



## Sistemi Informativi L-A

---

Anno Accademico 2007-2008

Home Page del corso:  
<http://www-db.deis.unibo.it/courses/SIL-A/>

Versione elettronica: [Intro2007.pdf](#)

Sistemi Informativi L-A



## Docente e Tutor

---

**Docente: Ilaria Bartolini**

DEIS - Facoltà di Ingegneria  
Viale Risorgimento, 2 - 40136, Bologna

Orario di ricevimento:

- **lunedì 16-18** (previo appuntamento)  
c/o palazzina IEIIT (ex-CSITE)

Contatti:

- Tel. 051 20 93550
- email: [ibartolini@deis.unibo.it](mailto:ibartolini@deis.unibo.it)
- Web: <http://www-db.deis.unibo.it/~ibartolini/>

**Tutor: Barbara Pettazoni**

Orario di ricevimento:

- **martedì 14-16** (previo appuntamento)  
c/o laboratorio CCIB-ST (LAB3)

Contatti:

- email: [barbara.pettazoni@studio.unibo.it](mailto:barbara.pettazoni@studio.unibo.it)

## Obiettivi del corso

- Il corso intende fornire gli strumenti necessari per imparare a **utilizzare le Basi di Dati relazionali**, presentando le nozioni di base del modello relazionale e dei linguaggi relativi (algebra relazionale e SQL)
- Il corso è propedeutico a quello di **Sistemi Informativi L-B**, in cui l'accento viene viceversa posto su aspetti di **progettazione e amministrazione** di Basi di Dati
- **N.B.** Il corso di **Sistemi Informativi L-A** è abbinato nel **I ciclo** di lezioni a quello di **Sistemi Informativi L-B**. Le lezioni dei due corsi **NON vengono svolte in parallelo**, pertanto le lezioni e le esercitazioni di laboratorio di SI L-B inizieranno al termine di quelle di SI L-A (dopo circa **5-6 settimane** di lezione)



## Orario

- **Orario delle Lezioni:**
  - Lunedì 12-14 aula **6.1**
  - Martedì 12-14 aula **3.3**
  - Giovedì 16-19 aula **3.3** (a partire da **GIOVEDÌ 04/10/07** utilizzeremo, a settimane alterne, anche l'ora **15-16**)
- **Orario del Laboratorio (LAB3):**
  - Venerdì 9-12: esercitazione guidata dal **tutor**
  - Venerdì 12-15: esercitazione **libera** utilizzando il materiale predisposto

➡ Si inizia **venerdì 05/10/07!** ☺

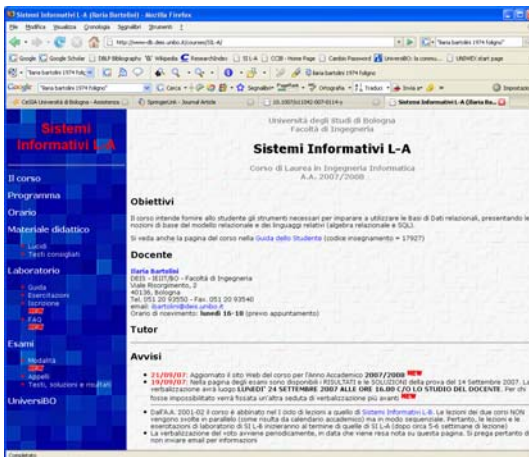
**N.B.** È **obbligatorio** iscriversi mediante la **lista UNIWEX** entro il **02/10/07!!**

## L'Home Page del corso

<http://www-db.deis.unibo.it/courses/SIL-A/>

Contiene:

- Copia dei lucidi in formato PDF
- Guida al lab.
- Testi e soluzioni delle esercitazioni svolte in lab
- Appelli d'esame
- Testi e soluzioni di prove d'esame
- Regole d'esame
- Bibliografia
- **Avvisi**
- ...



