

# Ereignisse

## Inhaltsverzeichnis

- [1 SQL](#)
- [2 Sage API-Aufruf](#)
- [3 JavaScript](#)
  - [3.1 Besonderheiten JavaScript](#)

Bei der Neuanlage, dem Bearbeiten oder dem Löschen bietet die Ereignisse die Möglichkeit zusätzliche Logik zu unterschiedlichen Zeitpunkten auszuführen.

**Ereignisse** (Symbol: Blitz): Zu bestimmten Ereignissen können SQL-Anweisungen, JavaScript-Anweisungen und Sage-API-Aufrufe hinterlegt werden.

## 1 SQL

Folgende Eingabefelder sind für ein Ereignis vom Typ SQL verfügbar:

Eingabefeld	Bedeutung
Bezeichnung	Zur internen Zuordnung/Verwaltung von Ereignissen Auslöser für das Ereignis, folgende Auswahl ist vorhanden:
Ereignis	<ul style="list-style-type: none"><li>- Neuanlage eines Datensatzes</li><li>- Vor Einfügen eines neuen Datensatzes</li><li>- Nach Einfügen eines neuen Datensatzes</li><li>- Vor Aktualisierung eines vorhandenen Datensatzes</li><li>- Nach Aktualisierung eines vorhandenen Datensatzes</li><li>- Vor Löschen eines Datensatzes</li><li>- Nach Löschen eines Datensatzes</li><li>- Beim Ändern eines Feldes</li><li>- Beim Öffnen eines Formulars mit Daten</li></ul>
Feld	Hinweis: Siehe Ausführungsreihenfolge in Abbildung 1. Nur bei dem Ereignis "Beim Ändern eines Feldes" vorhanden. Auswahl des Listenfeldes welches bei einer Änderung das Ereignis auslöst.
Typ	SQL (Sage API-Aufruf und JavaScript)
SQL-Skript	Das auszuführende Skript. Für Hilfe zu den Platzhaltern kann der Editor über den Button "SQL Skript" in der Aktionsleiste aufgerufen werden.
SQL-Bedingung	Die optionale Bedingung, welche zutreffen muss, damit das Ereignis ausgeführt wird. Für Hilfe zu den Platzhaltern kann der Editor über den Button "SQL Bedingung" in der Aktionsleiste aufgerufen werden.
Aktiv	Nur wenn die Checkbox aktiviert ist, wird das Ereignis ausgeführt.

Tabelle 1: Eingabefelder Ereignis (Liste)

Ansicht	Ereignisse und deren Besonderheiten
Aktivierte Mehrfachauswahl	Es werden nur die Ereignisse "Vor Aktualisieren eines vorhandenen Datensatzes" und "Nach aktualisieren eines vorhandenen Datensatzes" ausgelöst.
Systemparameter Mandantennummer im AppBuilder (nicht für veröffentlichte Apps)	Der Parameter [\$System:Mandantennummer] ist im Kontext des AppBuilders immer "123". Unabhängig von dem für die App im Gateway hinterlegten Wert.

