

**Corso di Laurea in Scienze Turistiche
Informatica per il turismo
A.A. 2014/2015**

Comprensione e creazione di relazioni

«molti a molti»

«uno a molti»

«uno ad uno»

nei DBMS

DBMS E RELAZIONI

Un database relazionale è composto da dati e informazioni organizzati in diverse tabelle.

Una tabella definisce un singolo gruppo di informazioni specifiche, come la tabella dei clienti di un'azienda o la tabella dei prodotti venduti dall'azienda.

Ogni record all'interno della tabella deve comprendere un "campo chiave".

Un campo chiave è un campo o una combinazione di campi che identificano in modo univoco un record (riga) in una tabella.

Un campo chiave può essere di tipo "chiave primaria" o "chiave esterna".

- ✓ Una "chiave primaria" è un campo chiave che si trova nella stessa tabella del record che identifica.
- ✓ Una "chiave esterna" è un campo chiave che identifica un record in un'altra tabella.

DBMS E RELAZIONI

Relazioni "uno a molti"

Le relazioni più comunemente utilizzate nella creazione di database relazionali. Una riga di una tabella di un database può essere associata a una o (verosimilmente) a più righe di un'altra tabella. Un esempio di relazione "uno a molti" è un ordine con molte voci al suo interno. Inoltre, poiché le relazioni funzionano in entrambi i sensi, non è raro sentir parlare anche di relazioni molti a uno.

Relazioni uno a uno

Una riga di una tabella è associata ad una e una sola riga di un'altra tabella. Un esempio di relazione "uno a uno" è il seguente: una persona può avere un solo codice fiscale e ciascun codice fiscale può essere assegnato a una sola persona. Nella maggior parte dei casi non è necessaria una relazione "uno a uno" poiché i contenuti delle due tabelle possono essere combinati in un'unica tabella.

Relazioni "molti a molti"

Una o più righe di una tabella sono associate a una o più righe di un'altra tabella. Un esempio di relazione "molti a molti" è il caso di una tabella di clienti che possono acquistare più prodotti e lo stesso prodotto può essere acquistato da più clienti.

DBMS E RELAZIONI

Una chiave primaria è un campo di una tabella A (che chiameremo tabella primaria) che contiene valori non ripetuti che identificano in maniera univoca ciascun record.

Una chiave esterna è un campo di una tabella B (che chiameremo tabella secondaria) che ha lo stesso tipo di dati della chiave primaria e che viene utilizzato per legare i dati della tabella B a quelli della tabella A.

I clienti di una certa azienda possono essere associati, per esempio, agli ordini che hanno effettuato creando una relazione tra la tabella Clienti e la tabella Ordini tramite il campo *ID Cliente*.

ID Cliente = chiave primaria

CLIENTI		
<u>ID Cliente</u>	Cognome	Nome
1	Rossi	Mario
2	Verdi	Luigi

il campo ID Cliente appare in entrambe le tabelle

TABELLA: ORDINI		
<u>ID Ord.</u>	<u>ID Cliente</u>	<u>Data ord.</u>
10058	2	22/05/97
10059	2	15/06/97

ID Cliente = chiave esterna

DBMS E RELAZIONI

LA RELAZIONE UNO A MOLTI

è il tipo di relazione più frequente in un Data Base relazionale: in questo tipo di relazione ad un record della tabella A possono corrispondere più record della tabella B, mentre ad un record della tabella B corrisponde un solo record della tabella A.

Un esempio di relazione uno a molti è quello visto sopra tra clienti e ordini. Un altro caso di relazione di questo tipo può essere quella esistente tra fornitori e articoli di un'azienda commerciale, nell'ipotesi che ogni prodotto venga fornito da un unico fornitore e che ogni fornitore possa fornire più prodotti.

FORNITORI

ID Forn.	Nome fornitore	Indirizzo
1	Rossi s.r.l.	Via Stella, 21 - Palermo
2	Bianchi s.p.a.	Via Etnea, 18 - Catania
3	Verdi s.r.l.	Via Roma, 42 - Palermo

.....un fornitore.....può vendere più prodotti...

PRODOTTI			
ID Prod.	Nome prodotto	Categoria	ID Forn.
1	IBM 200	Computer	1
2	Laser SWX Lux	Stampante	1
3	Windows Kit	Software	2

...ma ciascun prodotto ha un solo fornitore

DBMS E RELAZIONI

LA RELAZIONE UNO A UNO

In questi tipi di relazioni, ad un record della tabella A può corrispondere un solo record della tabella B, e viceversa.