Introduzione

Sistemi Informativi L-A

5

## Programma del corso: in aula

- **Lezioni in aula:**
  - Sistemi informativi
  - Sistemi di gestione di basi di dati (DBMS)
  - Il modello relazionale dei dati
  - L'algebra relazionale
  - Il linguaggio SQL (Structured Query Language)
  - L'interfaccia JDBC (Java Data Base Connectivity)
- **Esercitazioni in aula:**
  - Modello relazionale
    - normalizzazione di strutture nidificate
  - L'algebra relazionale
  - Il linguaggio SQL
  - Applicazioni Java basate su JDBC

Introduzione

Sistemi Informativi L-A

6



## Programma del corso: in laboratorio

- **Esercitazioni guidate dal tutor** in laboratorio:
    - Il sistema **IBM DB2 Universal Database (UDB)**
    - Definizione di tabelle in DB2 UDB
    - Scrittura di interrogazioni da **Command Center**
    - Applicazioni Java basate su **JDBC**
  - **N.B. Il Laboratorio è parte integrante del corso!!** Frequentarlo è importante:
    - Alcune problematiche si capiscono meglio mettendo in pratica i concetti teorici appresi a lezione
    - Acquisire manualità nell'uso di strumenti di gestione di basi di dati è fondamentale
- (Per saperne di più sulle regole generali di laboratorio si prega di consultare la **guida** sulla home page del corso)



## Testi consigliati

- Per **sostenere l'esame è sufficiente il materiale reso disponibile sul sito**. Per chiunque voglia comunque avere a disposizione delle fonti in cui gli argomenti trattati a lezione vengono svolti in maniera più estesa sono consigliati i seguenti **testi**:

### Testo consigliato:

- P. Atzeni, S. Ceri, S. Paraboschi, R. Torlone. "**Basi di Dati: modelli e linguaggi di interrogazione**", seconda edizione, McGraw-Hill Italia, 2006 (*il testo è espressamente pensato per un corso di 5-6 crediti del nuovo ordinamento*)

### Altri testi interessanti:

- D. Chamberlin: "**A Complete Guide to DB2 Universal Database**", Morgan Kaufmann
- R. van der Lans: "**Introduzione a SQL**", Addison-Wesley, 2001
- N. Gehani: "**The Database Book: Principles & Practice using MySQL**", Silicon Press, 2006
- P. Ciaccia, D. Maio: "**Lezioni di Basi di Dati**", Esculapio, 2001 (*pur contenendo materiale utile per il corso - modello relazionale, algebra e SQL - ha i suoi punti di forza nel trattamento di aspetti "fisici" della tecnologia dei DBMS*)



## Modalità d'esame (1/2)

- L'esame di Sistemi Informativi L-A prevede una prova "integrata" che viene svolta in laboratorio. Tipicamente comprende:
  - scrittura di una semplice applicazione Java con SQL
  - risoluzione di interrogazioni SQL in modalità interattiva
  - esercizi di normalizzazione e algebra relazionale
  - domanda/e di teoria
- L'ammissione all'esame è subordinata all'iscrizione obbligatoria e vincolante dello studente tramite liste UNIWEX. L'iscrizione deve avvenire entro quattro giorni dalla data dell'esame, pena la mancata ammissione allo stesso



## Modalità d'esame (2/2)

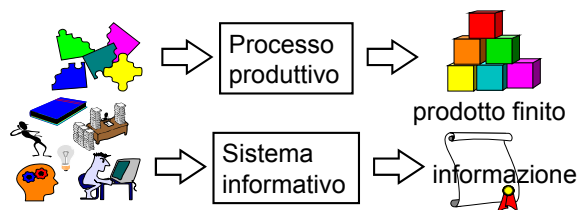
- Durante l'esame non è consentito uscire dalla sala terminali
- La consegna dell'esame avviene in modalità telematica
- Il voto acquisito in una prova ha validità illimitata nel tempo
- Chi si presenta a una prova perde l'eventuale voto precedentemente acquisito ("presentarsi" significa: iscriversi alla prova e ritirare il testo della prova stessa)
- Non c'è alcuna restrizione sul numero di prove che si possono sostenere
- L'orale viene svolto solo su richiesta del docente e in quei casi in cui è in dubbio la sufficienza o la lode, o quando vi sono seri elementi che mettono in dubbio la "paternità" dell'elaborato d'esame
- I risultati delle prove vengono esposti sul sito Web del corso

## Appelli d'esame

Ciclo	Appello	Data
I (24/09/07-07/12/07)	1	14 Dicembre 2007
I	2	11 Gennaio 2008
II (21/01/08-15/03/08)	3	18 Aprile 2008
III (21/04/08-14/06/08)	...	...

