2019-07-31

1.0
Instructions

Statistics Sweden
IT-department

Customisation instructions for PxWeb

Introduction

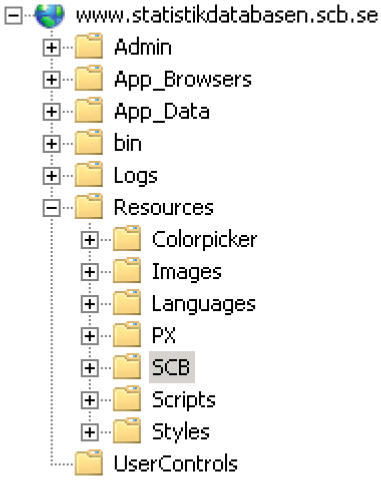
This document provides a short description of a possible approach to customise PxWeb to get the look and feel of an existing website. The instruction is written in conjunction with a project with aim to customise PxWeb to appear integrated with Statistics Sweden´s website www.scb.se, hence the steps and the examples below are described with those conditions in mind.

Basic knowledge about HTML, CSS and java script is needed to manage a successful customisation.

The elements adjusted are images, styles, texts and links, where images and styles affects the file structure of the website while texts and links is changed by code and text modifications in the existing PxWeb files.

Structure

To keep control of modifications of PxWeb and be able to keep changes in an easy way even if a new setup of the website is done, it is a good idea to set up a separate structure that contains the modifications. In this case this is made by adding customisation folder called “SCB” below the “Resources” folder in the PxWeb installation.



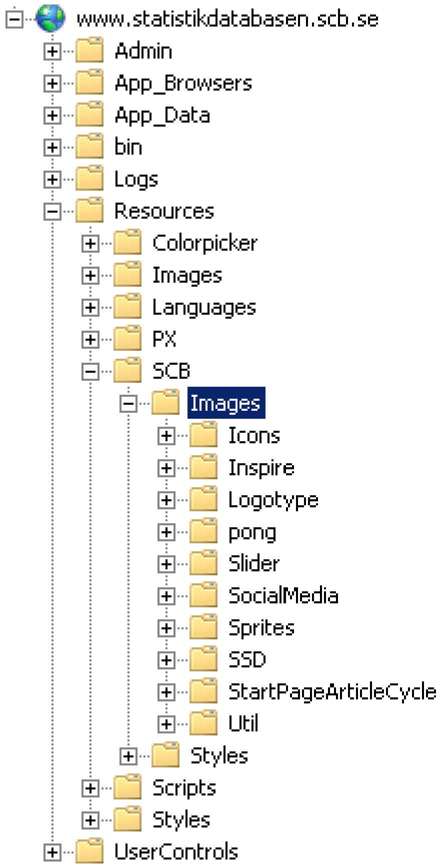
Images

In the customisation folder “SCB” folder structures for two sets of image types is created.

1. The default graphical elements for PxWeb (the content of “\PXWeb\Resources\Images” folder in the PxWeb installation). Some or all of those images may be replaced to match the new look of the website and to keep control of the changes , they are copied into a separate folder which in this example is called “SSD”. An example of PxWeb original graphical elements is

“mandatory.gif”: *, which is used to mark mandatory variables.

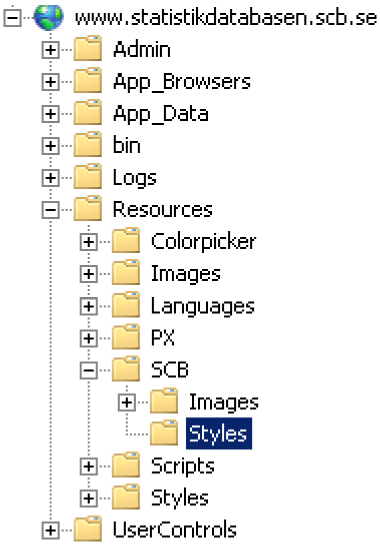
2. The graphical elements that is not originally a part of PxWeb, but will be used in the customised website as additions or referenced by imported style sheets. Since the goal is to get the PxWeb installation to appear like www.scb.se with a minimum of changes, the folder structure for images and its content is copied from that site and pasted into the “Images”-folder in the customisation folder “SCB”.



```
www.statistikdatabasen.scb.se
    Admin
    App_Browsers
    App_Data
    bin
    Logs
    Resources
        Colorpicker
        Images
        Languages
        PX
        SCB
            Images
                Icons
                Inspire
                Logotype
                pong
                Slider
                SocialMedia
                Sprites
                SSD
                StartPageArticleCycle
                Util
            Styles
        Scripts
        Styles
    UserControls
```

