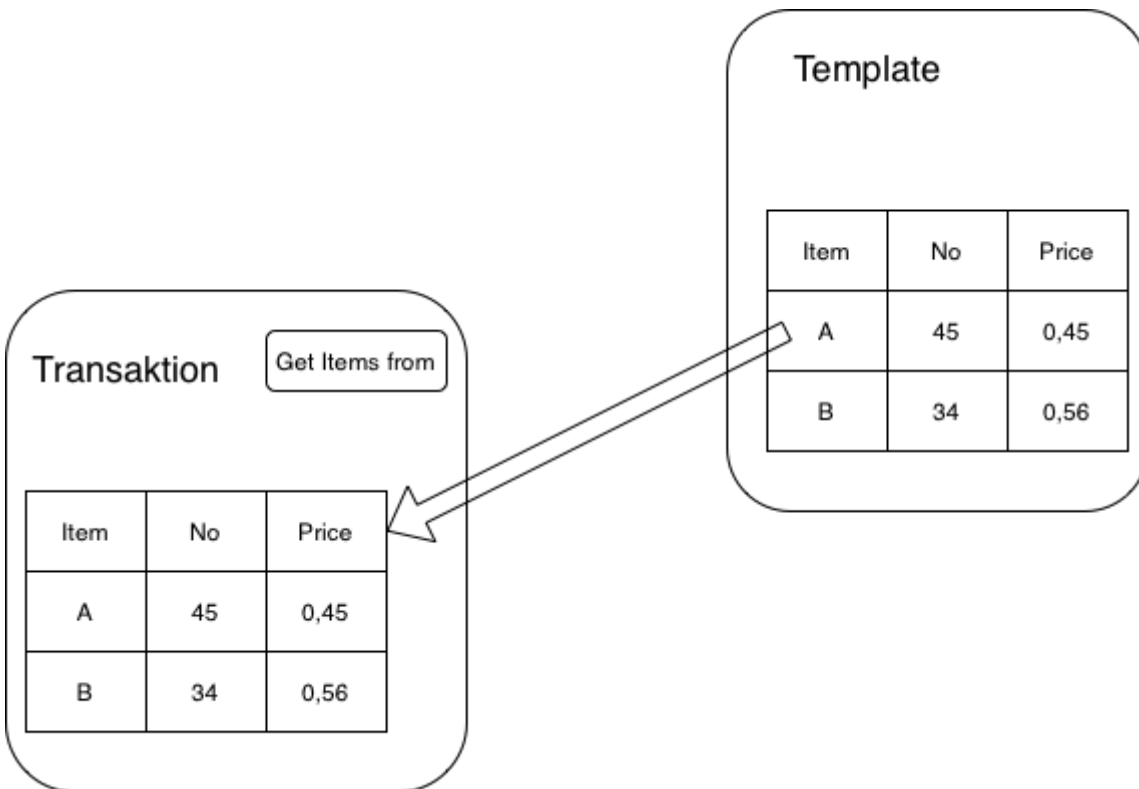


# Tabelleninhalte aus einer Tabelle befüllen

## Einleitung

In dieser Anleitung finden wir ein Beispiel wie wir eine Tabelle in einer Transaktion mit Inhalten aus einer Tabelle einer Vorlage füllen kann.



# DocType Template erstellen

Zunächst wollen wir ein Dokument erstellen in welchem wir unser Template festhalten, damit wir diese als Vorlage für unsere Transaktion verwenden können. Dazu erstellen wir hier Transaction Template.

Das Transaction Template nehmen wir in diesem Beispiel als Platzhalter. Es könnte z.B. auch eine Angebotsvorlage sein in welcher wir eine bestimmte Sammlung von Artikeln halten welche wir immer wieder zum Einsatz bringen wollen.

Im gezeigten Beispiel haben wir ein Template mit dem Namen Colors erstellen und mit den werden Red und Green.

