



# **Form Master Responsive**

## User Manual

Created: Saturday, April 30, 2016

Copyright © Code 5 Systems, LLC.. All Rights Reserved.



# Form Master Responsive

copyright © Code 5 Systems, LLC.. All rights reserved.  
<http://www.code5systems.com>

The information contained in this document is subject to change without notice.  
This document contains proprietary information which is protected by copyright.  
All rights are reserved. No part of this document may be photocopied, reproduced,  
or translated to another language without the prior written consent of Code 5 Systems, LLC..



# Table of Contents

<b>Chapter 1: Welcome</b>	1
Support	1
<b>Chapter 2: What's New</b>	3
<b>Chapter 3: Installation</b>	4
System Requirements	4
Installing Form Master Responsive	4
<b>Chapter 4: Getting Started</b>	6
About Form Master Responsive	6
Form Submit Workflow	6
Form Elements	7
Menu Items	8
<b>Chapter 5: Form Settings</b>	10
Basic Settings	10
Validation Options	12
Form Wizard Options	13
Responsive Web Options	15
Email Settings	15
Auto Responder Settings	17
Email Template & Format Settings	18
Redirect After Submit Options	19
Web Service Options	19
<b>Chapter 6: Basic Functions</b>	21
Form Layout	21
Tab Index	23
Localization	23
File Management	24
<b>Chapter 7: Working With Fields</b>	26
Field Settings	26
Input control	26
Auto Fill	29
List Options	31
Other Options	32
Adding Fields	34
Textbox	34
Checkbox	34
Dropdown List	35
Listbox	35
Checkbox List	36

Radiobutton List .....	36
Label .....	37
File Upload .....	37
Submit Button .....	38
Hidden Field .....	38
HTML Editor .....	39
Calculated Fields .....	40
<b>Chapter 8: Cascading Dropdowns .....</b>	<b>41</b>
Static Data .....	41
Dynamic Data .....	43
<b>Chapter 9: Multi-Page Forms .....</b>	<b>47</b>
Multi-Page Form Options .....	47
Working with Multi-Page Forms .....	48
<b>Chapter 10: Validation .....</b>	<b>51</b>
Required Field Validator .....	51
Regular Expression Validator .....	52
Compare Validator .....	52
Checkbox List Validator .....	53
Validation Summary .....	53
<b>Chapter 11: Data Manager .....</b>	<b>55</b>
The Connection String .....	55
Map Existing Table .....	56
Table From Form .....	57
Form From Table .....	58
The Mapped Form .....	59
Edit Mappings .....	61
Multi Table Mapping .....	62
External Table Mask .....	63
<b>Chapter 12: Formatting .....</b>	<b>67</b>
Form Formatting .....	67
Email Formatting .....	68
XSL Templates .....	69
Tokenized Templates .....	69
Custom Template .....	69
<b>Chapter 13: Reporting .....</b>	<b>71</b>
<b>Chapter 14: Messaging .....</b>	<b>72</b>
Form Heading .....	72
Form Thank You .....	73
Auto Responder .....	74
<b>Chapter 15: How to Videos .....</b>	<b>76</b>
Getting Started .....	76

Fields .....	77
Adding a Textbox .....	77
Adding a Checkbox .....	78
Adding a Listbox .....	78
Adding a Checkbox List .....	79
Adding a Radiobutton List .....	80
Adding a Label .....	80
Adding a File Upload .....	81
Adding a Submit Button .....	82
Adding a Hidden Field .....	82
Adding an HTML Editor .....	83
Cascading Dropdowns .....	84
Dynamic Data .....	84
Multi-Page Forms .....	84
Updatable Forms .....	85
<b>Chapter 16: Advanced Functions</b> .....	<b>87</b>
Silent Post .....	87
Field Rules .....	87
User Manager .....	89
Updatable Forms .....	90
Transaction Option .....	90
<b>Chapter 17: Index</b> .....	<b>92</b>



# Form Master Responsive Online Help

Welcome to the Form Master Responsive online help system. Browse through the help pages by clicking on the icons below or selecting pages in the table of contents to the left. To quickly find specific product information, enter search criteria in the search box above and click the search button.

**Note:** Some of the videos and screen shots show older versions of Form Master Responsive but are still valid.



What's New



Installation



Getting Started



Form Settings



Basic Functions



Working With Fields



Cascading  
Dropdowns



Multi-Page Forms



Validation



Data Manager



Formatting



Reporting



Messaging



How to Videos



Advanced Functions

## Ask Us



If you're unable to find what you're looking for in this help system, try these alternative resources:

- [Our Website](#)
- [Knowledgebase](#)

or contact our support team:  
Email:

[support@code5systems.com](mailto:support@code5systems.com)

### Most popular pages

- [Welcome](#)
- [Form Layout](#)
- [Getting Started](#)
- [Advanced Functions](#)
- [Updatable Forms](#)
- [Input control](#)
- [Multi-Page Form Options](#)
- [Support](#)
- [System Requirements](#)
- [Basic Functions](#)

[Home](#) > [Welcome](#) > [Support](#)

## Support

## Get help from Customer Support or other resources

### Ask Someone

---

Post a support request or offer a suggestion at our online [Help Desk](#).

### Knowledgebase

---

Search the [Knowledgebase](#) for common bugs, tips & tricks, and compatibility issues with other software.

### Customer Support

---

Contact our customer support department by email at [Email Support](#)

See also



If you're unable to find what you're looking for in this help system, try these alternative resources:

- [Help Desk](#)
- [Knowledgebase](#)

Contact our support department:  
Email: [Email Support](#)

---

[Home](#) > [What's New](#)

## What's New

Form Master Responsive introduces several new features to make creating forms faster and easier.

- Responsive (table-less) Web Design Option
- Calculated Fields
- Placeholders for Text Fields
- Option to use DotNetNuke styls error message format
- Option to use Form Master default field formatting

Form Master Responsive also has improvements in these areas

- Improved Multi-Page Form Performance
- Redesigned Form Manager User Interface
- Improved Reporting Options

Form Master Responsive has removed support for:

- .Net Framework 3.5
- DotNetNuke Version 5.x

In addition, Form Master Responsive has been converted from VB.NET to C# for the sanity's sake.

**Note:** Parts of this user guide depict older versions of Form Master but are nonetheless still relevant to [Product Name]. Linked videos may also show older versions in name but the functionality is the same.

### See also

[Welcome](#)  
[Installation](#)  
[Getting Started](#)  
[Form Settings](#)  
[Basic Functions](#)  
[Working With Fields](#)  
[Cascading Dropdowns](#)  
[Multi-Page Forms](#)  
[Validation](#)  
[Data Manager](#)  
[Formatting](#)  
[Reporting](#)  
[Messaging](#)  
[How to Videos](#)  
[Advanced Functions](#)

[Home](#) > [Installation](#)

# Installation

Articles in this section



[System Requirements](#)



[Installing Form Master Responsive](#)

See also

[Welcome](#)  
[What's New](#)  
[Getting Started](#)  
[Form Settings](#)  
[Basic Functions](#)  
[Working With Fields](#)  
[Cascading Dropdowns](#)  
[Multi-Page Forms](#)  
[Validation](#)  
[Data Manager](#)  
[Formatting](#)  
[Reporting](#)  
[Messaging](#)  
[How to Videos](#)  
[Advanced Functions](#)

[Home](#) > [Installation](#) > [System Requirements](#)

## System Requirements

Platform:

**DotNetNuke® Version 6.2.0 or Later.** Form Master Responsive has been tested to DotNetNuke® Version 7.2 which was the latest version available when this help was published.

Database:

Form Master Responsive is designed to use the Microsoft® SQL Server version listed below.

- SQL Server 2005 Express
- SQL Server 2005 Standard, Workgroup, and Enterprise
- SQL Server 2008 Express
- SQL Server 2008 Standard, Workgroup, and Enterprise
- SQL Server 2012 Express
- SQL Server 2012 Standard, Workgroup, and Enterprise

See also

[Installing Form Master Responsive](#)

[Home](#) > [Installation](#) > [Installing Form Master Responsive](#)

## Installing Form Master Responsive

**DO NOT DELETE ANY PREVIOUS VERSION OF FORM MASTER**

Data loss will occur if the module is deleted

**BACKUP YOUR DATABASE!!!****BACKUP YOUR FILES!!!****BEFORE INSTALLING ANY MODULE!!!**

\*\*\*\*\* IMPORTANT \*\*\*\*\*  
If you have modified the default TokenIzed or XSL templates, please  
backup these files prior to installing the module, then replace them  
after installation is complete.  
\*\*\*\*\*

1. Download the appropriate file for your DotNetNuke version
2. Login as host
3. Install the module
  - o DNN 6.x:
    - Navigate to Host > Extensions
    - Hover over the Extensions Manage Menu, select Install Extension Wizard
  - o DNN 7.x
    - Navigate to Host > Extensions
    - Click on the "Install Extension Wizard" button
4. Click the Browse button and find FormMaster\_RSP\_4.n.n.n\_DNN-6.x-7x\_PA.zip
5. Click the Next button
6. Click Next on Package Information
7. Click Next on Release Notes
8. Check Accept License Checkbox then click Next which installs the module
9. On the results page check that there were no errors installing the module
10. You are done

As always it is recommended that you install into a test environment similar to your production environment, and of course backup all data and disk files before installing or upgrading any module in test or production.

See also

[System Requirements](#)

[Home](#) > [Getting Started](#)

# Getting Started

Articles in this section



[About Form Master Responsive](#)



[Form Submit Workflow](#)



[Form Elements](#)



[Menu Items](#)

[Getting Started Video](#)

### See also

[Welcome](#)  
[What's New](#)  
[Installation](#)  
[Form Settings](#)  
[Basic Functions](#)  
[Working With Fields](#)  
[Cascading Dropdowns](#)  
[Multi-Page Forms](#)  
[Validation](#)  
[Data Manager](#)  
[Formatting](#)  
[Reporting](#)  
[Messaging](#)  
[How to Videos](#)  
[Advanced Functions](#)

[Home](#) > [Getting Started](#) > [About Form Master Responsive](#)

## About Form Master Responsive

**Form Master Responsive** is a form creation module for the DotNetNuke® Content Management System. **Form Master Responsive** allows an average user to create a wide variety of input forms comprising all the common Input Controls available for any web page quickly and easily. As needs become more complex, **Form Master Responsive** provides the advanced user with the flexibility and tools to include a high degree of functionality in one complete package. Listed below are the functional areas which will be discussed in more detail later in this User Guide.

- [Form Creation](#)
- [Input Validation](#)
- [Data Storage](#)
- [Email](#)
- [Auto Responder](#)
- [Post Submission Options](#)
- [Data Retrieval](#)
- [User Registration](#)

### See also

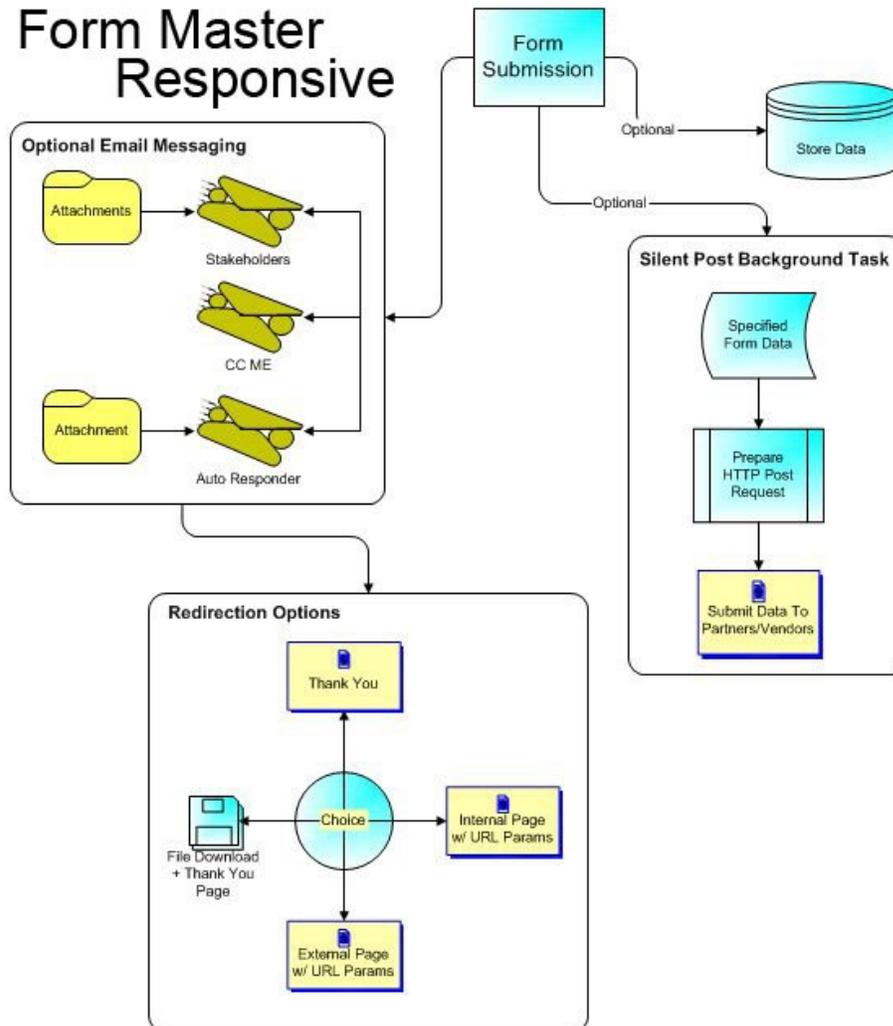
[Form Submit Workflow](#)  
[Form Elements](#)  
[Menu Items](#)

[Home](#) > [Getting Started](#) > [Form Submit Workflow](#)

## Form Submit Workflow



# Form Master Responsive

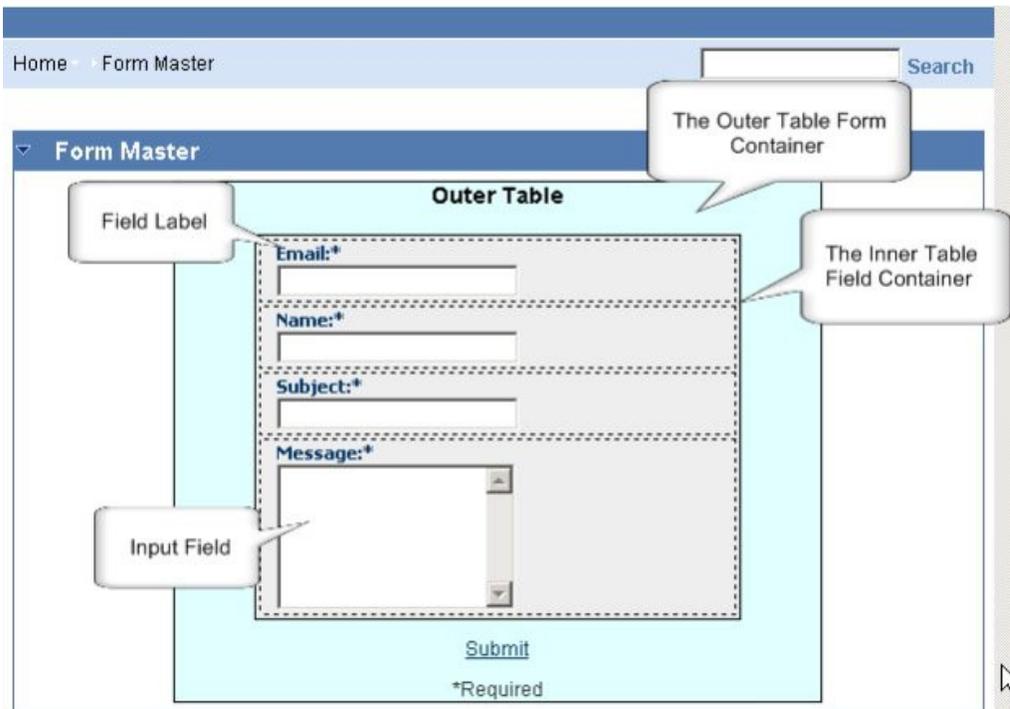


See also  
[About Form Master Responsive](#)  
[Form Elements](#)  
[Menu Items](#)

[Home](#) > [Getting Started](#) > [Form Elements](#)

## Form Elements

Form Master end user interface is made up of several HTML elements which will be referred to in other sections of the User Guide.



### See also

- [About Form Master Responsive](#)
- [Form Submit Workflow](#)
- [Menu Items](#)

[Home](#) > [Getting Started](#) > [Menu Items](#)

## Menu Items

- **Form**
  - **Edit Form:** This is where you will spend the majority of time. This section provides the tools to design the form
  - **Form Query:** The Form Query Editor allows for Getting Data from Database Tables, Spreadsheets, Text Files and other Data Sources supported by ADO.NET. This data can then be used to Auto Fill fields on the form based on the query Where clause. **Note:** Form Query is available for Super Users and users in the Administrators Role only.
  - **Field Rules:** The Field Rules page allows you to define rules which will show or hide fields on the form based on the value of other fields on the form. For example of Region is Other then show the Other Text Box field
- **Settings:**
  - **Form Settings:**
    - **Form Settings Tab:** Basic Settings, Validation Options, Form Wizard Options, Responsive Web Options
    - **Email Settings Tab:** Email Settings, Auto Response Settings, Email Template and Format Settings
    - **Redirection Settings Tab:** Redirect After Submit Options
    - **Other Settings Tab:** Web Service Options. Used to setup Authorization and get the URL for the Web Service. **Note:** Web Service Settings is available for Super Users and users in the Administrators Role only.
  - **Data Manager:** The Data Manager allows for Mapping Existing Fields to a Data Table, Create Table from Form, or Create Form from Table. **Note:** Data Manager is available for Super Users and users in the Administrators Role only.
  - **User Manager:** The User Manager page allows you to Register a User and include them in multiple Roles when the Form is Submitted. Users who are already Logged in, can be added to Roles. **Note:** User Manager is available for Super Users and users in the Administrators Role only.
- **Files:** Used to View and Manage uploaded Files
- **Messaging:**
  - **Form Heading:** Used for creating the form heading or introduction above the start of the form entry fields
  - **Form Thank You:** Used for creating the message displayed after the user submits the form
  - **Auto Responder:** Used for designing the Auto Responder Email
  - **Custom Template:** Used for designing the Custom Email Template
- **Report:** Used for viewing or downloading collected data
- **Help:**
  - **License Manager:** Allows SuperUsers to manage Module and Domain licensing (hosts only)
  - **Getting Started:** [Getting Stated Video](#)
  - **How To Videos:** [Helpful videos](#)
  - **User Guide:** Link to this user guide
  - **About Form Master:** View Release Notes, License Agreement and Version number information
- **Exit:** Leave the Form Master Edit menu

## See also

[About Form Master Responsive](#)  
[Form Submit Workflow](#)  
[Form Elements](#)

[Home](#) > [Form Settings](#)

## Form Settings

Articles in this section



[Basic Settings](#)



[Validation Options](#)



[Form Wizard Options](#)



[Responsive Web Options](#)



[Email Settings](#)



[Auto Responder Settings](#)



[Email Template & Format Settings](#)



[Redirect After Submit Options](#)



[Web Service Options](#)

### See also

- [Welcome](#)
- [What's New](#)
- [Installation](#)
- [Getting Started](#)
- [Basic Functions](#)
- [Working With Fields](#)
- [Cascading Dropdowns](#)
- [Multi-Page Forms](#)
- [Validation](#)
- [Data Manager](#)
- [Formatting](#)
- [Reporting](#)
- [Messaging](#)
- [How to Videos](#)
- [Advanced Functions](#)

[Home](#) > [Form Settings](#) > [Basic Settings](#)

## Basic Settings

## Basic Settings

Form Name:	<input type="text" value="Contact Us Web Form"/>
Outer Container Class:	<input type="text" value="OuterTableClass"/>
Main Form Container Width:	<input type="text"/>
Main Form Container Class:	<input type="text" value="MainFormTableClass"/>
Form Container Width:	<input type="text"/>
Form Container Height:	<input type="text"/>
Outer Container Alignment:	<input type="text" value="NotSet"/>
Inner Container Alignment:	<input type="text" value="Right"/>
Use Default Field Formatting	<input checked="" type="checkbox"/>
Survey Mode	<input type="checkbox"/>

- **Form Name:** The form name is used in the Subject for emails, and for the Cookie name when the form is in Survey Mode. Form names should be unique for each domain name for the Survey Mode feature to work properly.
- **Outer Container Class:** Optional Class name for the outer form container.
- **Main Form Container Width:** Sets the width of the outer container for the module. This container contains the heading text in row one and the form container in row two. Setting this value larger than the form container width allows for a wider heading than the form below.
- **Main Form Container Class:** The CSS Class name used for the container containing the form input fields.
- **Form Container Width:** The width of the container which contains all the controls. This container is inside the main form container which contains the header text in the first row and the form container in the second row. This value can be a whole number or a percentage.
- **Form Container Height:** The height of the container which contains all the controls. This container is inside the main form container which contains the header text in the first row and the form container in the second row. This value can be a whole number or a percentage. Percentage does not work with all browsers. Applies to non-responsive forms only.
- **Outer Container Alignment:** This is the outer container which contains the heading text in the top row and the form container in the second row. Applies to non-responsive forms only.
- **Inner Container Alignment:** This is the inner container which directly contains the fields and is contained in the outer form container. Applies to non-responsive forms only.
- **Use Default Field Formatting:** Check to use Form Master default field formatting CSS rules.
- **Survey Mode:** Check to put the Form in Survey Mode. In this mode, users will only be able to submit the form once. On future tries, they will be sent directly to the thank you page. This is controlled by UserID for Registered and Logged in users or by Cookie. This action is disabled for users in Edit roles.