Si tratta di relazioni poco usate nella pratica, in quanto nella maggior parte dei casi, le informazioni delle due tabelle possono essere contenute in un'unica tabella.

Si possono tuttavia verificare situazioni in cui una relazione uno a uno si rivela utile.

Si immagini, per esempio, che la Divisione Personale di un'azienda voglia distinguere tra dati pubblici dei dipendenti e dati privati.

Si potrebbe a tale scopo porre i dati pubblici in una tabella alla quale possono accedere tutti, mentre quelli privati in un'altra tabella ad accesso controllato; infine legare le due tabelle con una relazione uno a uno definendo in entrambe le tabelle la stessa chiave primaria.

DBMS E RELAZIONI

LA RELAZIONE MOLTI A MOLTI

ad un record della tabella A possono corrispondere più record della tabella B e, viceversa, ad un record della tabella B possono corrispondere più record della tabella A.

Questo tipo di relazione è possibile solo definendo una terza tabella, chiamata tabella di congiunzione, che dispone di due chiavi esterne, una che punta alla chiave primaria della tabella A, l'altra che punta alla chiave primaria della tabella B.

Una relazione **molti a molti** è quindi in realtà composta da due relazioni uno a molti con una terza tabella. Tale terza tabella può anche essere composta da due soli campi (le due chiavi esterne), se il suo scopo è solo quello di legare le due tabelle primarie. Il più delle volte si cercherà comunque di renderla più significativa, aggiungendo altri campi con informazioni pertinenti.

La tabella Ordini e la tabella Prodotti, ad esempio, hanno una relazione molti a molti definita creando due relazioni uno a molti con la tabella Dettaglio ordini.

DBMS E RELAZIONI

LA RELAZIONE MOLTI A MOLTI ,UN ESEMPIO:

Si tratterà un tipico scenario molti a molti: Studenti e Corsi.

- ✓ Avremo una tabella Studenti, contenente un record per ogni studente, e una tabella Corsi, contenente un record per ogni corso disponibile.
- ✓ Il campo chiave primaria IDStudente identifica in modo univoco uno studente della tabella degli studenti.
- ✓ Analogamente, il campo chiave primaria IDCorso identifica in modo univoco un corso della tabella dei corsi.
- ✓ Uno studente può seguire più corsi e un corso può avere più studenti.
- ✓ Correlare queste due tabelle sarebbe difficile senza ricorrere a una tabella associativa. La tabella associativa conterrà le chiavi primarie dalla tabella degli studenti e di quella dei corsi.
- ✓ Alla tabella associativa può essere attribuito il nome che si desidera. Per questo esempio, chiameremo la tabella associativa "Iscrizioni."

DBMS E RELAZIONI

LA RELAZIONE MOLTI A MOLTI ,UN ESEMPIO:

- 1) Creare una nuova tabella in Access e chiamarla "Iscrizioni"
- 2) Nella tabella "Iscrizioni", creare un campo chiave primaria chiamato "IDIscrizioni." Questa operazione non è necessaria per il corretto funzionamento della tabella associativa ma, nei database relazionali, è buona norma che ogni tabella abbia una chiave primaria.
- 3) Nella tabella "Iscrizioni" creare almeno un campo "IDStudente" e un campo "IDCorso". Poiché questi campi verranno popolati mano a mano che verranno creati record nelle tabelle "Studenti" o "Corsi", non è necessario impostare opzioni di campi ad inserimento automatico.
- 4) Se necessario, è possibile aggiungere altri campi alla tabella "Iscrizioni". Alcuni esempi di altri campi che potrebbe essere opportuno inserire nella tabella "Iscrizioni" sono un campo "Data", per tenere traccia del giorno in cui una persona ha iniziato un corso, o un campo "Costo" per tenere traccia della somma pagata per partecipare a un corso. Le tabelle associative solitamente contengono campi che potrebbe non avere senso inserire in nessuna delle altre tabelle.
- 5) Correlare le tabelle in base alle chiavi primarie ed esterne.

DBMS E RELAZIONI

LA RELAZIONE MOLTI A MOLTI ,UN ESEMPIO:

Se uno studente si è iscritto a tre corsi, ha un record nella tabella Studenti e tre record nella tabella "Iscrizioni", ovvero uno per ogni corso a cui si è iscritto.

Una conseguenza dell'uso di una tabella associativa è la possibilità di accedere ai campi e ai dati contenuti nelle tabelle senza dover creare una relazione distinta.

Nel nostro esempio, se necessario, è molto facile visualizzare le informazioni della tabella Studenti nella tabella Corsi e viceversa.



