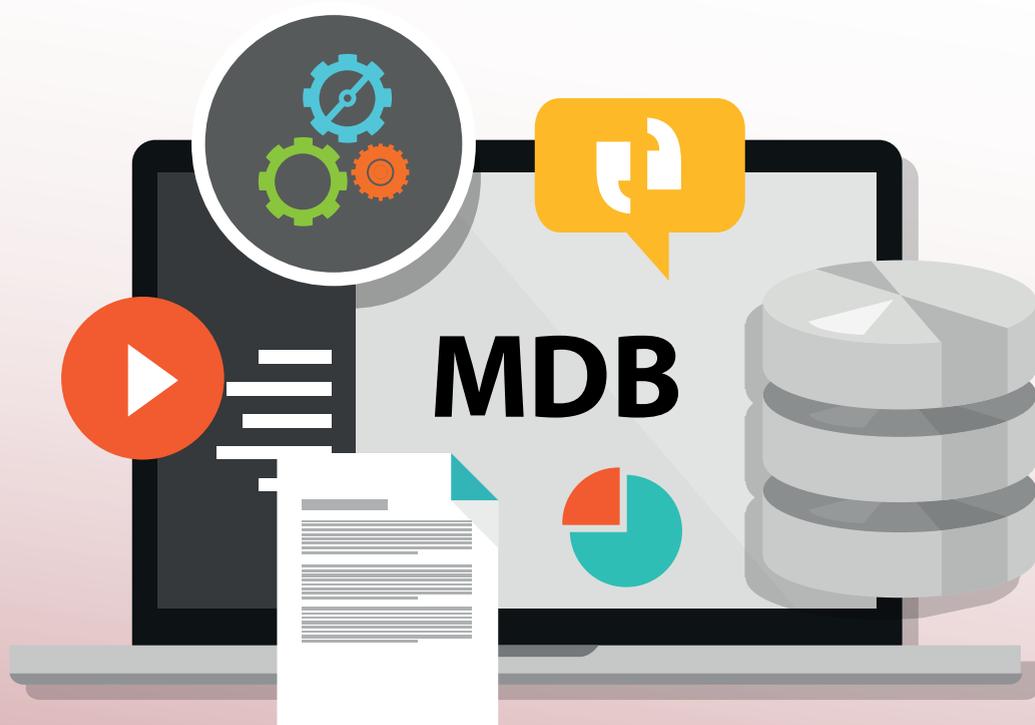


- Alessandra Salvaggio -

LAVORARECON

Microsoft Access 2016

Guida all'uso



Interfaccia e operazioni di base con la nuova versione del programma >>

Uso pratico di tabelle, maschere, query e report >>

Esempi pratici per familiarizzare con Access >>

Creazione e utilizzo delle macro >>

*pro
DigitalLifeStyle

*pro
DigitalLifeStyle

LAVORARECON

Microsoft

Access 2016

Guida all'uso

Alessandra Salvaggio

EDIZIONI
LSWR

Lavorare con Microsoft Access 2016

Autrice: Alessandra Salvaggio

Collana: DigitalLifeStyle^{*pro}

Editor in Chief: Marco Aleotti

Progetto grafico: Roberta Venturieri

Immagine di copertina: © bakhtiarzein | Fotolia

© 2017 Edizioni Lswr* - Tutti i diritti riservati

ISBN: 978-88-6895-446-8

I diritti di traduzione, di memorizzazione elettronica, di riproduzione e adattamento totale o parziale con qualsiasi mezzo (compresi i microfilm e le copie fotostatiche), sono riservati per tutti i Paesi. Le fotocopie per uso personale del lettore possono essere effettuate nei limiti del 15% di ciascun volume dietro pagamento alla SIAE del compenso previsto dall'art. 68, commi 4 e 5, della legge 22 aprile 1941 n. 633.

Le fotocopie effettuate per finalità di carattere professionale, economico o commerciale o comunque per uso diverso da quello personale possono essere effettuate a seguito di specifica autorizzazione rilasciata da CLEARedi, Centro Licenze e Autorizzazioni per le Riproduzioni Editoriali, Corso di Porta Romana 108, 20122 Milano, e-mail autorizzazioni@clearedi.org e sito web www.clearedi.org.

La presente pubblicazione contiene le opinioni dell'autore e ha lo scopo di fornire informazioni precise e accurate. L'elaborazione dei testi, anche se curata con scrupolosa attenzione, non può comportare specifiche responsabilità in capo all'autore e/o all'editore per eventuali errori o inesattezze.

