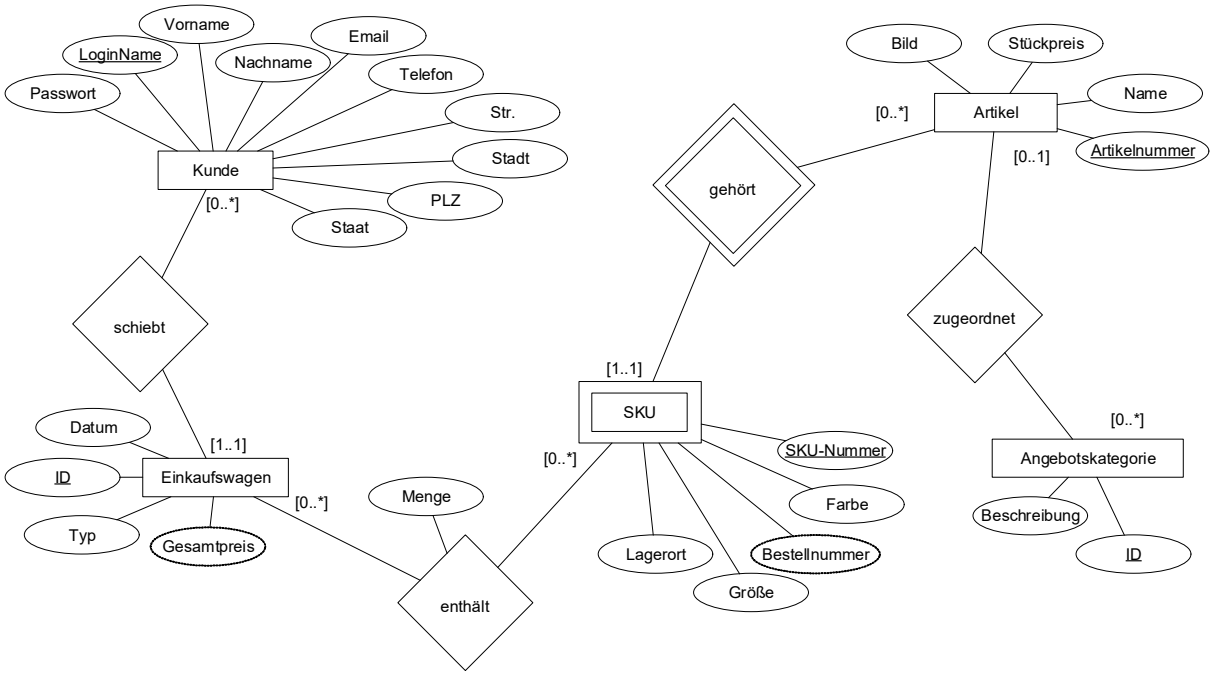
Exercise 09: Transformation into relational schemas

Task 1: Transformation of an ER model into a relational model

a) Transform the following ER model into a relational model using the rules for mapping from the lecture. Merge the resulting relations as much as possible.



Entities

Customer (LoginName, Password, FirstName, Surname, E-mail, TelephoneNumber, Street, City, Postcode, State)
ShoppingBasket (Date, ID, Type)
Item (ItemNumber, UnitPrice, Picture, Name)
SKU (SKU-Number, StorageLocation, Size, Colour, ItemNumber)
    ItemNumber *references* Item(ItemNumber)
OfferCategory (Description, ID)

Relationships

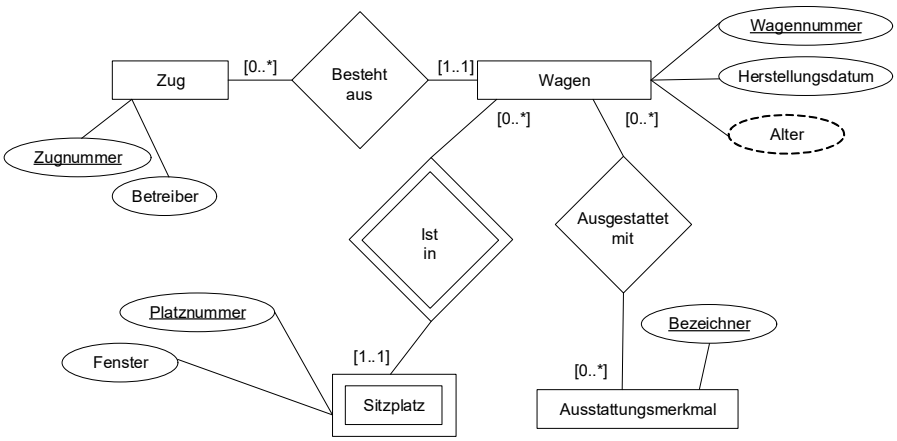
fills (LoginName not null, ID)
    LoginName *references* Customer (LoginName)
    ID *references* ShoppingBasket (ID)
contains (Quantity, ID, SKU-Number, ItemNumber)
    ID *references* ShoppingBasket(ID)
    SKU-Number *references* SKU(SKU-Number)
    ItemNumber *references* Item(ItemNumber)
assigned (ItemNumber, ID)
    ItemNumber references Item(ItemNumber)
    ID references OfferCategory(ID)

Merging

The 'fills' relationship is merged with the ShoppingBasket
ShoppingBasket' (ID, Type, Date, LoginName not null)
LoginName references Customer (LoginName)

The 'assigned' relationship is merged with Item
Item' (ItemNumber, Name, Price, Picture, ID)
ID references ItemCategory(ID)

b) Do the same with the following model.