Styles

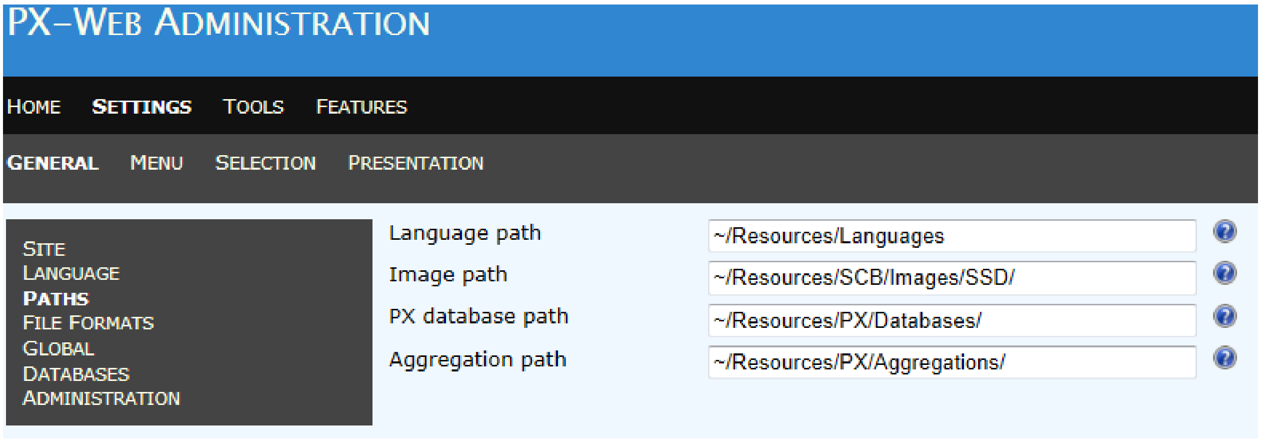
In customisation folder “SCB” a sub folder is created that will contain all style sheets for the site we want to mimic. The folder is called “Styles” and the style sheets from www.scb.se is copied into that.



```
www.statistikdatabasen.scb.se
    Admin
    App_Browsers
    App_Data
    bin
    Logs
    Resources
        Colorpicker
        Images
        Languages
        PX
        SCB
            Images
            Styles
        Scripts
        Styles
    UserControls
```

Images and paths

The set of default graphical elements for PxWeb that is going to be used in this customisation of PxWeb were in a previous step copied into a new location in the “customisation folder”. The image path setting in PxWeb administration therefore has to be changed to reflect the new path. That is made by changing the setting “Image path” in PxWeb administration interface.

