

Dispensa di database *Access*

Indice:

Database come tabelle; fogli di lavoro e tabelle	2
Database con più tabelle; relazioni tra tabelle	2
Motore di database, complessità di un database; concetto di tabella, query, linguaggio SQL.....	3
Tabelle, relazioni, query e insieme dinamico di risultati (dynaset) associato.....	3
Schede: inserimento di dati, ricerca e filtro	3
Report: presentazione dei dati e stampa.....	4
Macro e moduli: la programmazione in VBA; programmazione nelle schede e nei report; gli oggetti di access per la programmazione.....	4
Query: query di selezione e query di azione (viste e procedure).....	4
Linguaggio SQL.....	5
Strumenti fondamentali: connessioni e tabelle collegate. importazione ed esportazione di tabelle. compattazione.	5

Scopo e argomenti del manuale

Questo manuale, con il corso ad esso associato, ha come scopo quello di fornire delle basi di comprensione di un database prendendo access 97 come esempio per la costruzione e l'utilizzo di un database.



Cooperativa ALEKOS 20155 MILANO - V. Plana, 49 - P.IVA 11027820155

Tel 02 - 39264592 - Fax 02 - 700506084 info@alekos.net - www.alekos.net

Database come tabelle; fogli di lavoro e tabelle

Un database è essenzialmente un insieme di tabelle. La tabella è un insieme ordinato di dati e differisce da un foglio di calcolo (le tabelle riga-colonna di excel) per il fatto di avere i campi (le colonne) strutturati, ovvero il tipo di dati che possono essere inseriti in un campo è definito dal campo (per esempio non si può inserire una stringa se il campo è di interi), mentre in un foglio di calcolo ogni cella è libera. Questo comporta una diversa strutturazione dei dati e dei modi per accedervi (le colonne si chiamano campi e le righe record ed è un insieme di dati coerente) tale da rendere complesso un database. Si utilizza un database quando la mole di dati diventa significativa oppure quando i dati vengono divisi in più tabelle che sono in relazione tra loro. I database relazionali possono infatti legare più tabelle attraverso relazioni in modo molto efficiente.

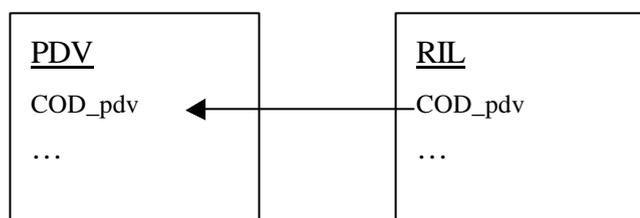
Database con più tabelle; relazioni tra tabelle

Un tipico esempio di relazione tra tabelle è la relazione a due anagrafica-inserimento temporale, come ad esempio i punti vendita e le rilevazioni fatte nei punti vendita.

La tabella dei punti vendita (PDV) ha un record per ogni punto vendita e viene aggiornata saltuariamente, mentre la tabella rilevamenti (RIL) ha un record per ogni rilevazione fatta e quindi viene incrementata continuamente.

Ipotizziamo un rilevamento di fatturato settimanale e dei voti sulla gestione, i campi delle tabelle possono essere:
 PDV: COD_pdv (intero), Ragione sociale (50 caratteri), Indirizzo (50 caratteri), Località (50 caratteri), Telefono (50 caratteri)

RIL: Data, COD_pdv (intero), fatturato_sett (intero doppia precisione), voto (intero)



La relazione avviene tramite il campo COD_pdv, identificativo (cioè non duplicabile) della tabella PDV. La relazione si dice uno a molti (uno su PDV, molti su RIL) quando il codice è duplicato in una tabella ed univoco nell'altra.

Motore di database, complessità di un database; concetto di tabella, query, linguaggio SQL

Mentre il motore di database è il programma che gestisce le richieste (query) al database, fatte secondo un linguaggio standardizzato (SQL, Structured Query Language), il programma che visualizza i risultati, che consente di creare le query, che crea maschere, report, e interfacce verso l'utente, sono programmi client database o ambienti di programmazione database. Posso quindi immaginarmi 5 livelli:

- I. il livello che contiene le tabelle, ovvero il formato dei file di dati e l'organizzazione logica di un database
- II. il livello del motore di database, che ha come obiettivo l'ottimizzazione degli accessi alle tabelle
- III. il livello di interfaccia utente al database che aiuta sia la gestione del database (creazione e modifica di tabelle e query)
- IV. il livello di ambiente di sviluppo per la costruzione del programma di utilizzo del database (creazione e modifica di maschere e report)
- V. il livello dell'utente, ovvero il programma per l'utente finale, come ad esempio il funzionamento delle maschere

Spesso un programma di database contiene più di uno di questi livelli, nel caso di access è possibile unire in un solo file (.mdb) tutti questi livelli tranne il motore database (livello II). Esso è infatti contenuto nella 'runtime' del programma access e può essere unito al file database creando un eseguibile proprio come farebbe un interprete come visual basic. Un file mdb (chiamiamolo così) contiene i dati (le tabelle), l'interfaccia al database (consente di creare tabelle e query), un ambiente di programmazione facilitato in linguaggio vb, il VBA (visual basic for application) che consente di creare maschere, report e codice di programmazione associato, un programma vero e proprio di livello utente per scrivere i dati. Tutto ciò genera confusione ma è comodo per iniziare ad affrontare i database oltre ad essere comodo per la trasportabilità di un database.

