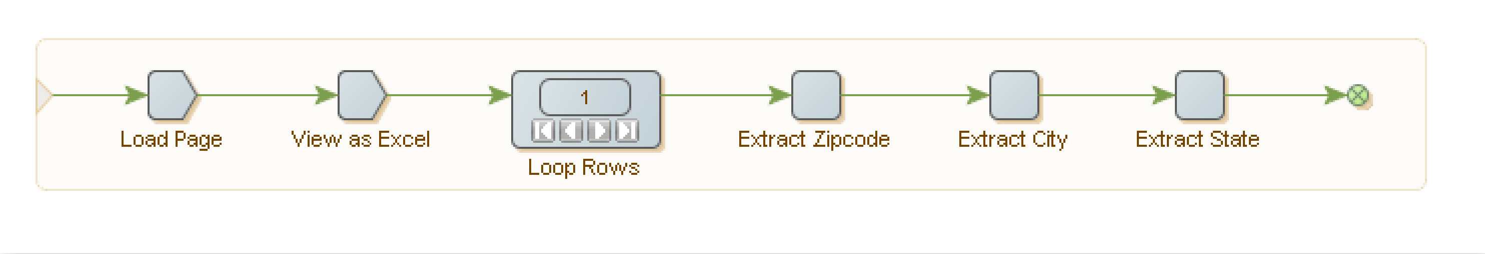
Converters:

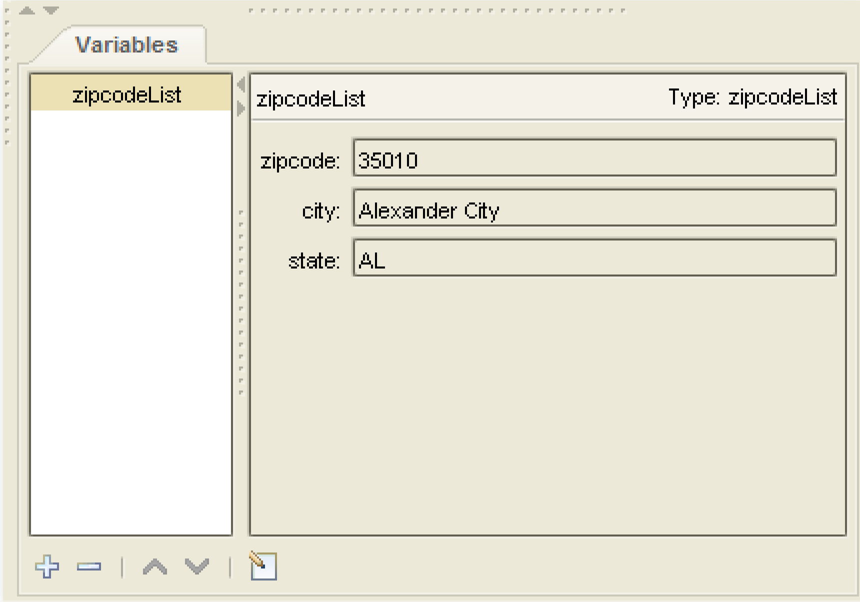
Variable: zipcodeList.zipcode

Of course, because this data is being extracted from Excel, Tag Finders are not used. Range Finders are used instead. These will work properly with no modification.

And Extract City and State

- Extract City and State the same way as you extracted the Zip Code.





