

# COSA E' UN DATA-BASE (DB) ?

- è l'insieme di dati relativo ad un sistema informativo

# COSA CARATTERIZZA UN DB ?

- la struttura dei dati
- le relazioni fra i dati

# I REQUISITI DI UN DB SONO:

- la ridondanza minima

i dati non devono essere inutilmente duplicati per problemi di:

- spazio,
- gestione,
- manutenzione,
- affidabilità,
- coerenza

- la permanenza dei dati

la base di dati è protetta contro eventi che possano minacciarne l'esistenza e/o l'integrità

- la condivisione dei dati

piu' utenti devono potere ad un tempo usare la stessa base di dati (supporto unico, aggiornamento unico, coerenza dei dati, affidabilità, ...)

# Schema di una base di dati

Descrizione della struttura dei dati di uno specifico contesto applicativo

# Istanza (o occorrenza) di una base di dati

Valore assunto dai dati di un certo DB in un certo istante

# Organizzazione degli archivi

Le informazioni contenute in memoria vengono organizzate in record logici

Num Record	Nome	Indirizzo	Telefono
1	Rossi Carlo	Via dei Tigli, 32	02-33249187

Un record è composto da campi

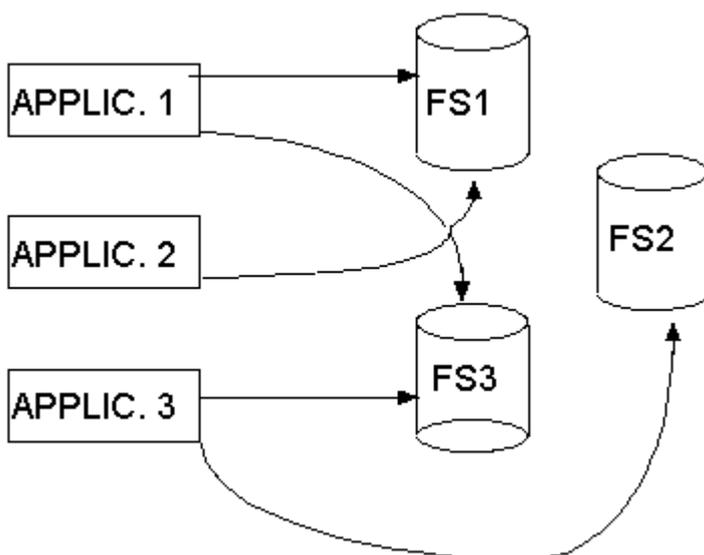
Più record possono essere contenuti in un file

## E' POSSIBILE CONDIVIDERE I DATI?

In un sistema informativo ogni utente fa uso dei vari tipi di dati in misura diversa ed in modo diverso (applicazioni diverse)

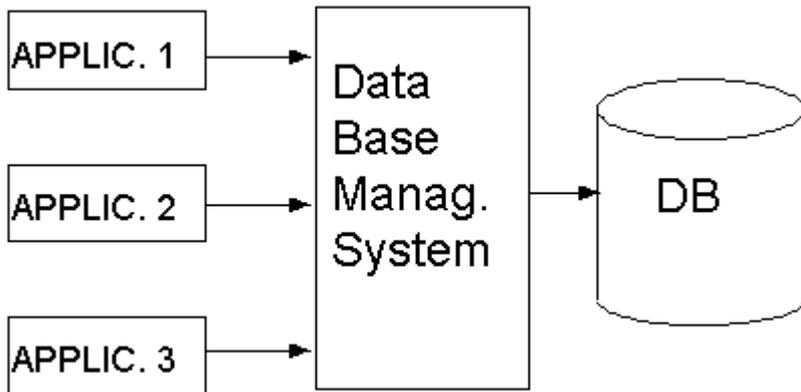
### IERI:

**Applicazioni diverse, file-system diversi**



### OGGI:

## Applicazioni diverse, DBMS, DB unico



## Problemi degli archivi separati

### 1. Inconsistenza e ridondanza dei dati

Vi possono essere differenze tra i valori relativi ad una stessa entità ma riportati in archivi diversi.

La duplicazione di dati crea spreco di memoria

### 2. Riservatezza dei dati

### 3. Integrità dei dati

L'integrità dei dati viene assicurata dai 'vincoli di consistenza'

Ad esempio un campo non può assumere valore negativo.

Con archivi separati, l'integrità dei dati viene affidata a programmi applicativi (soggetti ad errori)

### 4. Concorrenza

Gestire gli accessi contemporanei alla stessa informazione.

## Vantaggi del DBMS

### 1. I dati non sono duplicati

### 2. L'accesso ai dati avviene in base a privilegi fissati dal DBMS

### 3. I vincoli di consistenza possono essere fissati all'interno del DBMS

### 4. L'accesso concorrente ai dati è controllato dal DBMS che gestisce la mutua esclusione dei programmi

# Modelli di dati

- Gerarchico

Basato sugli alberi

- Reticolare

Basato sui grafi

- Relazionale

Basato sugli insiemi, dati strutturati in tabelle

- Ad oggetti

Basato sulle proprietà degli oggetti

## Linguaggi per la gestione dei dati

### DDL (data definition language)

- Per definire lo schema della base di dati

(-» Le definizioni dello schema costituiscono il 'Dizionario dei dati')

### DML (data manipulation language)

- Per inserire, cancellare, modificare i dati

- Per effettuare query

## Il modello relazionale

- Una base di dati relazionale è una collezione di relazioni

- Una relazione è una tabella costituita da un numero fisso di colonne (dette attributi) e un numero variabile di righe (dette tuple)

- Ciascuna colonna assume valori estratti da uno stesso dominio

- Il numero di colonne di una relazione si chiama grado, il numero di righe cardinalità

## Schema di una relazione

E' la descrizione della struttura della relazione

**Schema = Nome della relazione + [nome di attributo + dominio dell'attributo]**

## Esempio

Relation Conto\_corrente (Numero\_cc: integer, Nome: char(20), Indirizzo: char(20), saldo: integer)

## Istanza di una relazione

Insieme delle tuple presenti nella base di dati in un certo istante.

Nella definizione formale del modello relazionale è richiesto che le tuple siano tutte distinte.

## Restrizione di una tupla

La restrizione di un tupla sugli attributi A della relazione R (indicata con  $t[A]$ ), è data dalla lista dei valori assunti da t sugli attributi A

## Esempio

Sia  $t=(1, Rossi, v. Anemoni 5, L. 3.678.000)$  la tupla.

Allora una possibile restrizione è:  $t[Numero\_cc, Nome]=(1, Rossi)$

## Chiave di una relazione

E' un sottoinsieme K degli attributi che soddisfa le proprietà:

### ■ unicità

in qualunque istanza di R, non possono esistere due tuple distinte la cui restrizione su K sia uguale

### ■ minimalità

non è possibile sottrarre a K un attributo senza violare la condizione di unicità

In generale una relazione può avere più di una chiave

Esempio

Record	Nome	Indirizzo	Telefono
1	Rossi Carlo	Via dei Tigli, 32	02-33249187

Numero\_cc è una chiave

Nome non è una chiave

(Nome, Indirizzo) non è una chiave

## Chiave primaria

**Corrisponde alla chiave usata più frequentemente per accedere ai dati**

In genere si indica sottolineando gli attributi che la costituiscono:

Conto\_corrente (Numero\_cc, Nome, Indirizzo, saldo)

## Progettazione dei database



Le fasi di progettazione di un DB sono tre:

1 - Progetto CONCETTUALE (-> Schema concettuale)

2 - Progetto LOGICO (-> Schema logico)

3 - Progetto FISICO (-> Schema fisico)

### 1 - Progetto CONCETTUALE

Lo schema concettuale è la rappresentazione più astratta, la più vicina alla logica umana nella definizione di dati e relazioni. Spesso vengono usati i modelli entità-relazioni

### 2 - Progetto LOGICO

Lo schema logico dipende fortemente dal DBMS e dal suo modello logico dei dati. Esistono ad esempio DBMS gerarchici, reticolari e relazionali.

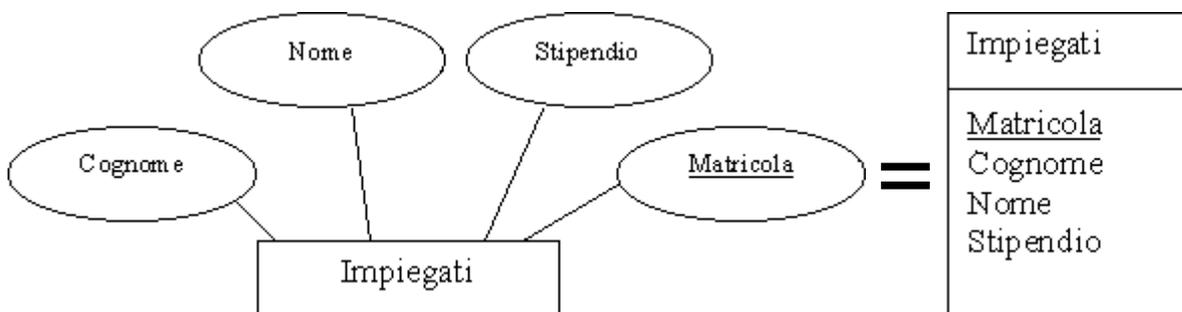
Nello schema logico vengono definite anche le viste (dette anche schemi esterni) cioè le parti del DB messe a disposizione delle applicazioni.

### 3 - Progetto FISICO

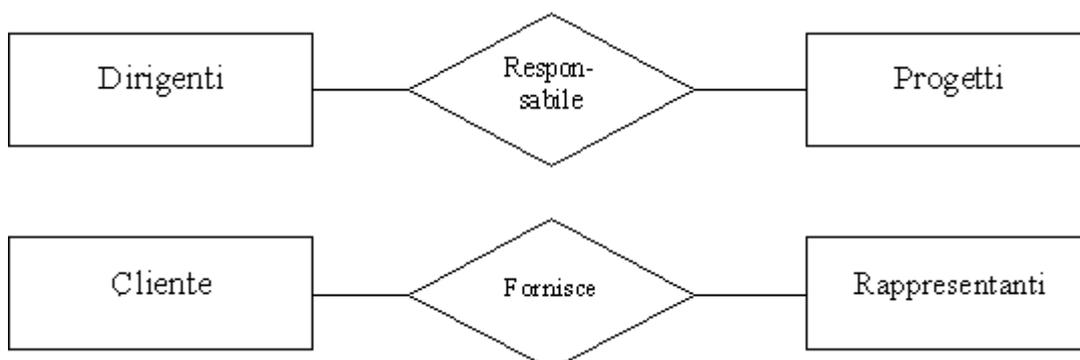
Lo schema fisico definisce come le strutture definite nel progetto logico vanno implementate nell'archivio e nel file system scelti.

## I diagrammi entità relazione (ERD)

- Entità** un qualsiasi oggetto concettuale che caratterizza l'applicazione in questione e che può essere individuato e distinto dagli altri
- Attributi** insieme di valori che caratterizzano un'entità
- Attributi chiave** insieme degli attributi sufficienti ad identificare univocamente un'entità all'interno di un certo insieme



- Relazioni** dipendenze o associazioni di interesse informativo tra le entità rappresentate

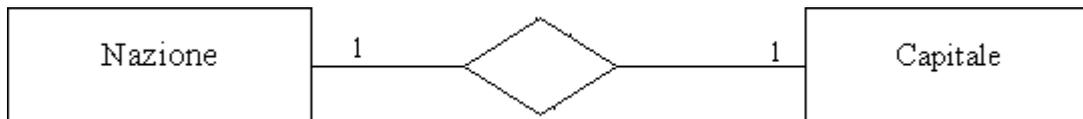


---

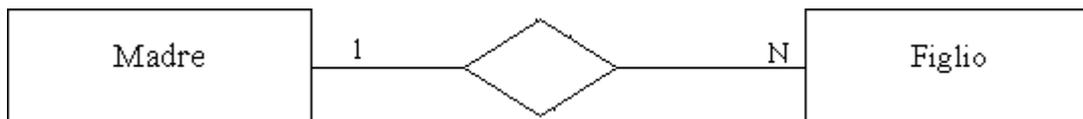
## Cardinalità delle relazioni

La relazione R che lega due entità E1 ed E2 può essere classificata in base alla sua cardinalità:

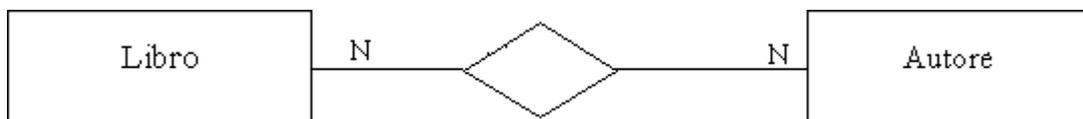
1. R ha cardinalità 1:1 (uno a uno) se ad un elemento di E1 può corrispondere un solo elemento di E2



2. R ha cardinalità 1:N (uno a molti) se ad ogni elemento di E1 possono corrispondere più elementi di E2



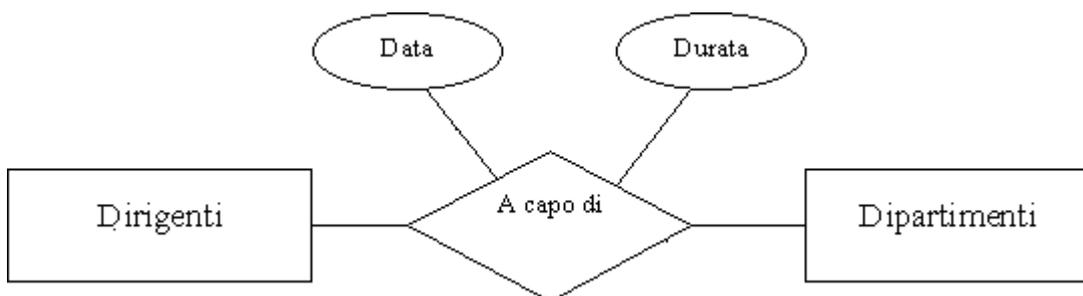
3. R ha cardinalità N:N (molti a molti) se ad ogni elemento di E1 possono corrispondere molti elementi di E2 e viceversa



---

## Attributi delle relazioni

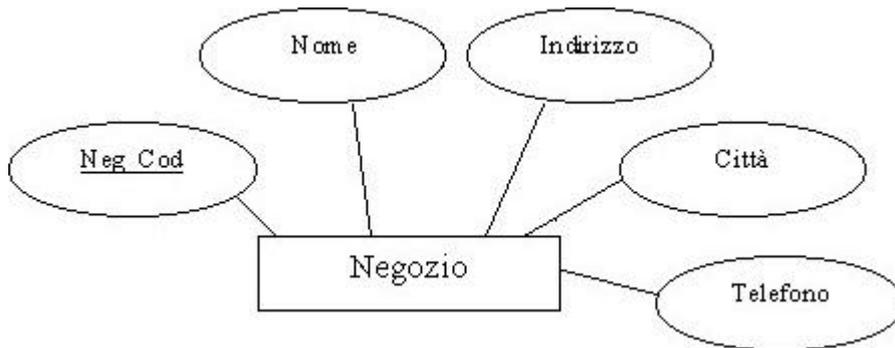
Le relazioni possono essere pensate come insiemi di elementi e quindi è possibile assegnare ad esse degli attributi



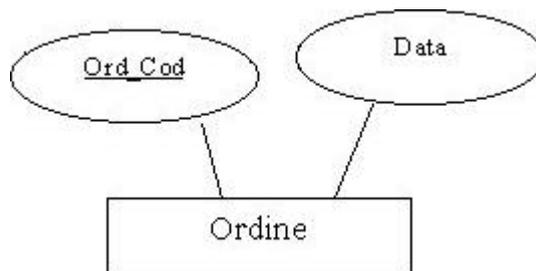
# Esempi di ERD

I seguenti diagrammi rappresentano una ipotetica schematizzazione delle entità Negozio, Ordine, Articolo e delle relazioni tra esse esistenti:

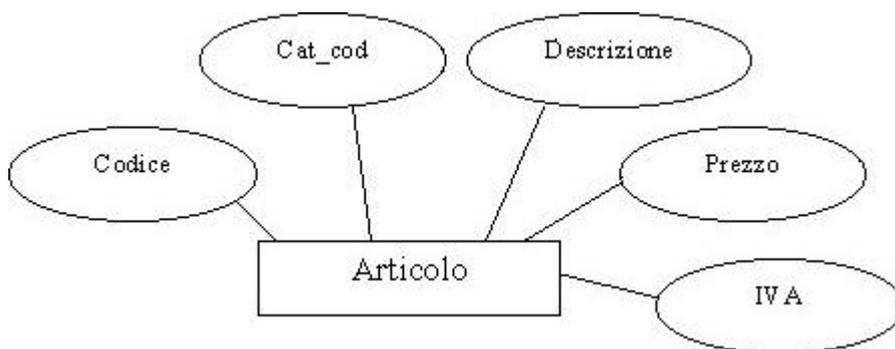
## Entità Negozio



## Entità Ordine



## Entità Articolo



## Relazioni tra le entità



N. protocollo	Uffici interessati				

NO

N. protocollo	Uffici interessati

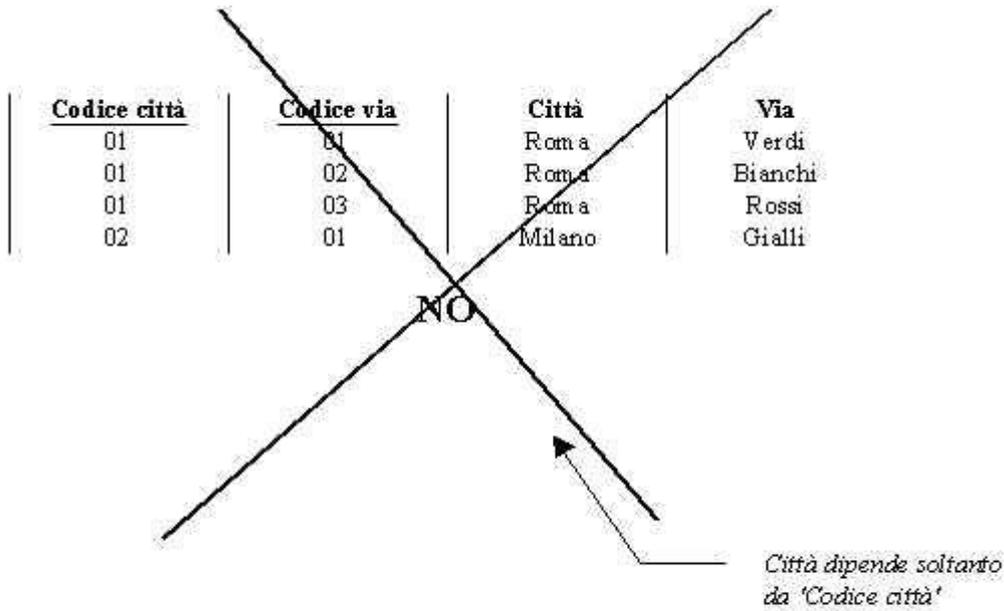
SI

---

## La seconda forma normale

**Condizione:** in una tabella con chiave a più attributi, ogni colonna non appartenente alla chiave deve dipendere dall'insieme delle colonne-chiave e non solo da una parte di esse.

**Normalizzazione:** Normalizzare vuol dire produrre tante tabelle che soddisfino la condizione nel caso in cui in quella originaria essa non sia soddisfatta



<u>Codice città</u>	<u>Codice via</u>	Via
01	01	Verdi
01	02	Bianchi
01	03	Rossi
02	01	Gialli

**SI**

<u>Codice città</u>	Città
01	Roma
01	Roma
01	Roma
02	Milano

**SI**

## La terza forma normale

**Condizione:** in una tabella la dipendenza fra le colonne deve essere basata soltanto sulla chiave primaria.

**Normalizzazione:** Normalizzare vuol dire produrre tante tabelle che soddisfino la condizione nel caso in cui in quella originaria essa non sia soddisfatta

N. protocollo	Mittente	Tipo	Urgenza
1	001	1	Si
2	001	2	No
3	003	1	Si
4	002	2	No
5	003	3	Si

NO

Tipo	Descrizione
1	Lettera
2	Memo
3	Telegramma

NO

*L'urgenza dipende dal tipo  
e non dal numero di  
protocollo*

<u>Tipo</u>	Descrizione	Urgenza
1	Lettera	Si
2	Memo	No
3	Telegramma	Si

SI

## Progetto concettuale

### Descrizione dell'archivio

Si vuole realizzare un archivio contenente degli **articoli**.

Ogni articolo è **pubblicato** su una **rivista**.

Ogni articolo è caratterizzato da un **titolo**, da alcune **parole chiave**, dalle **note**.

Ogni articolo **tratta di** un certo **argomento** (scelto da una apposito elenco).

