BPA
Platform

Technical Overview

Format as HTML Pro Tool

codeless
PLATFORMS

Copyright

Head Office:

Codeless Platforms

Suite 1 & 2 Bourne Gate

25 Bourne Valley Road

Poole

BH12 1DY

United Kingdom

Tel: +44 (0) 330 99 88 700

Email: enquiries@codelessplatforms.com

Trademarks

The information provided in this publication may contain errors, omissions, or typographical errors or may be out of date. Orbis Software Ltd T/A Codeless Platforms may change, delete, or update any published information at any time and without prior notice. The information published in this document is provided for informational purposes only and is not binding on Orbis Software Ltd T/A Codeless Platforms in any way except to the extent that it is specifically indicated to be so.

Contents

Introduction 1

Features 1

Technical Summary 2

Working with other Tools 2

Consuming From Other Tools 2

Objects Consumed 2

Exposing to Other Tools 3

Objects Exposed 3

Global Configuration 3

Step Configuration 4

About the General Tab 4

About the Data Tab 5

Document Generation Options 6

Grouping Your Data 6

Creating Sub Totals 8

Creating Grand Totals 8

Automatically Creating Groups and Sub and Grand Totals 8

Adjusting the Recordset Structure 8

Specifying the Recordset Relationships 9

About the Design Tab 10

Styling the HTML Template 10

Creating a Basic HTML Template 11

Checking the HTML Design 12

A Note About the BPA Platform-Specific HTML Classes 12

Using Format as HTML Pro's Advanced Features 13

About the Options Tab 15

Introduction

The **Format as HTML Pro** tool is used to create a task step that produces single or multiple HTML documents. You can choose to create the HTML template from scratch or import a pre-existing template. Recordsets and other task step properties can also be used to populate the HTML template to produce dynamic content documents. These documents can then be consumed and then delivered by Output or Execute task steps.

Features

- HTML 4.0 onwards supported
- Create or use existing HTML templates
  - Support for use of complex templates
- Create single or multiple documents as required
- Merge data from other task steps
- Support for multiple recordsets
- Support for input recordset customisation in the tool rather than in the consumed step
- Support for multiple tables, including having tables embedded in existing tables
- Support for image resource handling relative to the task step
- Freedom to display in other methods other than tables, for example, lists, free-form, and so on

Technical Summary

Working with other Tools

The **Format as HTML Pro** tool can directly interact with the following tools:

Consuming From Other Tools

The **Format as HTML Pro** tool can consume objects outputted by the following tools:

Icon	Tool Name	Tool Category
	Call Stored Procedure (OLEDB)	Input, Data Connectors, Output, and Execute
	Database Query (ODBC)	Input and Data Connectors
	Database Query (OLEDB)	Input and Data Connectors
	Import Flat File	Input
</>	**Convert XML to Recordset**	Format
	Filter Data	General

NOTE: The **Format as HTML Pro** can only consume from the **Call Stored Procedure (OLEDB)** tool when it is outputting recordsets.

Objects Consumed

The following objects, outputted by the above tools, can be directly consumed by the **Format as HTML Pro** tool:

- **Recordset** — Tabular data from any BPA Platform tool capable of exposing such data (see above)

Exposing to Other Tools

Objects exposed by the **Format as HTML** tool can be directly consumed by the following tools:

Icon	Tool Name	Tool Category
	Save File	Output
	Send Email (SMTP)	Output
	Transfer File (FTP)	Output
	Call COM Object	Execute
	Run External Program	Execute
	Run VBScript	Execute

Objects Exposed

The following objects, exposed by the **Format as HTML Pro** tool, can be directly consumed by the above tools:

- **Documents (HTML)** — Standalone HTML pages
- **RecordSource** — If an **Input Recordset** has been selected (see General tab), this contains the columns included in the recordset
- **Step Properties** — Standard step properties are available allowing you to use statistical data of the tool

Global Configuration

The **Format as HTML Pro** tool does not require any global configuration before being used in a task.

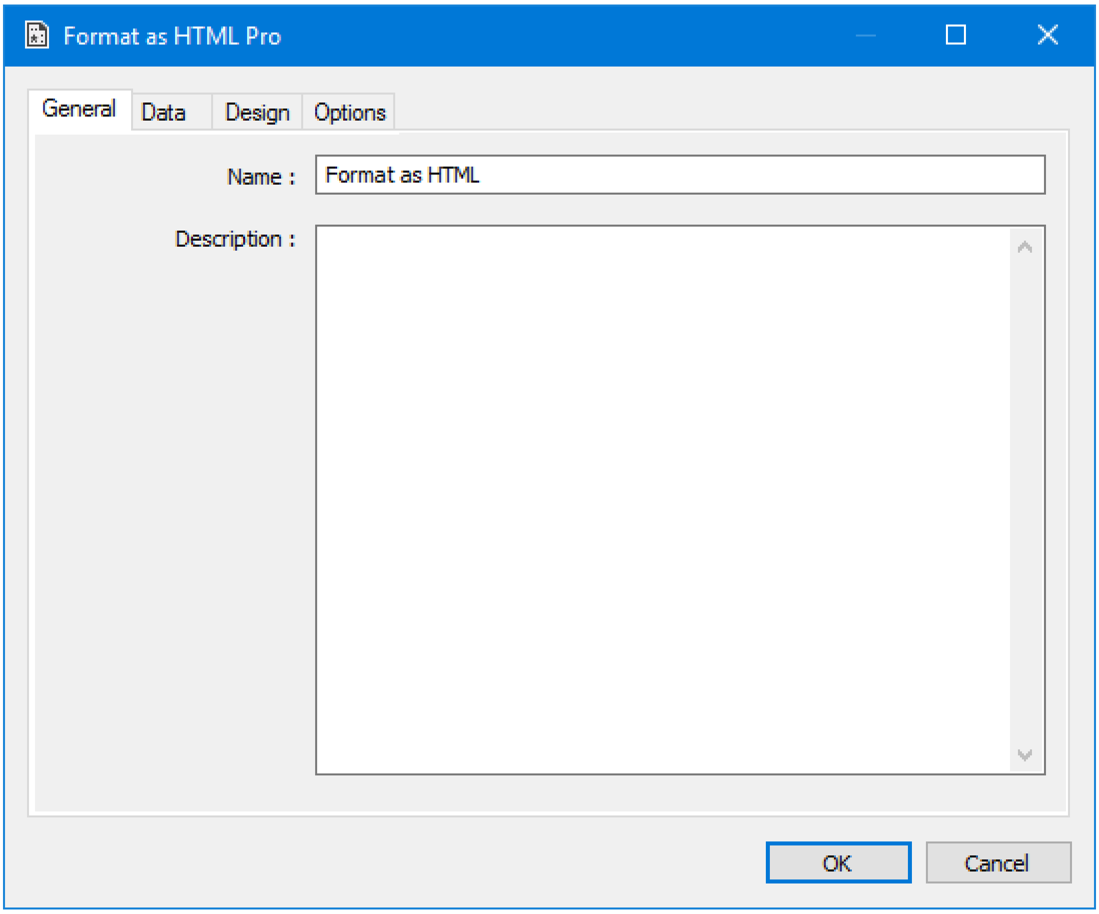
Step Configuration

To add a new **Format as HTML Pro** step to an existing task, you either:

- Click and drag the **Format as HTML Pro** icon from the **Task Browser** to the task **Design** area.
- From the task's **Design** tab, right-click on empty space and select ***New* > *Format* > *Format as HTML Pro.***

For a detailed description of how to create new tasks, refer to the product help.