Hier die wesentlichen Inhalte des DocTypes

E > DocType > Transaction Template

Suchen oder Befehl eingeben (Strg + G)

Hilfe

WS

Transaction Template

Gehen Sie zur Liste Transaction Template

<

>

Drucken

...

speichern

Felder

Felder

<input type="checkbox"/>	No.	Bezeichnung	Typ	Name	Zwinge...	Optionen	
<input type="checkbox"/>	1	Title	Daten	title	<input type="checkbox"/>		<a href="#">Bearbeiten</a>
<input type="checkbox"/>	2	Transaction Items	Tabelle	transaction_items	<input type="checkbox"/>	Transaction Items	<a href="#">Bearbeiten</a>

Zeile hinzufügen

Bezeichnung

Automatische Benennung

field:title

Naming Options:

- 1. **field:[fieldname]** - By Field
- 2. **naming\_series:** - By Naming Series (field called naming\_series must be present)
- 3. **Prompt** - Prompt user for a name
- 4. **[series]** - Series by prefix (separated by a dot); for example PRE.####
- 5. **format:EXAMPLE-{MM}morewords{fieldname1}-{fieldname2}-{#####}** - Replace all braced words (fieldnames, date words (DD, MM, YY), series) with their value. Outside braces, any characters can be used.

Großschreibung

☒ Umbenennen zulassen

Beschreibung

Dokumentationslink

URL für Dokumentation oder Hilfe

Und die dazu gehörige Untertabelle "Transaction Items"

E > DocType > Transaction Template Items

Suchen oder Befehl eingeben (Strg + G) | Hilfe | WS

Transaction Template Items

dies vor 3 Tag(en)

Administrator erstellte dies vor 3 Tag(en)

Modul \*

Custom

☒ Benutzerdefiniert?

☐ Beta

☐ Is Virtual

☒ Ist Untertabelle

Untergeordnete Tabellen werden in anderen DocTypes als Raster angezeigt

☐ Ist Baum

Baumstrukturen werden mit Nested Set implementiert

☒ Editierbares Raster

Felder

No.	Bezeichnung	Typ	Name	Zwinge...	Optionen	
<input type="checkbox"/> 1	Value 1	Daten	value_1	<input type="checkbox"/>		<a href="#">Bearbeiten</a>
<input type="checkbox"/> 2	Value 2	Daten	value_2	<input type="checkbox"/>		<a href="#">Bearbeiten</a>

Zeile hinzufügen

Bezeichnung

Automatische Benennung

Beschreibung

Ist der DocType erstellt, erstellen wir direkt eine Instanz.

