

CREAZIONE DI UN DB RELAZIONALE IN ACCESS

CONTENUTI:

- **Definizione della chiave primaria di una tabella**
- **Creazione di relazioni fra tabelle**
- **Uso del generatore automatico di Query**

A) Apertura del DB e creazione di una nuova tabella

Riapriamo con ACCESS il DB Biblioteca creato precedente. Aggiungiamo una nuova tabella di nome “Prestiti” a quelle già precedentemente create (libri e utenti). I campi della tabella “Prestiti” con i relativi tipi sono: utente (testo), libro (testo), data_prestito (data/ora), scadenza (data/ora), data_consegna (data/ora), consegnato (si/no). I campi di tipo data/ora devono avere il formato “data in cifre” e il campo “consegnato” ha come valore predefinito No. Salvare la struttura della tabella senza definire, per il momento, nessuna Chiave Primaria.

B) Chiave primaria

Una **chiave primaria** in una tabella è un campo il cui valore identifica in modo univoco un record nella tabella stessa. Detto così sembra molto complicato, ma non lo è.

1) Tabella “utenti”

Prendiamo ad esempio la tabella “utenti”. Il problema è quello di assicurarsi che ogni utente sia distinguibile da ogni altro in ogni caso possibile. In altre parole: vogliamo che ogni record (riga) nella tabella differisca da tutti gli altri per il valore di almeno un campo (colonna). Per esempio affidarsi al “cognome” dell'utente come chiave primaria non è una buona idea, perché potrebbero esistere più utenti con lo stesso cognome. Anche definendo la coppia “nome” e “cognome” come chiave primaria, non avremmo la certezza di poter distinguere ogni utente da tutti gli altri (quanti “Mario Rossi” ci sono?).

In realtà, riflettendo un attimo, l'unico campo che da solo permette di distinguere con assoluta certezza un utente da tutti gli altri è il “*codice fiscale*”. Questo campo è pertanto un'ottima scelta come chiave primaria della tabella “utenti”. Per promuovere il campo “codice fiscale” a chiave primaria, dobbiamo aprire la tabella “utenti” in modalità “struttura” e selezionare il campo. Facendo clic con il tasto destro del mouse, scegliamo l'opzione “chiave primaria”. Il simbolo di una chiave comparirà accanto al “*codice fiscale*” dell'utente.

 codicefiscale	Testo
---	-------

2) Tabella “libri”

Seguendo lo stesso procedimento precedente possiamo facilmente individuare una “chiave primaria” anche per la tabella “libri”: in questo caso si tratta della “*collocazione*” (ogni altra scelta infatti non garantirebbe in tutti i casi l'univocità del record). Seguendo lo stesso metodo illustrato prima, aprire la tabella “libri” in vista “struttura” e definire la collocazione del libro come chiave primaria della tabella.

Attenzione: se per caso era già stata inavvertitamente definita qualche altra chiave primaria (cioè se il simbolo della chiave compare accanto a qualche altro campo), rimuoverla prima di definire la nuova chiave primaria della tabella

3) Tabella “prestiti”

Per quanto riguarda la tabella “prestiti” sembra più difficile definire una “chiave primaria” perché in realtà nessuno dei campi della tabella permette di identificare univocamente il prestito (l'utente può avere chiesto in prestito più libri e lo stesso libro, nel corso del tempo, può essere stato prestato a più utenti).

Aggiungiamo in questo caso un nuovo campo alla tabella “prestiti” (menu “Inserisci/Righe”). Diamogli il nome “IDPrestito” (cioè “identificatore del prestito”, seguendo una convenzione di nomi largamente usata nei DBMS) e scegliamo come tipo “contatore”. Il tipo “contatore” è in pratica un numero intero progressivo che viene assegnato automaticamente da Access a ogni nuovo prestito (cioè a ogni nuovo *record* in tabella) e in tal modo identifica univocamente, senza possibilità di duplicati, un prestito da tutti gli altri. Ogni volta che non si riesce o è complicato

individuare una *chiave primaria*, è possibile ricorrere a un campo contatore per questo scopo. Definiamo dunque il campo “IDPrestito” come chiave primaria della tabella “Prestiti”.

