

Gestione dei dati (autore: Vittorio Albertoni)

Premessa

Nel documento dedicato al software libero per calcolare, parlando del modulo Calc di Libre Office, abbiamo già accennato alla possibilità di creare e gestire semplici database su una tabella di Calc, che offre funzioni di ricerca e ordinamento dei dati.

Se vogliamo creare un database per la nostra biblioteca, come minimo dovremo tener nota di autore e titolo, in modo da poter trovare un titolo che cerchiamo e da poter vedere che titoli abbiamo di un certo autore: per non parlare di un possibile interesse a tener nota anche dell'editore, del genere, dell'anno di stampa, dell'epoca e del Paese in cui è vissuto l'autore, dello scaffale della biblioteca in cui è collocato il volume, ecc. Ma limitiamoci, per il nostro esempio, a autore e titolo.

Se utilizziamo un foglio di calcolo intestiamo due colonne affiancate, una "autore" e una "titolo" e, riga per riga, inseriremo autore e titolo dei nostri libri, creando, per ognuno di essi, quello che si chiama un record del nostro database. Se abbiamo 18 libri di Ken Follett, dobbiamo ovviamente scrivere 18 titoli e dobbiamo anche scrivere inutilmente per 18 volte la stessa cosa, che è l'autore Ken Follett, augurandoci di scriverlo sempre nello stesso modo: se una volta scriviamo Ken Follett, un'altra scriviamo Follett Ken, un'altra ancora Follett K. e qualche volta sbagliamo pure in Folett, vai poi a tirar fuori dal database l'elenco dei titoli di Ken Follett che abbiamo in biblioteca!

Se invece utilizziamo un altro modello di data base in cui facciamo due tabelle, una per gli autori, l'altra per i titoli, entrambe con due colonne, scriveremo una sola volta - badando bene di scriverlo giusto - il nome dell'autore Ken Follett nella tabella degli autori, nella quale gli verrà a corrispondere, nella colonna a fianco, un numero assegnato solo a lui e scriveremo i nostri 18 titoli nella tabella dei titoli nella quale, nella colonna a fianco indicheremo il numero dell'autore: il numero che compare in entrambe le tabelle costituisce la relazione tra le due tabelle e il data base che funziona in questo modo si chiama database relazionale.

Abbiamo risparmiato spazio (per 18 volte a un nome e cognome abbiamo sostituito un numeretto) e abbiamo guadagnato in sicurezza (il numeretto è sempre quello e non ci sono possibilità di scrivere male l'autore).

L'esempio non dà l'idea adeguata del risparmio di spazio; per averla pensiamo se in una colonna ci dovesse essere l'intestatario di un conto corrente bancario, con il suo indirizzo, il suo codice fiscale, ecc. e nell'altra colonna l'importo di un'operazione bancaria in un database destinato a produrre l'estratto conto trimestrale per il correntista e pensiamo a una banca che abbia qualche centinaia di migliaia di correntisti con qualche milione di operazioni alla settimana e ci rendiamo conto dello spazio che si risparmia evitando di dover scrivere tutte le volte nome, cognome, indirizzo, codice fiscale, ecc. dei correntisti di fianco a ciascuna operazione bancaria.

Per non parlare della velocità con cui si possono realizzare ricerche e ordinamenti agendo su campi numerici piuttosto che su campi alfanumerici.

Chiarito che cosa intendiamo per database relazionale vediamo che strumenti ci offre il software libero in questo campo.

Indice

1	Libre Office, modulo Base	1
2	MySQL	4
3	Sqlite	4

1 Libre Office, modulo Base

Il modulo Base di Libre Office è un ottimo strumento per gestire database e si presta a creare e gestire database propri sullo stesso computer su cui è installato o a gestire database creati altrimenti e non necessariamente localizzati sul computer su cui è installato; per creare intendiamo stabilire che cosa deve contenere il database, in quali colonne, ecc. e per gestire intendiamo inserire nuovi dati, effettuare ricerche sui dati, ecc., cioè tutto ciò che si può fare con un database una volta che è stato creato.

