M153 Projekt

Autoverleih Datenbank



Inhaltsverzeichnis

[Einleitung 3](#_Toc137056745)

[Kurzbeschreibung 3](#_Toc137056746)

[Rapport 3](#_Toc137056747)

[ER-Diagramm 3](#_Toc137056748)

[Relationales Modell 4](#_Toc137056749)

[Abfragen/Manipulationen 5](#_Toc137056750)

[Alle Vermietungen pro Kunde anzeigen 5](#_Toc137056751)

[Preis pro Tag für Fahrzeug anzeigen 5](#_Toc137056752)

[Fahrzeug hinzufügen 6](#_Toc137056753)

[Kunde löschen 7](#_Toc137056754)

[Stored Procedures 8](#_Toc137056755)

[Sp\_InsertFahrzeug 8](#_Toc137056756)

[Sp\_DeleteKunde 9](#_Toc137056757)

[Stored Functions 10](#_Toc137056758)

[GetTotalVermietungenForKunde 10](#_Toc137056759)

[GetFahrzeugPricePerTag 11](#_Toc137056760)

# Einleitung

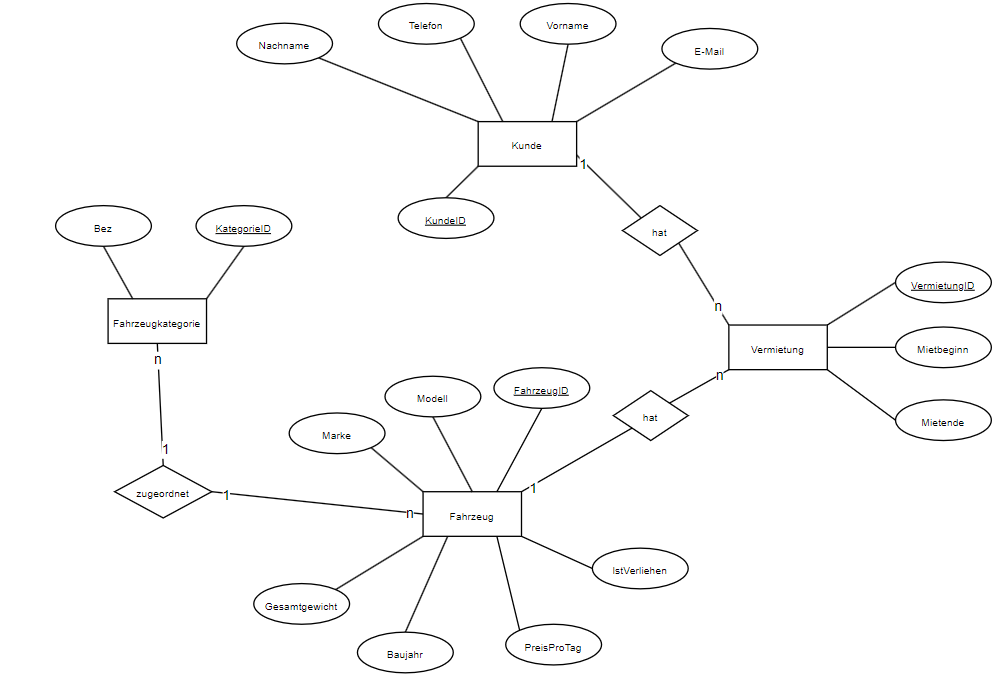
## Kurzbeschreibung

Das Projekt umfasst eine Datenbank für einen Autoverleihdienst. Die Datenbank mit dem Namen "AutoverleihDB" dient der Verwaltung von "Fahrzeugen", "Kunden", "Vermietungen" und "Fahrzeugkategorien". Die Datenbank besteht aus fünf Tabellen: "Fahrzeug", "Kunde", "Vermietung", "FahrzeugKategorie" und "FahrzeugKategorieZuordnung". Die Tabelle "Fahrzeug" enthält Informationen zu einzelnen Fahrzeugen wie "Marke", "Modell", "Baujahr" und "PreisProTag". Die Tabelle "Kunde" speichert Daten zu den Kunden wie "Vorname", "Nachname", "Telefon" und "E-Mail-Adresse". Die Tabelle "Vermietung" verknüpft Kunden und Fahrzeuge über Fremdschlüssel und speichert Informationen über den Mietzeitraum. Die Tabelle "FahrzeugKategorie" enthält verschiedene Fahrzeugkategorien mit optionalen Beschreibungen. Die Tabelle "FahrzeugKategorieZuordnung" dient der Zuordnung von Fahrzeugen zu den entsprechenden Kategorien. Das Projekt ermöglicht es, die Autoverleihdienste effizient zu verwalten und den Kunden eine reibungslose Erfahrung zu bieten.