E > Transaction Template

Suchen oder Befehl eingeben (Strg + G) | Hilfe | WS

Transaction Template

Filtern nach

Zugewiesen zu

Erstellt von

Edit Filters

Schlagworte

Tags anzeigen

Filter speichern

Name des Filters

List View

Hinzufügen Transaction Template

Name

☐ Name

☐ 6da55d2c0c

Filter

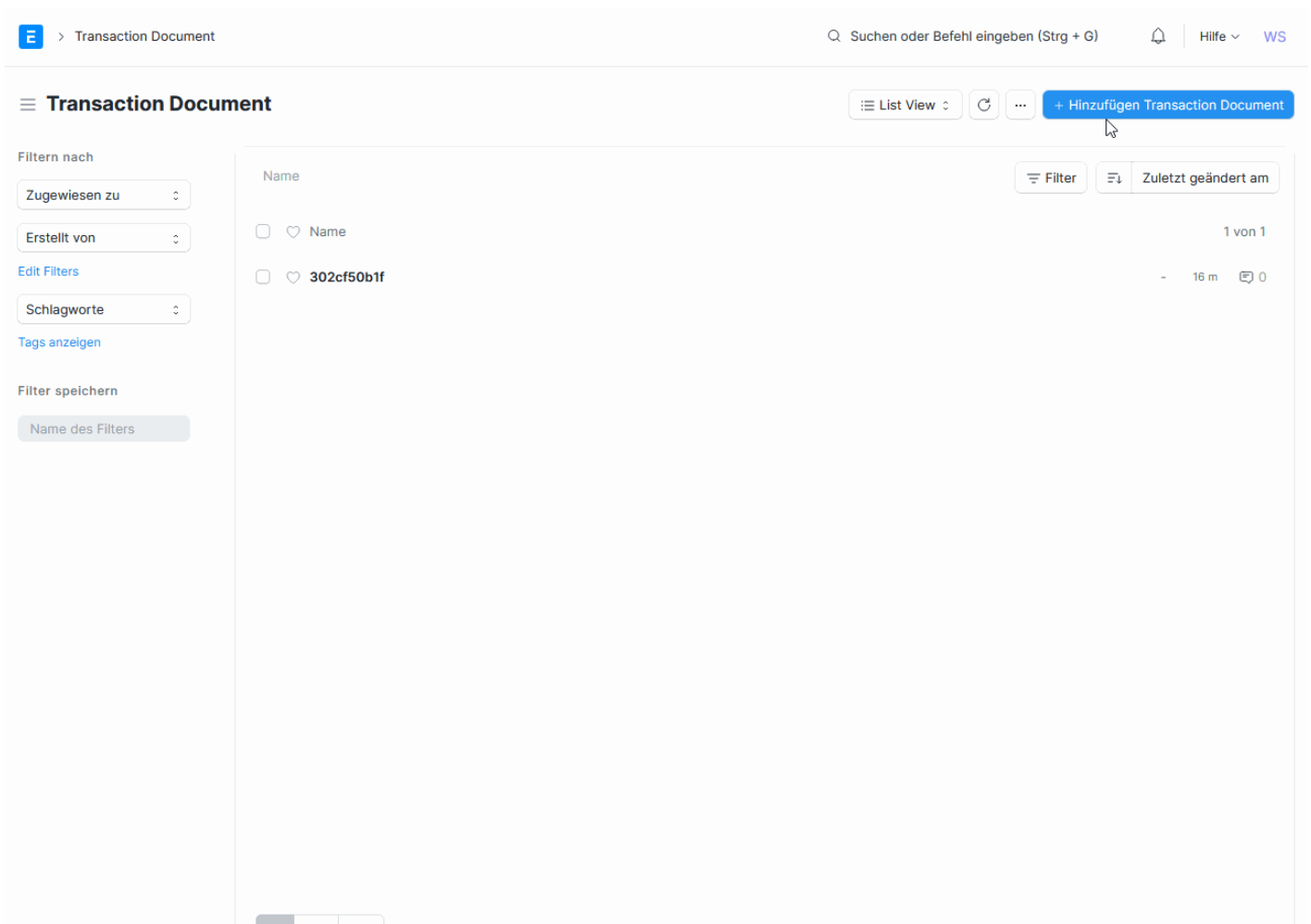
Zuletzt geändert am

1 von 1

3 d

# DocType Transaktion erstellen

Nun erstellen wir einen weiteren DocType der die eigentliche Transaktion repräsentiert. Diese kann z.B. ein Angebot sein in welches wir Artikel aus unserer Angebotsvorlage importieren. In diesem Beispiel importieren wir lediglich zwei Felder mit Werten vom Typ Data. Dem Beispiel folgend werden es der Wert Red und Green.



Der DocType ist wie folgt erstellt

E > DocType > Transaction Document

Suchen oder Befehl eingeben (Strg + G) | Hilfe | WS

Transaction Document

Gehen Sie zur Liste Transaction Document < > ... speichern

☐ Schnelleingabe  
Öffnen Sie ein Dialogfeld mit Pflichtfeldern, um schnell einen neuen Datensatz zu erstellen

Felder

Felder

<input type="checkbox"/>	No.	Bezeichnung	Typ	Name	Zwingend no...	Optionen	
<input type="checkbox"/>	1	Title	Daten	title	<input type="checkbox"/>		<a href="#">Bearbeiten</a>
<input type="checkbox"/>	2	Transaction Items	Tabelle	transaction_items	<input type="checkbox"/>	Transaction Items	<a href="#">Bearbeiten</a>

Zeile hinzufügen

Bezeichnung

Automatische Benennung  
field:title

Naming Options:  
1. field:[fieldname] - By Field  
2. naming\_series: - By Naming Series (field called naming\_series must be present)  
3. Prompt - Prompt user for a name  
4. [series] - Series by prefix (separated by a dot); for example PRE-####  
5. formatEXAMPLE-(MM)morewords(fieldname)-{fieldname2}-{#####} - Replace all braced words (fieldnames, date words (DD, MM, YY), series) with their value. Outside braces, any characters can be used.

Großschreibung

☒ Umbenennen zulassen

Beschreibung

Dokumentationslink

URL für Dokumentation oder Hilfe

Dazu die Untertabelle Transaction Items

E > DocType > Transaction Items

Suchen oder Befehl eingeben (Strg + G) | Hilfe | WS

Transaction Items

< > ... speichern

Custom

☐ Beta

☒ Ist Untertabelle

Untergeordnete Tabellen werden in anderen DocTypes als Raster angezeigt

☐ Ist Baum

Baumstrukturen werden mit Nested Set implementiert

☒ Editierbares Raster

☐ Is Virtual

Felder

Felder

<input type="checkbox"/>	No.	Bezeichnung	Typ	Name	Zwinge...	Optionen	
<input type="checkbox"/>	1	Value 1	Daten	value_1	<input type="checkbox"/>		<a href="#">Bearbeiten</a>
<input type="checkbox"/>	2	Value 2	Daten	value_2	<input type="checkbox"/>		<a href="#">Bearbeiten</a>

