Modul-153

Dokumentation Projekt «zoo»

Bachmann Erin & Wicki MOritz

2022

Inhaltsverzeichnis

[Projekt Beschreibung 2](#_Toc105532847)

[Thema Beschreibung 2](#_Toc105532848)

[ER-Diagramm 3](#_Toc105532849)

[Relationales Modell 3](#_Toc105532850)

[Triggers 3](#_Toc105532851)

[Trigger 1 3](#_Toc105532852)

[Trigger 2 4](#_Toc105532853)

[Trigger 3 4](#_Toc105532854)

[Arbeitsrapport 5](#_Toc105532855)

# Projekt Beschreibung

In dieser Gruppenarbeit entwerfen und erstellen wir zu einem frei gewählten Thema eine Datenbank. In unserem Beispiel ist dies eine Zoo-Datenbank.

Die Datenbank wird unter T-SQL mit dem Tool «SQL-Server Management Studio» erstellt.

Das Projekt umfasst dabei folgende Punkte:

* Eine Dokumentation mit ER-Diagramm, Datenbankbeschreibung
* SQL-Code mit welchem die Datenbank inkl. Trigger und Stored Procedures/Functions erstellt werden kann
* SQL-Code mit dem die Daten in die Datenbank eingetragen werden
* SQL-Code für Abfragen und Tests inkl. einer kurzen Testbeschreibung.
* einen einfachen Arbeitsrapport (mit Datum, Dauer, Beschreibung, Person pro ausgeführte Arbeit).

# Thema Beschreibung

Als Thema für das Projekt haben wir uns für eine Zoo-Datenbank entschieden. Die Datenbank wird aus 5 Tabellen bestehen. Zu diesen gehört:

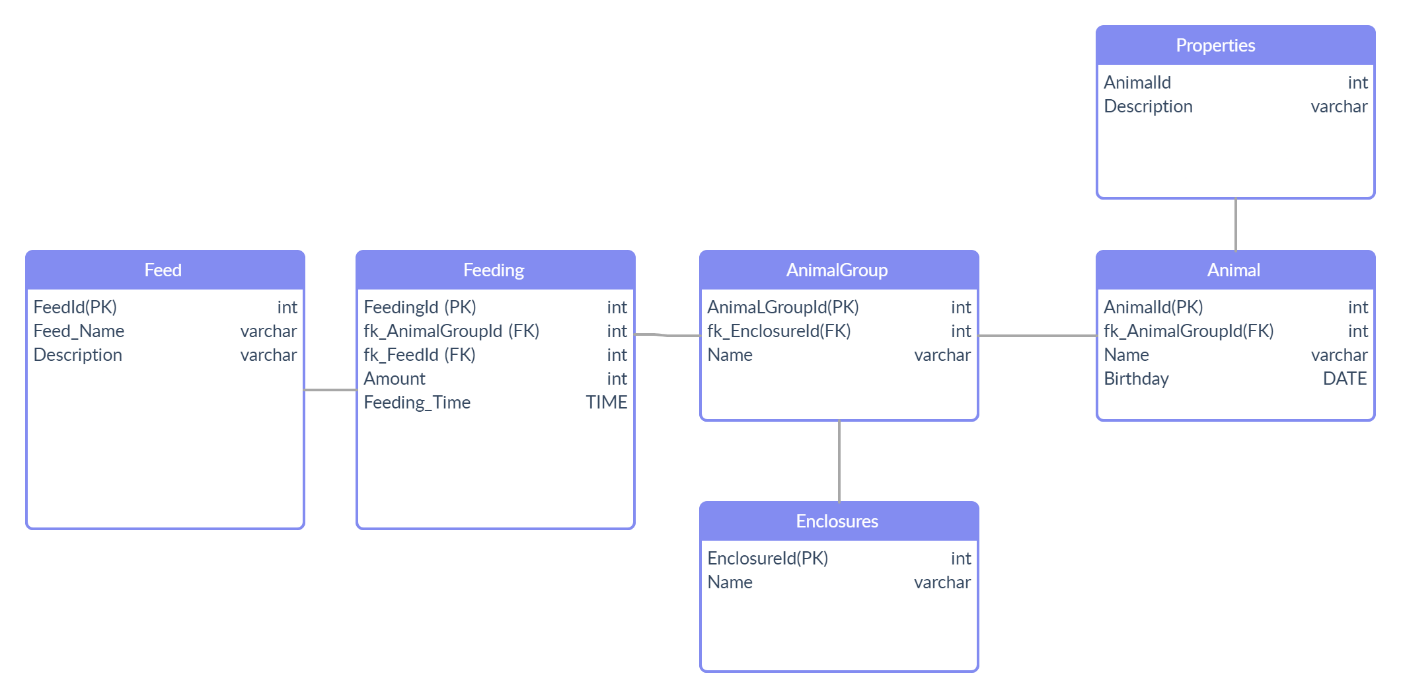
* Tier Tabelle
* Tier-Gruppe Tabelle
* Eigenschaften Tabelle
* Gehege Tabelle
* Fütterung Tabelle
* Futter Tabelle

Das Ganze ist folgendermassen aufgebaut:

* Ein Tier kann mehrere Eigenschaften haben.
* Ein Tier gehört zu einer Tiergruppe.
* Eine Tiergruppe gehört in 1 oder mehrere Gehänge.
* Eine Tiergruppe kann mehrere Fütterungen haben.
* Eine Fütterung besteht aus einem Futter.