zipcodeList Type: zipcodeList

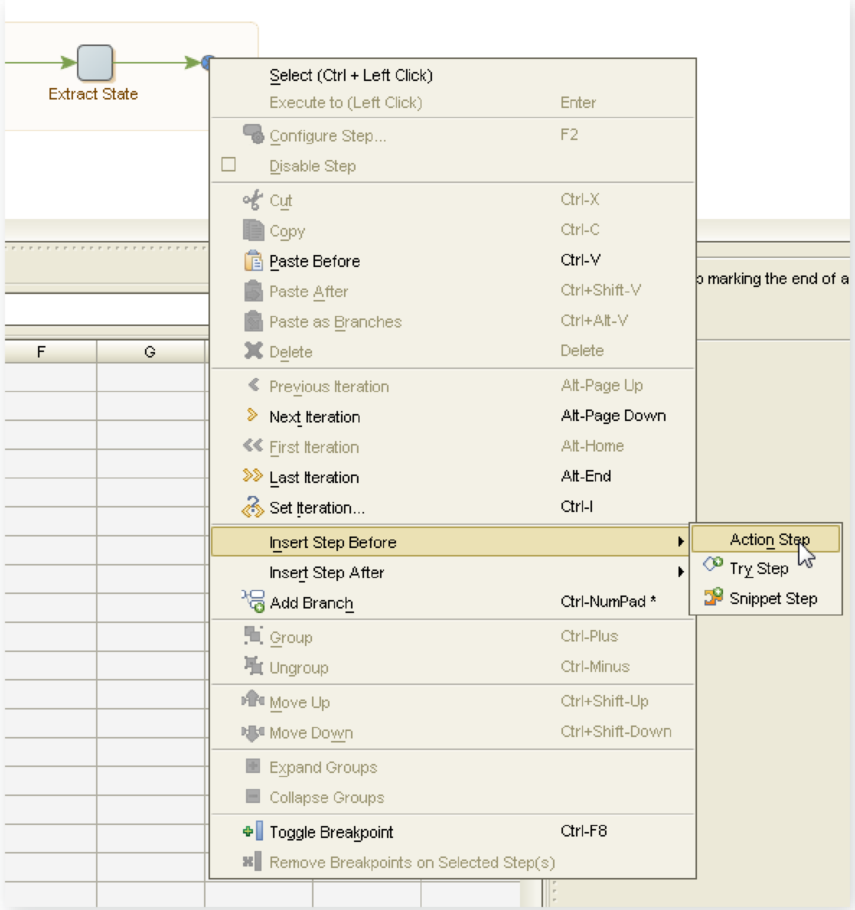
zipcode: 35010

city: Alexander City

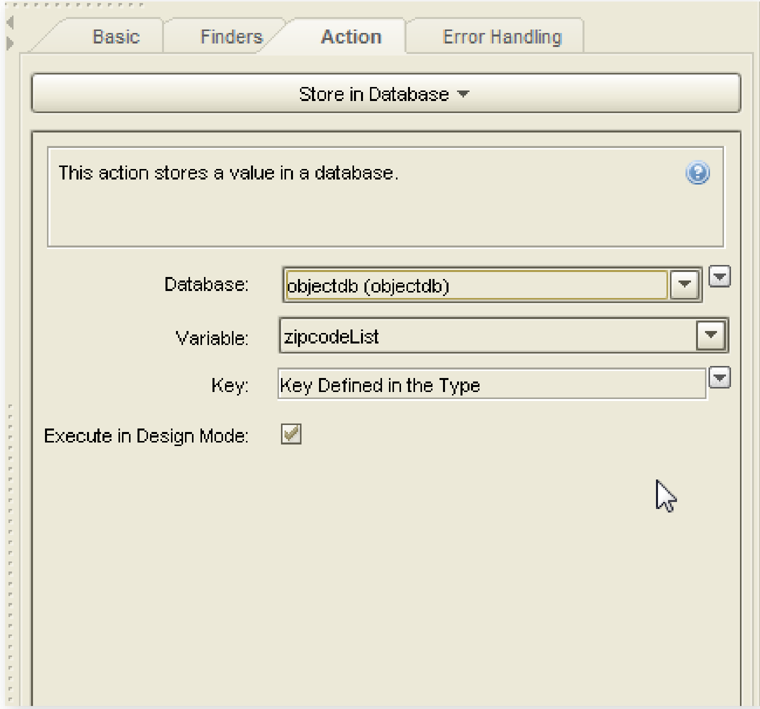
state: AL

Finally, Add a "Store in Database" Step

Add a new action step before the end step.



Select "Store in Database" as the action and select the database from the dropdown list.



This action stores a value in a database.

Database: objectdb (objectdb)