L'articolo è solitamente pubblicato in una determinata **rubrica** della rivista ed in una data **pagina**.

Ogni rivista ha un **nome**, il **numero**, il **mese** e l'**anno** di pubblicazione.

## Identificazione elementi dell'ERD

Le **entità** sono in genere rappresentate da sostantivi. Nel nostro caso possiamo identificare (in rosso): articolo, rivista, elenco argomenti.

Gli **attributi** (delle entità) sono identificabili come proprietà delle entità (in verde).

Le **relazioni** sono verbi che collegano più entità.

Nel nostro caso:

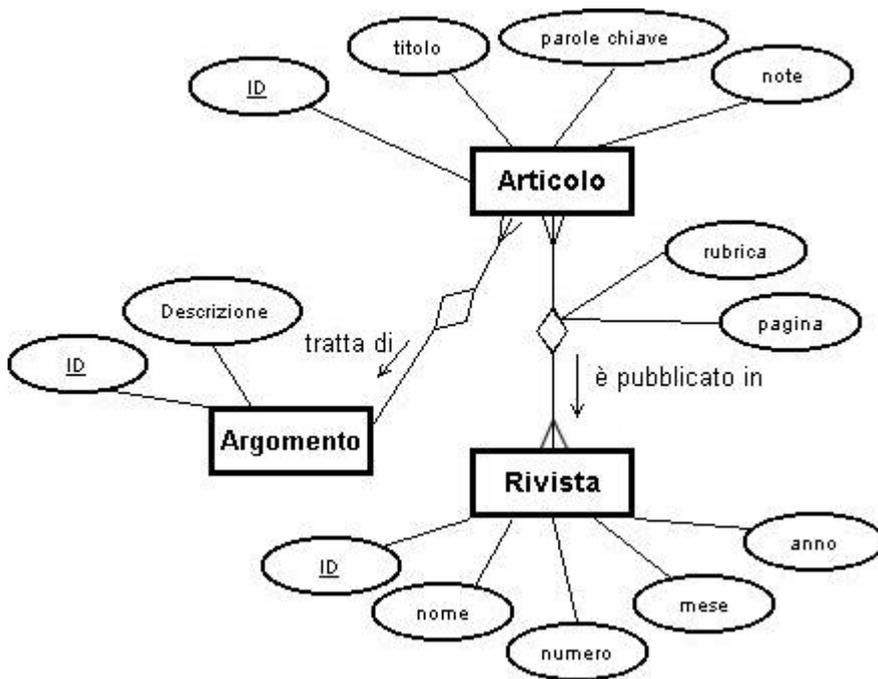
- Per l'entità **articolo** gli attributi sono: titolo, parole chiave, note.
- L'entità **argomento** è in relazione di tipo 'tratta\_di' con l'entità articolo; i suoi attributi sono: descrizione.
- L'entità **rivista** è in relazione di tipo 'è\_publicato' con l'entità articolo; i suoi attributi sono: nome, numero, mese, anno.  
Si noti che la relazione 'è\_publicato' è caratterizzata dagli attributi rubrica e pagina. Infatti l'articolo appartiene ad una rubrica in quanto pubblicato in una determinata pagina.

## Ulteriori affinamenti

Si potrebbe considerare Rubrica come un'entità

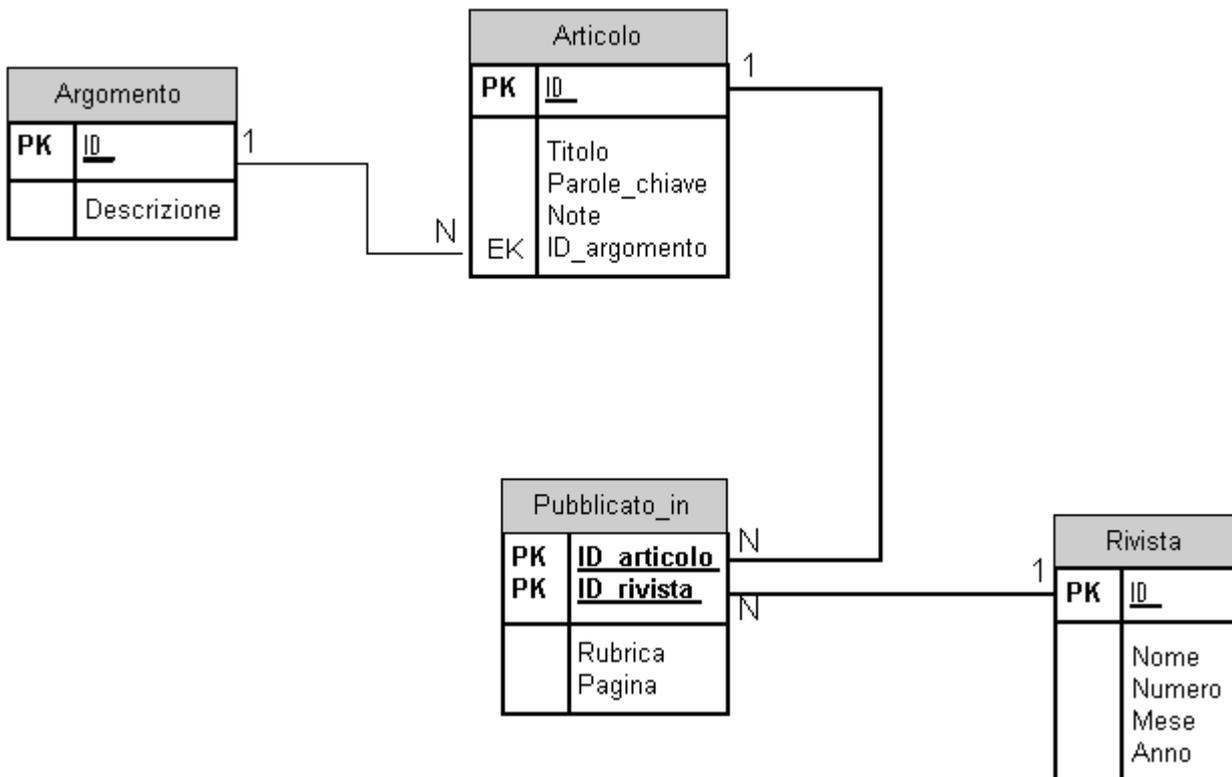
## ERD

Dalla descrizione prima vista si può trarre il seguente ERD:



## Progetto logico

Dall'ERD prima visto, si possono trarre le seguenti tabelle nelle quali sono state aggiunte le chiavi esterne necessarie alle relazioni:



## Tabelle

## Argomento

Campo	Tipo
ID	Contatore
Descrizione	Testo (25) (richiesto: si, Indice: No)

## Articolo

Campo	Tipo
ID	Contatore
Titolo	Testo (50) (richiesto: si, Indice: Si)
Parole chiave	Testo (50) (richiesto: no, Indice: No)
Note	Memo
ID_argomento	Intero lungo

## Rivista

Campo	Tipo
ID	Contatore
Nome	Testo (30) (richiesto: si, Indice: No)
Numero	Testo (5) (richiesto: si, Indice: No)
Mese	Intero (Valido se: ">=1 And <=12")
Anno	Intero (Valido se: ">=1980 And <=2050")

## Pubblicato\_in

Campo	Tipo
ID_articolo	Intero lungo
ID_rivista	Intero lungo
Rubrica	Testo (25) (richiesto: no, Indice: No)
Pagina	Intero

Si omette il progetto fisico.

# SQL e Access

Nella trattazione del linguaggio SQL si è dedicata particolare attenzione alla applicazione dei concetti mediante Microsoft Access.

E' disponibile il file Access ([esempi.mdb](#)) che contiene tutti gli esempi che verranno di seguito illustrati.

## Le relazioni

Nel modello relazionale, per relazione si intende una tabella caratterizzata da un numero fisso di colonne (attributi) ed un numero variabile di righe (tuple).

Ciascuna colonna assume valori tratti da uno stesso dominio.

In modo più formale si può così definire una relazione:

- Dati gli insiemi D1, D2, ..., Dn una relazione R tra questi n insiemi è un insieme di n-uple <d1, d2, ..., dn> dove d1 appartiene a D1, d2 appartiene a D2, ...
- D1, D2, ..., Dn sono detti domini, i loro nomi sono detti attributi, n è detto grado della relazione R.

## La creazione delle relazioni in SQL: CREATE TABLE

### Sintassi

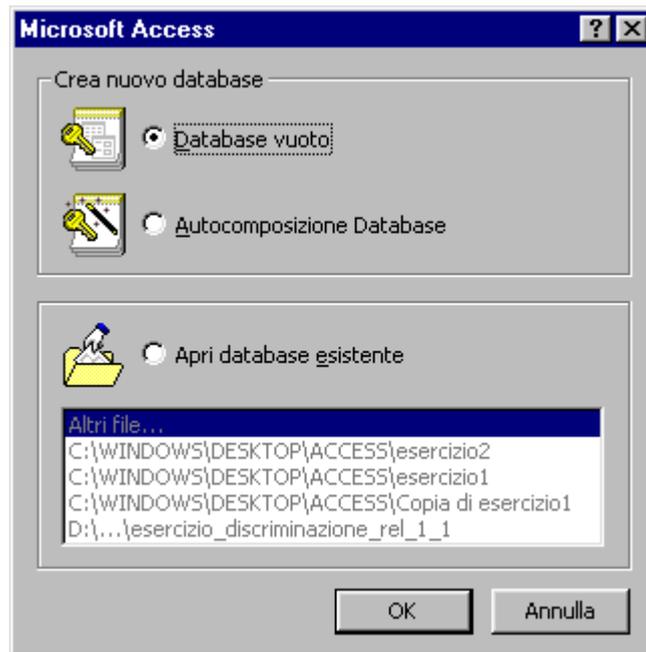
```
CREATE TABLE tabella (campo1 tipo [(dimensioni)] [NOT NULL] [indice1] [, campo2 tipo [(dimensioni)] [NOT NULL] [indice2] [, ...]])
```

### Esempio

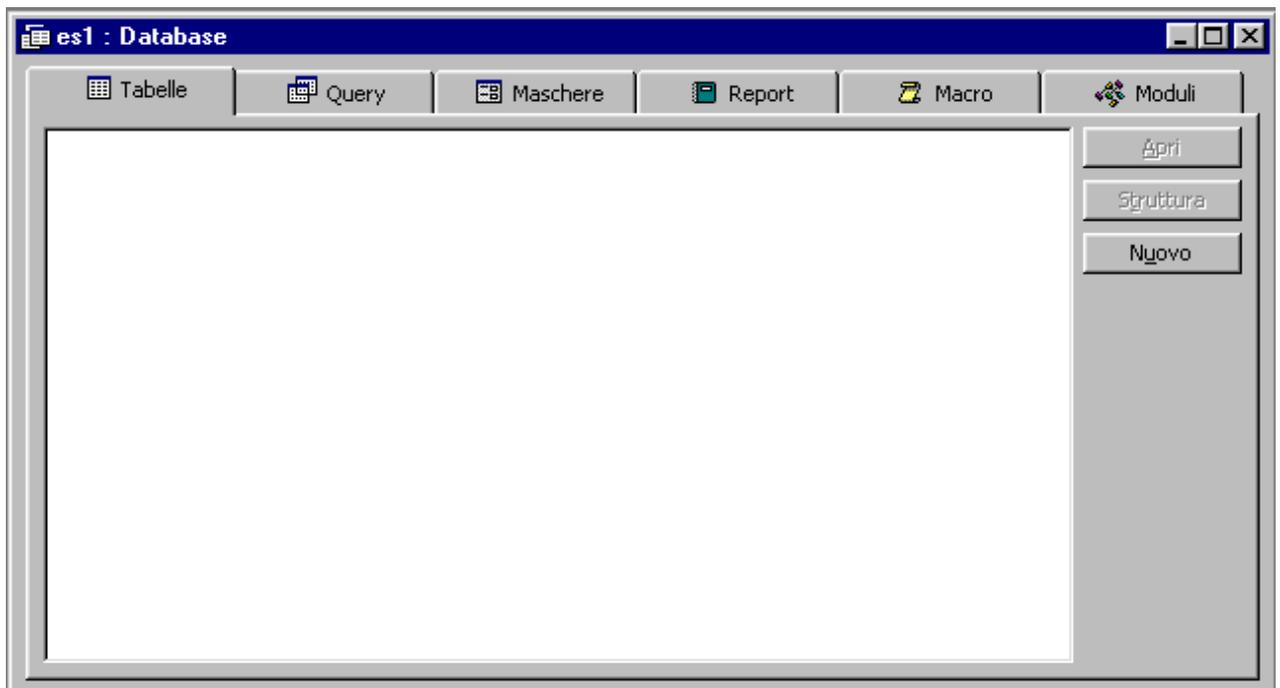
```
CREATE TABLE CONTO_CORRENTE  
(NUMERO_CC: INTEGER, PRIMARY KEY, NOT NULL,  
NAME: CHAR(20), NOT NULL,  
INDIRIZZO: CHAR (20),  
SALDO: INTEGER, NOT NULL)
```

## La creazione delle relazioni con Access

- Creare un nuovo database
  - Avviare Access
  - Scegliere database vuoto nella finestra:



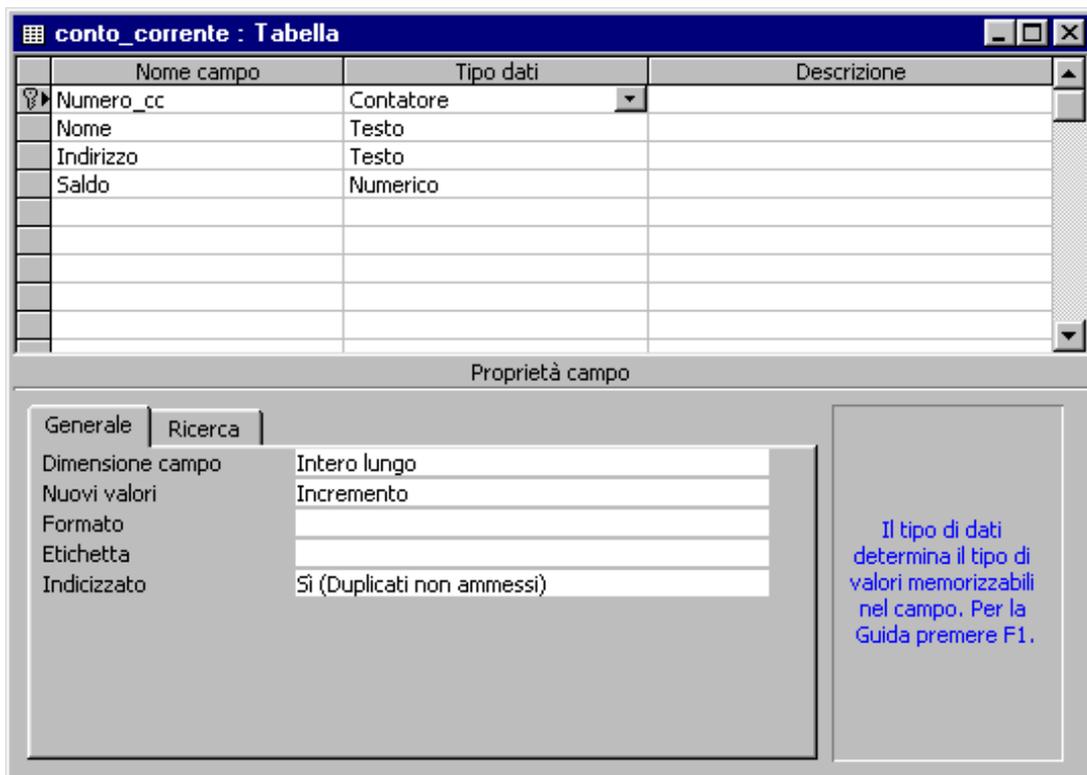
- Salvare il file  
Apparirà la finestra principale del db:



- Creare la tabella "conto\_corrente"
  - Scegliere nuovo nella finestra del database (cartella Tabelle)
  - Nella finestra nuova tabella scegliere Visualizzazione struttura



- Nella finestra struttura della tabella inserire le proprietà dei campi (vedi dopo):



- Dettaglio dei campi della tabella:

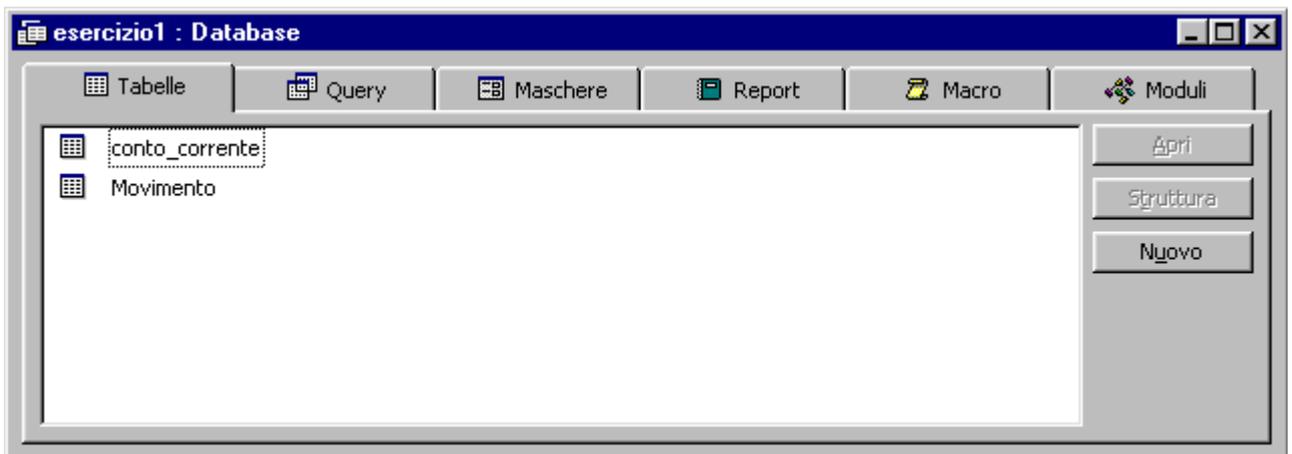
Nome	Tipo	Altre proprietà
Numero_cc	Contatore	Indicizzato: Sì (Duplicati non ammessi)
Nome	Testo	Dimensione campo: 50; richiesto: Sì; Consenti lunghezza zero: No; Indicizzato: Sì
Indirizzo	Testo	Dimensione campo: 50; richiesto: No; Consenti lunghezza zero: No; Indicizzato: No
Saldo	Numerico	Dimensione campo: intero lungo; Formato: valuta; Richiesto: No; Indicizzato: No

- Creazione della chiave primaria

1. Evidenziare la riga "Numero\_cc"
2. Dal menu "Modifica" scegliere "Chiave primaria"
3. Chiudere la finestra dal bottone di chiusura
4. Salvare la tabella con nome "conto\_corrente"

■ Inserire i dati nella tabella "conto\_corrente"

- Nella finestra del database, selezionare la tabella "conto\_corrente" e fare click sul bottone "Apri"



- Apparirà la tabella (inizialmente vuota) dove si potranno inserire i dati

	Numero_cc	Nome	Indirizzo	Saldo
▶	1	Rossi	v. Anemoni 5	L. 3.678.000
	2	Bianchi	v. Bolla 64	L. 664.000
	3	Brunelli	v. Po 41	L. 6.777.000
	4	Grandi	v. Romolo 3	L. 3.400.000
*	(Contatore)			L. 0

E' possibile passare dalla visualizzazione 'struttura' della tabella a quella 'foglio dati' mediante il bottone  e viceversa mediante .

## Indici in SQL

Un indice è una struttura dati che il DBMS associa alle tabelle per poter eseguire con maggiore efficienza la ricerca dei dati.

Ciascun indice è relativo ad una o più colonne di una tabella e permette di evitare l'analisi della intera tabella per effettuare una data ricerca.

