

Begriffe:

Zeile: Datenobjekt, Datensatz, Entität, Tupel

Spalte(+name): Attribut, Eigenschaft, Merkmal

Tabelle: Entitätsmenge, Datenobjektsmenge, Relation

Zelle: Value, Datenwert, Attributwert

Domäne: Wertebereich

Redundanz: Selbe Info mehreren Orten abgespeichert (z.B. Backup).

Dies sollte aber in der elektronischen Speicherung sonst vermieden werden, denn es ist wertlos und nutzlos und führt zu Problemen. Gift für Datenbank. Beispiel mit Uhrzeit.

Konsistenz -> Eindeutig, Aussagekräftig, Stimmt, Verhält, Frei von Widersprüchen und Redundanzen

Inkonsistenz -> Wenn es nicht eindeutig ist , nicht aktuell(Zürich – Zuerich, Zivilstand geändert), Redundanzen, Fehler bei referenzieren

Anomalien (inkonsistente Daten/Redundanzen):

Update-Anomalie (Änderungsanomalie): Wenn nur ein Teil auf dem Neusten Stand ist und der Rest nicht.

Delete-Anomalie (Löschungsanomalie): Wenn es nicht sicher ist, welche Tupel man löschen muss.

Insert-Anomalie (Einfügeanomalie): Wenn für das gleiche 2 verschiedene Bezeichnungen vorhanden sind z.B. Produktion und Production

Datenmanagement:

Unter dem Datenmanagement fasst man alle betrieblichen und technischen Funktionen der Datenarchitektur, der Datenadministration und der Datentechnik zusammen, die der unternehmensweiten Datenhaltung, Datenpflege und Datennutzung dienen.

Filesystem:

Oft redundant, Wiederverwendung schwierig, Integrität nicht gewährleistet, grosser Programmieraufwand

Datenbanken:

Klar strukturiert, keine Redundanz, Applikationsunabhängig, leicht erweiterbar, Integrität garantiert, sicher

Zusammengesetzte Attribute:

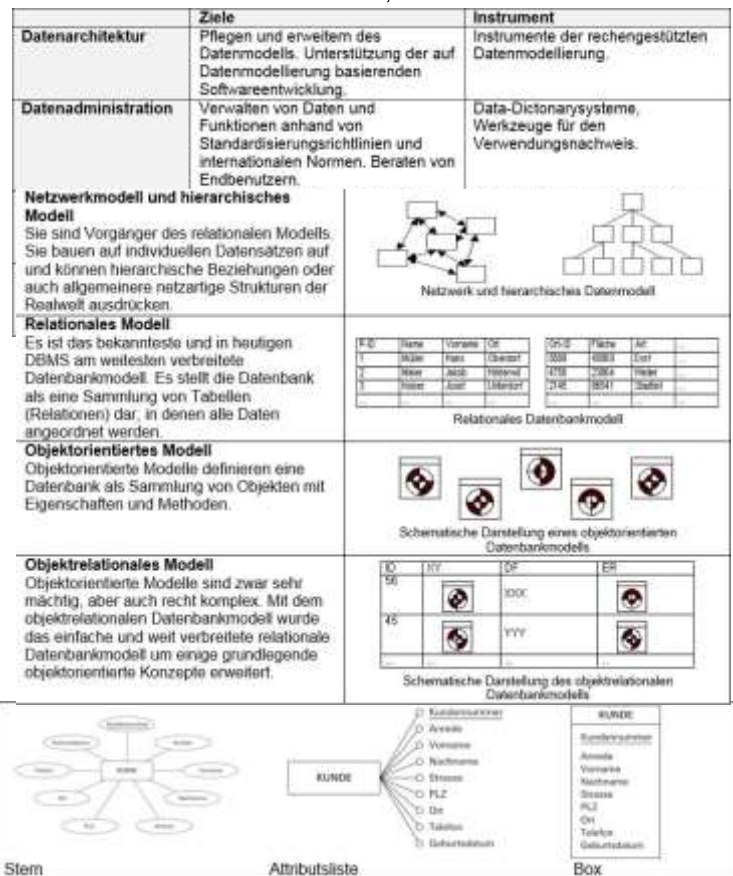
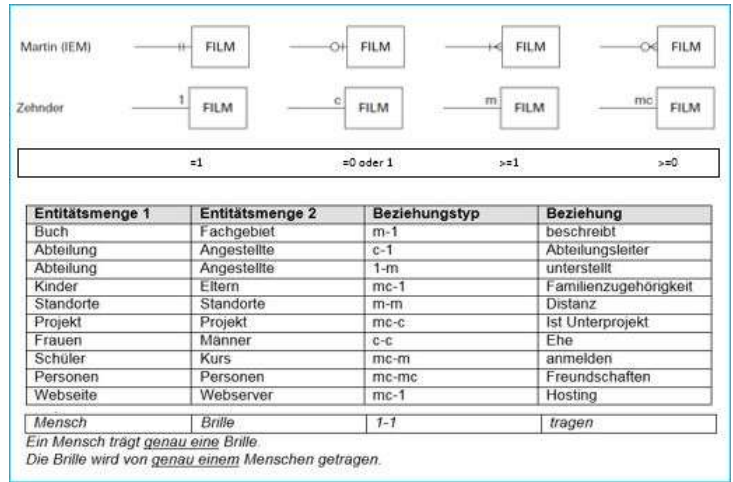
Adresse -> Strasse, Nr. ,PLZ, Ort