About the General Tab



Name : Format as HTML

Description :

The **General** tab is used to enter the following details for the step:

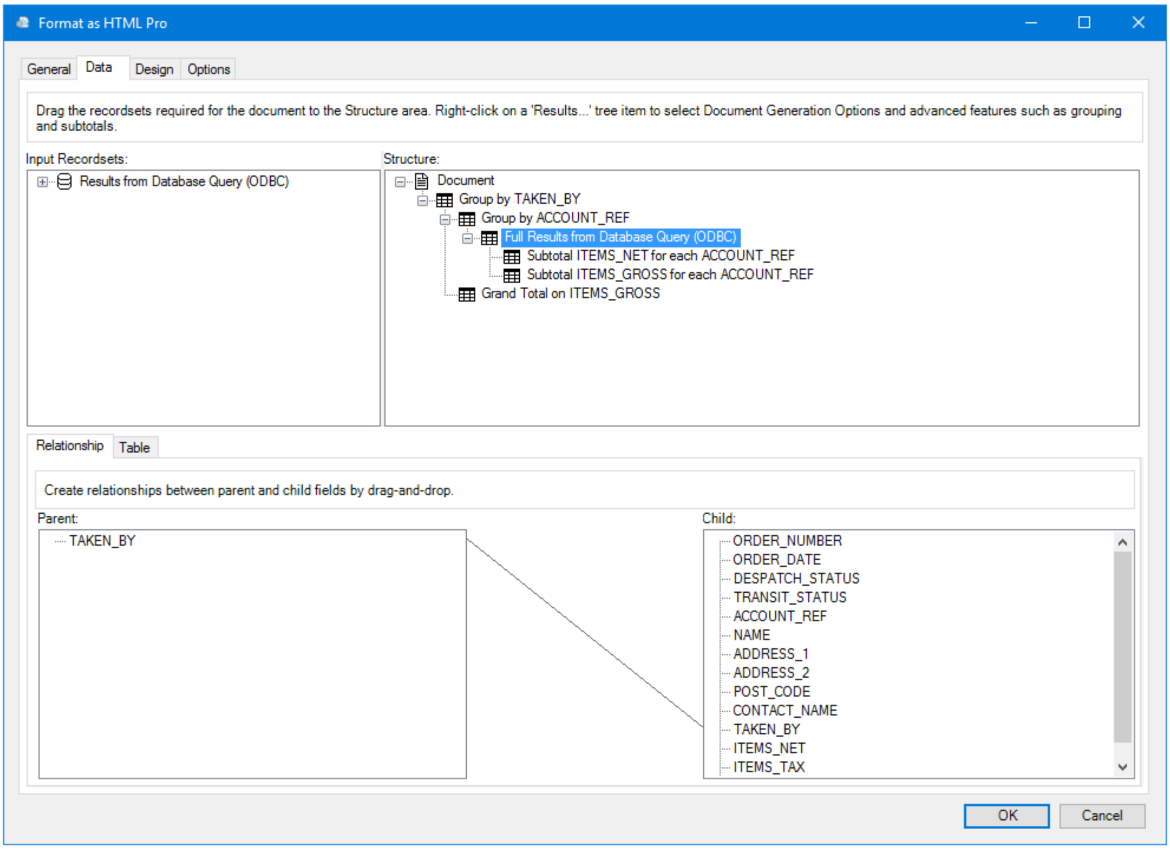
- **Name** — Enter a meaningful name for the step

TIP: If this task instance makes use of two or more **Format as HTML Pro** steps, ensure the **Name** used is unique for each individual step.

- **Description** — If required, enter a description of this step

About the Data Tab

You use the **Data** tab to specify and manipulate the data sources included in your HTML document.



Drag the recordsets required for the document to the Structure area. Right-click on a 'Results...' tree item to select Document Generation Options and advanced features such as grouping and subtotals.

Input Recordsets:

- Results from Database Query (ODBC)

Structure:

```
Document
  Group by TAKEN_BY
    Group by ACCOUNT_REF
      Full Results from Database Query (ODBC)
        Subtotal ITEMS_NET for each ACCOUNT_REF
        Subtotal ITEMS_GROSS for each ACCOUNT_REF
    Grand Total on ITEMS_GROSS
```

Create relationships between parent and child fields by drag-and-drop.

Parent:

- TAKEN_BY

Child:

- ORDER_NUMBER
- ORDER_DATE
- DESPATCH_STATUS
- TRANSIT_STATUS
- ACCOUNT_REF
- NAME
- ADDRESS_1
- ADDRESS_2
- POST_CODE
- CONTACT_NAME
- TAKEN_BY
- ITEMS_NET
- ITEMS_TAX

NOTE: This tab is not available if you do not have any steps capable of exposing a recordset in your task. In this scenario, the **Format as HTML Pro** step can only generate a single document — see If Not Consuming a Recordset.

Available recordsets from previous task steps are shown in the **Input Recordsets** pane. Drag the relevant recordsets whose data you want in the HTML document, to the **Structure** pane in the required data structure order.

TIP: If all records from a single input recordset are required without any customisation, drag the relevant **Results from ...** to the **Data Sources** pane and select the document generation options. If data customisation is required, carry out the following sequence before creating or importing the HTML template — note that you must have the required data structure for your HTML template to hand first.

TIP: When creating your data structure, the first nodes should be what you want to do to the recordset before specifying which recordset is involved.

Document Generation Options

Note that the **Document Generation Options** pane is only available for the parent-level results set under **Document**. Use this pane to determine whether a single document is created at task run-time or multiple documents, based on the following:

NOTE: The **Data** tab is not available if you do not have any steps capable of exposing a recordset in your task therefore you cannot choose the type of output. By default, only a single HTML document is generated at task run-time.

From the **Document Generation Options** pane, you specify when a new document is generated:

- **Single Document** — Generate a new document each time the **Format as HTML Pro** task step is run.
- **New Document when Column changes** — Generate a new document each time the value in a specified column changes. Select the relevant column.