## Creare un indice in SQL: CREATE INDEX ... ON ...

### Sintassi

```
CREATE [UNIQUE] INDEX indice  
ON tabella (campo [ASC|DESC][, campo [ASC|DESC], ...])
```

La parola riservata UNIQUE impedisce la presenza di duplicati nel campo indicizzato

### Esempio

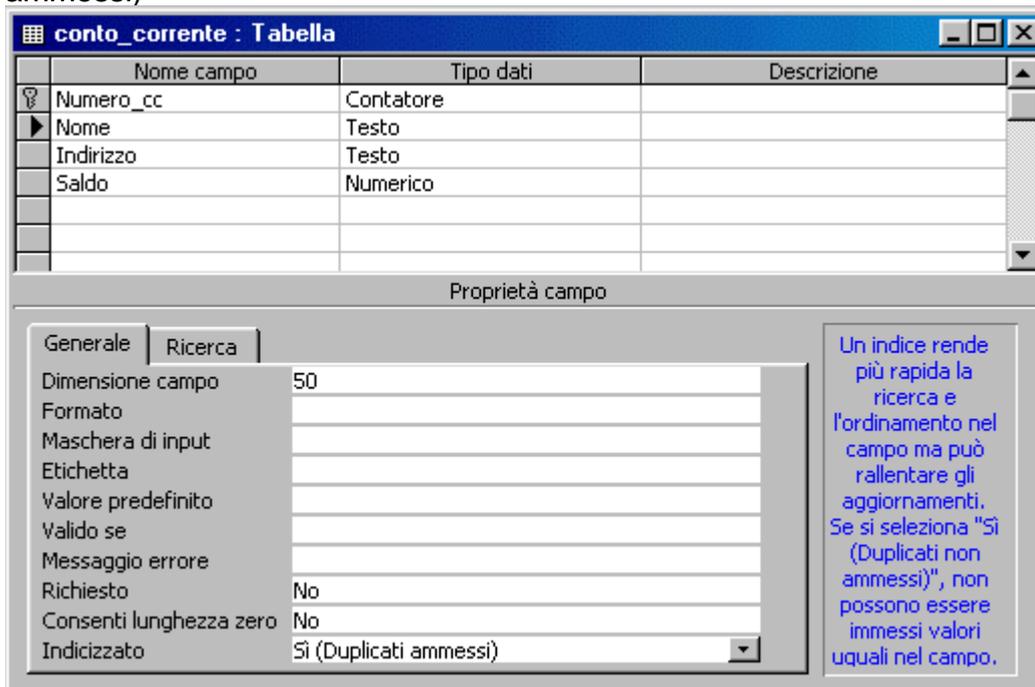
```
CREATE INDEX CONTO_CORRENTE_KEY ON CONTO_CORRENTE  
(NUMERO_CC)
```

## Indici in Access

### Indicizzare la relazione conto\_corrente in base all'attributo Nome

1. aprire la tabella conto\_corrente in modalità struttura
2. selezionare il campo Nome

3. Dal riquadro delle proprietà del campo, alla voce Indicizzato scegliere "Si (Duplicati ammessi)"



## Le operazioni di selezione

Data una tabella e una condizione logica definita sui suoi attributi, la selezione restituisce una tabella con gli stessi attributi di quella di partenza, ma con le sole tuple che soddisfano la condizione.

E' possibile scegliere quali attributi verranno restituiti dalla selezione.

## Le operazioni di selezione in SQL: SELECT

Seleziona dalle relazioni nomi\_relazioni le tuple degli attributi nomi\_attributi che soddisfano le condizioni condizioni\_ricerca

### Sintassi

```
SELECT nomi_attributi FROM nomi_relazioni WHERE  
condizioni_ricerca
```

## Esempio

```
SELECT SALDO FROM CONTO_CORRENTE WHERE NUMERO_CC =2
```

# Esempio 1 (SELECT)

## Descrizione

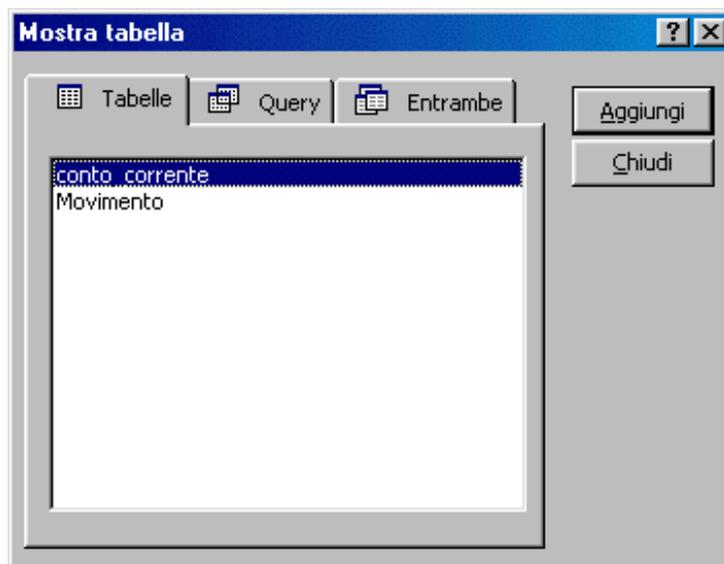
Estrarre nomi e indirizzi di tutti i correntisti

## La query da realizzare

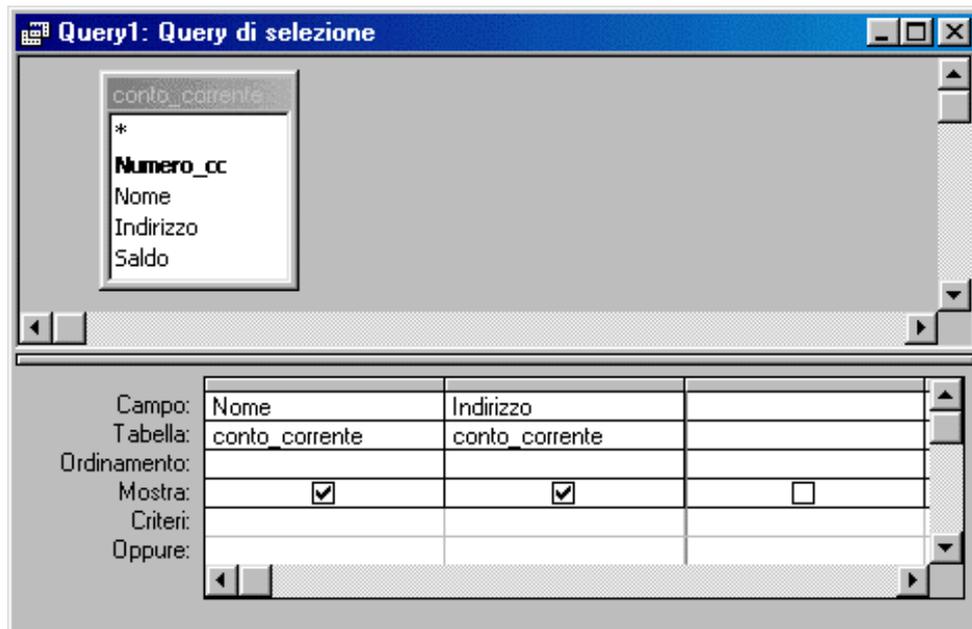
```
SELECT nome, indirizzo FROM conto_corrente
```

## La procedura in Access

- Dalla finestra del database selezionare la cartella Query, fare click su Nuovo e su visualizzazione Struttura.



- Selezionare la tabella conto\_corrente e poi il bottone Aggiungi.
- Nella finestra 'Query di selezione' scegliere nel rigo relativo al Campo: Nome e Indirizzo, come sotto illustrato.



## Risultato della query

Per eseguire la query fare click sul bottone .

La proiezione della relazione sui due attributi scelti sarà:

	Nome	Indirizzo
▶	Rossi	v. Anemoni 5
	Bianchi	v. Bolla 64
	Brunelli	v. Po 41
	Grandi	v. Romolo 3
*		

## Esempio 2 (WHERE)

### Descrizione

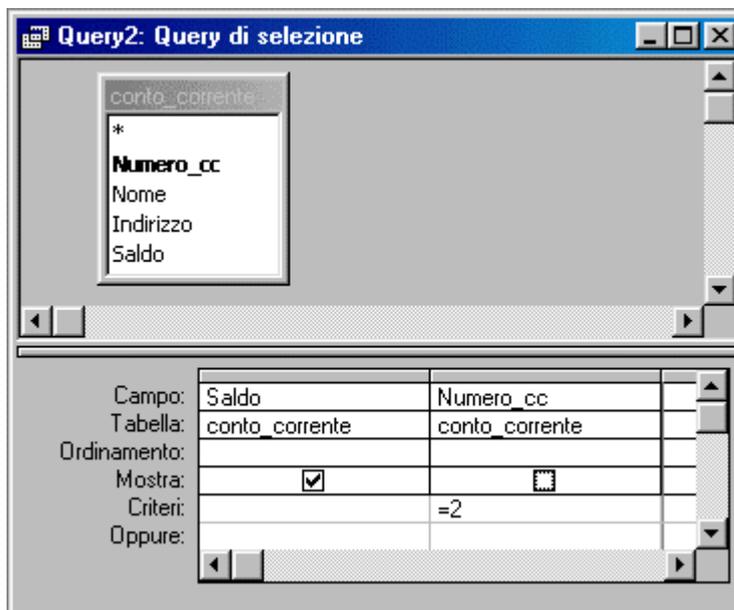
Estrarre il valore del saldo del conto corrente numero 2

### La query da realizzare

```
SELECT saldo FROM conto_corrente WHERE numero_cc =2
```

### La struttura della query in Access

Creare una nuova query scegliendo i campi saldo e numero\_cc. Alla voce Criteri per la colonna numero\_cc inserire =2 e disabilitare l'opzione Mostra (per non far apparire il campo nel risultato della query)



## Risultato della query

Eeguire la query. Il risultato sarà:



## Esempio 3 (connettivi logici in WHERE)

### Creazione della relazione Movimento

Per procedere alla esecuzione della query prevista in questo esempio sarà prima necessario creare una nuova tabella di nome "Movimento":



## Descrizione dei campi (attributi) della tabella (relazione):

Nome	Tipo	Altre proprietà
Numero_cc	Numerico	Dimensione campo: intero lungo; Richiesto: No; Indicizzato: No
Data_mov	Data/ora	Dimensione campo: data in cifre; Richiesto: No; Indicizzato: No
Numero_mov	Numerico	Dimensione campo: intero lungo; Richiesto: No; Indicizzato: No
Importo	Numerico	Dimensione campo: intero lungo; Formato: valuta; Richiesto: No; Indicizzato: No
Causale	Testo	Dimensione campo: 1; Richiesto: Si; Indicizzato: No

## Dati da inserire nella tabella

Movimento : Tabella					
	Numero_cc	Data_mov	Numero_mov	Importo	Causale
▶	1	14/01/99	1	L. 200.000	V
	1	27/01/99	1	L. 2.700.000	S
	4	27/01/99	1	L. 1.850.000	S
	3	25/01/99	1	-L. 650.000	A
	1	14/01/99	2	-L. 500.000	P
*	0		0	L. 0	

Record: 1 di 5

## Descrizione della query

Estrarre nome e indirizzo dei correntisti che hanno un movimento in data 27-1-99

## La query da realizzare

```
SELECT nome, indirizzo FROM conto_corrente, movimento
WHERE data_mov = 27-1-99 AND conto_corrente.numero_cc =
movimento.numero_cc
```

## La procedura da seguire in Access

Creare una nuova query, aggiungere le tabelle conto\_corrente e movimento. Impostare il criterio di selezione sulla data.

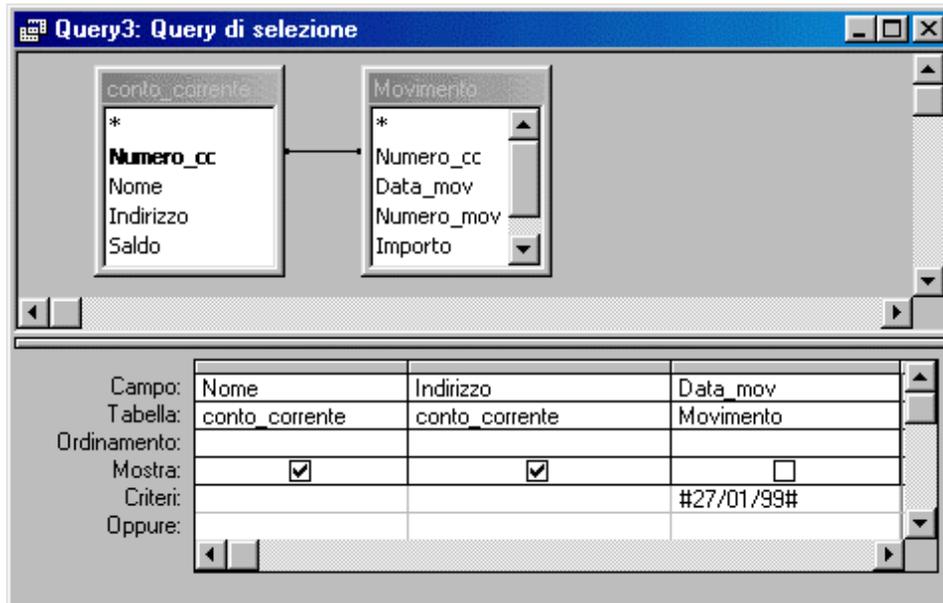
La condizione SQL "conto\_corrente.numero\_cc = movimento.numero\_cc" viene automaticamente interpretata da Access come un Join.

## JOIN

L'operazione di JOIN combina due relazioni in una sola concatenando le tuple che soddisfano la condizione di Join.

La relazione risultante ha grado pari alla somma dei gradi delle relazioni di partenza

## La struttura della query in Access



## Risultato della query

	Nome	Indirizzo
	Rossi	v. Anemoni 5
	Grandi	v. Romolo 3

## Il codice SQL generato

Scegliendo dal menu Visualizza l'opzione 'Visualizzazione SQL' si potrà esaminare il codice SQL prodotto dall'interazione grafica con il programma:

```
SELECT conto_corrente.Nome, conto_corrente.Indirizzo
FROM conto_corrente INNER JOIN Movimento ON
conto_corrente.Numero_cc = Movimento.Numero_cc
WHERE (((Movimento.Data_mov)=#1/27/99#));
```

## La clausola INNER JOIN

Access ha interpretato la corrispondenza dei numeri di conto corrente come una relazione di JOIN.

I join interni combinano i record da due tabelle ogni volta che in un campo comune ad entrambe le tabelle vengono individuati valori corrispondenti.

Esempio: è possibile utilizzare l'operazione INNER JOIN con le tabelle Reparti e Impiegati per selezionare tutti gli impiegati di ciascun reparto. Al contrario, per selezionare tutti i reparti, anche se alcuni non hanno impiegati, oppure per selezionare tutti gli impiegati,

anche se alcuni non sono assegnati ad un reparto, è possibile utilizzare l'operazione LEFT JOIN o RIGHT JOIN per creare un join esterno.

## Esempio 4 (connettivi logici in WHERE)

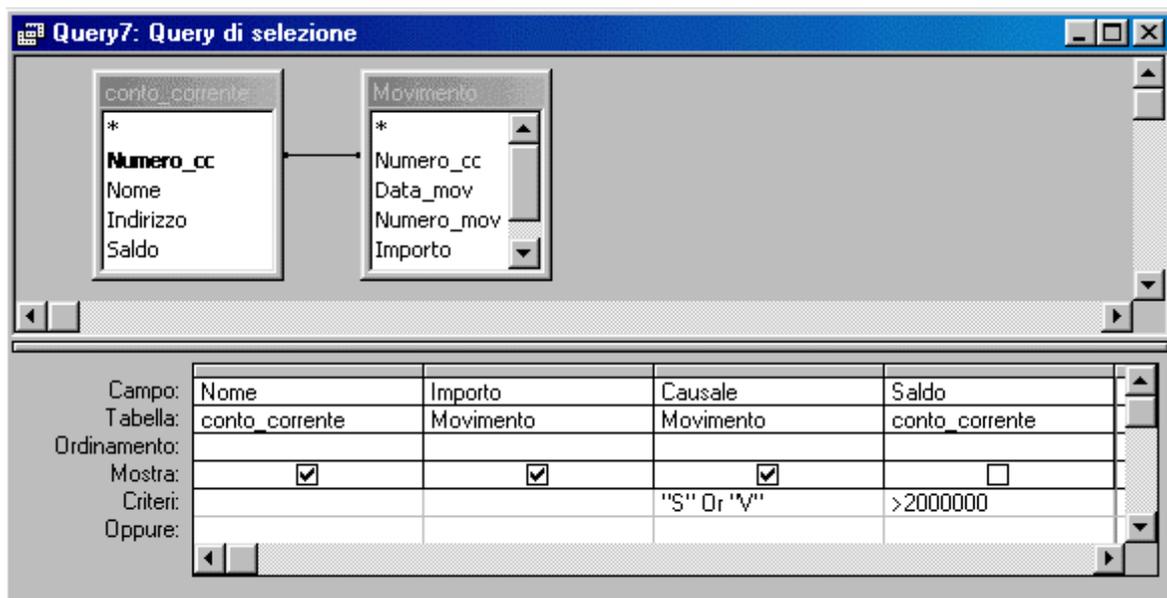
### Descrizione della query

Estrarre nome del correntista, importo e causale dei movimenti di tutti i correntisti con un saldo maggiore di due milioni e con un movimento che sia un versamento o un accredito di stipendio

### La query da realizzare è:

```
SELECT nome, importo, causale FROM movimento,
conto_corrente WHERE saldo>2.000.000 AND (causale=V OR
causale =S) AND conto_corrente.numero_cc =
movimento.numero_cc
```

### La struttura della query in Access



### Il risultato della query

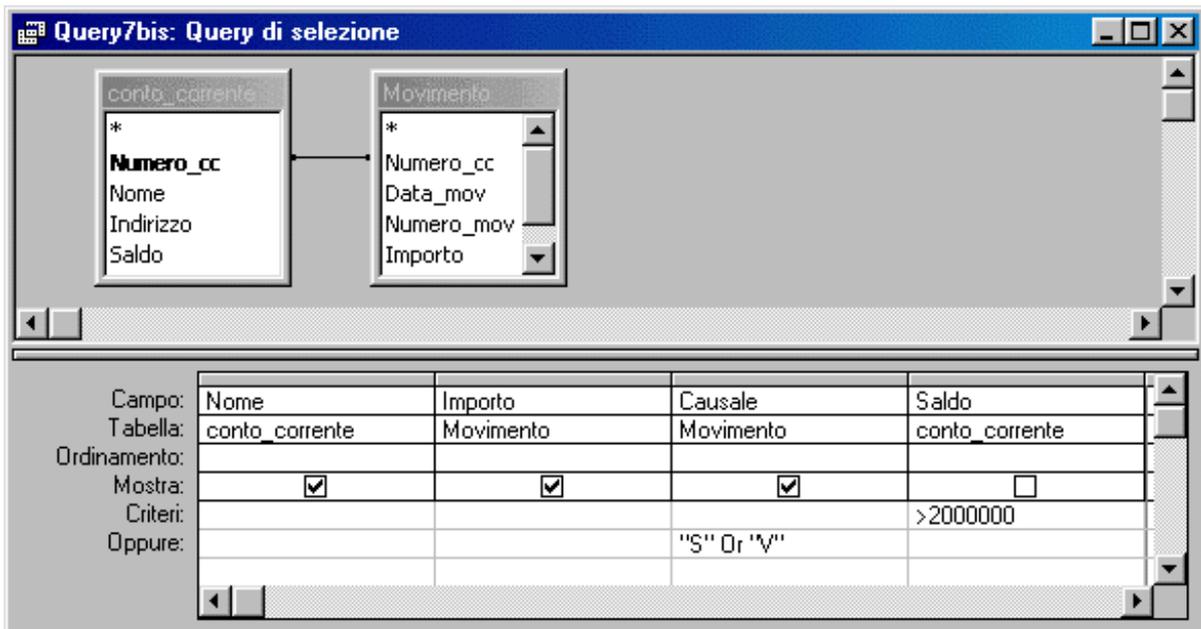
	Nome	Importo	Causale
▶	Rossi	L. 200.000	V
▶	Rossi	L. 2.700.000	S
▶	Grandi	L. 1.850.000	S
*			

Record: 1 di 3

## Il codice SQL generato

```
SELECT conto_corrente.Nome, Movimento.Importo,
Movimento.Causale
FROM conto_corrente INNER JOIN Movimento ON
conto_corrente.Numero_cc = Movimento.Numero_cc
WHERE (((Movimento.Causale)="S" Or
(Movimento.Causale)="V") AND
((conto_corrente.Saldo)>2000000));
```

## Si noti la differenza con la seguente interrogazione



## Il risultato della query

The screenshot shows a query editor window titled "Query7bis: Query di selezione" displaying the result of a query. The result is a table with the following columns: Nome, Importo, Causale. The data is as follows:

Nome	Importo	Causale
Rossi	L. 200.000	V
Rossi	-L. 500.000	P
Rossi	L. 2.700.000	S
Grandi	L. 1.850.000	S
Brunelli	-L. 650.000	A
*		

Record: 1 di 5

## Il codice SQL generato

```
SELECT conto_corrente.Nome, Movimento.Importo,
Movimento.Causale
FROM conto_corrente INNER JOIN Movimento ON
conto_corrente.Numero_cc = Movimento.Numero_cc
```

```
WHERE (((conto_corrente.Saldo)>2000000)) OR
(((Movimento.Causale)="S" Or (Movimento.Causale)="V"));
```

## Esempio 5 (operazioni in SELECT)

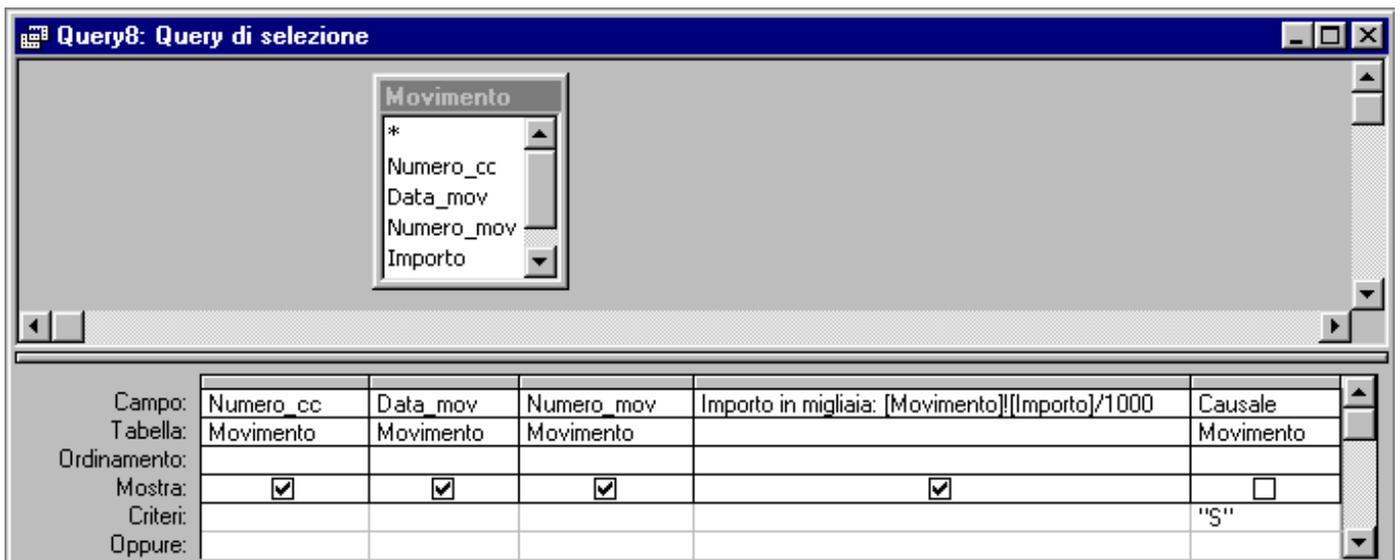
### Descrizione della query

Estrarre numero del conto, data, numero di movimento ed importo in migliaia di lire dei movimenti dovuti ad un accredito di stipendio

### La query da realizzare

```
SELECT numero_cc, data_mov, numero_mov, importo/1000
FROM movimento
WHERE causale =S
```

### La struttura della query in Access



Il campo "Importo in migliaia" è stato introdotto con il generatore di espressioni accessibile mediante il bottone .

