

# Budibase

Quelloffen, aber nicht kostenlos

Budibase ist eine quelloffene Low-Code-Plattform. Der Quellcode steht bei Github zur Verfügung. Dennoch ist der Einsatz der Software nicht gänzlich kostenlos; Budibase lässt sich nur mit einem npm-Paket betreiben, das direkt beim Hersteller bezogen werden muss. Dabei unterscheidet der Anbieter von einem im Funktionsumfang eingeschränkten Free-Plan bis hin zum Business-Plan mit allen Funktionen für 400 US-Dollar, beziehungsweise dem maßgeschneiderten Enterprise-Plan mit Preisen auf Anfrage. Unternehmen können Budibase entweder selbst on-Premises hosten oder als Cloud-Software vom Hersteller beziehen. Die Preise sind dabei identisch.

- Installation
  - Installation über einen Docker container mit ssl
- Budibase Internal DB
  - Beziehungen Relationships
- Budibase GUI Elemente
  - Ein Option Picker als Relationship Picker zum Filtern zweckentfremden
  - URL Variablen und öffnen im neuen Fenster
  - Einen copy to clipboard Button über eine embed Komponente
  - Mit einer Embed Komponente den Webseiten Titel ändern.
  - Mit einer Embed Komponente alle Attachments in einem Attachmentfield zum Download bereitstellen

# Installation

# Installation über einen Docker container mit ssl

## Voraussetzungen:

Docker: <https://www.docker.com/get-started>

## Docker installieren

Siehe Buch -> [Installation](#)

## Budibase installieren

Als Befehl ohne ssl einfach port 10000

```
sudo docker run -d -t \  
  --name=budibase \  
  -p 10000:80 \  
  -v /home/duffy/budibase/data:/data \  
  --restart unless-stopped \  
  budibase/budibase:latest
```

## Environment Parameter

Variablenname	Beschreibung
CUSTOM_DOMAIN	Wenn eine Domain im Format domain.com festgelegt ist, versucht Budibase automatisch, ein SSL-Zertifikat zu erstellen und HTTPS für diese Domain zu verwenden. Die Domain muss bereits auf den Budibase-Container gerichtet sein, damit dies korrekt überprüft werden kann.
INTERNAL_API_KEY	Ein API-Schlüssel, der zum Zugriff auf viele Kernkomponenten verwendet werden kann. Dieser sollte auf eine zufällige Zeichenkette aktualisiert werden.

Variablenname	Beschreibung
JWT_SECRET	Ein geheimer Schlüssel, der zum Sichern aller Sitzungen mit Budibase verwendet wird. Dies sollte auf eine zufällige Zeichenkette aktualisiert werden. Beachten Sie, dass das Ändern dieser Zeichenkette alle vorhandenen Sitzungen ungültig macht.
MINIO_ACCESS_KEY und MINIO_SECRET_KEY	Diese beiden Umgebungsvariablen sollten auf eine Kombination aus zufälligen Zeichenketten festgelegt werden, um den Zugriff auf MinIO abzusichern. Diese können auch verwendet werden, um sich im MinIO-Browser anzumelden, wenn gewünscht.
REDIS_PASSWORD	Das Passwort, das zum Zugriff auf die Redis-Instanz verwendet wird. Dies sollte auf eine zufällige Zeichenkette festgelegt werden, um die Sicherheit zu gewährleisten.
COUCHDB_USER und COUCHDB_PASSWORD	Diese beiden Umgebungsvariablen definieren den CouchDB-Benutzernamen und das Passwort, die zum Zugriff auf den Hauptadministrationsbenutzer verwendet werden. Diese können auch verwendet werden, um auf die Fauxton-Benutzeroberfläche zuzugreifen.

## oder als Composer Datei mit ssl

Dazu ein neues Verzeichnis namens certs im Verzeichnis wo die composer Datei liegt / hinsoll erstellen.

```
mkdir certs
```

Nun ins Verzeichnis gehen und das Zertifikat anlegen.

```
cd /home/duffy/budibasedockertemplate/certs
openssl req -newkey rsa:4096 -x509 -sha256 -days 365000 -nodes -out selfsigned.crt -keyout private.key
```

nun die Fragen beantworten.

DE

und der Common Name als URL.

Bei selbstsigniert eigentlich egal. Hauptsache steht was drin.

Ausgabe:

```
.....++++
.....++++
writing new private key to 'private.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
```

For some fields there will be a default value,

If you enter '.', the field will be left blank.

-----

Country Name (2 letter code) [AU]:DE

State or Province Name (full name) [Some-State]:

Locality Name (eg, city) []:

Organization Name (eg, company) [Internet Widgits Pty Ltd]:

Organizational Unit Name (eg, section) []:

Common Name (e.g. server FQDN or YOUR name) []:budibase.local.lan

Email Address []:

Nun wieder in das Verzeichnis wo die composer-datei liegt zurück gehen.

```
cd ..
```

Nun wieder ein Verzeichnis zurück ins Template Verzeichnis und eine neue nginx Config namens "nginx-proxy.conf" anlegen

```
nano nginx-proxy.conf
```

Inhalt dieser Datei:

Client\_max\_body\_size gibt an wie groß eine Datei zum Upload sein darf. Braucht man größere Größen diese anpassen.

Wenn im nginx dieser Fehler im Log auftaucht:

```
Fehler: client intended to send too large bod
```

Nun der Inhalt

```
events {
    worker_connections 1024;
}

http {
    server {
        listen 80;
        return 301 https://$host$request_uri;
    }

    server {
        listen 443 ssl;
```

```
ssl_certificate /etc/ssl/certs/selfsigned.crt;
ssl_certificate_key /etc/ssl/private/private.key;

location / {
    proxy_pass http://budibase:80;
    proxy_set_header Host $host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto $scheme;
    client_max_body_size 100M;
}
}
```

Die composer Datei:

```
version: "3"

services:
  budibase:
    restart: unless-stopped
    image: budibase/budibase:latest
    expose:
      - "80"
    healthcheck: -> ein healthcheck der 3 mal überprüft ob der dienst läuft wenn nicht wird er als unhealthy.
    markiert. siehe unten bei restart on failure
    test: ["CMD", "curl", "--fail", "http://localhost:80/"]
    interval: 30s
    timeout: 10s
    retries: 3

volumes:
  - budibase_data:/data -> Bezieht sich unten auf das Volume, wo die Budibase Daten gespeichert werden
    sollen, also persistentlaufwerk
```

nginx:

image: nginx:stable

container\_name: nginx-proxy

volumes:

- ./nginx-proxy.conf:/etc/nginx/nginx.conf:ro -> Es können anstatt Verzeichnisse auch Dateien angegeben werden. Hoer nginx config. Im lesen Mdous

- ./certs/selfsigned.crt:/etc/ssl/certs/selfsigned.crt:ro Hier der Öffentliche Schlüssel für ssl auch lesend

- ./certs/private.key:/etc/ssl/private/private.key:ro Hier der Private Schlüssel für ssl auch lesend.

ports:

- "80:80" -> port 80 - weitergelietet auf port 80

- "443:443" -> 443 port auf 443 port

depends\_on:

- budibase -> braucht budibase gestartet

restart: unless-stopped -> wenn oben budibase fehlerhaft ist startet der nginx mit neu

volumes:

budibase\_data: -> name

driver: local

driver\_opts:

type: none

o: bind

device: /home/duffy/budibasedata -> verzeichnis wo die budibase daten liegen auf dem host

## Und Komplette ohne Erläuterung

version: "3"

services:

budibase:

restart: unless-stopped

image: budibase/budibase:latest

expose:

- "80"

healthcheck:

test: ["CMD", "curl", "--fail", "http://localhost:80/"]

interval: 30s

timeout: 10s

retries: 3

volumes:

- budibase\_data:/data

```
nginx:
  image: nginx:stable
  container_name: nginx-proxy
  volumes:
    - ./nginx-proxy.conf:/etc/nginx/nginx.conf:ro
    - ./certs/selfsigned.crt:/etc/ssl/certs/selfsigned.crt:ro
    - ./certs/private.key:/etc/ssl/private/private.key:ro
  ports:
    - "80:80"
    - "443:443"
  depends_on:
    - budibase
  restart: unless-stopped
```

```
volumes:
  budibase_data:
    driver: local
    driver_opts:
      type: none
      o: bind
      device: /home/duffy/budibasedata
```

Nun den container starten/aktualisieren.

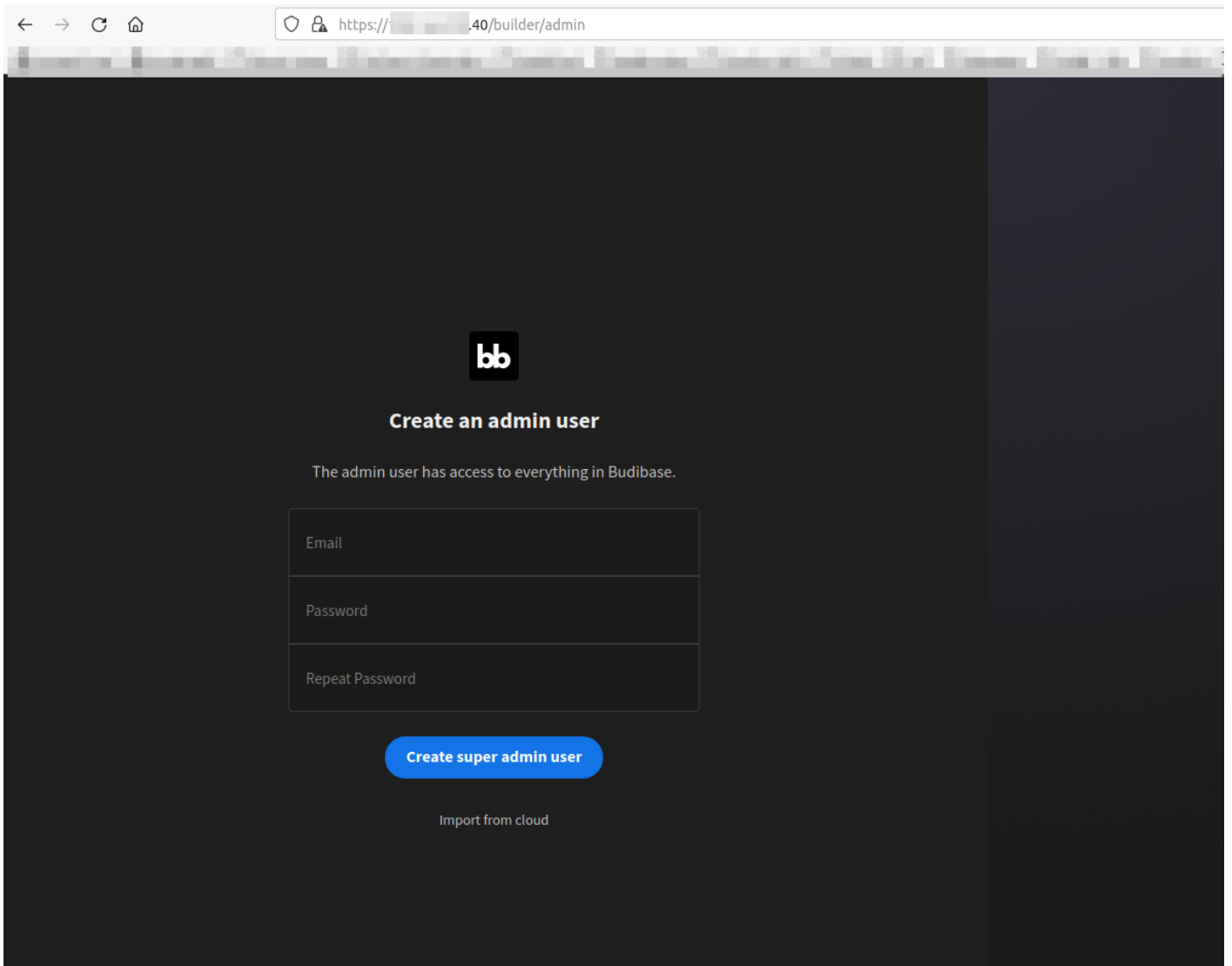
```
docker-compose down
docker-compose up -d
```

Fürs aktualisieren erst down nehmen und dann up.  
Für den ersten start nur up.

```
docker-compose up -d
```

Nun kann ein Webbrowser geöffnet werden, mit https. Das zertifikat beim ersten aufruf natürlich zur ausnahme hinzufügen

[https://ip\\_vom\\_dockerhost](https://ip_vom_dockerhost)



Nun kann ein Admin Benutzer erstellt werden.

# Budibase Internal DB

# Beziehungen Relationships

## Beschreibung:

Ein Nachteil von der Budibase DB ist, das in einigen GUI Elementen wie auch im Automator die Verknüpften Tabellen Felder nicht angezeigt werden. Aber es gibt Gott sei Dank ein Workaround dafür.

Wir wollen in diesem Beispiel bestellungen kunden zu ordnen.

## Ausgangssituation.

In unserem Beispiel haben wir eine Tabelle Kunden und eine Tabelle Bestellungen. Wir haben die Anzahl der Felder so minimalistisch gelassen, weil es hier ums Prinzip geht

Die Tabelle Kunden hat folgende Felder.

AutoID  
Name  
Vorname

Die Tabelle Bestellungen hat folgende Felder

AutoID  
artikel  
anzahl  
preis  
gesamtpreis

Der Gesamtpreis ist vom Typ Formula. Also ein berechneter Wert.  
Die formel dazu heißt

```
{{ multiply anzahl preis }}
```

Beziehung herstellen

Wir gehen auf die Tabelle Kunden und erstellen ein neues Feld mit dem Namen bestellung.  
als Type wählen wir Relationship  
Dann wählen wir die Tabelle aus mit der Verknüpft werden soll.

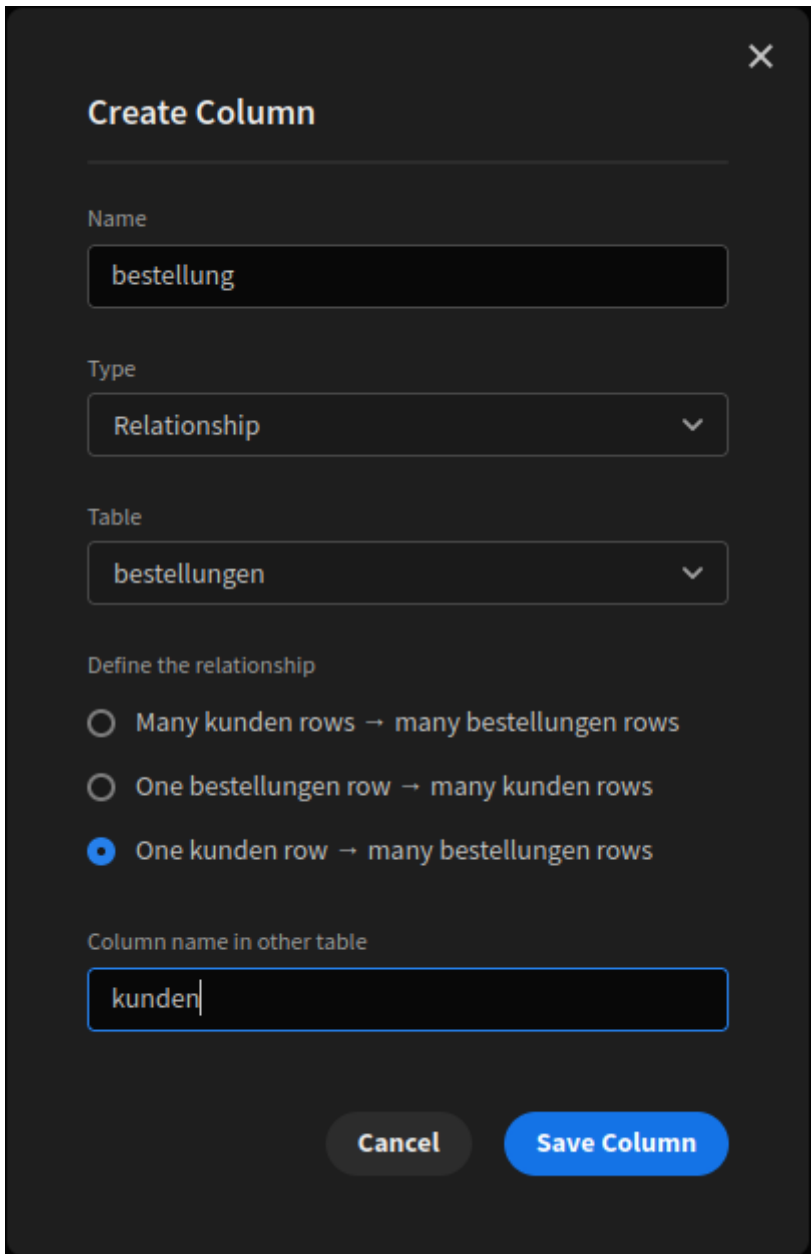
also Bestellungen

Welche Beziehung soll es sein.

Hier kann ein Kunde mehrere Bestellungen, Also one kunden row -> many bestellungen rows  
nun wird auch automatisch ein feld in der tabelle bestellungen angelegt.

Wir müssen nur noch den namen angeben.

Kunden oder kunde macht vielleicht sinn



**Create Column**

Name  
bestellung

Type  
Relationship

Table  
bestellungen

Define the relationship

Many kunden rows → many bestellungen rows

One bestellungen row → many kunden rows

One kunden row → many bestellungen rows

Column name in other table  
kunden

Cancel Save Column

Über add row fügen wir zwei kunden ein.

Max Mustermann

und Anna Schulz

Jetzt sieht unsere Tabelle Kunden so aus.

123 Auto ID	Name	Vorname	bestellung
1	Mustermann	Max	
2	Schulz	Anna	
+ CTRL ↵			

Nun legen wir eine Bestellung an, werden aber gleich sehen das da was nicht stimmt bzw blöd ist

Eine neue Row in bestellungen anlegen.

Artikel intel i5

preis 150

anzahl 2

und nun können wir den kunden auswählen.

und da haben wird, es werden die IDs vom Kunden angezeigt.

