

Corso di “Basi di Dati”

Università degli studi di Salerno

Docente: Prof. Giuseppe Polese

Descrizione del Corso:

Il corso intende fornire gli elementi per

- conoscere la struttura e le funzioni di un sistema informativo,
- conoscere le tecnologie, i modelli ed i linguaggi utilizzati per la progettazione, l'implementazione e la gestione delle basi di dati
- saper progettare e produrre basi di dati di buona qualità partendo dai requisiti utente
- saper estrarre informazioni da una base di dati
- saper progettare applicazioni software che necessitano di interagire con una base di dati.

La prima parte del corso fornisce una panoramica sul ruolo delle Basi di Dati nei Sistemi Informativi e sui Sistemi di Gestione di Basi di Dati. Dopo aver introdotto i concetti fondamentali di dato, informazione, database e dbms, si passa allo studio dell'analisi dei requisiti, della modellazione concettuale e logica, mediante opportuni modelli dei dati. In particolare, dopo aver introdotto il modello Entità/Relazioni (ER), vengono studiate le tecniche di ristrutturazione di uno schema concettuale, onde prepararlo alla traduzione verso un modello logico. A tale scopo, dopo una trattazione del modello relazionale e dei suoi vincoli, vengono descritte le regole per la traduzione di uno schema ER ristrutturato in uno schema logico relazionale.

Nella seconda parte del corso si passa allo studio dei linguaggi di definizione e manipolazione di basi di dati relazionali. In particolare, viene affrontato lo studio dell'algebra relazionale e del linguaggio SQL, unitamente alla libreria di funzioni JDBC per l'utilizzo di SQL nei programmi scritti in linguaggio Java. Successivamente, viene trattata l'organizzazione fisica dei dati e le strutture di indicizzazione. Infine, il corso chiude con dei cenni sulla normalizzazione delle basi di dati relazionali.

L'attività di laboratorio è rivolta in una prima fase alla conoscenza di strumenti di supporto alla progettazione delle basi di dati. Successivamente, viene introdotto il DBMS relazionale MySQL, le cui funzionalità vengono sperimentate attraverso la creazione di applicazioni per basi di dati convenzionali in domini di interesse. Infine, vengono svolte alcune fasi del progetto di corso, riguardante la progettazione ed implementazione di un'applicazione di tipo gestionale.

Testi Consigliati:

Rif1) P. Atzeni, S. Ceri, S. Paraboschi, R. Torlone, “ Basi di dati - Modelli e linguaggi di interrogazione”, McGraw-Hill, Quinta Edizione.

Rif2) R.A. Elmasri, S.B. Navathe, “ Sistemi di Basi di Dati – Fondamenti”, 5^a edizione, Pearson - Addison Wesley, 2004.

Prerequisiti:

Fondamenti di informatica, concetti di programmazione iterativa; capacità di scrivere ed eseguire semplici programmi che gestiscono archivi mediante file system.

Metodo di Insegnamento:

1. Lezioni Frontali (60%).
2. Esercitazioni (40%).

Metodo di Valutazione:

La valutazione avverrà tramite un progetto, una prova scritta ed una prova orale. Quest'ultima costituisce la prova finale e vi si accede solo dopo aver superato con sufficienza la prova scritta ed il progetto. Quest'ultimo può essere sviluppato individualmente o in gruppi di 2-3 persone. Solo in presenza di progetti di maggiore complessità il docente può, in via del tutto eccezionale, autorizzare gruppi composti da più persone. Il singolo studente o gruppo può scegliere tra due tipologie di progetti: progetto di corso o extracorso. Il progetto di corso viene sviluppato durante alcune delle ore di esercitazione ed è soggetto a delle scadenze temporali per la consegna dei relativi moduli, tra cui il modulo finale deve essere consegnato poco dopo la fine del corso. Viceversa, il progetto extracorso può essere sviluppato su un dominio applicativo a scelta e può essere completato prima o dopo aver superato la prova scritta, ma comunque, prima della prova orale o, al limite, può essere discusso contestualmente alla stessa.