Darstellung:

TABELLENNAME(Attr1(#),Attr2(#)...))

ERD: (Aufbau danach kann es implementiert werden)

Phase	Tätigkeit
Schritt 1	Tabelle identifizieren.
Schritt 2	Die Attribute der Tabellen definieren.
Schritt 3	Die Beziehungen zwischen den einzelnen Tabellen klären.
Schritt 4	Die Tabellen mit Attributen und die Beziehungen zwischen den Tabellen als Modell aufzeichnen .



Begriffe:	Datentypen:
NOT NULL: Spalte muß mit einem Wert ungleich NULL gefüllt sein. UNIQUE: Spalte oder Spaltenkombination ist eindeutig. PRIMARY KEY: Spalte oder Spaltenkombination ist Primärschlüssel. FOREIGN KEY: Spalte oder Spaltenkombination muß in einer separaten Tabelle als Schlüssel vorhanden sein. ON DELETE CASCADE: Löschen eines Datensatzes führt zum kaskadierenden Löschen der, über foreign key constraints verbundenen, Datensätze. CHECK: Boolescher Ausdruck ist wahr	Nvarchar Bigint , bit, int, char, date, datetime, nchar, text, null (leer)

Befehl	Beschreibung
SELECT * FROM Table1 UNION SELECT * FROM Table2	Resultate vereinen

Data Definition Language (DDL)

Befehl	Beschreibung
CREATE / ALTER / DROP TABLE / DATABASE / INDEX	Erstellen / ändern / löschen
CREATE TABLE BUCH_STICHWORT(Buch_StichwortID bigint IDENTITY (1,1) NOT NULL, StichwortID bigint NULL, Titel nvarchar (50) NULL, BuchID bigint NULL, CONSTRAINT PK_BUCH_STICHWORT PRIMARY KEY (Buch_StichwortID), CONSTRAINT FK_BUCH_STICHWORT_BUCH FOREIGN KEY (BuchID) REFERENCES BUCH (BuchID), CONSTRAINT FK_BUCH_STICHWORT_STICHWORT FOREIGN KEY (StichwortID) REFERENCES STICHWORT (StichwortID) -- Beziehungen BUCH --> BUCH STICHWORT ALTER TABLE BUCH_STICHWORT ADD CONSTRAINT FK_BUCH_STICHWORT_BUCH FOREIGN KEY (BuchID) REFERENCES BUCH (BuchID) ALTER TABLE BUCH_STICHWORT DROP CONSTRAINT FK_BUCH_STICHWORT_BUCH CREATE INDEX "INDEX_NAME" ON "TABELLEN_NAME" (SPALTEN_NAME) CREATE INDEX IDX_ADRESSE ON ADRESSE (Strasse, PLZ, Land) DROP INDEX INDEX_NAME ON TABELLEN_NAME	Tabelle erstellen Datentyp meistens bigint oder nvarchar(x) Zuweisen eines Primary Key Zuweisen eines Foreign Key Referenz zur anderen Tabelle und dort den Primary Key nehmen
ALTER TABLE BUCH_STICHWORT ADD CONSTRAINT FK_BUCH_STICHWORT_BUCH FOREIGN KEY (BuchID) REFERENCES BUCH (BuchID)	Foreign Key nachträglich einfügen
ALTER TABLE BUCH_STICHWORT DROP CONSTRAINT FK_BUCH_STICHWORT_BUCH	Foreign Key löschen / Entsprechenden Constraint Name brauchen
CREATE INDEX "INDEX_NAME" ON "TABELLEN_NAME" (SPALTEN_NAME) CREATE INDEX IDX_ADRESSE ON ADRESSE (Strasse, PLZ, Land) DROP INDEX INDEX_NAME ON TABELLEN_NAME	Aufbau Index Beispiel Index Löschen Index