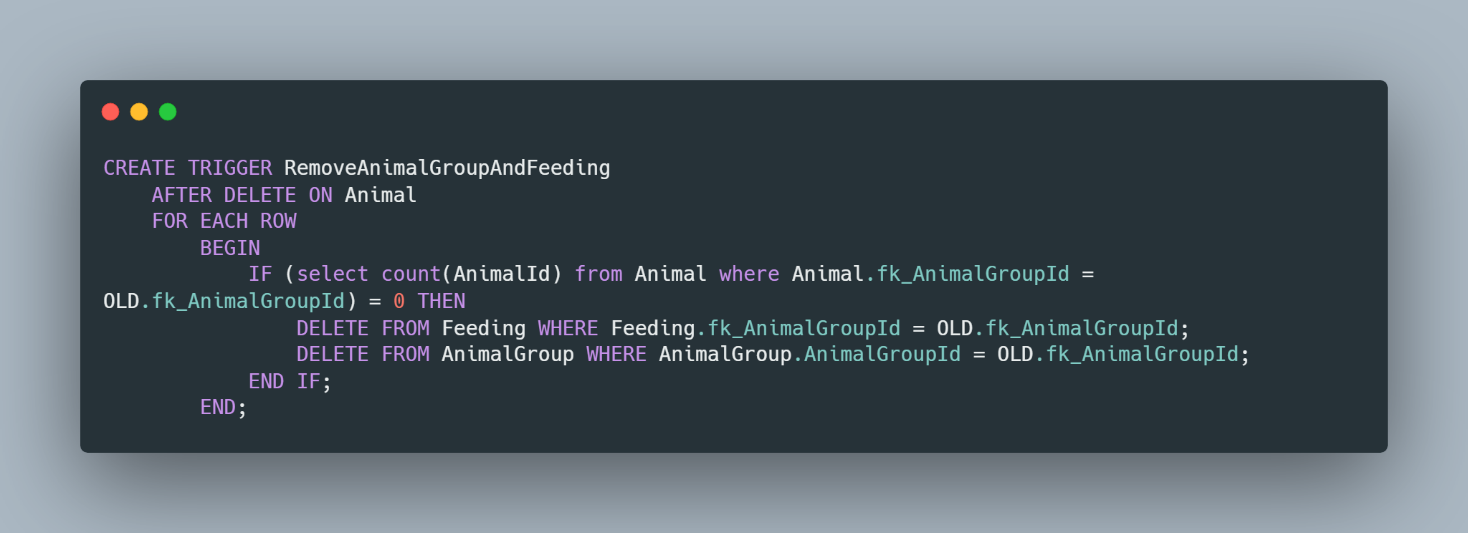
# ER-Diagramm

# Relationales Modell



# Triggers

## Trigger 1

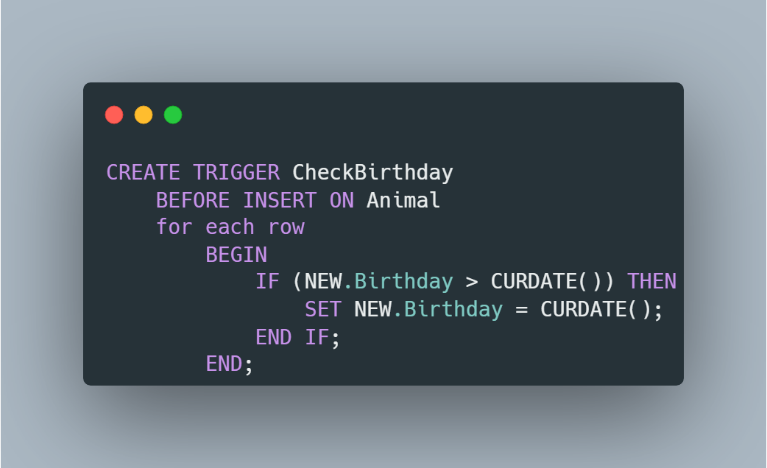


Der Trigger RemoveAnimalGroupAndFeeding wird immer nach dem Löschen von einem Tier ausgeführt. Dieser Trigger stellt sicher, dass eine Tiergruppe, welche gar keine Tiere mehr enthält, gelöscht und nicht mehr gefüttert wird.