Ein bisschen blöd sich die ids zu merken wer nun Max oder Anna ist

123 Auto ID	artikel	anzahl	preis	gesamtpreis	kunden
1	intel i5	2	150	300	+ Search by Auto ID 1 2
+ CTRL ↵					

Grundsätzlich erstell ich noch ein Feld in der tabelle mit dem namen der tabelle und anzeige dran.

Bei Kunden zum Beispiel. kundenanzeige.

Es gibt bei feldern ein Flag das geschaltet werden kann, das dieses als Anzeige Benutzt werden soll, bei Beziehungen.

Wir erstellen ein neues Feld kundenanzeige vom typ Formula

Wir wollen das in diesem Feld der Vorname und Nachname zusammen angezeigt wird.

Das ist die Formel. Über das Blitzsymbol kann man schnell die Formel zusammen klicken.

```
{{ Vorname }} {{ Name }}
```



## Create Column

Name

kundenanzeige


Type

Formula



Required

Use as table display column

Formula type 

Dynamic



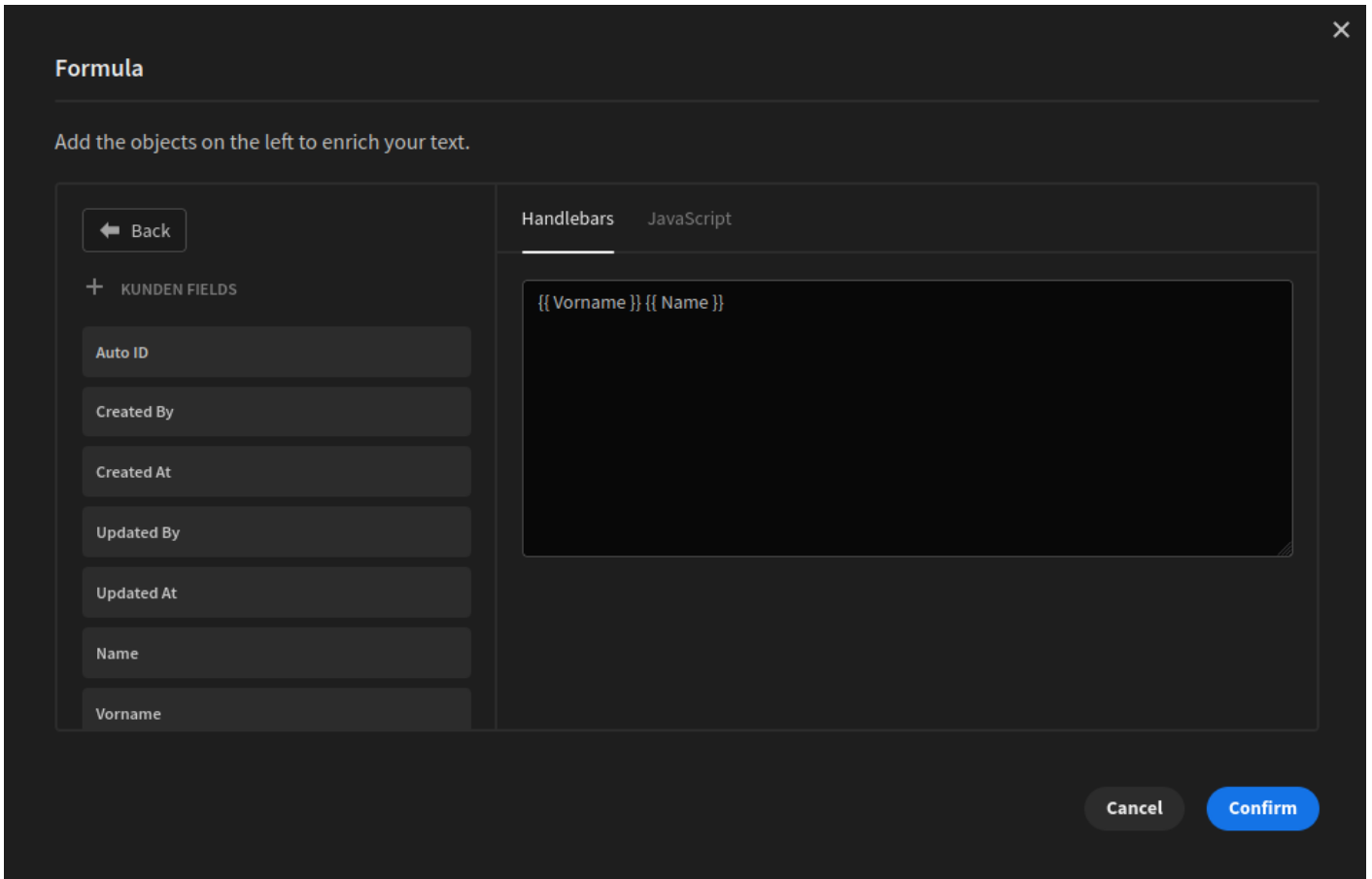
Formula

{{ Vorname }} {{ Name }}



Cancel

Save Column



Und nun kommt der Clou. Den schieber use as table display coloumn aktivieren.  
In einer tabelle kann das immer nur ein Feld sein. Würded Ihr ein anderes Feld dies einhaken,  
würde hier der Haken wieder rausgehen.

×

## Create Column

---

Name

Type

Formula ▼

Required

Use as table display column

Formula type ⓘ

Dynamic ▼

Formula

⚡

Cancel Save Column

Nun gehen wir wieder in unsere Bestellungen und wählen den Kunden aus.  
Tadaaa, jetzt ist es lesbarer als nur die IDs.

✕

## Edit Row

---

Artikel

Anzahl

Preis

Kunden

- Choose an option*
- Anna Schulz ✓
- Max Mustermann

Nun wollen wir natürlich eine App draus bauen.  
Allerdings kann Budibase nicht auf die verlinke id zugreifen.  
Wir erstellen uns zwei screens.  
Eine mit ner Tabelle für die kunden.  
Dazu Komponente table block hinzufügen  
und die Tabelle kunden auswählen

GENERAL

Name

Title  ⚡ ⌵

---

TABLE

Data  ⌵

Columns

Sort by  ⌵

Sort order  ⌵

Size  ⌵

Scroll limit  ⌵

Paginate

Quiet

Compact

Row selection

Row selection is only compatible with internal or SQL tables

Filtering

Empty text  ⚡ ⌵

Search fields  ⌵

Only the first 5 fields will be used

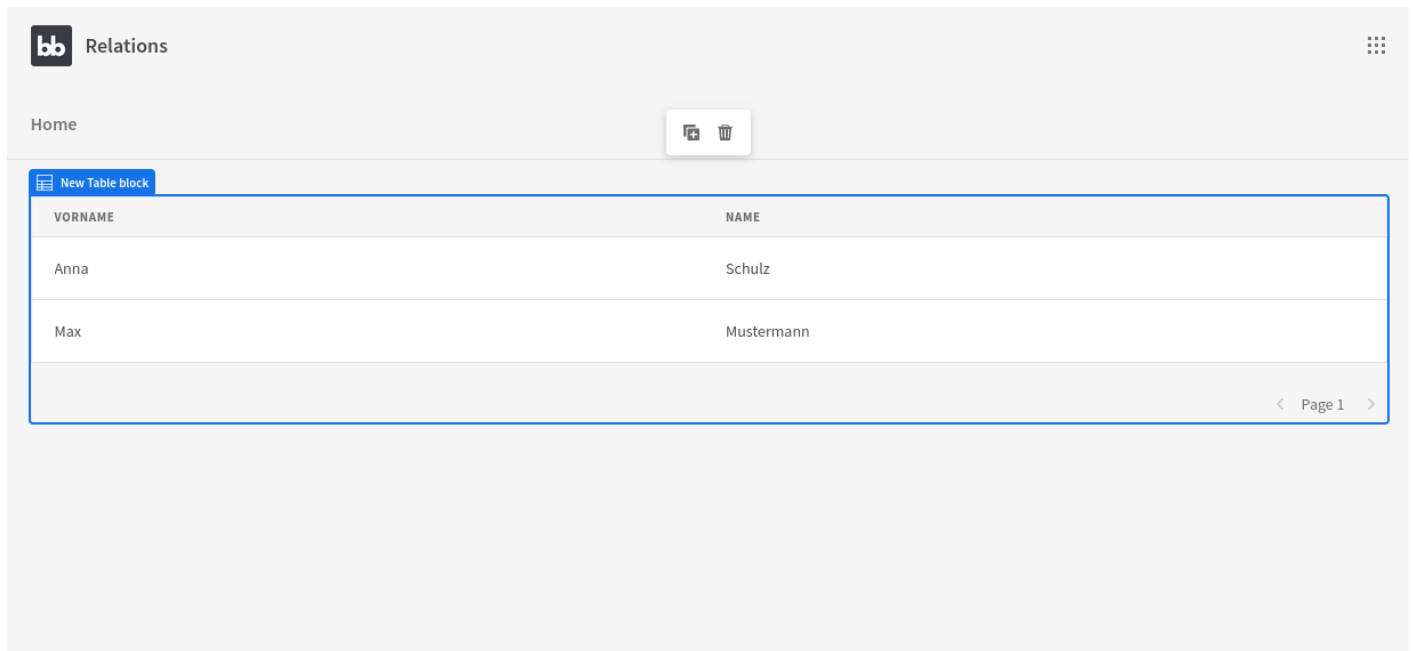
Jetzt konfigurieren wir noch die columns. Denn die Bestellungen und die Kundenzeile wollen wir da nicht drin haben.

In unserem Beispiel sollen die Bestellungen über einen Button geöffnet werden als modal. So dass alle Bestellungen für den Kunden auf einen Blick sind.

Und das ist der Grund warum in Budibase ein Workaround nutzen muss, denn Relationship Felder werden nicht angezeigt.

Aber dazu später.

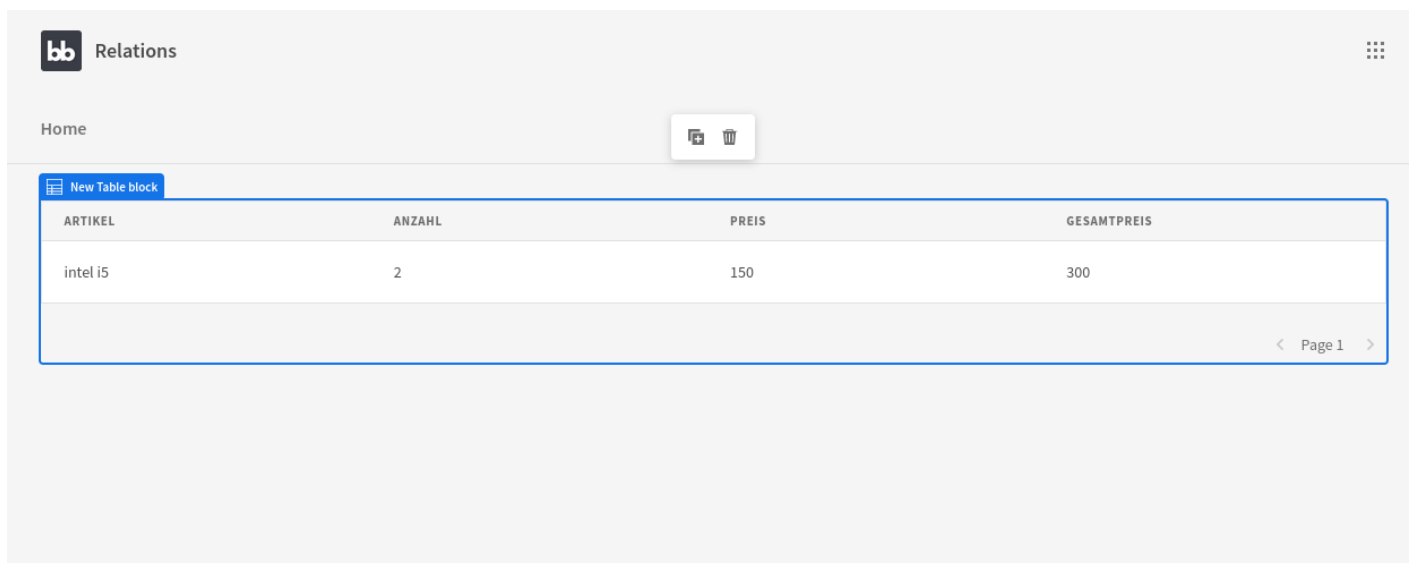
Unser Screen sieht jetzt so aus



Nun erstellen wir einen zweiten screen mit einer Tabelle die die Bestellungen beinhaltet. ich hab den Screen Details genannt.

Wieder einen Tabell hinzugefügt und die Spalten angepasst.

Die Spalte Kunde brauchen wir nicht, da ja nur bestellungen von einem kunden aufgerufen werden. Unser Screen sieht dann so aus.



Nun gehen wir in den ersten screen zurück und erstellen einen Button in der Tabelle



Table

	NAME	VORNAME
New Button	Muster...	Max

< Page 1 >

Diesem Button geben wir folgende Eigenschaften. Dazu klicken wir den Button an.  
Nun geben wir den button einen Namen buttonBestellungen anzeigen  
Dann Anzeigetexte Betsellungen öffnen.

GENERAL

Name

Text  ⚡ ⌵

Variant  ⌵

Size  ⌵

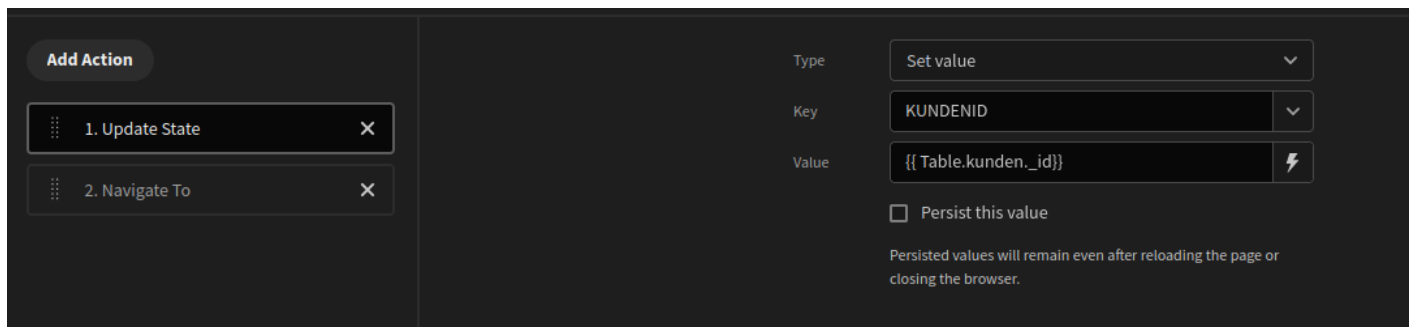
Quiet

Disabled

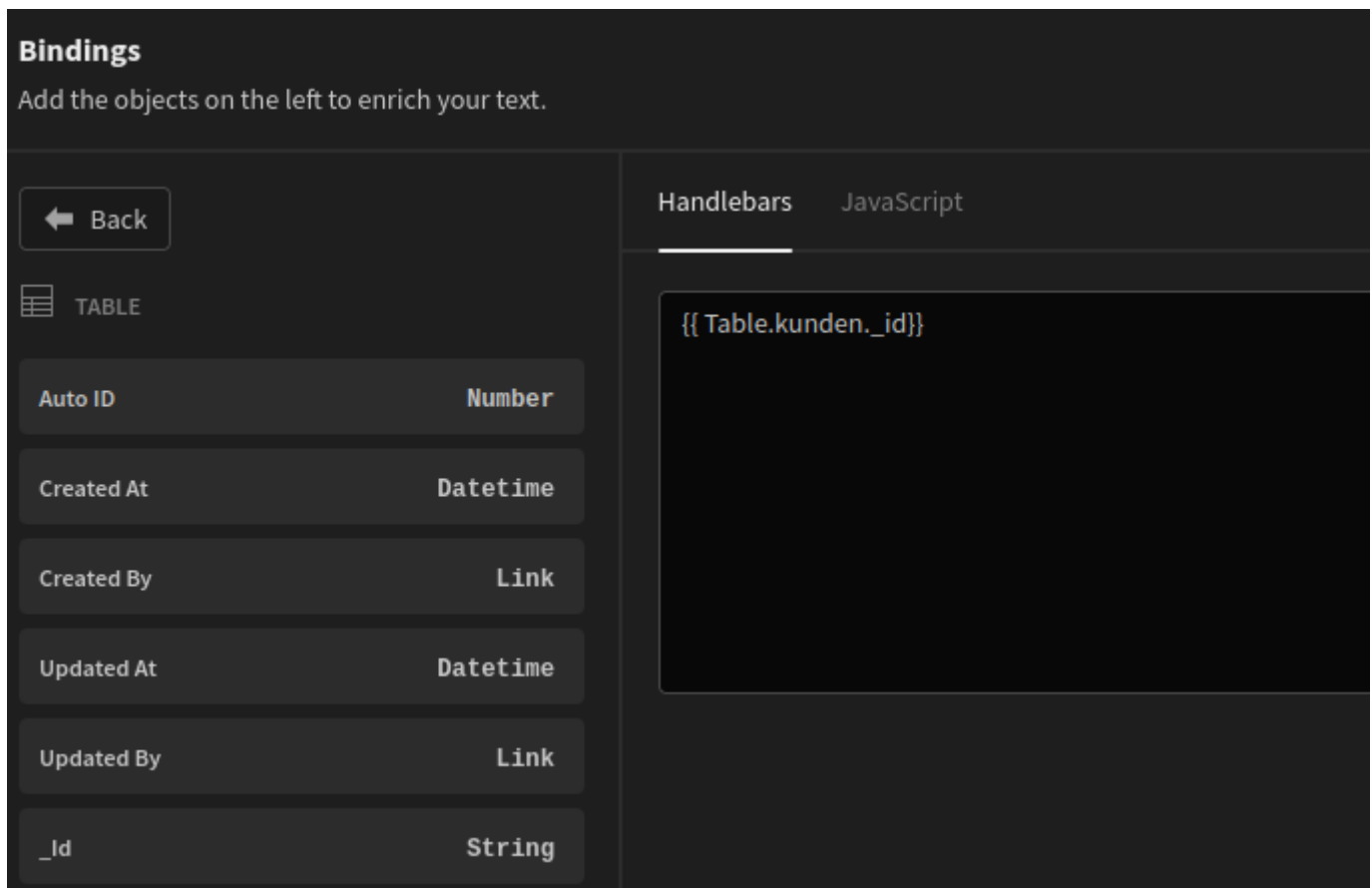
On click **No actions set**

Nun auf define actions klicken.  
Dort Update State Action hinzufügen

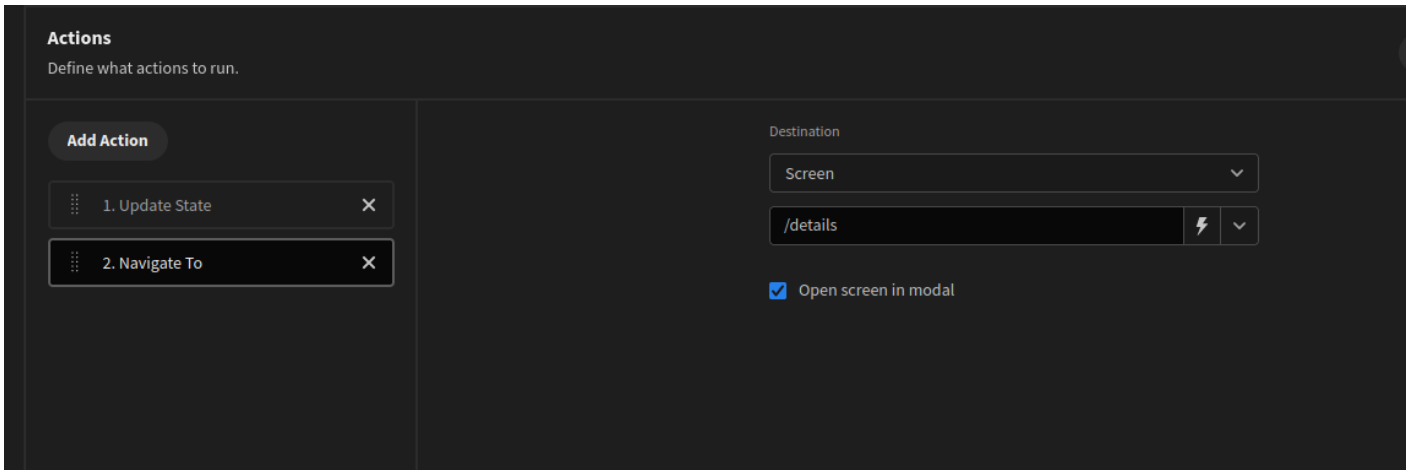
Eine Variable mit KUNDENID angeben und dann auf den Blitz klicken



