

History

This document was originally published in the July 2007 issue of Cognos SupportLink and tested with Cognos 8 and 8.2. The concepts still apply to IBM Cognos 8.3 and 8.4, however the implementation may be different. An update to this document is forthcoming.

Introduction

Cognos 8 and Report Studio use Cascading Style Sheets, CSS, to provide default styles, (i.e. colors, fonts, sizes) to objects like tables and lists. While Cognos provides many methods for modifying the default styles, we can often run into problems with reusability.

Another common issue is modifying the default styles programmatically. In order to apply stripes or freeze table headers we often need to add small bits of JavaScript to individual reports or physically separate the column headers from the report. This inhibits reusability. Luckily Microsoft Internet Explorer supports a method of combining small snippets of JavaScript with CSS, thus providing easily reusable styles.

In the following examples I will take advantage of 2 Advanced CSS techniques, CSS parent child relationships in CSS and Internet Explorer's implementation of CSS expressions to provide rich reusable styles.

Report CSS

The default formatting for Report Studio reports are controlled from the CSS file GlobalReportStyles.css, located in the install dir \webcontent\schemas. This file contains default formatting for most report objects, but for our example we are only concerned with the first 3 entries that affect the List Report object. Scanning the file we find the following section:

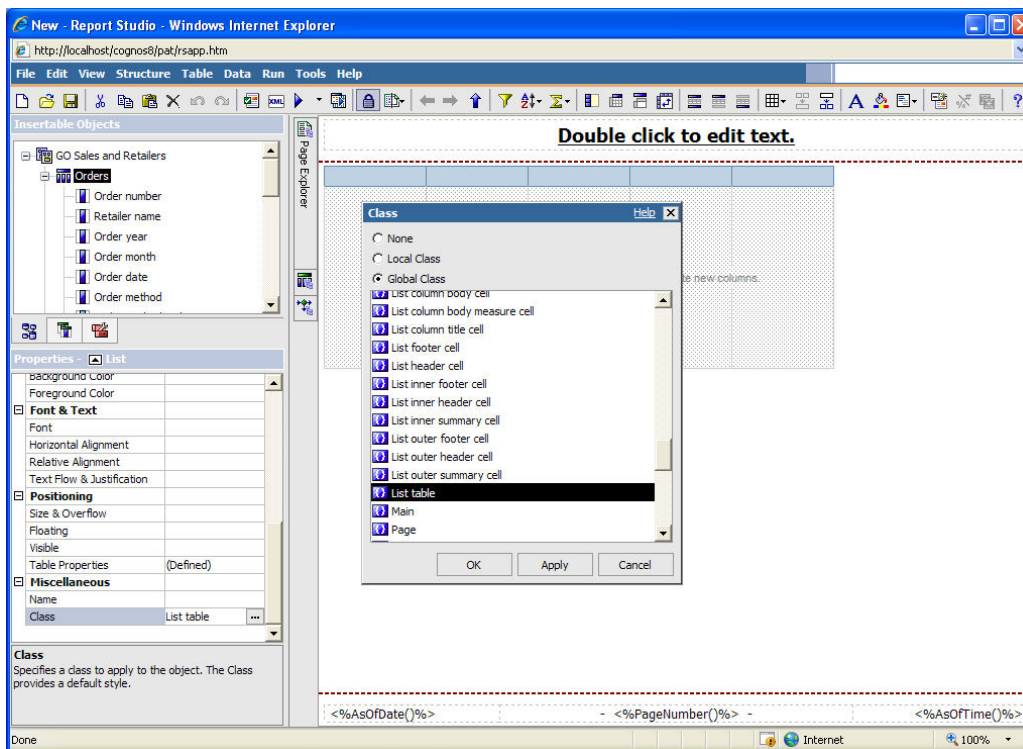
```
/* List Styles */
/* ----- */

.ls /* list */
{
    font-size: 8pt;
}

.lt /* list column title cell */
{
    text-align: center;
    vertical-align: top;
    line-break: strict;
    border: 1px solid #608BB4;
    background-color: #BFD2E2;
    padding: 3px 5px;
}

.lc /* list column body cell */
{
    vertical-align: top;
    line-break: strict;
    border: 1px solid #CCCCCC;
    padding: 3px 5px;
}
```

What we see are classes and their default formatting properties. For example, the class .ls indicates that any object with the class property set to ls will have a font size of 8pt. We can overwrite these properties in Report Studio either through the properties pane or with local classes. These classes are given more descriptive names and we can access them in report studio through the properties pane.