Zeile hinzufügen

Bezeichnung

Automatische Benennung

Naming Options:

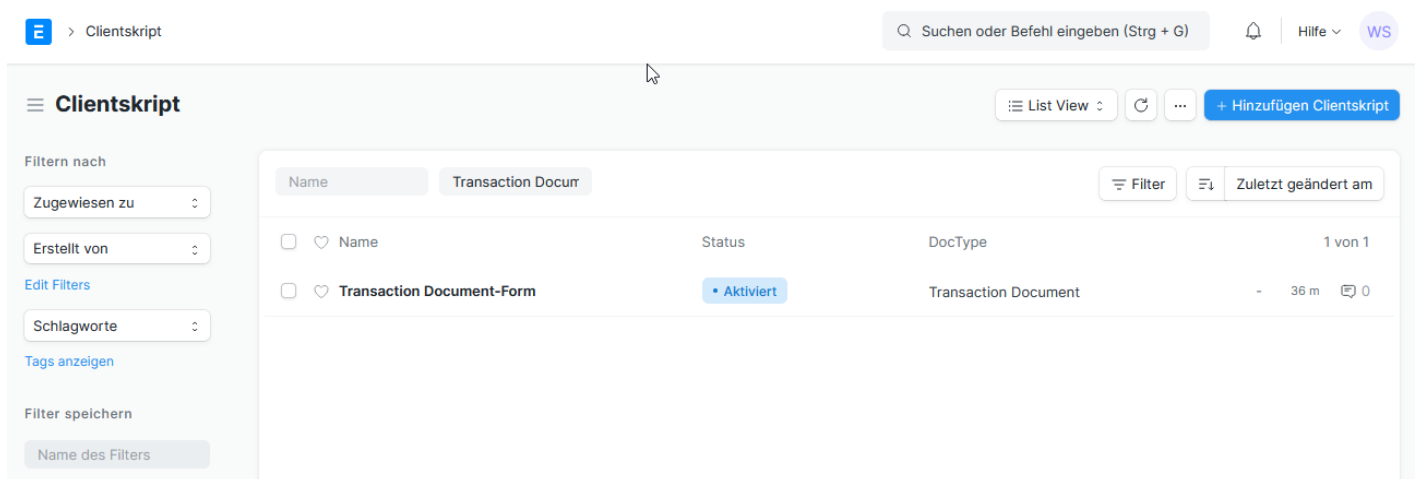
Beschreibung

Button in Transaktion erstellen

Nun wollen wir über einen Button die Möglichkeit bereit stellen, dass wir die Daten aus dem Template in die Transaction übertragen können. Dazu erstellen wir ein Client Script welches einen Button auf unser DocType Transaction bereitstellt über welchen wir per Dialog aus einem Template wählen können.

Nun haben wir alles eingebaut um eine neue Transaktion zu erstellen und dort Werte aus einem Template zu holen.

