

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 koiert 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

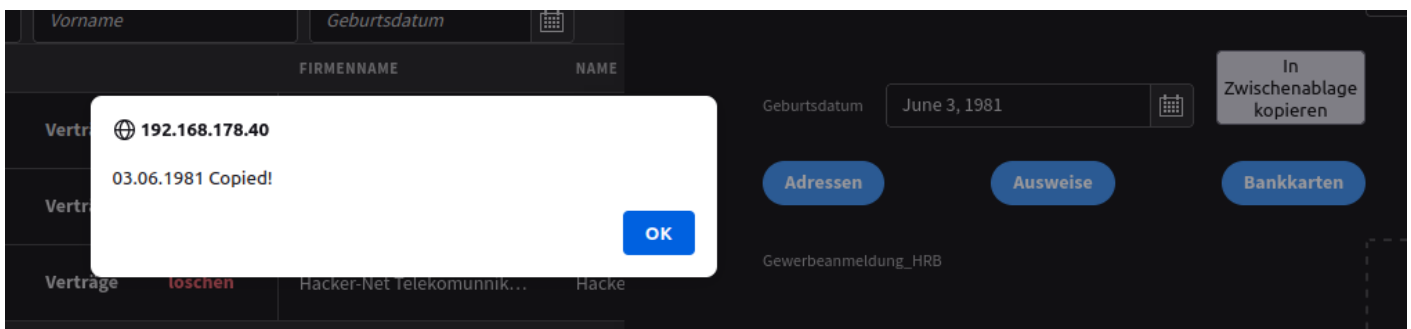
```
<button onclick="( function()  
{ navigator.clipboard.writeText('{{ Form.Fields.rechnungsadressemeldeadresse }}').then(function (){alert ('  
{{ Form.Fields.rechnungsadressemeldeadresse }} in die Zwischenablage kopiert!')}} )  
)">Copy</button>
```

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 ( $("Form.Fields.rechnungsadressemeldeadressepicker") == rows[i]._id
    {
      //nun speichern wir die einzelnen Felder in den Variablen um sie dann weiter zu verwenden
      street = rows[i].strasse;
      ort = rows[i].ort;
      plz = rows[i].plz;
    }
  }
//Hier setzten wir unsere Variablen zusammen
let text = street + " " + plz + " " + " " + ort;
//Hier erstellen wir den Button mit dem Clipboard und setzten dort unsere Text Varibalen ein
let button = `<button name="rechnungsadressemeldeadresse" onclick="( function()
{
  navigator.clipboard.writeText('${text}').then(function (){alert ('${text} Copied!')}) }
)()">Copy Link</button>`;
//Die JavaScript funktionen gegben in Budibase immer das was angezeigt werden soll zurück, in unserem
Beispiel der Button.
return 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 = `
```

Version #10

Erstellt: 17 Juni 2023 10:37:56 von Admin

Zuletzt aktualisiert: 30 Juni 2023 04:11:17 von Admin