L'Editore ha compiuto ogni sforzo per ottenere e citare le fonti esatte delle illustrazioni. Qualora in qualche caso non fosse riuscito a reperire gli aventi diritto è a disposizione per rimediare a eventuali involontarie omissioni o errori nei riferimenti citati.

Tutti i marchi registrati citati appartengono ai legittimi proprietari.

EDIZIONI
LSWR

Via G. Spadolini, 7
20141 Milano (MI)
Tel. 02 881841
www.edizionilswr.it

Printed in Italy

Finito di stampare nel mese di gennaio 2017 presso "LegoDigit" Srl., Lavis (TN)

(*) Edizioni Lswr è un marchio di La Tribuna Srl. La Tribuna Srl fa parte di LSWR GROUP.

Sommario

INTRODUZIONE	9
1. I DATABASE	11
Tipologie di database	12
I database relazionali.....	12
Gli elementi costitutivi dei database relazionali	12
2. L'INTERFACCIA E LE OPERAZIONI PIÙ COMUNI.....	19
Creare un database desktop.....	20
La barra multifunzione.....	30
La scheda File	34
La barra di accesso rapido	39
Visualizzare gli oggetti.....	42
3. IL FORMATO DI FILE DI ACCESS 2016.....	43
4. PROGETTARE UN DATABASE	47
I tipi di dati	48
Le relazioni	52
5. CREARE UNA TABELLA	55
Le maschere di input	66
La ricerca guidata.....	69
Altri tipi di dati	72
Apportare modifiche in visualizzazione Foglio dati.....	74
6. USARE UNA TABELLA	75
7. IMPORTARE DATI ESTERNI.....	81
8. CREARE TABELLE DA METTERE IN RELAZIONE CON ALTRE TABELLE	91

9.	CREARE UNA TABELLA CON UNA CHIAVE MULTICAMPO.....	95
10.	UNA TABELLA CON UN CAMPO A VALORE MULTIPLO	97
11.	IMPOSTARE LE RELAZIONI TRA LE TABELLE	103
12.	MANIPOLARE I DATI INSERITI NELLE TABELLE.....	109
	Ordinare i dati	110
	Filtrare i dati.....	111
	Calcolare un totale.....	114
13.	LE MASCHERE.....	117
	Creare una maschera standard.....	117
	La visualizzazione Layout.....	119
	Creare una maschera con la creazione guidata.....	122
	I temi.....	125
	La maschera divisa.....	126
14.	LA STRUTTURA DELLE MASCHERE.....	129
	Elementi di una maschera.....	132
	Maschere e controlli associati	132
	Ordine di tabulazione.....	134
	Organizzazione degli elementi.....	135
	Creare una maschera in visualizzazione Struttura	137
	Formattare la maschera.....	143
15.	CREARE UNA MASCHERA A ORIGINE MULTIPLA	147
16.	INSERIRE IN UNA MASCHERA CONTROLLI NON ASSOCIATI	157
	Casella combinata di ricerca	157
	Distinguere visivamente la casella combinata	160
	Inserire un campo calcolato	161
	Usare una maschera a origine multipla.....	164
17.	INTRODUZIONE ALLE QUERY	167
	Creare una query con la creazione guidata	168
	Analizzare una query.....	170
18.	CREARE UNA QUERY IN VISUALIZZAZIONE STRUTTURA	173
	Ordinare i dati in una query	175
	Le proprietà delle query	177
19.	QUERY CON CRITERI	181
	Selezionare i record in base a un intervallo di date presenti in un campo.....	185
	Usare i caratteri jolly	188