Dies wird gemacht, indem geschaut wird, ob die Anzahl der Tiere in der Tiergruppe des gelöschten Tieres gleich 0 ist.

Ist dies der Fall, enthält die Tiergruppe keine Tiere mehr, sodass die Fütterung der Tiergruppe, und die Tiergruppe selbst gelöscht werden kann.

## Trigger 2



Der Trigger CheckBirthday wird immer ausgeführt, bevor ein Neues Tier hinzugefügt wird. Es wird geschaut, ob in dem Insert Statement ein Geburtstagsdatum vorkommt, welches in der Zukunft liegt. Falls der Fall ist, wird dieses Datum auf das heutige Datum gesetzt. Dadurch haben wir keine Daten welche gar nicht möglich sind, und bekommen keine Fehler bei der Berechnung des Alters.

## Trigger 3



Der Trigger RemoveEnclosuresIfNotUsed steht stark in Verbindung mit dem Trigger RemoveAnimalGroupAndFeeding.   
  
Der Trigger löscht ein Gehege, falls dieses nicht mehr von einer Tiergruppe gebraucht wird. Dies ist der Fall, sobald der Trigger RemoveAnimalGroupAndFeeding eine Tiergruppe löscht. Aus diesem Grund wird dieser Trigger immer ausgeführt, sobald eine Tiergruppe Manuell oder per Trigger gelöscht wird.

Die Löschung des Trigger wird also gemacht, falls die Anzahl der Tiergruppen, welche ein Gehege benutzen = 0 ist.

# Procedures

## Procedure 1

Diese Simple Prozedur namens «GetAge» gibt das Alter eines spezifischen Tieres zurück.   
Dazu nimmt die Prozedur das Geburtstagsdatum entgegen und gibt das Alter in einer Natürlichen Zahl zurück.

Durch die Funktion datediff() kann die Anzahl Tage zwischen 2 Daten berechnet werden. Um nun das Alter in Jahren zu bekommen, muss das Ganze noch durch 365 geteilt werden.

Wichtig ist, dass beim Aufrufen der Prozedur zuerst eine Subquerry durchgeführt werden muss, um das Geburtstagsdatum eines bestimmten Tieres zu bekommen.

## Procedure 2

# Arbeitsrapport

Wir haben uns dazu entscheiden für unseren Arbeitsrapport Github zu benutzen. Durch Github können wir direkt miteinander Arbeiten und Zeitgleich haben wir einen Automatischen Arbeitsrapport.

Die Commit-Messages dienen dabei als Indikator was von wem gemacht wurde. Sie werden in diesem Projekt nach folgendem Schema erstellt:

[Name] Commit Message

Dadurch haben wir ein Einheitliches Schema und es ist einfach zu erkennen was von wem zu welchem Zeitpunkt gemacht wurde.

**Die Commits aus Github werden in der folgenden Tabelle laufen ergänzt.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ersteller | Commit Message | Datum | Dauer | Link |
| Erin | Initial commit | 25.05.2022 | 1min | [Github.com](https://github.com/EriN-B/M153-Zoo/commit/a0cc1f9aea275b39e2823033b8775764bca86ce0) |
| Erin | Create Document Structure | 25.05.2022 | 10min | [Github.com](https://github.com/EriN-B/M153-Zoo/commit/fe0a77cfcc0c2bd3067c5b3dc29e6aa5c94b5549) |
| Erin | Update Documentation | 25.05.2022 | 20min | [Github.com](https://github.com/EriN-B/M153-Zoo/commit/c938da068dd840a47afae576bb60da1b0e68027f) |
| Erin | Update Documentation | 25.05.2022 | 20min | [Github.com](https://github.com/EriN-B/M153-Zoo/commit/e5de12fe73670c4aba02547d911e9503ff580e28) |
| Erin | Update Create Script | 01.06.2022 | 1h | [Github.com](https://github.com/EriN-B/M153-Zoo/commit/953dff6bbd09d18afa766be3633d5ab1bb47e13f) |
| Erin | Update Create,Insert and Query | 06.06.2022  (Auf Github das falsche Datum, da umgebungsvariable verändert) | 1h | [Github.com](https://github.com/EriN-B/M153-Zoo/commit/deb40adfcccad076be257c8733b91fd13c4d5ab0) |
| Moritz | Add Size to Animal | 07.07.2022 | 15min | [Github.com](https://github.com/EriN-B/M153-Zoo/commit/4a838dfb383a6db6344ca8845216dad2c7414011) |
| Moritz | Add Values of Size  to Animal in Insert | 07.07.2022 | 15min | [Github.com](https://github.com/EriN-B/M153-Zoo/commit/482d421e24351db55bd3f0d886fe9fc059eee826) |