By default we see a name assigned to the class in this case “List Table” refers to the class “.Is”

Modifying the CSS

At this point it would be easy to simply change the entry in the CSS file to whatever format we want, however the format change would affect every list report. We could make the change in the report at the local class level; however that would only affect lists in the current report. We could also create new class entries for the List, the Column Heading, and the Body Columns, but this approach means that we would manually have to change the class property for each of those objects and items.

Instead we want to create classes that would apply across all reports and still take advantage of the default classes assigned to the list object when it is created. To do this we will take advantage of Inheritance or Parent Child relationships in CSS. A class can belong to another class as a child, meaning that the class is only applied if its parent class is assigned to an object as well.

Sample Report

Let's create a simple report for this example. Using the Go Sales and Retailers package open Report Studio and create a new list report. From the Products Query Subject drag Product Line, Product Type, and Product Name into the list. In order to make the list more manageable lets add the following filter; Product Line = "Golf Equipment".

The sample report should now look like this:

Product line	Product type	Product name
Golf Equipment	Golf Accessories	Course Pro Gloves
Golf Equipment	Golf Accessories	Course Pro Golf and Tee Set
Golf Equipment	Golf Accessories	Course Pro Golf Bag
Golf Equipment	Golf Accessories	Course Pro Umbrella
Golf Equipment	Irons	Hailstorm Steel Irons
Golf Equipment	Irons	Hailstorm Titanium Irons
Golf Equipment	Irons	Lady Hailstorm Steel Irons
Golf Equipment	Irons	Lady Hailstorm Titanium Irons
Golf Equipment	Putters	Blue Steel Max Putter
Golf Equipment	Putters	Blue Steel Putter
Golf Equipment	Putters	Course Pro Putter
Golf Equipment	Woods	Hailstorm Steel Woods Set
Golf Equipment	Woods	Hailstorm Titanium Woods Set
Golf Equipment	Woods	Lady Hailstorm Steel Woods Set
Golf Equipment	Woods	Lady Hailstorm Titanium Woods Set

Save the report.

Minimal Styling

Make a backup copy of the GlobalReportStyles.css. Then add the following to the bottom of the file.

1. `/* Custom Styles */`
This line is a comment to let us know where our style begins
2. Add an entry `.blue { }` under the comment
blue is the name of our class. The formatting code usually goes between the curly brackets `{ }`, but in this case the class is simply going to be used as a parent class.
3. Copy and paste the first 3 `.classes` from the list classes under the comment `/* List Styles */`
We are going to use these 3 classes to help format our list
4. Add the following entries `table{ }`, `row{ }`.
We are going to apply formatting to these objects. These are objects as opposed to classes
5. Finally insert the parent class `.blue` in front of each subsequent entries as `.blue .ls { ... }`
This will distinguish the classes that we modify from the original entries. This way they can share the same class name, but the modified versions will only be applied when the `.blue` class is applied first

The entries should now look like this:

```
/* Custom Style */
.blue{ }
.blue table { }
.orange .tr{ }

.blue .ls /* list */
{
    font-size: 8pt;
}

.blue .lt /* list column title cell */
{
    text-align: center;
    vertical-align: top;
    line-break: strict;
    border: 1px solid #608BB4;
    background-color: #BFD2E2;
    padding: 3px 5px;
}

.blue .lc /* list column body cell */
{
    vertical-align: top;
    line-break: strict;
    border: 1px solid #CCCCCC;
    padding: 3px 5px;
}
```

