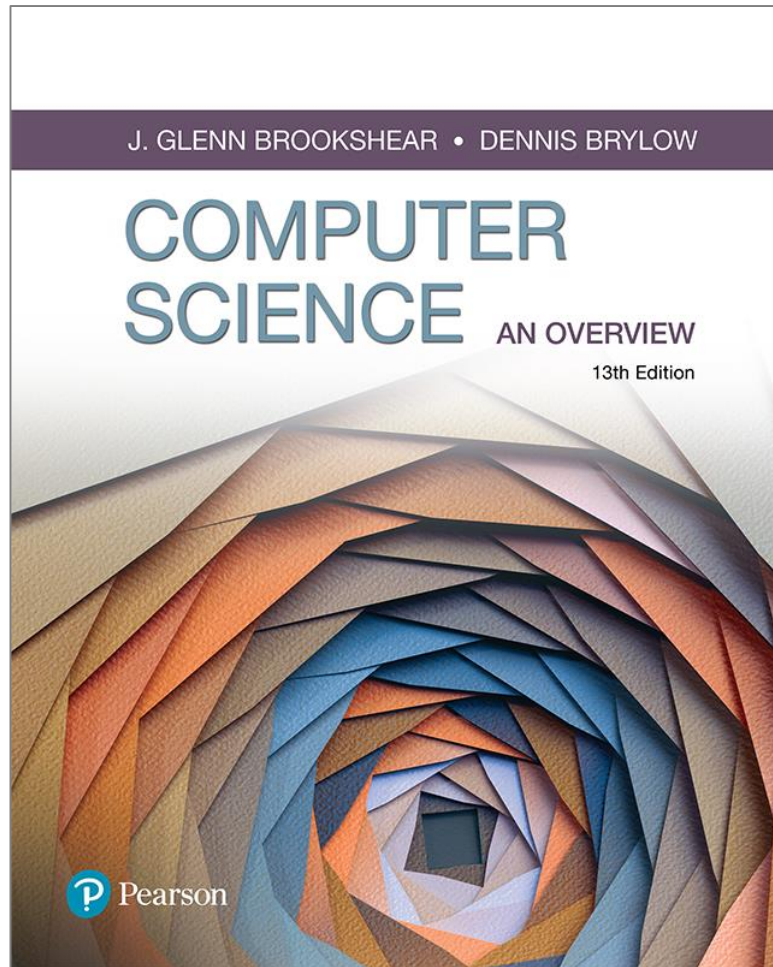


# Computer Science An Overview

13<sup>th</sup> Edition



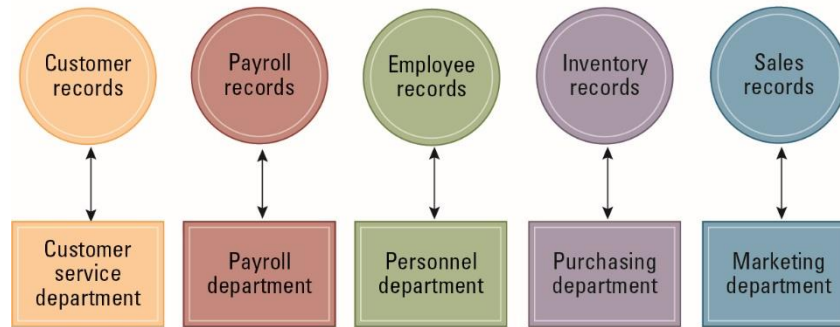
## Chapter 9 Database Systems

# Fondamenti delle basi di dati

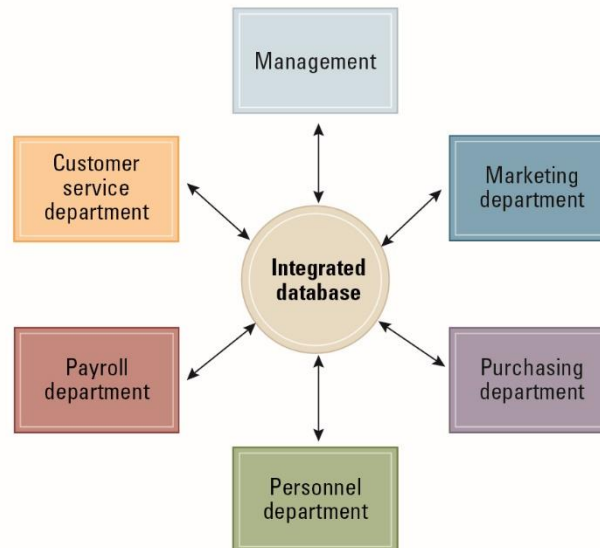
Database: una collezione di dati che è multidimensionale nel senso che contiene collegamenti interni tra gli elementi in modo che le sue informazioni siano accessibili da diversi punti di vista.

