

Tutorial
Create Custom
CSS Skin



Table of contents

Introduction	3
Planning the application	5
Create elements needed for the customization	7
Create the new skin	9
Other Resources	13
Copyright	14

Introduction

In this tutorial you will learn how to adapt the MX Kollection 3 elements to better fit your existing design and style. All this is done by using the fact that most of the HTML elements inserted by the MX Kollection 3 wizards and server behaviors use CSS files.

Besides choosing from one of the skins provided, you can also create a skin that integrates with your existing site smoothly, in a simple manner.

The application used in this tutorial is the web-site of a small company, that uses for the design an existing template, and for the dynamic elements MX Kollection3 Bundle.

The modules needed to complete this tutorial are:

- The Transaction Engine
- NeXTensio

For the basic operations requested in order to create a new skin, basic knowledge of graphic editing is required, while for more advanced changes, you will need some knowledge of CSS.

In this tutorial, you will customize an application that has been already created, that contains three pages:

- A NeXTensio List used to display the company products
- A NeXTensio Form used in conjuncture with the list, to allow editing of elements
- A Contact Form, that allows users send their messages through e-mail to the company.

The application uses a template design, and all that has to be changed are the list, form and contact form appearance, through the CSS files.

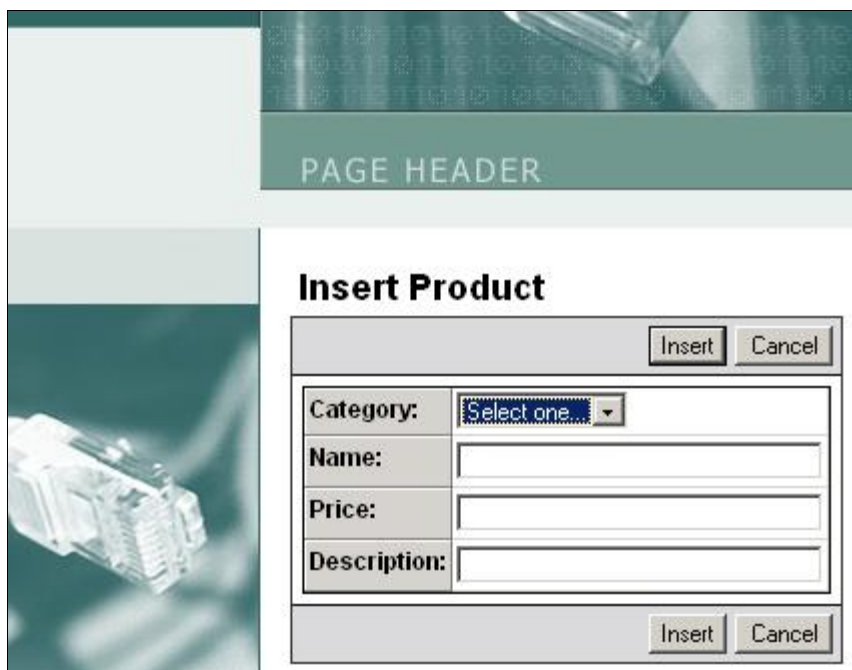
The tutorial wil not present how to create the NeXTensio list or form, or the contact form. These elements can be found in one of the other tutorials (for the NeXTensio list and form, see the Job site tutorial > Improve the Job Site > Receive job ads by e-mail ; to learn how to create the contact form that sends the e-mail message, see the How to Send Form Contents by E-mail (without using a database) tutorial.).