Now modify the entries to match the code below. For the .lt class we added an optional background image to provide a subtle 3d effect. If it is not found the .lt class will use a light blue color for the background.

```
/* Custom Style */

.blue{
background-color: #FFFFFF;
}
.blue table{
border:none;
}

.blue row {}

.blue .lt {
font: bold 11px "Trebuchet MS", Verdana, Arial, Helvetica, sans-serif;
color: #4f6b72; border-right: 1px solid #C1DAD7;
border-bottom: 1px solid #C1DAD7;
border-top: 1px solid #C1DAD7;
letter-spacing: 2px;
text-align: left;
padding: 6px 6px 6px 12px;

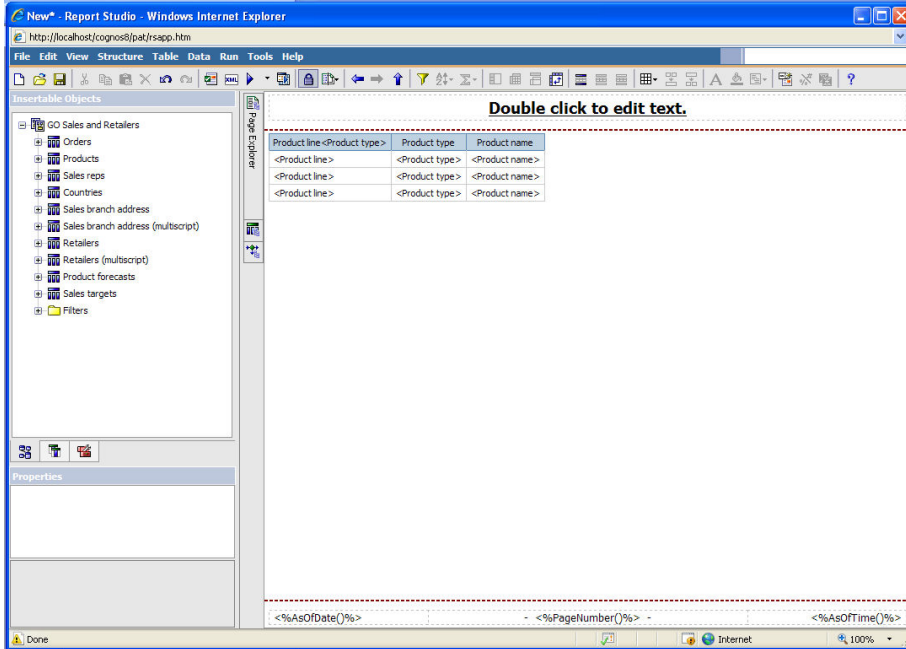
background: #CAE8EA url(../common/images/bg_header.jpg) no-repeat;
}

.blue .lc {
border-right: 1px solid #C1DAD7;
border-bottom: 1px solid #C1DAD7;
padding: 6px 6px 6px 12px;
color: #4f6b72;}

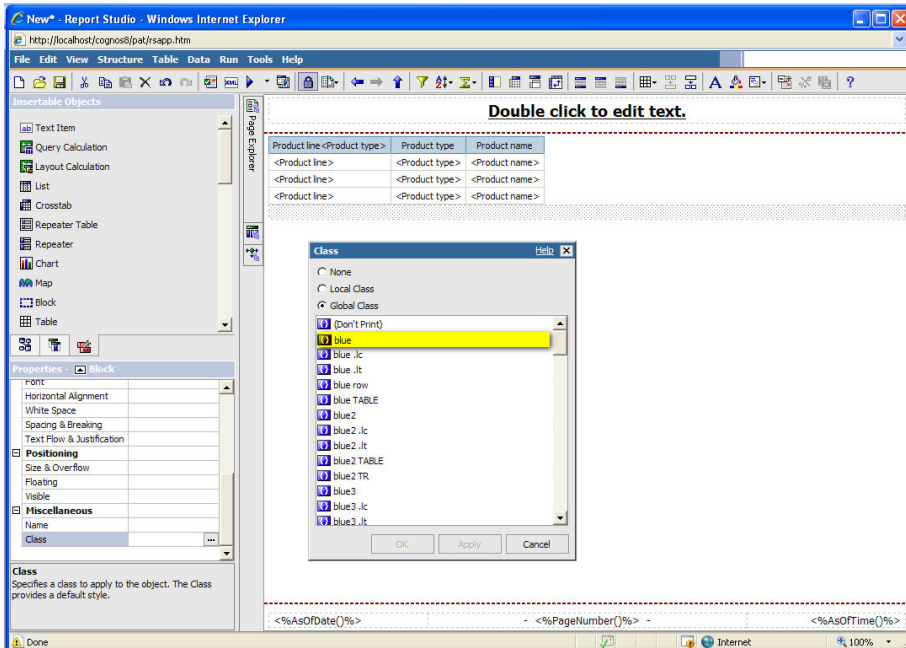
```