Allow Anonymous Uploads <small>i</small>	<input type="checkbox"/>
Authenticate File Downloads <small>i</small>	<input checked="" type="checkbox"/>
Save Data to Database <small>i</small>	<input checked="" type="checkbox"/>
Use Captcha <small>i</small>	<input type="checkbox"/>
Captcha Characters: <small>i</small>	<input type="text"/>
Hide Posted By in reports <small>i</small>	<input checked="" type="checkbox"/>
Silent Post Url: <small>i</small>	<input type="text"/>
Suppress Required Field Text <small>i</small>	<input type="checkbox"/>
Required Field Symbol <small>i</small>	<input type="text" value="*"/>

- **Allow Anonymous Uploads:** Check this to allow users who are not logged in to upload files when a File Upload field is placed on the form.
- **Authenticate File Downloads:** Check this to require that a user be already Logged in and be in an Edit Role for the module in order to access files which have been uploaded when clicking from a link in an Email.
- **Save Data to Database:** Check to save the user submitted data to the database.
- **Use Captcha:** Check to use Captcha to prevent web robots from submitting the form.
- **Captcha Characters:** List of Characters to used for the Captcha Image. Enter at least 12 unique upper and lower case Characters
- **Hide Posted By in Reports:** Check this to suppress the Posted By Column which is generated. The information contains the IP Address of the User and Additional information if the user is logged in.
- **Silent Post Url:** Url to which a Silent Post of Form Data is made on Submit.
- **Suppress Required Field Text:** Check to suppress the display of the default "\*\*Required" text on the bottom of the form.
- **Required Field Symbol:** Symbol or Image used to mark a Field as Required  
Image Tag Example: `&nbsp;`

### See also

[Validation Options](#)  
[Form Wizard Options](#)  
[Responsive Web Options](#)  
[Email Settings](#)  
[Auto Responder Settings](#)  
[Email Template & Format Settings](#)  
[Redirect After Submit Options](#)  
[Web Service Options](#)

[Home](#) > [Form Settings](#) > [Validation Options](#)

## Validation Options

## Validation Options

Enable Client Script <small>i</small>	<input checked="" type="checkbox"/>
Use Form Error Formatting <small>i</small>	<input type="checkbox"/>
Use Validation Summary <small>i</small>	<input type="checkbox"/>
Summary Heading Text <small>i</small>	<input type="text"/>
Validation Summary CSS Class: <small>i</small>	<input type="text" value="NormalRed"/>
Show Validation Summary <small>i</small>	<input checked="" type="checkbox"/>
Show Validation Summary Message Box <small>i</small>	<input type="checkbox"/>
Supress Field Validation Messages <small>i</small>	<input type="checkbox"/>
Suppressed Field Character <small>i</small>	<input type="text" value="*"/>
Validation Summary Mode <small>i</small>	<input type="text" value="Bullet List"/>
Validation Summary Cell Alignment <small>i</small>	<input type="text" value="Center"/>

- **Enable Client Script:** When checked, enables field validation in the user's browser. When un-checked all validation is done on the Server which forces a reload of the page.
- **Use Form Error Formatting:** Uses CSS class c5sFormError for all validation formatting overriding values set in the individual validator controls
- **Use Validation Summary:** Check to use a Validation Summary Control on the Form. A Validation Summary shows all the Validation Errors in one central location and/or in an Alert popup box.
- **Summary Heading Text:** The Text which will appear before any error messages.
- **Validation Summary CSS Class:** The CSS class which is used for the displayed Error Messages.
- **Show Validation Summary:** Check to show the Validation Summary on the Page when a Validation Error Occurs.
- **Show Validation Summary Message Box:** Check to show an Alert Box when Form Validation Fails.
- **Suppress Field Validation Messages:** When using the Validation Summary, check this option to stop the Validation Messages from showing under the Fields.
- **Suppressed Field Character:** When Supress Field Validation Messages is On. This value will be output to the right of the Field. This value can be any character, string, or an Image Tag Example: ``
- **Validation Summary Mode:** Error Messages can be displayed in three modes: List, Bullet List, or Paragraph.
- **Validation Summary Cell Alignment:** Select the Cell Alignment for the Error Messaging.

### See also

[Basic Settings](#)  
[Form Wizard Options](#)  
[Responsive Web Options](#)  
[Email Settings](#)  
[Auto Responder Settings](#)  
[Email Template & Format Settings](#)  
[Redirect After Submit Options](#)  
[Web Service Options](#)

Home > Form Settings > Form Wizard Options

## Form Wizard Options

## Form Wizard Options

Button Type: 	<input type="text" value="Button"/>
Show Wizard Steps: 	<input checked="" type="checkbox"/>
Start Button Text: 	<input type="text" value="Start &gt;&gt;"/>
Next Button Text: 	<input type="text" value="Next &gt;&gt;"/>
Previous Button Text: 	<input type="text" value="&lt;&lt; Prev"/>
Finish Button Text: 	<input type="text" value="Finish"/>
Step Text: 	<input type="text" value="Step"/>
Step Class Name: 	<input type="text" value="Head"/>
Steps Alignment: 	<input type="text" value="Left"/>
Legend Class: 	<input type="text" value="Head"/>
Legend Alignment: 	<input type="text" value="Center"/>
Step Title Class Name: 	<input type="text" value="Head"/>
Step Title Alignment: 	<input type="text" value="Center"/>
Button Class: 	<input type="text" value="CommandButton"/>

- **Button Type:**
  - Button: Standard HTML button
  - LinkButton: Hyperlink style button
  - ImageButton: Allows for Uploading/Selecting a custom image for each button type
- **Show Wizard Steps:** When checked, shows the short page name text as steps at the top of the form
- **Start Button text:** The text to be used for the Button or LinkButton text for the first step in the form wizard
- **Next Button Text:** The text to be used for the next Button or LinkButton on all intermediate steps in the form wizard
- **Previous Button Text:** The text to be used for the previous Button or LinkButton on all intermediate steps in the form wizard
- **Finish Button Text:** The text to be used for the Button or LinkButton text for the final step in the form wizard
- **Step Text:** Text for the Wizard Step Number above the Short Name. Leave blank to not use a Step Number.
- **Step Class Name:** The CSS Class name for the text shown at the top of a multi-page Form Wizard.
- **Steps Alignment:** Horizontal alignment for the Wizard Steps at the top of a Multi-Page Form.
- **Legend Class:** CSS Class for the page Short Name text. Note: Use C5SLLC\_hidden to hide this text.
- **Legend Alignment:** Legend Alignment for Multi-Page Short Name in the Form Wizard
- **Step Title Class Name:** The CSS Class name for the step Title shown at the top of a Multi-page Form Wizard.
- **Step Title Alignment:** Horizontal alignment for the Wizard Title at the top of a Multi-Page Form.
- **Button Class:** Class to be applied to the Form Wizard Buttons or LinkButtons

## See also

[Basic Settings](#)  
[Validation Options](#)  
[Responsive Web Options](#)  
[Email Settings](#)  
[Auto Responder Settings](#)  
[Email Template & Format Settings](#)  
[Redirect After Submit Options](#)  
[Web Service Options](#)

Home > Form Settings > Responsive Web Options

## Responsive Web Options

**Responsive Web Options**

Use Responsive Layout: i

Use Hard Row Breaks: i

Group Width: i

Label Width: i

Label Align: i

Cell Height: i

- **Use Responsive Layout:** Check to use a Responsive Web (Table-Less) form layout rendering.
- **Use Hard Row Breaks:** Check to use Hard Row Breaks in your Responsive Form Layout. Checking this may make the form have varying controls on each row depending on the width of the device being used to view the form.
- **Group Width:** The combined width of the field label and input control. Can be affected by vaules set in individual controls in the control edit page.
- **Label Width:** This value specifies the width of all labels in a form. This value can be overridden by each form field. This is effective when there is more than one column and when the field labels are set to left.
- **Label Align:** Alignment of the Label text this can be overridden in each controls Label.
- **Cell Height:** Height for the Cell containing the Label and the Input Control.

## See also

[Basic Settings](#)  
[Validation Options](#)  
[Form Wizard Options](#)  
[Email Settings](#)  
[Auto Responder Settings](#)  
[Email Template & Format Settings](#)  
[Redirect After Submit Options](#)  
[Web Service Options](#)

Home > Form Settings > Email Settings

## Email Settings

### Email Settings

Send To Action: i

To Email: i

Send From Action: i

From Email: i

ReplyTo Email: i

ReplyTo Email: i

Show CC Me on Form i

CC Me Email: i

BCC: i

Include Direct Data Link in Email i

Use Form Field Value for Email Subject i

Field Value to Append to Subject i

Second Field Value to Append to Subject i

- **Send to Action:** There are four options described below:
  - **None:** No email is sent when the form is submitted
  - **Use Admin Email:** Email is sent to the Site Administrator when form is submitted
  - **User Defined:** Email is sent to the email address entered in the "To Email:" textbox. Multiple recipients can be entered separated by a semi-colon.
  - **Use Form Value:** Email is sent to the email address in a selected form field. This is useful for example: A Dropdown List could contain Regions with the value of each region being an email address of the person responsible for covering that region.
- **To Email:** Depending on the value of the "Send To Action" this can be:
  - A Disabled Textbox
  - An Enabled Textbox
  - A Dropdown list of Fields defined on the form
- **Send From Action:** This is important when the recipient of the email hits Reply in their Email client. There are three options described below:
  - Use Admin Email: The email is sent from the Site Administrator
  - **User Defined:** Email is sent from the email address entered in the "From Email:" textbox
  - **Use Form Value:** Email is sent from the email address in a selected form field.
- **From Email:** Depending on the value of the "Send From Action" this can be the same as the "To Email" field in #5 above
- **Show CC Me on Form:** This will enable/disable the checkbox at the bottom of the form so users can be Carbon Copied on the email which is sent.
- **CC Me Email:** When "Show CC Me on Form" is checked allows for the selection of the form field to use to send a copy to the user.
- **BCC:** Use the Blind Carbon Copy to send an email to any email address entered in the Textbox. Multiple recipients can be entered separated by a semi-colon.
- **Use Form Field for Email Subject:** When checked, the Field Value specified in the Dropdown List will be used as the Subject line.
- **Include Direct Data Link in Email:** Check to include a direct link to the record in Reports in the email.
- **Field Value to Append to Subject:** Specify the Field Value to Append to the Subject.
- **Second Field Value for Email Subject:** Optionally Select a Second Field Value to Append to the Subject Line selected above.

## See also

[Basic Settings](#)  
[Validation Options](#)  
[Form Wizard Options](#)  
[Responsive Web Options](#)  
[Auto Responder Settings](#)  
[Email Template & Format Settings](#)  
[Redirect After Submit Options](#)  
[Web Service Options](#)

[Home](#) > [Form Settings](#) > [Auto Responder Settings](#)

## Auto Responder Settings

### Auto Responder Settings

Send Auto Responder i

Auto Responder Send To Field i Email ▼

Auto Responder From Email: i sales@mycompany.com

Auto Response Subject: i MySubject

Auto Responder File Attachment: i

File Location: Root ▼

File Name: <None Specified> ▼

[Upload New File](#)

- **Send Auto Responder:** If Checked, the form will send an Email Auto Responder using the Field value selected in the Auto Responder Send To Field Dropdown List.
- **Auto Responder Send To Field:** Select the field to use as the Send To email address for the Auto Responder.
- **Auto Responder From Email:** Specify the Auto Responder From Email address. For Example ([sales@mycompany.com](mailto:sales@mycompany.com)).
- **Auto Response Subject:** Specify the Email Subject for the Auto Responder Subject Line
- **Auto Responder File Attachment:** Specify a File to be attached to the Auto Responder Email. Usefull for Software Trials etc.

## See also

[Basic Settings](#)  
[Validation Options](#)  
[Form Wizard Options](#)  
[Responsive Web Options](#)  
[Email Settings](#)  
[Email Template & Format Settings](#)  
[Redirect After Submit Options](#)  
[Web Service Options](#)

## Email Template & Format Settings

### Email Template and Format Settings

Email Template Type ⓘ

Email Endoding ⓘ

Text Template: ⓘ

Send HTML formatted Email ⓘ

Send Multipart Email ⓘ

Use custom Email XSL Stylesheet ⓘ

Upload Custom Email Stylesheet: ⓘ

- **Email Encoding** : Form most installations using the Default value will work just fine. If there are problems with the email containing strange characters, experimenting with the other encodings may help with the problem.
- **Email Template**: Form Master offers two Template styles for formatting the email message which are described below:
  - **Text Template**: The text templates can be used for "Plain Text", or "Html" formatting. For each format the file contains text with replaceable tokens to insert the form data. The files can be found in the Templates folder in the Module root. Copying these files to a new folder under the Templates folder allows for creating multiple templates on a Host.

To Make Templates which are Specific to Individual Portals Create a Folder Under the Templates Fold corresponding the the Portal's PortalID for example Templates/0

Each format is made up of three files with the .TXT extension for "Plain Text" and .HTM for "Html":

- HEADER
- ITEM
- FOOTER

- **XSL Templates**: The XSL templates are available for those users familiar with XSL style sheet transformations. For version 1.1, this was the default method for formatting the email message.

The XSL Templates offer the advantage of being able to supply a unique custom template for each instance of Form Master without affecting any other instance. A sample data document located as "assets/xml/SampleData.xml" from the module root can be used to develop the XSL style sheet.

The "Text", or "Html" format default style sheets can be found at "assets/xslt", EmailSend.xslt and EmailSendHtml.xslt. It is recommended that not to overwrite these files, but rather upload a custom style sheet using the "Upload Custom Email Style Sheet" option.

- **Custom Template**: A Custom Template can be designed using the HTML Editor and the Field Tokens defined when creating the Form. This Template type is by Default sent in HTML format.
- **Text Template**: Only "default" is available in the Dropdown List until new templates are added by copying the default files to a new folder as noted in 19.1 above. When new templates are available the folder name will appear in the Dropdown List.
- **Send HTML formatted Email**: By default Form Master sends email in "Plain Text" format, If "Html" format is needed, simply check this box, and the default template with be used, for either style of template. (Custom Template is HTML always)
- **Send Multipart Email**: Check to Send HTML email with a Text Section for Email Readers which cannot of choose not to show HTML (Note: Only applies to HTML emails) Checked by Default.
- **Use custom Email XSL Stylesheet**: To use a custom XSL style sheet check this checkbox.
- **Upload custom Email Stylesheet**: File Upload to upload a custom style sheet when #22 is checked

### See also

[Basic Settings](#)  
[Validation Options](#)  
[Form Wizard Options](#)  
[Responsive Web Options](#)  
[Email Settings](#)  
[Auto Responder Settings](#)

[Redirect After Submit Options](#)  
[Web Service Options](#)

[Home](#) > [Form Settings](#) > [Redirect After Submit Options](#)

## Redirect After Submit Options

**Redirect After Submit Options**

**Redirection After Submit:** ⓘ

**Link Type:**

None

URL ( A Link To An External Resource )

Page ( A Page On Your Site )

File ( A File On Your Site )

- **Redirection After Submit:** There are four options for redirection described below:
  - **None:** No redirection the Thank You page is shown by default
  - **URL:** This allows the user to enter any valid Internet Address.
  - **Page:** This allows the user to select a page within the site
  - **File:** This allows the user to select a file on the server for download to the user's browser. When this option is used the default Thank You message is shown and for most Browsers (User Agents), the file download should automatically start. A link to the file is provided for those browsers which block automatic downloads.

### See also

[Basic Settings](#)  
[Validation Options](#)  
[Form Wizard Options](#)  
[Responsive Web Options](#)  
[Email Settings](#)  
[Auto Responder Settings](#)  
[Email Template & Format Settings](#)  
[Web Service Options](#)

[Home](#) > [Form Settings](#) > [Web Service Options](#)

## Web Service Options

**Web Service Options**

**Web Service Access Roles** ⓘ

Administrators     Subscribers     Translator (en-US)     Unverified Users

Registered Users

**URL for Form Master Web Service** ⓘ

- **Web Service Access Roles:** Check the Roles to enable Access to Web Services. To disable Web Services un-check all Roles.

- **URL for Form Master Web Service:** Copy this URL into your Web Service Client Application or Code.

See also

[Basic Settings](#)  
[Validation Options](#)  
[Form Wizard Options](#)  
[Responsive Web Options](#)  
[Email Settings](#)  
[Auto Responder Settings](#)  
[Email Template & Format Settings](#)  
[Redirect After Submit Options](#)

---

Home > Basic Functions

## Basic Functions

Articles in this section



[Form Layout](#)



[Tab Index](#)



[Localization](#)



[File Management](#)

### See also

- [Welcome](#)
- [What's New](#)
- [Installation](#)
- [Getting Started](#)
- [Form Settings](#)
- [Working With Fields](#)
- [Cascading Dropdowns](#)
- [Multi-Page Forms](#)
- [Validation](#)
- [Data Manager](#)
- [Formatting](#)
- [Reporting](#)
- [Messaging](#)
- [How to Videos](#)
- [Advanced Functions](#)

Home > Basic Functions > Form Layout

## Form Layout

The Form Master Responsive Form Layout page allows you to add, delete, and Move fields in a visual designer. The designer includes several options which are described following the Screenshots shown below.