4) Verifica funzionamento

Per verificare il funzionamento della chiave primaria, proviamo per esempio ad aprire la tabella “libri” in modalità Foglio Elettronico e aggiungiamo un nuovo record, cioè un libro qualsiasi. Proviamo però ad usare come collocazione del libro una collocazione già usata per un altro libro. Come si può notare, ACCESS impedisce la duplicazione della chiave primaria.

C) Inserimento dati nella tabella Prestiti

Tornando ora alla tabella “Prestiti”, se vogliamo identificare univocamente un utente, bisognerà indicarne il relativo codice fiscale (ogni altro dato infatti potrebbe essere duplicato). Allo stesso modo per identificare una particolare copia di un determinato libro occorre indicarne la collocazione. Pertanto, per evitare confusioni, la soluzione migliore è rinominare: utente -> codice fiscale e libro -> collocazione. Per rinominare i campi aprire la tabella Prestiti in visualizzazione struttura.

Proviamo ora a inserire qualche dato nella tabella “Prestiti”. Il campo “consegnato” (di tipo “sì/no”) appare come una casella nella quale è possibile inserire un segno di spunta.

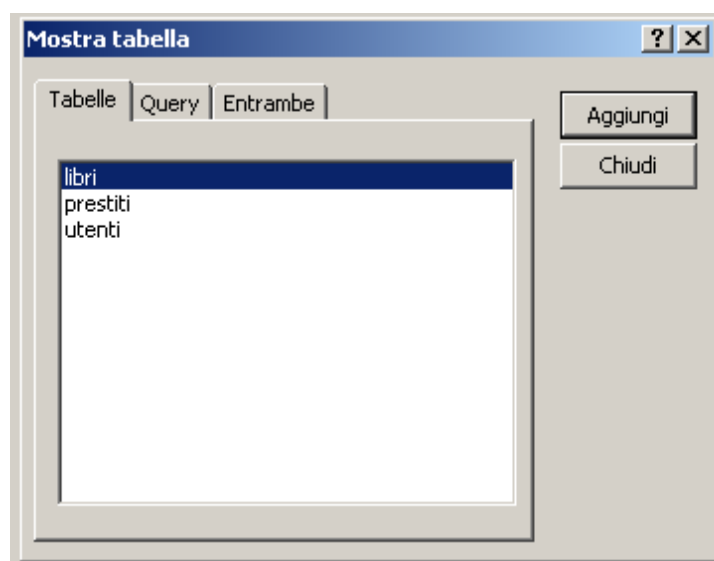
Un problema che sorge subito è quello del riempimento dei campi “codice fiscale” e “collocazione”. Prendiamo per esempio il campo “codice fiscale”. E’ molto importante che il codice fiscale inserito corrisponda effettivamente a un utente registrato alla biblioteca. In caso di errori (per esempio digitando un codice fiscale sbagliato o inserendo il codice fiscale di un utente che non è stato ancora registrato nella tabella Utenti), si verrebbe a creare un problema di **incongruenza** all'interno del DB, ovvero ci sarebbe un prestito registrato a nome di un utente che non esiste.

Un problema perfettamente analogo potrebbe verificarsi con il campo “collocazione”. Il libro preso in prestito deve già comparire nella tabella “Libri” e dev’essere identificato con precisione e senza possibilità di errore.

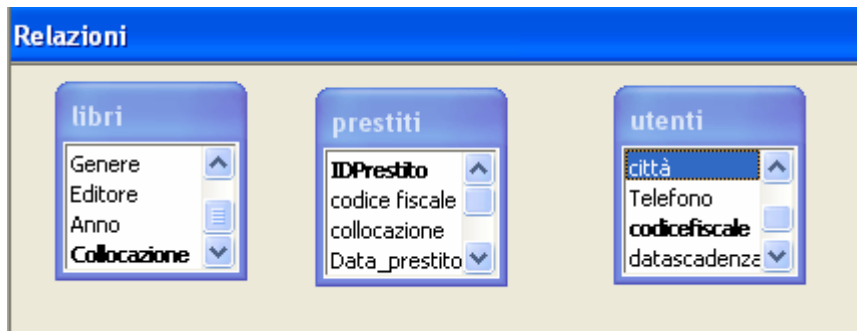
D) Creare una relazione

Abbiamo visto come si fa a definire una *chiave primaria* per una tabella e qual è il suo significato. Non abbiamo però ancora visto in pratica a cosa serve la *chiave primaria*, cioè quale sia il suo scopo.