Applying the Minimal Style

To test the new style lets open the sample report

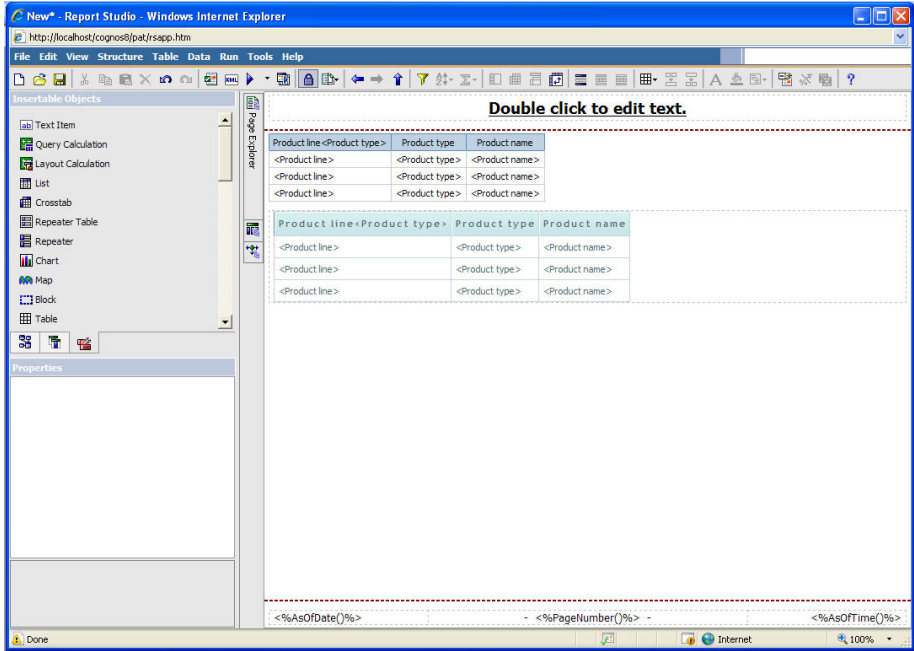


From the insertable objects pane drag a block onto the report below the list. Select the block and from the properties pane select the class property and select the blue class.



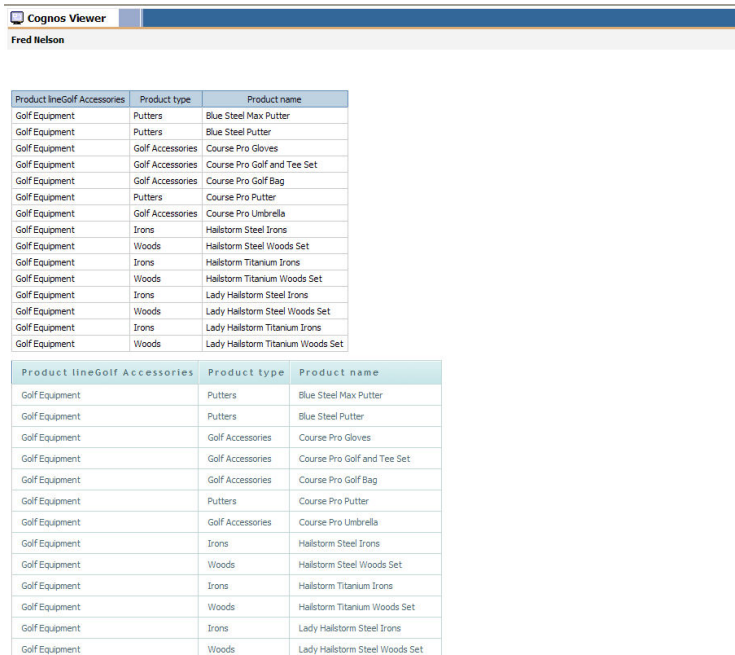
We now applied the parent class to the block any object with the classes's .list, .lt, .lc placed inside the block should take on the style properties we created.

To test this, select the list and copy it. Highlight the block and paste the copy into it.



As you can see the list now takes on the new style immediately simply by placing it in the block with the class property set to blue. If we take it out of the block or remove the class blue from the block the list report will revert back to the default style.

If we run the report we see the style's full effect.



By using this method we are able to take advantage of the default class properties and facilitate re-use. The class property blue can be applied to either report pages or blocks, in order to apply the style formatting to any list inside of it. We can also place multiple blocks on a single report each with a different style.