Selezionare record che soddisfano piÙ criteri	190
Query su un campo di ricerca a valore multiplo.....	193
20. QUERY CON CAMPI CALCOLATI.....	197
Creare calcoli in una query in visualizzazione Struttura.....	200
Scrivere le espressioni con il Generatore	205
Campi creati con funzioni	207
Campi che coinvolgono piÙ query.....	209
Frequenza di distribuzione: una tabella di supporto	213
21. LE QUERY CON PARAMETRI	217
Parametri su piÙ campi	220
Parametri basati su un'espressione	221
Parametri con date.....	224
22. AUTOCOMPOSIZIONI DI QUERY AVANZATE.....	227
Query a campi incrociati	227
Query di ricerca dei duplicati	231
Query di ricerca dati non corrispondenti	236
23. TIPI DI JOIN.....	239
Self join.....	244
Riepilogo dei tipi di join disponibili	247
24. LE QUERY DI COMANDO.....	249
Query di aggiornamento	249
Query di creazione tabella.....	251
Query di aggiornamento	255
Query di eliminazione.....	257
Evitare la richiesta di conferma per l'esecuzione delle query di comando	259
25. I REPORT	261
Creare un report in visualizzazione Layout.....	261
Creare un report con una creazione guidata.....	264
Analisi di un report creato in vista Layout	265
26. CREARE UN REPORT IN VISUALIZZAZIONE STRUTTURA.....	267
Migliorare l'aspetto di un report.....	272
Livelli di gruppo.....	278
Temi	284
Esportare il report	284
27. LA FORMATTAZIONE CONDIZIONALE.....	285
Le barre di dati	288

28. ETICHETTE.....	291
29. LE MACRO DI DATI	295
Una macro più complessa.....	299
30. MACRO INCORPORATE NELLE MASCHERE E MACRO INDIPENDENTI	311
Una maschera per selezionare un parametro.....	313
31. LE MACRO NELLE CREAZIONI GUIDATE DEI PULSANTI E LE MACRO AUTONOME	325
Le macro nelle creazioni guidate dei pulsanti.....	325
Le macro autonome.....	330
32. MACRO, CASELLE COMBinate IN CASCATA E FUNZIONI VARIE	333
Leggere il valore da una tabella con una macro	337
Abilitare e disabilitare un campo via macro.....	339
33. MACRO PER AVVIARE QUERY DI AZIONE.....	343
34. UN'APPLICAZIONE DATABASE	347
Configurare le opzioni di avvio del database.....	353
Dividere un database	354
INDICE ANALITICO	359

Introduzione

Office 2016 è la nuova versione della suite da ufficio di Microsoft. In questo libro vi presenteremo il nuovo Access 2016 nel modo più semplice ed esaustivo possibile.

Questo libro è rivolto a due generi di lettori: innanzitutto, è per chi non ha mai usato Access o lo ha usato poco. Questi lettori troveranno una guida passo passo per cominciare a orientarsi con il programma. Dopo i primi rudimenti, passeremo ad argomenti più avanzati che permetteranno di approfondire la conoscenza di Access. Tutto questo porterà ad aumentare la produttività in ufficio, nello studio e nel lavoro.

Il libro, però, è utile anche per chi già conosce Access nelle versioni precedenti, in particolare per chi passa alla versione 2016 dalla 2003 o precedenti, saltando la versione 2007: qui vi presenteremo le novità e vi aiuteremo a ritrovare i vecchi comandi. I cambiamenti, soprattutto nell'interfaccia, sono talmente importanti che, a un primo approccio, potrebbero mettere in difficoltà anche l'utente più smaliziato. Abbiamo dunque cercato di evidenziare il più possibile le novità e gli aspetti in grado di generare confusione. Gli utenti più esperti potranno utilizzare questo libro per orientarsi nelle novità e come manuale di consultazione rapida, per risolvere i problemi puntuali che possono presentarsi nel lavoro di ogni giorno.

I lettori verranno guidati passo passo con esempi pratici, in modo da poter trarre il meglio da Access ed essere subito operativi.

Trovate il materiale per svolgere gli esercizi del libro sul booksite:

<http://www.sos-office.it/libroAccess2016.html>.

I database

Prima di cominciare a parlare di Microsoft Access, impareremo a conoscere i database in generale.

Il termine database, in senso stretto, indica una qualsiasi raccolta di dati (base dati). La guida del telefono, un libro di ricette, l'archivio di un ufficio: ogni tipo di archivio è un database.

Nel campo dell'informatica, i database sono raccolte di dati computerizzate. Il termine database, però, viene usato impropriamente anche per designare quei programmi che permettono di gestire dati memorizzati in un archivio elettronico.

Il nome corretto per questa classe di programmi è **DataBase Management Systems** (DBMS), ossia sistemi per la gestione di basi dati. Tuttavia, dato l'uso comune, impiegheremo il termine database per designare sia l'archivio dei dati sia il programma capace di gestire tale archivio.

Sul mercato sono presenti molti database con diverse caratteristiche. Uno dei più diffusi è **Microsoft Access**, che deve il suo grande successo anche al fatto di essere inserito nel pacchetto **Office** di **Microsoft**.

