

# Subforms

Form ID	Table Name
leaveApplication	leave_application
leaveDetails	leave_details

**Relationship:** In this example, the 'parent' form (*leave application*) has one 'child' form (*leave details*).

**Creating the Child Form:** leaveDetails

The screenshot shows the Joget Form Builder interface. At the top, there is a navigation bar with the Joget logo and 'FORM BUILDER' text. To the right of the logo are three buttons: 'Form Builder', 'Datalist Builder', and 'Userview Builder'. Below the navigation bar is a main workflow area with four steps: 'Design Form' (Design and layout), 'Properties' (Form properties), 'Preview' (Preview form), and 'Save' (Save form). On the left side, there is a 'Basic' widget palette containing: Hidden Field, Text Field, Password Field, Text Area, SelectBox, CheckBox, Radio, and Date Picker. The main workspace displays a form titled 'Leave Details' with a dashed border. The form contains three fields: 'leave\_application\_id' (a hidden field with a 'HIDDEN FIELD' label and a 'Drag This Column' instruction), 'Reason' (a text field), and 'Justification' (a text area). Below the form, there is a link for 'ADVANCED: JSON Definition'.

First, you will need to create a child form or the form that you would want to embed in your parent form. In the child form, add a field that will be used to store the ID of the parent form. In this example, we added the field `leave_application_id`.

**Creating The Parent Form:** leaveApplication

joget | FORM BUILDER Form Builder Datalist Builder Userview Builder

**Design Form** Design and layout **Properties** Form properties **Preview** Preview form **Save** Save form

**Basic**

- Hidden Field
- Text Field
- Password Field
- Text Area
- SelectBox
- CheckBox
- Radio
- Date Picker

**Leave Application**

*Drag This Column*

Name

leave\_details\_id  HIDDEN FIELD

**Leave Details**

Reason

Justification

In the parent form, you can (optionally) add a field that will be used to store the ID of the child form. In this example, the field `leave_details_id` has been added. This is only needed in situations where there is a one-to-one relationship between the parent form and the child form. You can then embed your child form in the parent form by dragging the **Sub Form** form element into the section that you want the subform to appear in.

**Linking the Child Form to the Parent Form**

**Property Editor**

**Edit SubForm**

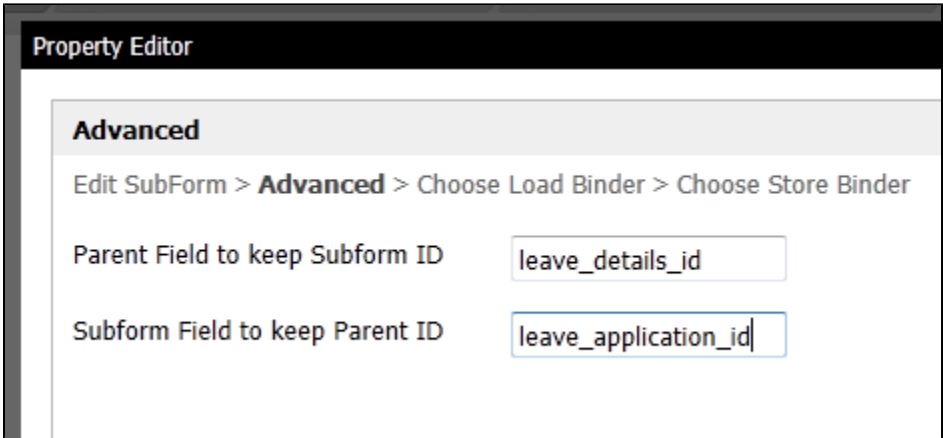
Edit SubForm > Advanced > Choose Load Binder > Choose Store Binder

ID \*

Label

Form ID \*

Readonly



To link the child form to its parent, you will need to edit the properties of the subform input that you added to your parent form in the previous step. The following fields are available:

- Parent Field to keep Subform ID: ID of the field in the parent form that will be used to store the ID of the child form
- Subform Field to keep Parent ID: ID of the field in the child form that will be used to store the ID of the parent form

**Using Your Newly Created Form**

When using your parent form in a process activity, the system automatically fills in the IDs of the parent and the child forms upon form submission.

**One-to-One Relationship**

In a one-to-one relationship, you will only have to select a form to be used in the subform properties. The child record will use the parent form ID as primary key. In this case, 'Parent Field to keep Subform ID' can be set so that the parent form has a reference to the child record.

Sample data

Table: leave\_application

id	leave_details_id	....
1550	1550	....

Table: leave\_details

id	leave_application_id	
1550	1550	....