Nun als Wert die ID von der Tabell Kunden angeben.  
Über den Asistenten -> Table -> \_id  
schnell ausgewählt



Nun ein navigate als zweite Aktion hinzufügen. Dort denn screen details auswählen:  
ob im Vollbild oder Modal geöffnet werden soll bleibt euch überlassen.  
ich find das Modal aber an Budibase ja so genial, deshalb lass ich Modal drin.



So nun kommt der eigentliche Workaround, aber erstmal schauewn wir uns an, wass denn nicht funktioniert.

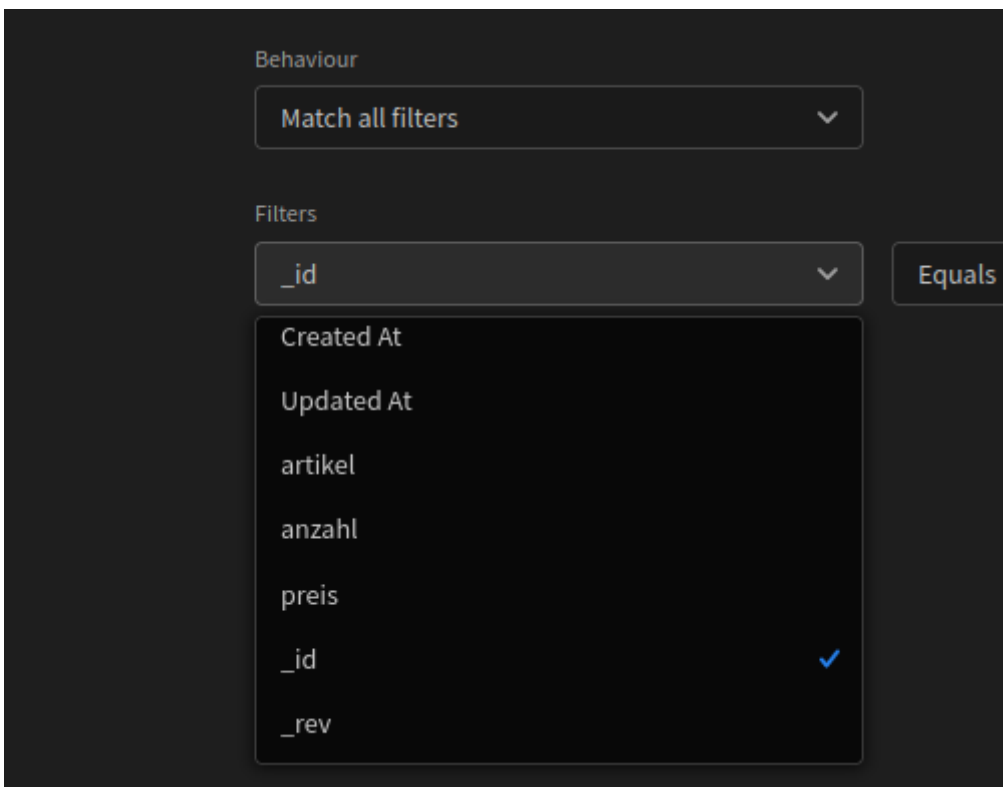
Dazu gehen wir in den Screen details.

klicken unseren Dataprovider an

und gehen auf define Filters.

Denn hier wollen wir ja nur die Betsellungen vom kunden mit der ID die vir in die variable gesetzt haben lesen.

Aber hier ist das Problem kein Feld kunden in der auswahl!!!!



Da behlfen wir uns eines kleinen tricks.

Wir gehen die Tabelle Bestellen zurück und fügen ein neues Feld hinzu.

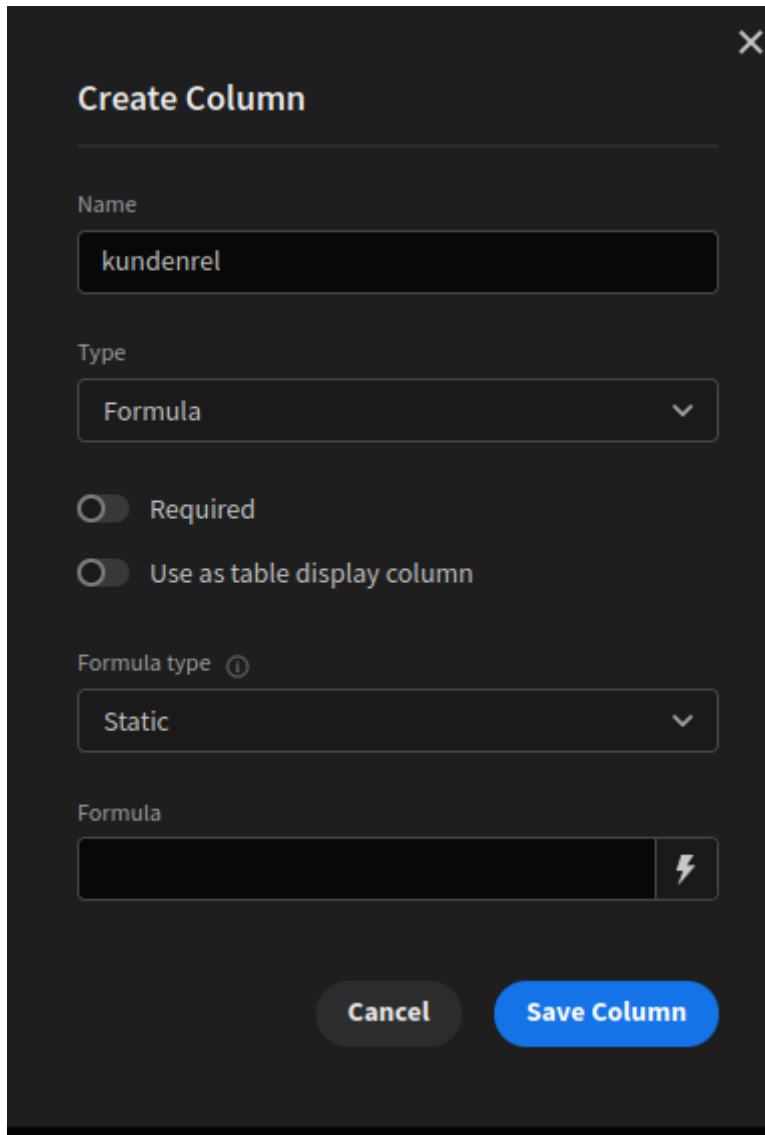
Dieses nennen wir kundenrel.

Ich mach das immer so, das ich in einer tabelle wo ich ein relationsfeld habe, dieses feld nochmals erstelle allerdings mit rel am ende und vom typ formula. Dann nur static und folgende Formel rein. wichtig unter javascript!!!

wichtig ist auf die Groß und Kleinsschreibung des Feldnamens zu achten.  
Deshalb schreib ich alle Feldnamen in tabellen immer alles klein

```
return $("kunden") ? $("kunden").map(x => x._id) : ""
```

Nun in der Tabelle bestellungen das neue Feld erstellen mit Static



**Create Column**

Name  
kundenrel

Type  
Formula

Required

Use as table display column

Formula type ⓘ  
Static

Formula

Cancel Save Column

Nun nachdem auf dem Blitz geklickt wurde auf Javascript klicken und die oben genannte formel angeben.

wichtig ist auf die Groß und Kleinsschreibung des Feldnamens zu achten.  
Deshalb schreib ich alle Feldnamen in tabellen immer alles klein

### Formula

Add the objects on the left to enrich your text.

Text    JavaScript

```
1 return ·$( "kunden" ) ·? ·$( "kunden" ) ·.map (x ·=> ·x ·.id) ·: ·" "
```

Search for bindings

- + Adressen Fields >
- + Created By Relationships >
- + Updated By Relationships >
- + Kunde Relationships >
- + Vertragerechnungsadressemeldeadresse >

⚡ Add available bindings by typing \$ or use the menu on the right

Cancel    Confirm

Nun sieht das ganze so aus und speichern

### Create Column

Name

Type

Formula

Required

Use as table display column

Formula type ⓘ

Static

Formula

(JavaScript function)

⚡

Cancel Save Column

Nun haben wir einen pointer auf die ID.

	kunden	kundenrel	Cre
1		ro_ta_3bcdd0e8f47847e7a36a3	info@h

Jetzt legen wir noch eine Bestellung an, damit wir zwei Datensätze haben. Nachdem wir diese angelegt haben, sind in unserer Datenbank zwei Kunden und zwei Bestellungen. Wir haben die Bestellungen jeweils einem Kunden zugeordnet. Da wir den Filter noch definiert haben werden beide angezeigt

bb Relations ☰

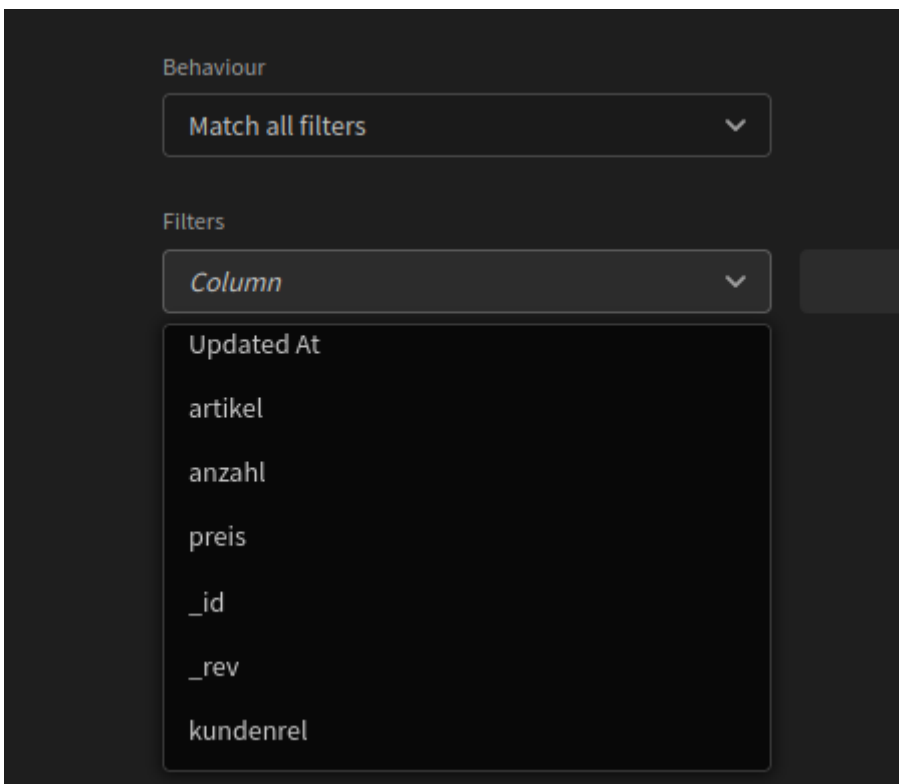
Home Details

ARTIKEL	PREIS	ANZAHL	GESAMTPREIS
intel i5	150	2	300
Desktiopschrank	250	1	250

< Page 1 >

Nun wieder auf define filzter im Dataprovider

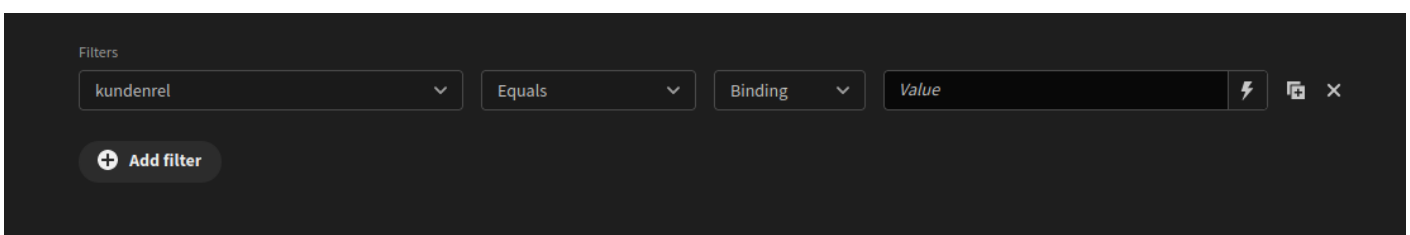
Jetzt steht kundenrel in der Auswahl. Jippie



Nun die Formel vervollständigen.

Dazu equal auswählen und bindings, dann wird der blitz freigeschaltet.

Dann auf den blitz klicken

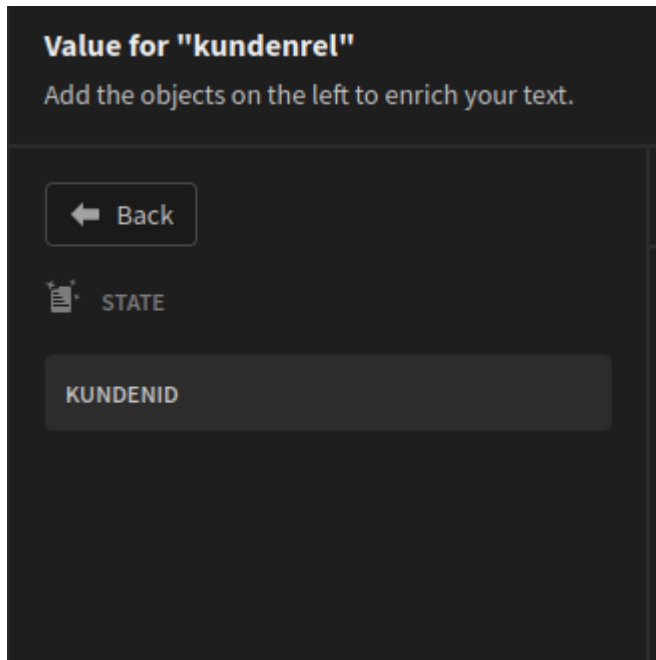


unter State werden alle Variablen angezeigt die angelegt wurden.

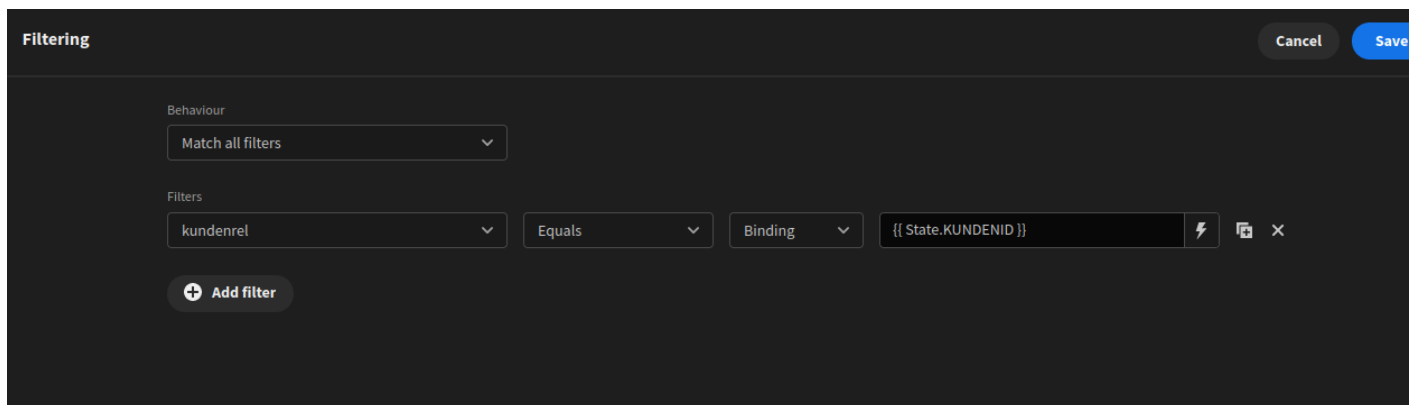
Nun auf State klicken und die Variable KUNDENID raussuchen.