Un database serve per immagazzinare informazioni, per recuperarle facilmente e velocemente e, soprattutto, per elaborarle. Un database, per esempio, permette che più agenzie di viaggi contemporaneamente prenotino voli aerei o posti sui treni o, ancora, consente di compiere operazioni sul proprio conto corrente anche da filiali diverse della banca. Con un database si può sapere quanti e quali clienti di una società non eseguono ordini da un determinato periodo... Con i database, quindi, si possono elaborare i dati in maniera complessa e non ci si limita alla semplice consultazione degli stessi.

Tipologie di database

I database si dividono in due grandi categorie:

- **semplici o piatti:** formati da una sola tabella, memorizzano in un unico archivio dati semplici che richiedono un'elaborazione minima. Sono utili per creare elenchi, come raccolte di indirizzi. Generalmente sono per uso personale. Esempi di database di questo tipo sono l'archivio creato da Word per i dati della stampa unione o le tabelle di Excel;
- **complessi:** sono nati per gestire una maggiore quantità di dati (per esempio, i database per la gestione dei conti correnti nelle banche). Sono rivolti a un uso più professionale e consentono che più utenti modifichino i dati in contemporanea (è il caso degli archivi per la prenotazione dei voli aerei, cui accedono in contemporanea più agenzie di viaggi, connesse con diversi terminali). La gestione non è semplice e rimangono alcuni problemi, come quello della sincronizzazione nell'aggiornamento dei dati. Nel caso dell'archivio della prenotazione dei voli, la non risoluzione di questo problema porta al cosiddetto *overbooking*.

I database complessi si dividono a loro volta in varie categorie, tra cui:

- **object oriented;**
- **relazionali:** sono i più diffusi e anche Access rientra in questa categoria.

I database relazionali

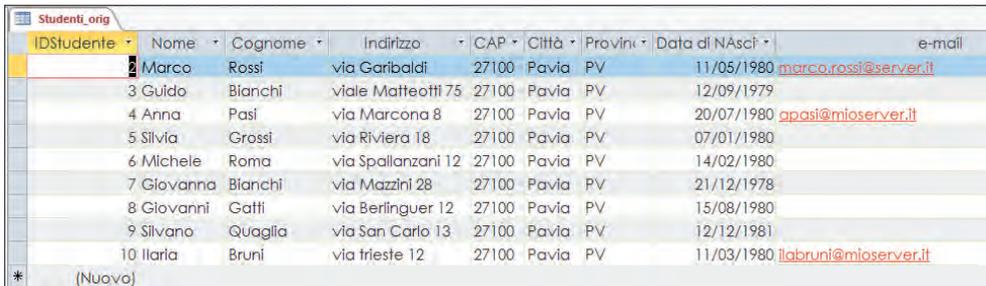
Questo tipo di database è nato alla metà degli anni Settanta. La sua teorizzazione si deve a E. F. Codd, un tecnico IBM che ha gettato le basi per la realizzazione dei moderni programmi di gestione dei database.

Nel suo libro, *A Relational Model of Data for Large Shared Data Banks*, Codd ha spiegato dal punto di vista teorico come devono essere immagazzinati i dati, cioè la loro organizzazione logica. Dal punto di vista pratico, ogni database relazionale presenta soluzioni diverse. Per esempio, Access memorizza tutto l'archivio in un unico grande file con estensione .accdb (.mdb nelle versioni precedenti alla 2007), mentre altri programmi dividono l'archivio in tanti file più piccoli, ma questo non influenza i concetti che portano alla strutturazione dei dati.

Gli elementi costitutivi dei database relazionali

In questo paragrafo faremo una rapida panoramica sugli elementi costitutivi di un database relazionale, mentre nei capitoli seguenti impareremo a creare e a utilizzare realmente questi elementi con Access. In particolare, in questo libro parleremo di tabelle, maschere, query e report.