## Sistemi Informativi

- Un sistema informativo (SI) è un **componente di una organizzazione** (azienda, ente, ...) il cui scopo è **gestire le informazioni** utili per gli scopi dell'organizzazione stessa  
**GESTIRE = acquisire, elaborare, conservare, produrre, distribuire**
- L'informazione è un bene a valore crescente, necessario per pianificare e controllare con efficacia le attività dell'organizzazione, e rappresenta la materia prima che viene trasformata dai sistemi informativi, così come i semilavorati vengono trasformati dai sistemi di produzione



## Il valore dell'informazione

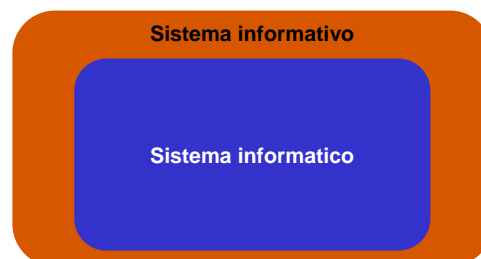
- L'informazione è una risorsa alla stessa stregua del capitale, delle materie prime, degli impianti e delle persone, e come queste ha un costo
- È quindi importante capire il valore effettivo dell'informazione



- Il livello più basso nella scala dei valori lo occupano i "dati grezzi", che possono definirsi informazione solo se interpretati nel giusto contesto  
"Mario", "Rossi" e "12345" sono dati  
Lo studente Mario Rossi ha numero di matricola 12345 è informazione

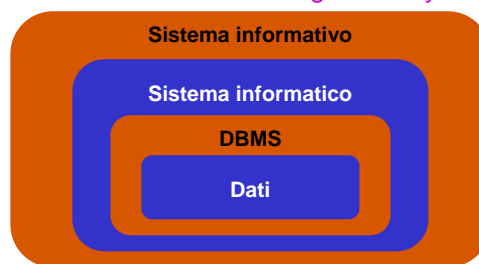
## Sistemi Informativi e Sistemi Informatici

- Un SI gestisce informazioni, ma ciò non significa necessariamente fare ricorso a strumenti automatici propri della tecnologia dell'informazione (IT)  
Banche e servizi anagrafici esistono da secoli!
- La parte automatizzata di un SI viene più propriamente denominata Sistema Informatico



## Dati e Basi di Dati

- Il modo più comune con cui un sistema informatico gestisce le informazioni è attraverso la **representazione codificata dei dati di interesse**
- Intuitivamente, una **Base di Dati (DB - Data Base o Database)** può pensarsi come una collezione di dati che rappresentano le informazioni di interesse per un'organizzazione
- In termini più precisi, un DB è **una collezione di dati gestita da un DBMS**  
**DBMS = Data Base Management System**



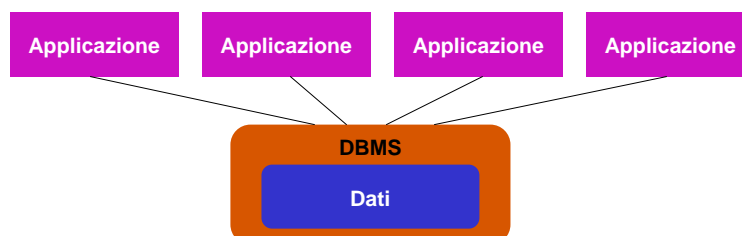
Introduzione

Sistemi Informativi L-A

15

## Dati e applicazioni: il ruolo dei DBMS

- In qualsiasi organizzazione sono molteplici le applicazioni del sistema informatico che utilizzano gli stessi dati (o quasi)  
**Immatricolazione, RegistrazioneEsami, DomandaLaurea, PianidiStudio, ...**  
sono tutte procedure applicative che **condividono** i dati di uno studente
- Un DBMS è un sistema software in grado di **gestire collezioni di dati che sono condivise da più applicazioni e utenti** (e molto altro ancora...)



Introduzione

Sistemi Informativi L-A

16



## DBMS: caratteristiche di base

- Le caratteristiche di un DBMS non si limitano ovviamente alla sola condivisione dei dati e verranno trattate nel seguito con maggior dettaglio
- ... ma è importante avere chiaro da subito che un DBMS:
  - è in grado di gestire **grandi quantità di dati** (Giga-Tera byte e oltre)
  - è in grado di garantirne la **persistenza** (anche a fronte di guasti)
  - offre una “visione strutturata” dei dati che dipende dal **modello (logico) dei dati** supportato

