

Informatica per Statistica

Riassunto della lezione del 14/12/2011

Igor Melatti

Introduzione all'implementazione di basi di dati e all'SQL

- Una volta che la progettazione logica è terminata (e si ha quindi a disposizione uno schema di basi di dati annotato), occorre usare un DBMS per realizzare la base di dati stessa
- Qui si suppone di usare un DBMS semplice come quello contenuto in LibreOffice (<http://www.libreoffice.org/download/>)
 - versione open-source di Access, cui somiglia molto
 - le istruzioni in questo documento possono essere facilmente usate anche per Access
- Per prima cosa, occorre creare le tabelle listate nello schema di base di dati di partenza
- Dopodiché, si scrive l'SQL per le interrogazioni
 - linguaggio standard con cui si scrivono interrogazioni per una base di dati
 - ad esempio, “listare tutte le persone che abbiano almeno 25 anni, mostrando nome e cognome” diventa qualcosa del tipo `SELECT Nome, Cognome FROM Persona WHERE Eta >= 25`
 - in SQL ci sono due tipi di interrogazioni: DDL (*Data Definition Language*) e DML (*Data Manipulation Language*)
 - il DDL serve per creare tabelle o modificare lo schema di tabelle già create
 - il DML serve per:
 1. cercare dati di interesse (interrogazioni vere e proprie)
 2. modificare l'istanza di base di dati (aggiungendone di nuovi, cancellandone di vecchi o modificandone di esistenti)

- Le tabelle si possono creare in 2 modi: usando l'SQL o l'interfaccia grafica
 - con l'SQL si userebbe la parte DDL
 - questa parte non è attualmente supportata in LibreOffice, mentre lo è in Access
 - con l'interfaccia grafica è semplice: si supponga di voler creare una tabella T
 - si va sulla sezione “Tabelle” o “Tables”
 - * a seconda della lingua usata nell'installazione
 - si seleziona la voce che permette di creare tabelle
 - * tipicamente qualcosa come “Create Table in Design View”
 - compare una tabella in cui immettere nella prima colonna il nome delle colonne di T , mentre il tipo (dominio) delle colonne di T va nella seconda
 - i domini sono a scelta tra quelli predefiniti, ovvero
 - * stringa, ovvero successione di caratteri; occorre sempre dire qual è la lunghezza massima consentita (quanti caratteri ci saranno al massimo in una stringa)
 - * numero esatto: intero o frazionario in virgola fissa (in questo caso, si può precisare quante cifre dopo la virgola)
 - * numero approssimato: in virgola mobile, come i float e i double del C
 - * tempo: date e affini
 - occorre specificare quali sono gli attributi chiave: li si seleziona e li si dichiara primary key
 - è possibile specificare i vincoli di integrità referenziale
 - * va specificata tramite una funzionalità “Relazioni” o “Relationships”, che si può scegliere dal menù a tendina superiore (tipicamente è dentro la voce “Strumenti” o “Tools”)
 - * si scelgono le due tabelle tra le quali esiste il vincolo da inserire
 - * si trascina il campo sul quale il vincolo è definito sopra quello dal quale deve prendere i valori
 - * graficamente, verrà mostrato da una linea con n e 1 ai suoi estremi
- Per creare le interrogazioni, si usa l'SQL
 - esistono meccanismi che “aiutano” a scrivere l'SQL per una interrogazione, più o meno efficaci; qui verranno ignorati

- per gli scopi di questo corso, una tipica interrogazione SQL per la ricerca dei dati è scritta in questa maniera:

```
SELECT <ListaNomiColonne> FROM <NomeTabella> WHERE  
<Condizione>;
```

- * dove <NomeTabella> è ovviamente il nome di una tabella presente nella base di dati
 - * dove <ListaNomiColonne> è una lista di nomi di colonne (separati da virgole) appartenenti alla tabella usata nel FROM
 - * se non ci fosse il WHERE (il che è permesso dall'SQL), allora l'effetto sarebbe di stampare tutte le righe della tabella <NomeTabella>, ma per ognuna di queste righe si stamperebbero solo i dati nelle colonne elencate in <ListaNomiColonne>
 - * se l'interrogazione richiede di stampare i dati di tutte le colonne, allora o si scrivono tutte le colonne in <ListaNomiColonne> oppure si mette semplicemente *
 - * ad esempio, `SELECT * FROM Persona;` stampa i dati di tutte le colonne di tutte le righe della tabella Persona
 - * se non si vogliono tutte le righe, ma solo quelle che soddisfano una qualche condizione booleana su(i dati contenuti ne)lle colonne, allora si usa la *clausola WHERE*
 - * ad esempio, se si vogliono tutti i dati di tutti gli impiegati che prendono tra 1000 e 1200 euro come stipendio mensile, e tale informazione è contenuta nell'attributo `StipendioMensile`, allora la query corretta è `SELECT * FROM Impiegato WHERE StipendioMensile >= 1000 AND StipendioMensile <= 1200;`
 - * le condizioni contenute nella clausola WHERE possono contenere confronti che coinvolgano sia nomi di colonne che costanti, combinati tra loro con i connettivi logici AND, OR e NOT
 - ovviamente, le costanti devono essere dello stesso dominio delle colonne
 - i confronti disponibili, con il rispettivo simbolo, sono: uguale =, diverso <>, maggiore >, minore <, maggiore uguale >=, minore uguale <=
- invece, per modificare l'istanza di base di dati (inserzione/cancellazione di intere righe, o modifica di dati già immessi) si suppone qui di usare gli strumenti grafici
 - * facendo doppio clic su una tabella viene mostrato il suo contenuto
 - * è sufficiente agire su questa visualizzazione per modificare l'istanza di base di dati