Per quanto concerne la prova scritta, oltre alla risoluzione di alcuni esercizi legati alla progettazione e realizzazione di una base di dati, viene solitamente chiesto lo sviluppo di query e/o di piccoli programmi Java/JDBC/MySQL.

In alternativa alla prova scritta, gli studenti che seguono il corso con una certa regolarità (valutata attraverso firme di presenza), tenendosi al passo con gli argomenti trattati, possono optare per una valutazione in itinere. Questa consiste nello svolgimento di circa 3 classwork (brevi prove scritte).

Il voto finale dell'esame viene derivato a partire dal voto della prova scritta che, tramite la valutazione del progetto, può essere incrementato di circa 2 trentesimi. Il punteggio così ottenuto può essere incrementato di un ulteriore 20-25% circa con la prova orale. Ad esempio, se il voto della prova scritta è 23/30 e si è fatto un ottimo progetto, si arriva ad un punteggio base di 25/30, potendo poi aspirare al voto di 30/30 (eventualmente con lode) tramite la prova orale. Coloro che optano per la valutazione tramite prove in itinere devono aver sostenuto almeno 2 delle 3 prove previste, riportando la sufficienza ad almeno 2 di esse. Per costoro, in luogo del voto della prova scritta, costituisce voto di partenza la media dei voti riportati sui 2 classwork migliori.

Programma dettagliato del corso

1. Concetti Introduttivi

(Rif.1: Capitolo 1; Rif.2: Capitoli 1 e 2)

In questa parte del corso viene fatta una panoramica generale sul mondo delle basi di dati, includendo definizioni base, modelli, strumenti, linguaggi e figure professionali coinvolte.

Contesti applicativi delle Basi di Dati

Cenni sui Sistemi Informativi

Definizione di Base di Dati

Proprietà delle Basi di Dati

Sistemi di Gestione di Basi di Dati (DMBS):

- Funzionalità dei DBMS
- Principali DBMS
- Basi di Dati e DBMS: Il Sistema di Base di Dati (Database System)

Confronto con approccio File Processing:

- Problemi di duplicazione ed inconsistenza dei dati
- Indipendenza Programma-Dati

Figure professionali coinvolte

Modelli:

- Concetti di Modello ed Astrazione
- Modelli dei dati
- Modelli Concettuali, Logici (o rappresentazionali) e Fisici

Schemi e Istanze di Basi di dati

Architettura a 3 livelli (3-schema) di un Sistema di Base di Dati:

- Livello Esterno, Logico e Fisico
- Mapping tra i livelli
- Indipendenza Logica e Fisica
- Vantaggi e svantaggi dell'architettura a 3 livelli

Classificazione di DBMS:

- Classificazione basata sul modello dei dati supportato
- Classificazione basata sui livelli supportati

Linguaggi di DBMS:

- Linguaggi di definizione dati (DDL)
- Linguaggi di manipolazione dati (DML)
- Linguaggi di definizione di viste (VDL)
- Linguaggi di definizione della struttura fisica dei dati (SDL)
- DML dichiarativi e procedurali

- Embedding di DML in linguaggi di programmazione general-purpose

Interfacce di DBMS:

- Interfacce a Menu
- Interfacce form-based
- Interfacce per DBA

Utility di un DBMS:

- Utility di caricamento
- Utility di monitoraggio prestazioni
- Utility di Backup
- Utility di Organizzazione file

2. Progettazione Concettuale di una base di dati

(Rif.1: Capitoli 6 e 7; Rif.2: Capitolo 3)

Scopo di questa parte è quello di trattare le tecniche di progettazione concettuale di una base di dati. Pertanto, viene fatta una panoramica sulle metodologie di progettazione e sui modelli concettuali disponibili, fornendo una descrizione dettagliata del modello Entità-Relazioni. Successivamente, vengono studiate le tecniche analisi delle prestazioni, nonché le tecniche di ristrutturazione di schemi concettuali ER. Queste ultime fasi nel riferimento bibliografico n. 1 vengono inquadrare nell'ambito della progettazione logica, anche se gli autori di altri libri non sempre convergono su questa visione.