Le tabelle



IDStudente	Nome	Cognome	Indirizzo	CAP	Città	Provincia	Data di NAsci	e-mail
2	Marco	Rossi	via Garibaldi	27100	Pavia	PV	11/05/1980	marco.rossi@server.it
3	Guido	Bianchi	viale Matteotti 75	27100	Pavia	PV	12/09/1979	
4	Anna	Pasi	via Marcona 8	27100	Pavia	PV	20/07/1980	apasi@mioserver.it
5	Silvia	Grossi	via Riviera 18	27100	Pavia	PV	07/01/1980	
6	Michele	Roma	via Spallanzani 12	27100	Pavia	PV	14/02/1980	
7	Giovanna	Bianchi	via Mazzini 28	27100	Pavia	PV	21/12/1978	
8	Giovanni	Gatti	via Berlinguer 12	27100	Pavia	PV	15/08/1980	
9	Silvano	Quaglia	via San Carlo 13	27100	Pavia	PV	12/12/1981	
10	Ilaria	Bruni	via trieste 12	27100	Pavia	PV	11/03/1980	labruni@mioserver.it

Figura 1.1 - Una tabella.

Simile a un foglio di Excel, la tabella in un database relazionale è un insieme di dati omogenei: per esempio, nell'archivio di una banca, i nomi di tutti i correntisti.

La tabella è formata da colonne e righe. La colonna è una raccolta verticale di dati, mentre la riga è una raccolta orizzontale di dati.

Ogni colonna contiene un tipo di dati (per esempio "Nome", "Cognome") ed è chiamata anche **campo**. Le righe, invece, contengono dati diversi relativi a un solo elemento e sono chiamate anche **record**.

L'intersezione tra una riga e una colonna si chiama **campo** (*field* in inglese), proprio come le colonne, che riuniscono campi dello stesso tipo. Per esempio, nella Figura 1.1 la terza colonna è il campo "Cognome".

È più efficiente dividere i dati in più tabelle piccole, piuttosto che avere una sola grossa tabella. Il fatto di dividere i dati in più tabelle permette, infatti, di evitare duplicazioni di dati (ridondanze) e vuoti, riducendo lo spreco di spazio. Inoltre, nel recupero dei dati, i campi vuoti sono elementi di disturbo, mentre la ridondanza crea complicazioni nell'aggiornamento: il dato ripetuto deve essere aggiornato non una volta sola, ma tutte le volte che appare.

I dati vengono divisi, come abbiamo visto, in più tabelle che devono poi essere messe **in relazione** (da qui il nome database relazionali) tra loro, per ripristinare il collegamento logico dei dati.

La relazione tra tabelle è realizzata mediante il ricorso ad alcuni campi speciali detti **chiavi**. In ogni tabella esiste una **chiave primaria**, che serve a rendere **univoci** i record (per esempio, nella Figura 1.1 le due persone che si chiamano Bianchi di cognome sono chiaramente distinte dalla chiave numerica IDStudente).

La chiave primaria può essere **naturale** o **surrogata**. Nel primo caso, il campo che funge da chiave primaria contiene un dato che sarebbe comunque memorizzato nella tabella. Per esempio, per rendere univoche le righe di una tabella con i dati degli stu-

denti di una facoltà universitaria, si potrebbe usare il loro numero di matricola o, più in generale, il loro codice fiscale: si tratta di dati che occorre comunque memorizzare nella tabella e che, per loro natura, sono adatti a fungere da chiave primaria. Se la tabella non ha un campo adatto a questo scopo, si ricorre alla chiave **surrogata**. A dire la verità, le chiavi surrogate sono più frequenti di quelle naturali, per la loro facilità di utilizzo. Una chiave surrogata non è altro che un campo numerico, spesso chiamato ID (Identificativo), aggiunto alla tabella al solo scopo di rendere univoche le sue righe. La tabella della Figura 1.1 utilizza la chiave surrogata IDStudente.

Le tabelle possono avere anche una **chiave secondaria** o **esterna**, che le mette in relazione con le chiavi primarie di altre tabelle.

Per chiarire meglio quanto appena esposto, possiamo ricorrere a un facile esempio. Immaginiamo di voler creare un database con i numeri di telefono. Quali dati dobbiamo memorizzare? Per esempio, potremmo aver bisogno di:

- nome;
- cognome;
- telefono dell'ufficio;
- telefono di casa;
- cellulare;
- fax.

