

# Event - Löschen (im Anschluss)

 Sprache: **Deutsch** · [English](#)

## Event - Löschen (im Anschluss)

Ereignis-Regel „Löschen (im Anschluss)“: Interface, Kontext und ein Beispiel – so reagierst du auf dieses Nuclos-Ereignis.

REFERENZ

ENTWICKLER

STAND: JUL 2026

GILT FUER NUCLOS 4.2026.X

### Auf dieser Seite

- [Wann greift die Regel?](#)
- [Struktur](#)
- [Zuweisung](#)
- [Beispiel](#)
- [Verwandte Seiten](#)

## Wann greift die Regel?

Diese [serverseitige Regel](#) wird ausgeführt **nach** dem erfolgreichen Löschen eines Datensatzes. Sie kann einem **Businessobjekt** zugewiesen werden.

## Struktur

Grundgerüst der Regelklasse (der Regeleditor legt es automatisch an):

```
package org.nuclet.lager;

import org.nuclos.api.rule.DeleteFinalRule;
import org.nuclos.api.context.DeleteContext;
import org.nuclos.api.annotation.Rule;
import org.nuclos.api.exception.BusinessException;

/** @name
 * @description
 * @usage
 * @change
 */
@Rule(name="Loeschen Lagerposition im Anschluss", description="Loeschen Lagerposition im Anschluss")
public class LoeschenLagerpositionImAnschluss implements DeleteFinalRule {

    public void deleteFinal(DeleteContext context) throws BusinessException {
    }
}
```



### Kontext: DeleteContext

Der Kontext DeleteContext liefert das betroffene BusinessObject über context.getBusinessObject(...); zusätzlich stehen die Funktionen des RuleContext (Provider) bereit. Eine BusinessException bricht den Vorgang ab.

## Zuweisung

Im [Server-Regelmanager](#) per Drag-and-Drop dem Ziel zuweisen. Optional lässt sich die Ausführung auf einen **Status** oder eine **Aktion** einschränken; die Reihenfolge steuerst du über die Pfeil-Buttons.

| Regel                            | Status | Aktion |
|----------------------------------|--------|--------|
| LoeschenLagerpositionImAnschl... |        |        |




*Zuweisung „Löschen (im Anschluss)“.*

## Beispiel

Das folgende Beispiel soll den Aufbau einer Regel vom Typ „Löschen im Anschluss“, und die Weiterverarbeitung der übergebenen Inhalte veranschaulichen.

```
package org.nuclet.company;

import java.util.Date;
import java.util.List;
import org.nuclos.api.annotation.Rule;
import org.nuclos.api.businessobject.BusinessObject;
import org.nuclos.api.businessobject.Query;
import org.nuclos.api.context.DeleteContext;
import org.nuclos.api.exception.BusinessException;
import org.nuclos.api.provider.BusinessObjectProvider;
import org.nuclos.api.provider.QueryProvider;
import org.nuclos.api.rule.DeleteFinalRule;

/** @name
 *  @description
 *  @usage
 *  @change
 */
@Rule(name="BestellungFinalDelete", description="BestellungFinalDelete")
public class BestellungFinalDelete implements DeleteFinalRule {

    public void deleteFinal(DeleteContext context) throws BusinessException {

        Bestellung businessObject = context.getBusinessObject(Bestellung.class);

        Query<History> allHistoryEntries = QueryProvider.create(History.class);
        allHistoryEntries.where(History.EntityId.equals(businessObject.getId()));

        List<History> results = QueryProvider.execute(allHistoryEntries);

        for (History b : results) {
            if (context.isLogical()) {
                b.setBemerkung("logical delete");
            } else {
                b.setBemerkung("physical delete");
            }
            BusinessObjectProvider.update(b);
        }
    }
}
```

## Verwandte Seiten

 **Regeln (serverseitig)**

Grundlagen.

 **Server Regelmanager**

Zuweisen.

Öffnen

Öffnen



## Ausführungsreihenfolge der Regeln

Reihenfolge.

Öffnen