# File vs database organization

a. File-oriented information system



b. Database-oriented information system



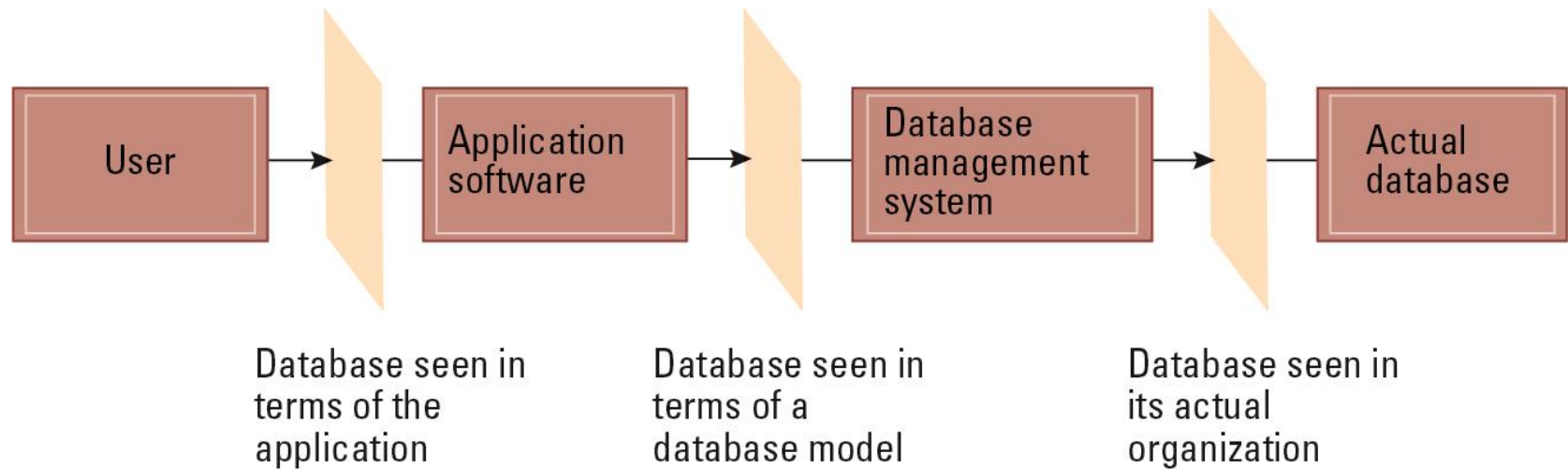
# Il ruolo degli schemi

- **Schema:** è una descrizione di tutta la struttura della base di dati utilizzata dal software di gestione.
- **Subschema:** descrive solo la parte di base di dati attinente alle particolari necessità di un utente.

# Database Management Systems

- **Database Management System (DBMS):** un livello di software che manipola un database in risposta alle richieste delle applicazioni
- **Database Distribuito:** Un database memorizzato su più macchine
  - DBMS maschera questo dettaglio organizzazionale ai suoi utenti
- **Indipendenza dei dati:** possibilità di modificare l'assetto organizzativo della base di dati senza modificare il software applicativo.

# Implementazione di un database



# Modelli di database

- **Modello di Database** : struttura concettuale di un database
  - Relational database model
  - Object-oriented database model

# Il modello relazionale

- **Relazione:** una tabella rettangolare
  - **Attributo:** una colonna della tabella
  - **Tupla:** una riga della tabella



# Relazione contenente informazioni sui dipendenti

Empl Id	Name	Address	SSN
25X15	Joe E. Baker	33 Nowhere St.	111223333
34Y70	Cheryl H. Clark	563 Downtown Ave.	999009999
23Y34	G. Jerry Smith	1555 Circle Dr.	111005555
•	•	•	•
•	•	•	•
•	•	•	•

# Problemi dei progetti relazionali

- Evitare concetti multipli entro una relazione
  - Può condurre a dati ridondanti
  - Una tupla può essere anche cancellata, ma deve essere con informazione non correlata

# Migliorare un progetto relazionale

- **Decomposizione:** Dividere le colonne di una relazione in due o più relazioni, duplicando quelle colonne necessarie per mantenere le relazioni
  - Decomposizione **Lossless** o **nonloss** : una “corretta” decomposizione che non perde nessuna informazione.

# Una relazione contenente ridondanza

Empl Id	Name	Address	SSN	Job Id	Job Title	Skill Code	Dept	Start Date	Term Date
25X15	Joe E. Baker	33 Nowhere St.	111223333	F5	Floor manager	FM3	Sales	9-1-2009	9-30-2010
25X15	Joe E. Baker	33 Nowhere St.	111223333	D7	Dept. head	K2	Sales	10-1-2010	*
34Y70	Cheryl H. Clark	563 Downtown Ave.	999009999	F5	Floor manager	FM3	Sales	10-1-2009	*
23Y34	G. Jerry Smith	1555 Circle Dr.	111005555	S25X	Secretary	T5	Personnel	3-1-1999	4-30-2010
23Y34	G. Jerry Smith	1555 Circle Dr.	111005555	S26Z	Secretary	T6	Accounting	5-1-2010	*
•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•

# Base di dati dei dipendenti costituita da tre relazioni

**EMPLOYEE relation**

Empl Id	Name	Address	SSN
25X15	Joe E. Baker	33 Nowhere St.	111223333
34Y70	Cheryl H. Clark	563 Downtown Ave.	999009999
23Y34	G. Jerry Smith	1555 Circle Dr.	111005555
•	•	•	•
•	•	•	•
•	•	•	•

**JOB relation**

Job Id	Job Title	Skill Code	Dept
S25X	Secretary	T5	Personnel
S26Z	Secretary	T6	Accounting
F5	Floor manager	FM3	Sales
•	•	•	•
•	•	•	•
•	•	•	•

**ASSIGNMENT relation**

Empl Id	Job Id	Start Date	Term Date
23Y34	S25X	3-1-1999	4-30-2010
34Y70	F5	10-1-2009	*
23Y34	S26Z	5-1-2010	*
•	•	•	•
•	•	•	•
•	•	•	•

# Come trovare i reparti in cui ha lavorato il dipendente 23Y34

## EMPLOYEE relation

Empl Id	Name	Address	SSN
25X15	Joe E. Baker	33 Nowhere St.	111223333
34Y70	Cheryl H. Clark	563 Downtown Ave.	999009999
23Y34	G. Jerry Smith	1555 Circle Dr.	111005555
•	•	•	•
•	•	•	•
•	•	•	•

## JOB relation

Job Id	Job Title	Skill Code	Dept
S25X	Secretary	T5	Personnel
S26Z	Secretary	T6	Accounting
F5	Floor manager	FM3	Sales
•	•	•	•
•	•	•	•
•	•	•	•