Per vedere tutto questo, dal menu principale di Access scegliamo Strumenti/Relazioni. Dovremmo vedere una finestra di questo tipo:



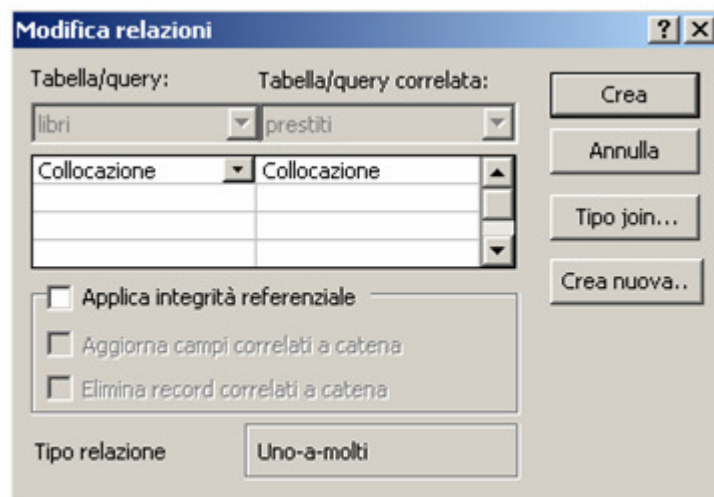
Scegliamo “Aggiungi” per tutte e tre le tabelle. Il risultato è il seguente:



Ogni tabella è rappresentata da un rettangolo contenente l'elenco di tutti i campi. La chiave primaria di ogni tabella è indicata in grassetto.

Abbiamo detto che vogliamo fare in modo che ogni prestito corrisponda a un solo utente e a un solo libro, senza possibili ambiguità, duplicazioni ed errori. Questo significa che dobbiamo creare due **relazioni**, rispettivamente fra la tabella “prestiti” e la tabella “libri” e fra la tabella “prestiti” e la tabella “utenti”.

Proviamo per prima cosa a creare una relazione fra la tabella “libri” e la tabella “prestiti”. Per fare questo facciamo clic col mouse sulla chiave primaria della tabella “libri” (cioè “Collocazione”) e trasciniamola sul campo “Collocazione” della tabella “Prestiti”. La tabella Libri (quella con la chiave primaria) viene detta *tabella primaria* (o anche *tabella genitore* o *master*), mentre la tabella Prestiti viene detta *tabella correlata* (o *figlia*). A questo punto rilasciamo il tasto del mouse e ci appare una finestra del tipo:



Spuntiamo la casella “Applica integrità referenziale”. L’*integrità referenziale* è una funzione con cui Access garantisce che ogni record della tabella correlata corrisponda opportunamente a un record della tabella primaria. In parole più semplici: Access impedisce, per esempio, l’eliminazione di un libro o di un utente, senza che vengano eliminati anche tutti i prestiti di quel libro o intestati a quell’utente.

ATTENZIONE: Se sono stati già inseriti dei dati errati nella tabella prestiti (p.es. un codice fiscale di un utente che non esiste), ACCESS segnalerà un messaggio di errore quando si tenterà di applicare l’integrità referenziale alla relazione. Individuare la causa di errore e correggerla.

Facciamo adesso clic sul pulsante “Crea”. Il risultato è il seguente:



La linea indica che abbiamo stabilito una relazione fra il campo “Collocazione” (*chiave primaria* della tabella “libri”) e il campo “Collocazione” della tabella “prestiti” (detta *chiave esterna* della nostra relazione). Il simbolo ∞ accanto a “Collocazione” in “Libri” significa che ogni record della tabella libri (cioè in pratica ogni libro in biblioteca) può essere dato in prestito nel tempo molte volte a più utenti. In altre parole, ogni record di “Libri” (tabella primaria) può corrispondere a molti record di “Prestiti” (tabella correlata). Una relazione di questo tipo è detta “**uno a molti**”.

Seguendo lo stesso procedimento, creiamo una relazione fra “codice fiscale” in “utenti” e “Utente” in “prestiti”. Ecco il risultato finale:



Abbiamo in questo modo creato un **Data Base Relazionale**, cioè un DB nel quale le tabelle sono collegate fra di loro attraverso relazioni.