The pages look as follows in the browser (no customization yet):

- The NeXTensio product list:

No.	Idctg:	Name:	Price:	Description:	
<input type="checkbox"/>	1	Cameras	Logitech advanced ca...	150	The advanced Logitec... edit delete
<input type="checkbox"/>	2	Cameras	Logitech camera	100	This streamlined Log... edit delete
<input type="checkbox"/>	3	Cameras	Logitech professiona...	200	The Professional Log... edit delete

- The NeXTensio form for one element



Insert Product

Category:

Name:

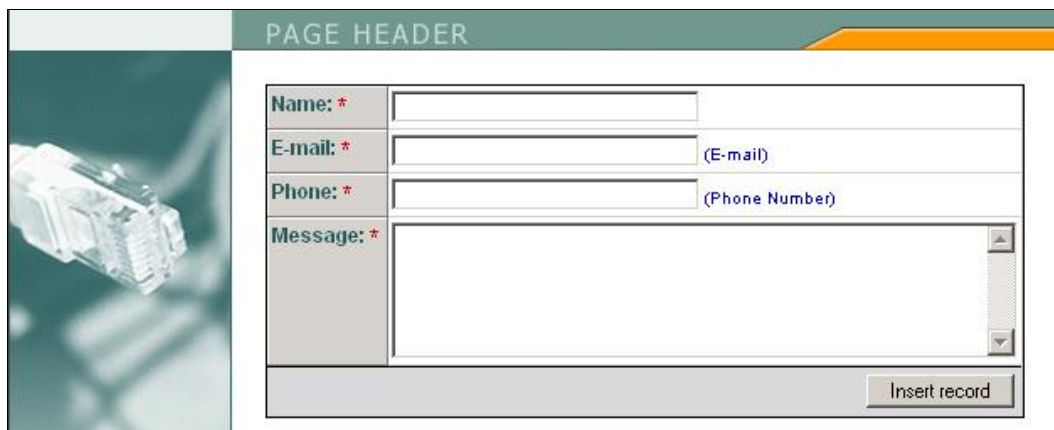
Price:

Description:

Insert Cancel

Insert Cancel

- The contact form



NAME

Name: *

E-mail: * (E-mail)

Phone: * (Phone Number)

Message: *

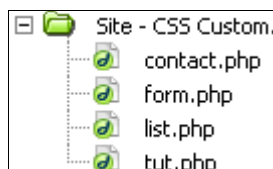
Insert record

Planning the application

This first section of this tutorial helps you create all the files and database tables needed for the application.

Before you start building this application, make sure you have a correctly configured **Dreamweaver** site, and a working database connection. For more instructions regarding such actions, consult the *Getting started* help file, which can be found in **Help -> InterAKT -> Getting Started**.

Through the tutorial, you will have to create several files in your site's root. The file structure will look as in the example below, and you can create it easily by unpacking the *zip* file corresponding to your server model from `\tutorials\Custom CSS Skin\` in your site root:



Besides these files that are the actual pages that will be viewed from the browser, some more files and folders will appear in your site structure:

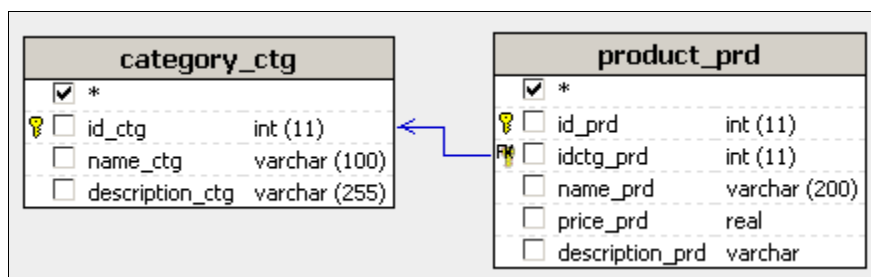
- Some images, that come with the site template
- The **Dreamweaver** created Connection folder
- The includes folder, created when an extension is used on a page.

Included in the zip file you can also find the template that has been used in the tutorial, as well as the new skin's images and **CSS** files.

After having dealt with the file and folder structure, it is time to create the database structure that will be used throughout the site's pages. The database is simple in nature, as it is not the scope of this tutorial, and contains only two tables:

- *product_prd* - the table that stores the company's product list
- *category_ctg* - this table stores the different product categories.