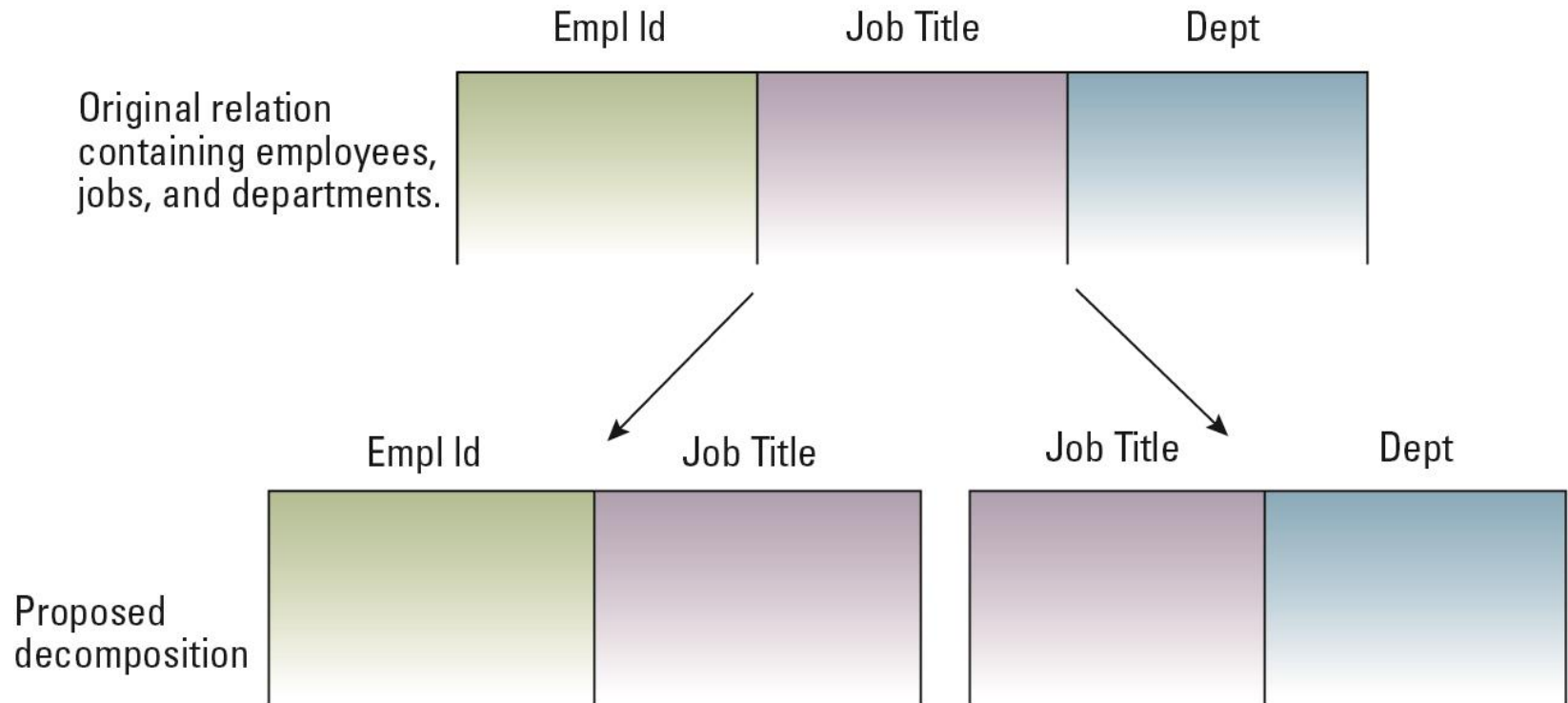
are contained in the personnel and accounting departments.

## ASSIGNMENT relation

Empl Id	Job Id	Start Date	Term Date
23Y34	S25X	3-1-1999	4-30-2010
34Y70	F5	10-1-2009	*
23Y34	S26Z	5-1-2010	*
•	•	•	•
•	•	•	•
•	•	•	•

The jobs held by employee 23Y34

# Relazione e scomposizione proposta



# Operazioni sulle relazioni

- **SELECT:** estrae righe
- **PROJECT:** estrae colonne
- **JOIN:** unisce relazioni diverse in un'unica relazione



# Operazione SELECT

**EMPLOYEE relation**

EmplId	Name	Address	SSN
25X15	Joe E. Baker	33 Nowhere St.	111223333
34Y70	Cheryl H. Clark	563 Downtown Ave.	999009999
23Y34	G. Jerry Smith	1555 Circle Dr.	111005555
•	•	•	•
•	•	•	•
•	•	•	•

**NEW** ← SELECT from **EMPLOYEE** where EmplId = '34Y70'

**NEW relation**

EmplId	Name	Address	SSN
34Y70	Cheryl H. Clark	563 Downtown Ave.	999009999

# Operazione PROJECT

**EMPLOYEE relation**

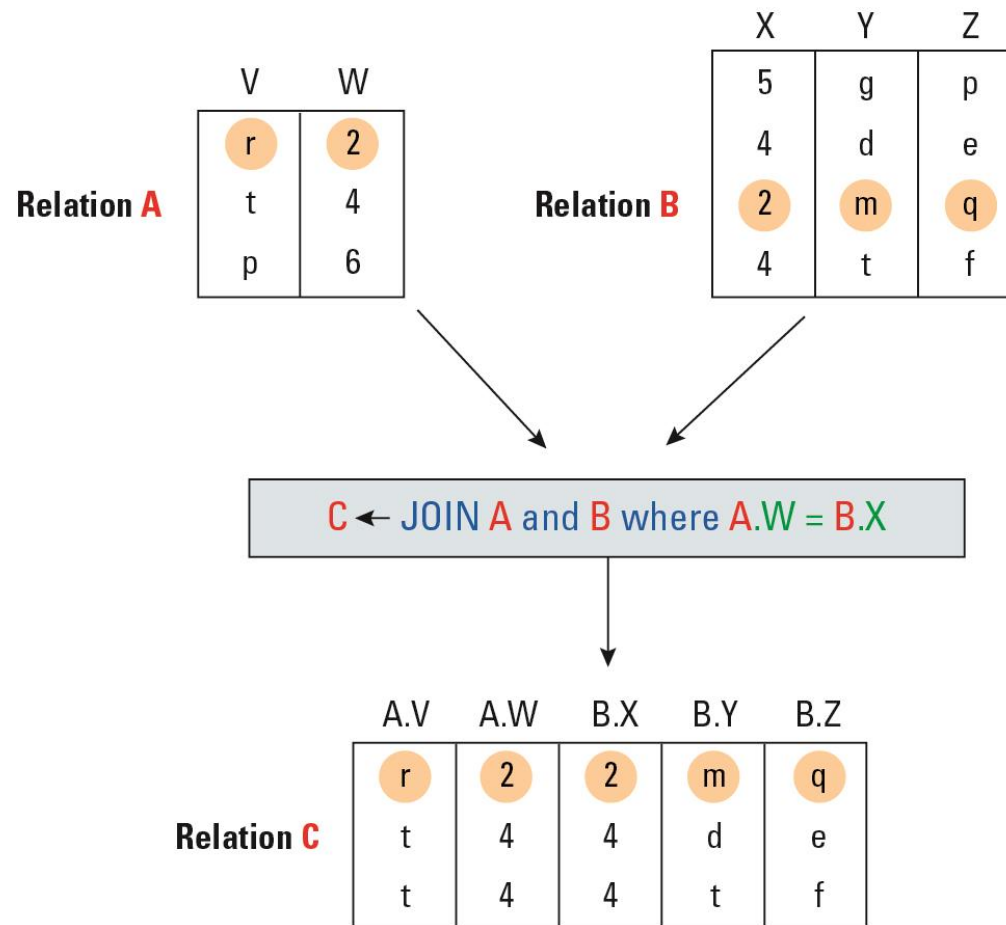
EmplId	Name	Address	SSN
25X15	Joe E. Baker	33 Nowhere St.	111223333
24Y70	Cheryl H. Clark	563 Downtown Ave.	999009999
23Y34	G. Jerry Smith	1555 Circle Dr.	111005555
•	•	•	•
•	•	•	•
•	•	•	•