Attenzione:

Quando si crea una relazione fra due tabelle, uno dei due campi dev'essere la chiave primaria della tabella. Per l'esattezza, la chiave primaria dev'essere la parte '1' della relazione, mentre la chiave esterna (cioè il campo relazionato sulla tabella correlata) è indicata con il simbolo di “infinito”.

Chiudiamo la finestra “Relazioni” salvando il risultato.

E) Una nota sui tipi e nomi delle chiavi correlate

Quando si mettono in relazione due tabelle mediante una chiave primaria e una chiave esterna, l'unica regola imposta da Access è che i tipi dei rispettivi campi siano fra di loro compatibili, cioè possano accogliere gli stessi dati. In altre parole, sarebbe sbagliato definire la chiave primaria di tipo *Data* e la chiave correlata di tipo *Sì/No*. La regola più semplice da seguire è fare in modo che i due campi abbiano sempre lo stesso tipo.

Per quanto riguarda i nomi, Access non impone nessuna regola particolare. Tuttavia è certamente più semplice dare lo stesso nome alle due chiavi. Per esempio, volendo, avremmo potuto chiamare “Libro” e “Utente” i due campi della tabella “Prestiti”, come abbiamo fatto all'inizio, ma questo può portare a confusione. D'altra parte occorre prestare attenzione a non confondersi neppure così: i campi “collocazione” in Libri e in Prestiti sono due campi diversi, appartenenti a tabelle differenti e sono collegati solo dopo che è stata stabilita una relazione fra loro.


G) Inserimento dei dati

Inserendo i dati nella tabella *prestiti* bisogna a questo punto fare in modo che il *codice fiscale* corrisponda al *codice fiscale* di un utente registrato nella tabella *utenti* e la *collocazione* corrisponda a un libro presente in *libri*.

Se per esempio apriamo la tabella *prestiti* in modalità Foglio elettronico e proviamo a inserire un codice fiscale e una collocazione inesistenti, otteniamo un messaggio di errore di questo tipo:

prestiti : Tabella							
	IDPrestito	codice fiscale	collocazione	Data_prestito	Scadenza	Data_consegna	Consegnato
✎	1	323	3238	03/04/2007	08/02/2009	05/04/2002	<input checked="" type="checkbox"/>
*	(Contatore)						<input type="checkbox"/>

Microsoft Office Access

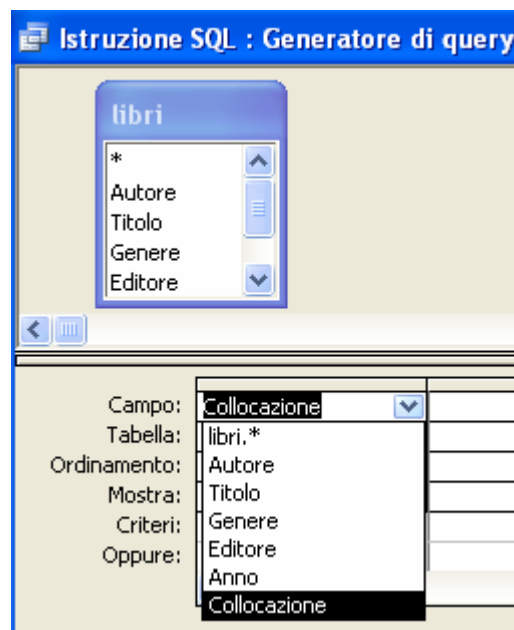
 Impossibile aggiungere o modificare il record. Nella tabella "libri" è necessario un record correlato.

OK ?

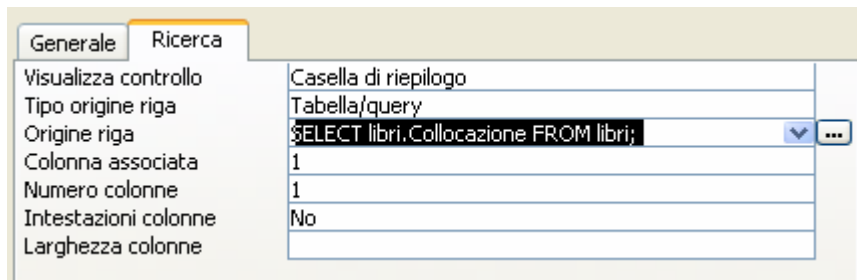
Per assicurare la corrispondenza fra i campi potremmo aprire contemporaneamente tutte e tre le tabelle in modalità Foglio dati e poi copiare e incollare i rispettivi valori dei campi da una tabella all'altra. Non è certamente una soluzione comoda!