TIP: You can nominate multiple columns to generate new documents from. If so, the nominated columns are read as an *OR* statement rather than an *AND*. You do not need to restructure your recordset data to achieve this; each record is read and processed in turn.

- **One Document for each row** — Generate a new document for every record in the input recordset.

TIP: If required, you do not have to include nominated columns in your outputted HTML document.

Grouping Your Data

This feature acts in the same manner as the `GROUP BY` condition in SQL. If your input recordset data is not grouped, you are getting generating a **New Document when Columns change**, or wanting sub-totals generated per group, you should create a group node. Group nodes control which records are returned; rather than saying "here is my data, group it" (recordset node as the parent to the group node), group nodes work in the opposite manner by saying "group the following data by these columns" (group node as the parent to the recordset).

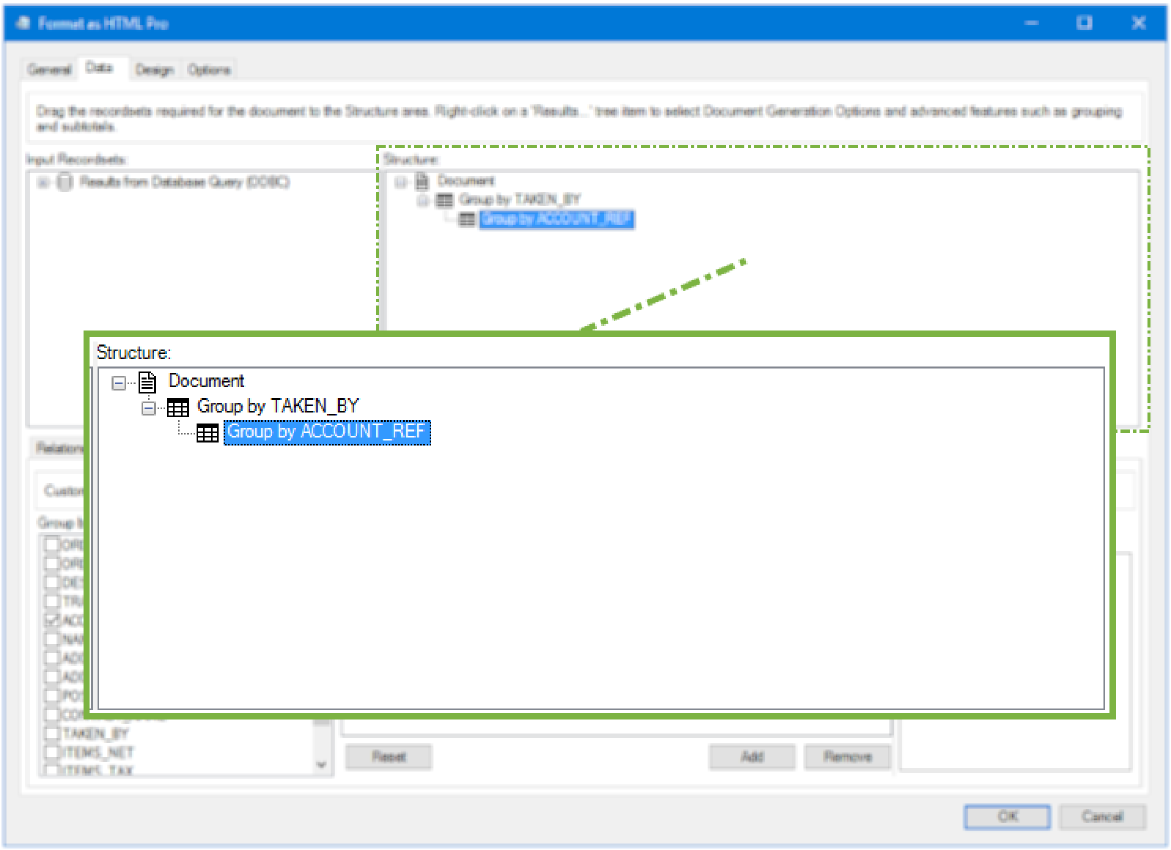
NOTE: When creating the group node, if more than one column is selected to group on, the grouping is composite rather than hierarchical. For a detailed description of how to create hierarchical groupings, refer to the product help.

To add a group node, you must convert the recordset node added previously to the group template (right-click the node > ***Templates* > *Convert to Group***). Then select the relevant **Column** and **Key**.

TIP: To see all grouped recordset data without any further manipulation, it is this node you drag to the HTML area in the **Design** tab (select **Table with Children** from the resultant menu).

Creating Hierarchical Groupings

To create a hierarchical grouping, you add the second group as child nodes to the initial group node created above:



Drag the recordsets required for the document to the Structure area. Right-click on a 'Results...' tree item to select Document Generation Options and advanced features such as grouping and subtotals.

TIP: If the child group must maintain an existing relationship higher in the **Structure**, select both the field to group by and the relationship fields. Then add relationships from all fields in this group node — see Specifying the Recordset Relationships.

Creating Sub Totals

You can create sub-totals for any required columns on either the group node (to generate a sub-total per group) or the recordset node (to generate a sub-total per recordset *not* per record).

To add sub-totals to the output, right-click on the required node and select ***Templates > Add Subtotals***.

Creating Grand Totals

Grand totals are sums of all values, typically currency, in a column in the recordset. They are added as child nodes to either the group node (to generate a grand -total per group) or the recordset node (to generate a total per recordset *not* per record).

You create grand totals in a similar way to creating groups.

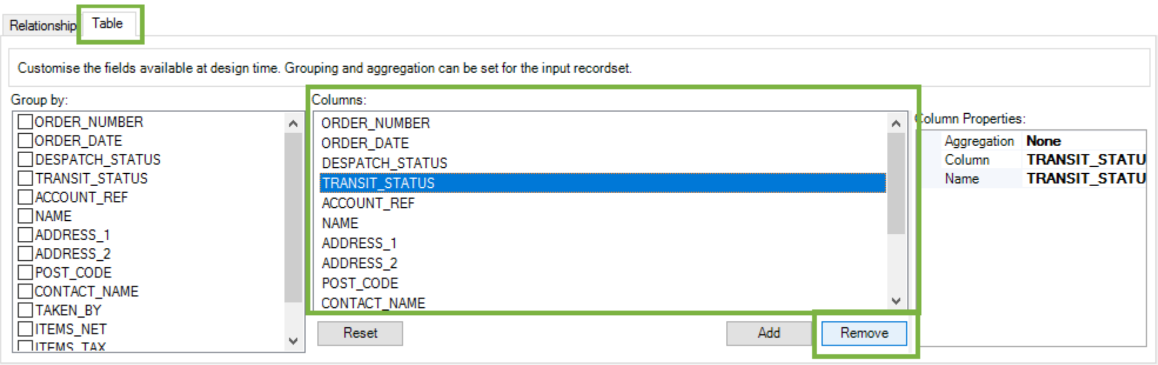
Automatically Creating Groups and Sub and Grand Totals

If consuming a single input recordset, you can create HTML tables in a similar vein to those available in the **Format as HTML** tool, with the sub and grand totals as part of the main table rather than embedded as sub-tables. To do this, select the **Convert to Flat Table** template for the input recordset.