**MAIL** ← PROJECT Name, Address from EMPLOYEE

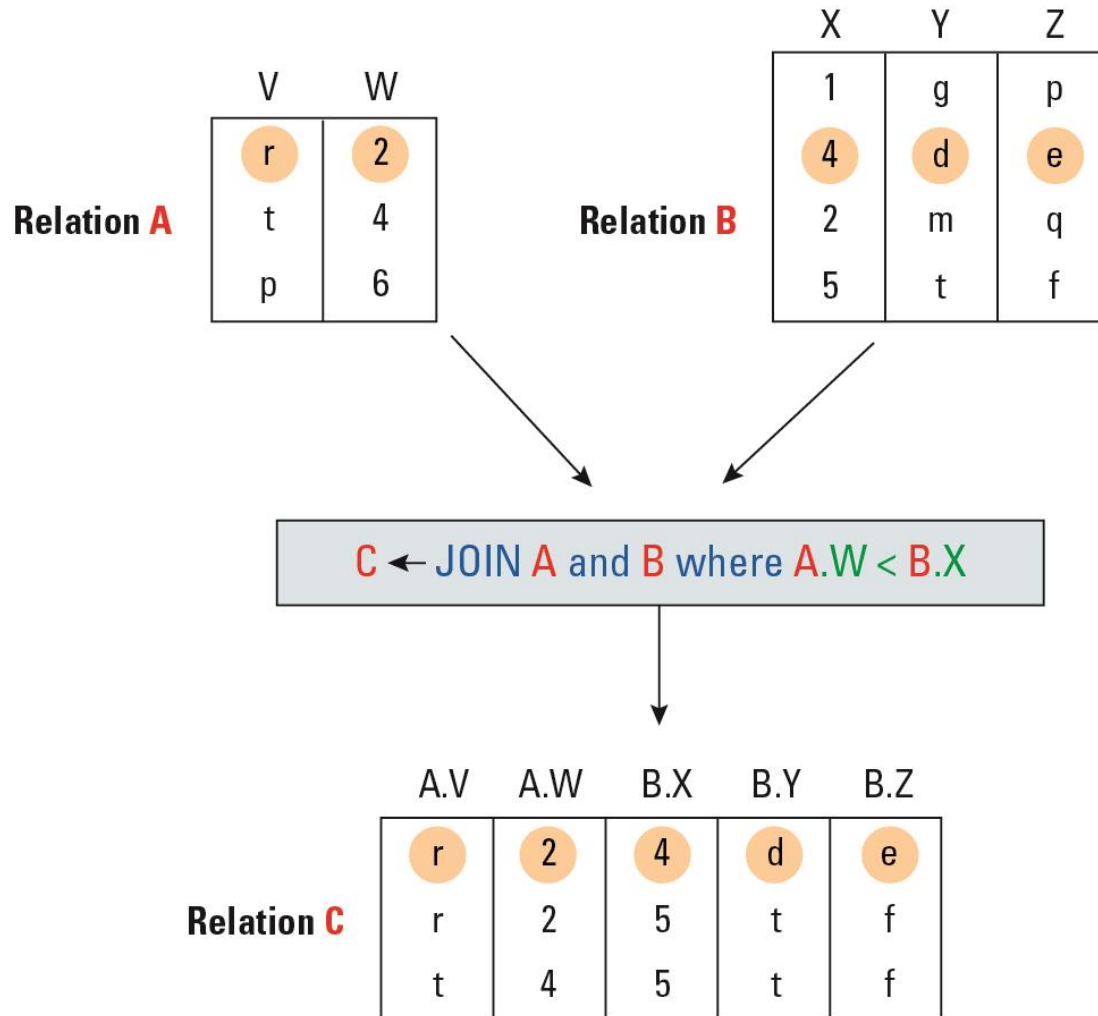
**MAIL relation**

Name	Address
Joe E. Baker	33 Nowhere St.
Cheryl H. Clark	563 Downtown Ave.
G. Jerry Smith	1555 Circle Dr.
•	•
•	•
•	•

# Operazione JOIN



# Altro esempio operazione Join



# Un'applicazione dell'operazione Join

**ASSIGNMENT relation**

Empl Id	Job Id	Start Date	Term Date
23Y34	S25X	3-1-1999	4-30-2010
34Y70	F5	10-1-2009	*
25X15	S26Z	5-1-2010	*
•	•	•	•
•	•	•	•
•	•	•	•

**JOB relation**

Job Id	Job Title	Skill Code	Dept
S25X	Secretary	T5	Personnel
S26Z	Secretary	T6	Accounting
F5	Floor manager	FM3	Sales
•	•	•	•
•	•	•	•
•	•	•	•

**NEW1** ← JOIN ASSIGNMENT and JOB where ASSIGNMENT.JobId = JOB.JobId

**NEW1 relation**

ASSIGNMENT Empl Id	ASSIGNMENT Job Id	ASSIGNMENT StartDate	ASSIGNMENT TermDate	JOB Job Id	JOB JobTitle	JOB SkillCode	JOB Dept
23Y34	S25X	3-1-1999	4-30-2010	S25X	Secretary	T5	Personnel
34Y70	F5	10-1-2009	*	F5	Floor manager	FM3	Sales
25X15	S26Z	5-1-2010	*	S26Z	Secretary	T6	Accounting
•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•

