

Multiple Child Form Guide

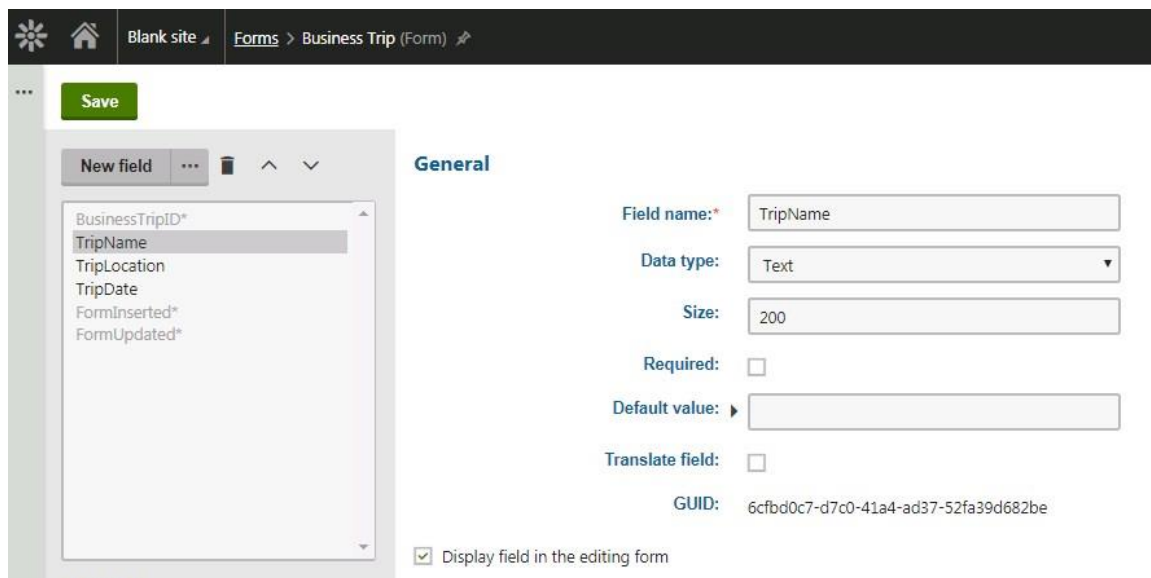
The Multiple Child Form Control allows you to add a many-to-1 sub form onto an existing form object. An example of this would be a Form concerning a Trip that contains multiple Sub-forms describing your trip's expenses.

SETUP

To use this tool, once installed into Kentico, first make sure to Resign all your Macros (System -> Macros -> Signatures, Resign all Macros)

Next, create your Main Form (in our Example, we will use a form called "Business Trip"). We'll do this in the Forms since this will email out the Business Trip later to a manager.

Add your normal fields as you would.



The screenshot shows the Kentico Forms editor interface. At the top, there's a navigation bar with a home icon, a 'Blank site' link, and a breadcrumb trail 'Forms > Business Trip (Form)'. Below this is a green 'Save' button. The main area is divided into two panels. On the left, a 'New field' panel shows a list of fields: 'BusinessTripID*', 'TripName' (selected), 'TripLocation', 'TripDate', 'FormInserted*', and 'FormUpdated*'. On the right, the 'General' configuration panel for the 'TripName' field is displayed. It includes fields for 'Field name:' (TripName), 'Data type:' (Text), 'Size:' (200), 'Required:' (checkbox), 'Default value:' (text box), 'Translate field:' (checkbox), and 'GUID:' (6cfbd0c7-d7c0-41a4-ad37-52fa39d682be). At the bottom of the 'General' panel, there is a checkbox labeled 'Display field in the editing form' which is checked.

Then, create your Child Form ("Business Expense") and let's add some fields, such as "ExpenseName" and "ExpenseCost". We'll do this in the Custom Tables as these are simply pieces of data that will belong to the parent.

Lastly, we need to add a field that will house a unique identifier for the Parent (Business Trip) this Child (Business Expense) belongs to. Make sure this field is HIDDEN as this is filled in by the tool later.

Some examples will be "ParentBusinessExpenseID" (int), "ParentBusinessExpenseGUID" (guid), "ParentBusinessExpenseCodeName" (text). Doesn't matter what you name it.