Wenn wir nachher noch mehrere Screens haben, wir können diese Variable immer wieder überschreiben, beim button. Denn es kann ja immer nur ein fenster gleichzeitig angezeigt werden. Deshalb ist in meinem gesamten Programm auch KUNDENID verwendbar.

Weil wo ichs brauche schreib ich die neue ID rein, über die Buttons im Home screen oder wo auch immer ich das brauche.

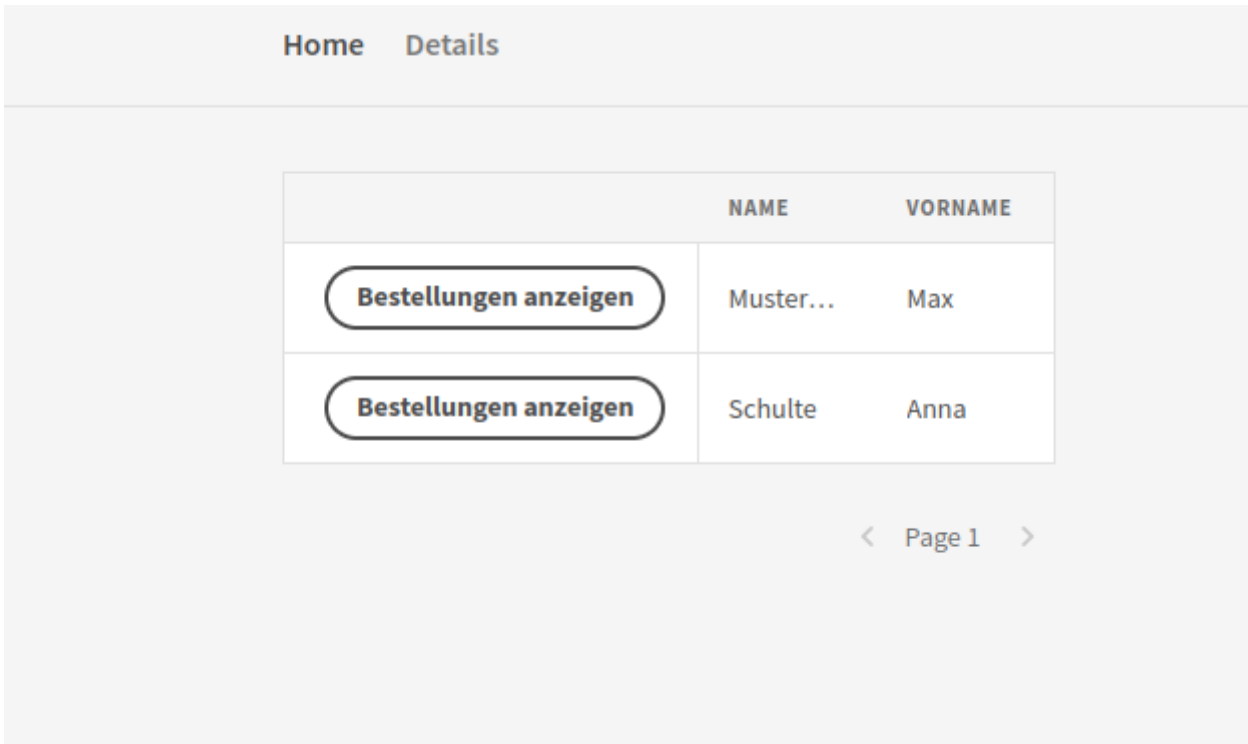


Nun sieht die Formel so aus, noch speichern fertig.

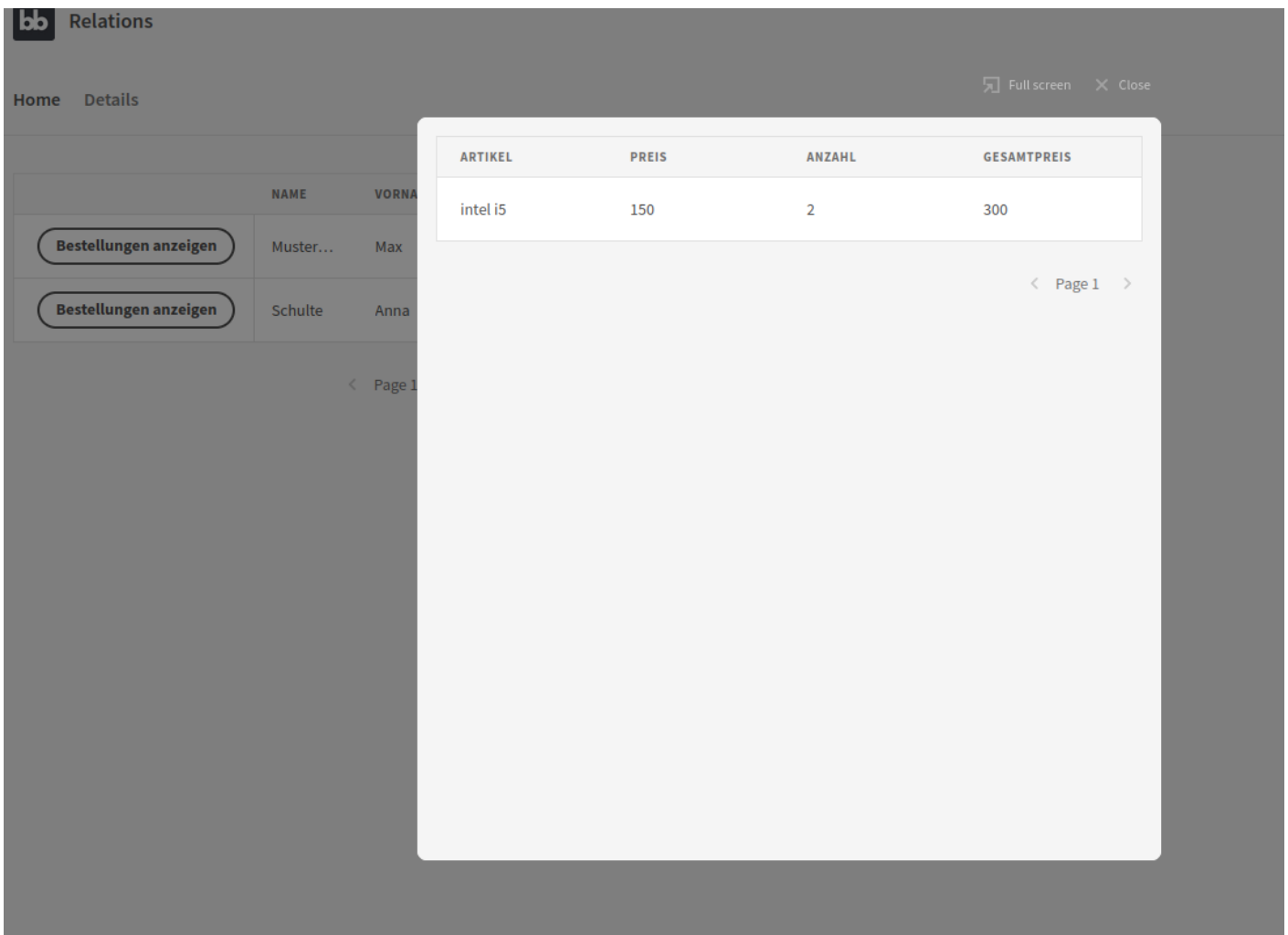


Nun können wir auf preview gehen und auf den Home screen

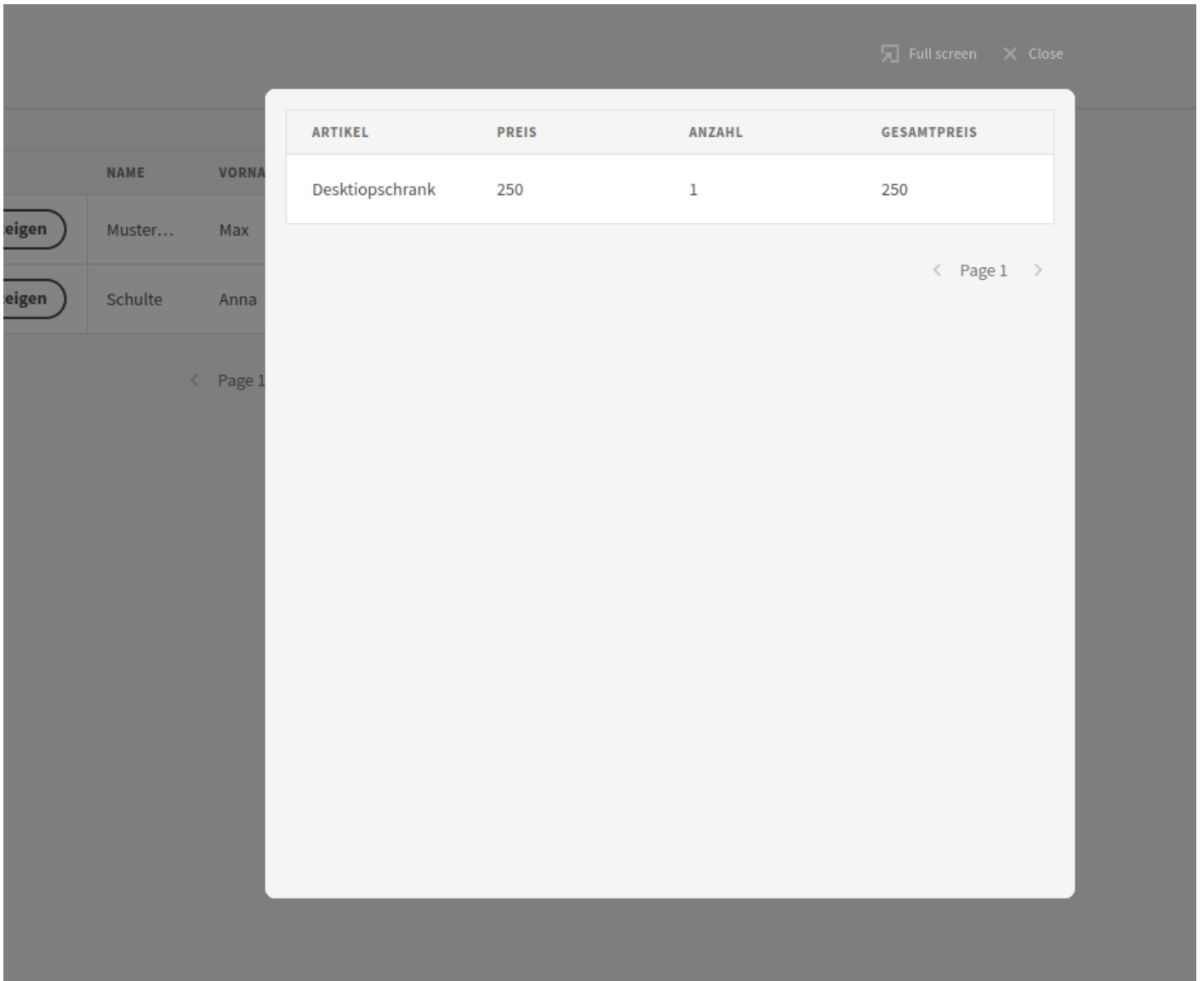
Nun bei dem jeweiligen Kunden auf bestellungen öffnen klicken



Und eh voila, ein Modal mit den Bestellungen für den Kunden, in diesem Falle Max



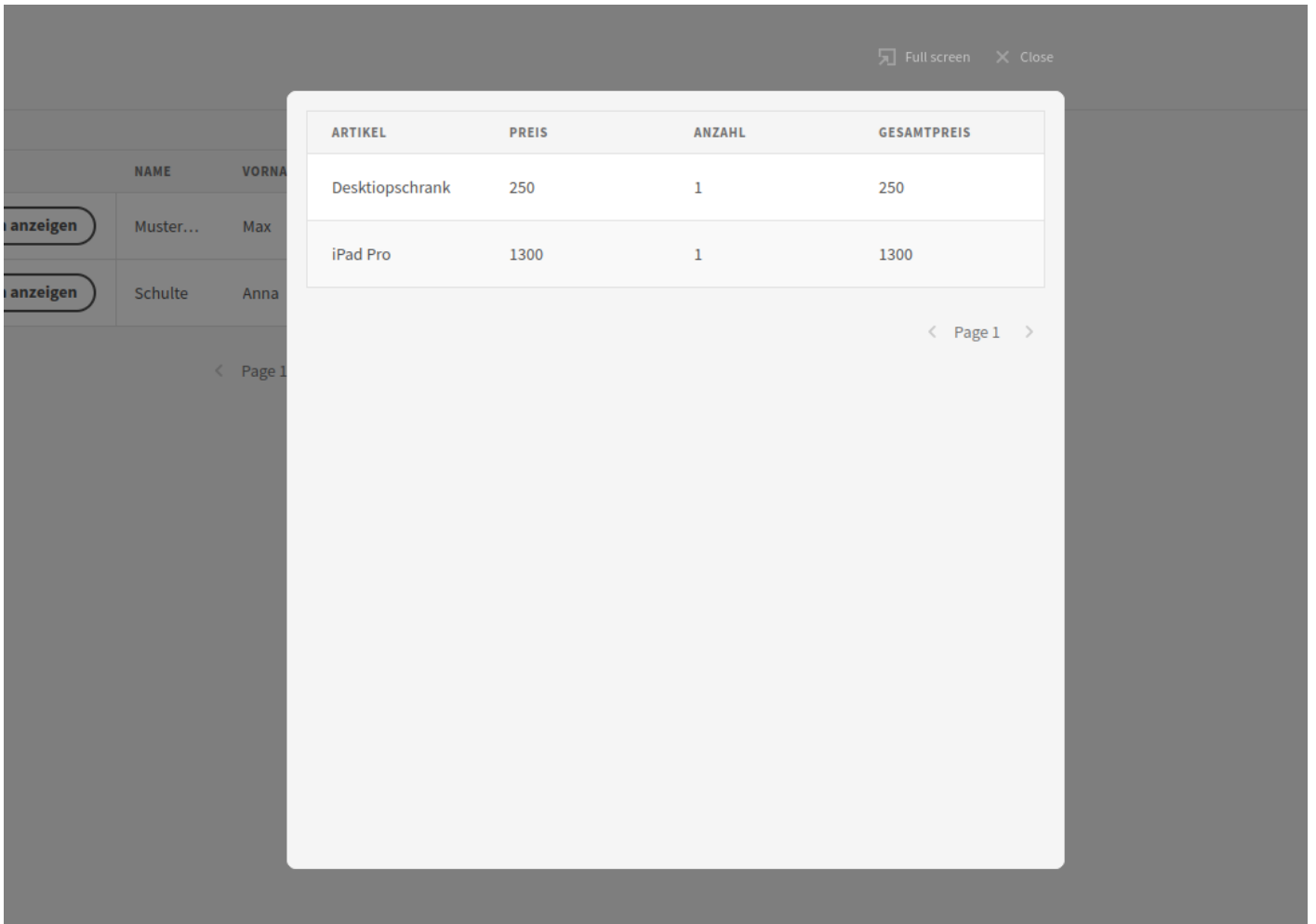
Und hier Anna



Wir legen noch eine Bestellung für Anna an.

Dazu wieder in die Data wechseln und eine row anlegen in bestellungen und Anna zuordnen.

Nun hat Anna zwei Bestellungen.



Fertig, kleiner Trick große wirkung, mit dem extra Feld in das die id kopiert wird.

# Budibase GUI Elemente

# Ein Option Picker als Relationship Picker zum Filtern zweckentfremden

## Beschreibung:

Leider gibt es in Budibase keinen Relationship Picker (Dropdownmenü wo man die Liste nochmals filtern kann.

Zum Beispiel Zeige aus der Liste nur einträge von Kunden so und so an.

In Unserem Beispiel haben wir folgende Tabellen.

Kunden

Adressen

Verträge.

Bei Kunden und adressen reicht der Relationship picker noch aus, da wir eine Tabelle filtern.

Sprich wir filtern die Tabelle Adressen nach einem Kunden.

Also einfache Verschachtelung. Kunde -> Adressen.

Spannend wird es jetzt mit den Verträgen. Denn ein Vertrag hat mehrere Relationen.

Ein Vertrag hat eine Relation zu Kunden und eine Relation zu den Adressen.

Denn ein Kunde kann mehrere Adressen haben, genauso wie ein Kunde mehrere Verträge haben kann.

Mit dem normalen Relationship Picker, werden dann wenn beim Vertrag die Adresse ausgewählt werden kann, leider alle Adressen angegeben, von allen Kunden.

Wir möchten aber gerne das nur die Adressen von dem Kunden dessen Vertrag gerade bearbeitet wird aufgelistet wird.

Das ganze ist ein wenig tricky, aber es geht.

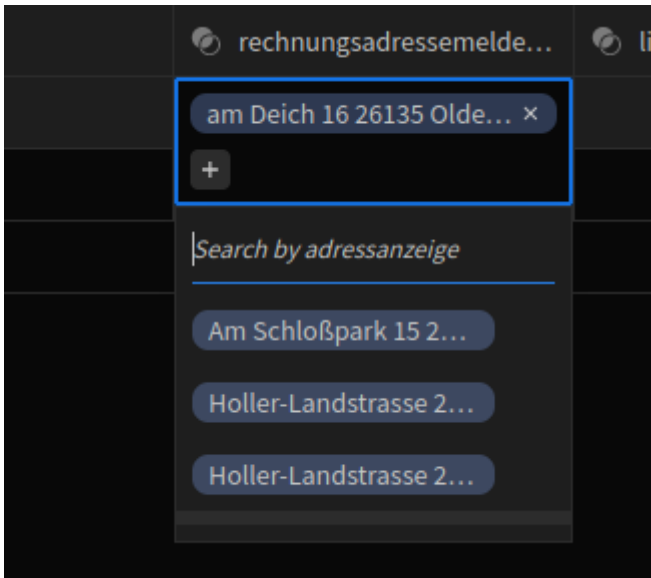
Hier zur veranschaulichung aus der Tabelle Verträge.