# Structured Query Language (SQL)

- Operazioni per manipolare tuple
  - insert
  - update
  - delete
  - select

# SQL Examples

- ```
SELECT EmpId, Dept
FROM Assignment, Job
WHERE Assignment.JobId = Job.JobId
      AND Assignment.TermData = '*';
```
- ```
INSERT INTO Employee
VALUES ('43212', 'Sue A. Burt',
      '33 Fair St.', '444661111');
```

# SQL Examples (continued)

- DELETE FROM Employee  
WHERE Name = 'G. Jerry Smith';
- UPDATE Employee  
SET Address = '1812 Napoleon Ave.'  
WHERE Name = 'Joe E. Baker';



# Chiave di una relazione

Identificazione univoca di ogni entità

qualsiasi insieme di attributi: chiave  
candidata

Chiave primaria

la chiave candidata che è stata scelta

Hair Color	Eye Color
Blond	Blue
Brown	Green
Black	Brown
Red	Blue
Blond	Blue

chiave primaria

ID	Hair Color	Eye Color
1	Blond	Blue
2	Brown	Green
3	Black	Brown
4	Red	Blue
5	Blond	Blue

# Prodotto tra tabelle

Crea una super-tabella che ha tutti i campi di entrambe le tabelle originali

Tabella1 × Tabella2

# polari

	Nome	Capitale	Lat
▶	Falkland Islands	Stanley	-51.42
	Faroe Islands	Torshavn	62.00
	Greenland	Nuuk	64.20
	Iceland	Reykjavik	64.13
	New Zealand	Wellington	-41.28
	Finland	Helsinki	60.17
	Svalbard	Longyearbyen	78.13

# poli

	NS
▶	Nord
	Sud
☀	

polari X poli →

	Nome	Capitale	Lat	NS
	Falkland Islands	Stanley	-51.42	Nord
	Falkland Islands	Stanley	-51.42	Sud
	Faroe Islands	Torshavn	62.00	Nord
	Faroe Islands	Torshavn	62.00	Sud
	Greenland	Nuuk	64.20	Nord
	Greenland	Nuuk	64.20	Sud
	Iceland	Reykjavik	64.13	Nord
	Iceland	Reykjavik	64.13	Sud
	New Zealand	Wellington	-41.28	Nord
	New Zealand	Wellington	-41.28	Sud
	Finland	Helsinki	60.17	Nord
	Finland	Helsinki	60.17	Sud
	Svalbard	Longyearbyen	78.13	Nord
	Svalbard	Longyearbyen	78.13	Sud

(polari X poli) ((Lat>0 AND ... 'Sud')) →

	Nome	Capitale	Lat	NS
	Falkland Islands	Stanley	-51.42	Sud
	Faroe Islands	Torshavn	62.00	Nord
	Greenland	Nuuk	64.20	Nord
	Iceland	Reykjavik	64.13	Nord
	New Zealand	Wellington	-41.28	Sud
	Finland	Helsinki	60.17	Nord
	Svalbard	Longyearbyen	78.13	Nord

# Unione di tabelle

Concatena due tabelle in una unica  
devono avere esattamente gli stessi  
attributi

Tabella1 + Tabella2

## equatoriali

	Nome	Capitale	Lat
▶	Kenya	Nairobi	-1.27
	Uganda	Kampala	0.20
	Ecuador	Quito	-0.22
	Gabon	Libreville	0.30
	Kiribati	Tarawa	1.40
	Nauru	Yaren	-1.00
	Singapore	Singapore	1.30
	Sao Tome	Sao Tome	0.00

## polari

	Nome	Capitale	Lat
▶	Falkland Is	Stanley	-51.42
	Faroe Islar	Torshavn	62.00
	Greenland	Nuuk	64.20
	Iceland	Reykjavik	64.13
	New Zeala	Wellington	-41.28
	Finland	Helsinki	60.17
	Svalbard	Longyearbye	78.13

polari + equatoriali →

	Nome	Capitale	Lat
▶	Kenya	Nairobi	-1.27
	Uganda	Kampala	0.20
	Ecuador	Quito	-0.22
	Falkland Is	Stanley	-51.42
	Faroe Islar	Torshavn	62.00
	Gabon	Libreville	0.30
	Greenland	Nuuk	64.20
	Iceland	Reykjavik	64.13
	Kiribati	Tarawa	1.40
	Nauru	Yaren	-1.00
	New Zeala	Wellington	-41.28
	Finland	Helsinki	60.17
	Singapore	Singapore	1.30
	Sao Tome	Sao Tome	0.00
	Svalbard	Longyearbye	78.13

# Join

Genera una nuova tabella che contiene solo le righe corrispondenti tra le due tabelle

ogni riga con tutte le colonne delle due tabelle

Tabella1 Join Tabella2 On Corrispondenza



# Città

# wDB1

ID	NomeCittà	Paese	Popolazione
412	Gaza	GAZA	409'680
413	Bhubaneswar	IND	411'542
414	Bologna	I	411'803
415	Culiacan	MEX	415'046
416	Lyon	F	415'487
417	Ashgabat	TM	416'400
418	Kisangani	ZRE	417'517
419	Kaliningrad	R	419'000

Nome	Co...	Capitale	Lat	Lon	Popol	Area
Philippines	RP	Manila	14.58	120.98	74480848	299764
Vietnam	VN	Hanoi	21.03	105.87	73976973	329560
Iran	IR	Tehran	35.68	51.42	66094264	1648000
Egypt	ET	Cairo	29.87	31.33	63575107	1001450
Turkey	TR	Ankara	39.95	32.88	62484478	780580
Thailand	THA	Bangkok	13.73	100.50	58851357	514000
United Kin	GB	London	51.48	0.00	58489975	244820
France	F	Paris	48.82	2.48	58317450	547030
Italy	I	Rome	41.80	12.60	57460274	301230
Ethiopia	ETH	Addis Ababa	9.00	38.75	57171662	1127130