Adjusting the Recordset Structure

By default, all columns in the consumed recordset(s) are included in the HTML output without any further data manipulation. You can tailor what is included in your HTML document to suit.

If required, you can exclude recordset columns from your HTML document. Highlight the relevant column in the **Columns** pane and click **Remove**.



Customise the fields available at design time. Grouping and aggregation can be set for the input recordset.

Group by:

- □ORDER_NUMBER
- □ORDER_DATE
- □DESPATCH_STATUS
- □TRANSIT_STATUS
- □ACCOUNT_REF
- □NAME
- □ADDRESS_1
- □ADDRESS_2
- □POST_CODE
- □CONTACT_NAME
- □TAKEN_BY
- □ITEMS_NET
- □ITEMS_TAX

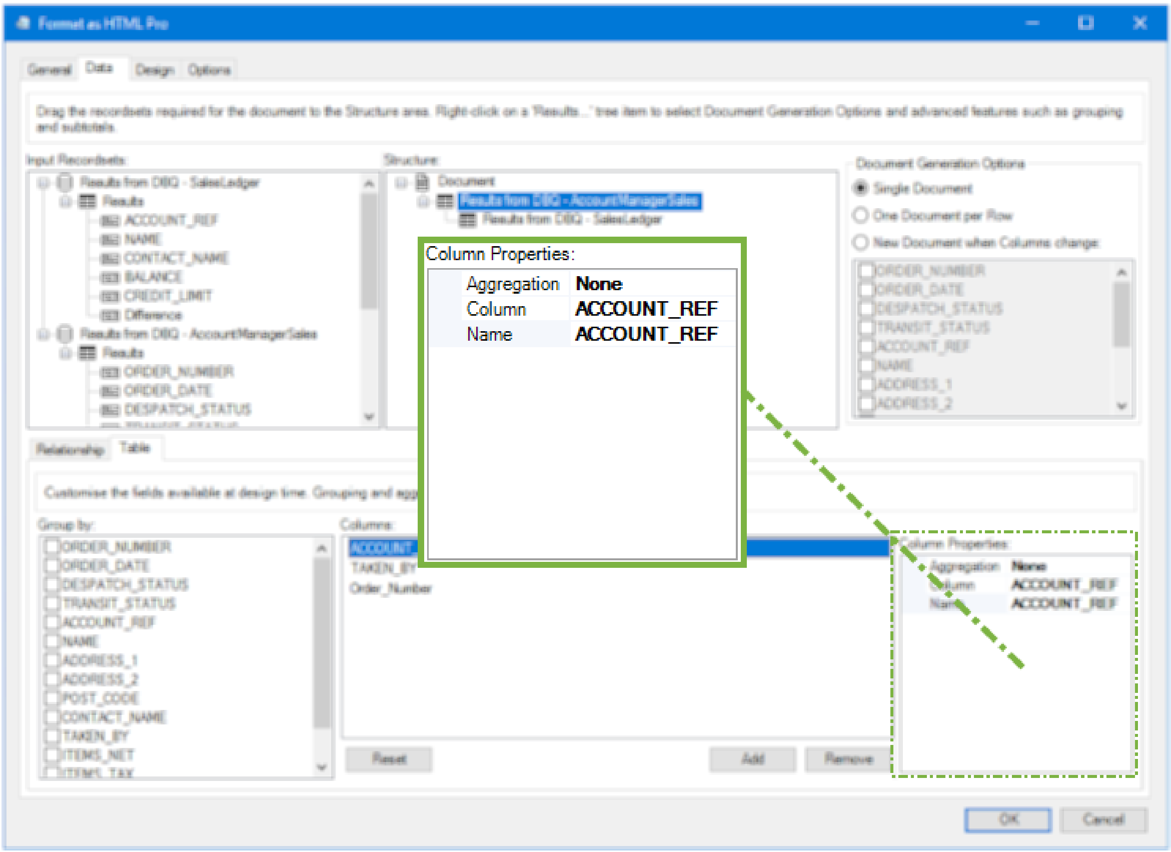
Columns:

- ORDER_NUMBER
- ORDER_DATE
- DESPATCH_STATUS
- TRANSIT_STATUS
- ACCOUNT_REF
- NAME
- ADDRESS_1
- ADDRESS_2
- POST_CODE
- CONTACT_NAME

Column Properties:

Aggregation	None
Column	TRANSIT_STATU
Name	TRANSIT_STATU

You can adjust recordset columns to suit — by changing the column name to a user-friendly name in the template, or by changing the aggregation type. Highlight the column and adjust the **Column Properties** to suit.



Drag the recordsets required for the document to the Structure area. Right-click on a 'Results...' tree item to select Document Generation Options and advanced features such as grouping and subtotals.

Input Recordsets:
- Results from DBQ - SalesLedger
  - Results
    - ACCOUNT_REF
    - NAME
    - CONTACT_NAME
    - BALANCE
    - CREDIT_LIMIT
    - Difference
- Results from DBQ - AccountManagerSales
  - Results
    - ORDER_NUMBER
    - ORDER_DATE
    - DESPATCH_STATUS

Structure:
- Document
  - Results from DBQ - AccountManagerSales
    - Results from DBQ - SalesLedger

Document Generation Options
- ● Single Document
- ○ One Document per Row
- ○ New Document when Columns change

Customise the fields available at design time. Grouping and agg

Group by:
- □ ORDER_NUMBER
- □ ORDER_DATE
- □ DESPATCH_STATUS
- □ TRANSIT_STATUS
- □ ACCOUNT_REF
- □ NAME
- □ ADDRESS_1
- □ ADDRESS_2
- □ POST_CODE
- □ CONTACT_NAME
- □ TAKEN_BY
- □ ITEMS_NET
- □ ITEMS_TAX