Meglio procedere in un altro modo.

- apriamo la tabella *Prestiti* in visualizzazione struttura
- posizioniamo il mouse sul campo *collocazione* e apriamo la scheda *Ricerca*
- scegliamo Casella combinata o Casella di riepilogo in Visualizza controllo e facciamo clic sui puntini accanto alla voce Origine riga e scegliamo la tabella Libri
- appare la scheda Istruzione SQL.; scegliamo Collocazione dall'elenco dei campi
- chiudiamo infine la scheda



A questo punto osserviamo che nella riga Origine riga è comparsa una “strana” scritta “SELECT libri.Collocazione FROM libri;”. Si tratta di una istruzione in un linguaggio detto SQL (Structured Query Language).

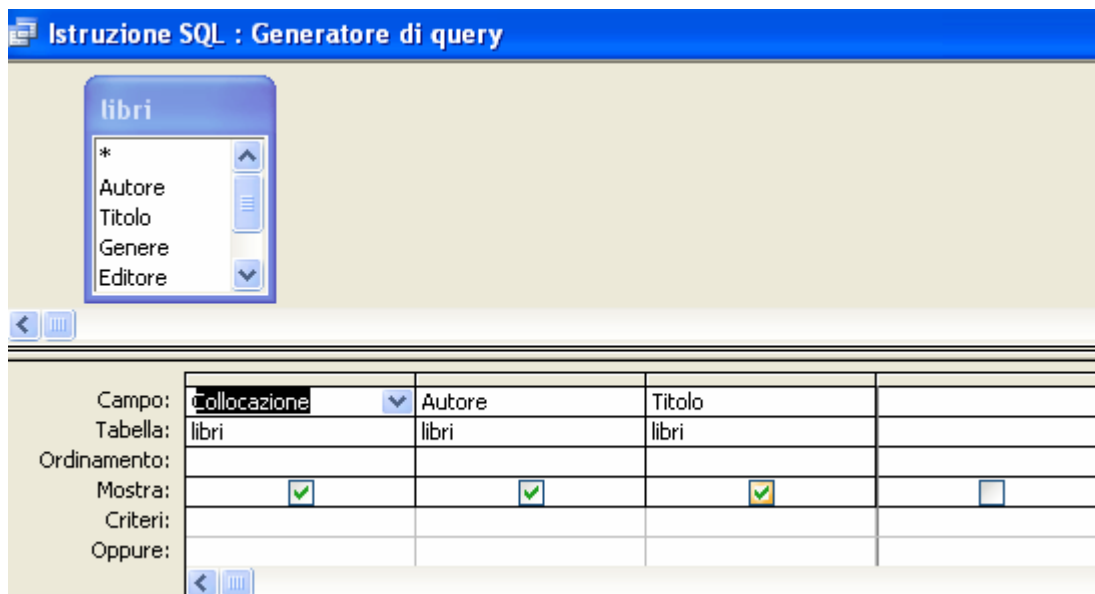


Il suo significato non è poi tanto misterioso. Analizzandola un attimo vediamo che
 SELECT libri.Collocazione FROM libri;
 significa semplicemente “SELEZIONA IL CAMPO Collocazione DALLA TABELLA libri”.

Apriamo ora la tabella prestiti in modalità Foglio dati e vedremo che ora, in corrispondenza del campo Collocazione, compare un elenco di valori possibili selezionati dalla tabella Libri. A questo punto dobbiamo semplicemente selezionare dall’elenco il libro che è stato preso in prestito.