Città Join wDB1 on Città.paese=wDB1.Codice

ID	NomeCittà	Paese	Popolazione	Nome	Co...	Capitale	Lat	Lon	Popol	Area
413	Bhubaneswar	IND	411'542	India	IND	New Delhi	28.58	77.20	95210769	3287590
414	Bologna	I	411'803	Italy	I	Rome	41.80	12.60	57460274	301230
415	Culiacan	MEX	415'046	Mexico	MEX	Mexico City	19.40	-99.20	95772462	1972550
416	Lyon	F	415'487	France	F	Paris	48.82	2.48	58317450	547030
417	Ashgabat	TM	416'400	Turkmenis	TM	Ashgabat	37.58	58.20	4149283	488100
418	Kisangani	ZRE	417'517	Zaire	ZRE	Kinshasa	-4.30	15.20	46498539	2345410
419	Kaliningrad	R	419'000	Russia	R	Moscow	55.77	37.67	14817848	1707520
420	Charlotte	USA	419'539	United Sta	USA	Washington	38.50	-77.00	26647627	9372610
421	Saltillo	MEX	420'947	Mexico	MEX	Mexico City	19.40	-99.20	95772462	1972550
422	San Salvador	EC	422'570	El Salvador	EC	San Salvador	13.70	-89.70	5828087	31040

# Gestione dell'integrità delle basi di dati

- **Transazione:** una sequenza di operazioni che deve avvenire assieme
  - Ex.: trasferimento fondi tra conti bancari
- **Transaction log:** Un record non volatile di ogni attività di transazione, costruito prima che la transazione sia stata eseguita
  - **Commit point:** Il punto al quale una transazione è stata memorizzata nel log
  - **Roll-back:** Il processo di annullamento di una transazione

# Gestione dell'integrità delle basi di dati (II)

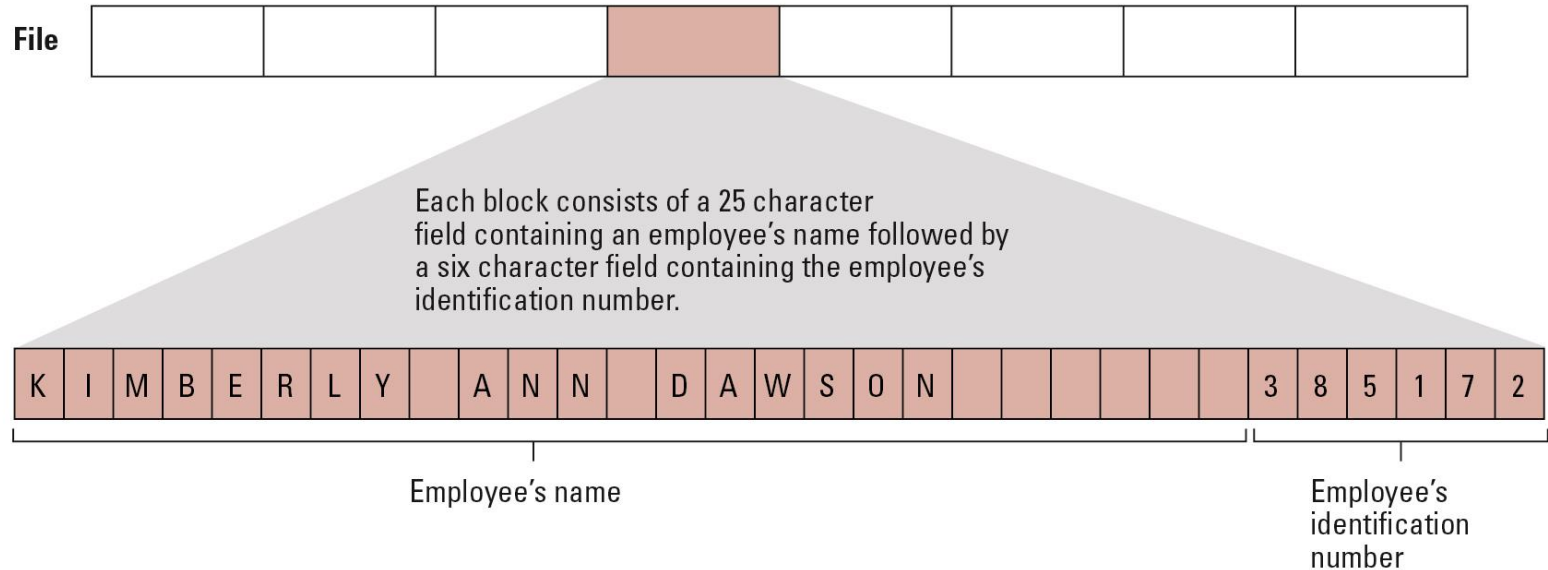
- Problemi di accesso simultaneo
  - Problema del totale errato
  - Problema dell'aggiornamento perso
- **Locking**: preventing others from accessing data being used by a transaction
  - **Shared** lock: usato quando vengono letti dati
  - **Exclusive** lock: usato quando vengono modificati dati

# Strutture di file tradizionali

- **File sequenziali:** Un file il cui contenuto può essere letto solamente in ordine
  - Il lettore deve essere capace di rilevare l'end-of-file (EOF)
  - I dati possono essere memorizzati in record logici, ordinati da un campo chiave

# Struttura di un semplice file degli impiegati implementato come file di testo

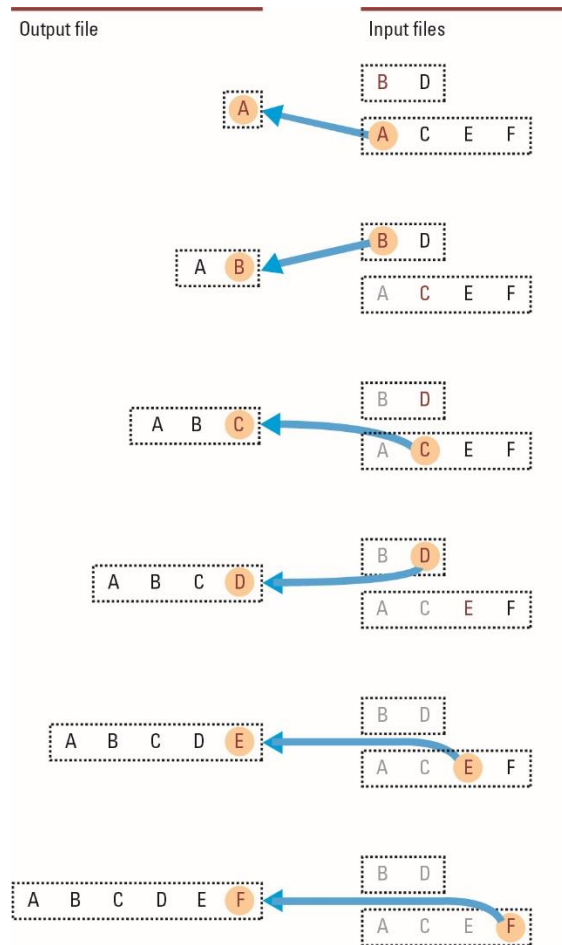
File consists of a sequence of blocks each containing 31 characters.