Advanced Styling

Add Stripes and Text Transform

In this example we will combine CSS and JavaScript to programmatically affect the content of the list report.

First we will add a CSS property called text-transform. The property text-transform can affect text with 3 values:

capitalize	Will change the text so each word starts with a capital letter
uppercase	Will change the text so all text is in capital letters
lowercase	Will change the text so all text is only in lower case letters

From our previous example lets add the following to the .blue .lt { } CSS entry

```
text-transform: uppercase;
```

This will make all text in the list headers capitalized

Next we will use a CSS expression to programmatically add stripes to the list report. A CSS Expression is a small piece of JavaScript that can be added directly to a CSS property.

Again from our previous example add the following line to the .blue .tr { } entry

```
.blue2 tr{  
background-  
color:expression(rowIndex%2?(parentNode.parentNode.rowHighlightColor||"#F5FAFA"): "");  
color:#797268;  
}
```

I won't go into the JavaScript, but this line of code looks for every other row and sets the background color to #F5FAFA, which is a very light blue. This provides subtle differences in row colors.

Our CSS entries should now look like the following:

```
.blue{
background-color: #FFFFFF;
}
.blue table{
border:none;
}

.blue tr{
background-
color:expression(rowIndex%2?(parentNode.parentNode.rowHighlightColor||"#F5FAFA"): "");
color:#797268;
}

.blue .lt {
font: bold 11px "Trebuchet MS", Verdana, Arial, Helvetica, sans-serif;
color: #4f6b72; border-right: 1px solid #C1DAD7;
border-bottom: 1px solid #C1DAD7;
border-top: 1px solid #C1DAD7;
letter-spacing: 2px;
text-transform: uppercase;
text-align: left;
padding: 6px 6px 6px 12px;
background: #CAE8EA url(../common/images/bg_header.jpg) no-repeat;}

.blue .lc {
border-right: 1px solid #C1DAD7;
border-bottom: 1px solid #C1DAD7;
padding: 6px 6px 6px 12px;
color: #4f6b72;}
```

If we run our previous example we see that the header titles are now all caps and every other row is a light blue. As we already mentioned we only need to assign one class, in this case blue, to a parent object for the list to take on the new style and not affect any previously created reports

Cognos Viewer
Fred Nelson

Product line	Product type	Product name
Golf Equipment	Golf Accessories	Course Pro Gloves
Golf Equipment	Golf Accessories	Course Pro Golf and Tee Set
Golf Equipment	Golf Accessories	Course Pro Golf Bag
Golf Equipment	Golf Accessories	Course Pro Umbrella
Golf Equipment	Irons	Hallstorm Steel Irons
Golf Equipment	Irons	Hallstorm Titanium Irons
Golf Equipment	Irons	Lady Hallstorm Steel Irons
Golf Equipment	Irons	Lady Hallstorm Titanium Irons
Golf Equipment	Putters	Blue Steel Max Putter
Golf Equipment	Putters	Blue Steel Putter
Golf Equipment	Putters	Course Pro Putter
Golf Equipment	Woods	Hallstorm Steel Woods Set
Golf Equipment	Woods	Hallstorm Titanium Woods Set
Golf Equipment	Woods	Lady Hallstorm Steel Woods Set
Golf Equipment	Woods	Lady Hallstorm Titanium Woods Set

PRODUCT LINE	PRODUCT TYPE	PRODUCT NAME
Golf Equipment	Golf Accessories	Course Pro Gloves
Golf Equipment	Golf Accessories	Course Pro Golf and Tee Set
Golf Equipment	Golf Accessories	Course Pro Golf Bag
Golf Equipment	Golf Accessories	Course Pro Umbrella
Golf Equipment	Irons	Hallstorm Steel Irons
Golf Equipment	Irons	Hallstorm Titanium Irons
Golf Equipment	Irons	Lady Hallstorm Steel Irons
Golf Equipment	Irons	Lady Hallstorm Titanium Irons
Golf Equipment	Putters	Blue Steel Max Putter
Golf Equipment	Putters	Blue Steel Putter
Golf Equipment	Putters	Course Pro Putter
Golf Equipment	Woods	Hallstorm Steel Woods Set
Golf Equipment	Woods	Hallstorm Titanium Woods Set

The beauty of this method for striping a table is that it is only one line of code and can be reused across all reports.