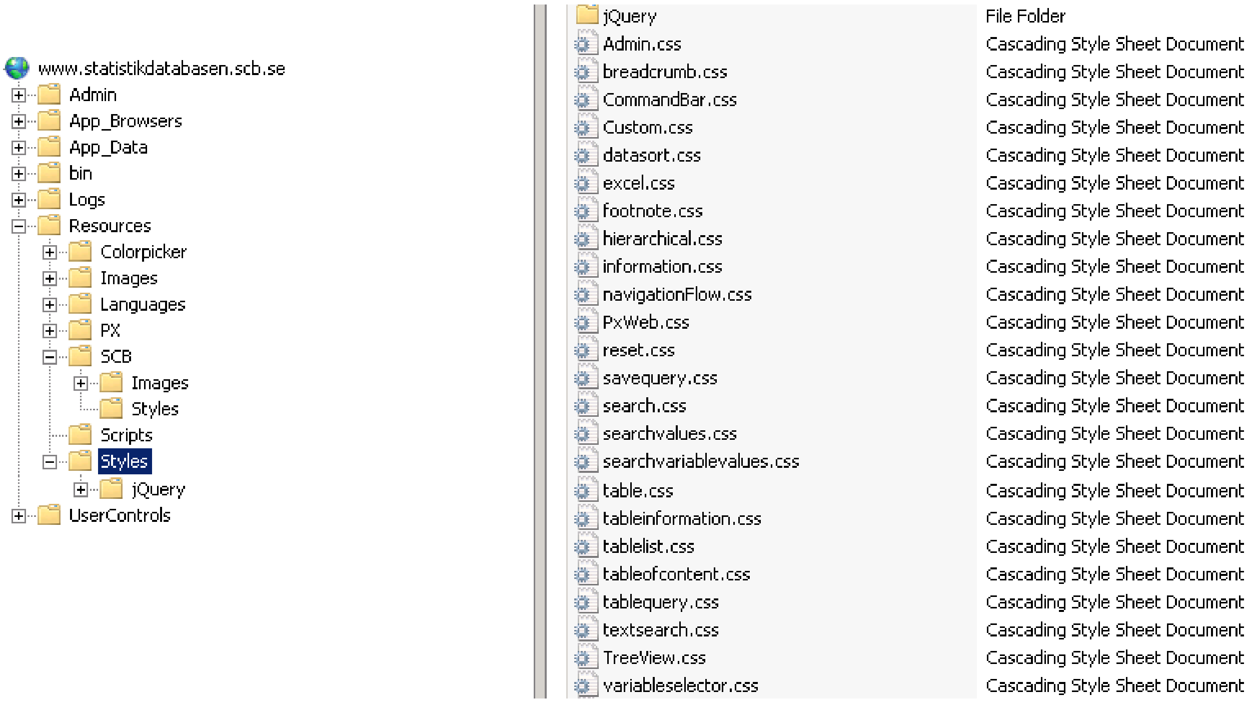


Language path	~/Resources/Languages
Image path	~/Resources/SCB/Images/SSD/
PX database path	~/Resources/PX/Databases/
Aggregation path	~/Resources/PX/Aggregations/

When an original graphical element for PxWeb are replaced in the customised version, the new image should be named as the image they replace. The image is then put in the designated folder below the customisation folder. In this example the replacement images are put in “Resources/SCB/Images/SSD”.

Styles

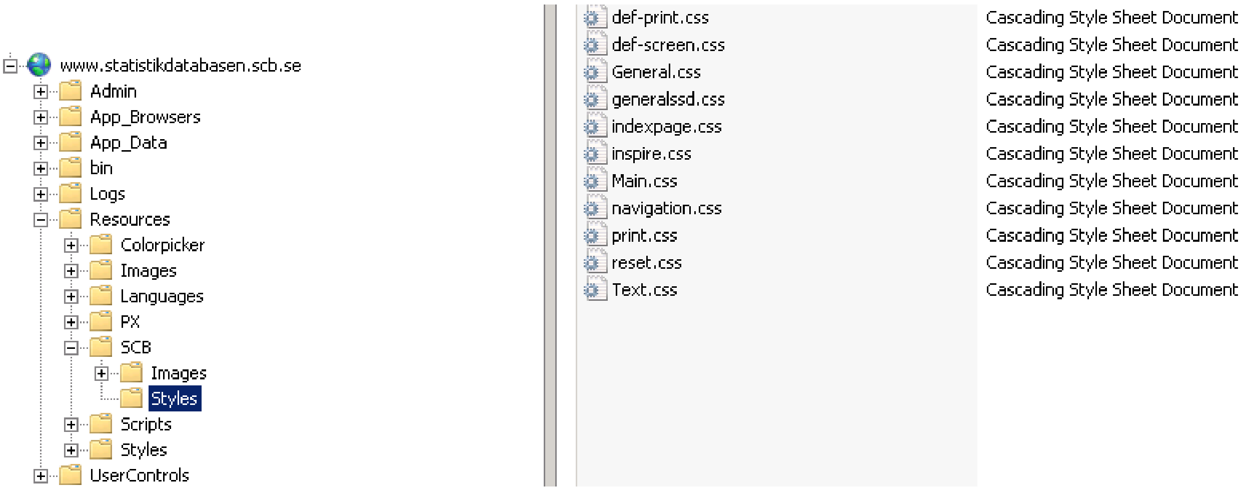
All styles used in the customised installation of PxWeb are defined in the PxWeb style sheets and in the style sheets added to the customisation folder.



```
www.statistikdatabasen.scb.se
+ Admin
+ App_Browsers
+ App_Data
+ bin
+ Logs
- Resources
    + Colorpicker
    + Images
    + Languages
    + PX
    - SCB
        + Images
          Styles
      Scripts
    - Styles
        + jQuery
+ UserControls
```