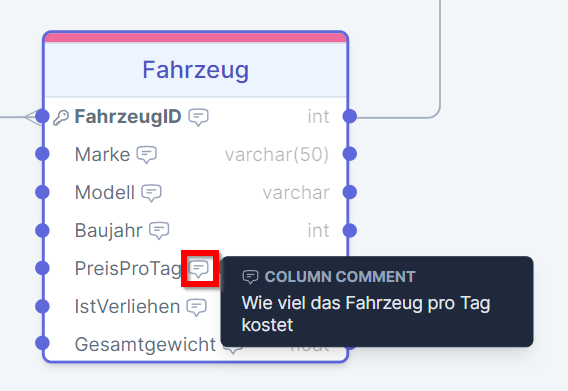
## Rapport

Wir haben den Rapport mit [Github](https://github.com/L1NN1I/M153_Projekt/commits/main) gemacht.

# ER-Diagramm



# imageRelationales Modell

Diagramm mit Beschreibung der Felder:   
<https://drawsql.app/teams/autoverleih/diagrams/autoverleih/embed>

# Abfragen/Manipulationen

## Alle Vermietungen pro Kunde anzeigen

Ein Bild, das Text, Screenshot, Software enthält.

Automatisch generierte Beschreibung

Mit dieser Funktion wird die Anzahl aller Vermietungen pro Kunde angezeigt.

## Preis pro Tag für Fahrzeug anzeigen

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

Mit dieser Funktion wird der Preis pro Tag für ein Fahrzeug ausgegeben.

## Fahrzeug hinzufügen

Ein Bild, das Text, Screenshot, Software enthält.

Automatisch generierte Beschreibung

Mithilfe dieser Funktion kann man ein Fahrzeug hinzufügen.

## Kunde löschen

Ein Bild, das Text, Screenshot, Software enthält.

Automatisch generierte Beschreibung

Mithilfe dieser Prozedur kann man einen Kunden löschen. Dafür muss man zuerst aber alle Vermietungen löschen da es sonst zu einem Fehler kommt.

Stored Procedures

Sp\_InsertFahrzeug

Ein Bild, das Text, Screenshot, Software enthält.

Automatisch generierte Beschreibung

Mithilfe der Prozedur «sp\_InsertFahrzeug» kann einfach ein neues Fahrzeug hinzugefügt werden. Beim Ausführen der Prozedur muss Marke, Modell, Baujahr und Preis pro Tag des Autos als Parameter mitgegeben werden. ReturnValue ist der Output der zurückgegeben wird. Als erstes wird geprüft, ob alle Parameter vorhanden sind. Falls nicht wird ein Fehler zurückgegeben. Die 11 nach der Fehlerbeschreibung heisst, dass der User den Fehler selbst beheben kann (die Skala geht von 1 bis 25, wobei 25 das Schlimmste ist. Die 1 ist der Status des Fehlers. Wir wussten nicht, was wir da mitgeben sollten, also haben wir uns für 1 entschieden. Hier geht die Skala von 0 bis 255. Mit Try/Catch wird nun versucht die Daten einzufüllen. Falls dies nicht funktioniert, geht es ins Catch uns es gibt einen Fehler.

## Sp\_DeleteKunde

Ein Bild, das Text, Screenshot, Software, Display enthält.

Automatisch generierte BeschreibungMithilfe der Prozedur «sp\_DeleteKunde» kann einfach ein Kunde gelöscht werden. Beim Ausführen der Prozedur muss die ID des Kunden als Parameter mitgegeben werden. Als erstes wird geprüft, ob die KundenID als Parameter übergeben wurde. Falls nicht wird ein Fehler zurückgegeben. Die 16 nach der Fehlerbeschreibung heisst, dass der User den Fehler selbst beheben kann und der Fehler Mittelschwer ist. Mit Try/Catch wird nun versucht den Kunden zu löschen. Falls dies nicht funktioniert, geht es ins Catch uns es gibt einen Fehler.

Stored Functions

GetTotalVermietungenForKundeEin Bild, das Text, Screenshot, Software enthält.

Automatisch generierte Beschreibung

Mithilfe der Funktion «GetTotalVermietungenForKunde» kann die Anzahl Vermietungen pro Kunde festgestellt werden. Beim Ausführen der Prozedur muss die ID des Kunden als Parameter mitgegeben werden. Als erstes wird geprüft, ob die KundenID als Parameter übergeben wurde. -1 ist ein Fehlercode für leere Argumente. Nun wird der Select ausgeführt. Wenn nun 0 zurückkommt wird NULL zurückgegeben. Ansonsten wird die Anzahl Vermietungen zurückgegeben.

## GetFahrzeugPricePerTag

Ein Bild, das Text, Screenshot, Software enthält.

Automatisch generierte Beschreibung

Mithilfe der Funktion «GetFahrzeugPricePerTag» kann der Preis pro Tag für ein bestimmtes Fahrzeug festgestellt werden. Beim Ausführen der Prozedur muss die ID des Fahrzeugs als Parameter mitgegeben werden. Es wird geprüft, ob die FahrzeugID als Parameter korrekt übergeben wurde. -1 ist ein Fehlercode für leere Argumente. Nun wird der Select ausgeführt und anschliessend der Preis pro Tag zurückgegeben.