### Il risultato della query

Numero_cc	Data_mov	Numero_mov	Importo in migliaia
1	27/01/99	1	2700
4	27/01/99	1	1850
(Contatore)			

Record: 3 di 3

## Il codice SQL generato

```
SELECT Movimento.Numero_cc, Movimento.Data_mov,  
Movimento.Numero_mov, [Movimento]![Importo]/1000 AS  
[Importo in migliaia]  
FROM Movimento  
WHERE (((Movimento.Causale)="S"));
```

## La clausola ORDER BY

Ordina i record risultanti da una query in base a uno o più campi specificati in ordine crescente o decrescente.

### Sintassi

```
SELECT elencocampi  
FROM tabella  
WHERE criteriselezione  
[ORDER BY campo1 [ASC | DESC ][, campo2 [ASC | DESC ]][,  
...]]]
```

## Esempio 6 (ORDER)

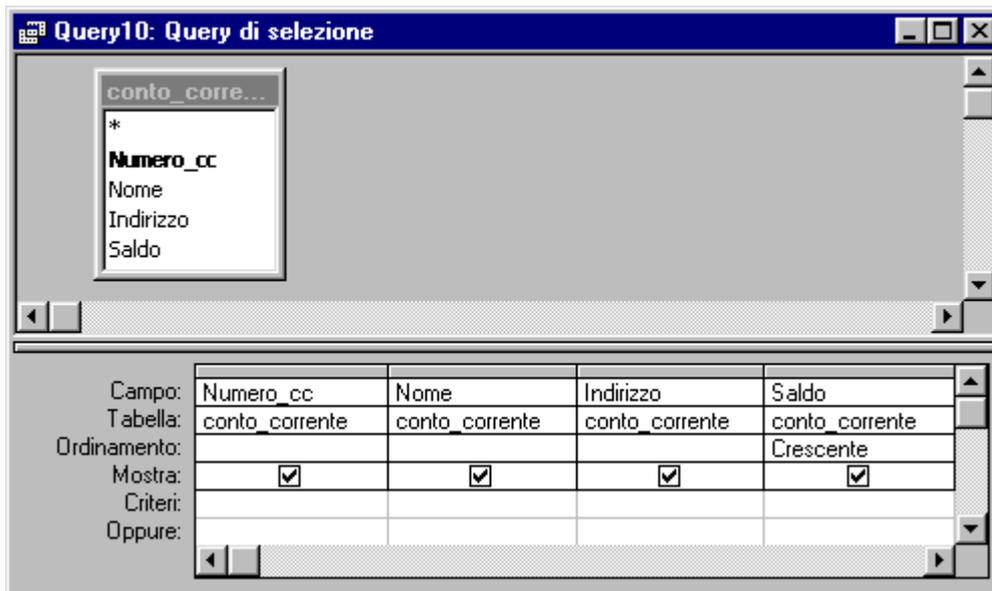
### Descrizione della query

Estrarre l'intera relazione conto\_corrente in cui i conti siano ordinati per valori crescenti del saldo

### La query da realizzare

```
SELECT *  
FROM conto_corrente  
ORDER BY saldo
```

### La struttura della query in Access



## Il risultato della query

Numero_cc	Nome	Indirizzo	Saldo
2	Bianchi	v. Bolla 64	L. 664.000
4	Grandi	v. Romolo 3	L. 3.400.000
1	Rossi	v. Anemoni 5	L. 3.678.000
3	Brunelli	v. Po 41	L. 6.777.500
(Contatore)			L. 0

Record: 5 di 5

## Il codice SQL generato

```
SELECT conto_corrente.Numero_cc, conto_corrente.Nome,
conto_corrente.Indirizzo, conto_corrente.Saldo
FROM conto_corrente
ORDER BY conto_corrente.Saldo;
```

### Si noti che:

Nella finestra della struttura della query si può impostare l'ordinamento in modo crescente o decrescente per ciascun campo

## La clausola GROUP BY

Raggruppa tutte le tuple che assumono lo stesso valore su alcuni attributi (detti attributi di raggruppamento) costituendo una partizione della relazione in classi di equivalenza.

## Sintassi

```
SELECT elencocampi
FROM tabella
WHERE criteri
GROUP BY elencocampiraggruppamento
```

## Le funzioni aggregatrici

Su una partizione si possono applicare delle funzioni aggregatrici dei valori numerici:

- MIN** (calcola il minimo valore dell'attributo cui è applicata per ciascuna delle classi di equivalenza)
- MAX** (calcola il massimo valore dell'attributo cui è applicata per ciascuna delle classi di equivalenza)
- SUM** (calcola la somma dei valori dell'attributo cui è applicata per ciascuna delle classi di equivalenza)
- AVG** (calcola la media dei valori dell'attributo cui è applicata per ciascuna delle classi di equivalenza)
- COUNT** (conta il numero di tuple che costituiscono ciascuna delle classi di equivalenza)

## La clausola HAVING

Permette di specificare condizioni sulle classi di equivalenza create dalla clausola GROUP BY.

## Sintassi

```
SELECT elencocampi
FROM tabella
WHERE criteriselezione
GROUP BY elencocampiraggruppamento
HAVING criteriraggruppamento
```

## Esempio 7 (GROUP, HAVING)

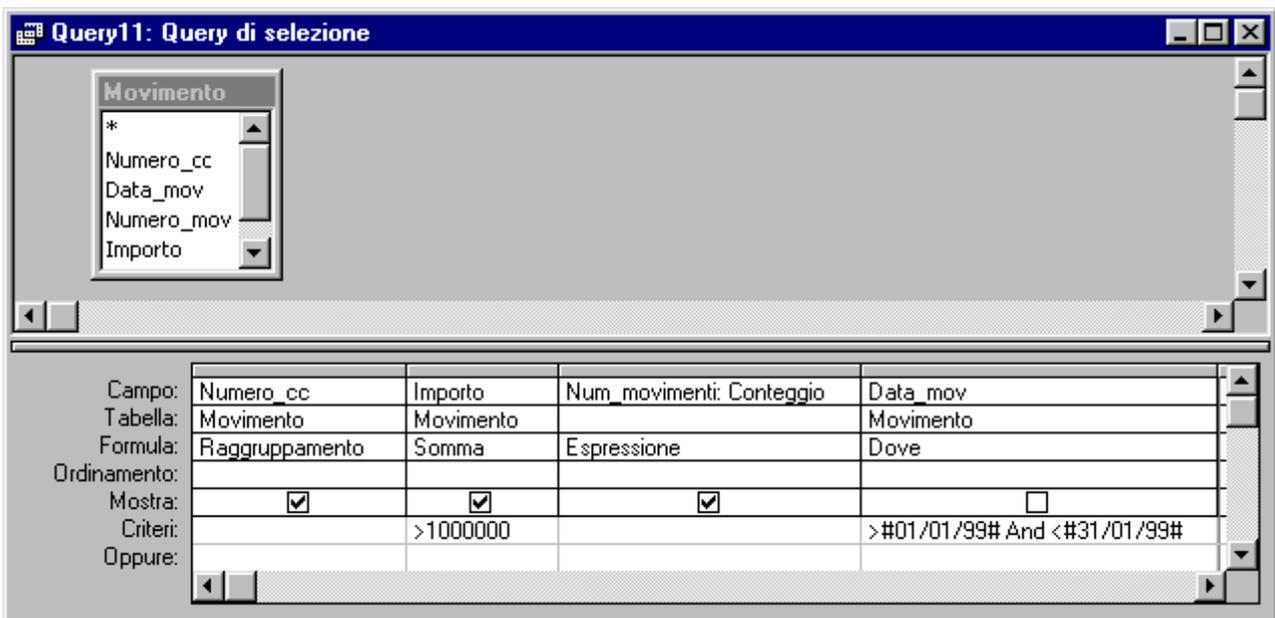
### Descrizione della query

Estrarre numero corrente, soma degli importi e numero dei movimenti di tutti i conti correnti i cui movimenti abbiano superato la cifra complessiva di un milione nel corso del gennaio 1999

## La query da realizzare

```
SELECT numero_cc, SUM(importo), count(*)
FROM movimento
WHERE data_mov > 1-1-99 AND data_mov < 31-1-99
GROUP BY numero_cc
HAVING SUM (Importo) > 1.000.000
```

## La struttura della query in Access



Per far apparire la riga relativa al campo Formula fare click con il pulsante destro sulla griglia della finestra e scegliere Totali dal menu.

## Il risultato della query

Numero_cc	SommaDiImporto	Num_movimenti
1	2400000	3
4	1850000	1

Record: 2 di 2

## Il codice SQL generato

```
SELECT Movimento.Numero_cc, Sum(Movimento.Importo) AS
SommaDiImporto, Count(*) AS Num_movimenti
FROM Movimento
```

```
WHERE (((Movimento.Data_mov)>#1/1/99# And
(Movimento.Data_mov)<#1/31/99#))
GROUP BY Movimento.Numero_cc
HAVING (((Sum(Movimento.Importo))>1000000));
```

## Annidamento di blocchi di istruzioni SQL

Il connettore IN permette di annidare blocchi di istruzioni SQL.

Nell'esempio seguente, vi sono due blocchi di istruzioni connessi mediante IN.

### Sintassi

```
SELECT nome, indirizzo
FROM conto_corrente
WHERE numero_cc IN
SELECT numero_cc
FROM movimento
WHERE data_mov=27-1-99
```

### Esecuzione

1. viene eseguito il secondo blocco (che restituisce una serie di numeri di conto)
2. viene eseguito il primo blocco (che seleziona gli attributi nome e indirizzo dalle tuple il cui numero di cc è stato selezionato nel secondo blocco)

## Esempio 8 (annidamento di blocchi)

### Descrizione della query

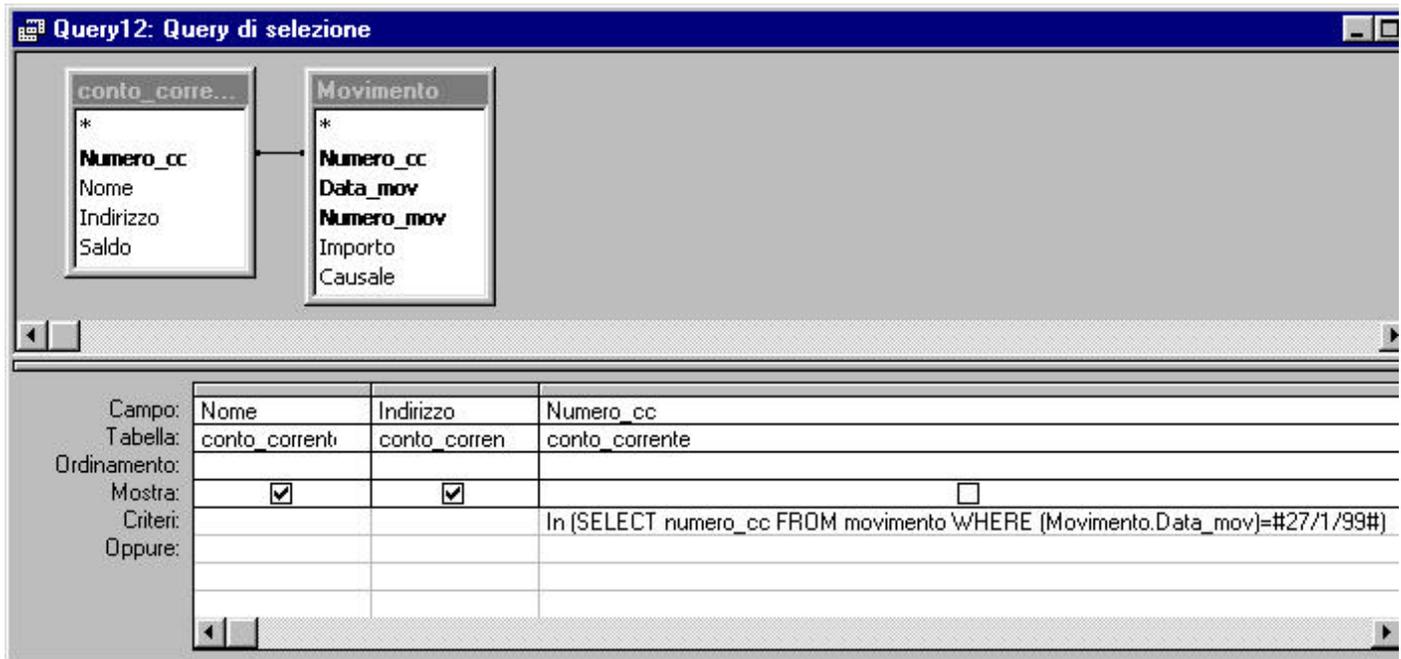
Estrarre nome e indirizzo dei correntisti che hanno un movimento in data 27-1-99

### La query da realizzare

```
SELECT DISTINCT nome, indirizzo
FROM conto_corrente
WHERE numero_cc IN
SELECT numero_cc
FROM movimento
WHERE data_mov=27-1-99
```

Il predicato DISTINCT esclude i record che contengono dati duplicati nei campi selezionati.

### La struttura della query in Access



## Il risultato della query

	Nome	Indirizzo	Data_mov
▶	Rossi	v. Anemoni 5	14/01/99
▶	Rossi	v. Anemoni 5	14/01/99
▶	Rossi	v. Anemoni 5	27/01/99
▶	Grandi	v. Romolo 3	27/01/99
*			

## Il codice SQL generato

```
SELECT DISTINCT conto_corrente.Nome,
conto_corrente.Indirizzo
FROM conto_corrente AS conto_corrente_1, Movimento AS
Movimento_1, conto_corrente INNER JOIN Movimento ON
conto_corrente.Numero_cc = Movimento.Numero_cc
WHERE (((conto_corrente.Numero_cc) In (SELECT numero_cc
FROM movimento WHERE (Movimento.Data_mov)=#27/1/99#)));
```

# INSERT

Inserisce in una relazione le tuple specificate

## Sintassi

```
INSERT INTO Relazione VALUES (<tupla>, <tupla>, ...)
```

# Esempio 9 (INSERT)

## Descrizione della query

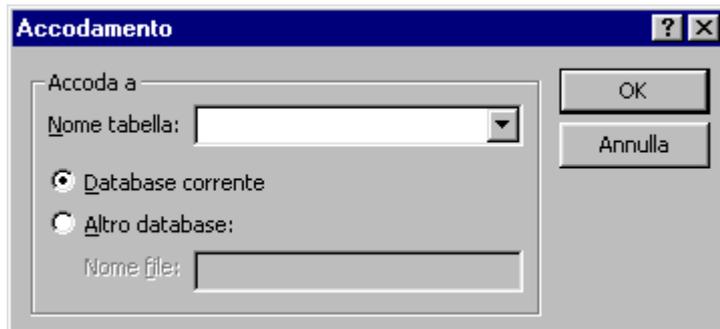
Inserire un nuovo correntista nella tabella conto\_corrente

## La query da realizzare

```
INSERT INTO conto_corrente
VALUES (<5, Cannetti, v. Tirolo 5, 100.000);
```

## La struttura della query in Access

Si tratta di una query di accodamento. Dopo aver creato la nuova query, scegliere "Query di accodamento" dal menu "Query". Nella finestra seguente scegliere conto\_corrente dal menu a tendina 'Nome tabella'.



Passare alla visualizzazione SQL della query.

Introdurre il seguente codice:

```
INSERT INTO conto_corrente
VALUES ('5', 'Cannetti', 'v. Tirolo 5', '100.000');
```

## Il risultato della query

Tabella conto\_corrente prima dell'esecuzione della query:

	Numero_cc	Nome	Indirizzo	Saldo
	1	Rossi	v. Anemoni 5	L. 3.678.000
	2	Bianchi	v. Bolla 64	L. 664.000
	3	Brunelli	v. Po 41	L. 6.777.500
	4	Grandi	v. Romolo 3	L. 3.400.000
	(Contatore)			

Record: 5 di 5

Tabella conto\_corrente dopo l'esecuzione della query (si noti che la query va eseguita con il bottone ):



Numero_cc	Nome	Indirizzo	Saldo
1	Rossi	v. Anemoni 5	L. 3.678.000
2	Bianchi	v. Bolla 64	L. 664.000
3	Brunelli	v. Po 41	L. 6.777.500
4	Grandi	v. Romolo 3	L. 3.400.000
5	Cannetti	v. Tirolo 5	L. 100.000
	(Contatore)		L. 0

## Il codice SQL generato

```
INSERT INTO conto_corrente  
VALUES ('5', 'Cannetti', 'v. Tirolo 5', '100.000');
```

## DELETE

Cancella da una relazione le tuple che soddisfano un predicato di selezione

## Sintassi

```
DELETE FROM Relazione WHERE Predicato
```

## Esempio 10 (DELETE)

### Descrizione della query

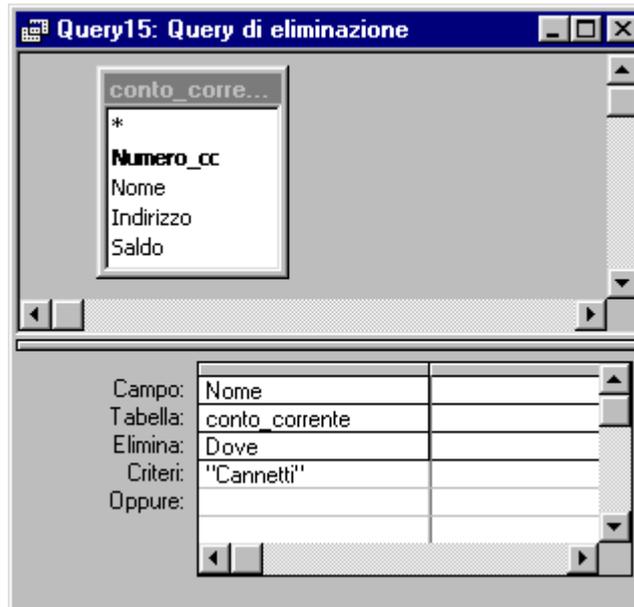
Eliminare il conto intestato a Cannetti

### La query da realizzare

```
DELETE FROM conto_corrente  
WHERE nome = Cannetti
```

### La struttura della query in Access

Si tratta di una query di eliminazione. Dopo aver creato la nuova query, scegliere "Query di eliminazione" dal menu "Query".