jQuery	File Folder
Admin.css	Cascading Style Sheet Document
breadcrumb.css	Cascading Style Sheet Document
CommandBar.css	Cascading Style Sheet Document
Custom.css	Cascading Style Sheet Document
datasort.css	Cascading Style Sheet Document
excel.css	Cascading Style Sheet Document
footnote.css	Cascading Style Sheet Document
hierarchical.css	Cascading Style Sheet Document
information.css	Cascading Style Sheet Document
navigationFlow.css	Cascading Style Sheet Document
PxWeb.css	Cascading Style Sheet Document
reset.css	Cascading Style Sheet Document
savequery.css	Cascading Style Sheet Document
search.css	Cascading Style Sheet Document
searchvalues.css	Cascading Style Sheet Document
searchvariablevalues.css	Cascading Style Sheet Document
table.css	Cascading Style Sheet Document
tableinformation.css	Cascading Style Sheet Document
tablelist.css	Cascading Style Sheet Document
tableofcontent.css	Cascading Style Sheet Document
tablequery.css	Cascading Style Sheet Document
textsearch.css	Cascading Style Sheet Document
TreeView.css	Cascading Style Sheet Document
variableselector.css	Cascading Style Sheet Document

Style sheets included in PxWeb is located under [Site]/Resources/Styles



```
- www.statistikdatabasen.scb.se
    + Admin
    + App_Browsers
    + App_Data
    + bin
    + Logs
    - Resources
        + Colorpicker
        + Images
        + Languages
        + PX
        - SCB
            + Images
              Styles
        + Scripts
        + Styles
    + UserControls
```

def-print.css	Cascading Style Sheet Document
def-screen.css	Cascading Style Sheet Document
General.css	Cascading Style Sheet Document
generalssd.css	Cascading Style Sheet Document
indexpage.css	Cascading Style Sheet Document
inspire.css	Cascading Style Sheet Document
Main.css	Cascading Style Sheet Document
navigation.css	Cascading Style Sheet Document
print.css	Cascading Style Sheet Document
reset.css	Cascading Style Sheet Document
Text.css	Cascading Style Sheet Document

Style sheets copied from another website, in this case www.scb.se, are added to the customisation folder.

Some of the styles defined in either the original PxWeb style sheets or in the imported style sheets probably has to be changed to get the exact right layout. All style additions, changes and extensions are made in the Custom.css file that is included in PxWeb.

If the style sheets imported from another website, in this example the style sheets from www.scb.se, contains image or file references, they have to be changed to reflect their new relative paths. With the file structure described in the example above the image paths in the style sheets will look like this: `url('../Images/[subfolder]/[name]')`.

For example the CSS properties for elements with id `TopSearch` and class `icon14` are defined in the style sheets for www.scb.se as

```
#TopSearch .icon14 {

    height: 25px;

    background: url('../Images/Sprites/scb_sprites_14px.png')
-1px -32px no-repeat;

}
```

The image reference in the customised PxWeb solution are then changed by the following addition to Custom.css:

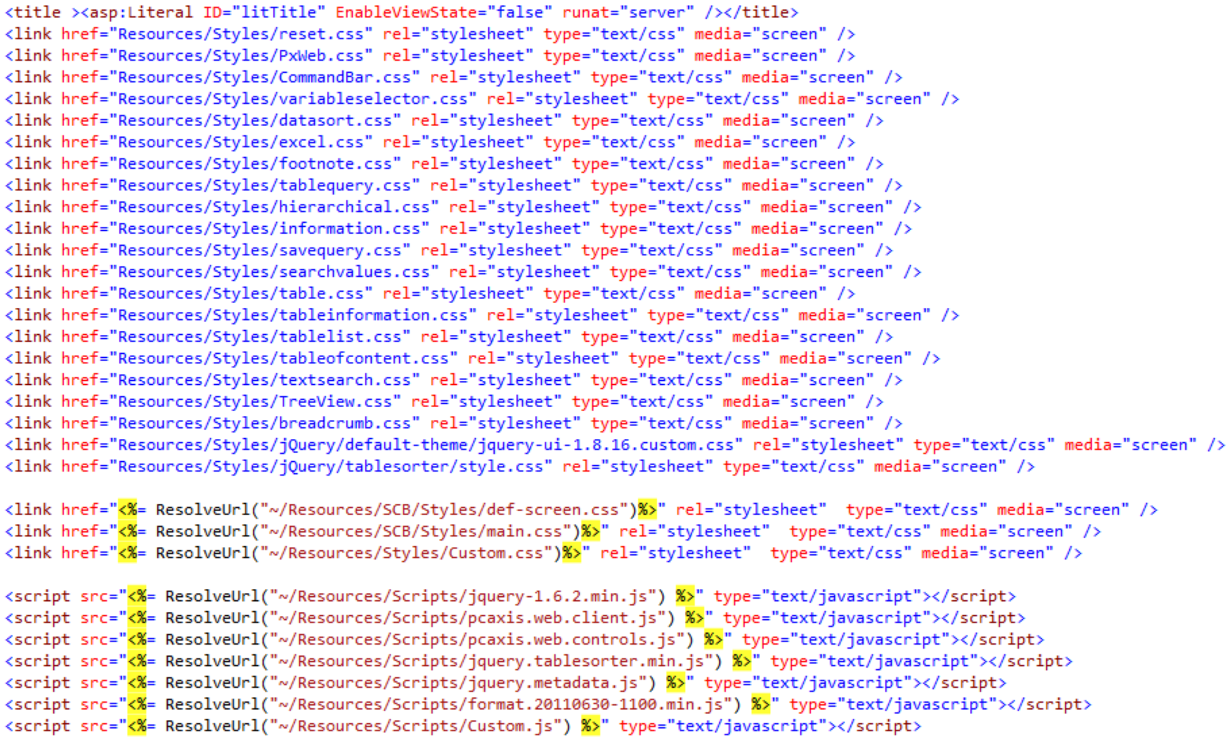
```
#TopSearch .icon14 {

    background:
url('../SCB/Images/Sprites/scb_sprites_14px.png') -1px -32px
no-repeat;

}
```

Code

Links to style sheets and script files

Links to style sheets and script files with content common for all parts of PxWeb are placed inside the `<head>` element in top of PxWeb.master. The style sheets and in the customisation folder should also be added there, just above Custom.css so that their contents overrides all other styles except the styles in Custom.css. Custom.css is as mentioned above the file where all modifications and extensions to the base styles are made. For the same reasons all script files (if any) in the customisation folder should be linked to just above the link to the file Custom.js.

Example: Below is a part of the code in PxWeb.master with the style sheets “def-screen.css” and “main.css” added from the customisation folder.



```
<title ><asp:Literal ID="litTitle" EnableViewState="false" runat="server" /></title>
<link href="Resources/Styles/reset.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/PxWeb.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/CommandBar.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/variableselector.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/datasort.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/excel.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/footnote.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/tablequery.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/hierarchical.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/information.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/savequery.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/searchvalues.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/table.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/tableinformation.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/tablelist.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/tableofcontent.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/textsearch.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/TreeView.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/breadcrumb.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/jQuery/default-theme/jquery-ui-1.8.16.custom.css" rel="stylesheet" type="text/css" media="screen" />
<link href="Resources/Styles/jQuery/tablesorter/style.css" rel="stylesheet" type="text/css" media="screen" />

<link href="<%= ResolveUrl("~/Resources/SCB/Styles/def-screen.css")%>" rel="stylesheet"  type="text/css" media="screen" />
<link href="<%= ResolveUrl("~/Resources/SCB/Styles/main.css")%>" rel="stylesheet"  type="text/css" media="screen" />
<link href="<%= ResolveUrl("~/Resources/Styles/Custom.css")%>" rel="stylesheet"  type="text/css" media="screen" />

<script src="<%= ResolveUrl("~/Resources/Scripts/jquery-1.6.2.min.js") %>" type="text/javascript"></script>
<script src="<%= ResolveUrl("~/Resources/Scripts/pcaxis.web.client.js") %>" type="text/javascript"></script>
<script src="<%= ResolveUrl("~/Resources/Scripts/pcaxis.web.controls.js") %>" type="text/javascript"></script>
<script src="<%= ResolveUrl("~/Resources/Scripts/jquery.tablesorter.min.js") %>" type="text/javascript"></script>
<script src="<%= ResolveUrl("~/Resources/Scripts/jquery.metadata.js") %>" type="text/javascript"></script>
<script src="<%= ResolveUrl("~/Resources/Scripts/format.20110630-1100.min.js") %>" type="text/javascript"></script>
<script src="<%= ResolveUrl("~/Resources/Scripts/Custom.js") %>" type="text/javascript"></script>
```