Hier sind zwei Verträge.

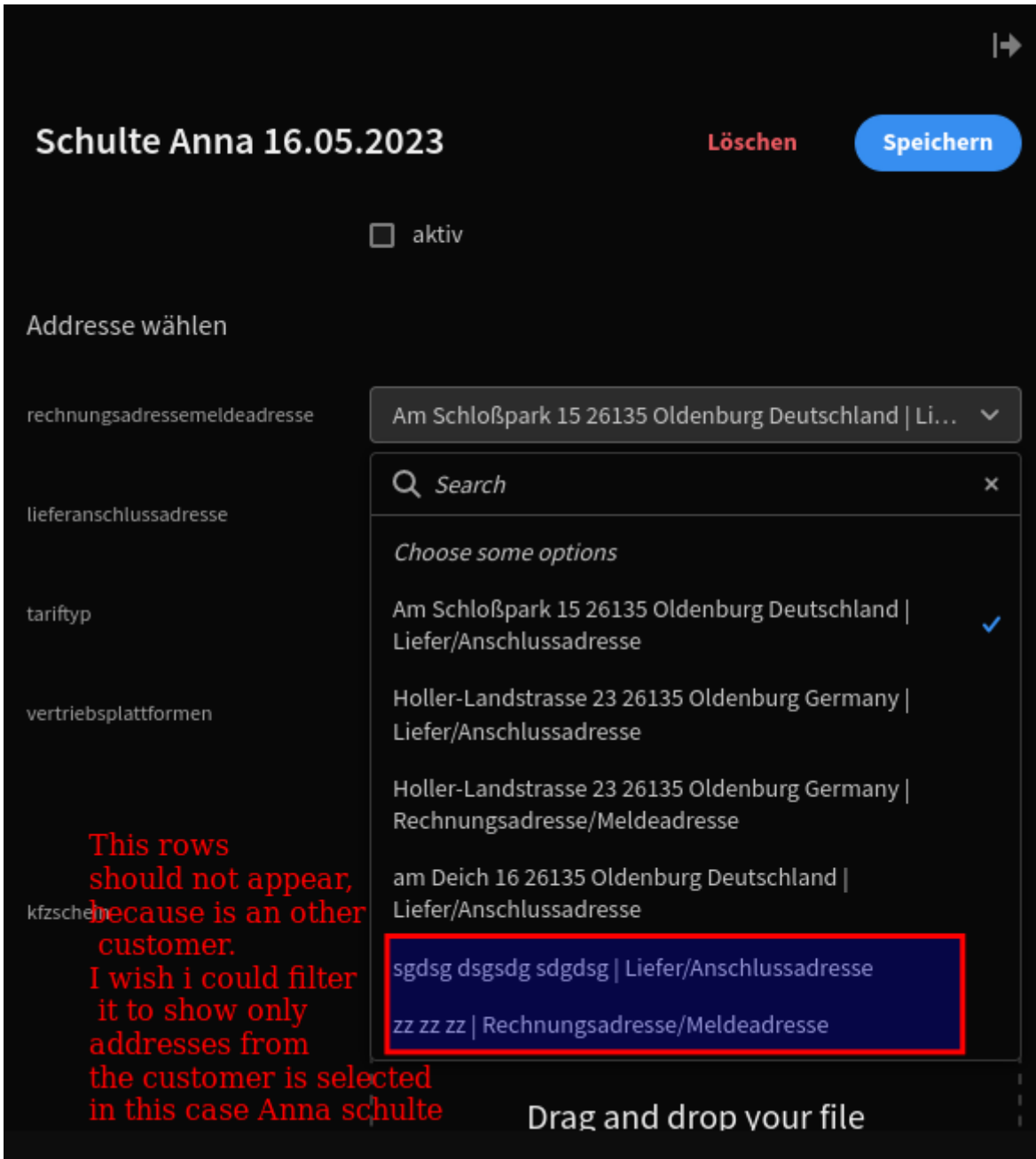
Von Anna die Rechnungsadresse am Deich und von John die Holler Landstraße.

	rechnungsadressemelde...	lieferanschlussadresse	kunden
	am Deich 16 26135 Olden...		Schulte Anna 16.05.2023
	Holler-Landstrasse 23 261...		example.com John Doe 03...

Klicken wir im Data view auf den Eintrag um die Straße auszuwählen, werden uns alle angezeigt. Dieses Verhalten wäre im Designview mit nem Relationship picker genau das selbe.



Hier dann in einem Relationship Picker im Design view, also unserer App / Screen



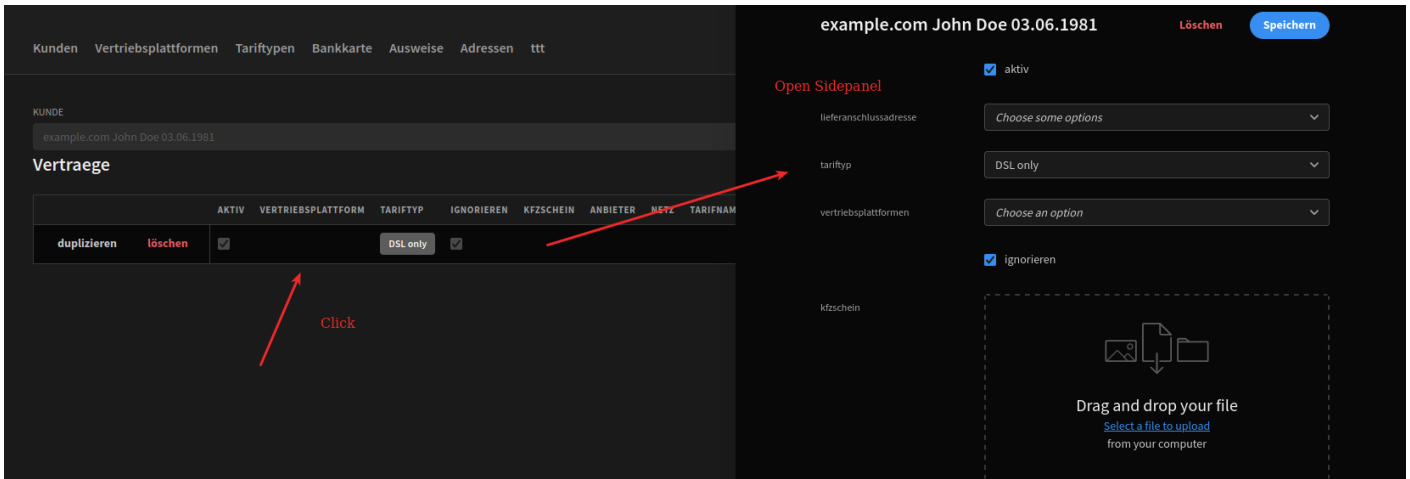
Die letzten beiden Adressen sollten nicht angezeigt werden, da diese nicht zu Anna gehören. Da wir dies aber mit dem Reaktionspicker nicht filtern können, müssen wir uns etwas anderes überlegen.

## Der Workaround

Dazu gehen wir in das bearbeiten Sidepanel / Screen oder hinzufügen Sidepanel / screen wo wir einen Vertrag editieren.

Ich habe in meinem Beispiel ein Sidepanel.

Wenn auf den Vertrag geklickt wird, öffnet sich ein Sidepanel



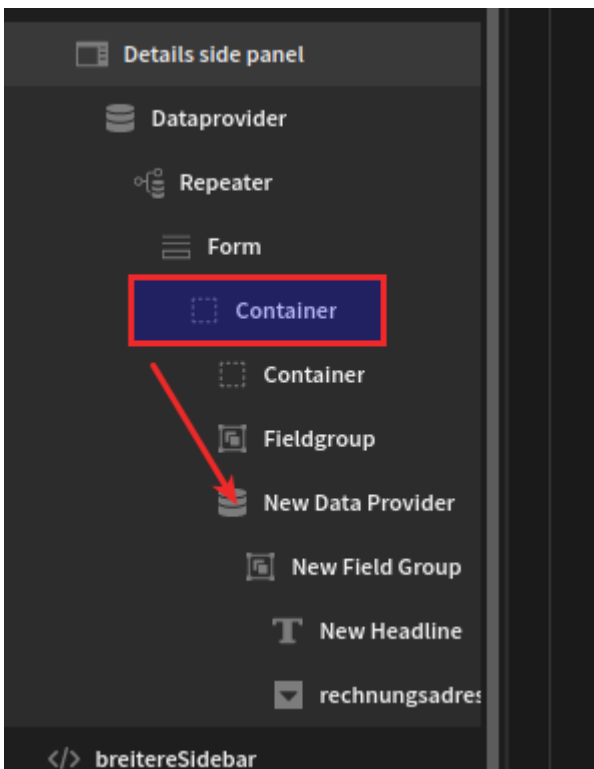
Nun in den Data view Modus. In das Sitepanel gehen.

Nun im Container einen Neuen Data Provider hinzufügen.

Warum genau an der Stelle, damit sich später unser Option Picker dem Sidepanel von der Größe anpasst.

Diesen Data Provider schieben wir dann ganz nach unten (STRG+Pfeilstate drücken zum verschieben)

So das es so aus sieht. Die Fieldgroup und den Picker gibt es noch nicht. Den erstellen wir gleich. Ist aber schon auf dem Bild.



Nun klicken wir auf den New Data Provider

Wir wählen als Daten unsere Tabelle Adressen aus.

**New Data Provider**

Settings Styles Conditions

GENERAL

Name

Data

Filtering

Sort column

Sort order

Limit

Paginate

Pagination is only available for data stored in tables

Klicken dann auf define Filter.

Wir wählen als feld kundenrel aus, oder wie auch immer ihr das Feld genannt habt, wenn ihr eine interne DB verwendet.

Wenn Ihr mysql verwendet wird eure direkte relationsfeld zum kunden angezeigt, die ihr dann warscheinlich kunden nennt.

Da ich aber die interne DB verwende hab ich ein zweites Feld als pointer. Dieses wähle ich und dann noch bindings auswählen.

Wer mehr zu den Workaround wissen will : [Beziehungen Relationships](#)

Behaviour

Filters

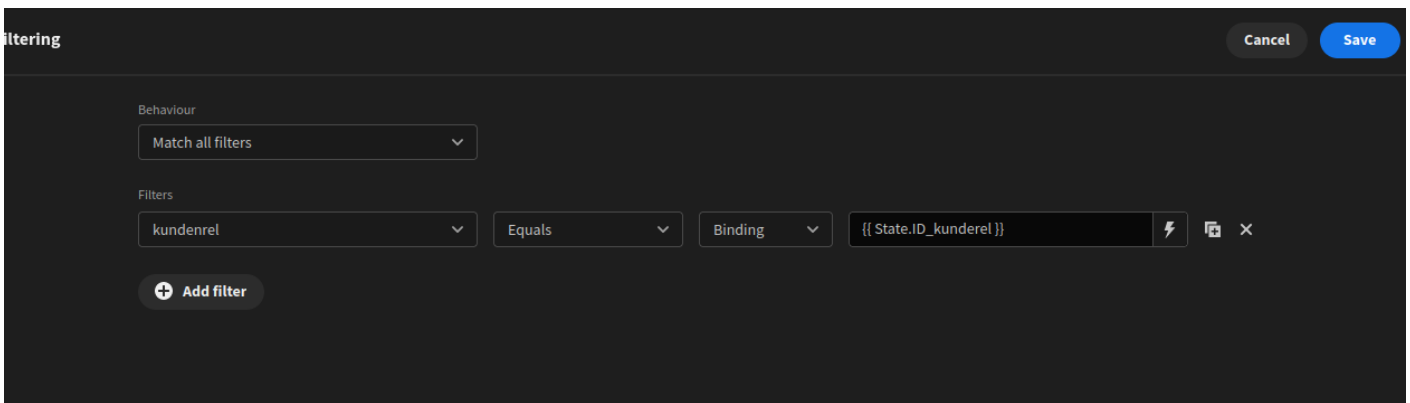
Equals

Binding

Nun auf den Blitz klicken und euren State für die kundenid raussuchen die ja schon vorher gespeichert war, da Ihr ja nur Verträge von Kunde XY angezeigt bekommt. So filtern wir dann auch die Adressen nur nach diesem Kunden durch die Stateid

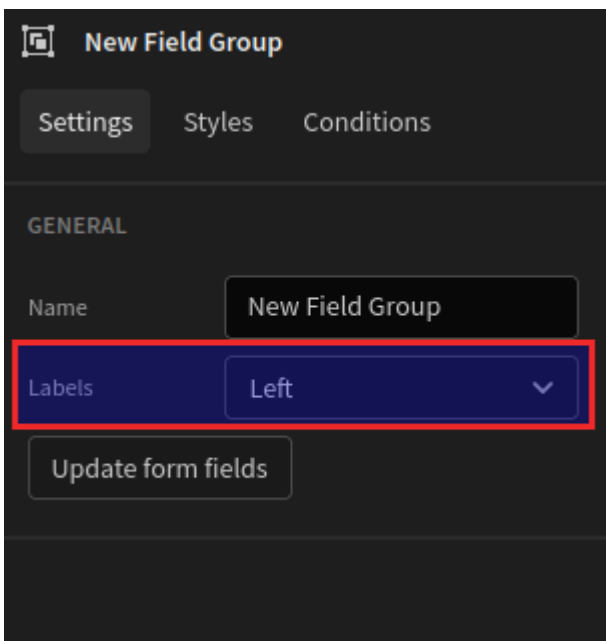


So sieht dann der fertige Filter aus.  
Speichern und fertig.



Somit wäre unser filter fertig.

Nun noch eine Fieldgrupper unterhalb des neuen Data Provider anlegen, damit sich die Elemnet die in die Fieldgroup dann anpassen. Hier wählen wir noch aus, das Labels also beschriftungen auf der Linken Seite seien sollen.



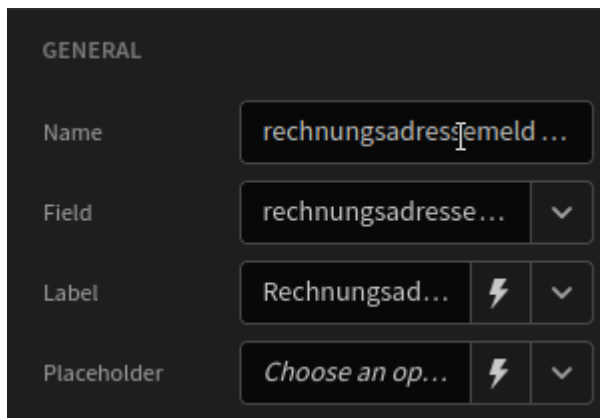
Nun erstellen wir in dieser Group unseren Option Picker.  
Fangen wir mit den ersten 4 Feldern an:

Name : name der das Steuerlement bennent, aht keine Auswirkung auf Tabellenfelder etc. Nur für die reine GUI

Field : Manuell per Hand eintragen. Und zwar erstellen wir dadurch ein neues feld. Name beliebig, der Name darf nur nicht in der Tabelle Verträge vorhanden sein.

Denn das wird ein neues Formularfeld, das uns zum zuweisen in der Tabelle zur verfügung steht. ich habe es rechnungsadressepicker genannt. Dann weis man gleich das ist das Pickerfeld.

Label: Das ist die Beschreibung in der GUI vom Picker



The image shows a configuration form with a dark background and light text. The form is titled "GENERAL" and contains four rows of configuration options:

- Name:** A text input field containing "rechnungsadressemeld ...".
- Field:** A dropdown menu with the text "rechnungsadresse..." and a downward arrow.
- Label:** A dropdown menu with the text "Rechnungsad...", a lightning bolt icon, and a downward arrow.
- Placeholder:** A dropdown menu with the text "Choose an op...", a lightning bolt icon, and a downward arrow.

Nun geht es weiter

Beim Typ kann man auswählen Selct (Also llste) oder Punkte (Wir nehmen Liste)

Default value machen wir zum Schluss, hier eben überspringen

Options source, wählen wir Data Provider aus.

Nun können wir unter Options provider unseren Data Provider auswählen, den eben gerade erstellten

Unter Label coulmn können wir aus der Tabelle Adressen auswählen, welche Spalte benutzt werden soll das zur Anzeige dient.

In diesem Fall für uns die Adressen, denn die IDS kann ja keiner Lesen.

Und value Column, das ist das was tatsächlich der Wert ist, sprich die Variable hinter dem Picker. In unserem Fall die `_id`. Denn wir wollen ja die `_id`, in dem Feld Rechnungsadresse im der Tabelle Verträge speichern.

Type	Select
Default value	{{ Repeater.ve... ⚡
Autocomplete	<input checked="" type="checkbox"/>
Alphabetical	<input checked="" type="checkbox"/>
Disabled	<input type="checkbox"/>
Options source	Data provider
Options provider	New Data Provider
Label column	adressanzeige
Value column	_id
Validation	Configure validation

Nun noch der Default value.

Wir wollen wenn der Wert in der Tabelle Verträge vergeben ist, das im Picker dieser dann schon drin ist.

Sonst ist mit aktualisieren eines Datensatzes ja blöd, man müsste sich das sonst ja vorher merken und wieder neu ausfüllen.

Dazu klicken wir auf den Blitz:

Und wählen:

Repeater

SEARCH

🔍 Search ✕

☰ FormMainTable >

🗄️ Dataprovider >

**🔗 Repeater >**

☰ Form >

Dann die Rechnungsadresse (wichtig den Link und nicht die rel falls intern DB verwendet wird)

### Bindings

Add the objects on the left to enrich your text.