Se volessimo memorizzare i dati in una sola tabella, potremmo avere il seguente risultato:

Nome	Cognome	Telefono ufficio	Telefono casa	Cellulare	Fax
Camilla	Testa	02/345678			
Gualtiero	Giusti	0382/99999	0382/456789	347/12340347	0382/933456
Leonardo	Rivolta	06/1234567	06/1234678		
Gabriella	Grossi	02/56789	02/433555		
Sara	Belli	0382/7456789		340/788788	
Stefano	Veroni	0382/723456		338/234567	0382/933333

È più efficiente dividere la tabella in due tabelle più piccole (nel caso di un numero maggiore di dati, le tabelle potrebbero essere più numerose):

NOTA

Per non creare esempi troppo lunghi, ci limitiamo a pochi dati, ma normalmente i database hanno tabelle molto più vaste. Con tabelle più ricche di dati è ancora più evidente l'efficienza della suddivisione dei dati in più tabelle.

Tabella nomi

Id	Nome	Cognome
1	Camilla	Testa
2	Gualtiero	Giusti
3	Leonardo	Rivolta
4	Gabriella	Grossi
5	Sara	Belli
6	Stefano	Veroni

Tabella telefoni

Id	Id_nomi	Telefono	Tipo Di telefono
1	1	02/345678	Ufficio
2	2	0382/99999	Ufficio
3	2	0382/456789	Casa
4	2	347/12340347	Cellulare
5	2	0382/933456	Fax
6	3	06/1234567	Ufficio
7	3	06/1234678	Casa
8	4	02/56789	Ufficio
9	4	02/433555	Casa
10	5	0382/7456789	Ufficio
11	5	340/788788	Cellulare
12	6	0382/723456	Ufficio
13	6	338/234567	Cellulare
14	6	0382/933333	Fax

Le tabelle così create rispettano le regole per la buona strutturazione dei database e si dicono **normalizzate**.

NOTA

È impossibile evitare completamente le ripetizioni: ovviamente nella colonna TIPO DI TELEFONO il dato sarà spesso ripetuto, ma trattandosi di un elenco limitato di possibilità, si potrebbe scegliere di assegnare sigle o codici a ciascuna opzione, per ridurre lo spazio occupato da queste informazioni.

Come si nota, ogni tabella ha una chiave primaria (ID) che identifica in modo univoco ciascun record, mentre la tabella TELEFONI ha anche una chiave secondaria (ID_NOMI) che mette in relazione i record della tabella stessa con i record della tabella NOMI: il primo record contiene un numero di telefono di Camilla Testa, il secondo, il terzo, il quarto e il quinto i numeri di Gualtiero Giusti. La Figura 1.2 mostra la relazione tra le due tabelle.

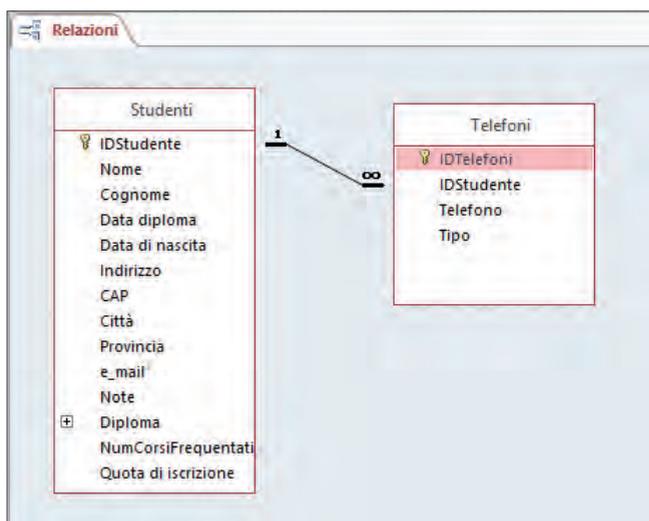


Figura 1.2 - Le relazioni tra le tabelle.

I dati sono divisi nelle tabelle secondo regole dette di **normalizzazione**. Riassumiamo brevemente queste regole.

La **prima regola di normalizzazione** dice che non ci devono essere righe duplicate nella tabella, i dati in una colonna devono essere dello stesso tipo e non ci devono essere voci multiple (in una colonna possono esserci solo nomi o solo cognomi, non nomi e cognomi).

La **seconda regola** dice che tutti i campi che non sono chiave primaria di una tabella devono dipendere esclusivamente dalla chiave primaria della riga in cui si trovano, altrimenti vanno spostati in un'altra tabella.

La **terza regola** dice che i campi non primari non devono essere interdipendenti tra loro (il valore di un campo non può variare al variare di un altro campo).

La **quarta regola** dice che i dati in una riga non possono avere più di una relazione tra loro.

Non preoccupatevi se tutto questo non vi è subito chiaro: piano piano, proseguendo la lettura dei prossimi capitoli, capirete meglio.