Static Header

In this example we will again take advantage of CSS expressions to embed JavaScript in the CSS entries. Here we will add a static non-scrolling header to the list. This will allow a long list report to be scrollable while the header always remains visible.

First we will set our parent object to a specific height and set it to scroll automatically. From our previous example add the following properties to the `.blue { }` CSS entry

```
.blue{  
    background-color: #FFFFFF;  
    border: solid 1px black;  
    width: 50%;  
    height: 150px;  
    overflow: auto;  
}
```

Second lets add the following to the class `.blue tr { }`

```
position:relative;  
top: expression(parentNode.parentNode.parentNode.parentNode.scrollTop-1);
```

This line add a small snippet of JavaScript to the CSS property "top" that will keep the header row from moving if the list report is scrolled downward.

Our CSS should now look like this:

```
.blue{
    background-color: #FFFFFF;
    border: solid 1px black;
    width: 50%;
    height: 150px;
    overflow: auto;
}

.blue table{
border:none;
}

.blue tr{
background-
color:expression(rowIndex%2?(parentNode.parentNode.rowHighlightColor||"#F5FAFA"): "");
color:#797268;
}

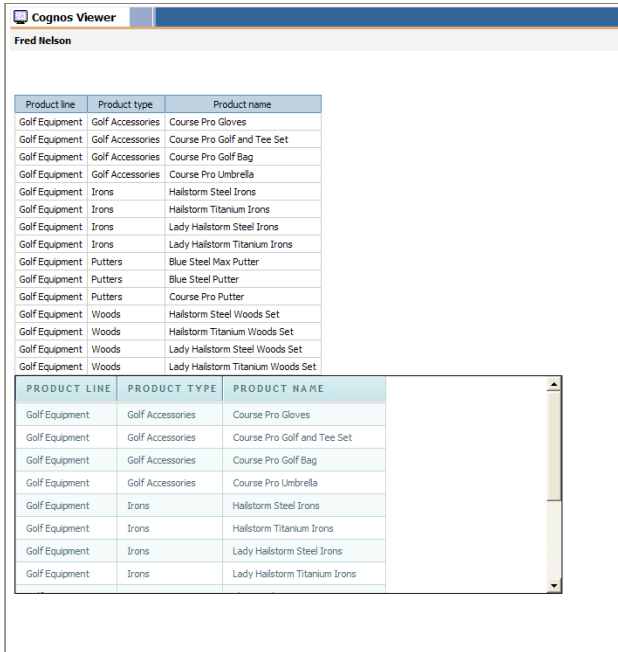
}

.blue .lt {
font: bold 11px "Trebuchet MS", Verdana, Arial, Helvetica, sans-serif;
color: #4f6b72; border-right: 1px solid #C1DAD7;
border-bottom: 1px solid #C1DAD7;
border-top: 1px solid #C1DAD7;
letter-spacing: 2px;
text-transform: uppercase;
text-align: left;
padding: 6px 6px 6px 12px;
background: #CAE8EA url(../common/images/bg_header.jpg) no-repeat;
position:relative;
top: expression(parentNode.parentNode.parentNode.parentNode.scrollTop-1);
}

}

.blue .lc {
border-right: 1px solid #C1DAD7;
border-bottom: 1px solid #C1DAD7;
padding: 6px 6px 6px 12px;
color: #4f6b72;}
```

When we run the report we can see that report has a scroll bar and that the header row stays in place as we scroll through the rows.



The screenshot shows the Cognos Viewer interface. At the top, there is a header bar with the Cognos logo and the text 'Cognos Viewer'. Below this, the user's name 'Fred Nelson' is displayed. The main content area contains a table with the following data:

Product line	Product type	Product name
Golf Equipment	Golf Accessories	Course Pro Gloves
Golf Equipment	Golf Accessories	Course Pro Golf and Tee Set
Golf Equipment	Golf Accessories	Course Pro Golf Bag
Golf Equipment	Golf Accessories	Course Pro Umbrella
Golf Equipment	Irons	Halstorm Steel Irons
Golf Equipment	Irons	Halstorm Titanium Irons
Golf Equipment	Irons	Lady Halstorm Steel Irons
Golf Equipment	Irons	Lady Halstorm Titanium Irons
Golf Equipment	Putters	Blue Steel Max Putter
Golf Equipment	Putters	Blue Steel Putter
Golf Equipment	Putters	Course Pro Putter
Golf Equipment	Woods	Halstorm Steel Woods Set
Golf Equipment	Woods	Halstorm Titanium Woods Set
Golf Equipment	Woods	Lady Halstorm Steel Woods Set
Golf Equipment	Woods	Lady Halstorm Titanium Woods Set

Below the main table, there is a smaller, scrollable version of the same table. This version has a double-line border around the header row and a vertical scrollbar on the right side. The data in this table is identical to the main table above.