The table and column structure is as shown in the following image:

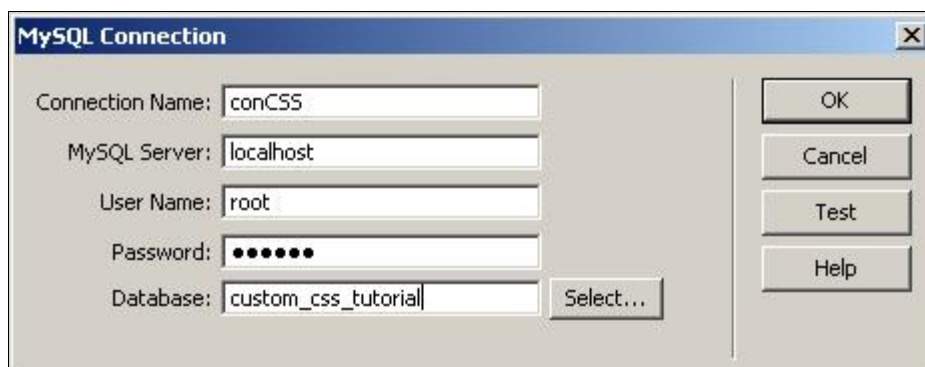


Note: The database diagram in the image above was built with **MX Query Builder** (also referred as **QuB**) to better illustrate the database structure. You do not need to build it in order to complete this tutorial.

You can find the scripts needed to create an identical table structure inside the downloaded package, in the `\tutorials\Custom CSS Skin\db\` folder, as an *sql* or *mdb* file, depending on the database server you intend to use. Import them in your database server management software (e.g. **PHPMyAdmin**, **Microsoft Access** etc).

The scripts already contains some data, in order for you to easily view the results.

Next, open the main *index* page and create a new connection named connAds and configure it to connect to your newly created database.



Create elements needed for the customization

In this section of the tutorial, you will identify all the elements that require customization, by analyzing the pages that contain **MX Kollection 3** elements (the list, form and contact page), and then create new elements that will match the template design.

The basic template page uses a green theme, seen below:



The **NeXTensio** list and form, as well as the contact form will replace the section containing text in the image above. However, none of the existing skins match the selected design. None of the skins use green for buttons or backgrounds.

The finished application must display links as buttons, and the button styles must match the general design: to use some green on the background.

The easiest way to start when creating a new skin is to start from an existing one, and modify its elements to integrate it into the design, as shown in the [User Manual](#). The skin used as a basis for this tutorial is the Aqua skin.

The first step to take before starting any work is to identify the elements you need changed. To accomplish this, open the pages in the browser, and decide which elements must be changed for each of them:

1. For the **NeXTensio List**: all buttons on the page: the buttons that replace the edit and delete links, the top and bottom rows buttons for adding a record, editing and deleting multiple records, and the navigation buttons.
2. For the **NeXTensio Form**: all available buttons.
3. For the Contact Form: only the submit button for the form must be changed, to match the rest of the site.

The main focus will be on the buttons. These must be created as images with a photo editing software (**Fireworks**, **Adobe Photoshop**, etc), based on the site color scheme.

To find out how many and what size the new images must be, open the `includes/skins/aqua/images` folder. In this folder, all images used for the skin are stored: buttons, backgrounds, widget and error elements. The files that pose the most interest at this point are:

- `button_big.gif` - it is displayed as the buttons on the top and bottom row
- `button_small.gif` - which is not used in the current skin

- `button_smallest.gif` - used for all other buttons on the list, form and contact form.

In order to see exactly which files are used by the skin and in which manner, simply note down the styles used by the different elements as shown by **Dreamweaver**, and then open the `nxt.css` file - for the **NeXTensio** list and form, and the `tng.css` file - for the contact form. Locate the style classes noted earlier, and check the files that are used.