Befehl	Beschreibung
CREATE / RESTORE BACKUP	Erstellen / Wiederherstellen
BACKUP DATABASE DBName TO DISK = 'path'	Backup erstellen
RESTORE DATABASE DBName FROM DISK = 'path'	Backup wiederherstellen

Data Manipulation Language (DML)

Befehl optional	Beschreibung
SELECT [AGGREGATION()] Attributname1 [AS „Alias“, Attributname2,... FROM TABELLENNAME [WHERE AttributnameN [=, <, <=, >, >=, LIKE, BETWEEN, IS, NOT] Datenwert] [AND, OR, NOT] [ORDER BY Attributname1 [DESC, ASC]] [GROUP BY Attributname1] [HAVING COUNT(*) > 2] [LIMIT N];	[Aggregation wie MIN/MAX/DISTINCT],[Ein Alias setzen] Eine Einschränkung setzen z.B. IS NULL, NOT IN [DESC=Absteigend ZYX ASC=Aufsteigend ABC] [Gruppieren] [Wenn eine Aggregatsfunktion gebraucht wird] [Eine Limite setzen]
INSERT INTO TABELLENNAME (Attributname1, Attributname2,...) VALUES ('Datenwert1','Datenwert2',...);	Eine Zeile hinzufügen
UPDATE TABELLENNAME SET Attributname1 = Datenwert1 AttributnameN = DatenwertN WHERE AttributnameN [=, <, <=, >, >=, LIKE, BETWEEN, IS, NOT] Datenwert [AND, OR, NOT]	Einen Datenwert aktualisieren
DELETE FROM TABELLENNAME WHERE AttributnameN [=, <, <=, >, >=, LIKE, BETWEEN, IS, NOT] Datenwert [AND, OR, NOT]	Eine Zeile löschen
SELECT AttributnameN FROM TabelleL [INNER / LEFT / RIGHT JOIN] TabelleR ON AttrL=AttrR;	Eine weitere Tabelle hinzufügen

Aggregatsfunktionen	Beschreibung
SUM	Summiert die Werte in einer Spalte. Der Datentyp muss numerisch sein!
AVG	Gibt den Mittelwert in einer Spalte zurück. Der Datentyp muss numerisch sein!
COUNT	Gibt die Anzahl von Einträgen in einer Gruppe zurück.
DISTINCT	Jeden Wert nur einmal
COUNT DISTINCT	Gibt die Anzahl von eindeutigen Werten ungleich NULL in einer Gruppe zurück.
MIN	Gibt den kleinsten Wert in einer Gruppe zurück.
MAX	Gibt den grössten Wert in einer Gruppe zurück.

Bestimmte Befehle	Beschreibung
SELECT ROUND(AVG(GehaltAttribut)/5,2)*5 FROM Tabelle;	Auf Franken runden
SELECT * FROM Personen WHERE Vorname BETWEEN ,F%' AND ,!%';	%, * heisst es könnte unendlich weiter gehen !, ? ist für ein Buchstabe