Dazu erstellen wir ein Custom Script



mit dem folgenden Code

```
// The fetch-from fields
var fields = [
    "value_1",
    "value_2"];

frappe.ui.form.on('Transaction Document', {
    refresh(frm) {
        var cur_frm = frm;
        console.log("Add button");
        frm.add_custom_button('Transaction Template', function () { frm.trigger('get_items') }, __('Get Items From'));
    },
    get_items(frm){
        start_dialog(frm);
    }
});
```

```

function start_dialog(frm) {
    let dialog = new frappe.ui.form.MultiSelectDialog({

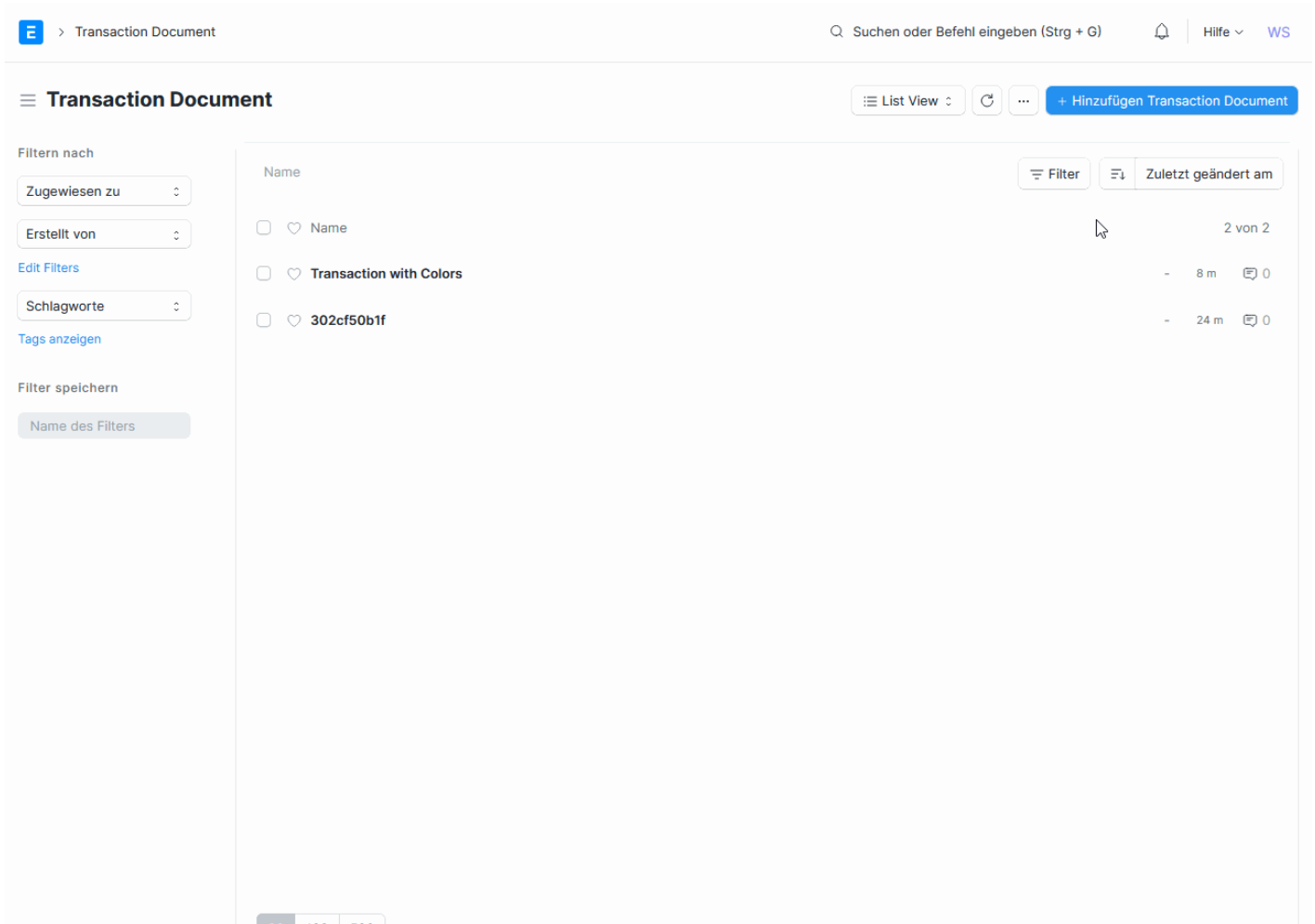
        // Read carefully and adjust parameters
        doctype: "Transaction Template", // Doctype we want to pick up
        target: cur_frm,
        setters: {
        },
        date_field: "creation", // "modified", "creation", ...
        get_query() {
            // MultiDialog Listfilter
            return {
                filters: { }
            };
        },
        action(selections) {
            var name = selections[0];
            frappe.db.get_doc("Transaction Template", name) // Again, the Doctype we want to pick up
                .then(doc => {
                    // Copy the items from the template and paste them into the cur_frm
                    for(var n = 0; n < doc.transaction_items.length; n++){
                        var item=doc.transaction_items[n];

                        // Copy-Paste Operation
                        var child = {};
                        for(var m = 0; m < fields.length; m++){
                            child[fields[m]] = item[fields[m]];
                        }
                        cur_frm.add_child("transaction_items",child);
                        cur_frm.refresh_fields("transaction_items"); // Refresh Tabelle
                    }
                });
        }
    });
}

```

# Demo

Hier sehen wir wie wir eine neue Transaktion erstellen und diese gleich mit den Werten einer Vorlage füttern.



## Hilfe Artikel

- DocType erstellen
- Client Script erstellen