Form Designer
Designer Settings

### Designer Settings

Label Position: ⓘ Left ▼

Cols: ⓘ 3 ▼

Rows: ⓘ 4 ▼

Pages: ⓘ Page 1 ▼

Page Short Name ⓘ Page 1

Page Title ⓘ

Show Title ⓘ

- **Label Position:** Specify where the Label for the Field will be shown. Options are:
  - **Above:** The text in the [Label Field](#) will be output above the Field in the same Table Cell
  - **Left:** The text in the Label Field will be output to the Left of the Field in a separate Table Cell
- **Cols:** The Number of columns in the form table
- **Rows:** The Number of rows in the form table
- **Pages:** List of pages contained in this form. Can be ignored for simple single page forms.
- **Mode:** Enables previewing of the form while editing
- **Page Short Name:** Short name for this page. Used in dropdowns and other editing controls. This name will be used as the steps text at the top of the form.
- **Page Title:** The Title for the page. This can optionally be shown as a header for each form page.
- **Show Title:** Indicate if the Title should be shown as a heading for this page.
- **Update:** Click the Update link to save changes to the settings listed above.
- **Reset Form:** Click the "Reset Form" link to delete all Fields from the form. **Note:** This cannot be undone.

C:1	
<b>R:1</b>	Email:* <input style="width: 80%;" type="text"/> <span style="color: red; font-weight: bold;">Please enter an email Email format is invalid</span> <span style="color: green; font-size: small;">Edit Del Move</span> CS: <span style="border: 1px solid #ccc; padding: 0 5px;">1</span> ▼ RS: <span style="border: 1px solid #ccc; padding: 0 5px;">1</span> ▼ TI: <span style="border: 1px solid #ccc; padding: 0 5px;">1</span> ▼
<b>R:2</b>	Add
<b>R:3</b>	Name:* <input style="width: 80%;" type="text"/> <span style="color: red; font-weight: bold;">Please enter a name</span> <span style="color: green; font-size: small;">Edit Del Move</span> CS: <span style="border: 1px solid #ccc; padding: 0 5px;">1</span> ▼ RS: <span style="border: 1px solid #ccc; padding: 0 5px;">1</span> ▼ TI: <span style="border: 1px solid #ccc; padding: 0 5px;">2</span> ▼

The top row of the form layout contain "C:" and the Column number. Clicking on for example "C:1" opens the Column Manager which will allow you to Insert or Delete Columns in the form.

The first Column of the form layout contains "R:" and the Row Number. Clicking on for Example "R:1" opens the Row Manager which will allow you to Insert or Delete Rows in the form.

- Add: The "Add" link opens the Input Control Form where a new Form Field can be added.
- Edit: The "Edit" link allows for editing a Form Field.
- Move: The "Move" link allows for moving a Field to another location on the form.
- CS: Column Span for the Field.\*
- RS: Rowspan for the Field.\*
- TI: Tab Index for the Field. When the user uses the Tab key to move between fields on a form the TI dropdown list can be used to control the order of the fields the will be move to when the Tab Key if pressed

[View the Form Layout Video](#)

\*This functionality is not available when "[Use Responsive Layout](#) " is checked.

See also

[Tab Index](#)  
[Localization](#)  
[File Management](#)

[Home](#) > [Basic Functions](#) > [Tab Index](#)

## Tab Index

The **Tab Index** Feature allows for the ordering of the input focus when the user presses the "Tab" key on their keyboard

The **Tab Index** is set by using the "TI:" dropdown in each field cell while in design mode. Label fields do not participate in the Tab Index.

The Ordering can be Left-Right, then Top-Down.

[Setting Left-Right then Top-Down Tab Index Presentation](#)

Or Top-Down, then Left-Right

[Setting Top-Down, then Left-Right Tab Index Presentation](#)

In reality, the ordering can be in any combination, these shown are the most common.

See also

[Form Layout](#)  
[Localization](#)  
[File Management](#)

[Home](#) > [Basic Functions](#) > [Localization](#)

## Localization

Form Master Responsive supports localization using the standard **DotNetNuke®** methods. The App\_LocalResources folder in the Module root can be used to create Resource files for any Locale.

The files Resource files are listed below:

- FormAbout.ascx.resx
- FormAutoRespMsg.ascx.resx
- FormCustomTemplate.ascx.resx
- FormDataServiceSettings.ascx.resx
- FormDispatcher.ascx.resx
- FormFieldRules.ascx.resx
- FormHeadingEdit.ascx.resx
- FormLicenseMgr.ascx.resx
- FormMaster.ascx.resx
- FormMasterDBMgr.ascx.resx
- FormMasterEdit.ascx.resx
- FormMasterFiles.ascx.resx
- FormMasterFormOptions.ascx.resx
- FormMasterFormQuery.ascx.resx
- FormMasterOptions.ascx.resx
- FormReport.ascx.resx
- FormThankYouEdit.ascx.resx
- FormUserManager.ascx.resx
- SharedResources.resx

In addition the DotNetNuke Membership field names for the Auto-Fill feature can be Localized using the Lists menu option on the Host Menu.

Form Master Responsive does not support Multi-Lingual content at this time.

See also

- [Form Layout](#)
- [Tab Index](#)
- [File Management](#)

Home > Basic Functions > File Management

## File Management

**Form Master Responsive** includes a **File Manager** page to manage the files uploaded by users.

The Files page allows for **Downloading**, **Searching**, and **Deleting** of uploaded files. **Sorting** is also available on the **File ID**, **File**, and **Date** columns.

When a file is uploaded when the "Save Data to Database", on the "Form Settings" is checked, the file cannot be deleted using the Files page since the files are linked to the data submitted by the user. These files are available here for download, but must be deleted with the related record

### Form File Manager

File Name: ?

1	Form Data ID	File ID	File	Date	Delete
	18	7	Bach.jpg	5/22/2013 2:50:00 PM	
	18	6	ConstantContact.txt	5/22/2013 2:50:00 PM	

1

**Files associated with a Data Record, must be deleted with that Record in Reports.**

See also

- [Form Layout](#)
- [Tab Index](#)
- [Localization](#)



[Home](#) > [Working With Fields](#)

## Working With Fields

Articles in this section



[Field Settings](#)



[Adding Fields](#)



[Calculated Fields](#)

See also

- [Welcome](#)
- [What's New](#)
- [Installation](#)
- [Getting Started](#)
- [Form Settings](#)
- [Basic Functions](#)
- [Cascading Dropdowns](#)
- [Multi-Page Forms](#)
- [Validation](#)
- [Data Manager](#)
- [Formatting](#)
- [Reporting](#)
- [Messaging](#)
- [How to Videos](#)
- [Advanced Functions](#)

[Home](#) > [Working With Fields](#) > [Field Settings](#)

## Field Settings

This opening paragraph should describe the feature that you are documenting. Explain how it is commonly used and what the benefits are. For example: The Widget Master email link allows you to easily send information about each widget to various departments within your company. Often, the feature that you are documenting can be best explained by walking the reader through step by step. Use screenshots to illustrate the steps where possible.

1. **Start the application by...**
2. **On the startup screen, click the...**
- 3.

See also

- [Adding Fields](#)
- [Calculated Fields](#)

[Home](#) > [Working With Fields](#) > [Field Settings](#) > [Input control](#)

## Input control

Common Options are common to mostl types of Form Fields i.e the Textbox, Dropdown List, etc.

### Basic Settings ^

Type: ⓘ

Label: ⓘ

Field Name: ⓘ

Disabled: ⓘ  Disable User Input

Hide Label: ⓘ  Hide Label:

Field Token: ⓘ

Exclude Field: ⓘ  Exclude from Email and Data

Allow Markup: ⓘ  Allow Markup (html)

Show on own row ⓘ  Show on own row

Calculation Formula ⓘ

[Calculation Helper](#)

- **Type:** Select the type of field needed
  - Textbox
  - Checkbox
  - DropDownList
  - ListBox
  - CheckBoxList
  - RadioButtonList
  - Label
  - File Upload
  - Submit Button
  - Hidden Field
  - HTML Editor
- **Label:** The visible Label for the input field.
- **Field Name:** The Name which will be used in the Database Columns, and for Columns in Reports
- **Disabled:** Disable User input ,useful if user need to see pre-filled data but is not allowed to change it
- **Hide Label:** Hides the Label specified in #1. This can be useful when a actual Label Control is used as a heading for a CheckBoxList for example.
- **Field Token:** A Replaceable Token which is replaced by the Submitted Data in Email messaging and the Thank You message. Usually formatted like [TOKENNAME]
- **Exclude Field:** Exclude the field from being sent in the Email & Stored in the Database. This is useful when the Silent Post Feature is being used and it is not necessary to store or email required Hidden fields used in the Silent Post
- **Allow Markup:** Check to allow markup (html tags) in field input.
- **Show on own row:** When Responsive Layout is checked in Form Settings, shows this control on it's own row. This has no effect in Table Layout.
- **Calculation Formula:** Enter an Excel style formula surrounded by Parenthesis Example: (\$A1+\$A2).

Calculation Helper ?
Calculation Helper

Undo   Clear

### Operators

+   -   ÷   ×   ^   ,   √   if   <   <=   >=   =   <>   mod   max   min   sum   avg   (   )

0   1   2   3   4   5   6   7   8   9   \$

### Page 1

SA11-FirstName	SA12-Address	SA13-PostalCode	SB11-LastName	SB12-City	SB13-BirthDate	SC11-Email
SC12-State	SC13-Total					

TODO: need a section on Calculations

## Layout & Style Settings

Width: ?

Label Align: ? NotSet ▼

Control Align: ? NotSet ▼

Label Class: ? SubHead

Height: ?

Class: ? NormalTextBox

- **Width:** Specify the Width of the input control in Pixels. Example: 250
- **Label Align:** The alignment of the Label in the Table Cell when Label Position is on Left
  - Not Set
  - Left
  - Center
  - Right
- **Control Align:** The alignment of the input control in the Table Cell
  - Not Set
  - Left
  - Center
  - Right
- **Label Class:** The CSS class applied to the Label Default is DotNetNuke "SubHead" class. Portal.css is the best place to define CSS Class Rules
- **Encrypt Field:** Useful for hiding Data on the Web Page such as a List of Email Addresses which are used for the Send To: address on a Lead Generation Form
- **Height:** Specify the Height of the input control in Pixels. Example: 150
- **Class:** The CSS class applied to the input control. Default is DotNetNuke "NormalTextBox". More than one class name can be used separated by a space which is normal for specifying class names at the HTML level. The field also allows for a second Class name separated by a semi-colon ";" which when present is applied to the Table Cell containing the input control. Portal.css is the best place to define CSS Class Rules

See also

[Auto Fill](#)

List Options  
Other Options

Home > Working With Fields > Field Settings > Auto Fill

## Auto Fill

### Fields for all Auto Fill Types

- **Encrypt Field:** Useful for hiding Data on the Web Page such as a List of Email Addresses which are used for the Send To: address on a Lead Generation Form.
- **Auto Fill Type:** Select from where the Auto Fill Data will be Supplied.

Input Control   Auto Fill   List Options   Other Options

Auto Fill Options

Encrypt Field:  Encrypt Field

Auto Fill Type: Default Text

Default Text:

### When Auto Fill Type is "Default Text":

**Default Text:** The Default Text for the Textbox Control. Data entered here will be populated to the field. The Default Text is not available for all field types

Auto Fill Options

Encrypt Field:  Encrypt Field

Auto Fill Type: Membership

Auto Fill From: << Select >>

Field Query Options

Connection String:

SQL Query:

- << Select >>
- Profile: BirthDate
- Name: Prefix
- Name: FirstName
- Name: MiddleName
- Name: LastName
- Name: Suffix
- Address: Unit
- Address: Street
- Address: City
- Address: Region
- Address: Country
- Address: PostalCode
- Contact Info: Telephone
- Contact Info: Email
- Contact Info: Cell
- Contact Info: Fax
- Contact Info: Website

### When Auto Fill Type is "Membership":

**Auto Fill From:** Select the Membership data which should be populated into the control. The Membership Auto Fill Type is not available for all field types.

Auto Fill Options

Encrypt Field:  Encrypt Field

Auto Fill Type:

Auto Fill Value:

When Auto Fill Types are: Session, Url, Cookie, Form Query, or Field Query:

**Auto Fill Value:** The data entered here has different a meaning based on what Auto Fill Type above is selected

- **Session:** The Session Variable name
- **Url:** The name of the Url Parameter
- **Form and Field Query:** The name of a column specified in the query.

Auto Fill Options

Encrypt Field:  Encrypt Field

Auto Fill Type:

Auto Fill From:

Field Query Options

- << Select >>
- HTTP\_HOST
- HTTP\_REFERER**
- HTTP\_USER\_AGENT
- PATH\_INFO
- QUERY\_STRING
- REMOTE\_ADDR
- REMOTE\_HOST
- URL
- SERVER\_NAME

**When Auto Fill Type is "Server Variables":**

**Auto Fill From:** Select the Server Variable data which should be populated into the control. The Server Variable Auto Fill Type is not available for all field types. [Product Name] does not support all possible Server Variables. [Use this link](#) to learn more about the Server Variables.

### Field Query Options

**About The Field Query** ⓘ

**Connection String:** ⓘ

**SQL Query:** ⓘ

**Query Tokens** ⓘ

**Text Column:** ⓘ

**Value Column:** ⓘ

Value	Text
AK	Alaska
AL	Alabama
AR	Arkansas
AZ	Arizona
CA	California
CO	Colorado

**The Field Query is a multi-purpose tool:**

1. Use it to query a Data Source for one Record then used the Auto Fill Value field to supply the column name to Auto Fill the control from the Column Value
2. Create a multi-row Query result and fill a List Control with the return values using the Text Column, and Value Column fields

**Fields:**

- **Connection String:** The Connection string to the Data Resource. Leave blank to use the Default System Connection String.
- **SQL Query:** The SQL Query to use for Data Access. Your query must start with "Select" Delete and Update queries are not allowed.
- **Query Tokens:** Replacable Tokens from Membership Url and Session parameter/variable names which can be used in Queries. Tokens must be used Unchanged to work. If you are working with [Cascading Dropdowns](#) , you can use the CASCADE token.
- **Text Column:** The name of the Column to use to populate the Text (visible) part of a List type control.
- **Value Column:** The name of the Column to use to populate the Value part of a List type control.

See also

[Input control](#)  
[List Options](#)  
[Other Options](#)

Home > Working With Fields > Field Settings > List Options

## List Options

Input Control
Auto Fill
List Options
Other Options

### List Control Options

Fill With States: ⓘ Fill List Items with States From Afghanistan ▼

Fill With Countries: ⓘ Fill List Items with Countries

Auto Fill Part: ⓘ Text ▼

List Data To Save: ⓘ Text ▼

Cascading List: ⓘ  Cascading

Fields:

- **Fill With States:** Fill an input field type of DropDownList, CheckBoxList, or RadioButtonList with States or Provinces.
- **Fill With Countries:** Fill an input field type of DropDownList, CheckBoxList, or RadioButtonList with Countries.
- **Auto Fill Part:** When using the Auto Fill for Regions or Countries, choose which part of the pair to use. Example: California (Text), or CA (Value). This choice is single use and the selection is not saved.
- **List Data To Save:** Select to save the "Text" or the "Value" to the database and send in the Email.
- **Cascading List:** Check if this List Control will be populated based on the value of the selected parent Dropdown List Control. See More information on [Cascading Dropdowns](#) .

### List Items

Text	Value	Selected	
		<input type="checkbox"/>	<a href="#">Add</a> <a href="#">Delete All</a>

Enter data for the Text part and data for the Value part of the list control, select if this item should be selected then click "Add"

See also

[Input control](#)  
[Auto Fill](#)  
[Other Options](#)

[Home](#) > [Working With Fields](#) > [Field Settings](#) > [Other Options](#)

## Other Options

### Text Box Options

Text Mode: ⓘ

Wrap Text: ⓘ  Wrap in Text Box

Columns: ⓘ

Rows: ⓘ

Use Calendar: ⓘ  Use Calendar

Max Length: ⓘ

Place Holder ⓘ

- **Text Mode:** Text mode for an input field of type TextBox.
- **Wrap Text:** Check this to automatically Wrap text as the User types in a Multiline Textbox.
- **Columns:** Number of columns for a TextBox input field (for downlevel browsers).
- **Rows:** Number of rows for a Multiline Text Mode TextBox input field (for downlevel browsers).
- **Use Calendar:** Check to use a Calendar Popup for this Textbox.
- **Max Length:** Maximum number of characters allowed in the input field.
- **Place Holder:** Text to display in the field as a user prompt.

### Validation Options

**Validators**

Validator Details	
Validator Type:	<input type="button" value="Add Validator"/>
<input type="text" value="RequiredFieldValidator"/> <input type="text" value="RequiredFieldValidator"/> <input type="text" value="RegularExpressionValidator"/> <input type="text" value="CompareValidator"/>	

Most Field Types support one or more validation controls. See the [Validation](#) topic for more information.

### Send Value By Options

Send Value By: ⓘ

Value Name: ⓘ

Cookie Expiration: ⓘ

You can send any values collected when a user submits a form to any type of system which can accept the HTTP verbs POST or GET. You can also send via a Session or Cookie variable which allows you to link values in multiple [Product Name] forms or other modules which can make use of Session Variables.

**Send By Methods:**

- Session
- Url

- [Cookie](#)
- [Silent Post](#)
- [Silent Post and Session](#)
- [Silent Post and Cookie](#)

See the [Silent Post](#) topic for more information.

### See also

[Input control](#)  
[Auto Fill](#)  
[List Options](#)

[Home](#) > [Working With Fields](#) > [Adding Fields](#)

## Adding Fields

This opening paragraph should describe the feature that you are documenting. Explain how it is commonly used and what the benefits are. For example: The Widget Master email link allows you to easily send information about each widget to various departments within your company. Often, the feature that you are documenting can be best explained by walking the reader through step by step. Use screenshots to illustrate the steps where possible.

1. **Start the application by...**
2. **On the startup screen, click the...**
- 3.

### See also

[Field Settings](#)  
[Calculated Fields](#)

[Home](#) > [Working With Fields](#) > [Adding Fields](#) > [Textbox](#)

## Textbox

The **Textbox** is the workhorse of most any form, and adding a Textbox only takes a few mouse clicks and keystrokes. In addition to adding a Textbox, this presentation will introduce:

- [Adding rows to the form](#)
- [Entering a Label for a field](#)
- [Using the Auto-Fill feature](#)
- [Introduce the field management controls](#)

[Adding a Textbox Video](#)

### See also

[Checkbox](#)  
[Dropdown List](#)  
[Listbox](#)  
[Checkbox List](#)  
[Radiobutton List](#)  
[Label](#)  
[File Upload](#)  
[Submit Button](#)  
[Hidden Field](#)  
[HTML Editor](#)

[Home](#) > [Working With Fields](#) > [Adding Fields](#) > [Checkbox](#)

## Checkbox

The **Checkbox** is a very simple field control to add to the form.

For the Checkbox the “Label Class” is used to format the text.