The width of the block can be changed at the report level to eliminate any unnecessary white space.

Also, if the header row seems jittery as you scroll through the report, try adding a double line border to the block.

As with the striping, this method allows us to use a single line of code to get the desired effect and reuse it across many reports.

Static Column

Finally in this example we will again use CSS expressions. This time we will add a static non-scrolling column to a wide report. This will allow a wide report to be horizontally scrollable while a column or a set of columns remain static.

For this example we will create the class independently of a parent class.

From our previous example copy the entries “.blue .lt” and “.blue .lc” to the bottom of the CSS file. Then rename the copied .blue .lt entry to .lockColHead and rename the .blue .lc entry to .lockCol.

```
.lockColHead{  
}  
  
.lockCol{  
}
```

Then add the following properties to each entry:

```
left: expression(parentNode.parentNode.parentNode.parentNode.scrollLeft); /*IE5+ only*/  
position: relative;
```

This line adds a snippet of JavaScript to the property left

Additionally add the z-index property to the .lockColHead class:
z-index:10;

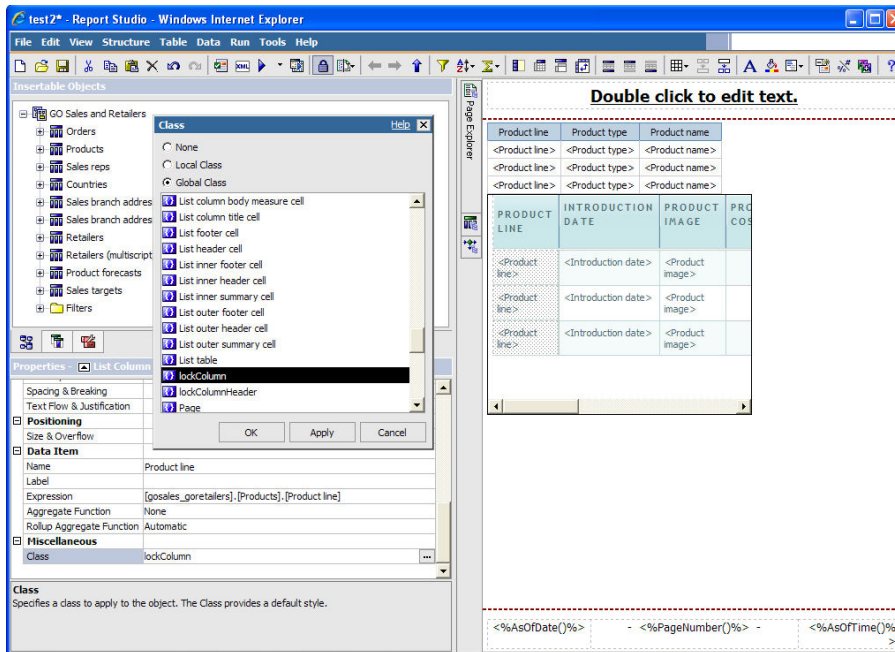
This line makes sure that the column header always remains visible as the columns or rows are scrolled past it.