In realtà la sola collocazione dell’opera potrebbe essere un’informazione troppo limitata per l’operatore che registra il prestito. Possiamo migliorare il nostro elenco a discesa. Per fare questo torniamo sulla tabella *prestiti* in visualizzazione struttura, apriamo di nuovo la scheda *Ricerca* e quindi facciamo clic sui puntini accanto a *Origine riga* per aprire la scheda *Istruzione SQL: Generatore di Query*. Impostiamo i parametri come mostrato in figura:

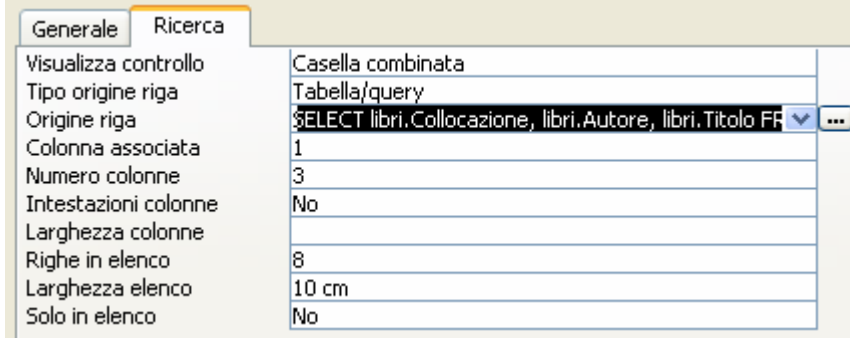


Si noti che abbiamo così selezionato, oltre alla Collocazione del libro, anche il suo Autore e Titolo. Chiudiamo la scheda e torniamo alla vista Struttura. Il valore di “Origine riga” adesso è diventato:

SELECT libri.Collocazione, libri.Autore, libri.Titolo FROM libri;

Anche questa è ovviamente una istruzione SQL, non molto difficile da comprendere. Come si vede Access genera automaticamente istruzioni SQL attraverso la scheda "Istruzioni SQL: Generatore di query". Conoscendo la sintassi del linguaggio sarebbe ovviamente possibile scrivere a mano queste istruzioni senza l'ausilio del *Generatore di query* automatico.

Per completare il nostro lavoro scegliamo Casella Combinata e impostiamo a 10 cm la larghezza dell'elenco:



Se adesso apriamo la tabella prestiti in modalità Foglio dati osserviamo che il nostro elenco a discesa contiene anche l'autore e il titolo dell'opera:

prestiti : Tabella							
	IDPrestito	codice fiscale	collocazione	Data_prestito	Scadenza	Data_consegna	Consegnato
▶	1		▼	03/04/2007	08/02/2009	05/04/2002	☑
*	(Contatore)		b4081	christie		10piccoli indiani	☐
			f3246	dahl		la fabbrica del cioc	
			h4321	brown		il codice da vinci	
			hfge2	ammaniti		io non ho paura	
			k6543	dante		divina commedia	
			l7890	pirandello		il fu mattia pascal	
			n6789	follet		le gazze ladre	
			p0987	manzoni		promessi sposi	▼

Selezionando un libro dall'elenco, il campo *collocazione* della tabella *prestiti* viene riempito automaticamente con la collocazione del libro.

Modificare allo stesso modo anche la visualizzazione del campo *codice fiscale* in modo tale che vengano visualizzati, oltre al *codice fiscale* dell'utente, anche il suo *nome e cognome*.

H) Esercizi di riepilogo

1) Accertarsi di aver compreso bene il significato e di saper definire correttamente i termini: chiave primaria, relazione, chiave esterna, tabella primaria, tabella correlata.

2) Creare un DB contenente due tabelle: la tabella degli insegnanti di una scuola (nome, cognome, indirizzo, telefono...) e la tabella delle materie insegnate per ogni classe (classe, materia, insegnante). Creare una relazione fra le due tabelle e riempirle con dati a piacere. Se occorre, migliorare la visualizzazione dei dati da inserire secondo quanto spiegato nel punto H precedente.

3) Creare un DB con le prenotazioni aeree di una agenzia turistica. Usare le seguenti tabelle: Prenotazioni (codice cliente, codice volo, codice impiegato, data prenotazione, data volo, costo, pagato), Voli (codice volo, compagnia aerea, aeroporto di partenza, aeroporto di destinazione, durata volo, tipo di volo [linea, charter]), Clienti (codice cliente, nome, cognome, indirizzo, telefono), Impiegati (codice impiegato, nome, cognome, indirizzo, telefono, livello salariale [impiegato semplice, direttore di sezione, direttore generale]). Se occorre, migliorare la visualizzazione dei dati da inserire secondo quanto spiegato nel punto H precedente.