[Adding a Checkbox Video](#)

### See also

[Textbox](#)  
[Dropdown List](#)  
[Listbox](#)  
[Checkbox List](#)  
[Radiobutton List](#)  
[Label](#)  
[File Upload](#)  
[Submit Button](#)  
[Hidden Field](#)  
[HTML Editor](#)

[Home](#) > [Working With Fields](#) > [Adding Fields](#) > [Dropdown List](#)

## Dropdown List

The **Dropdown List** and all the List type field controls share one common element which is the “List Item”.

A List Item is what appears in the Dropdown List when you click on the down arrow. A List Item is made up of two parts the “Text” and the “Value” parts. These two values can be the same, or as an example in a list of US states “New York” will be the Text part and “NY” will be the Value part.

With the Dropdown List, it is always a best practice to enter a Prompt for the user as the first item in the list for example: Enter “<< Select >>” for the Text part and “0” for the Value part. This will ensure that Validation can be applied to the field forcing the user to make a selection from the list. Without Validation, users will often submit the form without making a selection, submitting the first item in the Dropdown List as default.

Adding the Dropdown List will introduce new features, and make use of some which have already been covered:

- Use of the Auto-Fill feature which in the case of the Dropdown will find the correct value in the list and pre-select that for the user
- Manually adding a Item to the List Items form
- Adding List Items by using the Fill List Items with States from a selected Country feature

We will add a Required Field Validator to this Dropdown List in the Validation section.

[Add DropDown List Presentation](#)

### See also

[Textbox](#)  
[Checkbox](#)  
[Listbox](#)  
[Checkbox List](#)  
[Radiobutton List](#)  
[Label](#)  
[File Upload](#)  
[Submit Button](#)  
[Hidden Field](#)  
[HTML Editor](#)

[Home](#) > [Working With Fields](#) > [Adding Fields](#) > [Listbox](#)

## Listbox

The **Listbox** control it useful where there are a small number of items in a list for the user to choose from.

Typically, a Listbox can be in two modes, Single, or Multi select modes. By default, Form Master uses Multi select mode only. The reason for this is simple, for **Single selection mode**, a **Dropdown List**, or **Radiobutton List** provide a better user interface.

For Multi select mode, the preferred field control is the **Checkbox List** which offers a better user interface as there is no need for the user to know how to use

“Ctrl Click” Selects Multiple Items functionality.

The one advantage in using a **Listbox** is when a user must make a selection, as the Checkbox List cannot have a **Required Field Validator**, whereas the Listbox can.

In addition to adding a **Listbox**, this presentation will introduce:

- Adding a Column to the form
- Moving Controls between cells

[Adding a Listbox Video](#)

### See also

[Textbox](#)  
[Checkbox](#)  
[Dropdown List](#)  
[Checkbox List](#)  
[Radiobutton List](#)  
[Label](#)  
[File Upload](#)  
[Submit Button](#)  
[Hidden Field](#)  
[HTML Editor](#)

[Home](#) > [Working With Fields](#) > [Adding Fields](#) > [Checkbox List](#)

## Checkbox List

The **Checkbox List** is used when the number of items in the list is small, and the user is allowed to select multiple items in the list.

We will add a Required Field Validator to this Checkbox List in the Validation section

[Adding a Checkbox List](#)

### See also

[Textbox](#)  
[Checkbox](#)  
[Dropdown List](#)  
[Listbox](#)  
[Radiobutton List](#)  
[Label](#)  
[File Upload](#)  
[Submit Button](#)  
[Hidden Field](#)  
[HTML Editor](#)

[Home](#) > [Working With Fields](#) > [Adding Fields](#) > [Radiobutton List](#)

## Radiobutton List

The **RadioButton List** is used when the number of items in the list is small, and the user is allowed to select one item in the list. The procedure for adding a RadioButton List is the same as for the Checkbox List. The only two differences are:

1. There can only be one selected item in the List Items form, whereas the Checkbox List can have multiple items selected in the List Items form.
2. The Required Field Validator can be used to require that a user make a selection.

We will add a Required Field Validator to this RadioButton List in the Validation section

[Adding a Radiobutton List Video](#)

### See also

[Textbox](#)  
[Checkbox](#)  
[Dropdown List](#)  
[Listbox](#)

[Checkbox List](#)  
[Label](#)  
[File Upload](#)  
[Submit Button](#)  
[Hidden Field](#)  
[HTML Editor](#)

[Home](#) > [Working With Fields](#) > [Adding Fields](#) > [Label](#)

## Label

The **Label Control** can be used to insert any Html content including pictures to a Table Cell in the Form. The most likely use will be to introduce a section of the form or to provide instructions to the user which involve more than a one to three word prompt entered in the Label textbox on the "Input Control" form.

When used to provide instructions it is common to hide the label prompt for the target field control so as not to duplicate the field name.

We will use the **Label Control** to introduce "Interests", and "How Heard" fields. This will require us to insert a row to make room for the Label.

[Adding a Label Video](#)

### See also

[Textbox](#)  
[Checkbox](#)  
[Dropdown List](#)  
[Listbox](#)  
[Checkbox List](#)  
[Radiobutton List](#)  
[File Upload](#)  
[Submit Button](#)  
[Hidden Field](#)  
[HTML Editor](#)

[Home](#) > [Working With Fields](#) > [Adding Fields](#) > [File Upload](#)

## File Upload

The **File Upload Field** allows end users to upload files to a Server.

### File Storage

Files can be stored in the **Database** or in the **File System**. Even if the Form is not set to save the form submissions to the database the uploaded files are still stored in the database if the individual File Upload Field is set to that storage media. Multiple File Upload Fields can store the file in different locations, depending on the setting of the individual field.

### File Size\*

File size can be controlled by each field on the form, the default is 1000KB. In a form with multiple File Upload Fields, the module uses **Transactional Processing to Rollback** the entire form submission if any part of the submission fails, which would happen if a file were larger than the size set in the File Upload Field settings.

\*The default maximum size for all content uploaded to a server is set to 4096 KB by ASP.NET See Apendix A for instructions on modifying this value.

### File Types

Any File Type which the DNN Host has allowed can be uploaded using the File Upload Field. If a file type is not listed, it will be necessary to contact the DNN Host to add that extension to the allowable extensions for the DNN installation.

### Downloads

If emailing is enabled for the form a link to the uploaded file will be in the body of the email, also the file will be available for download from the Reports area of the Form Master Edit menu. By default Form Master requires that a user be in an **Authorized Edit Role** in order to download a file from the server, but Form Master can be set to allow anonymous file downloads if needed.

### Validation:

Form Master automatically adds a **Regular Expression Validator** for the File Upload Field to ensure that the File Type is one of the selected types. A Required Field Validator can be added when the end user must supply a file in order to submit the form.

[Adding a File Upload Video](#)

### See also

[Textbox](#)  
[Checkbox](#)

[Dropdown List](#)  
[Listbox](#)  
[Checkbox List](#)  
[Radiobutton List](#)  
[Label](#)  
[Submit Button](#)  
[Hidden Field](#)  
[HTML Editor](#)

[Home](#) > [Working With Fields](#) > [Adding Fields](#) > [Submit Button](#)

## Submit Button

The **Submit Button** comes in three types:

- Standard Button
- Link Button
- Image Button

Form Master automatically adds a Link Button to the form if no Submit Button is added by the form designer.

Form Master supports **Multiple Submit Buttons** on a single form, and each Button can be a different type. Multiple Submit Buttons are useful for example on a large form where some fields are required and others are optional. Required fields can be grouped at the top of the form with a Submit Button below the group. Another Submit Button can be added at the bottom of the form after the optional fields.

[Adding a Submit Button Video](#)

### See also

[Textbox](#)  
[Checkbox](#)  
[Dropdown List](#)  
[Listbox](#)  
[Checkbox List](#)  
[Radiobutton List](#)  
[Label](#)  
[File Upload](#)  
[Hidden Field](#)  
[HTML Editor](#)

[Home](#) > [Working With Fields](#) > [Adding Fields](#) > [Hidden Field](#)

## Hidden Field

The **Hidden Field** has many uses on a Web form. It can be used to provide data to the Database, Email Recipient, or Process which the user does not or should not see.

Consider a scenario where the user will be redirected to an affiliate web site after the Form Submission. A requirement of course in these cases is an Affiliate ID in the URL Querystring. Using the Hidden Field, an Affiliate ID can be added to the form which the user will not see. When setting up this field, make sure that Querystring Parameter name is what is expected by the Affiliate and mark this field to be sent by **Url** in the **Send Value By Options** of the Input Control Page. In this scenario, it is also necessary to make sure the form redirects after submission in the **Settings** > [Other Settings](#) page. It is also common to exclude Hidden Fields used in this way from being stored or sent in the email using the **Exclude from Email & Data** option as shown below.

### Hidden Field Screen Capture



contain a BIO for a user.

[Adding an HTML Editor Video](#)

### See also

- [Textbox](#)
- [Checkbox](#)
- [Dropdown List](#)
- [Listbox](#)
- [Checkbox List](#)
- [Radiobutton List](#)
- [Label](#)
- [File Upload](#)
- [Submit Button](#)
- [Hidden Field](#)

[Home](#) > [Working With Fields](#) > [Calculated Fields](#)

## Calculated Fields

[Product Name] includes the ability to do calculations on the data your users supply in the form. Using an Excel type of syntax, [Product Name] allows you to perform complex calculations. Calculation can be done across pages in multi-page forms allowing you to create a shopping cart effect.

### See also

- [Field Settings](#)
- [Adding Fields](#)

---

Home > Cascading Dropdowns

## Cascading Dropdowns

Articles in this section



[Static Data](#)



[Dynamic Data](#)

### See also

- [Welcome](#)
- [What's New](#)
- [Installation](#)
- [Getting Started](#)
- [Form Settings](#)
- [Basic Functions](#)
- [Working With Fields](#)
- [Multi-Page Forms](#)
- [Validation](#)
- [Data Manager](#)
- [Formatting](#)
- [Reporting](#)
- [Messaging](#)
- [How to Videos](#)
- [Advanced Functions](#)

Home > Cascading Dropdowns > Static Data

## Static Data

To use Static Data in a Cascading Dropdown Set (CDS) is only slightly more difficult than adding items to a standard Dropdown List. For this example we will use an automobile selection involving three Dropdownlists.

**We will start with the Make: Dropdown.**

This list control is being used as a Cascading Parent for: Model

### LIST ITEMS

Text	Value				
<< Select >>	0	False	<input type="checkbox"/>	<input type="checkbox"/>	Edit Delete
Ford	Ford	False	<input type="checkbox"/>	<input type="checkbox"/>	Edit Delete
Chevy	Chevy	False	<input type="checkbox"/>	<input type="checkbox"/>	Edit Delete
Jeep	Jeep	False	<input type="checkbox"/>	<input type="checkbox"/>	Edit Delete
<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Add Delete All

Top level parent items serve as categories for the next child dropdown

### VALIDATORS

Validator Details			
Validator Type: RequiredFieldValidator	Error Message: A value is required		
Error Class: NormalRed	Initial Value: 0	Edit	Delete
Validator Type: RequiredFieldValidator		Add Validator	

Update Cancel

Next is the Model: Dropdown

Fill With States:  Fill List Items with States From

Fill With Countries:  Fill List Items with Countries

Auto Fill Part:   Text  Value

List Data To Save:   Text  Value

Cascading List:   Cascading

Parent List Control

Prompt Text

Not Found Text

Select Text:

This list control is being used as a Cascading Parent for: Edition

### LIST ITEMS

Category	Text	Value	Selected		
Ford	Pinto	Pinto	False	<input type="checkbox"/>	<input type="checkbox"/>
Ford	Mustang	Mustang	False	<input type="checkbox"/>	<input type="checkbox"/>
Chevy	Malibu	Malibu	False	<input type="checkbox"/>	<input type="checkbox"/>
Chevy	Cavilier	Cavilier	False	<input type="checkbox"/>	<input type="checkbox"/>
Jeep	Cherokee	Cherokee	False	<input type="checkbox"/>	<input type="checkbox"/>
Jeep	Wrangler	Wrangler	False	<input type="checkbox"/>	<input type="checkbox"/>
<< Select >>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	Add Delete All

Parent is Make

Define items for each category in the parent dropdown list.

Use this dropdown to select a category when creating a new item

### VALIDATORS

Validator Type: RequiredFieldValidator	Add Validator		
---	---------------	--	--

Update Cancel

Next is the Edition: Dropdown

Fill With States:  Fill List Items with States From

Fill With Countries:  Fill List Items with Countries

Auto Fill Part:   Text  Value

List Data To Save:   Text  Value

Cascading List:   Cascading

Parent List Control

Prompt Text

Not Found Text

Select Text:

This list control is being used as a Cascading Parent for: Year

**LIST ITEMS**

Category	Text	Value	Selected				
Pinto	Pyro	Pyro	False	<input type="checkbox"/>	<input type="checkbox"/>	Edit	Delete
Pinto	Special	Special	False	<input type="checkbox"/>	<input type="checkbox"/>	Edit	Delete
Mustang	Sport	Sport	False	<input type="checkbox"/>	<input type="checkbox"/>	Edit	Delete
Mustang	Elite	Elite	False	<input type="checkbox"/>	<input type="checkbox"/>	Edit	Delete
Malibu	Limited	Limited	False	<input type="checkbox"/>	<input type="checkbox"/>	Edit	Delete
Malibu	Standard	Standard	False	<input type="checkbox"/>	<input type="checkbox"/>	Edit	Delete
Cavilier	Budget	Budget	False	<input type="checkbox"/>	<input type="checkbox"/>	Edit	Delete
Cherokee	Chief	Chief	False	<input type="checkbox"/>	<input type="checkbox"/>	Edit	Delete
Cherokee	Limited Edition	Limited Edition	False	<input type="checkbox"/>	<input type="checkbox"/>	Edit	Delete
Wrangler	2-WD	2-WD	False	<input type="checkbox"/>	<input type="checkbox"/>	Edit	Delete
Wrangler	4-WD	4-WD	False	<input type="checkbox"/>	<input type="checkbox"/>	Edit	Delete
Pinto	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Add	Delete All

**VALIDATORS**

**Validator Details**

Validator Type:

Parent is Model

Define items for each category in the parent dropdown list.

See also [Dynamic Data](#)

Home > Cascading Dropdowns > Dynamic Data

## Dynamic Data

Using Dynamic Data for Cascading Dropdowns involves knowledge of Structured Query Language (SQL) and the use of [Field Query Options](#). A Cascading Dropdown Set (CDS) involves at a minimum two Drop Down lists with one being the parent and the other being the child. The child can also become a Parent if a third dropdown is added to the set. Technically there is no limit to the number of dropdowns which can be in a set, but for practical purposes the number should be limited to 4 to 5 for both the Server machine CPU cycles and for the usability of the end user.

In this example we will use the classic Country, Region relationship where the Country dropdown will be the parent and the Region will be the child. We will use values from the DotNetNuke supplied List table to populate our dropdowns.

### FIELD QUERY OPTIONS

About The Field Query

Connection String:

SQL Query:

Test Query

Query Tokens: Name:  Insert Token

Text Column:  Value Column:

### SEND VALUE BY OPTIONS

Fill With States:  Fill List Items with States From:

Fill With Countries:  Fill List Items with Countries

Auto Fill Part:   Text  Value

List Data To Save:   Text  Value

Cascading List:   Cascading

This list control is being used as a Cascading Parent for: Region

### LIST ITEMS

Text	Value	State
<< Select >>	0	Fail
<input type="text"/>	<input type="text"/>	<input type="checkbox"/>

Add Delete All

### VALIDATORS

Validator Details		
Validator Type: RequiredFieldValidator	Error Message: Please select a country	Edit Delete
Error Class: NormalRed	Initial Value: 0	
Validator Type: <input type="text" value="RequiredFieldValidator"/>		Add Validator

Update Cancel

This query gets all the countries from the Lists Table

We used EntryID as the Value because the child query will key off this value for it's query.

When another dropdown has chosen this dropdown as a parent, this message will be shown.

Add a prompt and set the value to 0 for validation.

Make sure user selects a Country

See notes below for item by item description for the Child Dropdown

**About The Field Query** ?

**Connection String:** ?

**Initial Cascading Query:** ?

**SQL Query:** ?

Test Query

Test Query

**Query Tokens:** ? Name: Prefix Insert Token

**Text Column:** ? Text **Value Column:** ? Value

---

+
**SEND VALUE BY OPTIONS**

**Fill With States:** ? Fill List Items with States From Afghanistan

**Fill With Countries:** ? Fill List Items with Countries

**Auto Fill Part:** ?  Text  Value

**List Data To Save:** ?  Text  Value

**Cascading List:** ?  Cascading

**Parent List Control:** ? Country

**Prompt Text:** ? << Select a Country First >>

**Not Found Text:** ? Regions not found for Selected Count

**Select Text:** ? << Select a Region >>

### LIST ITEMS

Category	Text	Value	Selected
<< Select >>	<input style="width: 100%;" type="text"/>	<input style="width: 100%;" type="text"/>	<input type="checkbox"/> <span style="float: right;">Add Delete All</span>

### VALIDATORS

Validator Details			
Validator Type: RequiredFieldValidator	Error Message: Region is required		
Error Class: NormalRed	Initial Value: 0	Edit	Delete
Validator Type: <span style="border: 1px solid #ccc; padding: 2px;">RequiredFieldValidator</span>		Add Validator	

Update Cancel

**Field Descriptions:**

- Initial Cascading Query:** Because of the ASP.NET security model all values that may be submitted when the user completes the form must be present in the child dropdown when the page is first loaded. On page load, JavaScript will be used to empty the child dropdowns and add the Prompt Text. It is VERY IMPORTANT that this requirement be strictly adhered to as users will not be able to submit the form if it is not. In this case we get all the Regions defined in the Lists Table.
- SQL Query:** In the List Table we have a Parent ID column which is used for form a hierarchical relationship in the data. Regions will have a ParentID which is the EntryID for a Country. In this example #CASCADE:DO\_NOT\_EDIT# is used as a **TOKEN** and will be replaced with the selected EntryID when the user makes a selection from the Country dropdown list. The #CASCADE:DO\_NOT\_EDIT# TOKEN can be found in the Query Tokens: dropdown and inserted into the Query by clicking on the "Insert Token" button.
- Cascading List:** Check this to put a Dropdown in a new or existing CDS.
- Parent List Control:** When Cascading List is checked use this Dropdown to select the Parent dropdown for this child dropdown
- Prompt Text:** Text to be shown to the end user which tell them they musts select a value from the Parent dropdown first.
- Not Found Text:** Text to be shown to the end user when a selection they make in the parent dropdown returns not results
- Select Text:** Text to be used to prompt the user to select a value from this dropdown. A "0" is automatically added as a value so it can be used by the required field validator as shown above.

[Dynamic Data Video](#)

See also  
[Static Data](#)

[Home](#) > [Multi-Page Forms](#)

## Multi-Page Forms

Form Master Responsive supports Multi-Page forms which work in a Wizard style allowing the user to move forward and backwards through each page. Multi-Page forms support the following features.