Attualmente la finestra è posizionata sul record n. 1862 relativo al romanzo Cenere di Grazia Deledda e la figura ci aiuta a chiarire il meccanismo del database relazionale.

Questa finestra ci serve per immettere dati nella tabella dei titoli delle opere e vediamo che in questa tabella occorre inserire, in due finestrelle di testo delimitate con sfondo bianco, il titolo e un eventuale campo note, e, in tre finestrelle delimitate con sfondo bianco terminanti con un pulsante per l'apertura di un menu a tendina, i campi denominati idAutore, idGenere e idSupporto. Ciò che vediamo in queste finestrelle è il nome in chiaro dell'autore, del genere e del supporto, ma in realtà, nella tabella, ne scriviamo le rispettive id, cioè quei numeri che la tabella Autori assegna a Grazia Deledda, che la tabella Generi assegna al romanzo e che la tabella Supporti assegna al file epub. I nomi in chiaro si vedono in quanto, per inserire l'autore del titolo Cenere, abbiamo aperto il menu a tendina per la idAutore e ci siamo trovati di fronte un elenco da cui abbiamo selezionato Grazia Deledda, elenco che deriva dal collegamento instaurato tra il nostro formulario per la tabella Opere e la tabella Autori (v. figura 3).

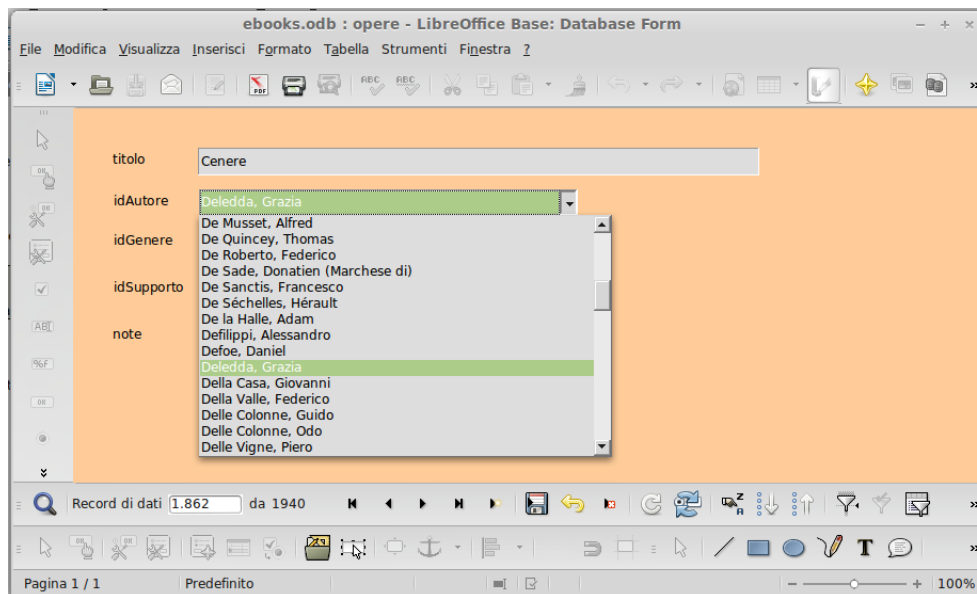


Figura 3: Inserimento dell'autore nel formulario per la tabella Opere

La voce Rapporti riguarda la strutturazione del modo con cui presentare i risultati di una ricerca. Attraverso una procedura guidata possiamo scegliere tra diverse impostazioni che ci vengono proposte ma, come sempre succede in questi casi, troviamo magari ghirigori di forma in eccesso e carenza di sostanza, comunque non troviamo la possibilità di fare quello che esattamente vorremmo fare noi.

Qui ci aiuta una formidabile possibilità che ci offre Libre Office, quella di collegare il database del modulo Base al foglio di calcolo del modulo Calc.

I database creati o ai quali ci colleghiamo con il modulo Base sono infatti “registrabili” in Libre Office.

