

Custom Fields

Product

Custom product fields can be created by creating a template called "productfields.cshtml" in the "config" theme.

To stop being overwritten on upgraded this is usually created at portal level

/Portals/<portalfolder>/Themes/config/default/productfields.cshtml

Although it can be created at the system level

(/DesktopModule/NBright/NBrightBuy/Themes/config/default), it will then be available across all portals that are using the system level theme. And will be overwritten on any upgrade to OpenStore.

Example

```
@inherits NBrightBuy.render.NBrightBuyRazorTokens<NBrightDNN.NBrightRazor>
@using NBrightDNN
@using Nevoweb.DNN.NBrightBuy.Components
@{
    var info = (NBrightInfo)Model.List.First();
}
<div> My custom field: @TextBox(info, "genxml/textbox/txttest2", "class='form-control'
maxlength='250'") </div>
<div class="col-sm-6">
    <label class="control-label">Custom Order</label>
    <!-- Field to accept localized data -->
    @TextBox(info, "genxml/lang/genxml/textbox/txttest1", "class='form-control' maxlength='250'")
    <!-- Field to accept non-localized data -->
    @TextBox(info, "genxml/textbox/txttest2", "class='form-control' maxlength='250'")
</div>
```

Displaying in Razor

```
@if (@info.GetXmlProperty("genxml/textbox/customfield") != "")
{
<div> My custom field: @info.GetXmlProperty("genxml/textbox/customfield") </div>
}
```

or for the localized version

```
@if (@info.GetXmlProperty("genxml/lang/genxml/textbox/customfield") != "")
{
<div> My custom field: @info.GetXmlProperty("genxml/lang/genxml/textbox/customfield") </div>
}
```

Model Custom Fields

Model custom fields work the same as product custom fields, but use a file name called: **modelfields.cshtml**

Client Custom Fields

Client custom fields work the same as product custom fields, but use a file name called: **clientfields.cshtml**