## Il risultato della query

Tabella conto\_corrente prima dell'esecuzione della query:

Numero_cc	Nome	Indirizzo	Saldo
1	Rossi	v. Anemoni 5	L. 3.678.000
2	Bianchi	v. Bolla 64	L. 664.000
3	Brunelli	v. Po 41	L. 6.777.500
4	Grandi	v. Romolo 3	L. 3.400.000
5	Cannetti	v. Tirolo 5	L. 100.000
(Contatore)			L. 0

Record: 6 di 6

Tabella conto\_corrente dopo l'esecuzione della query (si noti che la query va eseguita con il bottone ):

Numero_cc	Nome	Indirizzo	Saldo
1	Rossi	v. Anemoni 5	L. 3.678.000
2	Bianchi	v. Bolla 64	L. 664.000
3	Brunelli	v. Po 41	L. 6.777.500
4	Grandi	v. Romolo 3	L. 3.400.000
(Contatore)			

Record: 5 di 5

## Il codice SQL generato

```
DELETE conto_corrente.Nome  
FROM conto_corrente  
WHERE (((conto_corrente.Nome)="Cannetti"));
```

## UPDATE

Modifica il contenuto di alcuni attributi di una relazione per tutte le tuple che soddisfano una certa condizione

### Sintassi

```
UPDATE Relazione  
SET Attributo = Espressione  
WHERE Predicato
```

## Esempio 11 (UPDATE)

### Descrizione della query

Aumentare del 10% il saldo del conto del sig. Bianchi

### La query da realizzare

```
UPDATE conto_corrente  
SET saldo = saldo *1,1  
WHERE nome = Bianchi
```

### La struttura della query in Access

Si tratta di una query di aggiornamento. Dopo aver creato una nuova query scegliere "Query di aggiornamento" dal menu "Query"



*Il valore della riga 'Aggiorna a' per la colonna del campo Saldo è stato introdotto con il generatore di espressioni*

## Il risultato della query

Tabella conto\_corrente prima dell'esecuzione della query:

Numero_cc	Nome	Indirizzo	Saldo
1	Rossi	v. Anemoni 5	L. 3.678.000
2	Bianchi	v. Bolla 64	L. 664.000
3	Brunelli	v. Po 41	L. 6.777.500
4	Grandi	v. Romolo 3	L. 3.400.000
(Contatore)			

Tabella conto\_corrente dopo l'esecuzione della query (si noti che la query va eseguita con il bottone ):

Numero_cc	Nome	Indirizzo	Saldo
1	Rossi	v. Anemoni 5	L. 3.678.000
2	Bianchi	v. Bolla 64	L. 730.400
3	Brunelli	v. Po 41	L. 6.777.500
4	Grandi	v. Romolo 3	L. 3.400.000
*	(Contatore)		L. 0

## Il codice SQL generato

```
UPDATE conto_corrente SET conto_corrente.Saldo =
[conto_corrente]![Saldo]*1.1
WHERE (((conto_corrente.Nome)="Bianchi"));
```

## Esempio 11 (UPDATE)

### Descrizione della query

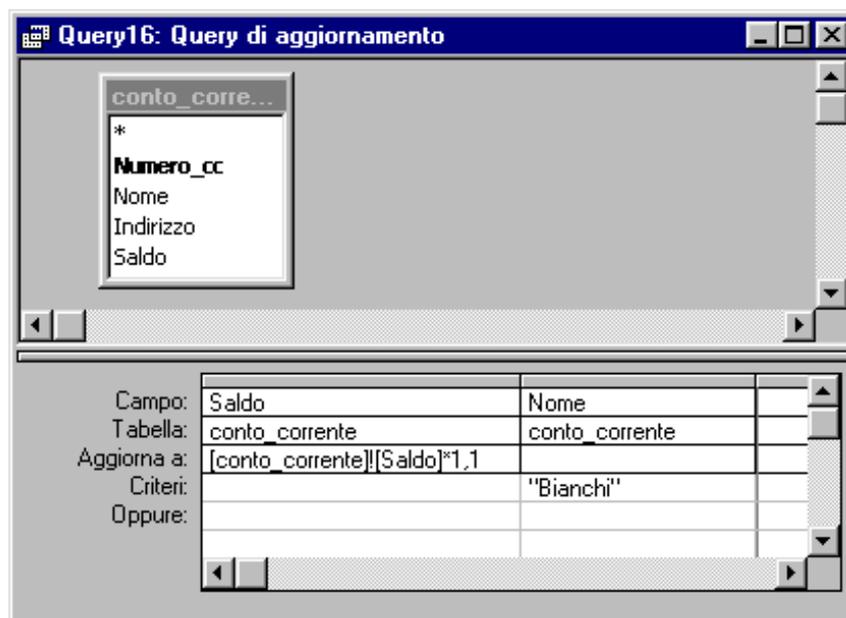
Aumentare del 10% il saldo del conto del sig. Bianchi

### La query da realizzare

```
UPDATE conto_corrente
SET saldo = saldo *1,1
WHERE nome = Bianchi
```

### La struttura della query in Access

Si tratta di una query di aggiornamento. Dopo aver creato una nuova query scegliere "Query di aggiornamento" dal menu "Query"



*Il valore della riga 'Aggiorna a' per la colonna del campo Saldo è stato introdotto con il generatore di espressioni*

### Il risultato della query

Tabella conto\_corrente prima dell'esecuzione della query:

conto_corrente : Tabella				
	Numero_cc	Nome	Indirizzo	Saldo
	1	Rossi	v. Anemoni 5	L. 3.678.000
	2	Bianchi	v. Bolla 64	L. 664.000
	3	Brunelli	v. Po 41	L. 6.777.500
	4	Grandi	v. Romolo 3	L. 3.400.000
▶		(Contatore)		

Record:  di 5

Tabella conto\_corrente dopo l'esecuzione della query (si noti che la query va eseguita con il bottone ):

conto_corrente : Tabella				
	Numero_cc	Nome	Indirizzo	Saldo
▶	1	Rossi	v. Anemoni 5	L. 3.678.000
	2	Bianchi	v. Bolla 64	L. 730.400
	3	Brunelli	v. Po 41	L. 6.777.500
	4	Grandi	v. Romolo 3	L. 3.400.000
*		(Contatore)		L. 0

Record:  di 4

## Il codice SQL generato

```
UPDATE conto_corrente SET conto_corrente.Saldo =
[conto_corrente].[Saldo]*1.1
WHERE (((conto_corrente.Nome)="Bianchi"));
```

# Approfondimenti su Access

Scopo di questo capitolo è l'approfondimento di alcune funzioni dell'Access (maschere, report, ...).

La trattazione prenderà spunto da due esempi concreti. In ambedue i casi si tratterà di realizzare un archivio per classificare degli articoli tratti da più riviste ed organizzati in base ad argomento, parole chiave, ...

Nel primo caso il problema verrà affrontato costruendo un DB non relazionale. La soluzione è decisamente da scartare dal punto di vista del modello dei dati ma essa dà luogo ad una sola tabella e ciò permetterà una più semplice presentazione di alcuni nuovi aspetti dell'Access. La seconda soluzione proposta userà un database relazionale.

1- [DB non relazionale](#)

2- [DB relazionale](#)

Per la consultazione off-line è possibile scaricare il file seguente:

[approfondimenti.pdf](#)

Per svolgere gli esercizi proposti sono necessari anche i seguenti file:

[elenco articoli.xls](#)

[elenco articoli 2.xls](#)

[elenco argomenti.xls](#)

[elenco riviste.xls](#)

# DB non relazionale

La struttura dell'archivio che si realizzerà è illustrata dal seguente ERD:



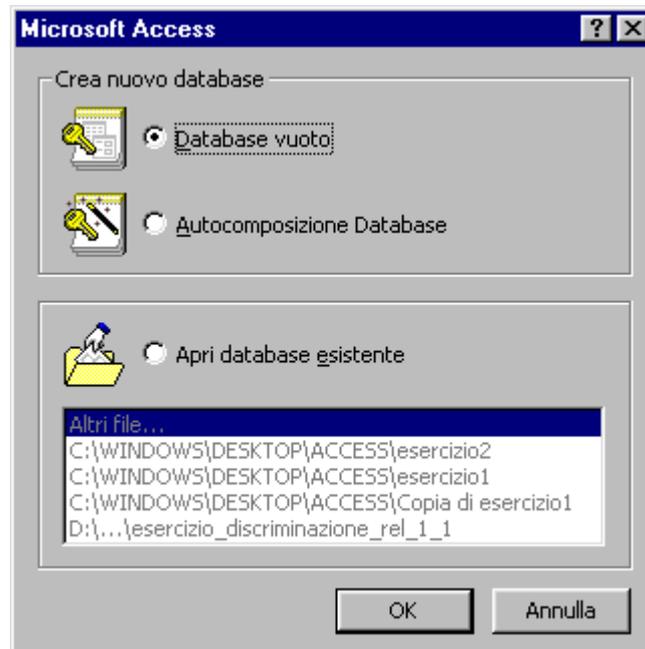
Si sottolinea nuovamente che dal punto di vista del modello dati questa struttura è decisamente non adeguata. Essa è stata scelta per favorire un più rapido apprendimento di alcune funzionalità dell'Access.

Una soluzione più valida per lo stesso archivio, viene proposta nell'esempio relativo al DB relazionale.

Il file contenente l'esempio svolto è: [esercizio1.mdb](#)

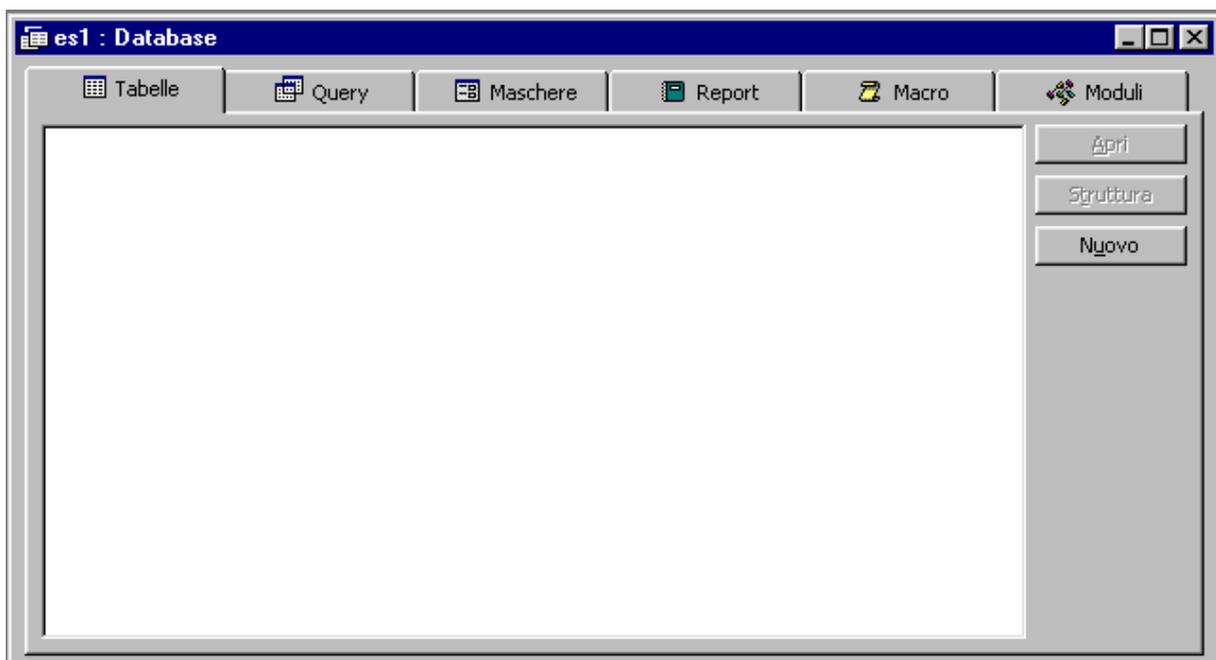
# Creare un nuovo database

Scegliere database vuoto nella finestra:



Salvare il file con il nome "es1"

Apparirà la finestra principale del db



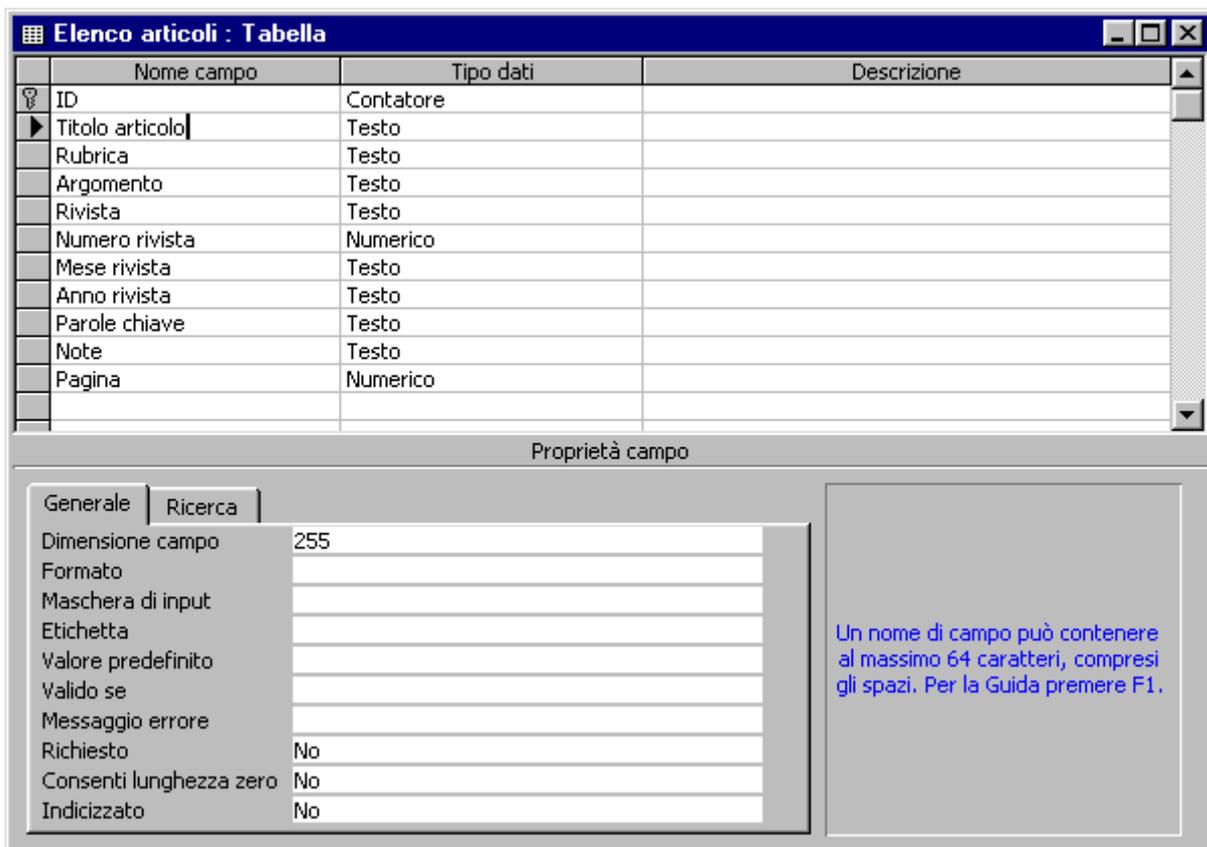
## Creare la tabella "elenco articoli"

Scegliere nuovo nella finestra del database (cartella Tabelle)

Nella finestra nuova tabella scegliere Visualizzazione struttura



Nella finestra struttura della tabella inserire le proprietà dei campi (vedi dopo)



Dettaglio dei campi della tabella:

Nome	Tipo	Note
ID	Contatore	Indicizzato: Sì (Duplicati non ammessi)
Titolo articolo	Testo	Dimensione campo: 255; richiesto: Sì; Consenti lunghezza zero: No; Indicizzato: No

Rubrica	Testo	Dimensione campo: 50; richiesto: No; Consenti lunghezza zero: No; Indicizzato: No
Argomento	Testo	Dimensione campo: 50; richiesto: No; Consenti lunghezza zero: No; Indicizzato: No
Rivista	Testo	Dimensione campo: 50; richiesto: Si; Consenti lunghezza zero: No; Indicizzato: si(duplicati ammessi)
Numero rivista	Numerico	Dimensione campo: intero; richiesto: No; Indicizzato: si(duplicati ammessi)
Mese rivista	Testo	Dimensione campo: 3; richiesto: No; Consenti lunghezza zero: No; Indicizzato: No
Anno rivista	Testo	Dimensione campo: 4; richiesto: No; Consenti lunghezza zero: No; Indicizzato: No
Parole chiave	Testo	Dimensione campo: 255; richiesto: No; Consenti lunghezza zero: No; Indicizzato: No
Note	Testo	Dimensione campo: 255; richiesto: No; Consenti lunghezza zero: No; Indicizzato: No
Pagina	Numerico	Dimensione campo: intero; richiesto: No; Indicizzato: No

## Creazione della chiave primaria

Evidenziare la riga "ID"

Fare click col pulsante destro del mouse su di essa

Scegliere "chiave primaria" dal menu

Chiudere la finestra dal bottone di chiusura

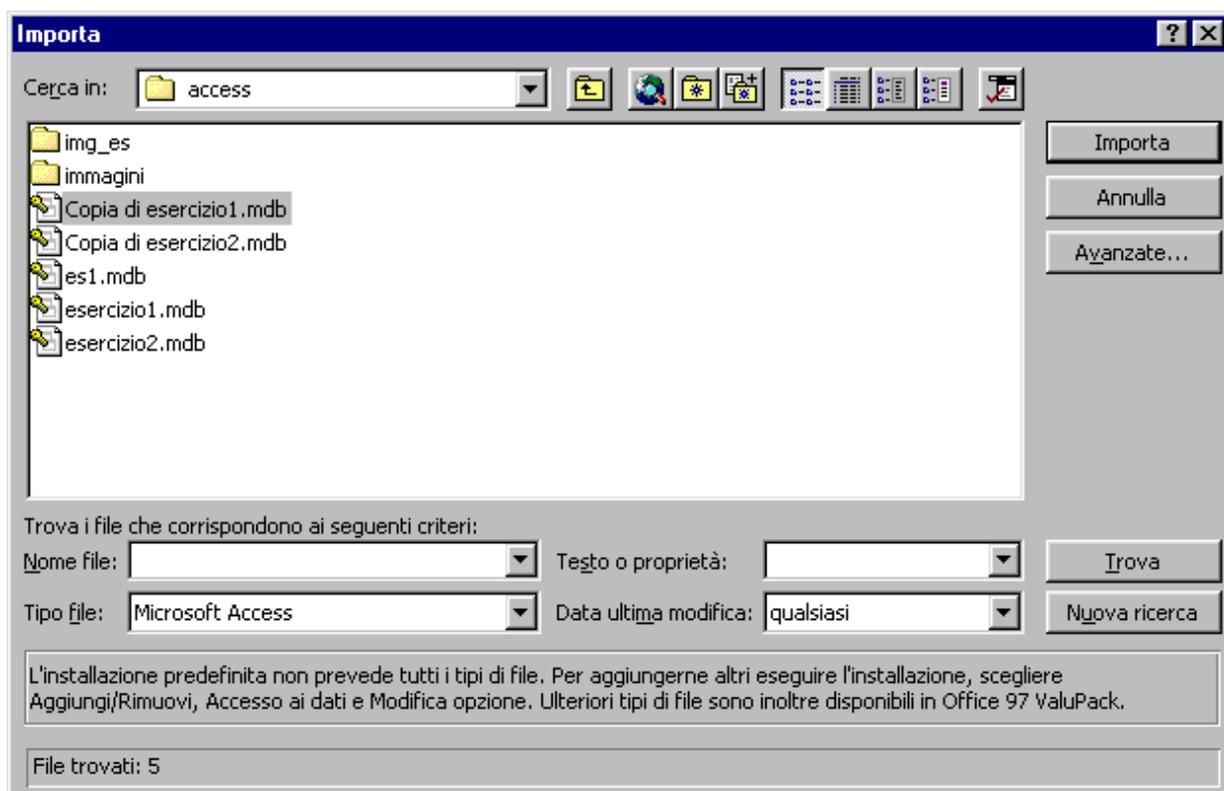
Salvare con nome "Elenco articoli"

## Importare i dati nella tabella

I dati verranno importati da una tabella Excel (file "[elenco articoli.xls](#)")

Dal menu file scegliere "Carica dati esterni" e poi "Importa"