Se apriamo il modulo Calc e premiamo F4 ci viene presentato l'elenco dei database registrati. Se il nostro database non è ancora registrato lo possiamo registrare facendo click destro nella finestrella dove dovrebbe comparire, scegliere DATABASE REGISTRATI... e seguire le istruzioni della finestra di dialogo che ci viene proposta.

Dei database registrati possiamo importare nel foglio di calcolo, con semplice trascinarsi del mouse, le tabelle e i risultati delle ricerche. Una volta che i risultati della nostra ricerca sono nel foglio di calcolo li possiamo manipolare graficamente come vogliamo, scegliendo tipi e dimensioni dei caratteri, intestazioni maggiormente esplicative delle colonne e, perchè no, arricchire il nostro rapporto con commenti e grafici.

Tutto ciò che abbiamo visto va ovviamente preparato nel modo dovuto: tabelle, relazioni, tipi di dato maschere di inserimento dei dati vanno progettati e realizzati; le ricerche bisogna saperle fare. Questi compiti non sono compiti banali e non è questo il luogo per approfondimenti.

Fortunatamente ci viene in aiuto la rete, dove troviamo parecchio materiale che ci guida a fare grandi cose: basta scrivere su una barra di ricerca “libreoffice base manuale” e ci verranno proposte molte guide, tra cui segnalo, per completezza e rigore, quella del prof. Crescenzo Gallo.

Per chi abbia già una qualche dimestichezza con i database le schede contenute nelle figure 4 e 5 penso siano sufficienti per rendersi conto di come funziona Base e per cominciare ad utilizzarlo.

Un avviso ai linuxiani: Libre Office molto spesso si installa automaticamente installando Linux ma non è detto che in questa sede venga installato anche il modulo Base. Nel caso lo si può installare successivamente ricorrendo al gestore dei programmi.

Il lavoro che facciamo con Base viene salvato in file con estensione `.odb`, che sono propri di Libre Office e non sono utilizzabili altrove. Nel caso il database sia esterno e collegato a Base, il file `.odb` contiene unicamente quanto serve per gestire il database ma il database rimane quello esterno e collegato. Nel caso, invece, il database sia stato anche creato con Base, il file `.odb` conterrà il database, che non si vedrà altrove se non esportandolo in un data base esterno: l'esportazione del solo database, per chi sa lavorare in questo campo, è comunque un'operazione banalissima (basta convertire le tabelle del database in file `.csv` e usare questi file per alimentare le tabelle del nuovo database).

2 MySQL

MySQL è il più famoso Relational database management system (RDBMS) che offra il mondo del software libero. Le piattaforme di riferimento native di questo potentissimo software sono Linux e Oracle Solaris, ma esso funziona praticamente su tutti i sistemi operativi, ivi compresi, per stare al mondo dei personal computer che qui ci interessa, Windows e OS X.

L'architettura di MySQL è client/server, nel senso che esiste un computer server su cui è collocato il database e esistono un qualsiasi numero di programmi applicativi client che comunicano con il server per interrogare, aggiornare o modificare il database. Il server può essere collocato sul nostro stesso computer, su un altro computer collegato al nostro in rete, o su Internet: in ogni caso, se il database è gigantesco e non può essere ospitato sul nostro piccolo PC, lo mettiamo su un computer più capiente, magari affittando spazio da un service provider presente su Internet, ma lo possiamo utilizzare dal nostro piccolo PC come se fosse lì.

A questo punto ci rendiamo conto che siamo di fronte a qualche cosa che esula dall'interesse dell'utente pur di buon livello, ma pur sempre dilettantistico-casalingo, al quale dedico queste note.

In una rassegna dedicata al software libero non si poteva comunque non ricordare che esiste MySQL e, per i volenterosi che vogliono approfondire, tutto si può trovare su *mysql.it*, da dove possiamo scaricare il software, oltre che su una vasta raccolta di libri e libretti che ne parlano.

