class ReportViewer

An object for a report representation, navigation and export

CONSTRUCTORS

- ReportViewer(string selector);

PROPERTIES

- string ReportName;
- void SetServiceUrl(string serviceUrl);
- string GetServiceUrl();
- List<ParameterModel> Parameters;
- ThumbnailSizeModel GetThumbnailSize();
- void SetThumbnailSize(int width, int height);

METHODS

- void RenderDocument();
- void LoadReportParameters();
- int GetCurrentPage();
- void SetCurrentPage(int pageNumber);
- void NextPage();
- void PrevPage();
- void FirstPage();
- void LastPage();
- bool IsFirstPage();
- bool IsLastPage();
- void SetPageView(PageViewBase viewModel);
- void LoadThumbnails();
- void ExportToExcel();
- void ExportToPdf();
- void ExportToHtml();
- void ExportToRtf();
- void ExportToXps();
- void ExportTo(string format);
- void Print();
- void SetPageCacheSize(int pageCountInCache);
- void LoadDocumentMapInfo();
- void SetDocumentMapControl(string selector);
- void SetThumbnailsControl(string selector);
- void SetInvisibleThumbnailCacheSize(int invisibleCacheSize);
- void GoToBookmark(string bookmark);
- void SetHyperlinkClickHandler(HyperlinkCallback hyperlinkCallback);
- void SetDocumentLoadingElement(Element indicatorElement);
- void SetPageLoadingElement(Element indicatorElement);

EVENTS

- ErrorCallback ErrorEvent;
- ReportParametersLoadedCallback ReportParametersLoadedEvent;
- DocumentInfoLoadedCallback DocumentInfoLoadedEvent;
- ThumbnailsInfoLoadedCallback ThumbnailsInfoLoadedEvent;
- PageCallback CurrentPageChangedEvent;
- PageInfoLoadedCallback PageInfoLoadedEvent;
- DocumentMapInfoCallback DocumentMapLoadedEvent;
- DocumentMapElementClickCallback DocumentMapElementClickEvent;

- PageCallback ThumbnailClickEvent;

CALLBACK DELEGATES

- void ReportParametersLoadedCallback(List<ParameterModel> parameters);
- void DocumentInfoLoadedCallback(List<ElementSizeModel> pages);
- void PageInfoLoadedCallback(int pageNumber, string pageContent);
- void ThumbnailsInfoLoadedCallback(List<string> thumbnailsLink);
- void DocumentMapInfoCallback(string documentTree);
- void PageCallback(int pageNumber);
- void HyperlinkCallback(string hyperlink);
- void DocumentMapElementClickCallback(string bookmarkName);
- void ErrorCallback(ErrorModel errorModel);

Constructors

ReportViewer(string selector);

Description
Creates the PerpetuumSoft.Reporting.WebViewer.Client.ReportViewer class instance.

Parameters
selector: A selector used to get an element which will be used for a report display.

Return value
Instance of ReportViewer class.

Properties

string ReportName;

Description
Gets or sets a report name.

void SetServiceUrl(string serviceUrl);

Description
Sets a service Url

string GetServiceUrl();

Description
Gets a service Url

List<ParameterModel> Parameters;

Description
Gets or sets a parameters list

ThumbnailSizeModel GetThumbnailSize();

Description
Gets a thumbnail size

Return value
PerpetuumSoft.Reporting.WebViewer.Model.ThumbnailSizeModel class (see class reference)

void SetThumbnailSize(int width, int height);

Description
Sets a thumbnail size

Parameters
width: thumbnail width
height: thumbnail height

Methods

void RenderDocument();

Description
Performs query to server for a document rendering and loading of document data onto the client

void LoadReportParameters();

Description
Performs query to server for parameters loading

int GetCurrentPage();

Description
Gets a number of a current page

Return value
Number of a current page

void SetCurrentPage(int pageNumber);

Description
Sets a current page with number equal to "pageNumber" and displays the page

Parameters
pageNumber: Current page number

void NextPage();

Description
Navigates to the next page

void PrevPage();

Description
Navigates to the previous page

void FirstPage();

Description
Navigates to the first page

void LastPage();

Description
Navigates to the last page

bool IsFirstPage();

Description
Checks if the current page is the first one

Return value
True - if the current page is the first oneFalse - if the current page is not the first one

bool IsLastPage();

Description
Checks if the current page is the last one

Return value
True - if the current page is the last oneFalse - if the current page is not the last one

void SetPageView(PageViewBase viewModel);

Description
Sets view model for reports pages display

Parameters
viewModel: Pages view model

void LoadThumbnails();

Description
Performs query for data about pages thumbnails from a server

void ExportToExcel();

Description
Performs query for document export into the Excel formatOpens a browser window and redirects a stream from server to this window

void ExportToPdf();