Lieferanschlussadresse	Link
Marktlokationsid	String
Netz	Options
Rechnungsadressemeldeadresse	Link
Rechnungsadressemeldeadresserel	Formula
Rufnummerzumtarif	Link

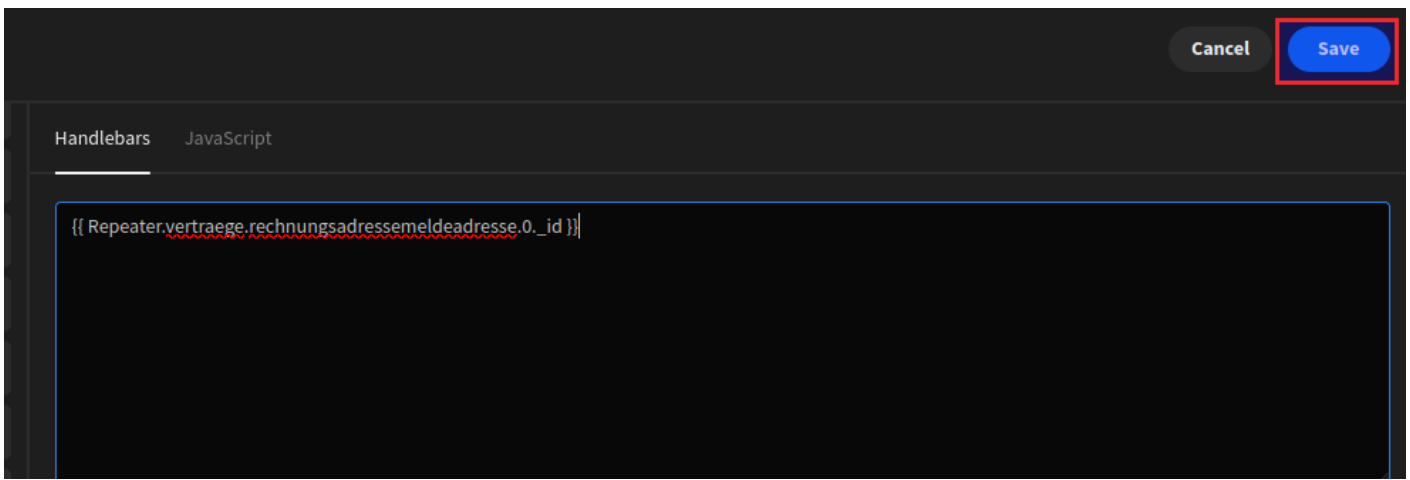
Nun sieht das so aus:

```
Handlebars  JavaScript
{{ Repeater.vertraege.rechnungsadressemeldeadresse }}
```

Jetzt müssen wir noch per Hand hinter dem Feldnamen `.0._id` hinzufügen. Denn das ist ein Array/Datensatz und wir wollen nur das ID Feld nicht den ganzen Datensatz. Denn dadurch wird dann im Picker der Wert gesetzt der zur Zeit in dem Datensatz in Verträge angeben ist. Ist n och keine Adresse angeben, wird glücklicher weise Choose angezeigt.

Dann siesht das ganze so aus, nun auf save klicken:

```
{{ Repeater.vertraege.rechnungsadressemeldeadresse.0._id }}
```

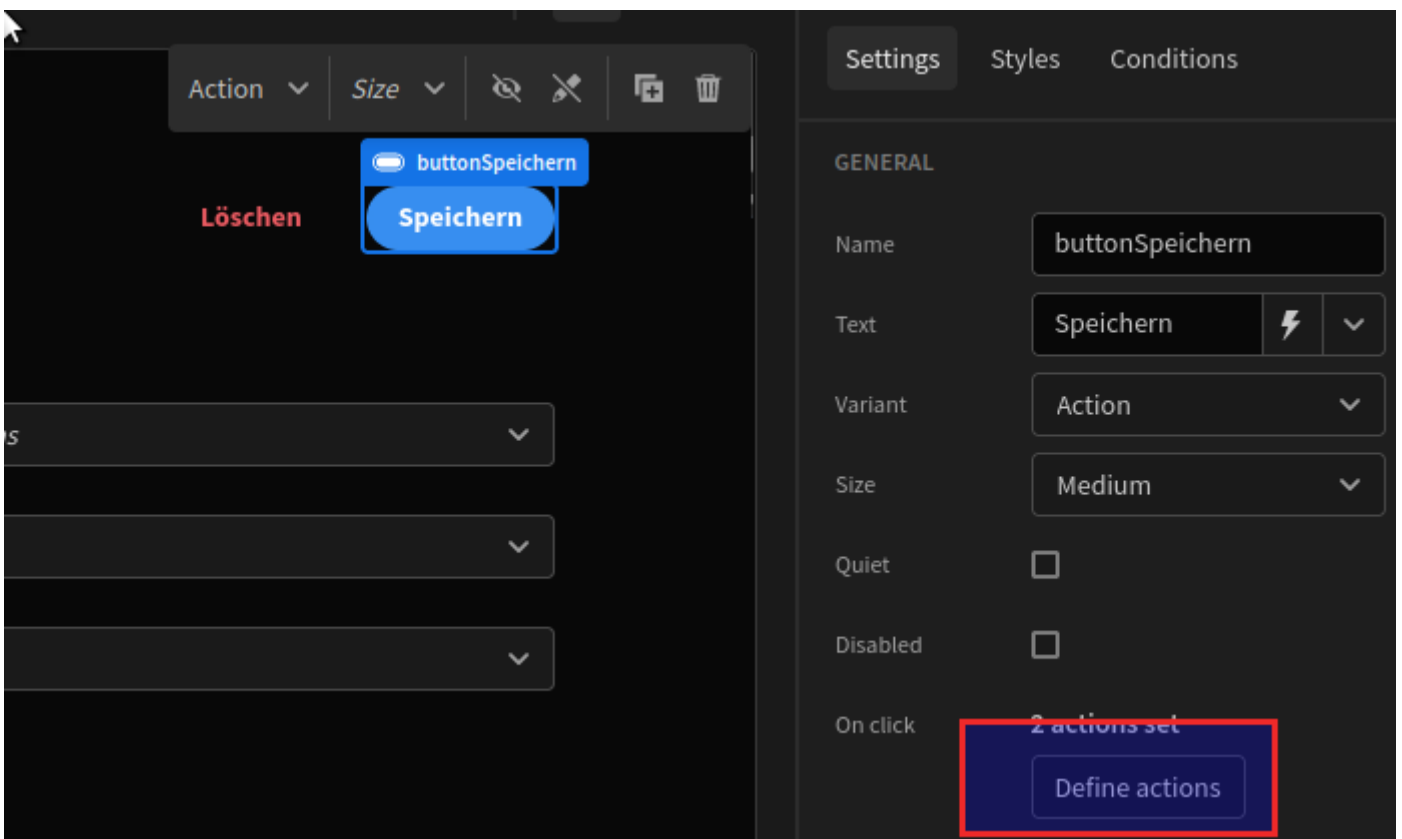


Nun sieht unsere Sidebar so aus.

Angezeigt wird sogar schon ein Wert, allerdings wenn wir diesen ändern würden und auf speichern klicken, passiert da noch nichts. Den default Wert laden, das geht natürlich schon, da wir das ja gerade definiert haben.

Also oben auf den Speichern button klicken.

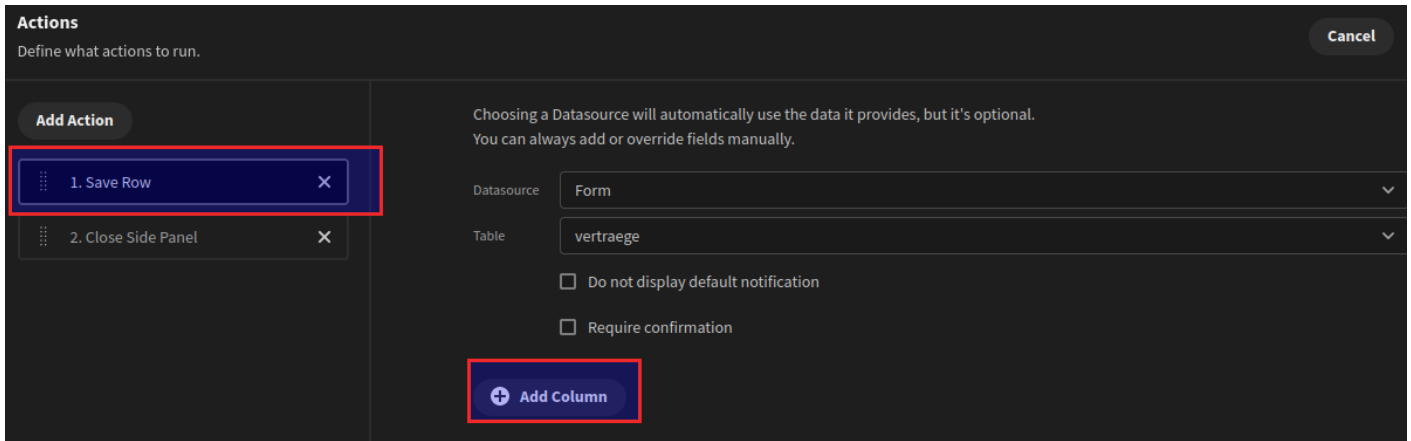
Dann auf define Actions klicken.



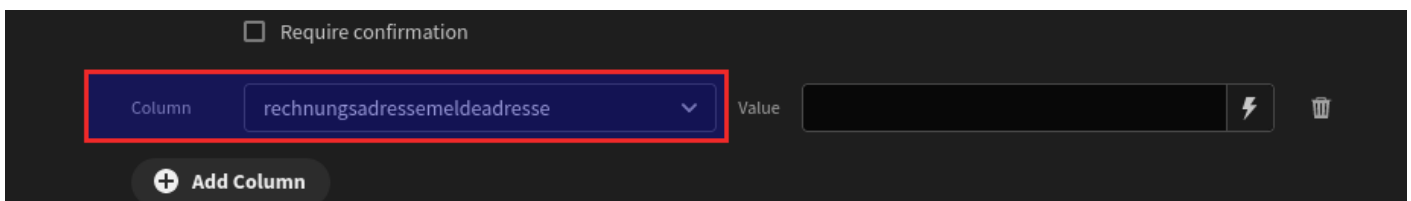
Nun auf Save row klicken.

Dann im Fenster auf Add column klicken.

Damit können wir eine spalte überschreiben mit einem anderen Wert.  
in diesem Fall aus die Rechnungsadresse aus unserem Picker.

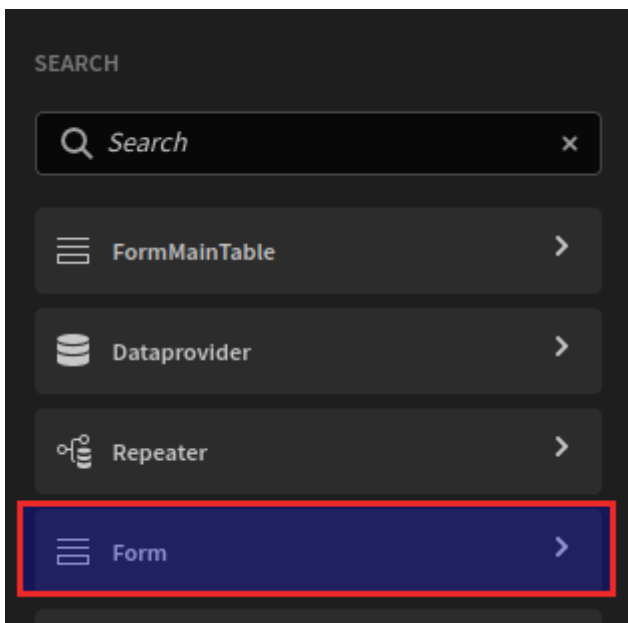


Nun die Rechnungsadresse auswählen, wichtig auch hier wieder die Link typ also die echte Relation nicht der selbsterstellte pointer. Entfällt natürlich bei einer MySQL wieder. Ist nur für interne DB. Sonst gäbe es ja nur den Link



Und bei Value auf den Blitz klicken.

Und bei auf das zweite Formular klicken, heißt bei mir Form. Dieses ist fürs Sidepanel. Denn FormMainTable ist für die Tabelle. Oder wie auch immer eure Formulare heißen



Und dort jetzt den picker auswählen

Netz	Options
Rechnungsadressemeldeadressepick	Options
Rufnummerzumtarif	Link
Tariflaufzeit	Options
Tarifname	String
Tariftyp	Link

Dann sieht das ganze so aus:  
Nun auf Save klicken.

Rechnungsadressemeldeadresse"

Cancel Save

Options Handlebars JavaScript

Options

Link

Options

String

Link

String

```

{{ Form.Fields.rechnungsadressemeldeadressepick }}

```

Nochmals auf Save:

Actions

Define what actions to run.

Cancel Save

Add Action

1. Save Row

2. Close Side Panel

Datasource Form

Table vertraege

Do not display default notification

Require confirmation

Column rechnungsadressemeldeadresse Value {{ Form.Fields.rechnungsadressemeldead ...

+ Add Column

## Ergebnis

Nun öffnen wir die App / aktualisieren die App mit F5 und gehen auf Kunden und klicken bei einem Kunden auf Verträge.

**Kunden**

Firmenname    Name    Vorname    Geburtsdatum    Strasse    **Kunden anlegen**

		FIRMENNAME	NAME	VORNAME	GEBURTSDATUM	DATENSCHUTZER
Öffnen	Adressen	Ausweise	Bankkarten	<b>Verträge</b>	löschen	Schulte Anna 16.05.2023
Öffnen	Adressen	Ausweise	Bankkarten	Verträge	löschen	example.com John Doe 03.06.1981

< Page 1 >

Nun öffnet sich die Seite mit den Verträgen und dort scrollen wir bis zu den Adressen:  
Wie zu sehen gehört der Vertrag Anna Schulte und die Adresse ist am Deich

KUNDE

Schulte Anna 16.05.2023

**Vertraege**

RVERTRAGSNUMMER	GUTSCHRIFT	VOLLMACHT	RECHNUNGSADRESSEMELDEADRESSE	LIEFERANSCHLUSSADRESSE	KUNDEN
			am Deich 16 26135 Oldenburg Deutschland   Liefer/Anschlussadresse		Schulte Anna 16.05.2023

Nun klicken wir auf den Vertrag und scrollen nach unten und sehen, am Deich wurde übernommen

gutschrift

vollmacht

Rechnungsadressemeldeadresse

Nun wählen wir eine andere Adresse und sehen es gibt nur 3 Adressen.  
Nämlich die , die Anna gehören. Gleichzeitig wählen wir jetzt zz zz zz aus

Search

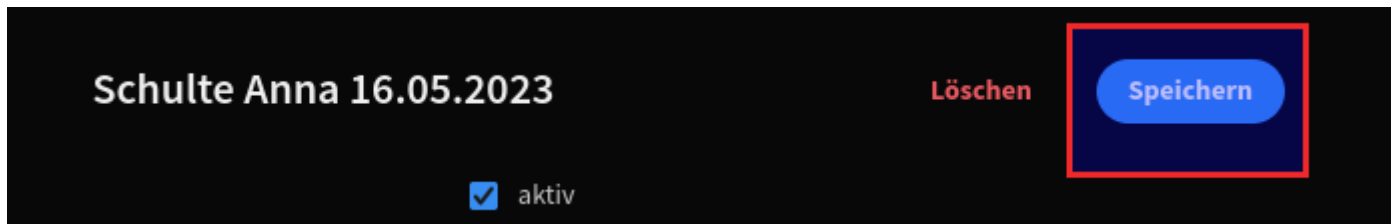
Choose an option

am Deich 16 26135 Oldenburg Deutschland | Liefer/Anschlussadresse ✓

sgdsg dsgsdg sgdsg | Liefer/Anschlussadresse

**zz zz zz | Rechnungsadresse/Meldeadresse**

Nun scrollen wir wieder nach oben und klicken auf speichern.



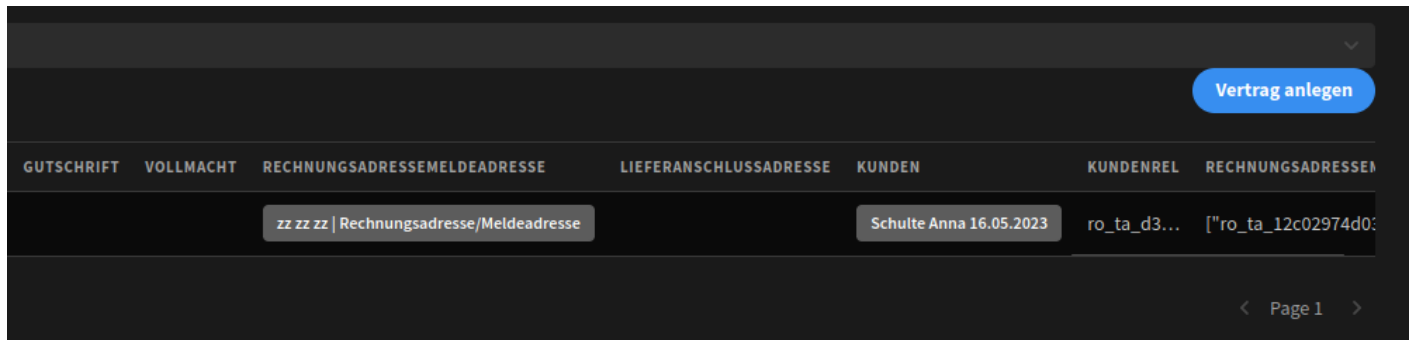
Schulte Anna 16.05.2023

Löschen

Speichern

aktiv

Nun wird automatisch der Vertrag aktualisiert mit der neuen Adresse.  
Mission erfolgreich.



Vertrag anlegen

GUTSCHRIFT	VOLLMACHT	RECHNUNGSADRESSEMELDEADRESSE	LIEFERANSCHLUSSADRESSE	KUNDEN	KUNDENREL	RECHNUNGSADRESSEN
		zz zz zz   Rechnungsadresse/Meldeadresse		Schulte Anna 16.05.2023	ro_ta_d3...	["ro_ta_12c02974d0:

< Page 1 >

# URL Variablen und öffnen im neuen Fenster

## Beschreibung:

Es gibt in Budibase zwei Möglichkeiten Daten in Variablen zu Speichern. einmal über States oder über URL Variablen.

In einer URL Variable sieht man den Inhalt in der URL. Also für sensible Daten die States benutzen.

Warum dann URL Variablen???

Weil States in einem neuen Fenster keine Gültigkeit mehr haben. States sind nur in einem Fenster gültig (Show Modal gilt auch noch) Da im Browserfenster geöffnet wird.

in der regel wird in einer URL Variable nur die ID und andere Parameter die dafür zuständig sind GUI elemente ein und auszublenden.