Ovviamente, uno dei programmi applicativi client con cui si può lavorare su un database MySQL è il modulo Base di Libre Office che abbiamo appena visto. All'apertura di Base basta scegliere COLLEGA A UN DATABASE ESISTENTE e, nel menu a tendina che compare, scegliere MYSQL, poi AVANTI e seguire la procedura guidata.

Una volta collegato il database a Base salviamo il nostro lavoro su un file `.odb` di Libre Office e, da qui in poi, lavoreremo su quello senza più bisogno di instaurare collegamenti.

3 Sqlite

Quando Hipp ha scritto questa minuscola libreria software, rendendola immediatamente di pubblico dominio, non pensava probabilmente che stava producendo il più efficiente motore di database che esista e che in pochi anni le installazioni del suo software avrebbero superato i 500 milioni, facendone il database più diffuso al mondo.

Efficiente in quanto siamo in presenza di un programma che occupa non più di 5 mega sul nostro computer e può gestire database grandi fino a 2 TB (cioè 2 mila miliardi di byte). Il database è contenuto in un solo file che può essere utilizzato su qualsiasi sistema operativo semplicemente copiandolo e incollandolo. E quando diciamo qualsiasi sistema operativo, diciamo proprio qualsiasi, perchè Sqlite, oltre a stare benissimo su Linux, Windows e OS X, è di casa su tutti i dispositivi Android, iOS8, Symbian OS, ecc., per cui ce lo troviamo su tablet e smartphone di tutti i tipi. I database che ho sul mio computer, tutti sqlite, li consulto anche sul mio telefonino.

La versione corrente si chiama **sqlite3** e la possiamo scaricare liberamente da *sqlite.org*, dove, purtroppo in lingua inglese, troviamo ampia documentazione per il suo utilizzo. L'installazione, avendo ovviamente scaricato la versione adatta al nostro sistema operativo, non presenta problemi.

Il problema è l'uso.

CREAZIONE DI FORMULARIO IN LIBRE OFFICE – BASE

E' semplice e immediata utilizzando la procedura guidata.

Se si vuole utilizzare un menu a tendina per una casella di inserimento dei dati:

1. si crea una tabella con due colonne:
 - nella prima, contenente il campo definito come chiave primaria, si mette il dato da inserire nella tabella principale del database (nel caso di campo autoincrementato questo dato sarà un numero progressivo),
 - nella seconda si mette la descrizione in chiaro che faremo comparire nel menu di scelta;
2. si crea una ricerca con Crea query in vista struttura riferita alla tabella appena creata, mettendo nella prima colonna il campo che contiene la descrizione in chiaro e nella seconda il campo contenente ciò che dovrà essere inserito nella tabella principale del database;
3. si apre il formulario creato con la procedura guidata in modalità modifica (click destro Modifica);
4. si seleziona la finestrella di inserimento cui si vuole applicare il menu a tendina (Ctrl click sinistro);
5. si fa click destro sulla selezione e si sceglie Sostituisci con Casella di Riepilogo;
6. si fa di nuovo click destro sulla selezione e si sceglie Campo di controllo...:
 - nella scheda Generale si mette Sì in Apribile,
 - nella scheda Dati, dove compare automaticamente il nome del campo destinatario dell'inserimento nella tabella principale del database:
 - si sceglie Query in Tipo del contenuto della lista,
 - si sceglie il nome della tabella creata al punto 2 in Contenuto elenco,
 - si sceglie 1 in Campo collegato.

CREAZIONE DI RICERCA IN LIBRE OFFICE - BASE

In vista struttura

Nello spazio apposito si inseriscono le tabelle coinvolte dalla ricerca, indicandone le relazioni con il mouse.

Nello spazio sottostante si indicano nelle colonne i campi da evidenziare con la ricerca e quelli implicati nell'indicazione dei criteri.

I criteri affiancati nelle colonne rispondono al legame logico **e** (si devono verificare tutti affinché avvenga la scelta).

I criteri incolonnati per riga rispondono al legame logico **o** (la scelta avviene al verificarsi di almeno uno).