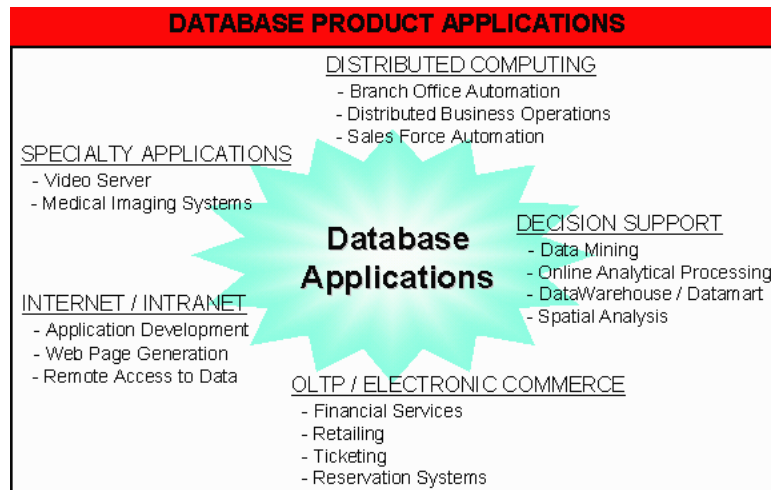
**RDBMS** = DBMS che supporta il **modello relazionale** dei dati

**Modello relazionale**  $\approx$  i dati sono rappresentati in forma tabellare

## Uno sguardo al mercato dei DBMS

- Oggi il mercato dei DBMS (il cui fatturato si aggira intorno a  **$8 \times 10^9$  US\$**) è dominato da un numero ristretto di produttori:
  - Oracle
  - IBM (**DB2 UDB**)
  - Microsoft (**SQL Server**)
  - Sybase
  - Borland
- A questi se ne aggiungono altri (*relativamente maturi*) del mercato “open-source”:
  - **mysql**
  - **PostgreSQL**
  - **DB2 Express-C**
  - **InterBase**
- Il fatturato globale del mercato dei DBMS (di cui l’**80%** è imputabile ai **RDBMS**) cresce di un fattore **> 10%** ogni anno

## Applicazioni dei Data Base



## Uno sguardo al corso

- Ricordiamo che esistono (almeno) 3 “punti di vista” possibili:
- **Utente**, ovvero come usare un DB
  - Richiede la conoscenza del modello dei dati, dei linguaggi supportati dal DBMS e delle modalità con cui un’applicazione può collegarsi a un DB
- **Progettista**, ovvero come progettare un DB
  - È necessario capire come i requisiti informativi di un’organizzazione complessa possono tradursi in strutture concrete
- **Amministratore**, ovvero come amministrare un DB
  - Richiede anche conoscenze su come è fatto un DBMS (in particolare per motivi di efficienza)

## Il Data Base della JLUNGA

Data	Ora	Cassa	Prodotto	Qtà	Importo
20-12-1997	17:53	21	Panettone GnamGnam 1Kg	2	26000
20-12-1997	17:53	21	Spumante Bollicin 1 lt.	5	60000
20-12-1997	18:01	21	Dentifricio WhiteTeeth	1	3400
20-12-1997	18:02	15	Spumante Bollicin 1 lt.	2	24000
21-12-1997	9:06	3	Caffè BlackMoka 250 gr.	1	4100

- Molte operazioni "semplici" di registrazione delle vendite
- Complesse operazioni di analisi delle vendite:
  - Importo complessivo delle vendite
  - Importo vendite per giorno, cassa, fascia oraria, ...
  - Correlazioni di vendita (chi compra il panettone, spesso compra anche lo spumante)

➡ L'obiettivo è permettere alle applicazioni di specificare queste operazioni nel modo più "uniforme" possibile e senza conoscere i dettagli dell'organizzazione fisica dei dati

## Un esempio: le casse il 20/12/1997

- Usando i file

```

openfile(f,`mypath/myfile`);
for i=1 to num_casse
  do somma[i] = 0;
while not eof(f) do
  read(f,rec);
  if rec.Data = `20/12/1997`
  then somma[rec.Cassa] +=
    rec.Importo
enddo;
closefile(f);
for i=1 to num_casse
  do print(i,somma[i]);

```

- Usando DBMS e SQL

```

Select  Cassa,sum(Importo)
        As Totale
From    Incassi
Where   Data = `20/12/1997`
Group by Cassa;

```

Cassa	Totale
21	350265
3	234150
15	415320

Legato a:

- path e file
- formato dei dati
- elaborazione sequenziale

## Un altro esempio: la cassa 21 nel 1997

```
openfile(f,`mypath/myfile`);
for i=1 to 365 * num_prodotti
  do somma[i] = 0;
while not eof(f) do
  read(f,rec);
  if rec.Cassa = 21 and
    year(rec.Data) = 1997 then
    i = converti(rec.Data, rec.Prodotto);
    somma[i] += rec.Importo
enddo;
closefile(f);
for i=1 to 365 * num_prodotti
  do print(i,somma[i]);
```

```
Select Data,Prodotto,sum(Importo)
From Incassi
Where Cassa = 21
And year(Data) = 1997
Group by Data,Prodotto
Order by Data,Prodotto;
```