- Field validation for each page so user cannot continue to the next page until the current page is properly filled out.
- Field rules which support showing or hiding fields on the current or other pages based on data supplied by the user.
- Custom Next, Previous, and Finish buttons:
  - Button
  - Link Button
  - Image Button
- Wizard Steps optionally listed at the top of the form
- Wizard Step description optionally shown at the top of the form
- Optional per page form Heading for instructions/explanations
- CSS class name for Steps, Title, and Button
- Ability to map all pages to one single external table, or map each page to separate tables using the [Data Manager](#)

### See also

[Welcome](#)  
[What's New](#)  
[Installation](#)  
[Getting Started](#)  
[Form Settings](#)  
[Basic Functions](#)  
[Working With Fields](#)  
[Cascading Dropdowns](#)  
[Validation](#)  
[Data Manager](#)  
[Formatting](#)  
[Reporting](#)  
[Messaging](#)  
[How to Videos](#)  
[Advanced Functions](#)

[Home](#) > [Multi-Page Forms](#) > [Multi-Page Form Options](#)

## Multi-Page Form Options

Form Master Responsive Multi-Page forms support the following options.

### Form Wizard Options

Button Type:	<input type="text" value="Button"/>
Show Wizard Steps:	<input checked="" type="checkbox"/>
Start Button Text:	<input type="text" value="Start &gt;&gt;"/>
Next Button Text:	<input type="text" value="Next &gt;&gt;"/>
Previous Button Text:	<input type="text" value="&lt;&lt; Prev"/>
Finish Button Text:	<input type="text" value="Finish"/>
Step Text:	<input type="text" value="Step"/>
Step Class Name:	<input type="text" value="Head"/>
Steps Alignment:	<input type="text" value="Left"/>
Legend Class:	<input type="text" value="Head"/>
Legend Alignment:	<input type="text" value="Center"/>
Step Title Class Name:	<input type="text" value="Head"/>
Step Title Alignment:	<input type="text" value="Center"/>
Button Class:	<input type="text" value="CommandButton"/>

The Form Wizard options allow you to customize the buttons used in the steps and other formatting options.

- **Button Type:**
  - Button: Standard HTML button
  - LinkButton: Hyperlink style button
  - ImageButton: Allows for Uploading/Selecting a custom image for each button type
- **Show Wizard Steps:** When checked, shows the short page name text as steps at the top of the form
- **Start Button Text:** The text to be used for the Button or LinkButton text for the first step in the form wizard
- **Next Button Text:** The text to be used for the next Button or LinkButton on all intermediate steps in the form wizard
- **Previous Button Text:** The text to be used for the previous Button or LinkButton on all intermediate steps in the form wizard
- **Finish Button Text:** The text to be used for the Button or LinkButton text for the final step in the form wizard
- **Step Text:** Text for the Wizard Step Number above the Short Name. Leave blank to not use a Step Number.
- **Step Class Name:** The CSS Class name for the text shown at the top of a multi-page Form Wizard.
- **Steps Alignment:** Horizontal alignment for the Wizard Steps at the top of a Multi-Page Form.
- **Legend Class:** CSS Class for the page Short Name text. Note: Use C5SLLC\_hidden to hide this text.
- **Legend Alignment:** Legend Alignment for Multi-Page Short Name in the Form Wizard
- **Step Title Class Name:** The CSS Class name for the step Title shown at the top of a Multi-page Form Wizard.
- **Step Title Alignment:** Horizontal alignment for the Wizard Title at the top of a Multi-Page Form.
- **Button Class:** Class to be applied to the Form Wizard Buttons or LinkButtons

See also

[Working with Multi-Page Forms](#)

## Working with Multi-Page Forms

Creating a Multi-Page form is as simple as adding a second page. This instantly converts a single page form to a multi-page form.

**Designer Settings**

<b>Label Position:</b>	<input type="checkbox"/>	<input type="text" value="Above"/>
<b>Cols:</b>	<input type="checkbox"/>	<input type="text" value="1"/>
<b>Rows:</b>	<input type="checkbox"/>	<input type="text" value="1"/>
<b>Pages:</b>	<input type="checkbox"/>	<input type="text" value="Page 1"/>
<b>Page Short Name</b>	<input type="checkbox"/>	<input type="text" value="Page 1"/>
<b>Page Title</b>	<input type="checkbox"/>	<input type="text" value=""/>
<b>Show Title</b>	<input type="checkbox"/>	<input type="checkbox"/>

**Form Designer**

Next give your pages a Page Short Name which will be used for the Steps and Page Title which will be shown for each page. You can choose to show the Page Title or not by checking or unchecking the Show Title checkbox. This allows you to use a Page Heading using the [Form Heading Manager](#) where you can define a Heading for each page.

**Designer Settings**

<b>Label Position:</b>	<input type="checkbox"/>	<input type="text" value="Above"/>
<b>Cols:</b>	<input type="checkbox"/>	<input type="text" value="1"/>
<b>Rows:</b>	<input type="checkbox"/>	<input type="text" value="1"/>
<b>Pages:</b>	<input type="checkbox"/>	<input type="text" value="Page 1"/>
<b>Page Short Name</b>	<input type="checkbox"/>	<input type="text" value="Contact &amp; Question"/>
<b>Page Title</b>	<input type="checkbox"/>	<input type="text" value="Contact &amp; Question Information"/>
<b>Show Title</b>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

**Form Designer**

Once the pages are named the Page Short Name shows up in the Pages Dropdown List which allows you to navigate between, add New, Copy or Delete pages. You can set the Cols and Rows for each page independently and you can even show the labels Above or on the Left differently on each page.

**Designer Settings**

<b>Label Position:</b>	T	Above	▼	
<b>Cols:</b>	T	1	▼	
<b>Rows:</b>	T	5	▼	
<b>Pages:</b>	T	Contact & Question		
<b>Page Short Name</b>	T	Contact & Question		
<b>Page Title</b>	T	Address		
<b>Show Title</b>	T	<New Page...>		
		<Copy Page...>		
		<Delete Page...>		

**Form Designer**

Once the Multi-Page configuration is done you can work with each page just as if it were a single page.

Multi-Page forms do have some constraints as listed below:

- Custom Buttons: While Form Master Responsive allows you to place a custom submit button on a Page it is not recommended and will only serve to confuse the user as the button will not navigate through the Wizard Steps.
- File Upload controls will only work correctly when they are placed on the last page of a Multi-Page form as they lose their value when the user navigates through the Wizard
- The Captcha control will only be placed on the last page of a Multi-Page form.

[Working with Multi-Page Forms Video](#)

See also

[Multi-Page Form Options](#)

Home > Validation

## Validation

Articles in this section



Required Field  
Validator



Regular Expression  
Validator



Compare Validator



Checkbox List  
Validator



Validation Summary

See also

[Welcome](#)  
[What's New](#)  
[Installation](#)  
[Getting Started](#)  
[Form Settings](#)  
[Basic Functions](#)  
[Working With Fields](#)  
[Cascading Dropdowns](#)  
[Multi-Page Forms](#)  
[Data Manager](#)  
[Formatting](#)  
[Reporting](#)  
[Messaging](#)  
[How to Videos](#)  
[Advanced Functions](#)

Home > Validation > Required Field Validator

## Required Field Validator

The **Required Field Validator** is used to ensure that a user enters a value in a Textbox, or selects a value from one of the **"List Item"** fields.

The Field Control types which support the Required Field Validator are:

- Textbox
- Dropdown List
- Listbox
- RadioButton List
- File Upload

The Required Field Validator has three attributes which can be modified when adding this Validator to a field:

- **Error CSS Class:** This is the CSS class name applied to the Error Message when displayed in the Browser. The DotNetNuke® class name "NormalRed", is used by default.
- **Error Message:** This is the message to be displayed in the Browser when the user input fails validation.
- **Initial Value:** The value entered here is used by the Required Field Validator to determine if the user supplied value is the same as the "Initial Value", if so then validation fails.

Using the Required Field Validator with the different field types is explained below:

### Adding a Required Field Validator to a Textbox or File Upload

Adding a Required Field Validator for a Textbox is straightforward, simply edit the field, click on "Add Validator" in the Validators Data Grid, modify the Error Messages, and click "Update". The Error Message is shown in the Designer.

### Adding a Required Field Validator to a Dropdown List

Adding a Required Field Validator to a Dropdown List requires some planning, as mentioned in Adding a Dropdown List, adding a prompt as the first item in the List Items data grid like "<< Select >>" for the Text part and "0" for the Value part gives us "0" to use as the **Initial Value**. If a user does not make a selection, validation will fail.

### Adding a Required Field Validator to a Listbox or RadioButton List

When an Initial Value is not used, adding a Required Field Validator to a Listbox or RadioButton list is the same as for the Textbox. When an Initial Value is used the steps are the same as for the Dropdown List.

This section will show how to require a user to agree to the Terms & Conditions in order to submit the form using a RadioButton List, but the steps would be the same for a Listbox. This section will also use a different form.

[Required Field Validator Presentation](#)

See also

[Regular Expression Validator](#)  
[Compare Validator](#)  
[Checkbox List Validator](#)  
[Validation Summary](#)

[Home](#) > [Validation](#) > [Regular Expression Validator](#)

## Regular Expression Validator

The **Regular Expression Validator** checks whether the value of an input control matches a pattern defined by a regular expression. This type of validation allows you to check for predictable sequences of characters, such as those in social security numbers, e-mail addresses, telephone numbers, and postal codes.

**The Regular Expression Validator does not require that a field contain a value, it only validates if a value is present.**

The use of Regular Expressions is beyond the scope of this User Guide, however there are many resources on the Internet to help learn how to create Regular Expressions for use in this Validator. Form Master provides 4 built-in expressions which are:

- E-Mail Address
- Internet URL
- U. S. Phone Number
- U. S. SSN
- U. S. or Canadian Postal Code

[Regular Expression Validator Presentation](#)

See also

[Required Field Validator](#)  
[Compare Validator](#)  
[Checkbox List Validator](#)  
[Validation Summary](#)

[Home](#) > [Validation](#) > [Compare Validator](#)

## Compare Validator

The **Compare Validator** is used to compare values in two different fields on the form, or compare the data entered to a specific type of data such as a Date.

**The Compare Validator does not require that a field contain a value, it only validates if a value is present.**

For example: Many forms ask the user to enter their email address then confirm the email address by entering it a second time to help the user avoid typographical errors. The Compare Validator is just the tool to use to solve this. The Compare Validator has four new attributes in the Validators Data Grid

- **Operator:** The values available here are:
  - **DataTypeCheck:** Use this when comparing the user entered value with a specific Data Type such as a Date
  - **Equal:** =; Compare value with another form field or with the Value to Compare. The remaining Operators share the same function.
  - **GreaterThan:** >
  - **GreaterThanEqual:** >=
  - **LessThan:** <
  - **LessThanEqual:** <=
  - **NotEqual:** <> or !=
- **Value to Compare:** When "Control to Compare" is "<< None >>", works with the Operator and Data Type to compare the user entered value.
- **Control to Compare:** Sets which form Field to Compare this Field to. Works with the Operator
- **Data Type:** When using the "DataTypeCheck" Operator forces the user to enter the proper type of data such as a Date or Whole Number

When the "Control to Compare" is not set to "<< None >>", the field will not be part of the data sent by email or saved to the database.

In this example, we will use the Compare Validator to make sure the user enters their email address without typographical errors.

[Compare to Field Validation Presentation](#)

## See also

[Required Field Validator](#)  
[Regular Expression Validator](#)  
[Checkbox List Validator](#)  
[Validation Summary](#)

[Home](#) > [Validation](#) > [Checkbox List Validator](#)

## Checkbox List Validator

The Checkbox List Validator can be used to make sure that a user checks a certain number of checkboxes. For Example we have a list of ten Options and a user must choose three of those Options to continue. See the Setup Screen Shot below.

Validators			
Validator Details			
<b>Validator Type:</b> ChkListValidator	<b>Error Message:</b> Please Select Three Options	Update Cancel	Delete
<b>Error Css Class:</b> NormalRed	<b>Minumum Checked:</b> 3		
<b>Maximum Checked:</b> 3			

## See also

[Required Field Validator](#)  
[Regular Expression Validator](#)  
[Compare Validator](#)  
[Validation Summary](#)

[Home](#) > [Validation](#) > [Validation Summary](#)

## Validation Summary

The Validation Summary allows for locating all the Validation Error Messages in one place on the form or in an Alert Message Box or Both.

The Validation Summary is useful on Long forms where a Validation Error may have occurred on a field at the top of the form which has scrolled out of the user's view. Then this happens the user may think that the page is not working properly and leave.

Validation Summary Options Include:

- Placement Top, Bottom, or any row in between.
- Optional Header Text
- Simple List, Bullet List, or Paragraph style
- Message Box Alert
- Suppress the default Error Message under the Field
- Customizable Invalid Field Indicator when Suppress Error Message is set
- Left, Center, and Right cell alignment

[View the Video](#)

## See also

[Required Field Validator](#)  
[Regular Expression Validator](#)  
[Compare Validator](#)  
[Checkbox List Validator](#)



---

Home > Data Manager

## Data Manager

Articles in this section



[The Connection String](#)



[Map Existing Table](#)



[Table From Form](#)



[Form From Table](#)



[The Mapped Form](#)



[Edit Mappings](#)



[Multi Table Mapping](#)



[External Table Mask](#)

**Form Master Responsive** Stores data as XML Fragments by default in a single Table Column. While convenient and easy to manage, this make it difficult for other applications to access the collected data. Form Master Responsive introduces the ability to store data in Standard Microsoft® SQL Server® Tables. Form Master 2008 offers three methods to help Map Form Fields to Standard SQL Tables as noted below.

When Forms are Mapped to an SQL Table, Form Master Responsive will continue to store the data in XML Fragments as a backup. This gives the user the ability to restore the data to a SQL Table by re-mapping the form.

**Note:** Data Manager is available for Super Users and users in the Administrators Role only.

### See also

- [Welcome](#)
- [What's New](#)
- [Installation](#)
- [Getting Started](#)
- [Form Settings](#)
- [Basic Functions](#)
- [Working With Fields](#)
- [Cascading Dropdowns](#)
- [Multi-Page Forms](#)
- [Validation](#)
- [Formatting](#)
- [Reporting](#)
- [Messaging](#)
- [How to Videos](#)
- [Advanced Functions](#)

---

Home > Data Manager > The Connection String

## The Connection String

Form Master Responsive allows the user to specify a Connection String to a MS-SQL Database external to the Database running the DotNetNuke® Application. Once a Form is Mapped to a Table the Connection String cannot be changed. If the Connection String needs to change for some reason, it will be necessary to Re-Map the Fields.

## FORM DATA TABLE MANAGER

Use a Separate External Table for Each Page in Form

Name Information ▼

Current Mapping
  Map Existing Table
  Create Table From Form
  Create Form/Page From Table

Connection String: Server=myServer;Initial Catalog=myDataBase;User Id=myUserName;Passwors=myF

Save Connection String

## CURRENT MAPPING INFORMATION

The Form is not mapped to any table, and is using the default XML Data Store.

See [www.connectionstrings.com](http://www.connectionstrings.com) for a good connection string reference

See also

[Map Existing Table](#)  
[Table From Form](#)  
[Form From Table](#)  
[The Mapped Form](#)  
[Edit Mappings](#)  
[Multi Table Mapping](#)  
[External Table Mask](#)

Home > Data Manager > Map Existing Table

## Map Existing Table

**Form Master Responsive** can map Form Fields to an Existing SQL Table.

## FORM DATA TABLE MANAGER

Current Mapping
  Map Existing Table
  Create Table From Form
  Create Form/Page From Table

Connection String:  

Save Connection String

Data Source Table List: Contacts ▼

Map Existing Table ✔

Existing Table Fields: ✔

Column	Type	Length	Prec	Scale	Nullable	Form Field	
ContactID	int	4	10	0	no	Key	
Name	nvarchar	50			no		<a href="#">Map</a> <a href="#">UnMap</a>
Email	nvarchar	100			no		<a href="#">Map</a> <a href="#">UnMap</a>
Subject	nvarchar	100			yes		<a href="#">Map</a> <a href="#">UnMap</a>
Message	nvarchar	250			yes		<a href="#">Map</a> <a href="#">UnMap</a>
SendInfo	bit	1			yes		<a href="#">Map</a> <a href="#">UnMap</a>

Form Fields: ✔

- Email
- Message
- Name
- SendInfo
- Subject

Commit Operation
Cancel

**Form Fields and Controls:**

- **Connection String:** The Connection string to the Data Resource. Leave blank to use the Default System Connection String. If defining a Connection string, it must be set before any Mappings are started since Mappings are based on what Tables exist in the Database.
- **Data Source Table List:** The List of Tables from the Default DNN or User Specified Database
- **Existing Table Fields:** The Grid shows the Fields in the Selected Table and allows for Mapping to Form Master Created Fields.
- **Form Fields:** List of Fields Defined in the Form used for Mapping to External Table Fields.

**The Grid Columns:**

- **Column:** The name of the Table Column
- **Type:** The Data Type of the Table Column
- **Length:** The allowable Length of data for the Column. This is useful primarily for Text type data and does not apply to Integer types or Boolean values. A discussion of SQL Data Types is beyond the scope of this User Guide.
- **Prec & Scale:** These are for use with Numeric & Decimal Data Types
- **Nullable:** If a column is not Nullable then a Required Field Validator will need to be included on the Form.
- **Form Field:** Initially empty, will show Form Field names as they are mapped to the Table.

#### The Process:

1. Select the **"Map Existing Table"** Radio Button
2. Optionally Specify a Connection String
3. Select a Table to map to in the **"Data Source Table List"** dropdown list. **Note:** the selected table must have an Auto Increment Identity Column
4. Select a Form Field Radio Button from the **"Form Fields"** list
5. Click on the "Map" link on the Table Column to be mapped to the Form Field. If a mistake is made the UnMap link can be used to remove the mapping.
6. When all Form Fields have been mapped, click the **"Commit Operation"** button.

#### See also

[The Connection String](#)  
[Table From Form](#)  
[Form From Table](#)  
[The Mapped Form](#)  
[Edit Mappings](#)  
[Multi Table Mapping](#)  
[External Table Mask](#)

Home > Data Manager > Table From Form

## Table From Form

**Form Master Responsive** can create an External Table from the Fields defined in the Form.

### FORM DATA TABLE MANAGER

Current Mapping
  Map Existing Table
  Create Table From Form
  Create Form/Page From Table

Connection String: 
  
Save Connection String

New Table Name:

**Create Table From Form**

Form Field	Type	Length	Prec	Scale	Nullable	Ignore
Email	NVarchar	100	18	0	<input type="checkbox"/>	<input type="checkbox"/>
Message	NVarchar	250	18	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Name	NVarchar	50	18	0	<input type="checkbox"/>	<input type="checkbox"/>
SendInfo	Bit	1	18	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Subject	NVarchar	100	18	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>

#### Form Fields:

- **Connection String:** The Connection string to the Data Resource. Leave blank to use the Default System Connection String. If defining a Connection string, it must be set before any Mappings are started since Mappings are based on what Tables exist in the Database
- **New Table Name:** The name for the new Table which will be created in the Database

#### The Grid Columns:

- **Form Field:** This will be the name of the column in the SQL Table. This value is taken from the "Field Name" value in the [Common Options](#) section.
- **Type:** The Data Type for the SQL Table Field
- **Length:** The Length of the SQL Field. For Text fields, Char, NChar, VarChar, NVarchar the value can be set to the greatest expected value for the field. In cases where the amount of data is unknown the Text or NText Data Types should be used as will allow more data than can be sent in a browser.
- **Prec. & Scale:** Precision is the number of digits in a number. Scale: Scale is the number of digits to the right of the decimal point in a number. Precision and Scale only apply to Numeric and Decimal Data Types and will be ignored otherwise

- **Nullable:** If Checked the SQL Table will allow Null as a Value in the Column
- **Ignore:** If Checked the Column will not be created in the SQL Table and no Mapping will be created for it. This is useful for Hidden Fields or Conformation type Fields which do not need to be stored.