# Procedura per fondere due file sequenziali

```
def MergeFiles (InputFileA, InputFileB, OutputFile):  
    if (ambedue i file di input all' EOF):  
        Stop, con OutputFile vuoto  
  
    if (InputFileA non all' EOF):  
        Dichiarare il suo primo record come record corrente  
  
    if (InputFileB non all' EOF):  
        Dichiarare il suo primo record come record corrente  
  
    while (nessuno dei due file di input all' EOF):  
        Poni il record corrente con il valore "più piccolo" del campo chiave in  
        OutputFile  
        if (tale record corrente è l'ultimo record del corrispondente file di input):  
            Dichiarare che quell file di input è all' EOF  
        else:  
            Dichiarare come record corrente del file di input il successivo record del  
            file  
  
    Iniziando dal record corrente del file di input che non è all' EOF, copia i  
    restanti record in FileOutput.
```

# Applicazione dell'algoritmo di fusione

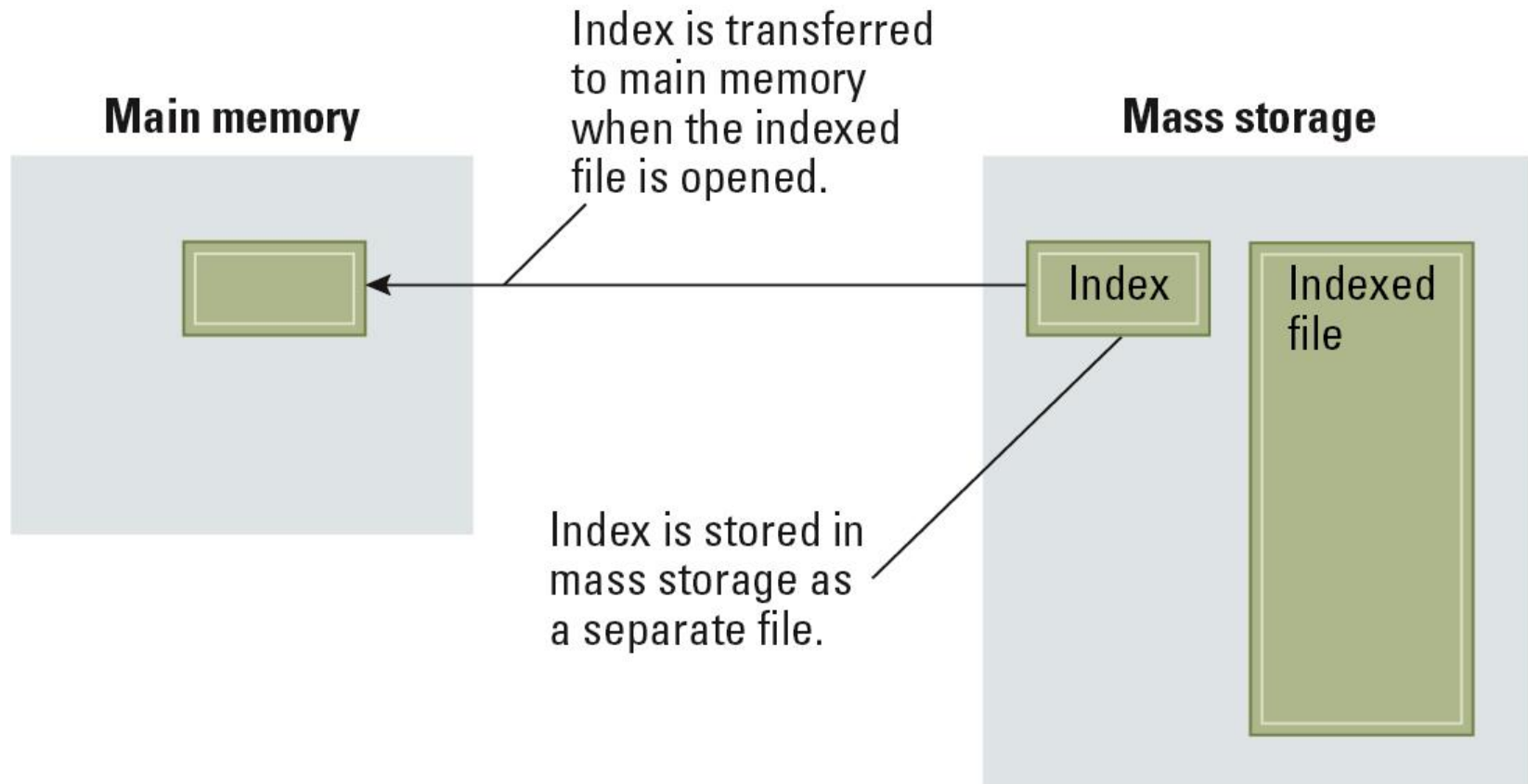


# File indicizzati

- **Index:** una lista di valori chiave la locazione dei loro record associati
  - Modo efficiente per identificare velocemente la locazione di un record desiderato
  - L'indice è memorizzato come un file separato