Le maschere

Servono a inserire i dati in un database. È anche possibile inserire i dati direttamente nelle tabelle ma, visto che i database relazionali sono spesso formati da tante tabelle, è più agevole creare una maschera per inserire i dati, i quali saranno poi divisi tra più tabelle. Access, come gli altri database, associa una maschera a una o più tabelle in cui registra i dati inseriti nella maschera. La maschera, come si vede nella Figura 1.3, assomiglia a un modulo da compilare, inserendo i dati richiesti negli spazi previsti.

Oltre a rendere più intuitivo l'inserimento dei dati, l'uso delle maschere permette di fornire valori per i campi in cui esiste un insieme ristretto di valori tra cui scegliere (per esempio, in un archivio di dischi, per il campo Supporto si potrà prevedere che nella maschera venga proposta l'alternativa CD, MP3, LP, Cassetta), senza l'obbligo da parte dell'utente di inserire manualmente il dato. Il vantaggio di questa soluzione è evidente: non solo velocizza l'inserimento dei dati, ma rende uniforme il formato del dato (sarà sempre CD e non Compact Disc, Cd...), rendendo più agevole la successiva ricerca.

Con le maschere, inoltre, è possibile inserire, vicino a ciascun campo, una spiegazione che aiuti a inserire correttamente i dati.

The screenshot shows a web application interface for managing student data. The title bar reads 'Studenti'. Below the title, there is a dropdown menu labeled 'Visualizza dati relativi a'. The main content area is divided into several sections. On the left, there are tabs for 'Dati studente', 'Telefoni', 'Certificazioni conseguite', and 'Valutazioni finali'. The 'Dati studente' tab is active. The form contains the following fields:

- IDStudente: Text input with the value '3'.
- Nome: Text input with the value 'Anna'.
- Cognome: Text input with the value 'Pasi'.
- Data di nascita: Text input with the value '20/07/1980'.
- Indirizzo: Text input with the value 'via Marcona 8'.
- Provincia: Dropdown menu with the value 'PV'.
- Città: Dropdown menu with the value 'Pavia'.
- CAP: Text input with the value '27100'.
- NumCorsiFrequentati: Text input.
- Quota di iscrizione: Text input.
- e_mail: Text input.
- Note: A large text area for entering notes.

Figura 1.3 - Una semplice maschera.

Report

I report, o rapporti, servono a presentare i dati. Generalmente sono concepiti per la stampa e presentano i dati organizzati secondo le necessità dell'utente. Grazie ai report, i dati si presentano in modo più elegante rispetto a come si presenterebbero stampando le singole tabelle. Inoltre, i dati saranno selezionati e strutturati nella maniera adeguata alle diverse esigenze.

Query

È il termine inglese per "ricerca". Infatti, una query permette di selezionare i dati in modo da visualizzare solo ciò che è necessario. Con le query si operano selezioni mirate per le diverse esigenze.

Per realizzare una query, occorre un linguaggio che dica al database quale tipo di ricerca eseguire, ossia che impartisca al database le nostre istruzioni. Access utilizza un linguaggio chiamato **SQL** (Structured Query Language). Per usare Access non occorre conoscere l'SQL, poiché il programma permette di creare query grafiche che sono automaticamente convertite in SQL.

Indici

Per rendere più veloci le query, il recupero e l'ordinamento dei dati, si possono creare degli indici. Nell'indice sono registrati i valori di una colonna e la loro posizione nella tabella. Tutti i campi di una tabella possono essere indicizzati. Non bisogna però eccedere nel creare indici, altrimenti tutte le operazioni sulla tabella risultano rallentate.

Macro

Una macro è un set di una o più azioni, ciascuna delle quali esegue una determinata operazione (per esempio, l'apertura di una maschera o la stampa di un report). Le macro consentono di automatizzare le attività comuni. È possibile, per esempio, eseguire una macro che stampa un report quando l'utente seleziona un pulsante di comando.

Moduli

Ospitano il codice VBA (Visual Basic for Applications). VBA è un linguaggio di programmazione inserito nei programmi Office e, quindi, anche in Access. Permette di espandere le capacità base del programma, con l'introduzione di nuove funzionalità, specifiche per il proprio database.

NOTA

Per maggiori informazioni sull'uso di VBA in Access, vi rimando al mio volume *Access: programmazione VBA*, edito da FAG.