**The Process:**

1. Select the "Create Table From Form" Radio Button
2. Enter a Name for the New Table (a name with no spaces or special charaters is recommended)
3. Choose the appropriate Data Type and Length for the Form Field
4. Check the Nullable Checkbox if it there is no Required Field Validator on the Field
5. Check if the Field should be Ignored
6. Click the "Commit Operation" Button

## See also

[The Connection String](#)  
[Map Existing Table](#)  
[Form From Table](#)  
[The Mapped Form](#)  
[Edit Mappings](#)  
[Multi Table Mapping](#)  
[External Table Mask](#)

Home > Data Manager > Form From Table

## Form From Table

**Form Master Responsive** can create a Form from the Fields in an External Table. Using this function will overwrite any existing form for this Module.

### FORM DATA TABLE MANAGER

Current Mapping
  Map Existing Table
  Create Table From Form
  Create Form/Page From Table

Connection String: 
  
Save Connection String

Data Source Table List:

Create Form From Table

Column	Type	Length	Prec	Scale	Nullable	Field Type	Form Field	Required
ContactID	int	4	10	0	no	TextBox	<input type="text"/>	<input type="checkbox"/>
Name	nvarchar	50			no	TextBox	Name <input type="text"/>	<input type="checkbox"/>
Email	nvarchar	100			no	TextBox	Email <input type="text"/>	<input type="checkbox"/>
Subject	nvarchar	100			yes	TextBox	Subject <input type="text"/>	<input type="checkbox"/>
Message	nvarchar	250			yes	TextBox	Message <input type="text"/>	<input type="checkbox"/>
SendInfo	bit	1			yes	CheckBox	SendInfo <input type="checkbox"/>	<input type="checkbox"/>

**Form Fields:**

- **Connection String:** The Connection string to the Data Resource. Leave blank to use the Default System Connection String. If defining a Connection string, it must be set before any Mappings are started since Mappings are based on what Tables exist in the Database
- **Data Source Table List:** The List of Tables from the Default DNN or User Specified Database

**The Grid Columns:**

- **Column:** The name of the SQL Table Column
- **Type:** The Data Type of the Table Column
- **Length:** The allowable Length of data for the Table Column. This is useful primarily for Text type data and does not apply to Integer types or Boolean values and is shown for setting up field validation. A discussion of SQL Data Types is beyond the scope of this User Guide.
- **Prec & Scale:** These are for use with Numeric & Decimal Data Types, and is shown for setting up field validation
- **Nullable:** If a column is not Nullable then the Required Checkbox should be checked.
- **Form Field:** The Form Field name which will be specified in the Field Name textbox in [Common Options](#) (this name must not contain spaces or special characters)

**The Process:**

1. Select the "Create Form From Table" Radio Button
2. Select the Field Type
3. Enter a Form Field Name or use the supplied Name
4. Check if the Field will have a Required Field Validator added
5. Click the "Commit Operation" Button

## See also

[The Connection String](#)  
[Map Existing Table](#)  
[Table From Form](#)  
[The Mapped Form](#)  
[Edit Mappings](#)  
[Multi Table Mapping](#)  
[External Table Mask](#)

Home > Data Manager > The Mapped Form

## The Mapped Form

Once a Form is mapped to a Table **Form Master Responsive** will show the page below. Once a Form is Mapped all mapping options and Connection String options will be un-available. In a newly mapped form which previously had saved data the option to Import the XML data will be available. Once the data has been imported the "Import Data Stored as XML" button will not be available.

### FORM DATA TABLE MANAGER

Current Mapping
  Map Existing Table
  Create Table From Form
  Create Form/Page From Table

Connection String:

Save Connection String

### CURRENT MAPPING INFORMATION

The Form Fields are Mapped to Table: [Contacts]

Column	Type	Length	Prec	Scale	Nullable	Form Field
Name	nvarchar	50			No	Name
Email	nvarchar	100			No	Email
Subject	nvarchar	100			No	Subject
Message	nvarchar	250			No	Message
SendInfo	bit	1			No	SendInfo

The Mappings can be deleted by using the **"Delete Mappings"** button. The Mappings can easily be done again which allows for easy changes to the database schema.

To Import data from the default data store to the mapped table click the **"Import Data Stored as XML"** button.

To Export data from the mapped table to the default data store click the **"Access Existing Data from Table"** button. The reason why you would do this is to enable reporting or editing of data which existed in the mapped table before mapping to the form.

## FORM DATA TABLE MANAGER

Current Mapping
  Map Existing Table
  Create Table From Form
  Create Form/Page From Table

Connection String:

Save Connection String

**Data Transfer Results**

Table Name: Contacts Total Records: 1 Successful Records: 1 Failed Records: 0

No Errors Occured During Transfer of Data

No Errors after clicking "Access Existing Data from Table"

Form Master 2008 will report the results of the import and show any records which failed the import process.

## FORM DATA TABLE MANAGER

Current Mapping
  Map Existing Table
  Create Table From Form
  Create Form/Page From Table

Connection String:

Save Connection String

**Data Transfer Results**

Table Name: Contacts Total Records: 2 Successful Records: 0 Failed Records: 2

FormDataID	Error Message	
120	Specified Column: 'Name-Was-Changed' does not exist in the External Table Change the External Table Column Name to import, or Delete the Record	Delete
157	Specified Column: 'Name-Was-Changed' does not exist in the External Table Change the External Table Column Name to import, or Delete the Record	Delete

The Error returned above indicates that the name of the External Column was changed after the Mapping for this field was done.

## FORM DATA TABLE MANAGER

Current Mapping
  Map Existing Table
  Create Table From Form
  Create Form/Page From Table

Connection String:

Save Connection String

**Data Transfer Results**

Table Name: Contacts Total Records: 2 Successful Records: 0 Failed Records: 2

FormDataID	Error Message	
120	No Data can be found for External Column: 'Name' Mapped to Form Field: 'Name' It is likely that the Form Field has been recreated after data collection had already started. Delete The Data cannot be imported.	Delete
157	No Data can be found for External Column: 'Name' Mapped to Form Field: 'Name' It is likely that the Form Field has been recreated after data collection had already started. Delete The Data cannot be imported.	Delete

The Error returned above indicates that a Form Field exists in the data which is no longer present in the form and has no mapping.

## FORM DATA TABLE MANAGER

Current Mapping
  Map Existing Table
  Create Table From Form
  Create Form/Page From Table

**Connection String:** 
  
Save Connection String

**Data Transfer Results**

**Table Name:** Contacts **Total Records:** 2 **Successful Records:** 0 **Failed Records:** 2

FormDataID	Error Message	
120	Data Type of 'bit for Field 'Name' is invalid for External Table Please Delete Mappings and Remap the External Table	Delete
157	Data Type of 'bit for Field 'Name' is invalid for External Table Please Delete Mappings and Remap the External Table	Delete

The Error returned above indicates that an External Table Column Data Type was changed after the mapping was completed.

### See also

[The Connection String](#)  
[Map Existing Table](#)  
[Table From Form](#)  
[Form From Table](#)  
[Edit Mappings](#)  
[Multi Table Mapping](#)  
[External Table Mask](#)

Home > Data Manager > Edit Mappings

## Edit Mappings

Once a Form has been Mapped to a SQL Table, the mappings can be edited. This may become necessary for example if a new field is added to the SQL Table and a new Form Field is created to collect the new data. In cases like this either the new SQL Table Field will need to allow Nulls or the new column must be populated with default values.

## FORM DATA TABLE MANAGER

Current Mapping
  Map Existing Table
  Create Table From Form
  Create Form/Page From Table

**Connection String:** 
  
Save Connection String

**Data Source Table List:**

**Map Existing Table**

**Existing Table Fields:**

Column	Type	Length	Prec	Scale	Nullable	Form Field		
ContactID	int	4	10	0	no	Key		
Name	nvarchar	50			no	Name	<a href="#">Map</a>	<a href="#">UnMap</a>
Email	nvarchar	100			no	Email	<a href="#">Map</a>	<a href="#">UnMap</a>
Subject	nvarchar	100			yes	Subject	<a href="#">Map</a>	<a href="#">UnMap</a>
Message	nvarchar	250			yes	Message	<a href="#">Map</a>	<a href="#">UnMap</a>
SendInfo	bit	1			yes	SendInfo	<a href="#">Map</a>	<a href="#">UnMap</a>

**Form Fields:**

Email  
 Message  
 Name  
 SendInfo  
 Subject

## See also

[The Connection String](#)  
[Map Existing Table](#)  
[Table From Form](#)  
[Form From Table](#)  
[The Mapped Form](#)  
[Multi Table Mapping](#)  
[External Table Mask](#)

Home > Data Manager > Multi Table Mapping

## Multi Table Mapping

When working with Multi-Page Forms you can map each page to a separate external table, however each table must reside in the same database. When multiple tables are mapped, the Primary Key for the first table will be used as a Foreign Key for the 2.. n table mappings so Joins can be used in Select statements when retrieving the data.

### FORM DATA TABLE MANAGER

Use a Separate External Table for Each Page in Form

Contact & Question ▾

Current Mapping
  Map Existing Table
  Create Table From Form
  Create Form/Page From Table

Connection String:

[Save Connection String](#)

### CURRENT MAPPING INFORMATION

The Form Fields are Mapped to Table: [Contacts]

Column	Type	Length	Prec	Scale	Nullable	Form Field
Name	nvarchar	50			No	Name
Email	nvarchar	100			No	Email
Subject	nvarchar	100			No	Subject
Message	nvarchar	250			No	Message
SendInfo	bit	1			No	SendInfo

[Delete Mappings](#)
[Edit Mappings](#)

A properly mapped multi-page form

 Not all Form Pages are Mapped to External Tables. Selected Options Require that all Pages be mapped.

[Form](#) [Settings](#) [Files](#) [Messaging](#) [Report](#) [Help](#) [Exit](#)

## FORM DATA TABLE MANAGER

Use a Separate External Table for Each Page in Form

Contact & Question ▾

Current Mapping  Map Existing Table  Create Table From Form  Create Form/Page From Table

Connection String: 

Save Connection String

### CURRENT MAPPING INFORMATION

The Form Fields are Mapped to Table: [Contacts]

Column	Type	Length	Prec	Scale	Nullable	Form Field
Name	nvarchar	50			No	Name
Email	nvarchar	100			No	Email
Subject	nvarchar	100			No	Subject
Message	nvarchar	250			No	Message
SendInfo	bit	1			No	SendInfo

Delete Mappings

Edit Mappings

When one of the pages in your multi-page form are not mapped a warning is shown until all pages are mapped.

See also

[The Connection String](#)  
[Map Existing Table](#)  
[Table From Form](#)  
[Form From Table](#)  
[The Mapped Form](#)  
[Edit Mappings](#)  
[External Table Mask](#)

[Home](#) > [Data Manager](#) > [External Table Mask](#)

## External Table Mask

For Security Reasons **Form Master Responsive** creates an External Table Mask List in Host > Lists which is used to Mask Tables in the List from the User during Table Mapping Operations in the Default Database. Only the Host Role can manage this list which is partially shown below.

List Name:  FormMasterExtTableMask	
Total:  102 entries	
<a href="#">Add Entry</a>	<a href="#">Delete List</a>
Text	Value
  Affiliates	Affiliates
  AnonymousUsers	AnonymousUsers
  aspnet_Applications	aspnet_Applications
  aspnet_Membership	aspnet_Membership

		aspnet_Profile	aspnet_Profile
		aspnet_Roles	aspnet_Roles
		aspnet_SchemaVersions	aspnet_SchemaVersions
		aspnet_Users	aspnet_Users
		aspnet_UsersInRoles	aspnet_UsersInRoles
		Assemblies	Assemblies
		Authentication	Authentication
		Banners	Banners
		C5SLLC_Form	C5SLLC_Form
		C5SLLC_FormData	C5SLLC_FormData
		C5SLLC_FormFieldRules	C5SLLC_FormFieldRules
		C5SLLC_FormFileData	C5SLLC_FormFileData
		C5SLLC_FormMapping	C5SLLC_FormMapping
		C5SLLC_FormPage	C5SLLC_FormPage
		C5SLLC_Global	C5SLLC_Global
		Classification	Classification
		ContentItems	ContentItems
		ContentItems_MetaData	ContentItems_MetaData
		ContentItems_Tags	ContentItems_Tags
		ContentTypes	ContentTypes
		Dashboard_Controls	Dashboard_Controls
		DesktopModulePermission	DesktopModulePermission
		DesktopModules	DesktopModules
		EventLog	EventLog
		EventLogConfig	EventLogConfig
		EventLogTypes	EventLogTypes
		EventQueue	EventQueue
		Files	Files
		FolderMappings	FolderMappings
		FolderMappingsSettings	FolderMappingsSettings
		FolderPermission	FolderPermission
		Folders	Folders
		HostSettings	HostSettings
		HtmlText	HtmlText
		HtmlTextLog	HtmlTextLog
		HtmlTextUsers	HtmlTextUsers
		LanguagePacks	LanguagePacks
		Languages	Languages

		Links	Links
		Lists	Lists
		Messaging_Messages	Messaging_Messages
		MetaData	MetaData
		ModuleControls	ModuleControls
		ModuleDefinitions	ModuleDefinitions
		ModulePermission	ModulePermission
		Modules	Modules
		ModuleSettings	ModuleSettings
		Packages	Packages
		PackageTypes	PackageTypes
		Permission	Permission
		PortalAlias	PortalAlias
		PortalDesktopModules	PortalDesktopModules
		PortalLanguages	PortalLanguages
		PortalLocalization	PortalLocalization
		Portals	Portals
		PortalSettings	PortalSettings
		Profile	Profile
		ProfilePropertyDefinition	ProfilePropertyDefinition
		RoleGroups	RoleGroups
		Roles	Roles
		Schedule	Schedule
		ScheduleHistory	ScheduleHistory
		ScheduleItemSettings	ScheduleItemSettings
		SearchCommonWords	SearchCommonWords
		SearchIndexer	SearchIndexer
		SearchItem	SearchItem
		SearchItemWord	SearchItemWord
		SearchItemWordPosition	SearchItemWordPosition
		SearchWord	SearchWord
		SiteLog	SiteLog
		SkinControls	SkinControls
		SkinPackages	SkinPackages
		Skins	Skins
		SystemMessages	SystemMessages
		TabModules	TabModules
		TabModuleSettings	TabModuleSettings

		TabPermission	TabPermission
		Tabs	Tabs
		TabSettings	TabSettings
		Taxonomy_ScopeTypes	Taxonomy_ScopeTypes
		Taxonomy_Terms	Taxonomy_Terms
		Taxonomy_Vocabularies	Taxonomy_Vocabularies
		Taxonomy_VocabularyTypes	Taxonomy_VocabularyTypes
		UrlLog	UrlLog
		Urls	Urls
		UrlTracking	UrlTracking
		UserAuthentication	UserAuthentication
		UserPortals	UserPortals
		UserProfile	UserProfile
		UserRoles	UserRoles
		Users	Users
		UsersOnline	UsersOnline
		VendorClassification	VendorClassification
		Vendors	Vendors
		Version	Version
		WebServers	WebServers
		Workflow	Workflow
		WorkflowStates	WorkflowStates

### See also

[The Connection String](#)  
[Map Existing Table](#)  
[Table From Form](#)  
[Form From Table](#)  
[The Mapped Form](#)  
[Edit Mappings](#)  
[Multi Table Mapping](#)

Home > Formatting

# Formatting

Articles in this section



[Form Formatting](#)



[Email Formatting](#)

## See also

[Welcome](#)  
[What's New](#)  
[Installation](#)  
[Getting Started](#)  
[Form Settings](#)  
[Basic Functions](#)  
[Working With Fields](#)  
[Cascading Dropdowns](#)  
[Multi-Page Forms](#)  
[Validation](#)  
[Data Manager](#)  
[Reporting](#)  
[Messaging](#)  
[How to Videos](#)  
[Advanced Functions](#)

Home > Formatting > Form Formatting

# Form Formatting

**Formatting** the form provides visual cues which draw attention to different sections of the form, this improves the **User Interface** by helping the user perform the task more efficiently, and helps deliver a better user experience..

Applying formatting to Form Master Responsive is done using **Cascading Style Sheets (CSS)**, and applying **Class Name Rules** to elements of the form. The logical place for the Style Rules is in Portal.css since this Stylesheet is specific to each Portal, but the Rules can be in any Stylesheet which gets loaded with the Form.

Class names can be applied to the elements below:

- Outer Table
- Inner Table
- Individual Table Cell within the Inner Table
- Labels for each individual fields in the form
- Individual fields within the form

The presentations below will show where and how the Class Names can be applied, and finally, the Class Name definitions.

[Formatting the Form Presentation](#)

#### The Class Definitions

```
.mainReport
{
    border: black 1px solid;
    border-left: black 1px solid;
    width: 600px;
    border-collapse: collapse;
    background:ivory;
}
.frmReport
{
    border: black 1px solid;
    width: 500px;
    border-collapse: collapse;
}
.frmPadding
{
    padding:3px 3px 3px 3px;
}
.addressInfo
{
    background:#eaeaea;
    padding:3px 0px 6px 6px;
}
.frmSeparator
{
    border-bottom:1px black solid;
    height:10px;
}
.widgetHeading
{
    background:brown;
    padding:3px 3px 3px 6px;
}
.chkLstShip
{
    background:wheat;
    width:250px;
    padding:3px;
}
.chkLstMake
{
    background:silver;
    width:250px;
    padding:3px;
}
```

See also

[Email Formatting](#)

[Home](#) > [Formatting](#) > [Email Formatting](#)

## Email Formatting

Articles in this section



[XSL Templates](#)



[Tokenized Templates](#)



[Custom Template](#)

---

Email can be sent in either HTML or Text Formatting. When sent in HTML format Form Master 2008 sends both the HTML and Text in a Multipart Alternative Email Message (Auto Responder messages).

See also

[Form Formatting](#)

[Home](#) > [Formatting](#) > [Email Formatting](#) > [XSL Templates](#)

## XSL Templates

### XSL Email Templates

Form Master Responsive offers two Template styles for formatting the email message which was previously covered in the **Other Settings** topic. This section will show how the XSL templates work.

Customizing the XSL templates will not be shown as this is beyond the scope of this User Guide

[XSL Email Templates Presentation](#)

See also

[Tokenized Templates](#)

[Custom Template](#)

[Home](#) > [Formatting](#) > [Email Formatting](#) > [Tokenized Templates](#)

## Tokenized Templates

Form Master Responsive offers two Template styles for formatting the email message which was previously covered in the **Other Settings** topic.

This section will show how the Tokenized templates work and how to use them to modify the body of the email sent from **Form Master Responsive**.