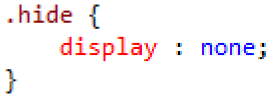
Hide original PxWeb content

When the PxWeb site is customised some of the original elements maybe has to be removed and/or be replaced by other elements. Since there are plenty of server elements (elements with the property `runat="server"`) it is often not possible to just remove that parts of the code.

One way to safely remove parts of the content in a PxWeb page is to hide it by applying a class on those elements.

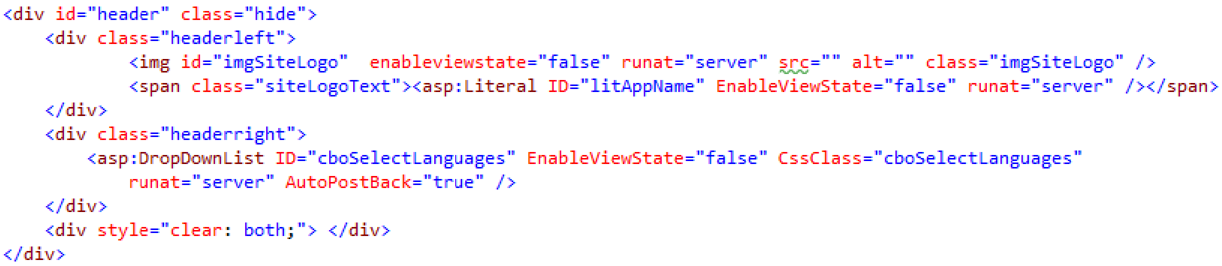
Example: The div “header” is changed so that it is not displayed or takes any space in the page.

The class “hide” is added to Custom.css:



```
.hide {
    display : none;
}
```

This class is used to hide the div “header” in PxWeb.master



```
<div id="header" class="hide">
    <div class="headerleft">
            <img id="imgSiteLogo"  enableviewstate="false" runat="server" src="" alt="" class="imgSiteLogo" />
            <span class="siteLogoText"><asp:Literal ID="litAppName" EnableViewState="false" runat="server" /></span>
    </div>
    <div class="headerright">
        <asp:DropDownList ID="cboSelectLanguages" EnableViewState="false" CssClass="cboSelectLanguages"
            runat="server" AutoPostBack="true" />
    </div>
    <div style="clear: both;"> </div>
</div>
```

The class “hide” is possible to use for all elements in the .aspx-pages, but if the element to hide is a server control it is more efficient to hide the element and its children using the property “Visible” since that affects the code output from the server.

Example: The visibility for the footnote control is changed so that neither the control or elements contained in it is rendered from the server.

```
<pxc:Footnote ID="SelectionFootnotes" runat="server"
Visible="false" />
```

Adding content

It is possible to add HTML and script content to all .aspx- and .ascx-files in PxWeb.

To insert a specific text or add a link, the recommended way is to use the language files.

Example: Add a link with text to a page in PxWeb

The following lines are added to pxlang.xml:

```
  <sentence name="MyCustomLinkUrl" value="http://www.scb.se"
/>

  <sentence name="MyCustomLinkText" value="Visit SCB" />
```

This code is added in the .aspx-file where the link should be displayed

```
<a href="<% =Master.GetLocalizedString("MyCustomLinkUrl")
%>">

<% =Master.GetLocalizedString("MyCustomLinkText") %></a>
```