In vista SQL

Attraverso il comando

SELECT <elenco dei campi che si vogliono evidenziare nel risultato della ricerca, separati da , >
FROM <elenco, separato da , , delle tabelle coinvolte nella ricerca>

WHERE separate eventualmente da **AND** si indicano le relazioni tra le tabelle coinvolte nella ricerca nella forma **tabella.campo = tabella.campo**
poi, sempre con **AND**, si fanno seguire le condizioni della ricerca nella forma
tabella.campo <operatore di confronto> <dato di confronto>, separate da **AND** o **OR**

Figura 4: Creazione di formulari e ricerche con Base

OPERATORI E COMANDI PER LA FORMULAZIONE DELLE CONDIZIONI IN LIBRE OFFICE – BASE

Le espressioni in formato stringa (testo tra singoli apici ') sono sensibili al maiuscolo e minuscolo.

I caratteri jolly sono:

- * per un numero imprecisato di caratteri,
- ? per un solo carattere.

Le espressioni in formato data (tra singoli apici ') disponibili sono AAAA-MM-GG HH:MM:SS, AAAA/MM/GG HH:MM:SS e AAAA.MM.GG HH:MM:SS

Operatore/Comando	La condizione è soddisfatta se ...
=	...il contenuto del campo è uguale all'espressione indicata
<>	...il contenuto del campo non corrisponde all'espressione indicata
>	...il contenuto del campo è maggiore dell'espressione indicata
<	...il contenuto del campo è minore dell'espressione indicata
>=	...il contenuto del campo è maggiore o uguale all'espressione indicata
<=	...il contenuto del campo è minore o uguale all'espressione indicata
IS EMPTY	...il campo è vuoto
IS NOT EMPTY	...il campo non è vuoto
LIKE <stringa>	...il campo contiene l'espressione indicata: - all'inizio (x*), - alla fine (*x), - all'interno (*x*).
NOT LIKE <stringa>	...il campo non contiene l'espressione indicata
BETWEEN X AND Y	...il campo contiene un valore compreso tra x e y
NOT BETWEEN X AND Y.	... il campo contiene un valore che non è compreso tra x e y
IN (a, b, c ...)	...il campo contiene una delle espressioni indicate a, b, c ...
NOT IN (a, b, c...)	...il campo non contiene alcuna delle espressioni indicate a, b, c ...

Figura 5: Formulazione delle condizioni in Base

Sqlite funziona a riga di comando, cioè in una finestra che accoglie comandi digitati da tastiera (finestra chiamata terminale in Linux, che in Windows è la finestra che si apre con il comando `cmd` e in OS X con il comando terminale su spotlight). Il comando per aprire `sqlite3` è `sqlite3`; se alla parola `sqlite3` facciamo seguire il nome di un database e il database esiste, esso viene aperto; se il database non esiste, esso viene creato con il nome che abbiamo indicato.

Da qui in poi, per riempire, aggiornare, consultare il database abbiamo a disposizione il linguaggio SQL, che è il linguaggio standard per dialogare con i database di tutti i tipi, e, per alcune funzionalità, il linguaggio proprio di `sqlite`: i comandi SQL si scrivono come tali e i comandi del linguaggio proprio si scrivono preceduti da un punto (per questo sono detti comandi puntati).

Le figure 6 e 7 riepilogano ad uso degli smanettoni.

Per semplificare esistono alcuni programmini scaricabili da Internet, che girano su Linux, Windows e OS X, tipo **Sqliteman** o **Sqlite Database Browser**, che ci consentono di utilizzare `sqlite` con interfacce grafiche. Con questi è quanto meno molto semplice creare il database, in quanto, senza scrivere alcun comando ma semplicemente seguendo procedure guidate, si impostano le tabelle con scelta dei vari tipi di dati da inserire nelle varie colonne, ecc.

Una volta creato il database (anche creato semplicemente avendogli dato un nome in `sqlite3`) lo possiamo collegare al modulo Base di Libre Office e facciamo tutto quello che vogliamo.