[Tokenized Email Templates Presentation](#)

See also

[XSL Templates](#)

[Custom Template](#)

[Home](#) > [Formatting](#) > [Email Formatting](#) > [Custom Template](#)

## Custom Template

By using Field [\[TOKENS\]](#) and the Custom Template, it is possible to completely customize the email which is sent

to the email addresses in the **SendTo:** field of the **Settings** > [Form Settings](#) page

**Email Custom Template**

Editor:  Basic Text Box  Rich Text Editor

ABC [Icons] Link to Portal Page... [Icons]

**B** *I* U abc [Icons] Paragraph Style Apply CSS Cl... [Icons]



A User has submitted a question on the Contact Form. Please find the information listed below.

Name:	[NAME]
Email:	[EMAIL]
Subject:	[SUBJECT]
Message:	[MESSAGE]

Design <> HTML Preview Words: 24 Characters: 149

Update

**Defined Tokens**

[NAME]  
[EMAIL]  
[SUBJECT]  
[MESSAGE]

- The body of the message can contain any valid HTML
- To Insert the user supplied data from the form just use any of the previously defined Field **[TOKENS]** from the Defined Tokens column on the right.
- When the Form is submitted, Form Master Responsive will replace the **[TOKENS]** with the information provided by the user.

See also

[XSL Templates](#)

[Tokenized Templates](#)

Home &gt; Reporting

# Reporting

## Form Report

From Date:  To Date:

[View](#)
[Delete](#)
[Excel®](#)
[Excel® 2007+](#)
[CSV](#)
[PDF](#)
[Word®](#)
[Tab](#)

	Form Data ID	Submit Date	First Name	Last Name	Email	Address	City	State	Total	Postal Code	Birth Date
Delete	Edit	30	10/20/2013 5:37 PM	John	Smith	jwsmith@somedomain.com	555 Any St	Any Town	California	12345	
Delete	Edit	1063	3/20/2014 6:24 AM	John	Locke	jwsmith@somedomain.com	555 Any St	Any Town	California	12345	2/11/2014 12:00:00 AM
			2/3/2014 7:37				555 Any	Any			

When "Save Data to Database" is checked in the "Form Options", reports will be available for viewing, editing, downloading, and of course deleting. Reporting and deleting is done based on a user specified date range, the default is a range covering the past 30 days from the current date.

### Viewing Reports

If files have been uploaded, links are available in the form to download the files. Each record can be edited, and even uploaded files can be replaced or added to records where files have not been uploaded.

Records can be deleted individually or by date range. Any uploaded files associated with any deleted record are deleted from the database or from the file system.

### Downloading Reports

Download of data is available in four formats:

- **Excel®**: Outputs HTML for earlier versions of Excel.
- **Excel 2007+:** Outputs XML for Excel 2007 and later.
- **CSV**: Comma Separated Values: This is a standard format for sharing information between disparate databases. The fields are separated by commas using double quotes as field delimiters.
- **PDF**: Adobe Acrobat® Portable Document Format.
- **Word®**: Microsoft Word document format.
- **Tab Delimited**: Again another standard format. Fields are separated by Tabs with no field delimiters.

### See also

[Welcome](#)  
[What's New](#)  
[Installation](#)  
[Getting Started](#)  
[Form Settings](#)  
[Basic Functions](#)  
[Working With Fields](#)  
[Cascading Dropdowns](#)  
[Multi-Page Forms](#)  
[Validation](#)  
[Data Manager](#)  
[Formatting](#)  
[Messaging](#)  
[How to Videos](#)  
[Advanced Functions](#)

[Home](#) > [Messaging](#)

## Messaging

Articles in this section



[Form Heading](#)



[Form Thank You](#)



[Auto Responder](#)

### See also

- [Welcome](#)
- [What's New](#)
- [Installation](#)
- [Getting Started](#)
- [Form Settings](#)
- [Basic Functions](#)
- [Working With Fields](#)
- [Cascading Dropdowns](#)
- [Multi-Page Forms](#)
- [Validation](#)
- [Data Manager](#)
- [Formatting](#)
- [Reporting](#)
- [How to Videos](#)
- [Advanced Functions](#)

[Home](#) > [Messaging](#) > [Form Heading](#)

## Form Heading

The **Form Heading** allow for the insertion of Html content at the top of the form, in the first row of the "Outer Table" element. For Multi-Page forms a different heading can be provided for each page.

The default text used to provide instructions when the Module is first placed on the page, should either be deleted or replaced by custom content for the current form



**Thank You Message** 

Note: Form Must be set to "Save Data to Database" in order to use Tokens

Editor:  Basic Text Box  Rich Text Editor

ABC         Link to Portal Page...                                     



Home > How to Videos

## How to Videos

This section contains a growing collection of Form Master Responsive videos showing how to do basic and advanced configuration and setup. [Subscribe to the Form Master Responsive Channel](#) so you'll know when a new video is published.

**Note:** The Videos are best viewed Full Screen.

Articles in this section



[Getting Started](#)



[Fields](#)



[Cascading  
Dropdowns](#)



[Multi-Page Forms](#)



[Updatable Forms](#)

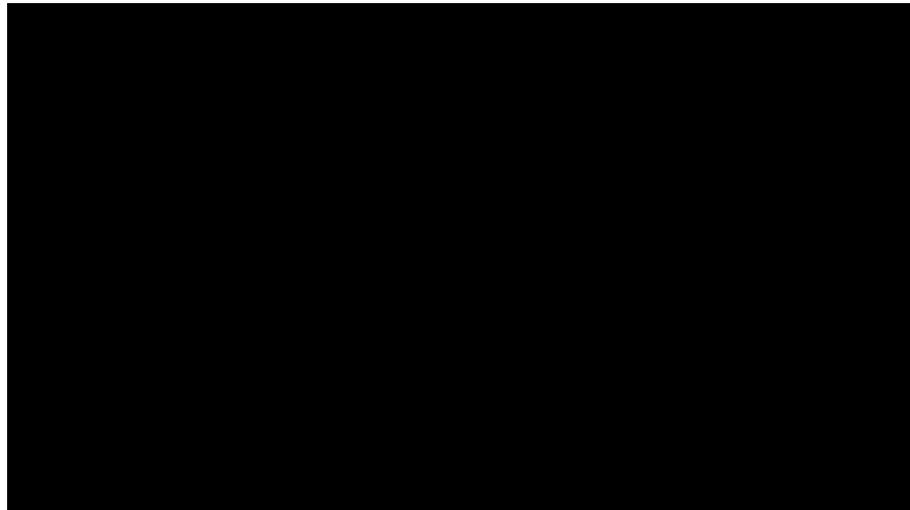
### See also

- [Welcome](#)
- [What's New](#)
- [Installation](#)
- [Getting Started](#)
- [Form Settings](#)
- [Basic Functions](#)
- [Working With Fields](#)
- [Cascading Dropdowns](#)
- [Multi-Page Forms](#)
- [Validation](#)
- [Data Manager](#)
- [Formatting](#)
- [Reporting](#)
- [Messaging](#)
- [Advanced Functions](#)

Home > How to Videos > Getting Started

## Getting Started

**Shows adding a Contact Us form from placing the module on the page to the completed form in 4 minutes**



[View Video in a new browser window](#)

See also

- [Fields](#)
- [Cascading Dropdowns](#)
- [Multi-Page Forms](#)
- [Updatable Forms](#)

[Home](#) > [How to Videos](#) > [Fields](#)

## Fields

This section contains Howto Videos on working with Form Master Responsive Fields

### Articles in this section



[Adding a Textbox](#)



[Adding a Checkbox](#)



[Adding a Listbox](#)



[Adding a Checkbox List](#)



[Adding a Radiobutton List](#)



[Adding a Label](#)



[Adding a File Upload](#)



[Adding a Submit Button](#)



[Adding a Hidden Field](#)



[Adding an HTML Editor](#)

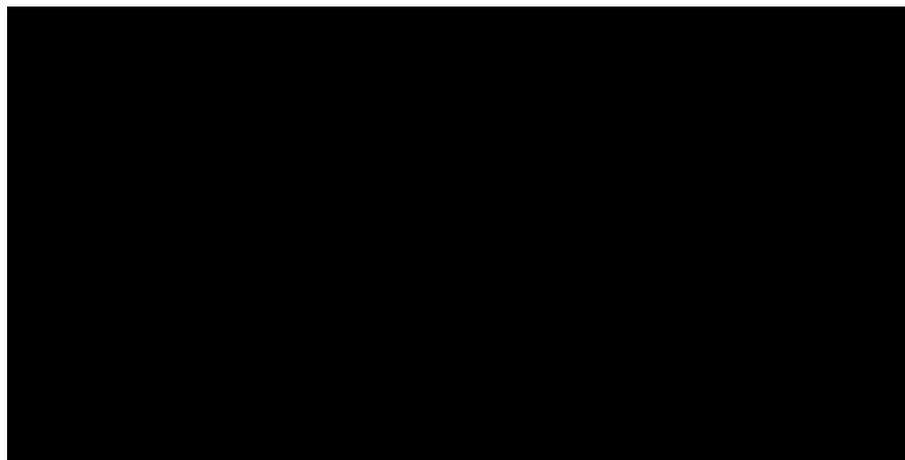
See also

- [Getting Started](#)
- [Cascading Dropdowns](#)
- [Multi-Page Forms](#)
- [Updatable Forms](#)

[Home](#) > [How to Videos](#) > [Fields](#) > [Adding a Textbox](#)

## Adding a Textbox

**Shows how to add a Textbox and work with the Rows in a form.**





[View Video in a new browser window](#)

See also

- [Adding a Checkbox](#)
- [Adding a Listbox](#)
- [Adding a Checkbox List](#)
- [Adding a Radiobutton List](#)
- [Adding a Label](#)
- [Adding a File Upload](#)
- [Adding a Submit Button](#)
- [Adding a Hidden Field](#)
- [Adding an HTML Editor](#)

[Home](#) > [How to Videos](#) > [Fields](#) > [Adding a Checkbox](#)

## Adding a Checkbox

**Shows how to add a single Checkbox.**



[View Video in a new browser window](#)

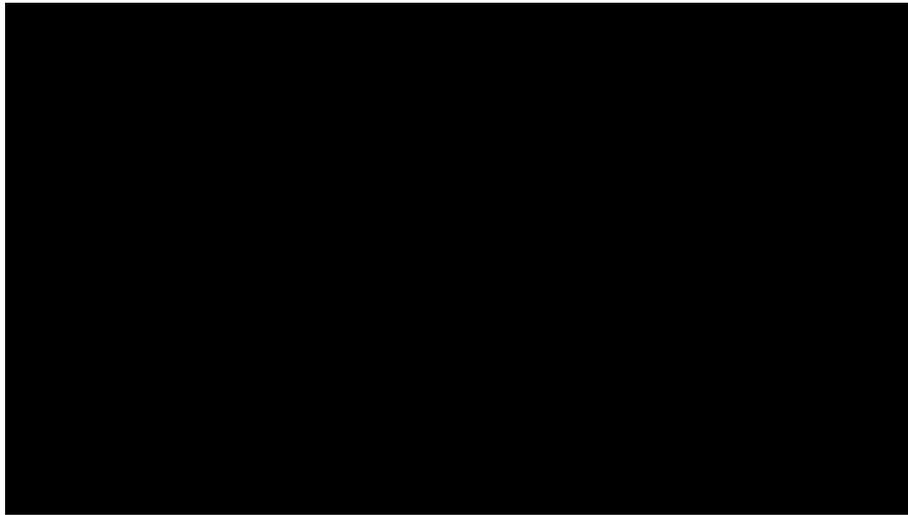
See also

- [Adding a Textbox](#)
- [Adding a Listbox](#)
- [Adding a Checkbox List](#)
- [Adding a Radiobutton List](#)
- [Adding a Label](#)
- [Adding a File Upload](#)
- [Adding a Submit Button](#)
- [Adding a Hidden Field](#)
- [Adding an HTML Editor](#)

[Home](#) > [How to Videos](#) > [Fields](#) > [Adding a Listbox](#)

## Adding a Listbox

Shows how to add a single Checkbox.



[View Video in a new browser window](#)

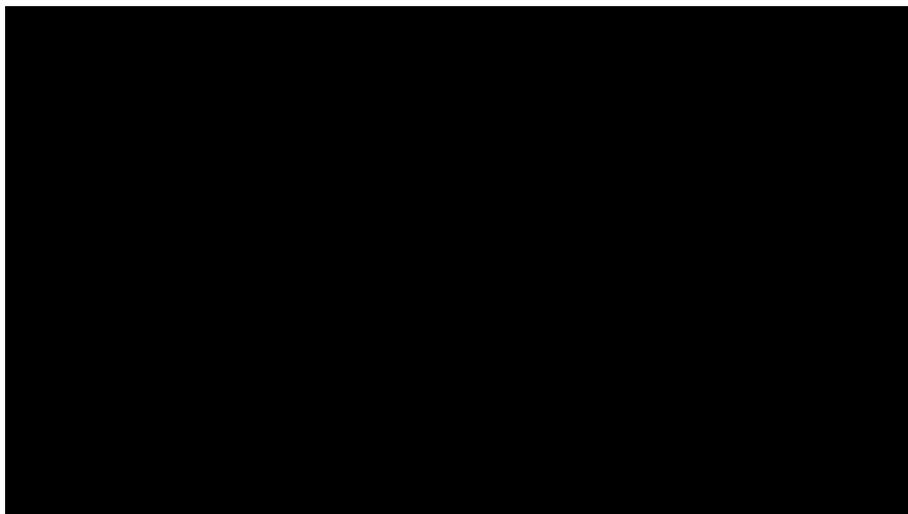
#### See also

- [Adding a Textbox](#)
- [Adding a Checkbox](#)
- [Adding a Checkbox List](#)
- [Adding a Radiobutton List](#)
- [Adding a Label](#)
- [Adding a File Upload](#)
- [Adding a Submit Button](#)
- [Adding a Hidden Field](#)
- [Adding an HTML Editor](#)

[Home](#) > [How to Videos](#) > [Fields](#) > [Adding a Checkbox List](#)

## Adding a Checkbox List

Shows adding a **Checkbox List** to the form. Also shows using **Column Span** for a more visually appealing for layout.



[View Video in a new browser window](#)

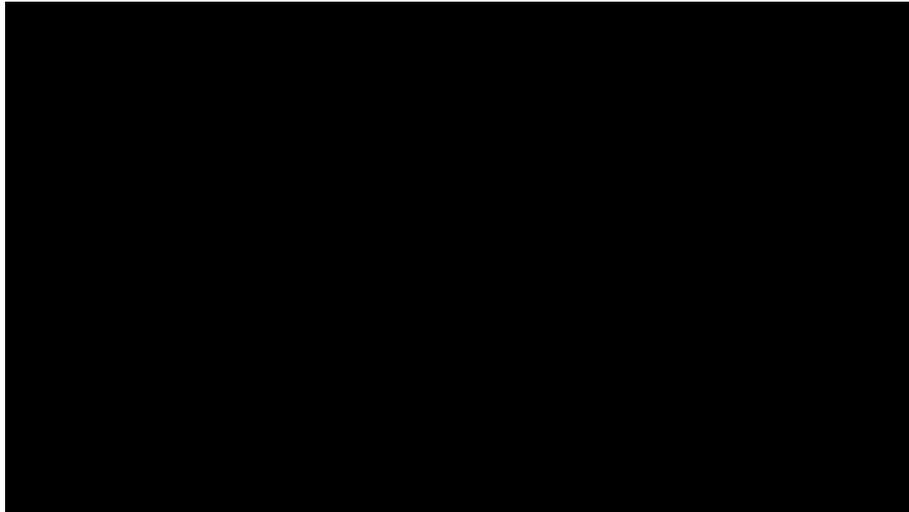
See also

- [Adding a Textbox](#)
- [Adding a Checkbox](#)
- [Adding a Listbox](#)
- [Adding a Radiobutton List](#)
- [Adding a Label](#)
- [Adding a File Upload](#)
- [Adding a Submit Button](#)
- [Adding a Hidden Field](#)
- [Adding an HTML Editor](#)

[Home](#) > [How to Videos](#) > [Fields](#) > [Adding a Radiobutton List](#)

## Adding a Radiobutton List

**Shows adding a Radiobutton List to the form. Also shows using the Row Manager to insert a row for the new field.**



[View Video in a new browser window](#)

See also

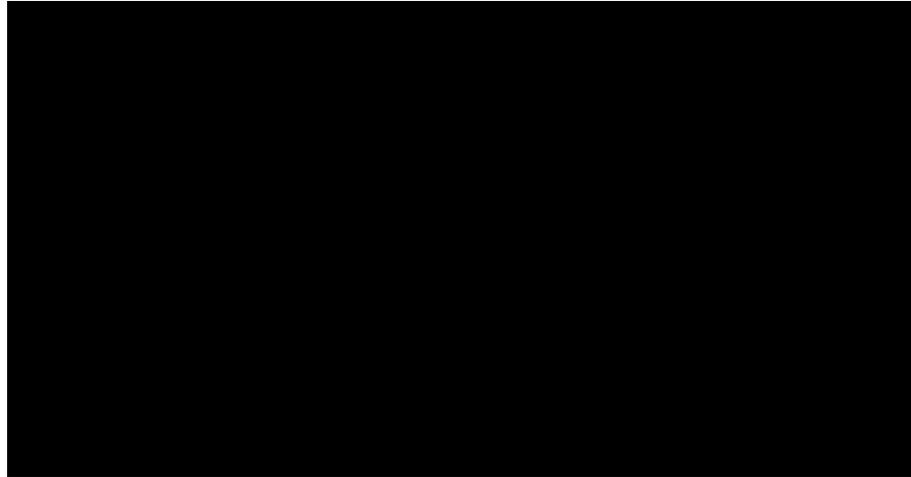
- [Adding a Textbox](#)
- [Adding a Checkbox](#)
- [Adding a Listbox](#)
- [Adding a Checkbox List](#)
- [Adding a Label](#)
- [Adding a File Upload](#)
- [Adding a Submit Button](#)
- [Adding a Hidden Field](#)
- [Adding an HTML Editor](#)

[Home](#) > [How to Videos](#) > [Fields](#) > [Adding a Label](#)

## Adding a Label

**Shows adding a Radiobutton List to the form. Also shows using the Row Manager to insert a row for the new field.**





[View Video in a new browser window](#)

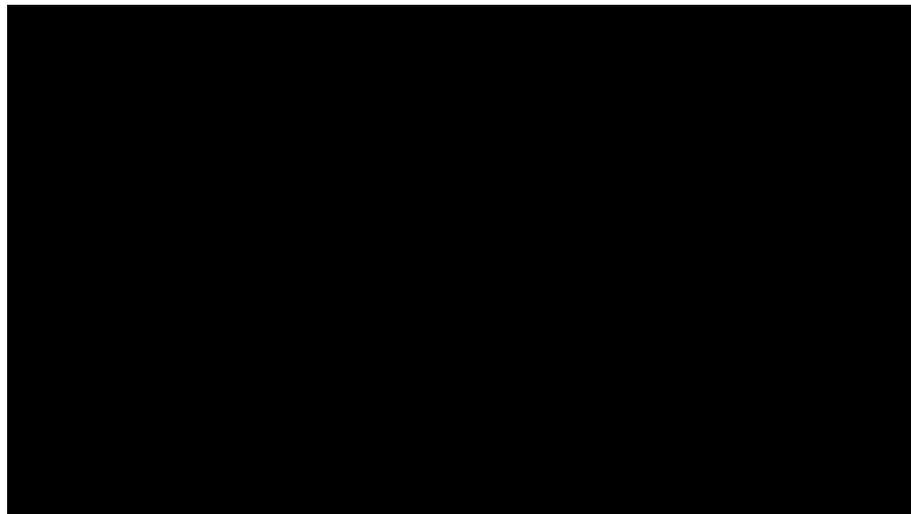
### See also

- [Adding a Textbox](#)
- [Adding a Checkbox](#)
- [Adding a Listbox](#)
- [Adding a Checkbox List](#)
- [Adding a Radiobutton List](#)
- [Adding a File Upload](#)
- [Adding a Submit Button](#)
- [Adding a Hidden Field](#)
- [Adding an HTML Editor](#)