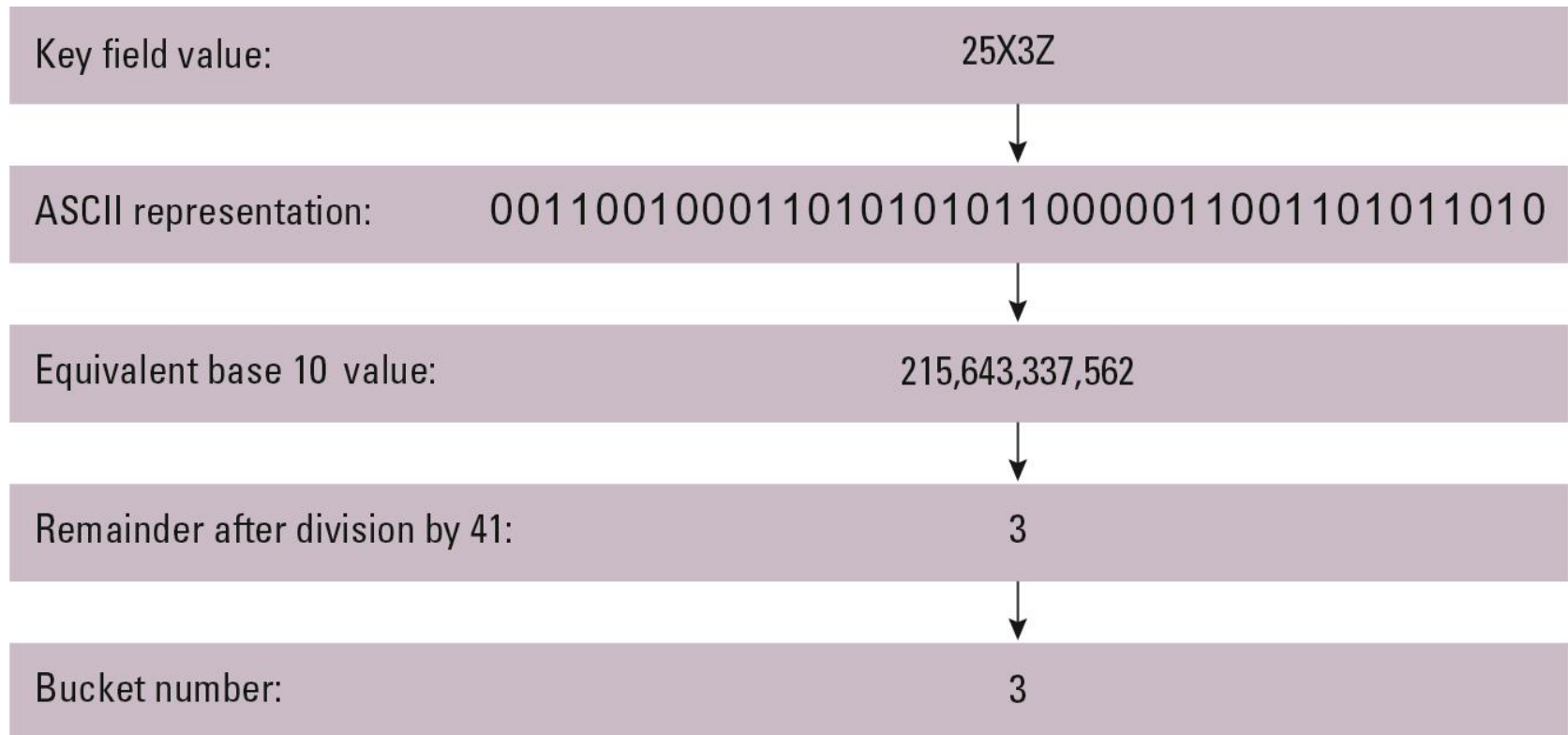
# Apertura di un file indicizzato



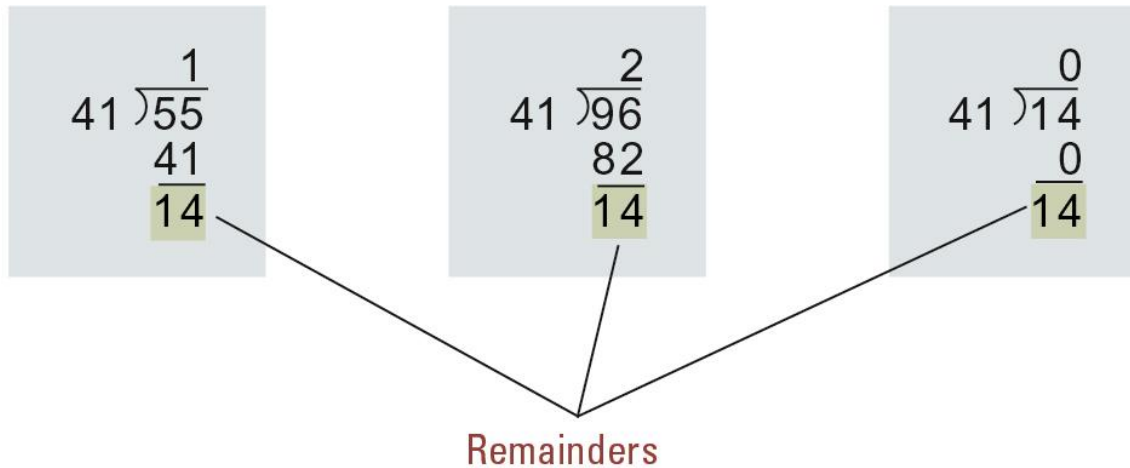
# Hash Files

- Ogni record ha un campo chiave
- Lo spazio di memorizzazione è diviso in **buckets**
- Una **funzione hash** computa un numero di bucket per ogni valore chiave
- Ogni record è memorizzato nel bucket corrispondente all'hash della sua chiave

# Hashing the key field value 25X3Z to one of 41 buckets



# Rudimenti di un Sistema hash



When divided by 41, the key field values of 14, 55, and 96 each produce a remainder of 14. Thus these records are stored in bucket 14.



# Collisions in Hashing

- **Collisione:** Il caso di due chiavi che danno lo stesso valore
  - Maggiori problem quando la tabella è piena più del 75%
  - Soluzione: aumentare il numero di buckets e ricostruzione dell'hash

# Data Mining

- **Data Mining:** Area dell'Informatica che tratta la scoperta di schemi ricorrenti nelle collezioni di dati
- **Data warehouse:** Raccolte di dati statiche
  - **Data cube:** Dati presentati da molte prospettive per abilitare il significato

# Strategie di Data Mining

- Class description
- Class discrimination
- Cluster analysis
- Association analysis
- Outlier analysis
- Sequential pattern analysis