Blank site
Custom tables > Business Expense (Custom table)

Save

New field

ItemID*
ExpenseName
ExpenseCost
ParentBusinessTripID*
ItemGUID*

Quick links:
[General](#)
[Field appearance](#)
[Editing control settings](#)
[Validation](#)
[CSS styles](#)
[HTML Envelope](#)
[Field advanced settings](#)

General

Field name:* ParentBusinessTripID
Data type: Integer number
Required: ☒
Default value:
GUID: 22061d09-ae15-482b-9068-dbcd83e7a987
☒ Display field in the editing form

Field appearance

Field caption: Parent Business Trip ID
Field description:
Explanation text:
Form control: Text box

Logic

So structurally, how this Many-to-1 relationship works, is you will have 1 Parent Entry (Business Trip), which will have a BusinessTripID of say "1", and many Children (Business Expenses), each with a ParentID field (ParentBusinessTripID) of "1", linking the Children with their Parent.

Configuration

On the Parent Form (Business Trip), we need to add a Field without database representation of type text (doesn't matter), and use the Form Control of "Multiple Child Form"

Save

New field

BusinessTripID*
TripName
TripLocation
TripDate
New field*
FormIsUpdated*
FormUpdated*

Quick links:
General
Field appearance
Editing control settings
Validation
CSS styles
HTML envelope
Field advanced settings

Field type

Field type: Field without database representation

General

Field name: ExpensesHere

Data type: Text

Size: 200

Required: ☐

Default value:

GUID:

☒ Display field in the editing form

Field appearance

Field caption: Expenses

Field description:

Explanation text:

Form control: Multiple Child Form

Next select the Child Class, and the Parent Form ID Field (Identifier from the current Parent Form), the Child Form Parent ID Field (the field on the Child Form that will store the Parent ID's value), and the Child ID Field (the RowID from the Child Form), as well as any Button Configuration. This tells the form what it needs in order to pull in and relate the children to the parent.

Save

Parent Form

New field

BusinessTripID*
ExpensesHere
ExpensesCost
ParentBusinessTripID*
RowID

Quick links:
General
Field appearance
Editing control settings
Validation
CSS styles
HTML envelope
Field advanced settings

Child Class Name: BusinessExpense

Alternative Form Name:

Minimum Children: 1

Maximum Children: 4

Parent Form ID Field: BusinessTripID

Child Form Parent ID Field: ParentBusinessTripID

Child ID Field: RowID

Styling

Add Remove Control Location: After Form

Child Form

Save

New field

BusinessTripID*
ExpensesHere
ExpensesCost
ParentBusinessTripID*
RowID

Quick links:
General
Field appearance
Editing control settings
Validation
CSS styles
HTML envelope
Field advanced settings

☐ Display field in the

Additional Options

Some additional options exist, such as an Alternative Form Name (so you can define how the child form will look), the minimum Children and Maximum to enforce that there should be at least X and no more than Y child items. The Styling allows you to determine where the Add / Remove child entries buttons will appear, styling, and any HTML before/middle/after so you can control the layout.

Testing out

Now simply add your Form onto a page (using something such as the on-Line form widget). You can now add multiple child items by clicking the + or -, then submit!

Trip Name:

Trip Location:

Trip Date:

Expense Name:

Expense Cost:

Expenses:

Expense Name:

Expense Cost:

Forms > Business Trip (Form)

Actions	BusinessTripID	Trip Name	Trip Location	Trip Date	Form inserted	Form updated
...	4	Kentico Road Show	Chicago	10/26/2017	10/26/2017 2:16:24 PM	10/26/2017 2:16:24 PM

Custom tables > Business Expense (Custom table)

The first 5 columns are displayed by default. Click 'Select displayed fields' button to modify the filter.

Actions	ItemID	Expense Name	Expense Cost	Parent Business Trip ID	GUID
...	8	Taxi	10	4	fc185348-7cb5-43b3-aaaa-03ceeacbac82
...	9	Food	20	4	2d739867-956a-43a5-8856-d655855ba01b

Validation

Child Forms must also validate before the entire thing can submit. Child Form Data May submit during this processes, but once a Form does submit, it removes any orphaned children.

Additional Fields

Alternative Form Name: Can select an alternative form of your child form.

Minimum Children: Minimum number of Child Forms Required

Maximum Children: Maximum number of Child Forms allowed

Control Location: If the Add / Remove should be placed to the before or after the Child Form Item

Control Style: Can use a Button, a Text Link, or an Image

Add Text: If Button/Link Text, the text of the add button

Add Image: If Image button, the URL to the image for the add button.

Remove Text: If Button/Link Text, the text of the remove button

Remove Image: If Image button, the URL to the image for the remove button.

HTML Before Buttons: This HTML will be placed before the Add/Remove buttons.

HTML After Buttons: This HTML will be placed after the Add/Remove buttons.

HTML Before Child Form: This HTML will go before each Child Form Item

HTML After Child Form: This HTML will go after each Child Form Item

HTML In Between Child Form Panels: This HTML will go between the Child Form Items.