

Datenbanksystemspezifikation – Kino

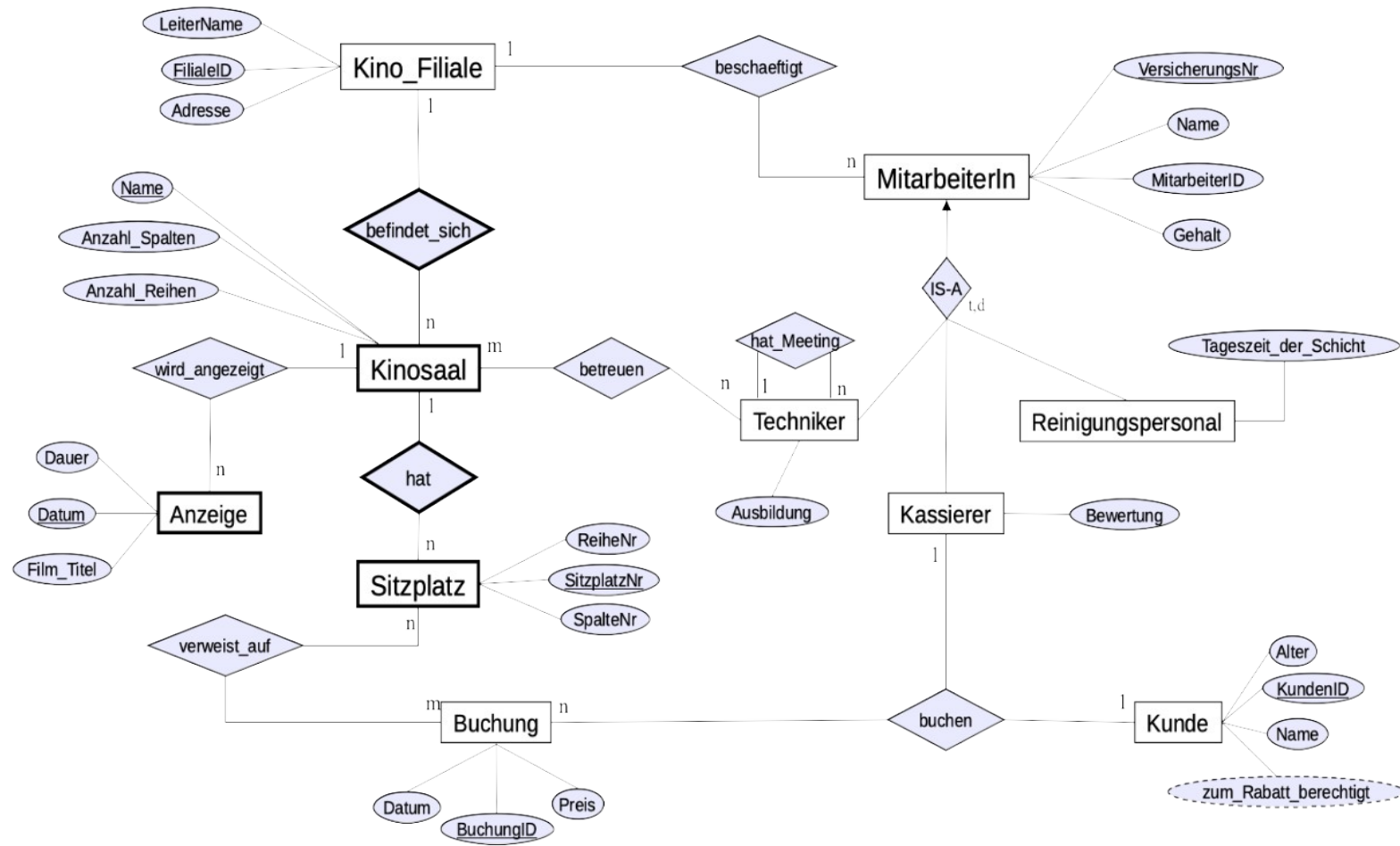
Peter Ivony - 11922002

Meilenstein 1a

Eine Filiale hat eine Leiter, eine Adresse und eine eindeutige ID. Sie beschäftigt viele MitarbeiterInnen, die können z.B.: Techniker, Kassierer oder Reinigungspersonal sein. Die MitarbeiterInnen sind durch ihre Versicherungsnummer identifiziert und ihre Gehalt, Name und IDs sind gespeichert. Weiterhin können TechnikerInnen miteinander Meetings organisieren. Eine Person vom Reinigungspersonal wird entweder zum Tagschicht oder Nachtschicht zugeordnet. Die Kassierer haben eine Bewertung ihrer Leistung und für die Techniker wird auch ihre Ausbildung gespeichert. Die Kassierer kann für eine Kunde eine Buchung erledigen, wobei der Preis, das Datum und eine eindeutige Buchung ID wird vermerkt. Für die KundInnen werden ihrer Namen, Alter und eine eindeutige Kunden ID gespeichert. Abhängig von ihrem Alter, wird es entschieden, ob sie zu einem Rabatt berechtigt sind.