Apparirà la finestra:



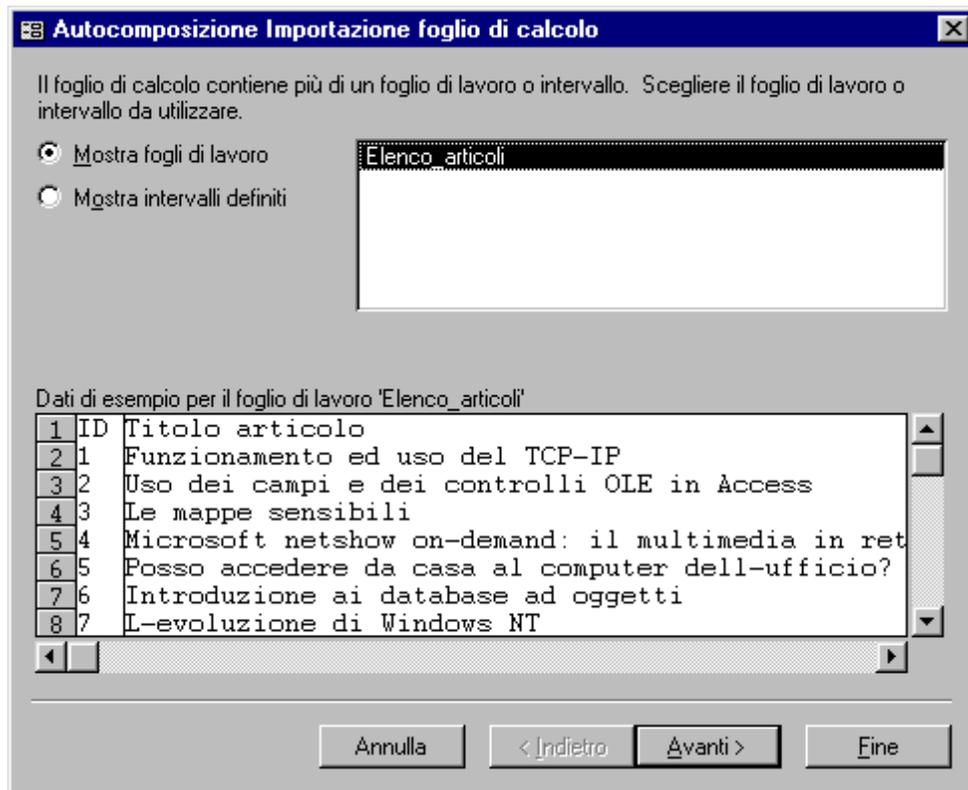
Dalla casella Tipo file scegliere "Microsoft Excel"

Usare la casella "cerca in" per aprire la cartella nella quale si trova il file "elenco articoli.xls"

Selezionare il file "elenco articoli.xls" e fare click su importa

Wizard di importazione

Apparirà la finestra:



Fare click sul bottone Avanti

Nella schermata successiva fare click sul bottone "Intestazioni di colonne nella prima riga"

Fare click sul bottone Avanti

Nella terza schermata selezionare l'opzione "tabella esistente" e scegliere la tabella "elenco articoli" dal menu a tendina

Fare click sul bottone Avanti

Fare click sul bottone Fine

## Verifica del risultato dell'importazione dei dati

Nella finestra del database evidenziare la tabella "elenco articoli" e fare click sul bottone Apri

Apparirà la finestra:

ID	Titolo articolo	Rubrica	Argomento	Rivista	Numero rivista
1	Funzionamento	-----	Internet, reti, te	MC Microcomp	169 G
2	Uso dei campi	-----	Database	MC Microcomp	169 G
3	Le mappe sens	-----	HTML	MC Microcomp	169 G
4	Microsoft netsh	-----	Internet, reti, te	PC Professiona	169 G
5	Posso acceden	-----	Sistemi operati	MC Microcomp	169 G
6	Introduzione ai	-----	Database	PC Professiona	54 G
7	L-evoluzione di	-----	Sistemi operati	PC Professiona	54 G
8	Due o tre cose	-----	Linguaggi di prc	PC Professiona	54 G
9	Mondi nei monc	-----	Internet, reti, te	PC Professiona	169 G
10	Programmare p	-----	Sistemi operati	PC Professiona	54 G
11	Rete ideale, arc	-----	Internet, reti, te	PC Professiona	68 D
12	Navigando nella	-----	Internet, reti, te	PC Professiona	68 D

Record: 1 di 71

Come evidenziato in basso la tabella contiene adesso 71 record

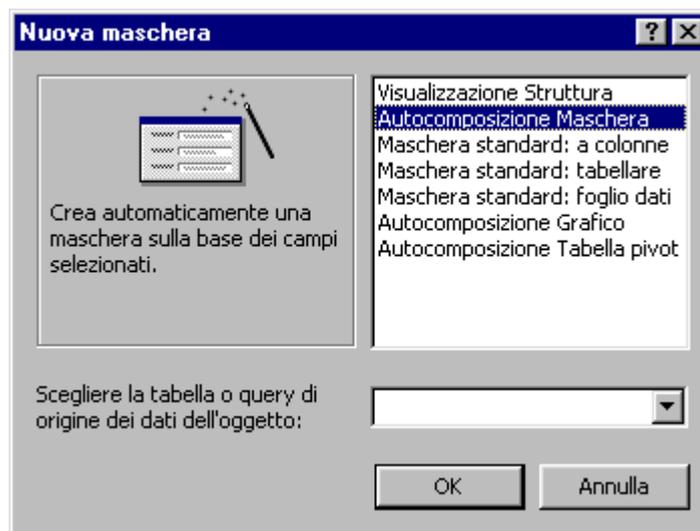
## Realizzare la maschera "elenco articoli"

### Autocomposizione

Dalla finestra del database scegliere la cartella Maschere

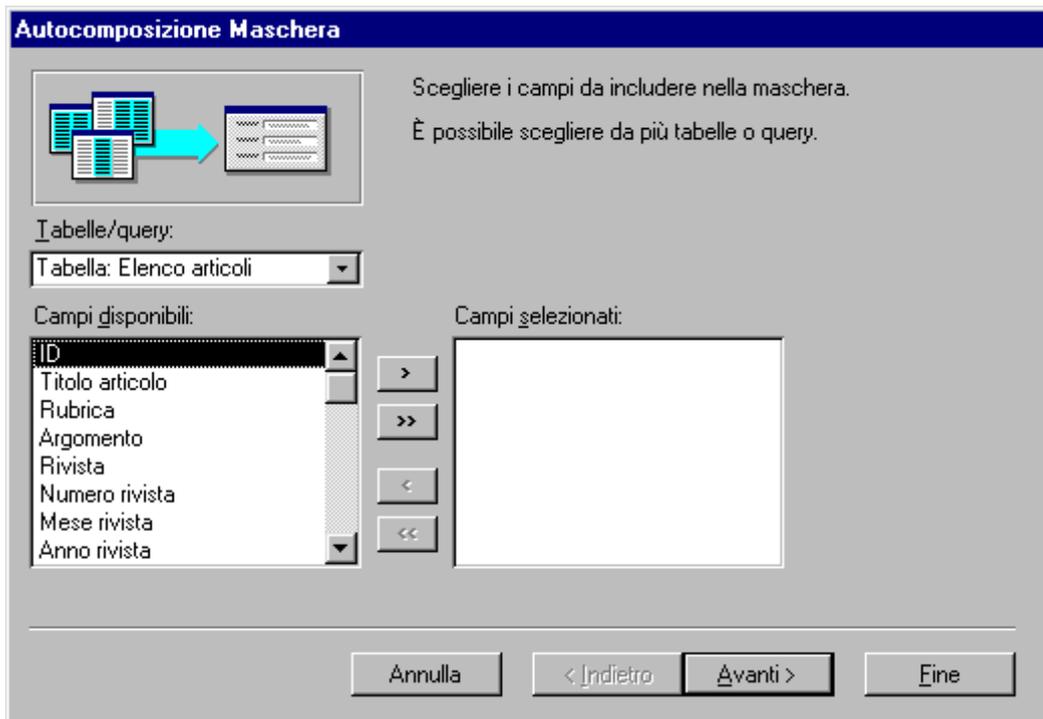
Fare click sul bottone Nuovo

Nella finestra seguente scegliere Autocomposizione maschera:



Scegliere anche la tabella "elenco articoli" dal menu a tendina in basso

Apparirà la prima schermata dell'autocomposizione:

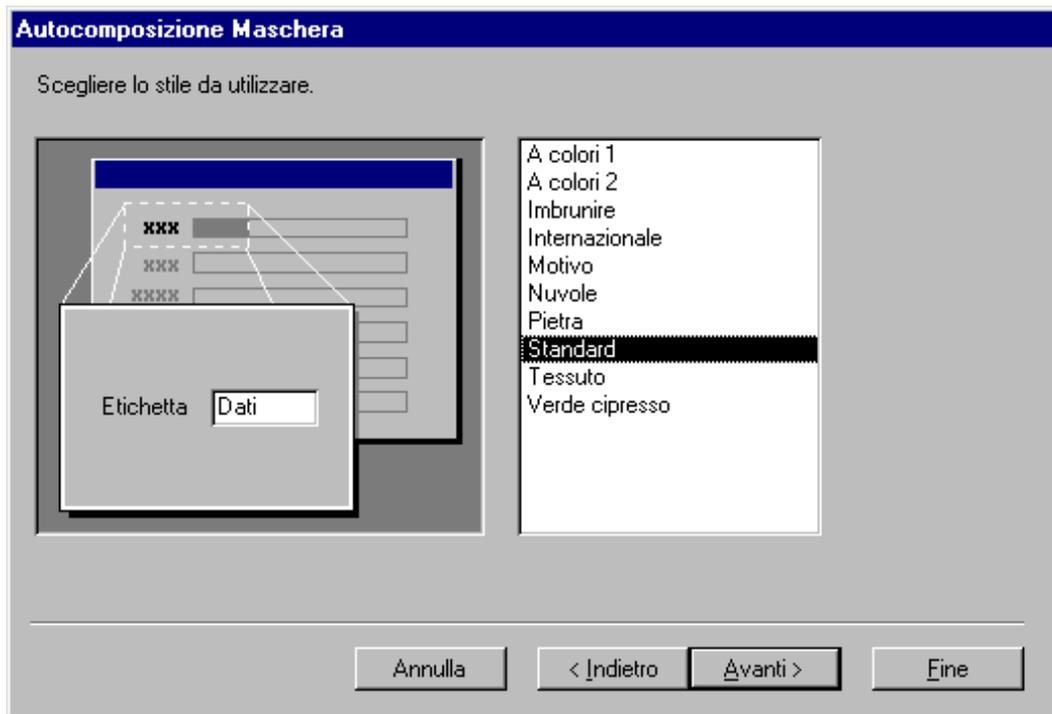


Spostare nell'elenco a sinistra tutti i campi tranne ID

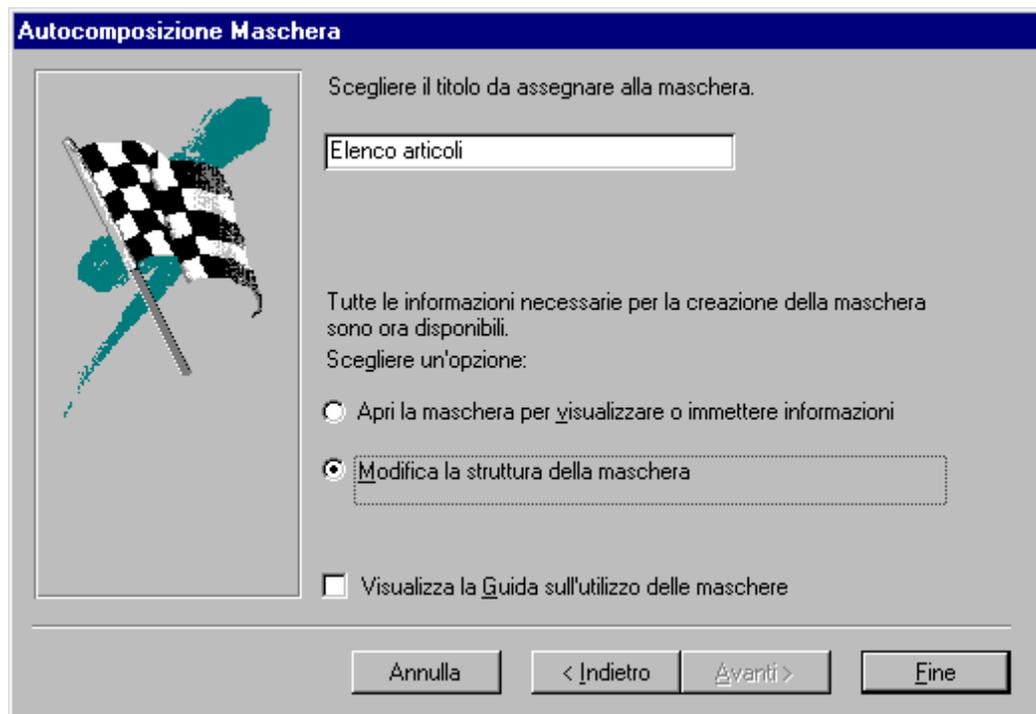
Nella successiva finestra scegliere 'A colonne'



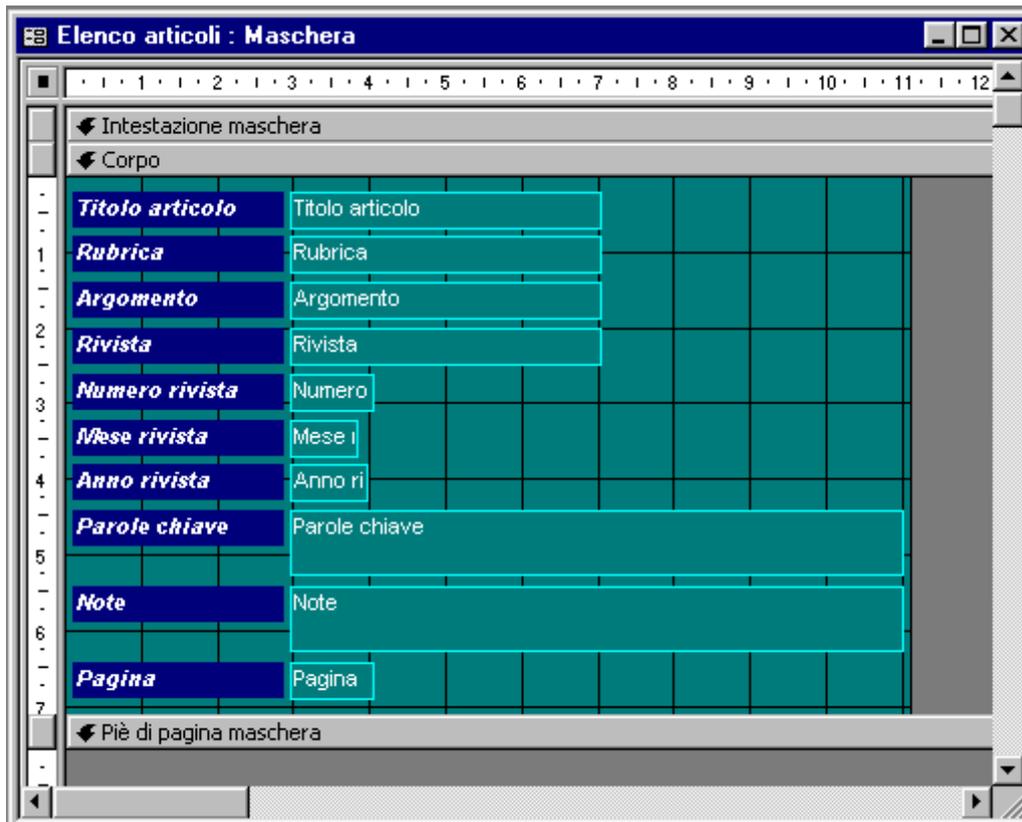
Nella finestra di scelta dello stile selezionare 'A colori 1'



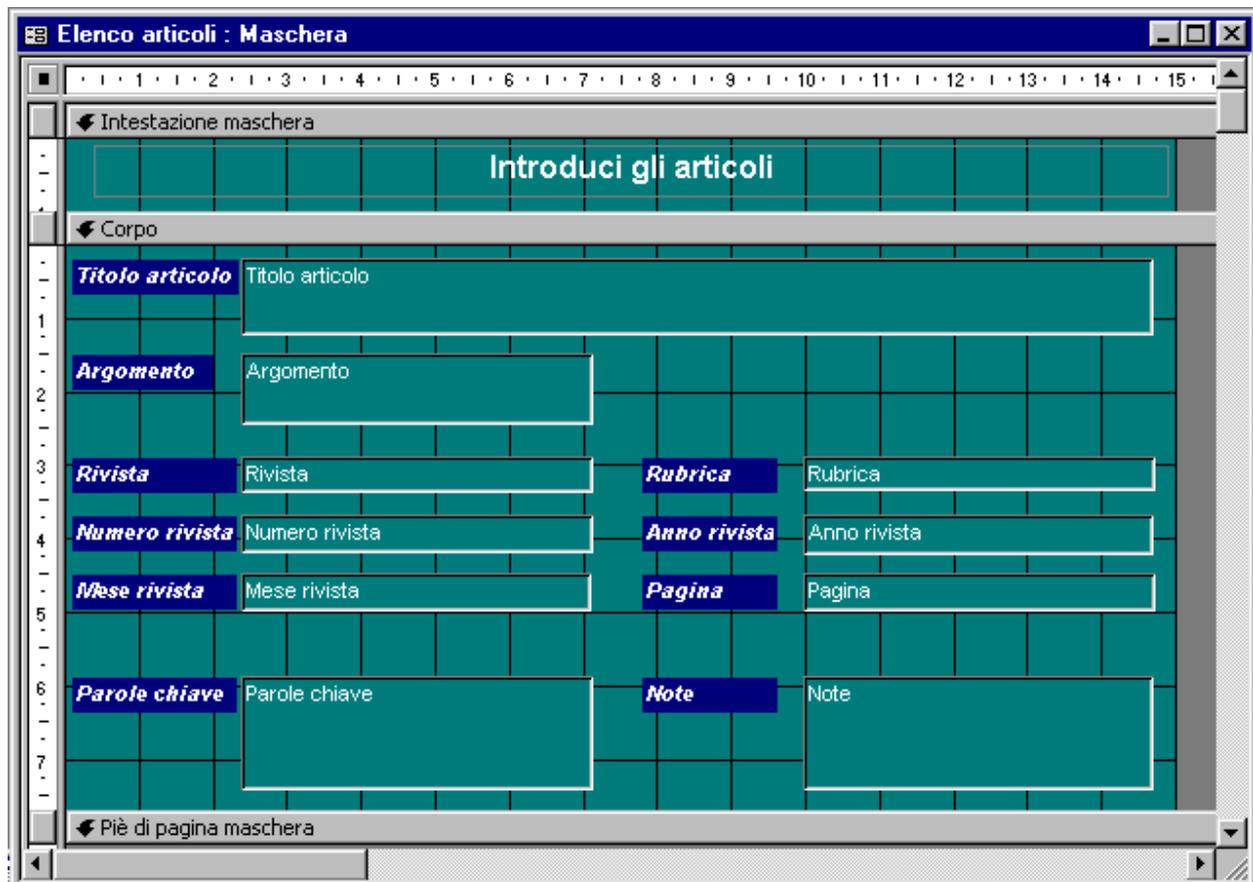
Nell'ultima finestra dell'autocomposizione impostare il nome della maschera (Elenco articoli) e scegliere 'Modifica la struttura della maschera'



Fare click sul bottone Fine, apparirà la maschera in visualizzazione struttura:



Modificarne l'aspetto come di seguito indicato:



Chiudere la finestra

Dalla finestra del database fare click su Apri per provare la maschera appena costruita

## Ordine di tabulazione (menu visualizza)

Aprire la maschera in modalità struttura

Per cambiare l'ordine di tabulazione scegliere dal menu Visualizza il comando 'ordine di tabulazione'

Trascinare i campi affinché appaiano nell'ordine indicato in figura:



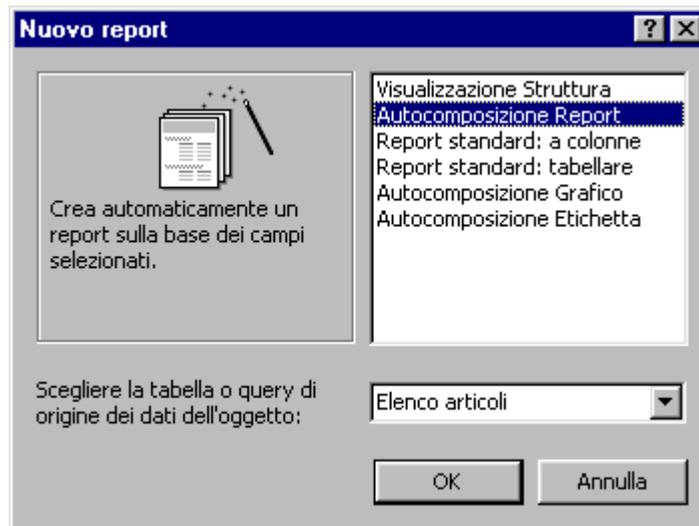
Provare nuovamente la maschera spostandosi da un campo all'altro con il tasto <TAB>

