

# Progettare Basi di Dati

Prof. Nicoletta D'Alpaos & Prof. Andrea Borghesan

Entità-Relazione

Elementi di informatica per l'economia

# Teoria della normalizzazione

La **teoria della normalizzazione** ha come scopo quello di fornire metodi per progettare basi di dati senza anomalie.

La realtà che si intende analizzare tramite una base di dati può avere più **schemi**, equivalenti fra loro, che la descrivono.

Alcuni di questi schemi possono presentare degli inconvenienti (ridondanze o anomalie di comportamento) che rendono inadeguato l'utilizzo efficiente delle informazioni.

# Anomalie di comportamento

- Ridondanza
- Anomalie da modifica
- Anomalie da inserzione
- Anomalie da cancellazione

# Anomalie di comportamento

- Con **ridondanza** si intende la non necessaria ripetizione della stessa informazione con il conseguente spreco di spazio;
- Con **anomalie da modifica** si intende la necessità di ripetere, in caso di cambiamento di una informazione, la modifica ovunque tale informazione è, duplicata;
- Con **anomalia da inserzione** si intende la necessità di inserire ulteriori informazioni, non strettamente necessarie per poter inserire un nuovo dato;
- Per **anomalie di cancellazione** si intende l'eliminazione di alcune informazioni in conseguenza della cancellazione di altre.

# *Esempio:*

Codice Cliente	Nome	Via	Codice Articolo	Quantità ordinata
10	Rossi Antonio	Piazza Garibaldi	A25	20
10	Rossi Antonio	Piazza Garibaldi	B10	10
20	Bianchi Maria	Viale Fermi	C12	3

*Si osserva che:*

- Le informazioni anagrafiche si ripetono tante volte quanti sono gli ordini fatti;
- Se un cliente cambia indirizzo, bisogna modificare più righe;
- Non si può caricare un nuovo cliente se questo prima non ha effettuato un ordine;
- Se un articolo deve essere eliminato (per esempio C12), si possono cancellare anche delle informazioni anagrafiche.

# Teoria della normalizzazione

La **teoria della normalizzazione** si occupa dei seguenti problemi su schemi (*soprattutto*) relazionali:

- Definire quando due schemi sono equivalenti
- Definire criteri di bontà per schemi (*forme normali*)  
(Una forma normale è una proprietà di uno schema relazionale che ne garantisce la “qualità”, cioè l’assenza di determinati difetti)
- Determinare metodi algoritmici per ottenere uno schema “migliore” ed equivalente a partire da uno schema non in forma normale (*normalizzazione*)

# Forme Normali

Relazione in 1NF

Relazione in 2 NF

Relazione in 3 NF

Relazione in BCNF

Relazione in 4 NF

Relazione in 5 NF

# Normalizzazione

- L'attività che permette di trasformare schemi non normalizzati in schemi che soddisfano una forma normale è detta **normalizzazione**;
- La normalizzazione va utilizzata come **tecnica di verifica dei risultati della progettazione** di una base di dati;
- Non costituisce quindi una metodologia di progettazione;



# Normalizzazione

Per costruire uno schema relazionale:

1. Si parte da un buon schema E-R e lo si “traduce”;
2. Si costruiscono direttamente le relazioni e poi si correggono quelle che presentano “anomalie”;
3. Si analizza uno schema relazionale già esistente e lo si modifica/completa;

# PRIMA FORMA NORMALE

**Ogni campo deve contenere un solo valore:** (la 1NF è chiamata anche **FORMA ATOMICA**).

Un campo non atomico può essere strutturato o multivalore:

- ogni attributo strutturato viene sostituito da tanti attributi quanti sono i valori che appaiono nell'attributo strutturato stesso;
- ogni attributo multivalore diventa un attributo semplice spezzando la tupla che contiene il valore in tante tuple quanti sono i valori descritti nell'attributo;

*(Se una tabella si trova nella prima forma normale, non c'è alcuna possibilità di suddividere ulteriormente i campi della tabella)*

# 1NF

*Esempio:*

Tabella STUDENTI:

**(#MATRICOLA, nomeCognome, indirizzo, voti)**

- “NomeCognome” e “indirizzo” sono campi strutturati e vengono divisi in “nome”, “cognome”, “via”, “cap”, “città”
- Voti è un campo multivalore, di conseguenza ogni riga viene divisa in più righe, una per ogni voto riportato

# 1NF

(continua):

Tabella STUDENTI:

(#MATRICOLA, nome, cognome, via, cap, città, voto)

Attenzione: “#Matricola” non può più essere chiave della relazione perché uno studente deve fare più esami:

(#MATRICOLA, #ESAME, nome, cognome, via, cap, città, voto)

# SECONDA FORMA NORMALE

Una tabella è in 2NF se è in 1NF e se tutti i suoi attributi sono dipendenti dall'intera chiave primaria.

- La tabella viene divisa in più tabelle che soddisfano la condizione.