In einer Filiale kann man verschiedene Kinosäle haben, die durch die Filiale und der Name des Kinosaals eindeutig identifiziert werden können. Bei diesen Sälen wird zusätzlich die Spaltenanzahl und die Anzahl der Reihen auch vermerkt. Kinosäle werden von mehreren Technikern betreut und ein(e) TechnikerIn kann auch mehrere Säle betreuen. In einem Saal findet man viele Sitzplätze, die mit einer eindeutigen Sitzplatznummer bezeichnet sind. Die Reihe- und Spaltennummer der Sitzplätze sind auch gespeichert. In dem Kinosaal werden viele Filmen angezeigt. Bei einer Anzeige wird der Titel und die Dauer des projizierten Films vermerkt und sie wird mithilfe das Datum der Anzeige und der Kinosaal eindeutig identifiziert. Eine Buchung verweist auf eine oder mehrere Sitzplätze bei genau einer Anzeige eines Films.

Meilenstein 1b



Meilenstein 2

Relationenschema für das DBS:

Kino_Filiale(FilialeID, LeiterName, Adresse)	PS = {FilialeID}
Kinosaal(FilialeID, Name_, Kapazitaet, Anzahl_Reihen)	PS = {FilialeID, Name_} FS = {FilialeID \diamond Kino_Filiale}
Sitzplatz(FilialeID, Name_, SitzplatzNr, ReiheNr, SpalteNr)	PS = {FilialeID, Name_, SitzplatzNr} FS = {Name_ \diamond Kinosaal, FilialeID \diamond Kino_Filiale}
Anzeige(FilialeID, Name_, Datum, Dauer, Film_Titel)	PS = {FilialeID, Name_, Datum} FS = {Name_ \diamond Kinosaal, FilialeID \diamond Kino_Filiale}
MitarbeiterIn(VersicherungsNr, Name_, Gehalt)	PS = {VersicherungsNr} FS = {FilialeID \diamond Kino_Filiale}
Techniker(VersicherungsNr, Gehalt, Ausbildung, getroffeneMitarbeiterInID, FilialeID)	PS = {VersicherungsNr} FS = {FilialeID \diamond Kino_Filiale, getroffeneMitarbeiterInID \diamond Techniker}
Kassierer(MitarbeiterInID, Name_, VersicherungsNr, Gehalt, Bewertung, FilialeID)	PS = {VersicherungsNr} FS = {FilialeID \diamond Kino_Filiale}
Reinigungspersonal(VersicherungsNr, Gehalt, SchichtTageszeit, FilialeID)	PS = {VersicherungsNr} FS = {FilialeID \diamond Kino_Filiale}
Kunde(KundenID, Name_, Age)	PS = {KundenID}
Buchung(BuchungID, Preis, Datum, MitarbeiterInID, KundenID)	PS = {BuchungID} FS = {VersicherungsNr \diamond Kassierer, KundenID \diamond Kunde}
verweist_auf(BuchungID, FilialeID, Name_, SitzplatzNr)	FS = {BuchungID \diamond Buchung, FilialeID, Name_, SitzplatzNr \diamond Sitzplatz}
betreuen(FilialeID, Name_, MitarbeiterInID)	FS = {FilialeID, Name_ \diamond Kinosaal, VersicherungsNr \diamond Techniker}

Meilenstein 4

Ich habe versucht mit PHP eine Webseite zu erstellen, wobei die registrierte Kunden eine Liste von Anzeigen abfragen können, und mit einer ‚admin‘-Login kann man die gespeicherte Daten vom DBMS beliebig abfragen/löschen/bearbeiten. Erstens machte ich dieses Teil. Die PHP Unterlagen sind geteilt in Ordnern, wo jedes Ordner beinhaltet die Code für eine der CRUD-Operationen. Bei der Java-Implementierung, hatte ich am Anfang 3 Klassen vorbereitet, die DatabaseHelper Klasse, was nur für die Kommunikation zwischen das Programm und das Datenbanksystem sicherstellt, eine RandomDataGenerator Klasse, die für das Einfügevorfahren semi-zufällige Daten generiert, und letztens die Hauptklasse, wo diese Operationen initiiert werden. Später brauchte ich für Abfragen eine TheaterRoom Klasse, was eine Repräsentation von einem Tupel aus der Kinosaal-Relation ist.