Il motore di database deve essere efficiente per cui può essere complicato al suo interno, ma l'utilizzo di un database potrebbe non essere influenzato da questo, spesso però la gestione di un database (livello I e III) avviene all'interno del programma e quindi ne è legato, così come la connessione al database da parte di un programma può avere qualche particolarità.

Tabelle, relazioni, query e insieme dinamico di risultati (dynaset) associato

La complessità di una database specifico, cioè di un insieme di dati, è data invece dalle relazioni tra tabelle. Può aiutare un diagramma nel quale per ogni tabella vengono visualizzati i campi con il tipo, le chiavi, le relazioni con la loro tipologia (uno-molti,...). In access la visualizzazione grafica può essere ottenuta con la creazione query.

Il dynaset è un insieme dinamico di record contenenti i dati estratti da una o più tabelle. Ad ogni query di selezione viene associato un dynaset i cui dati sono modificabili nel caso che la query sia modificabile.

Schede: inserimento di dati, ricerca e filtro

Le schede (o maschere) permettono di visualizzare i dati in una finestra per essere modificati. La visualizzazione può essere gabbellare, cioè presentare tutti i record per ogni riga, oppure a scheda, cioè una pagina per ogni record.

All'interno della scheda, così come in un form di visual basic, viene scritto il codice di programma associato ai controlli e alla maschera.

La maschera si disegna e si imposta in visualizzazione struttura, mentre la visualizzazione maschera, foglio dati e anteprima di stampa visualizzano i record, quindi in modalità utente.

Per cercare all'interno di un database in access, ci sono due strumenti standardizzati.

La ricerca: trova una parola in un campo scorrendo tutti i record

Il filtro: permette di restringere i record mostrati a quelli che soddisfano i criteri sui campi. I criteri sono simili al WHERE dell'istruzione SQL, come ad esempio:

indirizzo='milano'

*indirizzo like '*lan*'*

cod_pdv>5

cod_pdv between 3 and 5

indirizzo is not null

Report: presentazione dei dati e stampa

Per stampare un insieme di record (forma tabellare) o un record solo su un foglio (ad esempio una fattura), sono molto comodi i report che presentano i dati non modificabili come in una anteprima di stampa. Come per le maschere c'è il codice associato, ma i report non si usano per inserire e modificare i dati.

Macro e moduli: la programmazione in VBA; programmazione nelle schede e nei report; gli oggetti di access per la programmazione

Oltre al codice di programmazione associato ai controlli delle maschere e dei report, ci possono anche essere parti di programmazione comuni a tutto il database (moduli), così come semplici azioni (macro) utilizzate a livello di programma utente.

Le macro sono sostituibili dal codice e quindi possono essere convertite in moduli.

La programmazione utilizza il linguaggio VBA (Visual Basic for Application) per cui tutte le istruzioni, le parole chiave e la modalità di programmazione sono uguali al visual basic.

I moduli quindi si utilizzano soprattutto per le variabili globali, per le istruzioni non reattive solo ad una maschera e per le procedure ripetitive.

Query: query di selezione e query di azione (viste e procedure)

La tabella è un insieme di dati relativi ad uno stesso argomento, organizzato per campi e record; la query è invece una domanda di estrazione dei dati contenuti in una o più tabelle secondo un linguaggio codificato SQL, valido (più o meno) per tutti i database.

Le query non servono solo per visualizzare un insieme di risultati ma anche per modificare i dati: mentre le query di selezione (o anche la query a campi incrociati) vengono chiamate **viste** e non toccano i dati (una volta visualizzati i risultati si può modificare i record direttamente, la modifica viene attuata alla tabella o alle tabelle sottostanti. Le query di azione, o **procedure**, modificano direttamente i dati: query di creazione tabella, query di aggiornamento, query di accodamento, query di eliminazione.

Linguaggio SQL

Tutte le query create in access si possono anche scrivere e vedere nel formato SQL, il linguaggio standard per gestire i dati in un database. In questo modo è possibile avvicinarsi alla sintassi sql a partire dalla semplice creazione grafica di query di selezione o di procedure.

La più semplice istruzione SQL è composta da:

```
SELECT <campi>
FROM <tabelle>
WHERE <condizioni filtro>
[GROUP BY <campi di raggruppamento>]
ORDER BY <ordinamento>
SELECT * FROM indirizzi WHERE cod_pdv<12 ORDER BY citta
```

Strumenti fondamentali: connessioni e tabelle collegate. importazione ed esportazione di tabelle. compattazione.

In access si può accedere ai dati esterni sia importandoli in una tabella di access (i dati vengono copiati), sia collegando una tabella access alla risorsa esterna: in questo modo i dati modificati verranno modificati anche nella risorsa esterna. Si può in questo modo utilizzare access come ambiente di sviluppo e come programma utente (maschere, report) mentre i dati risiedono in un server database e si accede ad esso ad esempio tramite ODBC.

L'esportazione avviene esportando una tabella alla volta in una risorsa esterna, ad esempio ancora ODBC.

La compattazione del database access serve a liberare spazio, vengono cancellati in maniera definitiva i record e dati cancellati, vengono anche riposizionati al minimo i contatori.