For example, the Add new link on the **NeXTensio** bottom and top row uses the `KT_bottombuttons` class. This class is defined in the `nxt.css` file, as follows:

```
.KT_tng .KT_tngform .KT_topbuttons input, .KT_tng .KT_tngform
.KT_bottombuttons input{
font-family: 'Lucida Grande', Geneva, Arial, Verdana, sans-serif;
font-size: 12px;
width: 110px;
height: 20px;
border: 0px;
background-image: url(images/button_big.gif);
background-position: 0% 50%;
}
```

As you can notice from the code above, when viewed as buttons, the image used is `images/button_big.gif`. The path is relative to the **CSS** file, therefore the actual folder is the one specified earlier. Another important element supplied by the **CSS** code is the size of the image to use. In this case, the image to use for the button must have a width of 110 pixels, and a height of 20 pixels.

Do the same for the other buttons, to discover the file names and sizes

The edit and delete links shown next to each record in the **NeXTensio** list use the `button_smallest` file, with the following size: width: 50 pixels and height 20 pixels.

Once the required elements for the customization are identified, you can start creating them. If using the template provided with the tutorial, the button replacement images are also provided in the zip package.

Once the new images for the buttons are created, you can start creating the actual skin, and applying it onto your site.

Create the new skin

In this section of the tutorial you will actually create the new skin that uses your custom images for buttons, and some other changes.

To start creating the skin, follow the next steps:

1. Duplicate the **aqua** folder in the **includes/skins** folder in your site root. Rename the duplicate with the same name as the desired skin. Since the main color of this template is green, let's name the skin **green**.
2. Copy the *button_big.gif* and *button_smallest.gif* files from the zip package (or the folder where you created them) into the *includes/skins/green/images* folder. As mentioned in the previous page, the images folder stores all graphical elements used by the **CSS** files. When asked if you want to overwrite the existing files, click Yes. Also, copy the *preview.gif* file over the existing one in the *includes/skins/green* folder. This will allow you to identify the skin when selecting it in **Dreamweaver**.
3. Through this action, you have replaced the original buttons with the custom created ones. The buttons are already changed.
4. In order to see what changes are made, you must apply the skin. To do so, open the **InterAKT Control Panel > CSS Skin** and from the drop-down menu select the **green** entry.



5. Now upload the *includes* folder to your server. Notice how all buttons have already changed to the new images, without writing or changing a line of **CSS** code - for all pages: list, form and contact form:

Update Product

Insert as new	Update	Delete	Cancel
---------------	--------	--------	--------

Category:	<input type="text" value="Keyboards"/>
Name:	<input type="text" value="Microsoft natural keyboard"/>
Price:	<input type="text" value="53"/>
Description:	<input type="text" value="The most ergonomic keyboard"/>

Insert as new	Update	Delete	Cancel
---------------	--------	--------	--------

6. If you want to use buttons of a different size however, you must edit the *nxt.css* and *tng.css* file, the section corresponding to each button type (big and smallest) and set the new width and height.

7. The next step into creating a new skin is to replace the color used when highlighting list rows. At this point, it still uses the dashes if gray from aqua.

8. For the highlight colors, there are two options to set in the **CSS** file: the color to use when moving the mouse over each row, and the one to use when the row is checked (after being clicked once). To alter them, you must change the following **CSS** sections:

```
/* selected row (checkbox is clicked) */
.KT_tnglist tr.KT_highlight {
background-color: #E2E2E2;
}
.KT_tnglist tbody tr.KT_over {
background-color: #E2E2E2 ! important;
}
```

9. The first color to set is when the row is selected. Let's change the shade of gray to a shade of green: #BACDA0. When rows are selected, the list will look like the following image:

<input type="checkbox"/>	No.	Category:
<input type="checkbox"/>	1	Cameras
<input type="checkbox"/>	2	Cameras
<input type="checkbox"/>	3	Cameras
<input type="checkbox"/>	4	Cameras
<input checked="" type="checkbox"/>	5	Cameras
<input checked="" type="checkbox"/>	6	Keyboards
<input checked="" type="checkbox"/>	7	Keyboards
<input checked="" type="checkbox"/>	8	Keyboards
<input checked="" type="checkbox"/>	9	Mice
<input type="checkbox"/>	10	Mice

10. The second color to change is the color the row takes when the mouse is over it. Change it to #84D082. Save and upload the *nxt.css* file, and move the mouse over rows to see the result:

<input type="checkbox"/>	No.	Category:
<input type="checkbox"/>	1	Cameras
<input type="checkbox"/>	2	Cameras
<input checked="" type="checkbox"/>	3	Cameras
<input type="checkbox"/>	4	Cameras
<input type="checkbox"/>	5	Cameras
<input type="checkbox"/>	6	Keyboards
<input type="checkbox"/>	7	Keyboards
<input type="checkbox"/>	8	Keyboards
<input type="checkbox"/>	9	Mice
<input type="checkbox"/>	10	Mice

11. The last element to change regarding the **NeXTensio** list is the way even rows are colored. The **CSS** code handling this is the following:

```
.KT_tnglist tr.KT_even {
background-color: #F2F2F2;
}
```

12. To display even rows in a different shade, replace the existing color (#F2F2F2) to #D5E6C1. Upload the *nxt.css* file and refresh the page. Results are immediate.

<input type="checkbox"/>	No.	Category:
<input type="checkbox"/>	1	Cameras
<input type="checkbox"/>	2	Cameras
<input type="checkbox"/>	3	Cameras
<input type="checkbox"/>	4	Cameras
<input type="checkbox"/>	5	Cameras
<input type="checkbox"/>	6	Keyboards
<input type="checkbox"/>	7	Keyboards
<input type="checkbox"/>	8	Keyboards
<input type="checkbox"/>	9	Mice
<input type="checkbox"/>	10	Mice

13. When adding new links to a list or form, you might want to give them the same look as the rest. In order to benefit from the automatic conversion to a button, and the usage of skins for all manually added links, you need to set their style property to use the **KT_link** class.

These are the first steps in creating a skin to integrate **MX Kollection** elements into your site. Just a few image editing operations, some **CSS** knowledge and you're off.

Other Resources

Other Dreamweaver Extensions from InterAKT

- KTML
- MX Kart
- MX Site Search
- MX RSS Reader-Writer
- MX Dynamic Table Sorter
- MX Coder Pack
- MX Dynamic Charts
- MX CSS Dynamic Menus

Contact Us

- **Address:**
1-11 Economu Cezarescu ST, AYASH Center, 1st floor
Sector 6, ZIP 060754, Bucharest, Romania
- **Web:** <http://www.interaktonline.com/>
- **E-mail:** contact@interaktonline.com
- **Phone:** +4031 401.68.19 or +4021 312.51.91
- **Fax:** +4021 312.53.12

Copyright

Windows is a trademark of Microsoft, Inc.

Dreamweaver MX is a trademark of Macromedia, Inc.

Redhat is a trademark of Redhat, Inc.

Copyrights and Trademarks

Copyright 2000-2005 by InterAKT Online.

All Rights Reserved. This tutorial is subject to copyright protection.

PHAkt, ImpAKT, NeXTensio, MX Query Builder, Transaction Engine, MX Includes, KHTML, MX Kommerce, MX Kollection, MX Widgets, MX Looper, MX Dynamic Charts, MX CSS Dynamic Menus, MX Tree Menu, MX Form Validation, MX File Upload, MX Send E-mail, MX User Login, MX CSV Import-Export, MX Kart, MX Site Search, MX Dynamic Table Sorter, MX RSS Reader-Writer, MX Coder Pack, MX Dynamic Charts are trademarks of InterAKT Online.

All other trademarks are acknowledged as the property of their respective owners.

This document and the product to which it pertains are distributed under licenses restricting their use, copying, distribution, and decompilation.

No part of this document or of the associated product may be reproduced in any form by any means without prior written authorization of InterAKT Online, except when presenting only a summary of the tutorial and then linking to the InterAKT website.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Send comments and suggestions to products@interaktonline.com



InterAKT Online

Web: <http://www.interaktonline.com/>

E-mail: contact@interaktonline.com

Address: 1-11 Economu Cezarescu ST, AYASH Center, 1st floor, Sector 6, ZIP 060754, Bucharest, Romania

Phone: +4021 312.51.91

Fax: +4021 312.53.12