IMPOSTARE/MODIFICARE LE RELAZIONI

1. Innanzitutto non è possibile creare o modificare relazioni tra tabelle aperte, pertanto il primo passo è chiudere le tabelle aperte.
2. Quindi, fare clic sul pulsante **Relazioni** sulla barra degli strumenti nella finestra del Data Base: se è la prima volta che si definiscono o apportano modifiche alle relazioni del Data Base, la finestra appena aperta è vuota, e appare la finestra di dialogo **Aggiungi tabelle o query**, dalla quale selezionare le tabelle per le quali si vuole creare una relazione, quindi scegliere **Aggiungi**.
3. A questo punto trascinare il campo o i campi che si desidera correlare da una tabella (la *tabella primaria*) all'altra (la *tabella secondaria*). Normalmente il campo chiave primaria, visualizzato in grassetto, viene trascinato su un campo che deve condividere con il campo chiave primaria lo stesso tipo di dati e lo stesso genere di informazioni, anche se non necessariamente lo stesso nome.
4. Una volta visualizzata la finestra di dialogo **Relazioni**, accertarsi che si siano correlati i campi che si desiderava correlare, altrimenti modificarli.
5. Fare clic su **Crea**.

IMPOSTARE/MODIFICARE LE RELAZIONI

Nella finestra di dialogo **Relazioni** è inoltre possibile imporre la cosiddetta

Integrità referenziale

Con questo termine si tratta di un sistema di regole che assicura che le relazioni appena impostate si mantengano valide e che non si eliminino o si modifichino, per errore, i dati correlati.

Per impostare l'integrità referenziale è necessario rispettare le seguenti condizioni.

IMPOSTARE/MODIFICARE LE RELAZIONI

- Il campo corrispondente della tabella primaria è una chiave primaria o nel caso non lo fosse, almeno disponga di un indice univoco;
- I campi correlati contengano lo stesso tipo di dati con due possibili varianti, un campo *Contatore* può essere correlato ad un campo *Numerico* la cui proprietà *Dimensione campo* sia impostata a *Intero lungo* e un campo *Contatore* la cui proprietà *Dimensione campo* sia impostata a *ID replica* può essere correlato a un campo *Numerico* la cui proprietà *Dimensione campo* sia a sua volta impostata su *ID replica*;
- Nel caso in cui si trattasse di tabelle esterne al Data Base (fuori dal progetto) ma comunque accessibili, per impostare l'integrità referenziale è necessario che le tabelle siano state create con Access e che il Data Base in cui sono memorizzate sia aperto; non è possibile applicare l'integrità referenziale a tabelle correlate di Data Base in altri formati.

IMPOSTARE/MODIFICARE LE RELAZIONI

Se si seleziona la casella di controllo [Applica integrità referenziale](#) quando si crea una relazione, occorrerà in seguito rispettare alcune regole nell'immissione o nella eliminazione di dati.

1. Non si può immettere un valore nel campo chiave esterna se quest'ultimo non esiste anche nel campo chiave primaria: ad esempio, quando si aggiunge un prodotto alla tabella *Prodotti* è necessario che la chiave esterna ID fornitore esista (come chiave primaria, naturalmente) nella tabella *Fornitori*.
2. Inoltre, non si può cancellare un record da una tabella primaria (il record relativo ad un *fornitore*, per tornare all'esempio precedente), se esistono record corrispondenti in una tabella correlata (cioè *prodotti* venduti da quel fornitore).
3. Infine, non è possibile modificare un valore chiave primaria se il record corrispondente dispone di record correlati.

IMPOSTARE/MODIFICARE LE RELAZIONI

Attivare quindi la casella *Applica Integrità referenziale* della maschera di Fig. 5 significa rendere valide queste regole.

Ogni forzatura che si cercherà di fare verrà in questo caso evidenziata da un messaggio di avvertimento. Tuttavia Access, permette, sempre comunque mantenendo l'integrità referenziale, anche ignorando alcune regole, in particolare quelle sulla restrizione sulla modifica e eliminazione dei record correlati. Queste operazioni vengono compiute rispettivamente attivando le opzioni :

- ✓ *Aggiorna campi correlati a catena*
- ✓ *Elimina record correlati a catena*



Wisdom will give you morals
Knowledge will give you truth
Truth will give you freedom
Free knowledge will give you wisdom

**La saggezza vi darà la morale,
La conoscenza vi darà la verità,
La verità vi darà libertà,
La conoscenza libera vi darà la saggezza**

....

**GRAZIE
DELL'ATTENZIONE**