- sort implicito in "converti"
- memoria male utilizzata
- poco chiaro

## ... ma anche SQL può essere complesso!

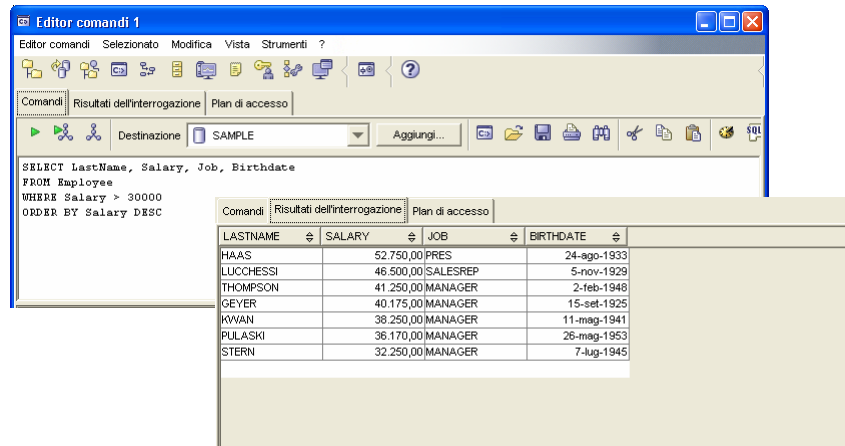
- Ad esempio, se si vuole trovare, per ogni cassa, il prodotto più venduto il 27/09/2001:

```
Create View CP(Cassa,Prodotto,SumQta)
As Select Cassa,Prodotto,sum(Importo)
From Incassi
Where Data = `27/09/2001`
Group by Cassa,Prodotto;

Select Cassa,Prodotto
From CP
Where SumQta = (Select max(CP1.SumQta)
From CP As CP1
Where CP1.Cassa = CP.Cassa);
```

## Come usare SQL (1)

- Le istruzioni SQL possono essere eseguite interattivamente...



The screenshot shows a window titled "Editor comandi 1" with a menu bar (Editor comandi, Selezionato, Modifica, Vista, Strumenti, ?) and a toolbar. Below the toolbar are tabs for "Comandi", "Risultati dell'interrogazione", and "Plan di accesso". The "Comandi" tab is active, showing the following SQL query:

```
SELECT LastName, Salary, Job, Birthdate
FROM Employee
WHERE Salary > 30000
ORDER BY Salary DESC
```

The "Risultati dell'interrogazione" tab is also active, displaying a table with the following data:

LASTNAME	SALARY	JOB	BIRTHDATE
HAAS	52.750,00	PRES	24-ago-1933
LUCCHESI	46.500,00	SALESREP	5-nov-1929
THOMPSON	41.250,00	MANAGER	2-feb-1948
GEYER	40.175,00	MANAGER	15-set-1925
KVWAN	38.250,00	MANAGER	11-mag-1941
PULASKI	36.170,00	MANAGER	26-mag-1953
STERN	32.250,00	MANAGER	7-lug-1945

## Come usare SQL (2)

- ... o inserendole nel codice di un'applicazione scritta in un linguaggio di programmazione "ospite" (ad es. Java)

```
System.out.println("Retrieve some data from the database...");
Statement stmt = con.createStatement();
ResultSet rs = stmt.executeQuery("SELECT * FROM employee");

// display the result set
while (rs.next()) {
    String a = rs.getString(1);
    String str = rs.getString(2);
    System.out.print(" empno= " + a);
    System.out.print(" firstname= " + str);
    System.out.print("\n");
}
rs.close();
stmt.close();
```



## Riassumiamo:

---

- Un **DB** è una collezione di dati (ad es. rappresentati in forma di tabelle) che rappresentano le informazioni di interesse per un'organizzazione
- Progettare un DB significa trovare un "buon modo" per strutturare i dati, che rispetti sia i requisiti propri del **Sistema Informativo** dell'organizzazione sia i **vincoli** che sui dati vengono imposti
- Un **DBMS** è un sistema software che gestisce DB
- Per eseguire operazioni su un DB è necessario scrivere istruzioni in un linguaggio (tipicamente dichiarativo: **SQL**) supportato dal DBMS