Sind es sensible GUI elemente die zum Beispiel nur bestimmte Benutzer sehen dürfen. Würde ich in der Datenbank ein extra Feld anlegen, was dann als condition vom user benutzt werden kann.

## Definition von URL Variablen

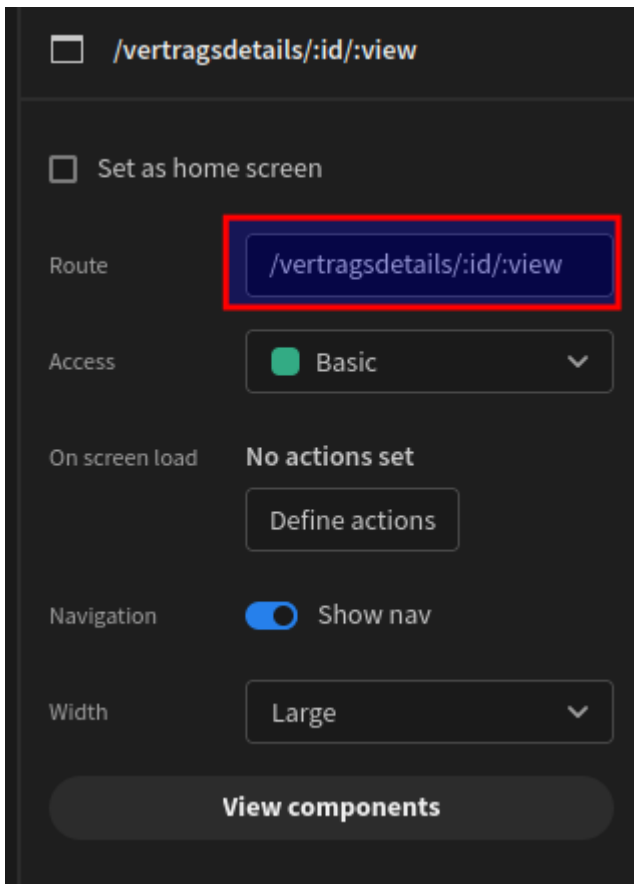
URL Variablen werden im Namen des Screens angehängen. eine Variable ist immer `/:varname`

```
/screenname/:id
```

Sollen mehrer Variablen definiert werden:

```
/screenname/:id/:meinwert
```

In meinem Beispielfeld die Variable id und die Variable view



## Auf Variablen zugreifen:

In dem Screen kann mit

```
{{ URL.id }} und {{ URL.view }}
```

drauf zugegriffen werden.

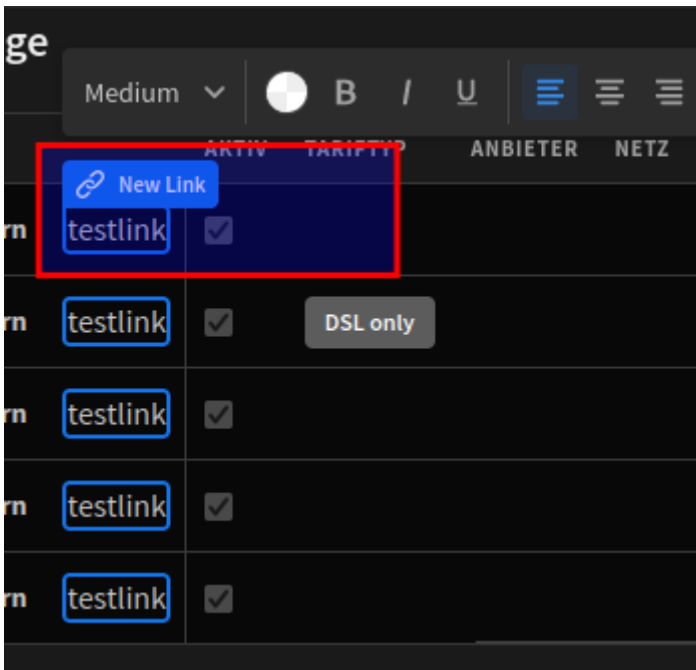
## Screen öffnen Variablen übergeben:

Übergeben werden die Variablen entweder manuell in einer Link Komponente oder über die Aktion Navigate to.

man kann sie natürlich auch manuell an die Adresszeile anfügen. z.b für Debug zwecke. Dann givibts eine andere Ansicht oder.

## Über Link Komponente:

Eine Linkkomponente erstellen in der Tabelle als Beispiel



Nun in den eigenschftaen des Links einen Namen vergeben und ein Ziel.  
Variablen werden einfach mit einem / angegeben

`/screenname/wert`

sollen mehre variablen übergeben werden einfach mit slash hintereinander.

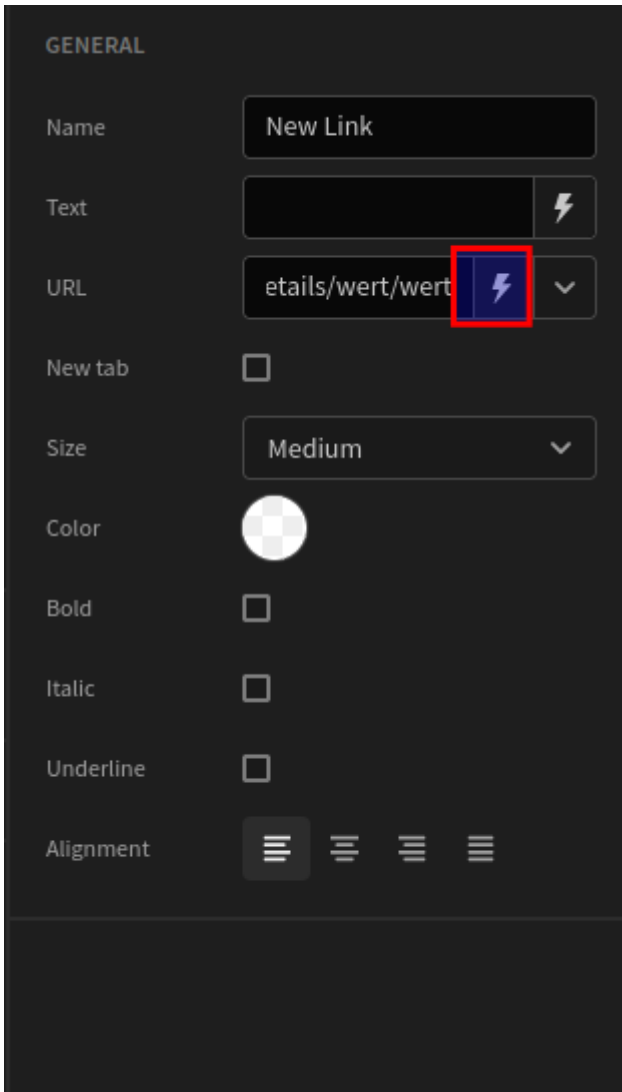
Auf die Reihenfolge achten, die Variablen werden in der Reihenfolge zugeordnet wie sie beim definieren angelegt wurden.

`/screenname/wert1/wert2`

In meinem Beispiel

`/vertragsdetails/{{ Table.vertraege._id }}/true`

Dazu auf den Blitz klciken

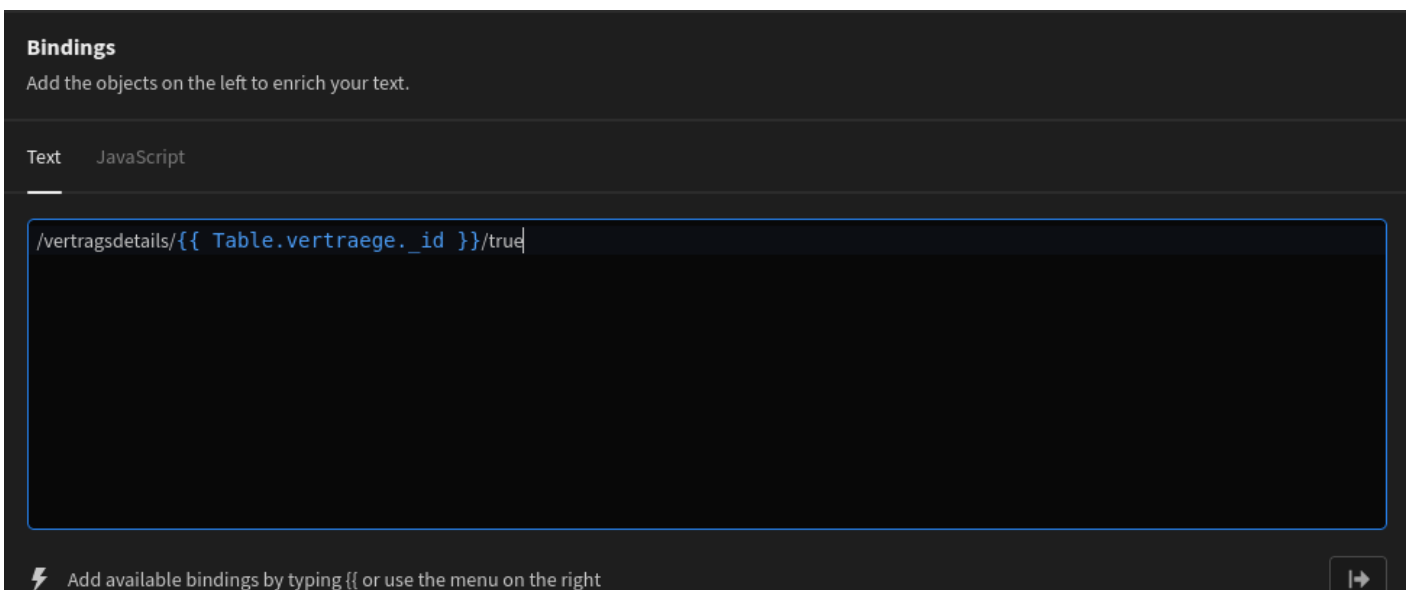


und dann dort eingeben.

Es können wieder Variablen übergeben werden wie auch, static text.

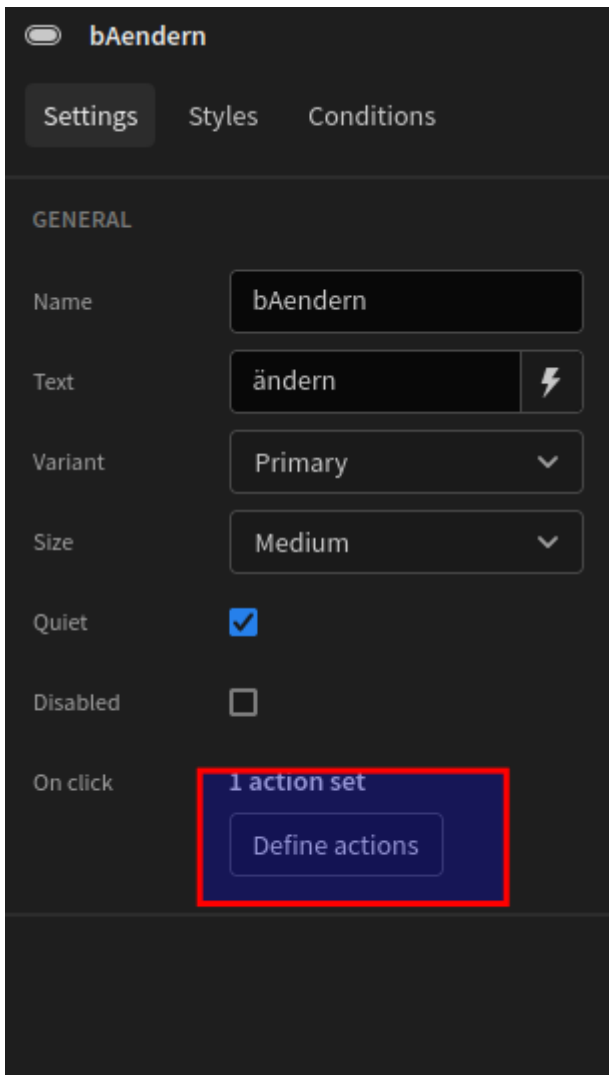
Der static text ist hier true.

Die Variable die ID von der row wo der Link drin ist



# Über einen Button mit Aktion Navigate To:

Im Button define Actions anklicken



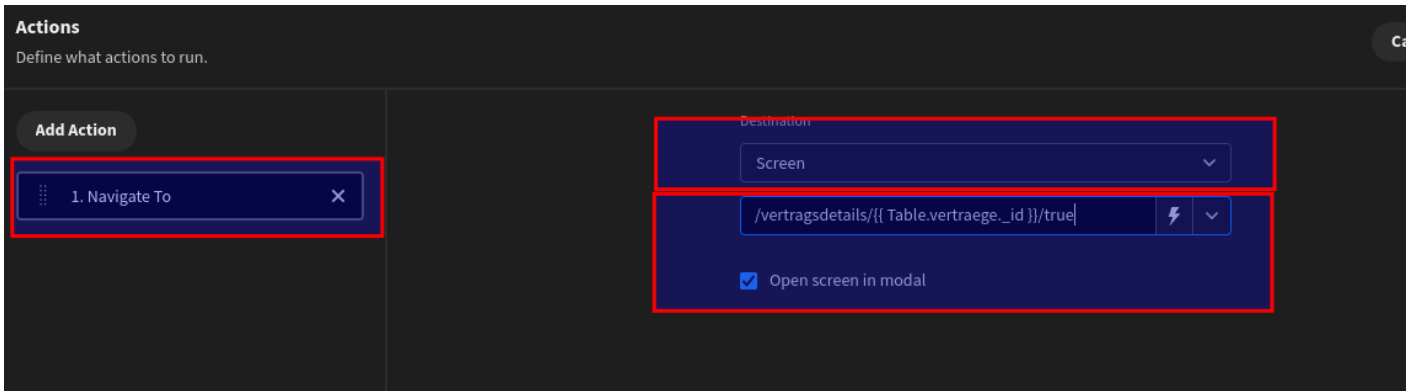
Aktion Naviagte to auswählen

Jetzt gibt es zwei Varianten Screen und URL

Screen:

- Dann Screen auswählen
- Screen mit Variablen eintragen, Ist genauso wie beim Link, zumindest wenn man Navigate to und dann Screen wählt.

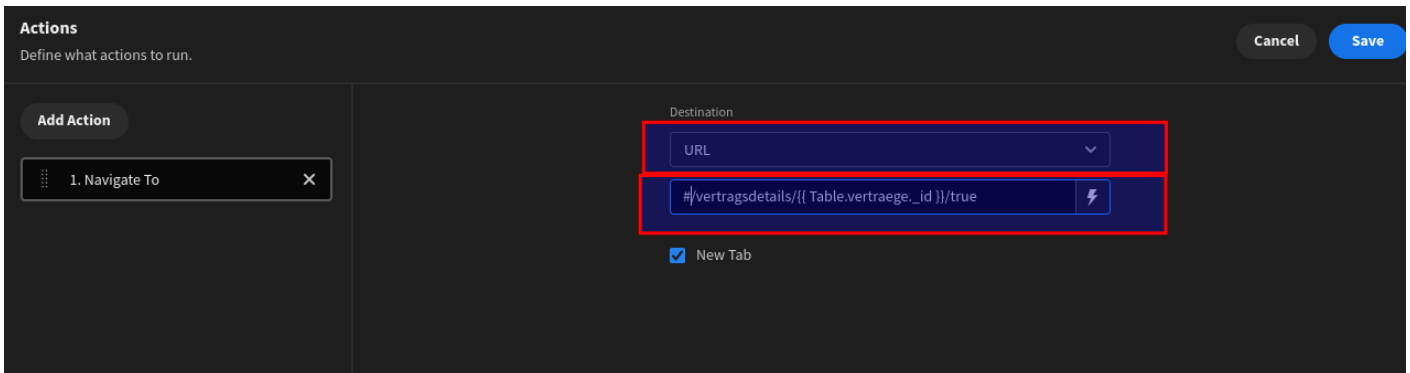
```
/vertragsdetails/{{ Table.vertraege._id }}/true
```



## URL:

Mit URL hab ich die Möglichkeit den Screen in einem neuen Fenster / Tab zu öffnen. Damit aber auch ein Screen geöffnet werden kann muss dem eine # vorangestellt werden. Sonst gibt es einen Zugriffsfehler.

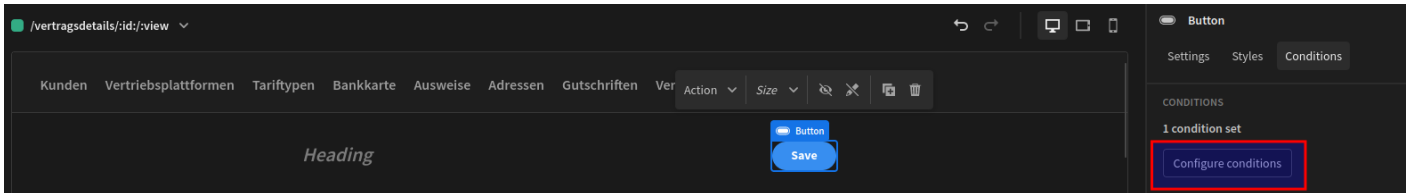
```
#/vertragsdetails/{{ Table.vertraege._id }}/true
```



Abrufen der URL Variablen im neuen Screen, als Beispiel mit einem Button der nicht dargestellt werden soll:

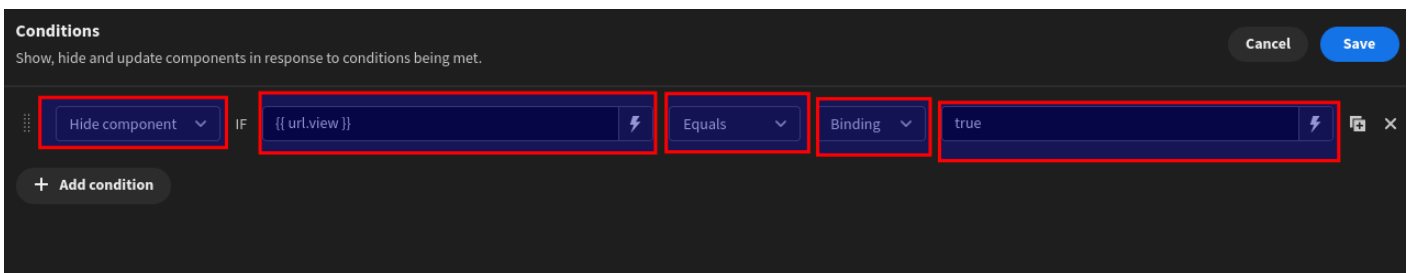
# Condition festlegen:

Da wir hier view auf true gesetzt haben wird im neuen Fenster der Button save nicht dargestellt. Da wir in dem Screen, den button unter condition folgenden filter hinzugefügt haben. Dazu in dem Screen auf den Button klicken der nicht dargestellt werden soll, wenn wie in unserem Beispiel view auf true gestellt ist. Dort dann auf configure conditions klicken



Und dann folgende condition hinzufügen.