Description
Performs query for document export into the Pdf formatOpens a browser window and redirects a stream from server to this window

void ExportToHtml();

Description
Performs query for document export into the HTML formatOpens a browser window and redirects a stream from server to this window

void ExportToRtf();

Description
Performs query for document export into the RTF formatOpens a browser window and redirects a stream from server to this window

void ExportToXps();

Description
Performs query for document export into the Xps formatOpens a browser window and redirects a stream from server to this window

void ExportTo(**string** format);

Description
Performs query for document export into specified by a user formatOpens a browser window and redirects a stream from server to this window

Parameters
format: A format which export should be performed into

void Print();

Description
Performs query for receiving of a document from server in the Html format for printing

void SetPageCacheSize(int pageCountInCache);

Description
Sets size of pages cache

Parameters
pageCountInCache: A number of pages which will be stored in cache

void LoadDocumentMapInfo();

Description
Performs query for loading of bookmarks tree

void SetDocumentMapControl(string selector);

Description
Sets web page element which will be used for display of bookmarks tree

Parameters
selector: An element which will be used for output of bookmarks tree

void SetThumbnailsControl(string selector);

Description
Sets web page element which will be used for display of pages thumbnails

Parameters
selector: An element which will be used for output of pages thumbnails

void SetInvisibleThumbnailCacheSize(int invisibleCacheSize);

Description
Sets the number of thumbnails, which will be loaded before and after the visible thumbnails

Parameters
invisibleCacheSize: Number of cached thumbnail

void GoToBookmark(string bookmark);

Description
Navigates to the element with "bookmark" name

Parameters
bookmark: Bookmark name

void SetHyperlinkClickHandler(HyperlinkCallback hyperlinkCallback);

Description
Sets a handler for hyperlink click event

Parameters
hyperlinkCallback: Hyperlink click event handler

void SetDocumentLoadingElement(Element indicatorElement);

Description
Sets web page element which will displayed during a document loading

Parameters
indicatorElement: An element which displays a document loading

void SetPageLoadingElement(Element indicatorElement);

Description
Sets web page element which will displayed during a document page loading

Parameters
indicatorElement: An element which displays a page content loading

Events

ErrorCallback ErrorEvent;

Description
Event raise when an error occurs

ReportParametersLoadedCallback ReportParametersLoadedEvent;

Description
Occurs when report parameters were received from a server

DocumentInfoLoadedCallback DocumentInfoLoadedEvent;

Description
Occurs when data about a document were received form a server

ThumbnailsInfoLoadedCallback ThumbnailsInfoLoadedEvent;

Description
Occurs when data about thumbnails were received from a server

PageCallback CurrentPageChangedEvent;

Description
Occurs when current page was changed

PageInfoLoadedCallback PageInfoLoadedEvent;

Description
Occurs when data about a page were received from a server

DocumentMapInfoCallback DocumentMapLoadedEvent;

Description
Occurs when data about bookmarks tree were received from a server

DocumentMapElementClickCallback DocumentMapElementClickEvent;

Description
Occurs when the element in bookmarks tree was clicked

PageCallback ThumbnailClickEvent;

Description
Occurs when a thumbnail was clicked

Callback Delegates

void ReportParametersLoadedCallback(List<ParameterModel> parameters);

Description
The callback function which is invoked when report parameters were received from a server

Parameters
`parameters:` Report parameters

void DocumentInfoLoadedCallback(List<ElementSizeModel> pages);

Description
The callback function which is invoked when data about a document were received from a server

Parameters
`pages:` Report pages list.

void PageInfoLoadedCallback(int pageNumber, string pageContent);

Description
The callback function which is invoked when data about a page were received from a server

Parameters
`pageNumber:` Page number.
`pageContent:` Page content.

void ThumbnailsInfoLoadedCallback(List<string> thumbnailsLink);

Description
The callback function which is invoked when data about thumbnails were received from a server

Parameters
`thumbnailsLink:` A list of link for thumbnails loading from a server

void DocumentMapInfoCallback(string documentTree);

Description
The callback function which is invoked when data about bookmarks tree were received from a server

Parameters
`documentTree:` Bookmarks tree

void PageCallback(int pageNumber);

Description
The callback function which is used to inform a user about change of a page number

Parameters
`pageNumber:` Page number

void HyperlinkCallback(string hyperlink);

Description
The callback function which is invoked when a user clicks a hyperlink in a report

Parameters
`hyperlink:` Hyperlink Url

void DocumentMapElementClickCallback(string bookmarkName);

Description

The callback function which is invoked when a user clicks an element in bookmarks tree

Parameters

`bookmarkName:` A name of bookmark to which a user should be navigated

void ErrorCallback(ErrorModel errorModel);

Description

The callback function which is invoked when an error occurs

Parameters

`errorModel:` Data about an error