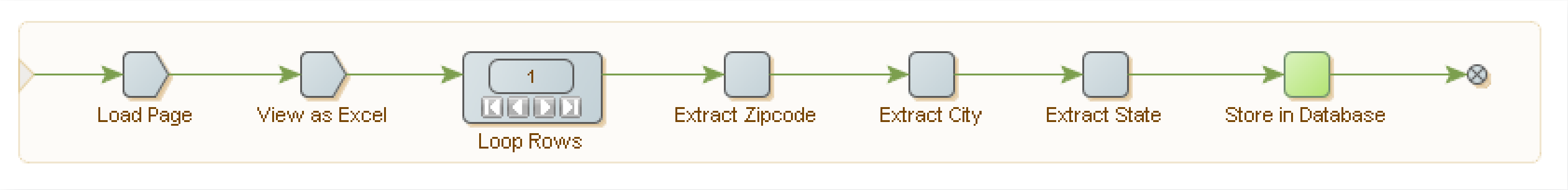
Variable: zipcodeList

Key: Key Defined in the Type

Execute in Design Mode: ☑

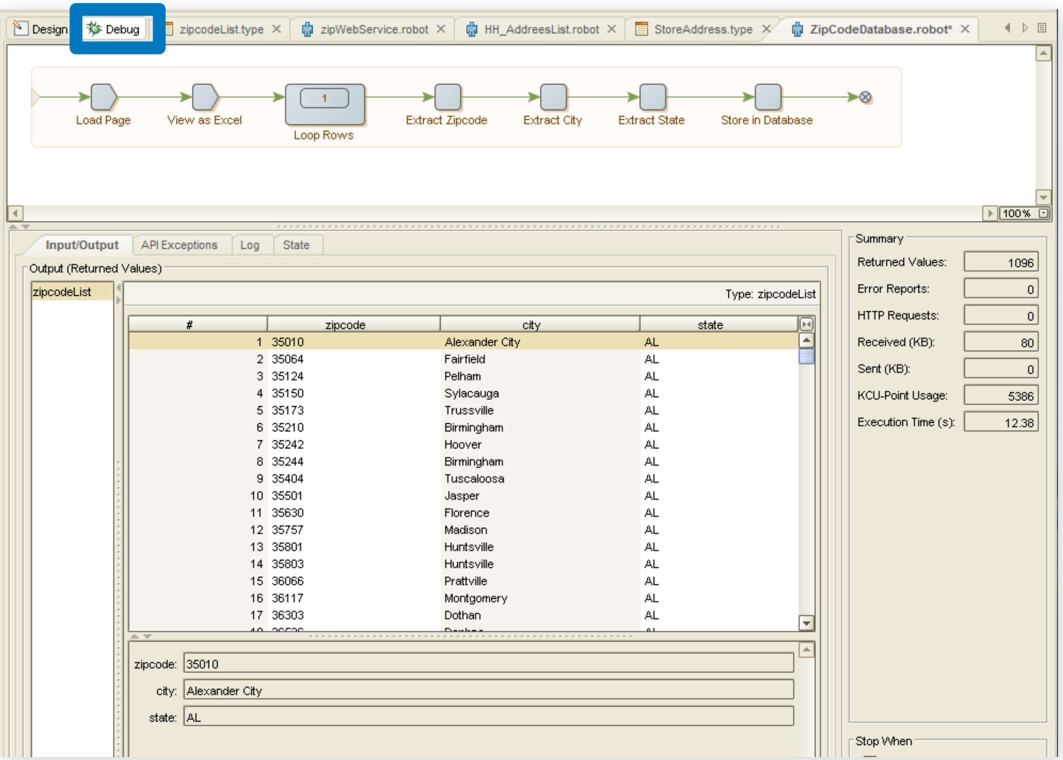
Your Completed Robot Looks Like This

- It's simple, but it works!



Running in Debug Mode

- IMPORTANT: Realize that running your Robot in Debug mode will output data to your database. If you wanted to test your Robot first, you could create a "Return Value" step and temporarily disable the "Store in Database" step.



#	zipcode	city	state
1	35010	Alexander City	AL
2	35064	Fairfield	AL
3	35124	Pelham	AL
4	35150	Sylacauga	AL
5	35173	Trussville	AL
6	35210	Birmingham	AL
7	35242	Hoover	AL
8	35244	Birmingham	AL
9	35404	Tuscaloosa	AL
10	35501	Jasper	AL
11	35630	Florence	AL
12	35757	Madison	AL
13	35801	Huntsville	AL
14	35803	Huntsville	AL
15	36066	Prattville	AL
16	36117	Montgomery	AL
17	36303	Dothan	AL