Il modo più semplice - relativamente - per instaurare il collegamento è ODBC (Open DataBase Connectivity). Occorre preventivamente installare sul proprio sistema operativo il driver ODBC per `Sqlite`, che troviamo sul sito ch-werner.de/sqliteodbc/, e poi indicare, attraverso il programma amministratore di collegamenti ODBC del nostro sistema operativo, i database che ci interessano come possibili sorgenti di dati. Fatto questo, all'apertura del modulo Base di Libre Office, scegliamo COLLEGA A UN DATABASE ESISTENTE, e dal menu a tendina scegliamo ODBC, poi AVANTI; nella finestra di dialogo che si apre premiamo il pulsante SFOGLIA e comparirà l'elenco dei database collegabili, tra i quali possiamo scegliere.

Una volta collegato il database a Base salviamo il nostro lavoro su un file `.odb` di Libre Office e, da qui in poi, lavoreremo su quello senza più bisogno di instaurare collegamenti.

Comandi speciali sqlite3 (comandi puntati)

La lista completa dei comandi puntati si ottiene con il comando puntato `<.help>` inserito al prompt di sqlite3:

```
sqlite> .help
```

I comandi più comunemente utilizzati sono i seguenti:

<code>.exit</code> (oppure <code>.quit</code>)	Provoca l'uscita dal programma sqlite3
<code>.header on off</code>	Inserisce o disinserisce le intestazioni delle colonne nell'output
<code>.import <file> <tabella></code>	Importa i dati di un file in una tabella (vedi <code>.separator</code>)
<code>.mode <mode></code>	Imposta il formato di output. <mode> sta per:
csv	valori separati da virgola
column	valori in colonna, allineati a sinistra (vedi <code>.width</code>)
line	un valore per riga
list	valori elencati e separati come scelto con il comando <code><.separator></code>
tabs	valori separati da uno spazio di tabulazione
tcl	valori come lista di elementi TCL
<code>.output <nome_file></code>	Redirige l'output sul file <code><nome_file></code>
<code>.output stdout</code>	Redirige l'output allo schermo
<code>.schema [<tabella>]</code>	Mostra lo statement SQL di creazione di tutto il data base o della tabella, se indicata.
<code>.separator '<stringa>'</code>	Indica il simbolo separatore per il mode list e per il comando <code><.import></code>
<code>.show</code>	Mostra il valore corrente dei vari settings
<code>.width {<numero>}</code>	Indica la larghezza delle colonne per il mode column
<code>.dump [<tabella>]</code>	Salva il database in formato SQL (se preceduto dal comando <code>.output <nome_file></code> lo salva in un file)
<code>.read <nome_file></code>	Ripristina il database leggendo il file in formato SQL creato con il comando <code><.dump></code>

Figura 6: Comandi puntati di sqlite 3

Comandi SQL riconosciuti da sqlite3

I comandi SQL si possono scrivere su più righe, andando a capo con Invio, e terminano con il ; .

ALTER TABLE	END TRANSACTION
ANALYZE	EXPLAIN
ATTACH DATABASE	INDEXED BY
COMMIT TRANSACTION	INSERT
CREATE INDEX	ON CONFLICT clause
CREATE TABLE	PRAGMA
CREATE TRIGGER	REINDEX
CREATE VIEW	REPLACE
CREATE VIRTUAL TABLE	ROLLBACK TRANSACTION
DELETE	SELECT
DETACH DATABASE	UPDATE
DROP INDEX	VACUUM
DROP TABLE	
DROP TRIGGER	funzioni aggregative
DROP VIEW	funzioni generali

Tipi di dato gestiti da sqlite3

tipo sqlite3	affinità con tipi contenenti la stringa
NULL	
INTEGER	INT
REAL	REAL FLOA DOUB
TEXT	CHAR CLOB TEXT
BLOB	

N.B.: il modificatore `AUTO_INCREMENT` in sqlite3 è scritto `AUTOINCREMENT`

Tutti i comandi, gli attributi e i modificatori possono indifferentemente essere scritti in maiuscolo o in minuscolo.

Figura 7: Comandi SQL riconosciuti da sqlite3