## Creare il report

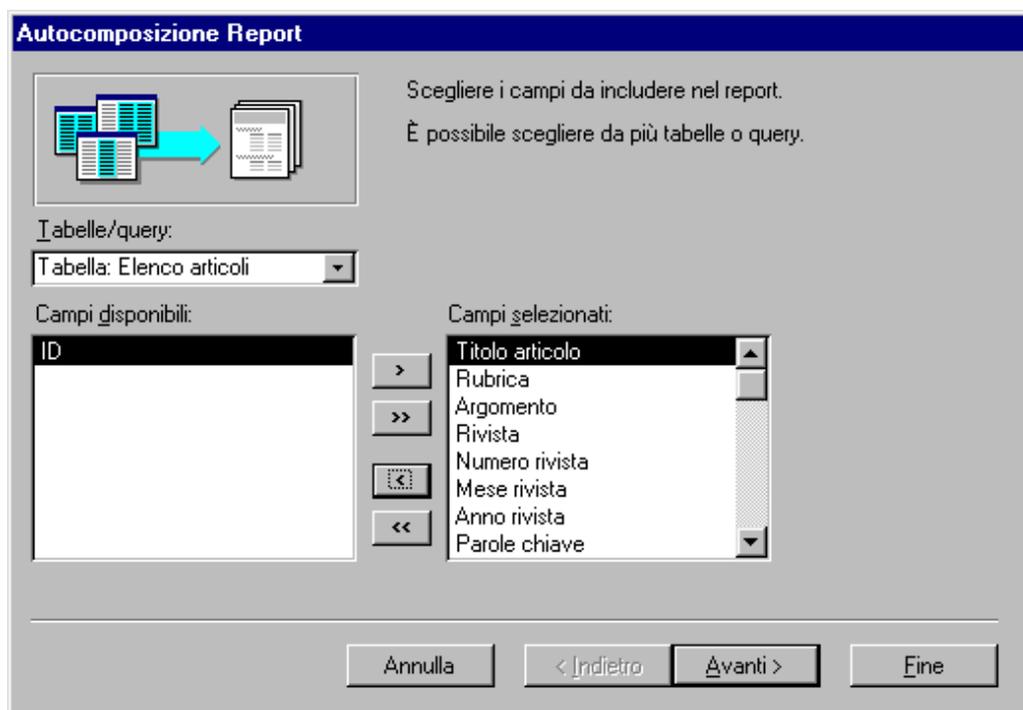
Dalla finestra del database scegliere la cartella 'Report'

Fare click sul bottone Nuovo

Nella finestra seguente scegliere 'Autocomposizione report' e poi selezionare la tabella 'elenco articoli' dal menu a tendina in basso

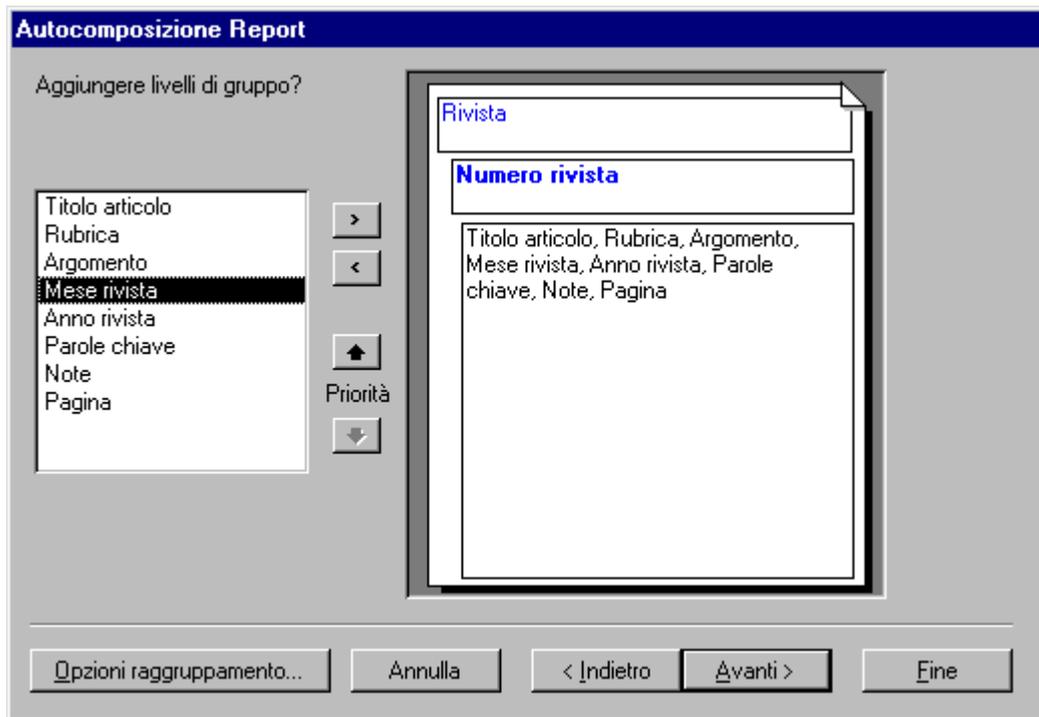


Nella finestra seguente spostare tutti i record tranne ID nella sezione 'campi selezionati'



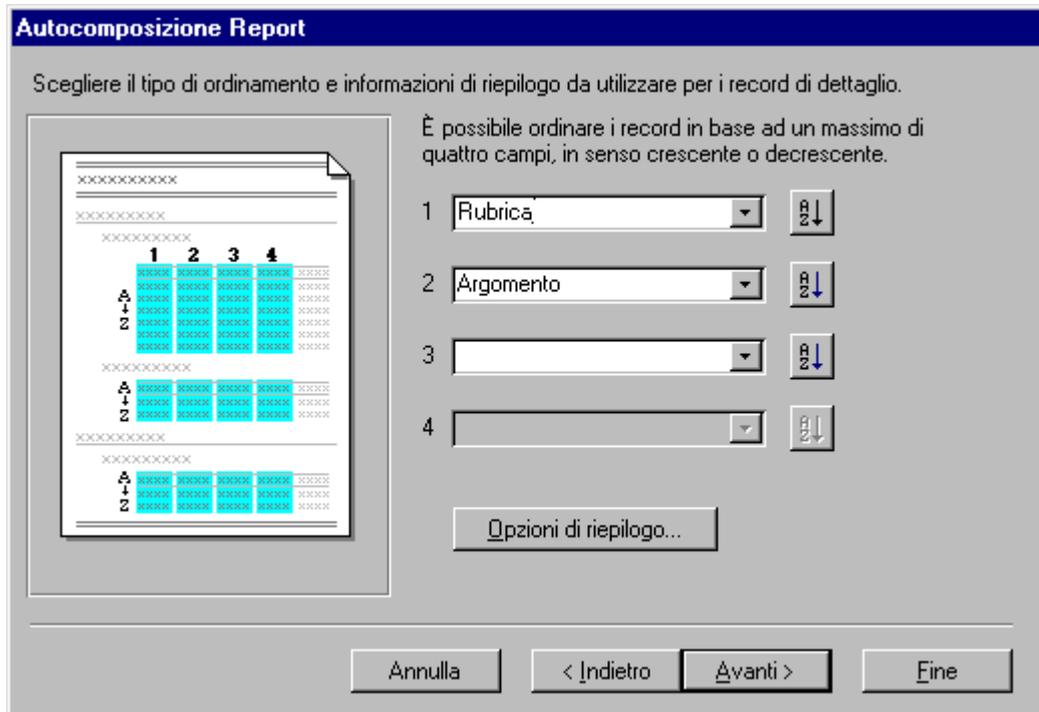
Fare click sul bottone Avanti

Nella finestra seguente raggruppare secondo 'Rivista' e 'Numero rivista'



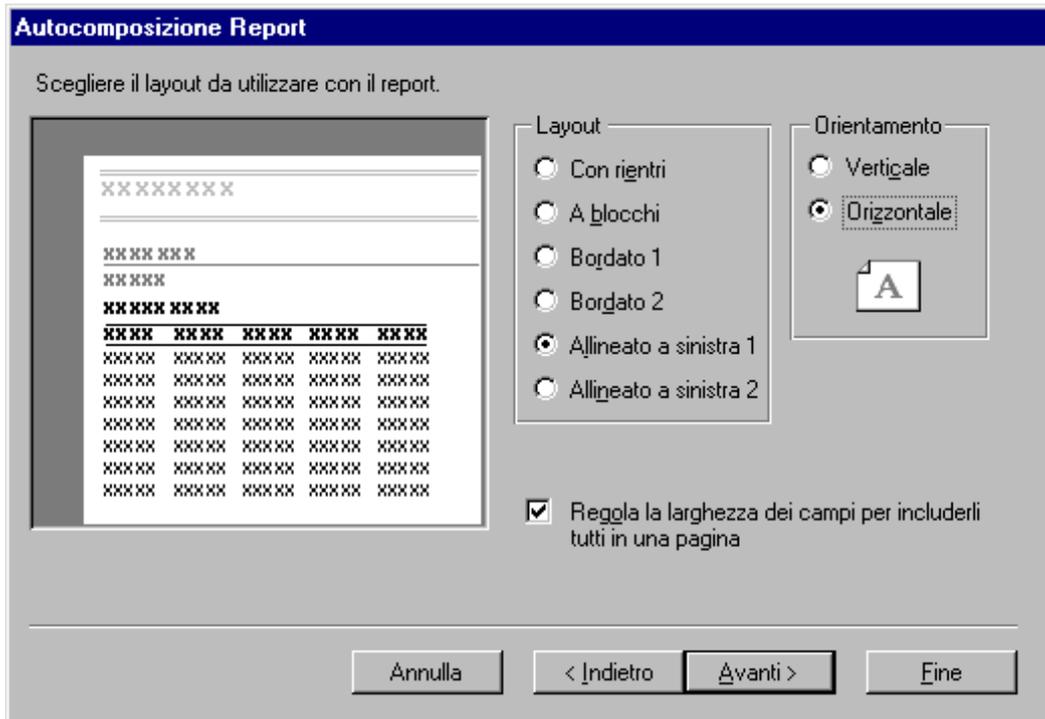
Fare click sul bottone Avanti

Nella finestra seguente ordinare per 'Rubrica' e 'Argomento' scegliendo dai menu a tendina i relativi campi



Fare click sul bottone Avanti

Nella finestra seguente scegliere il layout 'Allineato a sinistra 1', orientamento 'orizzontale', selezionare 'Regola la larghezza...'



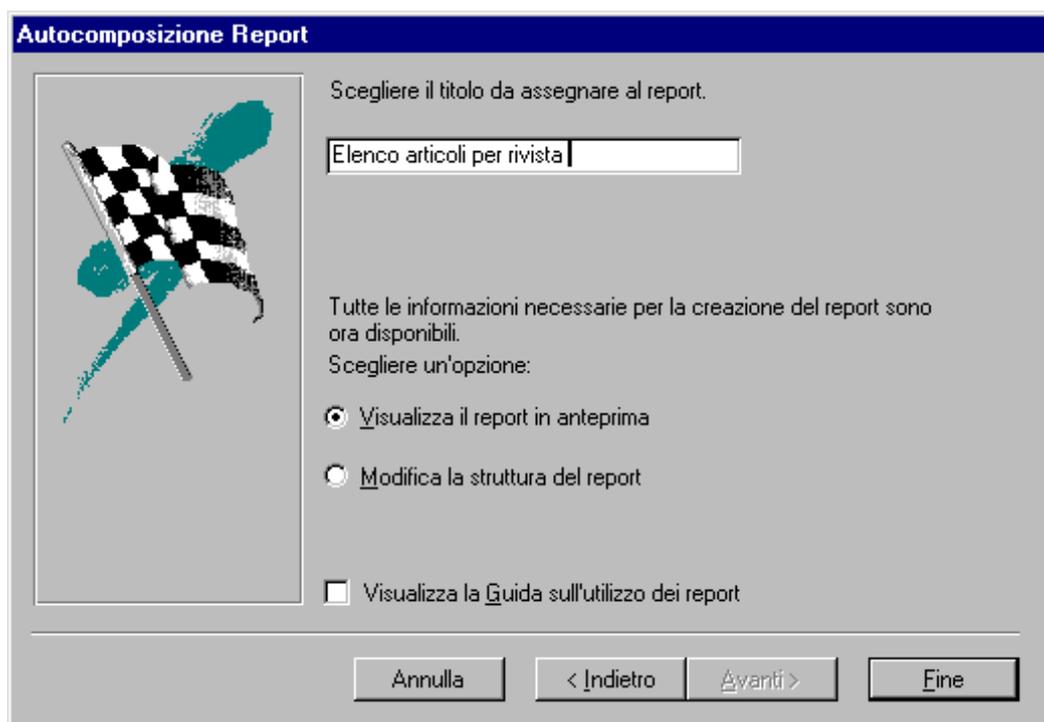
Fare click sul bottone Avanti

Dalla finestra seguente scegliere lo stile 'Aziendale' dalla lista



Fare click sul bottone Avanti

Nell'ultima finestra scegliere il nome da dare al report ('Elenco articoli per rivista')



Fare click sul bottone Fine

Apparirà l'anteprima del report:



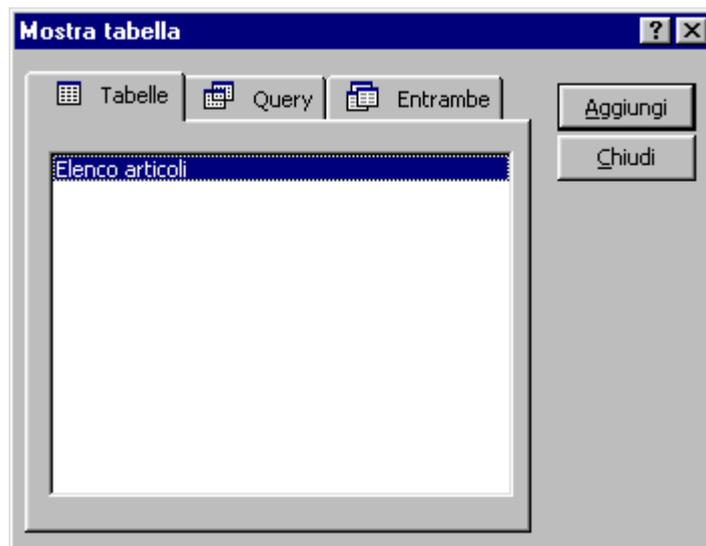
# Creare la query "cerca argomento"

Dalla finestra del database scegliere la cartella 'query'

Dalla finestra seguente selezionare 'Visualizzazione struttura'



Nella finestra seguente selezionare la tabella 'elenco articoli' e fare click sul bottone Aggiungi per inserire la tabella tra quelle usate per la query

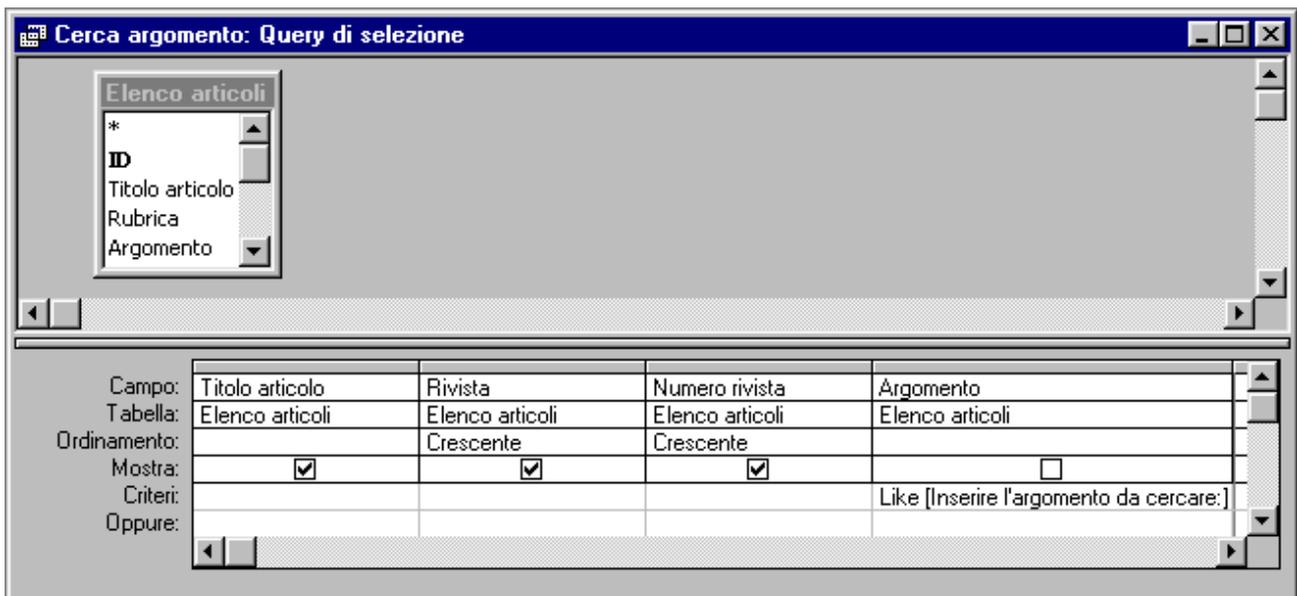


Chiudere la finestra dal bottone chiudi

Apparirà la finestra della query in modalità struttura



trascinare i campi dalla tabella in alto nei corrispondenti campi della riga 'Campo' fino ad ottenere la situazione di figura:



Nel campo 'Criteri' inserire il criterio di ricerca:

(per provare la query usare il bottone ; per tornare alla modalità struttura usare il bottone )

■ Nella colonna 'Argomento' provare

- = "software"
- = "software" or "hardware"
- = [Inserire l'argomento da cercare]
- Like "software"
- Like "\*ware"

*Like "????ware"*  
*Like [Inserire l'argomento da cercare]*  
*In ("Software";"Hardware")*  
*is null*  
*is not null*

■ Nella colonna del 'numero rivista' provare

*Between 54 And 66*  
*Not Between 54 And 66*  
*= 66*  
*> 66*  
*>= 66*

Ordinare in modo crescente per 'Rivista' e 'numero rivista' scegliendo la modalità dal menu a tendina del rigo 'Ordinamento'

## Creare la maschera "cerca argomento"

### Descrizione

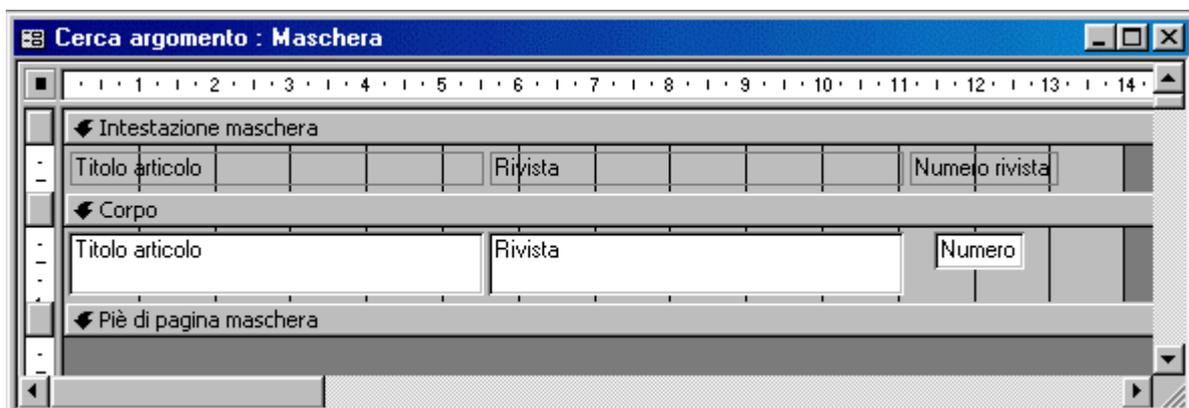
La maschera mostra gli articoli (soltanto campi 'titolo', 'rivista', 'numero rivista') corrispondenti al criterio indicato nella query 'cerca argomento'

Creazione della maschera

Procedere come visto precedentemente

1. Scegliere 'autocomposizione maschera' e la query "Cerca argomento" come origine dei dati
2. Inserire nei campi selezionati 'titolo articolo', 'rivista', 'numero rivista'
3. Scegliere il layout 'Tabulare'
4. Scegliere lo stile 'standard'
5. Denominare la maschera "cerca argomento"