Columns:
- ACCOUNT_REF
- TAKEN_BY
- Order_Number

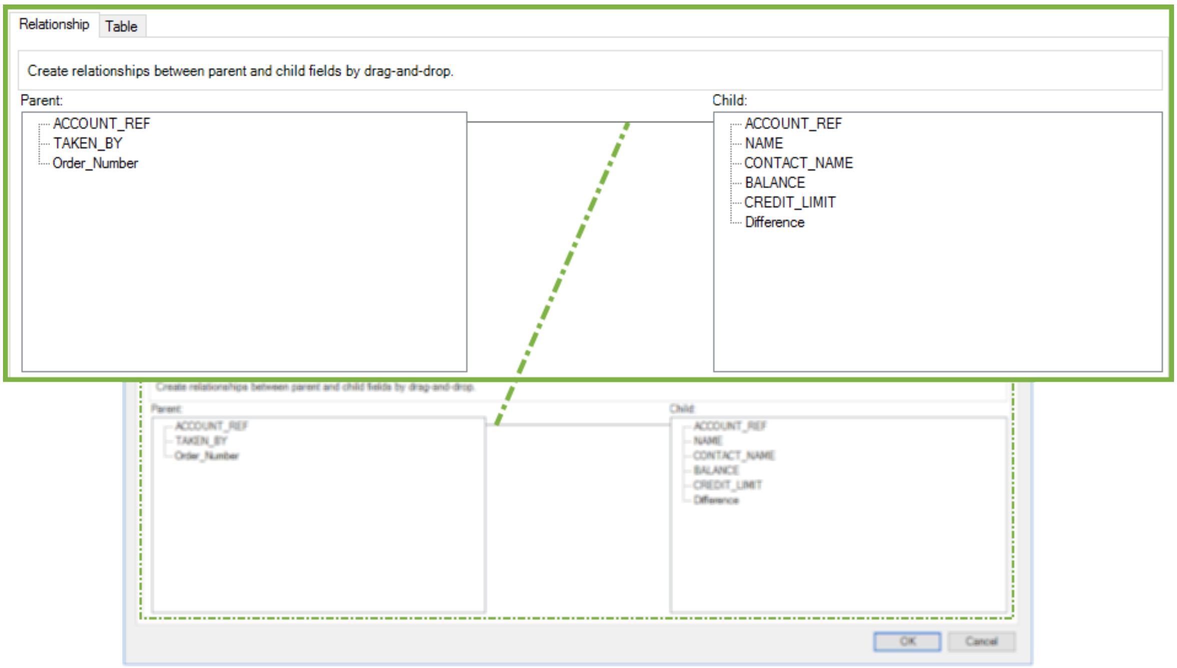
Column Properties:

Aggregation	None
Column	ACCOUNT_REF
Name	ACCOUNT_REF

By default, sub and grand total columns use the sum aggregation type. If required, you can create aggregations for other columns in the recordsets.

Specifying the Recordset Relationships

If your **Structure** has parent-child relationships, you must create the relationships between them to ensure the correct data is displayed in the document. You do this in the **Relationship** tab:



Create relationships between parent and child fields by drag-and-drop.

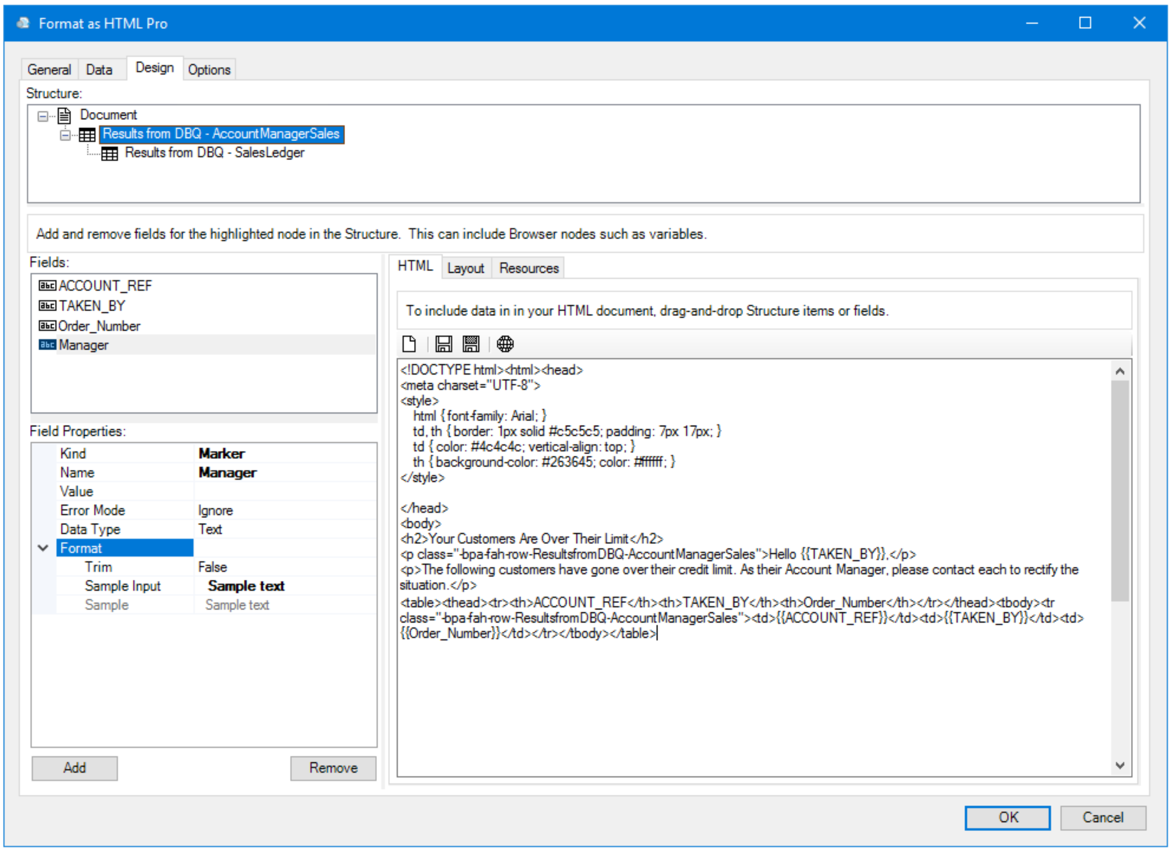
Parent:
- ACCOUNT_REF
- TAKEN_BY
- Order_Number

Child:
- ACCOUNT_REF
- NAME
- CONTACT_NAME
- BALANCE
- CREDIT_LIMIT
- Difference

Drag the relevant columns from the **Parent** pane to their equivalent in the **Child** pane. Do this for every parent child relationship you have created. Note that this feature is only available when the child node is highlighted in the **Structure** pane.

About the Design Tab

Create your HTML templates in the **Design** tab.



Add and remove fields for the highlighted node in the Structure. This can include Browser nodes such as variables.

