

## NewsAdDesigner Object

#	<i>Method</i>	<i>Description</i>
1	<b>void Initialize</b> ( int startPage, int pageCount, int pageRowCount, int pageColumnsCount, int blockWidth, int blockHeight, int newspaperColumnsCount)	This method which should be called first to initialize control with all needed information. If you try to call it twice you'll get exception. If you call other method without calling initialize previously you'll get an exception too. Parameters and type are self explanatory. Method will throw exception if parameters has not allowed value.
2	<b>bool HidePage</b> ( int pageNumber)	Hide page from designer. Page number is number which you see on the screen. To hide several pages call this method several times. Result show if page was hidden successfully. Last page can not be hidden.
3	<b>void ShowAllPages</b> ()	Get back to screen all pages hidden by HidePage method.
4	<b>void AddItem</b> ( String id, String caption, int pageNumber, int startPageRowNumber, int startPageColumnNumber, int pageRowCount, int pageColumnsCount, int color, ADODB.Stream image, bool freed, bool allowResize)	Adds item to page in designer. Id suppose to be Guid converted to string. Color is RGB coded color compatible with Delphi color dialog return value. Image can be null. All parameters describe position and look of the advertisement on the page. This method will throw exception if it's not possible to add item on specified position.
5	<b>void AddNewItem</b> ( String id, String caption, int pageNumber, int pageRowCount, int pageColumnsCount, int color, ADODB.Stream image, bool allowResize)	This method should be used to add new items to exact page specified by pageNumber. All parameters are the same as in AddItem method but two of them are "missing" startPageRowNumber and startPageColumnNumber. They would be calculated by control to fit advertisement. This method will throw exception if it's not possible to place item of specified dimensions on page. Method focuses on new Item.
6	<b>void ShowPics</b> ( bool captionVisibility)	Shows images for all items with images (image!=null). If captionVisibility is false the captions will be hidden for the items with images.
7	<b>void HidePics</b> ()	Hides images for all items.
8	<b>int GetItemsCount</b> ()	Return amount of the advertisements in this edition of the newspaper or magazine on all pages (visible and hidden).
9	<b>AdItem GetItem</b> ( int index)	Return information about exact advertisement. Index should be in range from 0 to GetItemsCount() - 1. Return type contains more detail information and is

#	<i>Method</i>	<i>Description</i>
		described below.
10	<b>AdItem GetSelectedItem ()</b>	This method get information about advertisement block which is currently selected in UI. If there is no such block then NULL is returned.
11	<b>void RemoveSelectedItem ()</b>	The method removes selected item from UI . If there is no selected item then method will not do anything. If you want to get information about item which you are deleting call GetSelectedItem before it.
12	<b>void Close ()</b>	Compulsory method to call before form with ActiveX closed
13	<b>void FreezeSelectedItem ()</b>	Freeze position and size of currently selected advertisement.
14	<b>void UnFreezeSelectedItem ()</b>	Unfreeze position and size manipulation of currently selected item.
15	<b>String GetFontFamily ()</b>	Return current font family for advertisement captions.
16	<b>float GetFontSize ()</b>	Return current font size for advertisement captions.
17	<b>int GetFontStyle ()</b>	Return current font style coded in integer for advertisement captions.
18	<b>void ChooseFont ()</b>	Show Font choose dialog and update font for all advertisements.
19	<b>void SetFontParameters (</b> <b>String family,</b> <b>float size,</b> <b>int style)</b>	This method allow to recreate font settings from external source.
20	<b>int GetCurrentPageNumber ()</b>	Return current page number which is shown in UI. Can be used to add user notes on each page.
21	<b>String GetPageUserNotes (</b> <b>int pageNumber)</b>	Return user notes below the page for specified page.
22	<b>void SetPageUserNotes (</b> <b>int pageNumber,</b> <b>String userNotes)</b>	Sets user notes below the page for specified page.
23	<b>void SetCurrentPageNumber (</b> <b>int pageNumber)</b>	Sets current page number.
24	<b>void SetSelectedItem (</b> <b>String id)</b>	Select item on UI by id and scroll view to show it.
25	<b>void Clear ()</b>	Clear all information about control and allow new initialization.
26	<b>int GetNewspaperColumnsCount ()</b>	Return amount of columns of the pages selected on UI.
27	<b>void SetNewspaperColumnsCount (</b> <b>int columnsCount)</b>	Set amount of columns of the pages selected on UI.

#	<i>Method</i>	<i>Description</i>
28	<b>bool</b> <b>IsPageVisible</b> ( <b>int</b> pageNumber)	Return true if the page is visible.
29	<b>void</b> <b>tsSetConnector</b> ( <b>Connector</b> connector)	Set Terrasoft Connector Object for TS PRESS license.
30	<b>void</b> <b>HidePages</b> ( <b>String</b> PagesNumber)	Hide pages from designer. Page number is number which you see on the screen. To hide several pages use string with numbers separated by ; Last page can not be hidden. For an example: "4;5;7"

### AdItem Object

#	<i>Method</i>	<i>Description</i>
1	<b>String</b> <b>GetId</b> ()	Retrieve GUID of the item as string
2	<b>String</b> <b>GetCaption</b> ()	Return caption of the block.
3	<b>int</b> <b>GetColor</b> ()	This method return color in RGB format
4	<b>int</b> <b>GetPageStartRowNumber</b> ()	Start row index for advertisement. Starts from 1 to count of rows on the page.
5	<b>int</b> <b>GetPageStartColumnNumber</b> ()	Start column index for advertisement. Starts from 1 to count of columns in the page.
6	<b>int</b> <b>GetPageRowsCount</b> ()	Return amount of rows used by advertisement. Could be 1 or more.
7	<b>int</b> <b>GetPageColumnsCount</b> ()	Return amount of columns used by advertisement. Could be 1 or more.
8	<b>int</b> <b>GetPageNumber</b> ()	Return page number as it is displayed in UI.
9	<b>bool</b> <b>IsFreezed</b> ()	Can be used to check if current advertisement is friezed in UI.