2.1. Metodologie e Modelli per il Progetto

(Rif.1: Capitolo 6; Rif.2: Capitolo 3 e Appendice A)

Cenni sul ciclo di vita di un sistema informativo

Processo di progettazione di una base di dati:

- Analisi dei requisiti
- Progettazione Concettuale
- Progettazione Logica
- Progettazione Fisica

Il Modello Entità-Relazione (ER):

- ER come modello concettuale dei dati
- Costrutti base di ER
 - Entità
 - Entità astratte e concrete
 - Istanze di entità
 - Rappresentazione grafica
 - Attributi
 - Attributi semplici/composti
 - Attributi a valore singolo/ multiplo
 - Attributi chiave di un'entità
 - Domini di attributi
 - Rappresentazione grafica

□Relazioni

- Relazioni ed istanze di relazione
- Grado di una relazione
- Relazioni ricorsive
- Attributi di relazione
- Ruoli delle entità che partecipano ad una relazione
- Vincoli di cardinalità e di partecipazione
- Rappresentazione grafica

- Gerarchie di Generalizzazione

2.2. Il Processo di Progettazione Concettuale

(Rif.1: Capitolo 7; Rif.2: Capitolo 3)

Analisi dei requisiti

Strategie di Progettazione Concettuale

Raffinamento dello schema concettuale

Criteri per la determinazione della qualità di uno Schema Concettuale

Cenni sugli strumenti CASE per la progettazione concettuale

2.3. Ristrutturazione di Schemi ER

(Rif.1: Capitolo 8; Rif.2: Capitolo 9)

Analisi delle prestazioni su schemi ER

Ristrutturazione dello schema ER:

- Analisi delle ridondanze
- Eliminazione delle generalizzazioni
- Partizionamento/accorpamento di concetti
- Scelta degli identificatori principali

3. Il Modello Relazionale

In questa parte viene trattato il modello logico relazionale ed i suoi vincoli, onde permettere di affrontare la traduzione di uno schema ER ristrutturato in uno schema logico relazionale.

(Rif.1: Capitolo 2; Rif.2: Capitolo 7)

Cenni sui modelli logici basi di dati

Relazioni e Tabelle

Attributi (Colonne) di Tabelle

Domini di attributi

Tuple (Righe) di Tabelle

Grado di una Tabella

Cardinalità di una Tabella

Schemi e Istanze di Tabelle

Schemi e Istanze di basi di dati relazionali

Vincoli del modello relazionale:

- Vincoli di dominio
- Vincoli di chiave

- Vincoli di integrità di entità
- Vincoli di integrità referenziale
 - Concetto di chiave esterna
 - Operazioni che possono violare l'integrità referenziale
 - Gestione di tentativi di violazione dell'integrità referenziale

4. Progettazione Logica di una base di dati

(Rif.1: Capitolo 8; Rif.2: Capitoli 7 e 9)

In questa parte vengono descritte le tecniche per la traduzione di uno schema concettuale ristrutturato in uno schema logico relazionale.

Traduzione di Entità e Attributi

Traduzione di relazioni

Traduzione di gerarchie di generalizzazione

Traduzione di Entità deboli

Documentazione di schemi logici

5. Manipolazione di Basi di Dati Relazionali

(Rif.1: Capitoli 3, 4 e 5; Rif.2: Capitoli 7, 8 e 12)

In questa parte vengono trattati i DML per la manipolazione di dati in un database relazionale. Dopo una trattazione del linguaggio teorico fornito dall'algebra relazionale, si passerà allo studio del linguaggio SQL. Infine, si studieranno le varie tecniche per l'utilizzo di SQL nei linguaggi di programmazione.