Kind	**Marker**
Name	**Manager**
Value	
Error Mode	Ignore
Data Type	Text
Format	
Trim	False
Sample Input	**Sample text**
Sample	Sample text

To include data in in your HTML document, drag-and-drop Structure items or fields.

```
<!DOCTYPE html><html><head>
<meta charset="UTF-8">
<style>
   html { font-family: Arial; }
   td, th { border: 1px solid #c5c5c5; padding: 7px 17px; }
   td { color: #4c4c4c; vertical-align: top; }
   th { background-color: #263645; color: #ffffff; }
</style>

</head>
<body>
<h2>Your Customers Are Over Their Limit</h2>
<p class="-bpa-fah-row-ResultsfromDBQ-AccountManagerSales">Hello {{TAKEN_BY}},</p>
<p>The following customers have gone over their credit limit. As their Account Manager, please contact each to rectify the
situation.</p>
<table><thead><tr><th>ACCOUNT_REF</th><th>TAKEN_BY</th><th>Order_Number</th></tr></thead><tbody><tr
class="-bpa-fah-row-ResultsfromDBQ-AccountManagerSales"><td>{{ACCOUNT_REF}}</td><td>{{TAKEN_BY}}</td><td>
{{Order_Number}}</td></tr></tbody></table>
```

Styling the HTML Template

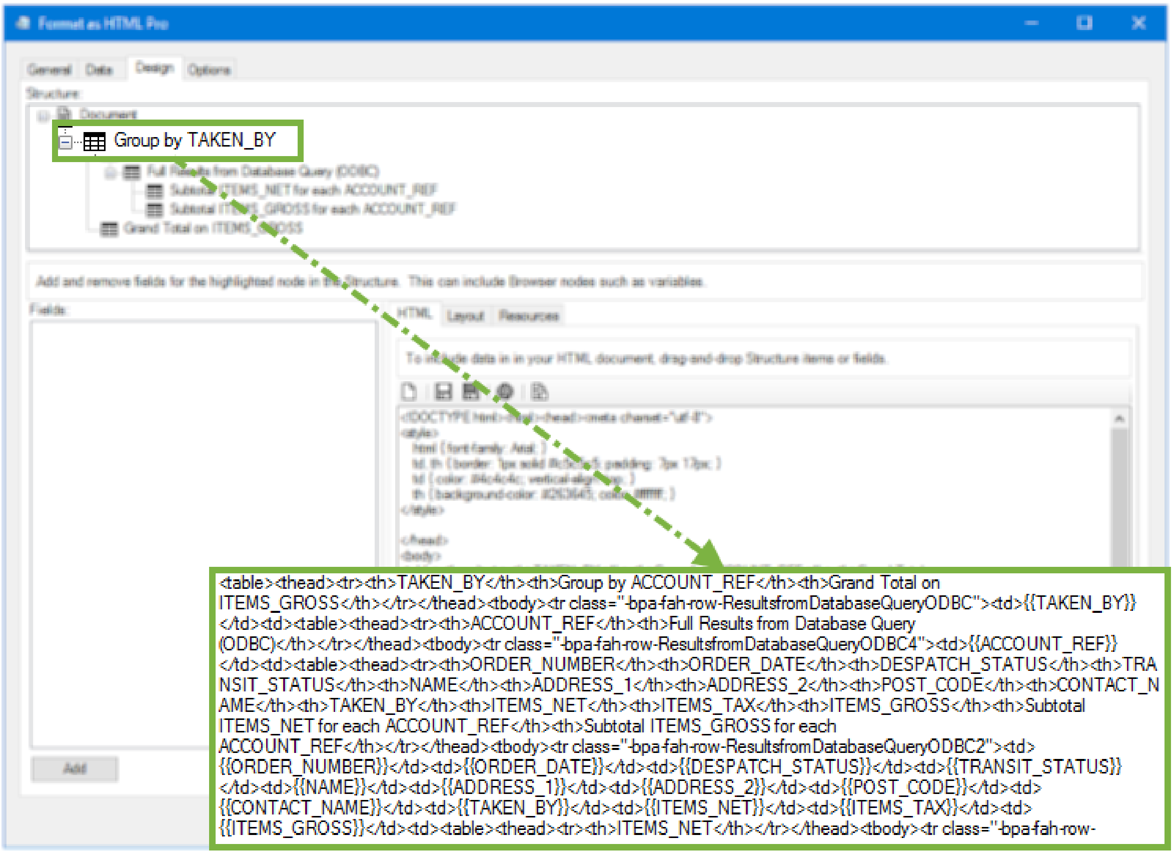
Basic in-line styles — for font and table styling — are available when you first create the HTML template.

You can either:

- Refer to a central stylesheet (CSS) via a URL
- Embed the CSS into the task step's local **Resources**
- Declare more in-line styles — this is recommended if CSS values are passed from an external location to the task step at run-time

Creating a Basic HTML Template

Use the **HTML** tab to add HTML elements to create the output. To add required BPA Platform data to your template, drag the results sets from **Structure** to the design area.


Format as HTML Pro
General
Data
Design
Options
Structure
Document
Group by TAKEN_BY
Full Results from Database Query (ODBC)
Subtotal ITEMS_NET for each ACCOUNT_REF
Subtotal ITEMS_GROSS for each ACCOUNT_REF
Grand Total on ITEMS_GROSS
Fields
HTML
Layout
Resources
Add

```
<table><thead><tr><th>TAKEN_BY</th><th>Group by ACCOUNT_REF</th><th>Grand Total on
ITEMS_GROSS</th></tr></thead><tbody><tr class="-bpa-fah-row-ResultsfromDatabaseQueryODBC"><td>{{TAKEN_BY}}
</td><td><table><thead><tr><th>ACCOUNT_REF</th><th>Full Results from Database Query
(ODBC)</th></tr></thead><tbody><tr class="-bpa-fah-row-ResultsfromDatabaseQueryODBC4"><td>{{ACCOUNT_REF}}
</td><td><table><thead><tr><th>ORDER_NUMBER</th><th>ORDER_DATE</th><th>DESPATCH_STATUS</th><th>TRA
NSIT_STATUS</th><th>NAME</th><th>ADDRESS_1</th><th>ADDRESS_2</th><th>POST_CODE</th><th>CONTACT_N
AME</th><th>TAKEN_BY</th><th>ITEMS_NET</th><th>ITEMS_TAX</th><th>ITEMS_GROSS</th><th>Subtotal
ITEMS_NET for each ACCOUNT_REF</th><th>Subtotal ITEMS_GROSS for each
ACCOUNT_REF</th></tr></thead><tbody><tr class="-bpa-fah-row-ResultsfromDatabaseQueryODBC2"><td>
{{ORDER_NUMBER}}</td><td>{{ORDER_DATE}}</td><td>{{DESPATCH_STATUS}}</td><td>{{TRANSIT_STATUS}}
</td><td>{{NAME}}</td><td>{{ADDRESS_1}}</td><td>{{ADDRESS_2}}</td><td>{{POST_CODE}}</td><td>
{{CONTACT_NAME}}</td><td>{{TAKEN_BY}}</td><td>{{ITEMS_NET}}</td><td>{{ITEMS_TAX}}</td><td>
{{ITEMS_GROSS}}</td><td><table><thead><tr><th>ITEMS_NET</th></tr></thead><tbody><tr class="-bpa-fah-row-
```

TIP: If you have created a specific data structure for your template (see About the Data Tab), drag the highest level node to the design area. This creates an embedded table structure.