# 2NF

*Esempio:*

Tabella STUDENTI:

(#MATRICOLA, #ESAME, nome, cognome, via, cap, città, voto)

- I campi “nome”, “cognome”, “via”, “cap”, “città” dipendono solo da una parte della chiave (#MATRICOLA)

Tabella STUDENTI:

(#MATRICOLA, nome, cognome, via, cap, città)

Tabella ESAMI:

(#MATRICOLA, #ESAME, voto)

# Qualità delle decomposizioni

Una decomposizione dovrebbe sempre soddisfare due proprietà:

- la decomposizione senza perdita, che garantisce la ricostruzione delle informazioni originarie
- la conservazione delle dipendenze, che garantisce il mantenimento dei vincoli di integrità originali

# TERZA FORMA NORMALE

Una tabella è in 3NF se è in 2NF e tutti gli attributi che non sono chiavi sono mutuamente indipendenti (eliminazione delle dipendenze transitive).

- La tabella viene divisa in più tabelle che soddisfano la condizione.



# 3NF

*Esempio:*

Tabella STUDENTI:

(#MATRICOLA, nome, cognome, via, cap, città, #isee, *tassa*)



- Con #isee si intende il codice dello scaglione di reddito che determina la *tassa* scolastica.

Tabella STUDENTI:

(#MATRICOLA, nome, cognome, via, cap, città, #isee)

Tabella TASSA:

(#ISEE, *tassa*)

Una lista di informazioni è così strutturata:

Codice	Articoli	Colore	<u>Cod. Colore</u>	Prezzo
1	Sedia	Rosso; Verde; Giallo	R; V; G;	100
2	Tavolo	Rosso; Verde;	R; V;	200
...				

Costruire da queste informazioni una base dati normalizzata

Una lista di informazioni è così strutturata:

<u>Codice Spettacolo</u>	<u>Replica</u>	Titolo	Data	Compagnia	Autore
X	1	Il berretto a sonagli	2 gen 05	Teatro nuovo	Pirandello
X	2	Il berretto a sonagli	3 gen 05	Teatro nuovo	Pirandello
Y	1	I rusteghi	4 gen 05	Veneziana	Goldoni
Y	2	I rusteghi	5 gen 05	Veneziana	Goldoni
Y	3	I rusteghi	7 gen 05	Veneziana	Goldoni
Z	1	Casa di bambole	8 gen 05	AllStars	Ibsen

Costruire da queste informazioni una base dati normalizzata

Una lista di informazioni è così strutturata:

<u>Numero Ordine</u>	<u>Progressivo Riga</u>	Codice Cliente	Codice Articolo	Quantità ordinata	Prezzo unitario
1	1	A10	12-500L	20	50
1	2	A10	24-400P	10	43
2	1	B20	45-200X	10	44
2	2	B20	12-500L	10	50
3	1	A10	12-500L	15	50

*(NOTA: un cliente può fare più ordini, il prezzo di un certo articolo non cambia)*

Costruire da queste informazioni una base dati normalizzata

Una lista di informazioni è così strutturata:

Titolo	Autore	Biografia	ISBN	Subject	Pag	Editore	Sede (editore)
Introduzione alle tecnologie WEB	Vito Roberto Marco Frailis	Roberto è professore ordinario a Udine, Frailis è dottorando in informatica	88386 6181	Informatica, WEB design, Ateneo	426	McGraw-Hill	Milano

Costruire da queste informazioni una base dati normalizzata

Una lista di informazioni è così strutturata:

Matricola	Nominativo	Indirizzo	Progetto	Descrizione	Ore lavorate
10	Mario Rossi	Via Leopardi, 15 Milano	P-10; P-21	Sviluppo sito web; Sviluppo grafica	80; 24
20	Maria Pia Bianchi	Piaz.le Mazzini 23, Milano	P-21; P-44; A-57	Sviluppo grafica; Acquisizione immagini; Documentazione	130; 10; 20
30	Gianni Verdi	Corso Pascoli, 100 Padova	P-44	Acquisizione immagini;	30
40	Carlo Manzi	Corso Petrarca, 44 Padova	P-30	Sviluppo XML;	81
.....	.....	.....	...		....

Costruire da queste informazioni una base dati normalizzata

Una lista di informazioni è così strutturata:

Titolo	Autore	ISBN	ID copia	Prestito
Zoologia	<u>Mitchell,</u> <u>Mutchmor,</u> <u>Dolphin</u>	88-08-12176	1, 2, 3, 4, 5	V, F, F, F; F
Il foglio elettronico per economia	<u>Atzeni, De</u> <u>Checchi,</u> <u>Sindoni,</u> <u>Tirelli</u>	88-386-6255	1, 2, 3, 4, 5, 6, 7, 8, 9, 10	V, V, V, V, V, V, F, F, V, V
Java 2. I fondamenti	<u>Horstmann,</u> <u>Cornell</u>	88-386-4367	1, 2	F, F

(NOTA: “prestito” contiene V (=vero) se il libro è prestato, F(= falso) se il libro è disponibile )

Costruire da queste informazioni una base dati normalizzata