Convert HTML to a ResultSet Table

Derek Pascarella
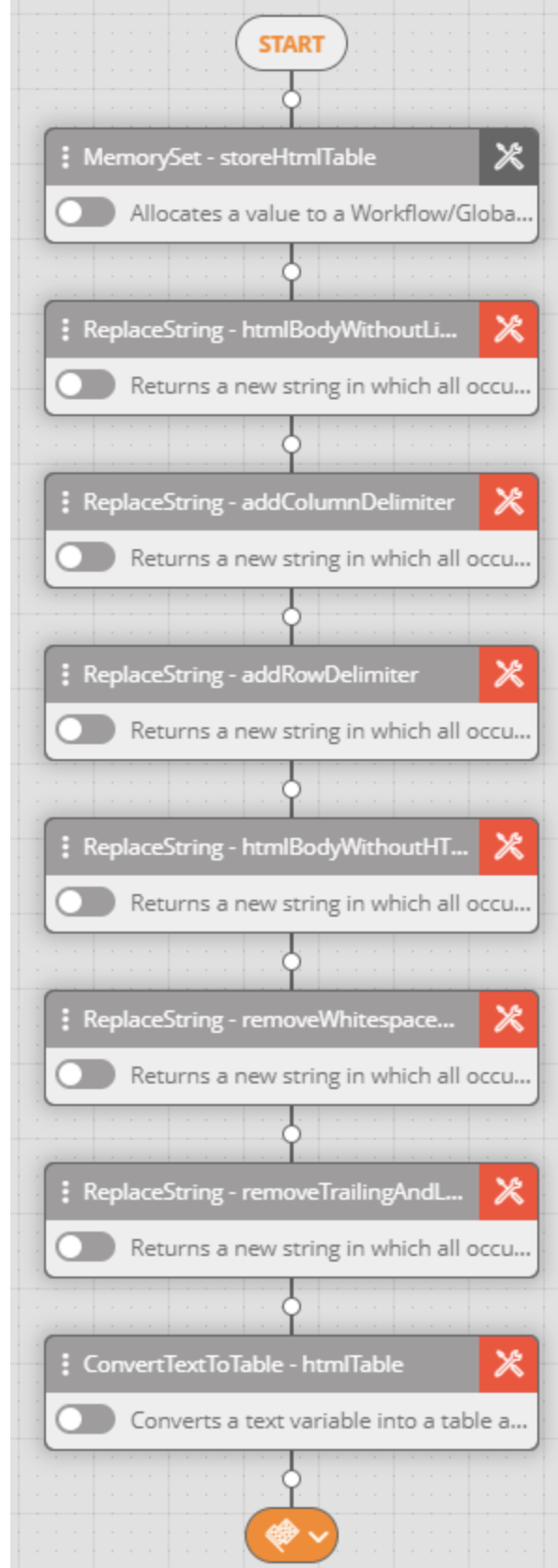Updated a few seconds ago

Applies To: Ayehu NG

Description

Ayehu NG utilizes tables as a useful and easy-to-read/manipulate format for both storing and retrieving data. Other formats are also available on the platform, like JSON and XML. Although there exists an easy and convenient **Table to HTML** activity for converting **Ayehu NG**'s native ResultSet table format to HTML, no such equivalent is available for converting HTML to a table.

The following tutorial outlines the simple steps needed to convert HTML to a table in an **Ayehu NG** workflow. Please note that you can also use this workflow template as a child workflow, which can be called from a parent workflow, so that it's reusable all throughout any of your workflows. For more information on parent and child workflows, see the following article: https://support.ayehu.com/hc/en-us/articles/360008089093

Workflow Overview

Below is a screenshot of an **Ayehu NG** workflow that contains a sample HTML table and then converts it to a ResultSet table. You can also download an export of this workflow attached to this article or on the Ayehu GitHub at https://raw.githubusercontent.com/Ayehu/custom-workflows/master/HTML%20Parsing%20Examples/Convert%20HTML%20Table%20To%20ResultSet%20Table.xml.

START

MemorySet - storeHtmlTable
Allocates a value to a Workflow/Globa...

ReplaceString - htmlBodyWithoutLi...
Returns a new string in which all occu...

ReplaceString - addColumnDelimiter
Returns a new string in which all occu...

ReplaceString - addRowDelimiter
Returns a new string in which all occu...

ReplaceString - htmlBodyWithoutHT...
Returns a new string in which all occu...

ReplaceString - removeWhitespace...
Returns a new string in which all occu...

ReplaceString - removeTrailingAndL...
Returns a new string in which all occu...