When dragging nodes into the design area, a few templates are available to you:

- **Node ID** — The internal unique identifier of the recordset node as plain text.
- **Table** — The recordset data displayed as a table. If this recordset has children, selecting this option won't include them in the HTML template. All records from the recordset are used in the output template.
- **Table with Children** — The recordset data, displayed as a table, with all child recordsets included as an embedded table per row. This option is only available when using the parent recordset. The image above had this option selected.
- **Flat Table** — If you have used the **Convert to Flat Table** template when building your recordset structure, use this option to output an HTML document in the same format as outputted by the **Format as HTML** tool.

TIP: By default, results sets are formatted as tables when first dragged into the design area; you can adjust the HTML to display the data as lists, free-form, and so on, ensuring that you keep in place the recordset value markers of `{{` and `}}`.

Checking the HTML Design

Use the **Layout** tab to check the design of your HTML template. Note that the editor available in the **Layout** tab is not designed to create the whole template. Instead, use it to get your template "off the ground" then switch back to the **HTML** tab to complete it.

Use the **Launch** button in either the **HTML** or **Layout** tabs to open your template, with placeholders, in the default browser for your machine.

A Note About the BPA Platform-Specific HTML Classes

A few things to note about the HTML:

- `class="-bpa-fah-row-[`*`node_id`*`]"` — This class is a BPA Platform-specific class. It handles looping through the node_id's input recordset and outputs each record found on a new line. You can add this class to any valid HTML element (`p`, `span`, `div`, and so on).
    - □ Putting `class="-bpa-fah-row-[`*`parent_node_id`*`]"` in a `<tr>` element renders a new row in the table for each record in the input recordset. If this was the `<table>` element, it generate a new table for every record.

TIP: To find the ID for a node, drag it from the **Structure** pane into some free space in the HTML design area and select **Node ID** from the menu.

- `class="-bpa-fah-node-[`*`node_id`*`]"` — This class is a BPA Platform-specific class. It handles looping through the node_id's input recordset and outputs each record contiguously.

NOTE: Do not apply CSS (styling) to these classes; it is ignored and removed at task run-time.

To see examples of usage for each class, refer to the product help and guides.

- Marker fields — Marker fields are those wrapped in `{{ }}`. These are placeholders for BPA Platform data; in this scenario, for recordset data. The markers must belong to a parent `-bpa-class` otherwise a validation error may occur.

Each BPA Platform-specific class is valid until the closing tag in which it resides — for example, if you put a class inside a `div`, all markers encountered in that `div` must belong to the declared node ID. Markers from other node IDs placed inside the `div` will generate an error unless their class is also declared inside the `div`. In fact, it is perfectly acceptable to declare a BPA Platform class inside the `body` tag, however, you should only declare the highest level parent node here. If your data structure has more than one parent node at the same level, you shouldn't declare any BPA Platform classes in the `body` tag.

Using Format as HTML Pro's Advanced Features

NOTE: Although HTML 4 and 5 are supported, XHTML elements are not.

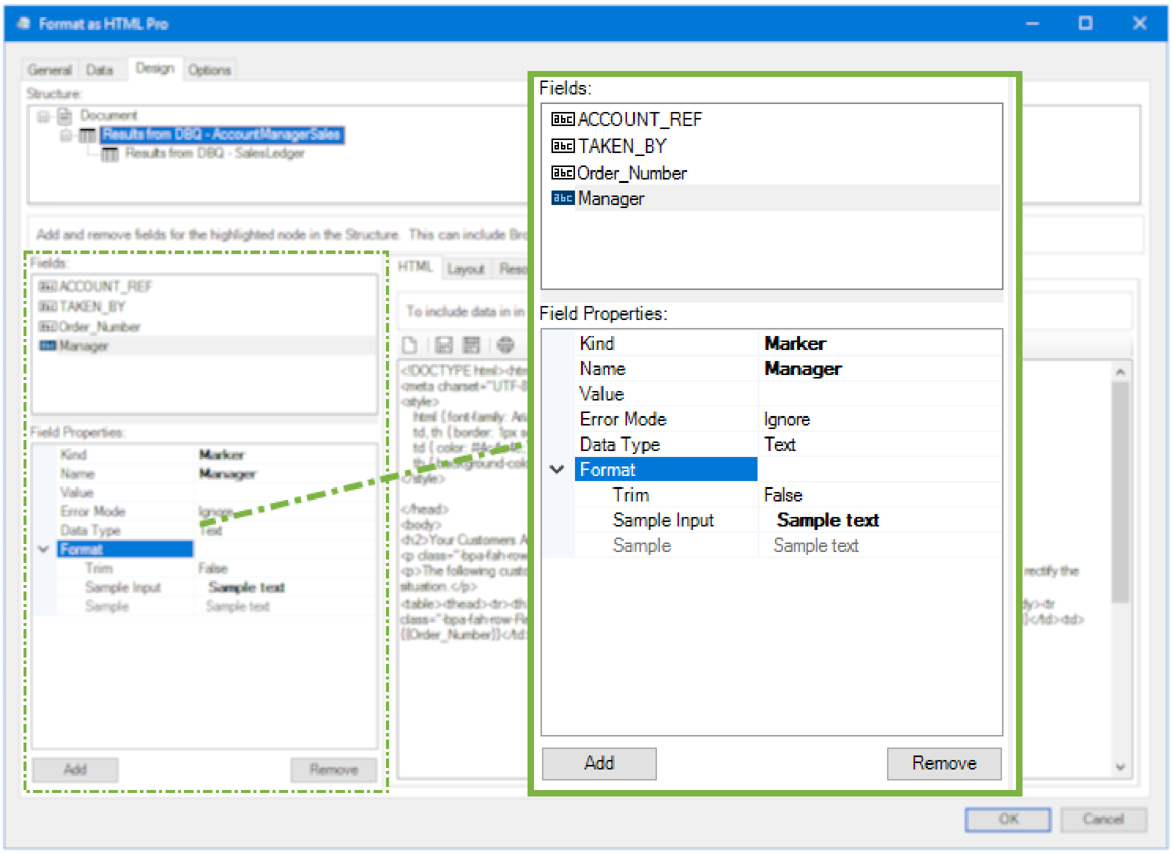
Importing HTML Templates

If you have created a template in an external HTML editor, you can import it into the **Format as HTML Pro** tool and add data sources to it — from the **HTML** tab's toolbar, click **Open**.

NOTE: The **Format as HTML Pro** tool behaves in the same manner when importing HTML templates as creating **New** templates — all customised HTML and local resources are erased.