5.1. Algebra Relazionale

(Rif.1: Capitolo 3; Rif.2: Capitolo 7)

Tabelle come relazioni

Concetto di operatore su relazioni

Operatori unari e binari

Operatori base:

- Selezione
- Proiezione
- Join

Concetto di Union-compatibility

Operatori insiemistici:

- Unione
- Intersezione
- Differenza

Insieme minimale di operatori dell'algebra relazionale

5.2. SQL

(Rif.1: Capitoli 4 e 5; Rif.2: Capitolo 8)

Cenni storici

Caratteristiche di SQL

Definizione dati in SQL:

- Creazione di Schemi
- Creazione di Tabelle
- Tipi di dati e domini
- Valori Null e di default
- Specifica di vincoli di integrità:
 - Il vincolo *Primary Key*
 - Il vincolo *Unique*
 - Il vincolo *Foreign Key (Triggers)*
 - Monitoraggio di eventi *Delete* e *Update*
 - Opzioni *Cascade*, *Set Null* e *Set Default*

Eliminazione di Schemi e Tabelle:

- Comandi *Drop Schema* e *Drop Table*
- Uso di opzioni *Restrict* e *Cascade*

Modifica di Schemi di Tabella:

- Aggiunta o rimozione di un attributo
- Ridefinizione di una colonna
- Aggiunta o rimozione di un vincolo

Manipolazione dati in SQL:

- Interrogazioni
 - Il comando *SELECT*
 - Uso di *SELECT* per Selezioni, Proiezioni e Join
 - Operatori aggregati
 - Operatori insiemistici
 - Interrogazioni nidificate
 - Ordinamento di Tuple
 - Clausole *Group by* e *Having*
- Aggiornamenti
 - Inserimento dati
 - Cancellazione dati
 - Modifica dati

Creazione e manipolazione di Viste (Tabelle Virtuali)

Creazione di Indici in SQL

Transazioni in SQL:

- Concetto di Transazione
- Proprietà ACID
- Transazioni in SQL
 - *Begin Transaction*
 - *End Transaction*
 - *Commit work*
 - *Rollback work*

6. Uso di SQL in linguaggi di programmazione general-purpose

(Rif.1: Capitolo 10; Rif.2: Sezioni 8.7 e 12.5)

Concetto di SQL Embedded

Cursori

SQL dinamico

ODBC e JDBC

Procedure SQL

7. Cenni sulla Normalizzazione di Schemi di Basi di Dati Relazionali

(Rif.1: Capitolo 9; Rif.2: Capitolo 10)

In questa parte vengono fornite linee guida per la valutazione degli schemi relazionali progettati. Inoltre, vengono forniti dei cenni sulla normalizzazione di schemi logici relazionali, illustrando vantaggi e limiti di questo approccio.

7.1. Analisi di schemi logici e dipendenze funzionali

(Rif.1: Capitolo 9; Rif.2: Capitolo 10 fino a sezione 10.2 inclusa)

Ridondanze negli schemi

Anomalie di inserimento, modifica e cancellazione

Dipendenze funzionali

7.2. Forme Normali

(Rif.1: Capitolo 9; Rif.2: Capitolo 10 sezioni 10.4 e 10.5)

Prima forma normale

Seconda forma Normale

Terza forma Normale

Definizione generale di seconda e terza forma normale

Confronto tra definizioni alternative di seconda e terza forma normale

Forma normale di Boyce-Codd

8. Organizzazione Fisica nei DBMS Relazionali

In questa parte viene trattata l'organizzazione fisica delle strutture che compongono uno schema di database relazionale, incluso gli indici per migliorare le prestazioni di accesso ai dati.

(Rif.1: Capitolo 11; Rif.2: Capitoli 13 e 14)

Caratteristiche della memoria secondaria

Organizzazione fisica delle relazioni

Gestione delle interrogazioni

Indici relazionali

9. Laboratorio

Le esercitazioni riguarderanno la progettazione di un intero database, attraverso le fasi di analisi dei requisiti, progettazione concettuale tramite lo strumento JDER o simili, progettazione logica ed implementazione con l'ausilio del DBMS Mysql. Successivamente, si passerà allo sviluppo di applicazioni Java che interagiscono con una base di dati.