zipcode: 35010

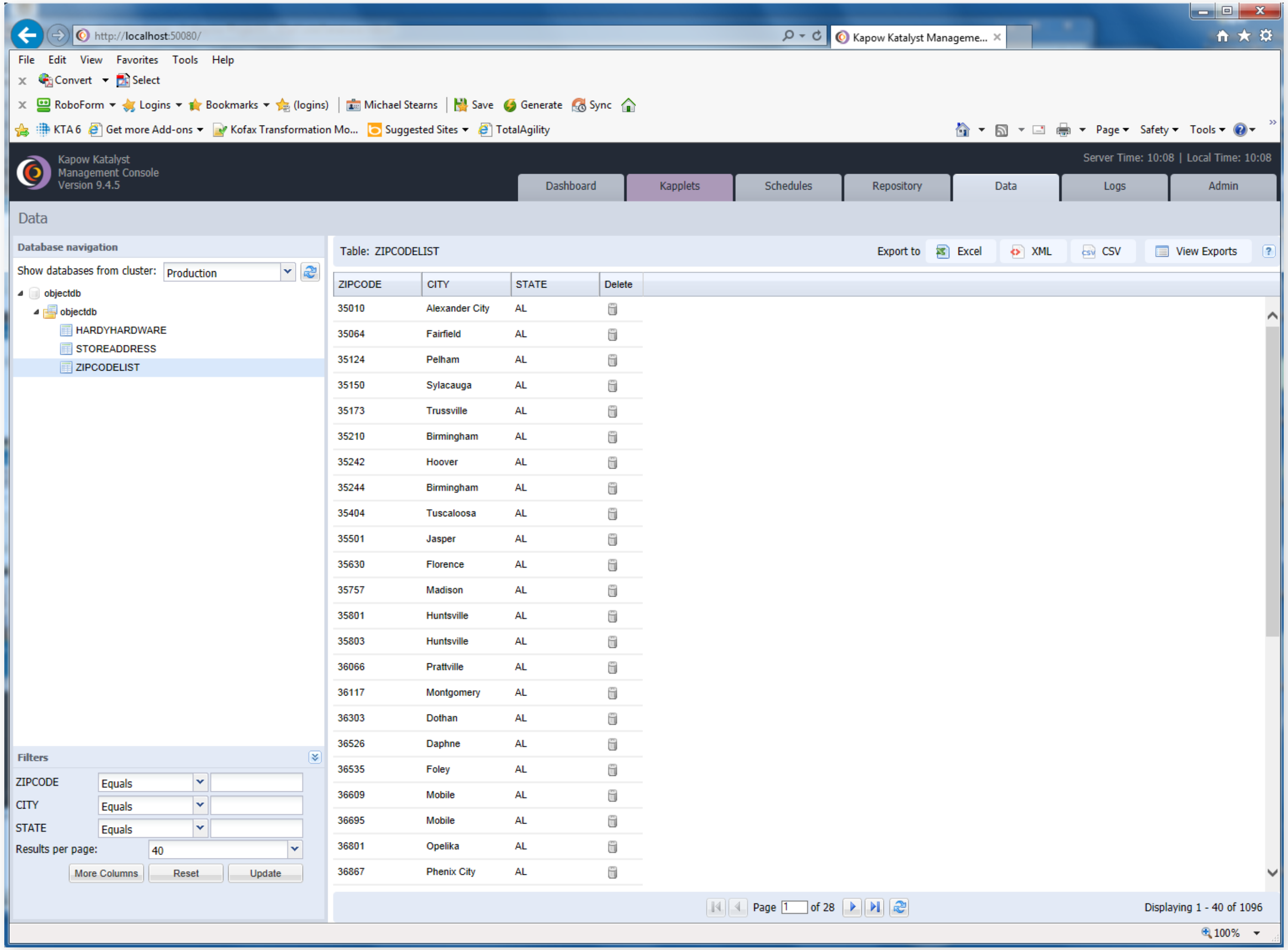
city: Alexander City

state: AL

Returned Values:	1096
Error Reports:	0
HTTP Requests:	0
Received (KB):	80
Sent (KB):	0
KCU-Point Usage:	5386
Execution Time (s):	12.38

Viewing Data in the Management Console

- By opening the Management Console, we can see that data has been successfully output and stored.


http://localhost:50080/
Kapow Katalyst Manageme...
File Edit View Favorites Tools Help
Convert Select
RoboForm Logins Bookmarks (logins) Michael Stearns Save Generate Sync
KTA 6 Get more Add-ons Kofax Transformation Mo... Suggested Sites TotalAgility
Page Safety Tools
Kapow Katalyst
Management Console
Version 9.4.5
Server Time: 10:08 | Local Time: 10:08
Dashboard
Kapplets
Schedules
Repository
Data
Logs
Admin
Export to Excel XML CSV View Exports
More Columns
Reset
Update
100%

Data

Database navigation

Show databases from cluster: Production

- objectdb
  - objectdb
    - HARDYHARDWARE
    - STOREADDRESS
    - ZIPCODELIST

Filters

ZIPCODE	Equals	
CITY	Equals	
STATE	Equals	

Results per page: 40

Table: ZIPCODELIST

ZIPCODE	CITY	STATE	Delete
35010	Alexander City	AL	🗑
35064	Fairfield	AL	🗑
35124	Pelham	AL	🗑
35150	Sylacauga	AL	🗑
35173	Trussville	AL	🗑
35210	Birmingham	AL	🗑
35242	Hoover	AL	🗑
35244	Birmingham	AL	🗑
35404	Tuscaloosa	AL	🗑
35501	Jasper	AL	🗑
35630	Florence	AL	🗑
35757	Madison	AL	🗑
35801	Huntsville	AL	🗑
35803	Huntsville	AL	🗑
36066	Prattville	AL	🗑
36117	Montgomery	AL	🗑
36303	Dothan	AL	🗑
36526	Daphne	AL	🗑
36535	Foley	AL	🗑
36609	Mobile	AL	🗑
36695	Mobile	AL	🗑
36801	Opelika	AL	🗑
36867	Phenix City	AL	🗑

Page 1 of 28

Displaying 1 - 40 of 1096

Demo & Lab

Extracting Data from Excel
Storing Data in a Database