[Home](#) > [How to Videos](#) > [Fields](#) > [Adding a File Upload](#)

## Adding a File Upload

**Shows how to add a file upload field to the form.**



[View Video in a new browser window](#)

### See also

- [Adding a Textbox](#)

- [Adding a Checkbox](#)
- [Adding a Listbox](#)
- [Adding a Checkbox List](#)
- [Adding a Radiobutton List](#)
- [Adding a Label](#)
- [Adding a Submit Button](#)
- [Adding a Hidden Field](#)
- [Adding an HTML Editor](#)

[Home](#) > [How to Videos](#) > [Fields](#) > [Adding a Submit Button](#)

## Adding a Submit Button

**Shows how to add three styles of Submit Buttons to a form.**



[View Video in a new browser window](#)

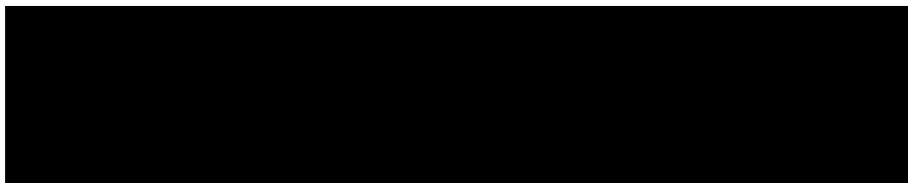
### See also

- [Adding a Textbox](#)
- [Adding a Checkbox](#)
- [Adding a Listbox](#)
- [Adding a Checkbox List](#)
- [Adding a Radiobutton List](#)
- [Adding a Label](#)
- [Adding a File Upload](#)
- [Adding a Hidden Field](#)
- [Adding an HTML Editor](#)

[Home](#) > [How to Videos](#) > [Fields](#) > [Adding a Hidden Field](#)

## Adding a Hidden Field

**Shows how to add a Hidden field to a form and populate it's value from a URL parameter.**





[View Video in a new browser window](#)

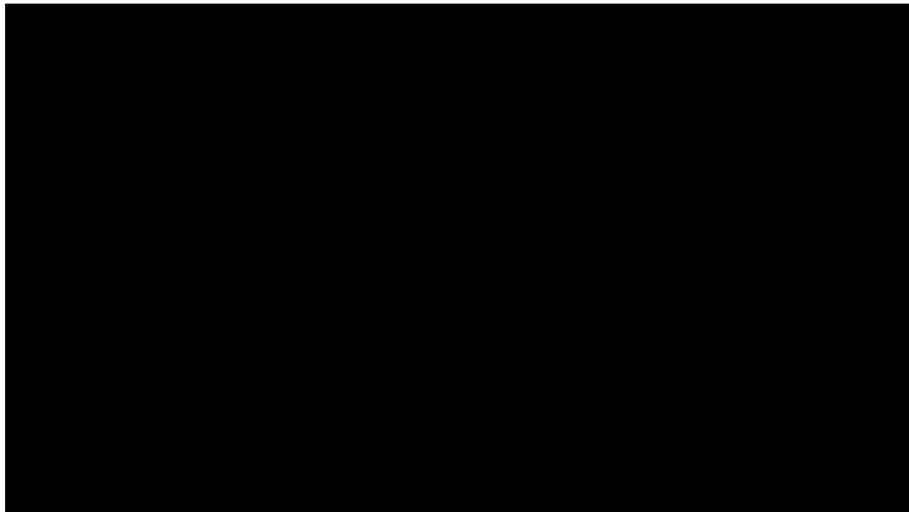
### See also

- [Adding a Textbox](#)
- [Adding a Checkbox](#)
- [Adding a Listbox](#)
- [Adding a Checkbox List](#)
- [Adding a Radiobutton List](#)
- [Adding a Label](#)
- [Adding a File Upload](#)
- [Adding a Submit Button](#)
- [Adding an HTML Editor](#)

[Home](#) > [How to Videos](#) > [Fields](#) > [Adding an HTML Editor](#)

## Adding an HTML Editor

**Shows how to add an HTML Editor to the form.**



[View Video in a new browser window](#)

### See also

- [Adding a Textbox](#)
- [Adding a Checkbox](#)
- [Adding a Listbox](#)
- [Adding a Checkbox List](#)
- [Adding a Radiobutton List](#)
- [Adding a Label](#)
- [Adding a File Upload](#)
- [Adding a Submit Button](#)

[Adding a Hidden Field](#)

[Home](#) > [How to Videos](#) > [Cascading Dropdowns](#)

## Cascading Dropdowns

Articles in this section



[Dynamic Data](#)

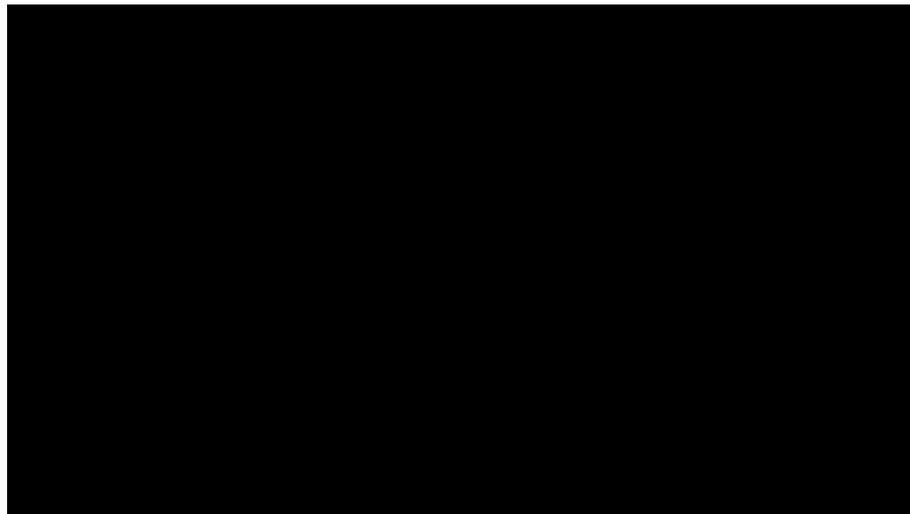
See also

[Getting Started Fields](#)  
[Multi-Page Forms](#)  
[Updatable Forms](#)

[Home](#) > [How to Videos](#) > [Cascading Dropdowns](#) > [Dynamic Data](#)

## Dynamic Data

**Shows how to create Cascading Dropdown Lists using Field Queries which fetch data from the database.**



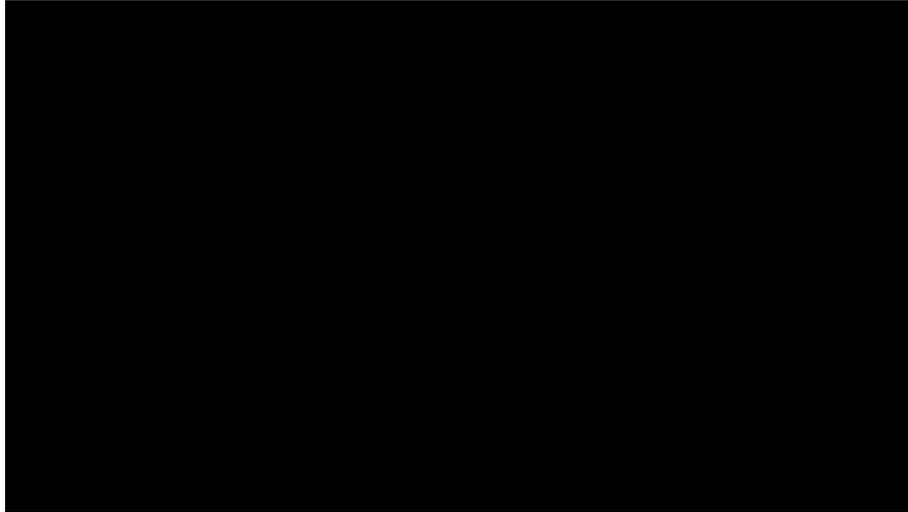
[View Video in a new browser window](#)

See also

[Home](#) > [How to Videos](#) > [Multi-Page Forms](#)

## Multi-Page Forms

Shows how to work with Multi-Page form including the [Form Wizard Options](#)



[View Video in a new browser window](#)

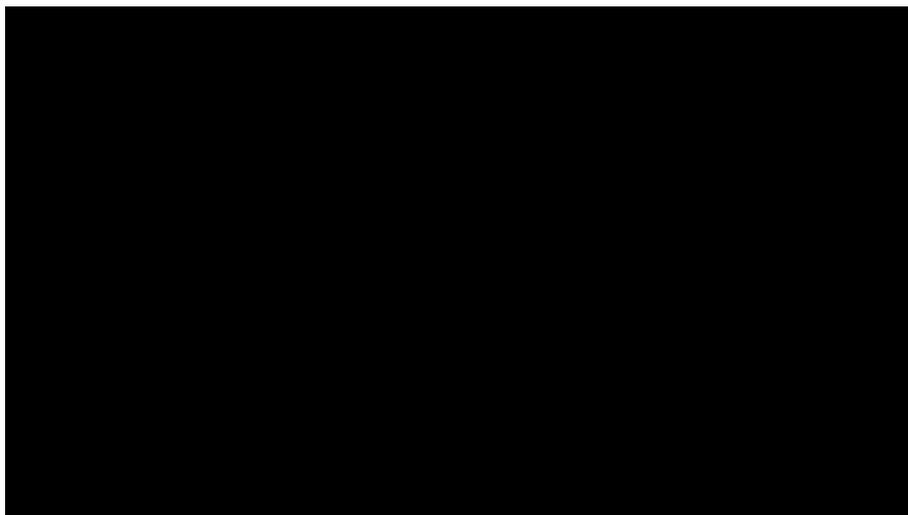
### See also

[Getting Started](#)  
[Fields](#)  
[Cascading Dropdowns](#)  
[Updatable Forms](#)

[Home](#) > [How to Videos](#) > [Updatable Forms](#)

## Updatable Forms

Shows how to work with [Updatable Forms](#)



[View Video in a new browser window](#)

See also

- [Getting Started](#)
- [Fields](#)
- [Cascading Dropdowns](#)
- [Multi-Page Forms](#)

[Home](#) > [Advanced Functions](#)

## Advanced Functions

Articles in this section

[Silent Post](#)[Field Rules](#)[User Manager](#)[Updatable Forms](#)[Transaction Option](#)

See also

[Welcome](#)  
[What's New](#)  
[Installation](#)  
[Getting Started](#)  
[Form Settings](#)  
[Basic Functions](#)  
[Working With Fields](#)  
[Cascading Dropdowns](#)  
[Multi-Page Forms](#)  
[Validation](#)  
[Data Manager](#)  
[Formatting](#)  
[Reporting](#)  
[Messaging](#)  
[How to Videos](#)

[Home](#) > [Advanced Functions](#) > [Silent Post](#)

## Silent Post

The **Silent Post** feature is useful for sending information to outside Partners, Vendors, or another process in the same Website.

A classic example of using the Silent Post feature is when using an outside Vendor to manage a Continuity Marketing Program, or CRM management. These programs are offered by many Vendors, the most notable are **Constant Contact®**, **AWeber®**, **Vertical Response®**, and **SalesForce®**.

In these services it is common for the vendor to supply code which shows a Form where an end user can sign up for a Mailing List, or some other service which the Vendor offers. The Vendors Client is instructed to place the code in their Web Page. This works fine for plain HTML pages which can have multiple Form Tags in the page. For DotNetNuke this becomes a problem, as it is difficult in ASP.NET and DotNetNuke to include a FORM tag in the page.

Using the Silent Post Feature a Vendor Form can be Mimicked with the appropriate end User Visible Fields and the required Vendor Hidden fields. When the Form Master 2008 Form is submitted the Vendor form is Created and Posted in the Background to the URL supplied by the Vendor in their Forms Tag "Action" attribute. The series of events is shown below

1. End User enters information into the Form Master form and Submits
2. Data is collected and stored according to the options specified
3. The Specified Silent Post Fields are used to create an HTTP Request to the Vendor URL
4. The end user does not know anything about the Silent Post activity, Form Master handles the End User as specified in Settings > Other Settings as usual.

For an example of setting up a form to work with SalesForce® please see this [Salesforce® Case Creation Form](#)

See also

[Field Rules](#)  
[User Manager](#)  
[Updatable Forms](#)  
[Transaction Option](#)

# Field Rules

Form Master Responsive supports conditional field display rules. For example in a Form where we have a DropDown List of Regions populated with US States and Canadian Provinces, is a User is not from one of these countries, we should provide a option in the DropDown List for "Other", and a TextBox where the user can fill in a Region. Field Rules will allow us to show the Other Region TextBox if the value of the DropDown List is "other" and hide it when it is not. Below is the Field Rules page.

## FORM FIELD RULES

When Source Field:  << Select >> ▼

Contains/Is:

Show Target Field(s):

Address1     Address2     City     Country  
 Email     Message     Name     OtherRegion  
 PostalCode     Region     SendInfo     Subject

Field rules will operate on all controls in a Cascading Dropdown Set.  
Dropdown controls which are children in a Cascading Dropdown set are not shown

## EXISTING RULES

Source Field	Contains/Is	Show Fields(s)	Delete
Region	other	OtherRegion	<a href="#">Delete</a>

- **When Source Field:** The Field for which the Value will cause other Fields to be shown or hidden. This Dropdown List contains the defined form Fields excluding incompatible types such as Hidden, File Upload, Button, and Label Fields.
- **Contains/Is:** For Checkboxes the default show value is "Checked". All other field types, the show value will be Contains. A Partial value will be matched for example:
  - A DropDown List of Regions contains "Other Region" and a TextBox should be shown when the user selects that value. If "Other" is entered into the Contains TextBox it will match the DropDown value.
  - The Contains TextBox can also contain a wildcard of \*.\* which will match any value Target Fields. This works for any type of Source Field.
- **Show Target Fields(s):** Select One or More Fields to Show when the Rule is True
- **Existing Rules:** The grid shows existing rules for this form.

**The Grid Columns:**

- **Source Field:** The Form Field which Triggers the Rule
- **Contains/Is:** The Value which the Source Field must Contain, or for Checkboxes be Checked to Trigger the Rule
- **The Grid Columns:** The list of Fields which will be shown when the Rule Condition is True.

**The Process:**

1. Select a Source Field from the "When Source Field:"
2. Enter a value which the Source Field must contain to Trigger the Rule. For Source Field of type Checkbox the value defaults to Checked
3. Select one or more fields to be the target of the Rule in the "Show Target Field(s)" Checkbox List
4. Click the "Add Rule" button

**Field Rule In Action:** Not Triggered

**Name:\***

**Email:\***

**Region:** << Select >> ▼

**Country:** << Select >> ▼

**Subject:\***

**Send Information**

**Message:\***

[Submit](#)

Field Rule In Action: Triggered

**Name:\***

**Email:\***

**Region:** Other ▼

**Country:** << Select >> ▼

**Other Region:**

**Subject:\***

**Send Information**

**Message:\***

[Submit](#)

**See also**

[Silent Post](#)  
[User Manager](#)  
[Updatable Forms](#)  
[Transaction Option](#)

[Home](#) > [Advanced Functions](#) > [User Manager](#)

## User Manager

Form Master Responsive can **Register a User** and add the user to any number of Roles when the Form is Submitted. If a user is already logged in the Registration will be ignored, but the user will still be added to any Roles which they are not already members of. Below is the User Manager page which is used to manage this feature.

**Note:** User Manager is available for Super Users and users in the Administrators Role only.

## USER MANAGER

Register User

User Name Field  Username

Password Field  Password

First Name Field  firstname

Last Name Field  lastname

Email Field  Email

User Roles  Customers  Form Master 2008 Source  ServiceAccess  Translator (en-US)  
 Form Master 2008  Form Master NextGEN  Subscribers  Trialware  
 Form Master 2008 Enterprise

[Update](#)

- **Register User:** Check to Register the User when the Form is Submitted. Fields to be used for the UserName, Password, First Name, Last Name, and Email must be defined. If user is logged in this setting is ignored.
- **User Name Field:** Field to be used as the User Name when Registering a User
- **Password Field:** Field to be used as the Password when Registering a User
- **First Name Field:** Field to be used as the First Name when Registering a User
- **Last Name Field:** Field to be used as the Last Name when Registering a User
- **Email Field:** Field to be used as the Email when Registering a User
- **User Roles:** Select The Roles the User should be Added to when Submitting the Form. Note: User Must be logged in or being Registered during submission.

### See also

[Silent Post](#)  
[Field Rules](#)  
[Updatable Forms](#)  
[Transaction Option](#)

[Home](#) > [Advanced Functions](#) > [Updatable Forms](#)

## Updatable Forms

Form Master Responsive can be configured to allow an authenticated user to update previously submitted data.

- The form must be on a page that requires the user to be logged in
- The form must be mapped to an external table using the [Data Manager](#)
- The form must include a field which represents the DotNetNuke generated UserID integer value
- The form must include a [Hidden Field](#) with the "Field Name" property set to DataRecordID
- The External Table mapping must include columns which can be mapped to the UserID and the DataRecordID fields in the form
- The form must include a Form Query to pre-populate the form on page load with the user's previously entered data

[Working with Updatable Forms Video](#)

### See also

[Silent Post](#)  
[Field Rules](#)  
[User Manager](#)  
[Transaction Option](#)

[Home](#) > [Advanced Functions](#) > [Transaction Option](#)

## Transaction Option

Form Master Responsive uses Transactions when the option to save to Database is checked. What the use of Transactions does is to ensure that all the updates which Form Master Responsive makes to multiple Database Tables and optionally the file system for file uploads when a form is submitted all succeed or

are all rolled back. This makes sure that the Database and optionally the file system are left in a consistent state.

The use of Transaction enlists resources of the host operating system which can span multiple machines, where Transaction is actually a Distributed Transaction. In some cases the machine configuration denies Distributed Transactions for most likely security reasons.

Form Master Responsive provides a method to turn off the use of Transactions by editing an XML file named Globals.xml file. After Form Master Responsive has been installed and the first instance placed on a page, this file will be available in /portals/[PORTALID].

To Disable Transactions, edit the <UseTransactions /> element so the value is False as shown in the image below. For previously installed instances of Form Master Responsive the <UseTransactions>False</UseTransactions> line will need to be added.

```
<FormMasterGlobals>
  <Portal PortalID="0">
    <UploadPath></UploadPath>
    <DoLogging>False</DoLogging>
  </Portal>
  <UseTransactions>False</UseTransactions>
</FormMasterGlobals>
```

### See also

- [Silent Post](#)
- [Field Rules](#)
- [User Manager](#)
- [Updatable Forms](#)

# Index