- Hide component, wenn die condition zu trifft. Bei show, würde das element nur angezeigt werden, wenn die condition gültig wäre.
  - if {{ unsere URL.view }} Variable
  - =
  - true
- ist, wird die komponente versteckt.



# Ergebnis auf der Seite:

Wird true übergeben, kein button da

aktiv	<input checked="" type="checkbox"/>	
vertriebsplattform		vertriebsplattform
marktlokationsid		marktlokationsid
hsn		hsn
tsn		tsn
kfzkennzeichen		kfzkennzeichen
fahrzeug		fahrzeug

Wird false übergeben

**Actions**  
Define what actions to run.

Cancel Save

Add Action

1. Navigate To

Destination

URL

`#/vertragsdetails/{{ Table.vertraege_id }} /false`

New Tab

ist der Button da

The image shows a dark-themed user interface with a form. In the top right corner, a blue rounded button labeled "Save" is highlighted with a red rectangular border. The form contains the following fields:

aktiv	<input checked="" type="checkbox"/>
vertriebsplattform	<input type="text" value="vertriebsplattform"/>
marktlokationsid	<input type="text" value="marktlokationsid"/>
hsn	<input type="text" value="hsn"/>
tsn	<input type="text" value="tsn"/>
kfzkennzeichen	<input type="text" value="kfzkennzeichen"/>

Wozu das ganze?

So kann ich mit einem Screen, mehrer Sachen machen und muss nicht screens doppelt designen ;-)

# Einen copy to clipboard Button über eine embed Komponente

## Beschreibung:

Dropdown Menüs und Datumsfelder lassen sich mit dem Mauscursor nicht kopieren.  
Also zweckentfremden wir die embed Komponente mit einem Javascript zum kopieren.

**!!!! Wichtig diese JavaScript Funktion funktioniert nur auf Webseiten mit HTTPS, unter HTTP wird vom Browser diese Funktion deaktiviert, es gibt dann auch keine Fehlermeldung, es passiert nix!!!!!!**

## Durchführung:

Eine embed Komponente an die gewünschte Stelle hinzufügen.

Dann diesen HTML-Code einfügen

**!!!WICHTIG bei Text nicht bei Javascript is ja HTML-Code!!!**

Die Variable value mit der Variable ersetzen (also zum Beispiel textfeld aus der kooiert werden soll.)

```
<button onclick="( function()  
{ navigator.clipboard.writeText('{{ value }}').then(function (){alert (' {{ value }} Copied!')}} )  
)">Copy Link</button>
```

Hier ein Beispiel mit Deustchem Text und echten Variablen

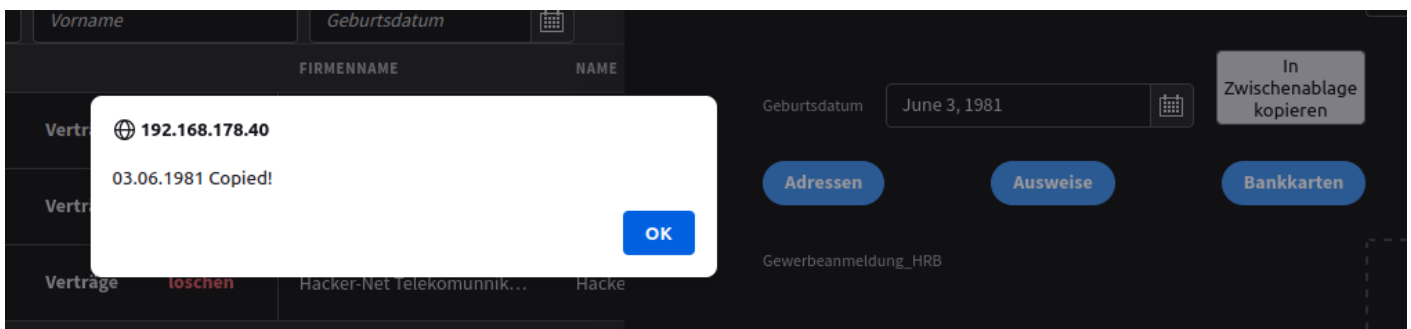
```
<button onclick="( function()  
{ navigator.clipboard.writeText('{{ Form.Fields.rechnungsadressemeldeadresse }}').then(function (){alert ('  
{{ Form.Fields.rechnungsadressemeldeadresse }} in die Zwischenablage kopiert!')}} )  
)">Copy</button>
```

# Hier ein Beispiel wenn ein Datum umformatiert werden soll.

Denn in Budibase wird ein Datumsfeld in diesem Format zurückgegeben, wenn es durch keine Komponente wie eine Datumskomponente umformatiert wird : 1981-06-03T00:00:00.000

```
<button onclick="( function()  
{  
  var value = '{{ value }}'; // Das Originaldatum  
  var date = new Date(value); // Erstellen eines neuen Date-Objekts mit dem Wert  
  var day = String(date.getDate()).padStart(2, '0'); // Extrahieren und Formatieren des Tages  
  var month = String(date.getMonth() + 1).padStart(2, '0'); // Extrahieren und Formatieren des Monats  
  var year = date.getFullYear(); // Extrahieren des Jahres  
  
  // Zusammenfügen der Teile in das gewünschte Format  
  var formattedDate = day + '.' + month + '.' + year;  
  
  navigator.clipboard.writeText(formattedDate).then(function (){  
    alert (formattedDate+' Copied!')  
  })  
})()">Copy Link</button>
```

Nun sieht die Rückgabe so aus:



Beispiel Option Picker, da Option Picker nur den Wert also in der Regel die ID zurückgeben und nicht das Label.

Hier ein Workaround.

hier wird der Button komplett mit einer JavaScript Funktion gebaut.

Also diesmal nicht auf Text sondern auf JavaScript gehen

```
Text JavaScript
1 const rows = $("dataRechnungsadressemeldeadresse.Rows");
2 let street = "strasse";
3 let ort = "ort";
4 let plz = "plz";
5 for (let i in rows) {
6   if ($("Form.Fields.rechnungsadressemeldeadressepicker"
7     ... {
```

Dort dann dieses Script einfügen

```
const rows = $("dataRechnungsadressemeldeadresse.Rows"); //holt alle rows vom Data Provider
//unsere einzelnen Felder als Variablen deklarieren
let street = "strasse";
let ort = "ort";
let plz = "plz";

//durch alle rows gehen und eine laufvariable i erstellen
for (let i in rows) {
  //wenn die row id von den rows mit dem picker übereinstimmen, denn der picker enthält ja nur die id dann ist
  das unser Datensatz
  if ($("Form.Fields.rechnungsadressemeldeadressepicker") == rows[i]._id)
  {
    //nun speichern wir die einzelnen Felder in den Variablen um sie dann weiter zu verwenden
    street = rows[i].strasse;
    ort = rows[i].ort;
    plz = rows[i].plz;
  }
}

//Hier setzten wir unsere Variablen zusammen
let text = street + " " + plz + " " + " " + ort;

//Hier erstellen wir den Button mit dem Clipboard und setzten dort unsere Text Varibalen ein
let button = `
```

Und hier der Komplette Code ohne Erläuterung

```
const rows = $("dataRechnungsadressemeldeadresse.Rows");
let street = "strasse";
let ort = "ort";
let plz = "plz";
for (let i in rows) {
  if ($("#Form.Fields.rechnungsadressemeldeadressepicker") == rows[i]._id)
  {
    street = rows[i].strasse;
    ort = rows[i].ort;
    plz = rows[i].plz;
  }
}
let text = street + " " + plz + " " + " " + ort;
let button = `
```

## Extrahieren der Straße Beispiel

```
//die funktion die die straße rausfiltert
function extractStreet(address) {
  let streetMatch = address.match(/^[^\d]*/);
  if (streetMatch === null) {
    return "Fehler: Keine Straße gefunden.";
  }
  return streetMatch[0].trim();
}

const rows = $("dataRechnungsadressemeldeadresse.Rows"); //holt alle rows vom Data Provider
//unsere einzelnen Felder als Variablen deklarieren
let street = "strasse";
let ort = "ort";
let plz = "plz";

//durch alle rows gehen und eine laufvariable i erstellen
for (let i in rows) {
```

```
//wenn die row id von den rows mit dem picker übereinstimmen, denn der picker enthält ja nur die id //dann ist das unser Datensatz
```

```
if ($("#Form.Fields.rechnungsadressemeldeadressepicker") == rows[i]._id)
```

```
{
```

```
    //nun speichern wir die einzelnen Felder in den Variablen um sie dann weiter zu verwenden
```

```
    street = rows[i].strasse;
```

```
    ort = rows[i].ort;
```

```
    plz = rows[i].plz;
```

```
}
```

```
}
```

```
//Hier setzen wir unsere Variablen zusammen
```

```
let text = extractStreet(street); //hier haben wir ein funktion oben die mit einem regulären ausdruck die Straße rausfiltert
```

```
//Hier erstellen wir den Button mit dem Clipboard und setzen dort unsere Text Variablen ein
```

```
let button = `<button name="rechnungsadressemeldeadresse" onclick="( function()
```

```
{
```

```
    navigator.clipboard.writeText('${text}').then(function (){alert ('${text} Copied!')} ) }
```

```
)">Copy Strasse</button>`;
```

```
//Die JavaScript funktionen geben in Budibase immer das was angezeigt werden soll zurück, in unserem
```

```
//Beispiel der Button.
```

```
return button;
```

## Extrahieren der Hausnummer Beispiel:

```
function extractHouseNumber(address) {
```

```
    let numberMatch = address.match(/d+[a-z]?/i);
```

```
    if (numberMatch === null) {
```

```
        return "Fehler: Keine Hausnummer gefunden.";
```

```
    }
```

```
    return numberMatch[0];
```

```
}
```

```
const rows = $("#dataRechnungsadressemeldeadresse.Rows"); //holt alle rows vom Data Provider
```

```
//unsere einzelnen Felder als Variablen deklarieren
```

```
let street = "strasse";
```

```
let ort = "ort";
```

```
let plz = "plz";
```

```
let text = "text"
```

```
//durch alle rows gehen und eine laufvariable i erstellen
for (let i in rows) {
    //wenn die row id von den rows mit dem picker übereinstimmen, denn der picker enthält ja nur die id //dann ist
das unser Datensatz
    if ($("#Form.Fields.rechnungsadressemeldeadressepicker") == rows[i]._id)
    {
        //nun speichern wir die einzelnen Felder in den Variablen um sie dann weiter zu verwenden
        street = rows[i].strasse;
        ort = rows[i].ort;
        plz = rows[i].plz;
    }
}
//Hier setzten wir unsere Variablen zusammen
text = extractHouseNumber(street);
//Hier erstellen wir den Button mit dem Clipboard und setzten dort unsere Text Varibalen ein
let button = `
```

# Mit einer Embed Komponente den Webseiten Titel ändern.

## Beschreibung:

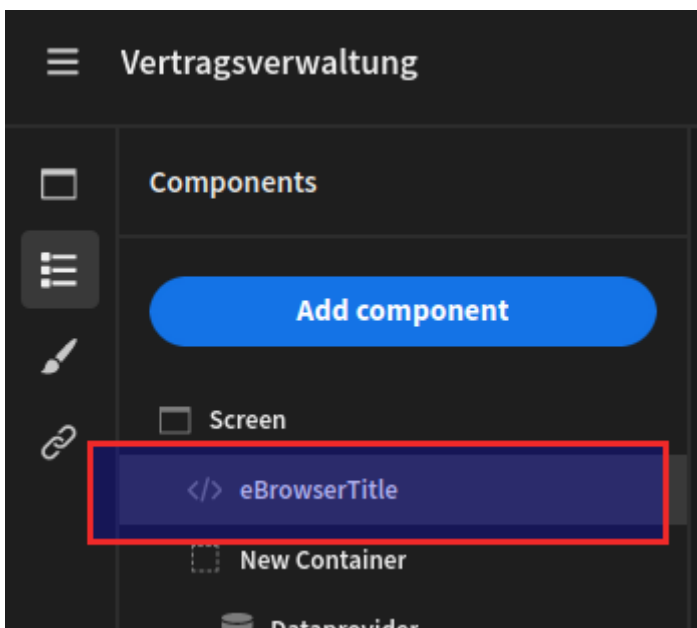
Es gibt Situationen, da möchte man den Webseitentitel gerne in Kundenname oder Vertragsnamen Nr etc. drin haben.

Da der Titel ja von Budibase vorgegeben wird, behelfen wir uns einer embed Komponente um den Titel zu setzen.

## Durchführung:

Eine embed Komponente erstellen und im Textteil folgendes einfügen.

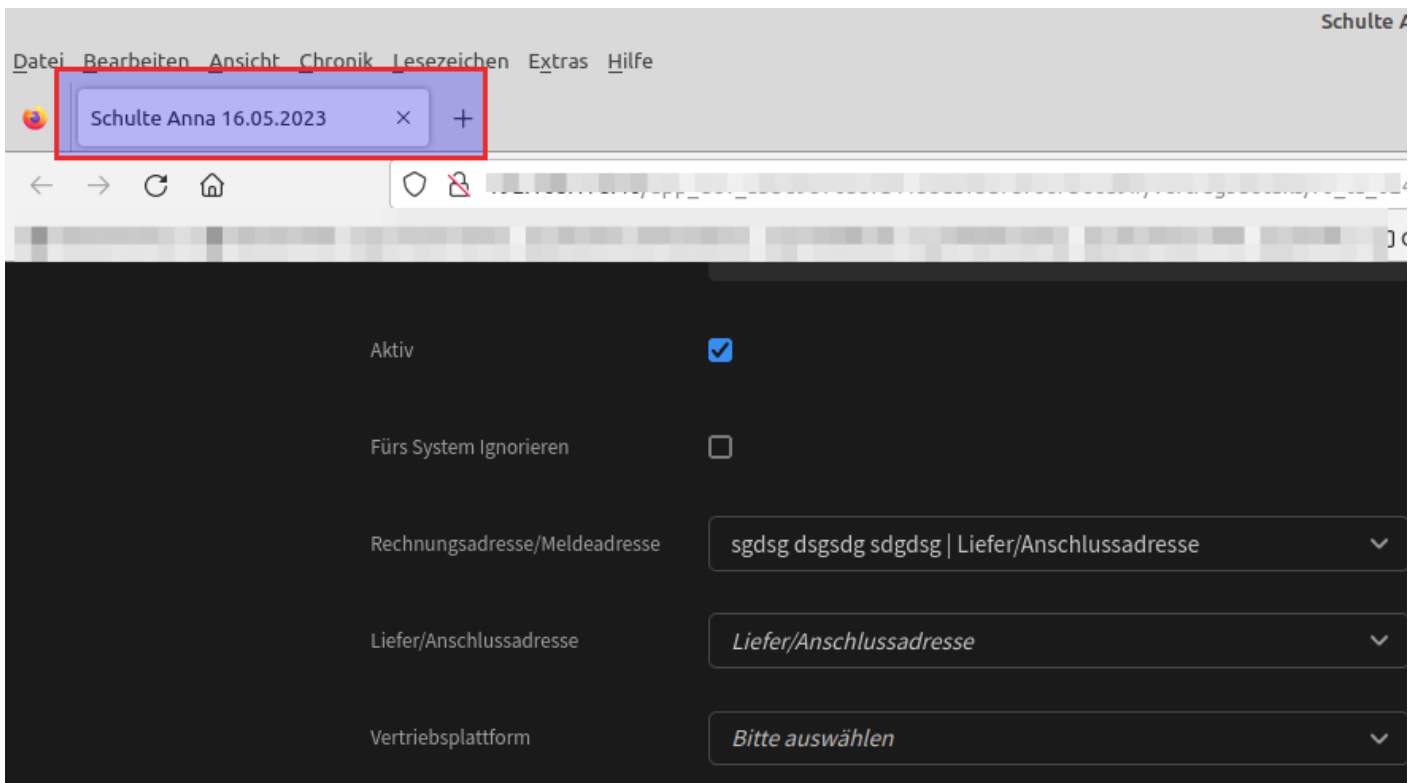
Die Embed Komponente muss die erste Komponente in der Reihenfolge sein



Die {{ variabel mit dem Text ersetzen wenn eine Variable gewünscht ist.

text natürlich den eigenen Bedürfnissen anpassen.

```
<img style="display: none;" src onerror="document.title = '{{ Variabel }}'" >
```



# Mit einer Embed Komponente alle Attachments in einem Attachmentfield zum Download bereitstellen

## Beschreibung:

Es gibt momente, da möchte man einfach auf dem Formular alle Attachments in einem Attachment Field als Download link im Formular haben.

Mit einem Repeater Block und darin enthaltenden Embed-Komponente, könnte man das so realisieren:

## Quelltext Embed Komponente:

Die variable Attachments enthält das AttachmentField aus dem Repeater.  
sind keine Attachments drin. Wird einfach der text ausgegeben, No Attachments here.  
Im Javascript Teil der Embed-Komponente

```
const attachments = $("New Repeater block.test.attachments");

if(attachments) {
  let html = "";
  for (const attachment of attachments) {
    html += `<a href="${attachment.url}" download>${attachment.name}</a><br>`;
  }
}
```

```
return html;  
}  
return 'No attachments here';
```