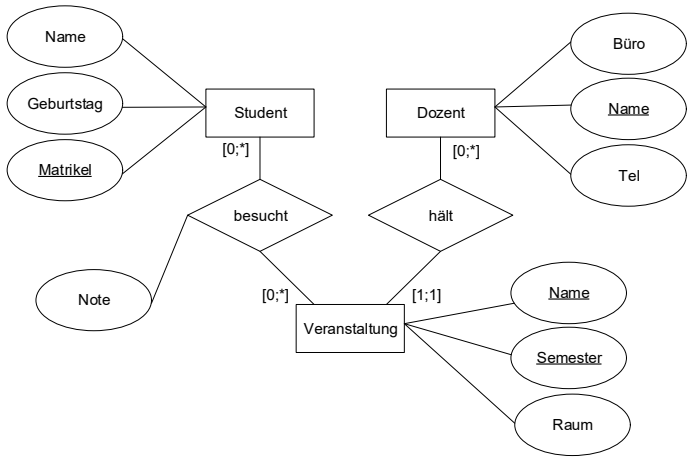


Entities
Train (TrainNumber, Operator)
Carriage (CarriageNumber, ManufacturingDate)
CarriageFeature (Identifier)
Seat (SeatNumber, CarriageNumber, Window)
CarriageNumber references Carriage(CarriageNumber)

Relationships
Consists_Of (TrainNumber, CarriageNumber)
TrainNumber references Train(TrainNumber)
CarriageNumber references Carriage(CarriageNumber)
Features(CarriageNumber, Identifier)
CarriageNumber references Train(CarriageNumber)
Identifier references CarriageFeature(Identifier)

Merging
Consists_Of is merged with Carriage:
Carriage' (CarriageNumber, ManufacturingDate, TrainNumber not null)
TrainNumber references Train(TrainNumber)

c) Do the same with the following model.

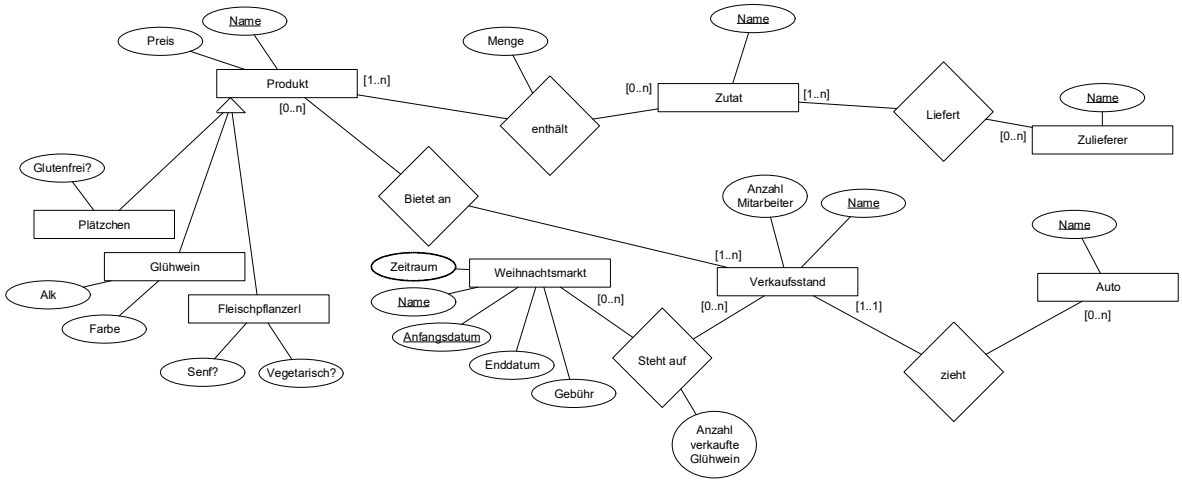


Entities
Student (Name, Date_of_birth, Matriculation)
Lecturer (Office, Name, Tel)
Event (Name, Semester, Room)

Relationships
attends (Matriculation, Name, Semester)
Matriculation *references* Student(Matriculation)
Name, Semester *references* Event(Name, Semester)
gives (EventName, Semester, LecturerName)
EventName, Semester *references* Event(Name, Semester)
LecturerName *references* Lecturer(Name)

Merging
gives is merged with Event
Event‘ (Name, Semester, Room, Lecturer not null)
Lecturer *references* Lecturer(Name)

d) Do the same with the following model.



Entities

Product (Name, Price, GlutenFree, Alc, Colour, Mustard, Vegetarian)
Ingredient (Name)
Supplier (Name)
Car (Name)
SalesBooth (Number of Employees, Name)
ChristmasMarket (Name, StartDate, EndDate, Fee)

Relationships

contains (ProductName, IngredientName, Amount)
ProductName *references* Product(Name)
IngredientName *references* Ingredient(Name)
offers (SalesBoothName, ProductName)
SalesBoothName *references* SalesBooth(Name)
ProductName *references* Product(Name)
delivers (SupplierName, IngredientName)
SupplierName *references* Supplier(Name)
IngredientName *references* Ingredient(Name)
pulls (SalesBoothName, CarName)
SalesBoothName *references* SalesBooth(Name)
CarName *references* Car(Name)
stands_on (NumberMulledWineSold, ChrMarketName, ChrMarketDate, SalesBoothName)
(ChrMarketName, ChrMarketDate) references ChristmasMarket (Name, StartDate)
SalesBoothName references SalesBooth (Name)

Merging

pulls is merged with SalesBooth
SalesBooth‘ (Number of Employees, Name, CarName not null)
CarName *references* Car(Name)