Adding Additional Fields to the Results Sets

If required, you can add additional fields and markers to your HTML template for other BPA Platform data outside of the recordsets or for static values. Typically, such fields are added to the **Document** level of the **Structure**, however, you can add additional fields and markers to any level of the structure.


Format as HTML Pro
General
Data
Design
Options
Structure:
Document
Results from DBQ - AccountManagerSales
Results from DBQ - SalesLedger
Add and remove fields for the highlighted node in the Structure. This can include Bro
HTML
Layout
Reso
Add
Remove
OK
Cancel

Fields:

- ACCOUNT_REF
- TAKEN_BY
- Order_Number
- Manager

Field Properties:

Kind	**Marker**
Name	**Manager**
Value	
Error Mode	Ignore
Data Type	Text
Format	
Trim	False
Sample Input	**Sample text**
Sample	Sample text

If changing the **Culture** format, the available list is determined by your operating system. For a detailed description of each culture code, refer to your operating system's documentation.

The **Raw** format available with the **Text** data type determines how HTML elements found in this field are processed:

- **True** — If data in the field contains HTML tags, enabling this option processes such data as HTML and outputs accordingly. For example, `<b>Name</b>,<em>Description</em>` would be outputted as **Name**, *Description*.
- **False** — (Default) All data in the field, including any HTML tags, is processed as plain text and outputted as such. For example, `<b>Name</b>,<em>Description</em>` would be outputted exactly as you see it here.

Managing Embedded Local Resources

A local resource is that which resides on the local machine and is referenced in the HTML — for example `<img src="C:\Users\Administrator\Desktop\format-as-html-pro.png" />`. The **Format as HTML Pro** tool can extract locally declared resources and cache them in the task step for faster retrieval. This changes the previous URL to `<img src="format-as-html-pro.png" />`.

The following embedded resources are supported:

- Images (namely `.bmp`, `.gif`, `.hdp`, `.jpeg`, `.jpg`, `.jxr`, `.png`, `.svg`, and `.wdp`)
- CSS files
- JavaScript (JS) files
- XML files
- Other `.htm` or `.html` files

> **NOTE:** When an HTML document referencing local resources is saved, transferred, emailed, and so on, the resources are delivered with the main document. If emailing, these are sent as attachments.

If caching of local resources is required, do one of the following:

- From the **Resources** tab, click **Add** to locate and **Open** the required resource

  To use the resource in your HTML template, format your HTML class value as `"`*`filename.file_extension`*`"` instead of specifying the whole path.

- Add the resource to your HTML, including the full path, as normal (for example, `<img src="C:\Users\Administrator\Desktop\myimage.png" />`)

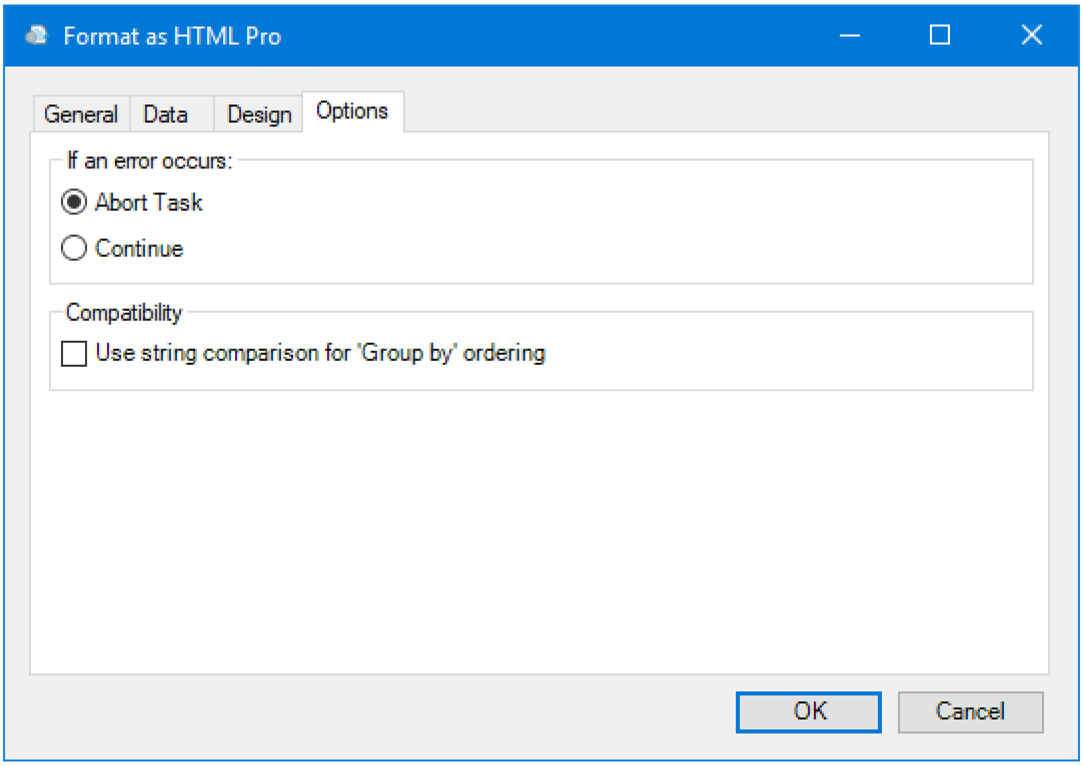
  When all resources have been added, click **Extract Resources**; all resources are added to the **Resources** tab.

NOTE: If importing an HTML template, all referenced files with the exception of internet references are automatically extracted and cached to the **Resources** tab. This includes images that are referenced:

- Relative to the local template, that is, `..\images\myimage.png`
- Using the full local path, that is, `C:\Users\Administrator\Desktop\myimage.png`
- Using a network path, that is, `\\mynetwork\images\myimage.png`

About the Options Tab

The **Options** tab allows you to define how errors in this step are handled at task runtime.



If an error occurs:

● Abort Task

○ Continue

Compatibility

☐ Use string comparison for 'Group by' ordering

If an error occurs, you can decide whether the step should **Continue** processing, or terminate the step immediately (**Abort Task**).

By default, data from grouped nodes are sorted according to their values; if grouping by a numeric field, this results in the following output:

1, 2, 3 ... 10, 11, 12 ... 20, 21, 22 ... 30, 31, 32, and so on.

If string-based sorting is required, enable **Use string comparison for 'Group by' ordering** in the **Compatibility** panel; numeric fields would then output as:

1, 10, 11 ... 2, 20, 21 ... 3, 30 31, and so on.

Want to learn more?

Discover how Codeless Platforms can help your business by improving performance, boosting efficiency and cutting costs.

+44 (0) 330 99 88 700

enquiries@codelessplatforms.com

www.codelessplatforms.com

codeless
PLATFORMS