ConvertTextToTable - htmlTable
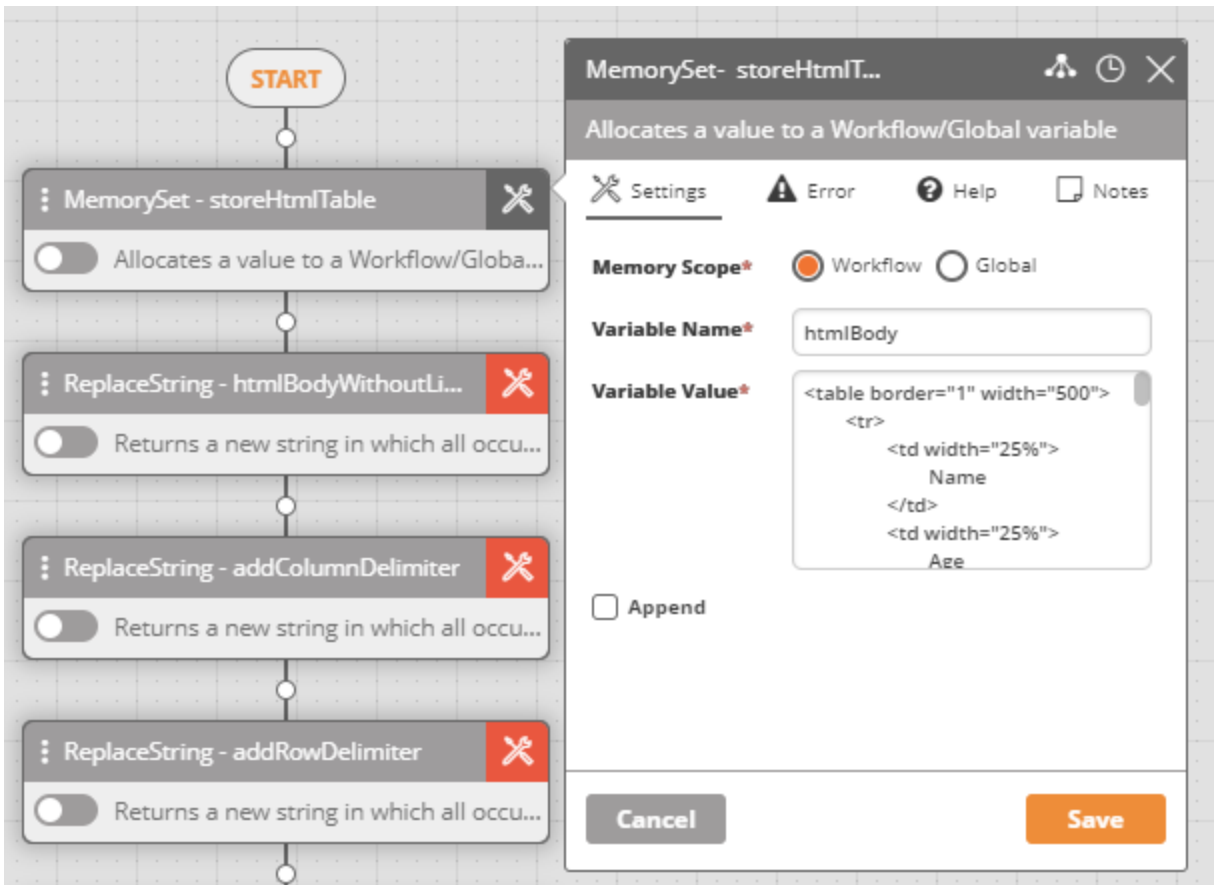Converts a text variable into a table a...

The workflow achieves this by taking the following steps:

1. Strips links from the HTML table text.
2. Add column and row delimiters to the HTML table text.
3. Strips any HTML tags from the text.
4. Removes all forms of extraneous whitespace from the text.

5. Converts the text to a table.

Activity Configuration

Now, let's look at an example of this workflow in action. The very first activity stores an HTML Table in the **%htmlBody%** variable.

The HTML Table from this activity is as follows:

```
<table border="1" width="500">
  <tr>
    <td width="25%">
      Name
    </td>
    <td width="25%">
      Age
    </td>
    <td width="25%">
      Location
    </td>
```

```
      <td width="25%">
        Occupation
      </td>
    </tr>
    <tr>
      <td width="25%">
        <a href="test.com">John Smith</a>
      </td>
      <td width="25%">
        32
      </td>
      <td width="25%">
        NYC
      </td>
      <td width="25%">
        Accountant
      </td>
    </tr>
    <tr>
      <td width="25%">
        Mary Johnson
      </td>
      <td width="25%">
        29
      </td>
      <td width="25%">
        CA
      </td>
      <td width="25%">
        Receptionist
      </td>
    </tr>
    <tr>
      <td width="25%">
        Dave Jones
      </td>
      <td width="25%">
        40
      </td>
      <td width="25%">
        FL
      </td>
      <td width="25%">
        Engineer
      </td>
    </tr>
</table>
```

After a successful execution of this workflow, the following native ResultSet table is returned.

Name	Age	Location	Occupation
John Smith	32	NYC	Accountant
Mary Johnson	29	CA	Receptionist
Dave Jones	40	FL	Engineer

Name	Age	Location	Occupation
John Smith	32	NYC	Accountant
Mary Johnson	29	CA	Receptionist
Dave Jones	40	FL	Engineer