The CSS entries should look like this:

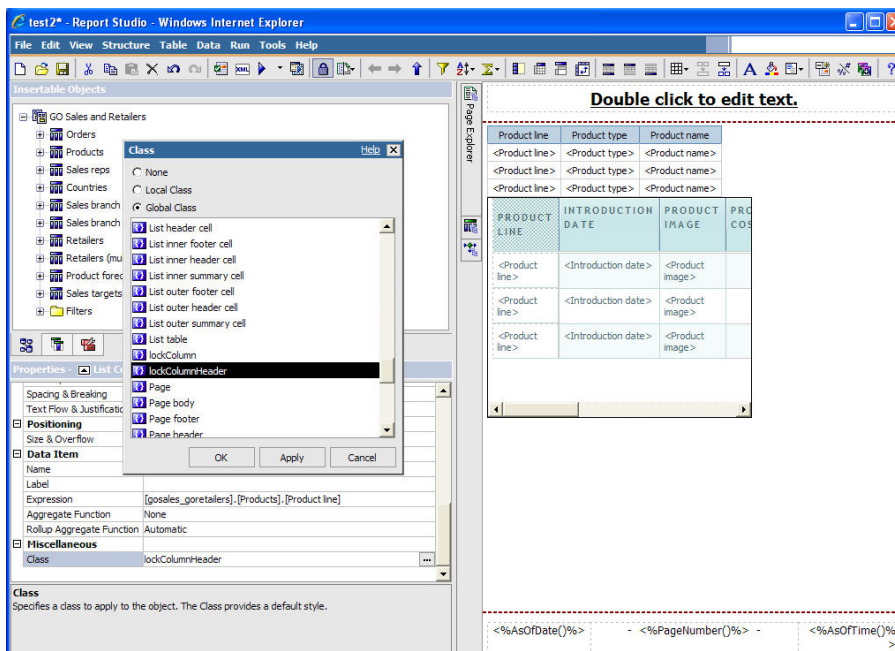
```
/*column Lock*/
.lockColumnHeader{
font: bold 11px "Trebuchet MS", Verdana, Arial, Helvetica, sans-serif;
color: #4f6b72; border-right: 1px solid #C1DAD7;
border-bottom: 1px solid #C1DAD7;
border-top: 1px solid #C1DAD7;
letter-spacing: 2px;
text-transform: uppercase;
text-align: left;
padding: 6px 6px 6px 12px;
background: #CAE8EA url(../common/images/bg_header.jpg) no-repeat;
top: expression(parentNode.parentNode.parentNode.parentNode.scrollTop-1);
left: expression(parentNode.parentNode.parentNode.parentNode.scrollLeft);
position: relative;
z-index:10;
}

.lockColumn{
border-right: 1px solid #C1DAD7;
border-bottom: 1px solid #C1DAD7;
padding: 6px 6px 6px 12px;
color: #4f6b72;
background-color:#FFFFFF;
left: expression(parentNode.parentNode.parentNode.parentNode.scrollLeft);
position: relative;
}
```

Let's open our example report and add additional columns to list report from the products query subject. Add enough columns that will force the report to scroll horizontally. Then select and highlight the first column and change the class property to lockColumn.



Select the header for the first column and change the class property to lockColumnHeader.



When we run the report we see that our styled report has a horizontal scrollbar. If we scroll to the right we see that the first column stays frozen in place.

Cognos Viewer
Fred Nelson

Golf Equipment	Irons	Lady Hallstorm Steel Irons	
Golf Equipment	Irons	Lady Hallstorm Titanium Irons	
Golf Equipment	Putters	Blue Steel Max Putter	
Golf Equipment	Putters	Blue Steel Putter	
Golf Equipment	Putters	Course Pro Putter	
Golf Equipment	Woods	Hallstorm Steel Woods Set	
Golf Equipment	Woods	Hallstorm Titanium Woods Set	
Golf Equipment	Woods	Lady Hallstorm Steel Woods Set	
Golf Equipment	Woods	Lady Hallstorm Titanium Woods Set	

PRODUCT LINE	PRODUCT TYPE	PRODUCT NAME	DESCRIPTION
Golf Equipment	Putters	Blue Steel Max Putter	Putter he composec single pie softest ca steel for feel.
Golf Equipment	Putters	Blue Steel Putter	This prem putter ha mixed wit lightweigh aluminum

Jun 1, 2007

- 1 -

The lockColumn class property can be applied to multiple columns in order to freeze more than one column.

Conclusion

The 2 advanced CSS concepts CSS Parent Child relationships and CSS Expressions provide report developers great flexibility.

CSS Parent Child relationships let us take advantage of the default classes assigned to objects by Report Studio, without affecting existing reports. It also provides us a method for easily applying styles to objects, by merely using the parent class on a parent object, such as a block or page.

CSS expressions are very powerful and let us encapsulate logic in the style sheet itself, providing an easy method for reusing small bits of JavaScript across many reports.