Tabelle 2: Besonderheiten in Verbindung mit Ereignissen

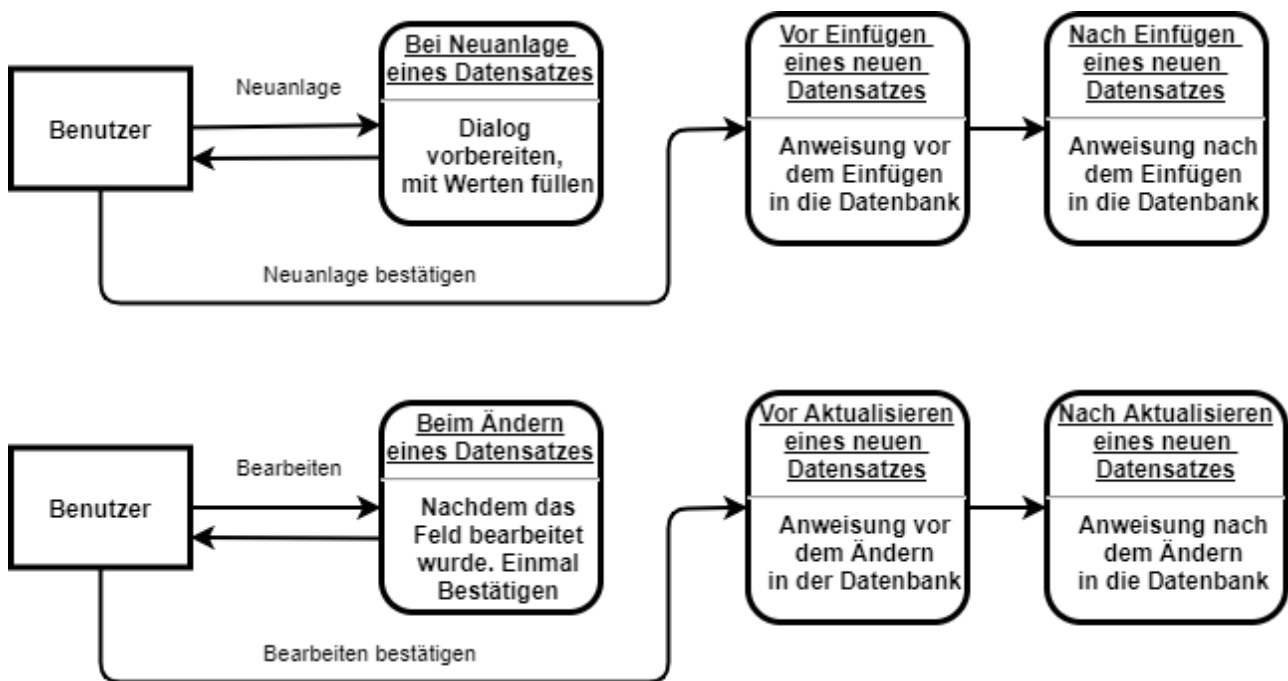


Abbildung 1: Ausführungsreihenfolge der Ereignisse für "Neuanlage" und "Aktualisieren"

## 2 Sage API-Aufruf

Dokumentation folgt.

## 3 JavaScript

Die gleichen Ereignisse wie für den Typ SQL.

Mit JavaScript werden die Ereignisse direkt im Browser des Benutzers ausgeführt. Das Skript muss in JavaScript Syntax hinterlegt sein.

Für JavaScript sind folgende Funktionen hilfreich, welche in dem Skript aufgerufen werden können.

Name	Funktion
<code>this.preventDefault();</code>	Bricht die Ausführung ab (z.B. das Speichern in einem Dialog).
<code>\$app.alert('Nachricht');</code>	Öffnet eine Nachricht in einem Dialog
<code>\$app.touch.notify('Nachricht');</code>	Zeigt eine Nachricht am unteren Rand an.
<code>\$app.input.focus({ fieldName: 'NameDesKategoriefelds' });</code>	Fokussiert ein Kategoriefeld welches über den Namen des Feldes angesprochen wird.

Beispiel:

Code

```
if ([$Field:FeldFuerBearbeiten] == '1') {  
    this.preventDefault();  
  
    $app.alert('DieEingabevonlistnichtbeiderNeuanlageerlaubt.').done(function(){  
        $app.input.focus({ fieldName:  
    });  
}
```

### 3.1 Besonderheiten JavaScript

Alle Anweisungen müssen mit einem Semikolon (;) abgeschlossen werden, sonst kann die JavaScript Anweisung in mydatastream nicht korrekt ausgeführt werden.

Bei Neuanlagen sind Felder ohne Wert "undefined". Es muss z.B. auf

Code

```
if ([$Field:FeldFuerBearbeiten] != undefined [...]
```

geprüft werden.