La struttura dovrebbe essere simile a questa:





La maschera finita avrà il seguente aspetto:



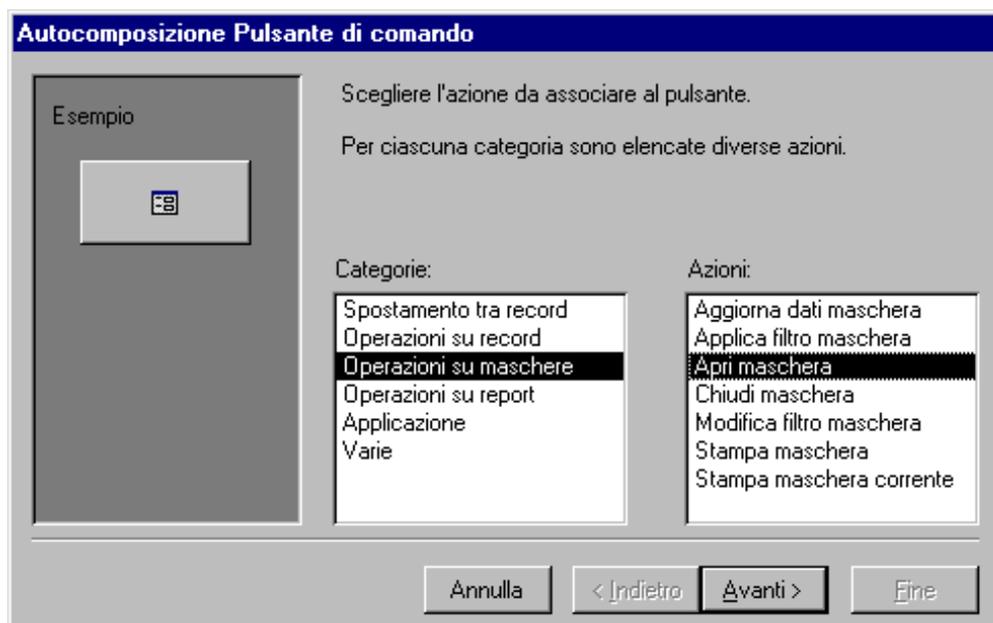
Inserire il titolo con il bottone , colorare il testo e scegliere le dimensioni del carattere

Inserire la linea orizzontale con il bottone , impostare lo stile tridimensionale con il menu degli effetti 3D:



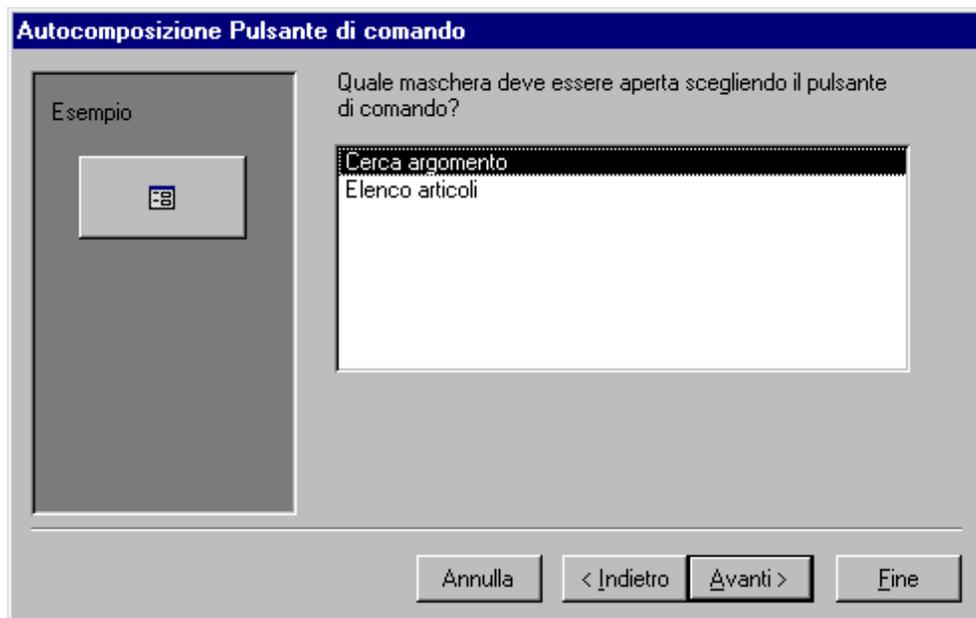
Inserire il bottone "Elenco di tutti gli articoli" con il bottone 

Dalla finestra di autocomposizione del bottone scegliere "operazioni su maschere" e "Apri maschere"



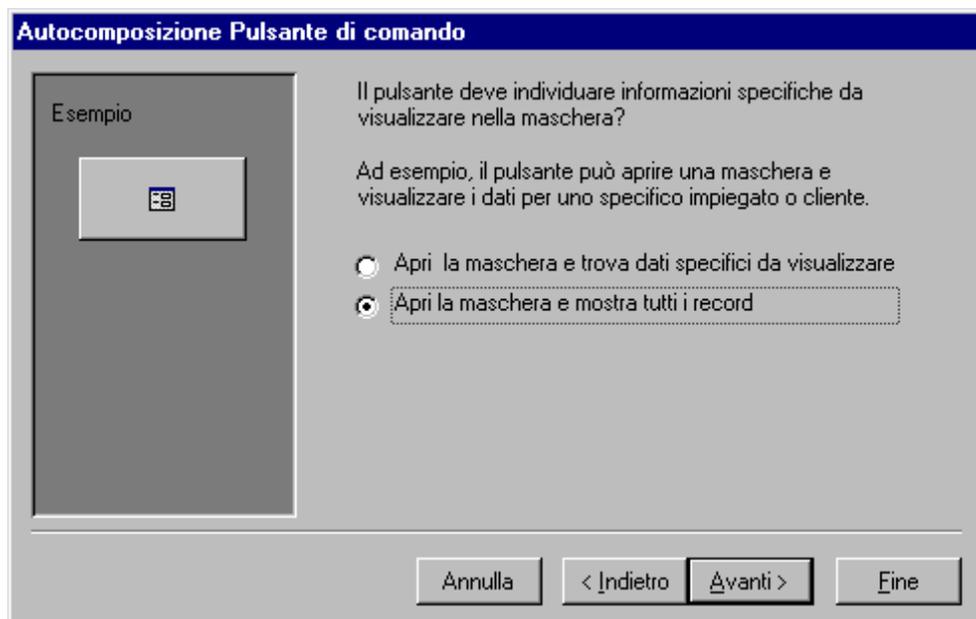
Fare click sul bottone Avanti

Dalla finestra di scelta della maschera selezionare "Elenco articoli"

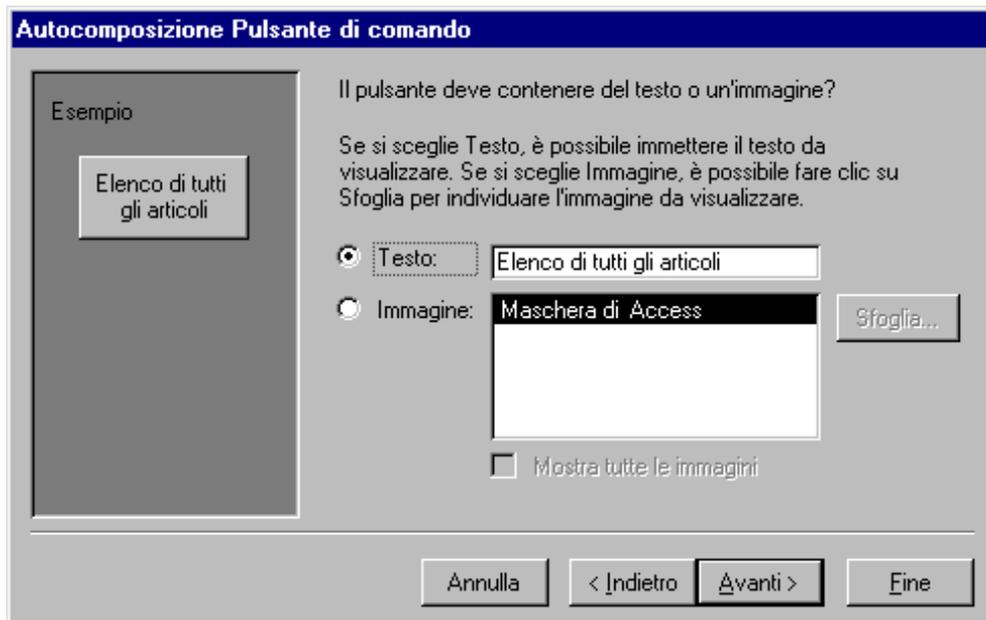


Fare click sul bottone Avanti

Dalla finestra seguente scegliere "Apri la maschera e mostra tutti i record"

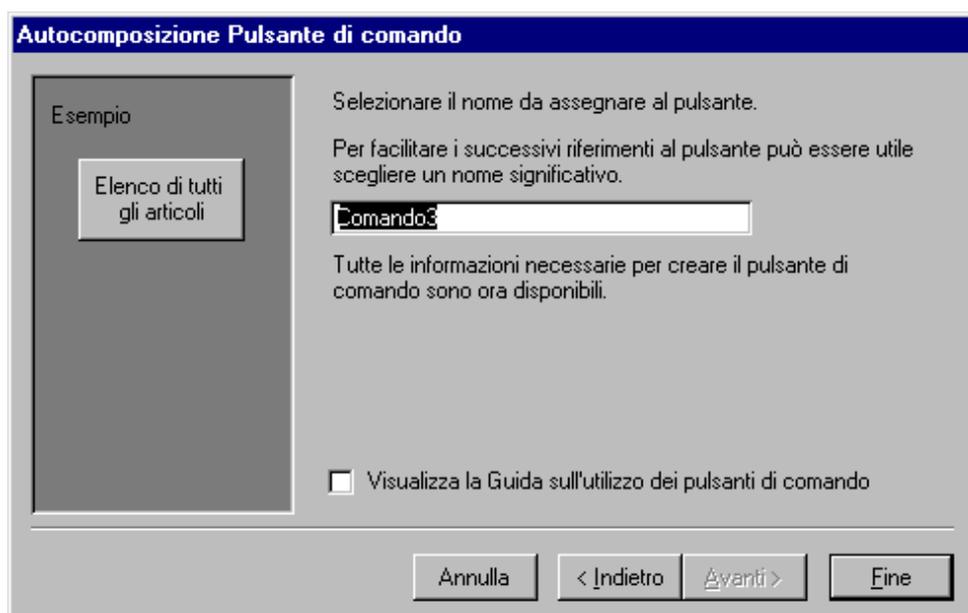


Dalla finestra riguardante l'aspetto del bottone scegliere il "Testo" ed inserire "Elenco di tutti gli articoli"



Fare click sul bottone Avanti

Nella finestra seguente si può scegliere un nome per il bottone



Fare click sul bottone Fine

Salvare la finestra (menu File, Salva) e scegliere il nome "Menu"

Inserire il bottone "Cerca argomento" con il bottone 

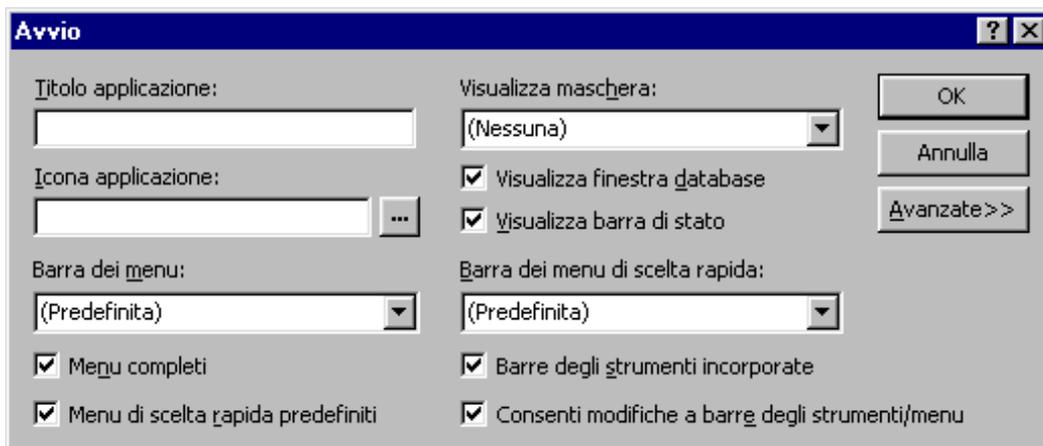
Procedere come già visto ma scegliere la maschera "cerca argomento"

Inserire il bottone "Anteprima report" con il bottone 

1. Nella prima finestra di autocomposizione scegliere 'operazioni su report' e 'anteprima report'
2. Nella finestra seguente scegliere il report "elenco articoli per rivista"
3. Nella finestra seguente inserire il testo "Anteprima report"
4. Nella finestra seguente scegliere il nome del bottone, fare click su fine

## Opzioni di avvio (menu 'Strumenti')

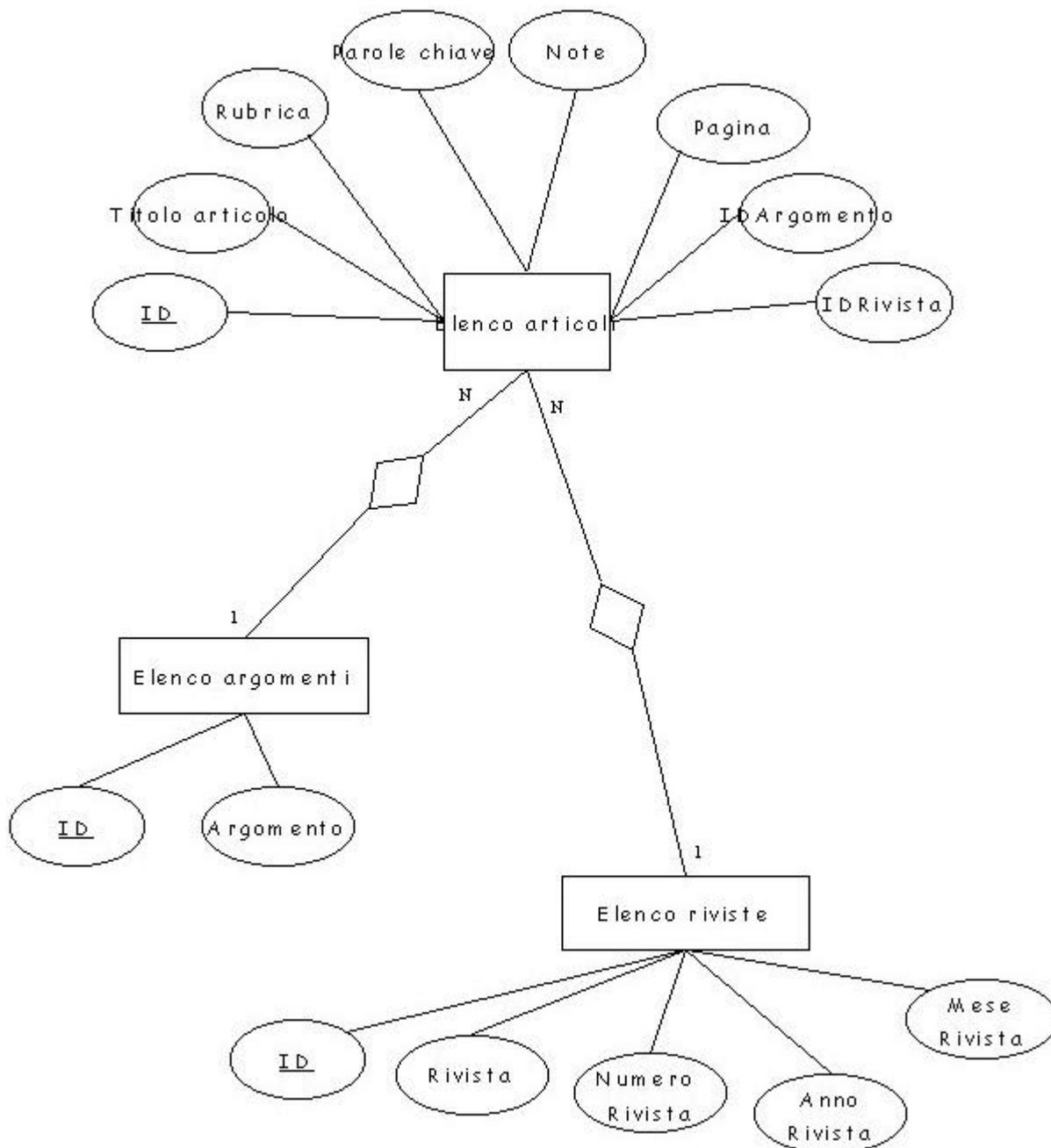
Per mostrare la maschera menu all'apertura del file bisogna scegliere la finestra "menu" dal menu a tendina "Visualizza maschera"



Impostare il titolo dell'applicazione ("Elenco articoli delle riviste di informatica")

## DB relazionale

La struttura dell'archivio che si realizzerà è illustrata dal seguente ERD:



Il file contenente l'esempio svolto è: [esercizio2.mdb](#)

## Creazione della struttura del database

### Creare la tabella 'elenco articoli'

Procedere come già visto a proposito dell'esercizio 1

Impostare i campi secondo la tabella seguente:

Nome	Tipo	Note
------	------	------

ID	Contatore	Indicizzato: Sì (Duplicati non ammessi)
Titolo articolo	Testo	Dimensione campo: 255; richiesto: Sì; Consenti lunghezza zero: No; Indicizzato: No
Rubrica	Testo	Dimensione campo: 50; richiesto: No; Consenti lunghezza zero: No; Indicizzato: No
Parole chiave	Testo	Dimensione campo: 255; richiesto: No; Consenti lunghezza zero: No; Indicizzato: No
Note	Testo	Dimensione campo: 255; richiesto: No; Consenti lunghezza zero: No; Indicizzato: No
Pagina	Numerico	Dimensione campo: intero; richiesto: No; Indicizzato: No
IDArgomento	Numerico	Dimensione campo: intero lungo; richiesto: No; Indicizzato: Sì (duplicati ammessi)
IDRivista	Numerico	Dimensione campo: intero lungo; richiesto: No; Indicizzato: Sì (duplicati ammessi)

## Creare la tabella 'elenco argomenti'

Procedere come già visto a proposito dell'esercizio 1

Impostare i campi secondo la tabella seguente:

Nome	Tipo	Note
ID	Contatore	Indicizzato: Sì (Duplicati non ammessi)
Argomento	Testo	Dimensione campo: 50; richiesto: No; Consenti lunghezza zero: No; Indicizzato: No

## Creare la tabella 'elenco riviste'

Procedere come già visto a proposito dell'esercizio 1

Impostare i campi secondo la tabella seguente:

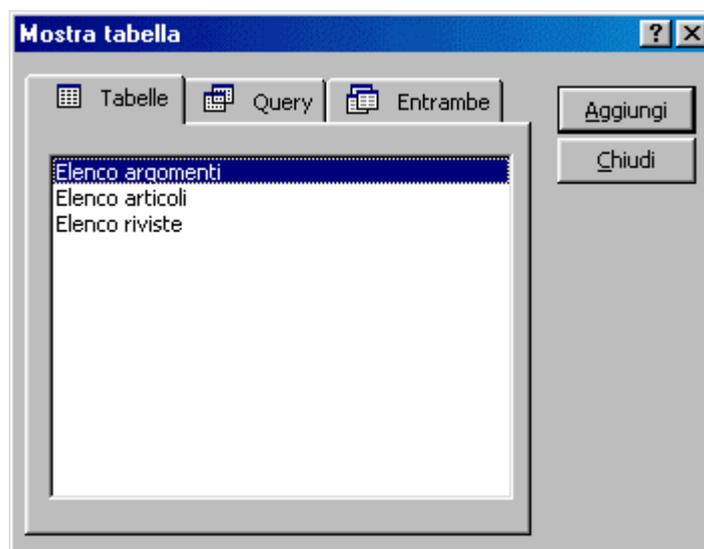
Nome	Tipo	Note
ID	Contatore	Indicizzato: Sì (Duplicati non ammessi)
Rivista	Testo	Dimensione campo: 50; richiesto: Sì; Consenti lunghezza zero: No; Indicizzato: sì(duplicati ammessi)
Numero rivista	Numerico	Dimensione campo: intero; richiesto: No; Indicizzato: sì(duplicati ammessi)
Mese rivista	Testo	Dimensione campo: 3; richiesto: No; Consenti

		lunghezza zero: No; Indicizzato: No
Anno rivista	Testo	Dimensione campo: 4; richiesto: No; Consenti lunghezza zero: No; Indicizzato: No

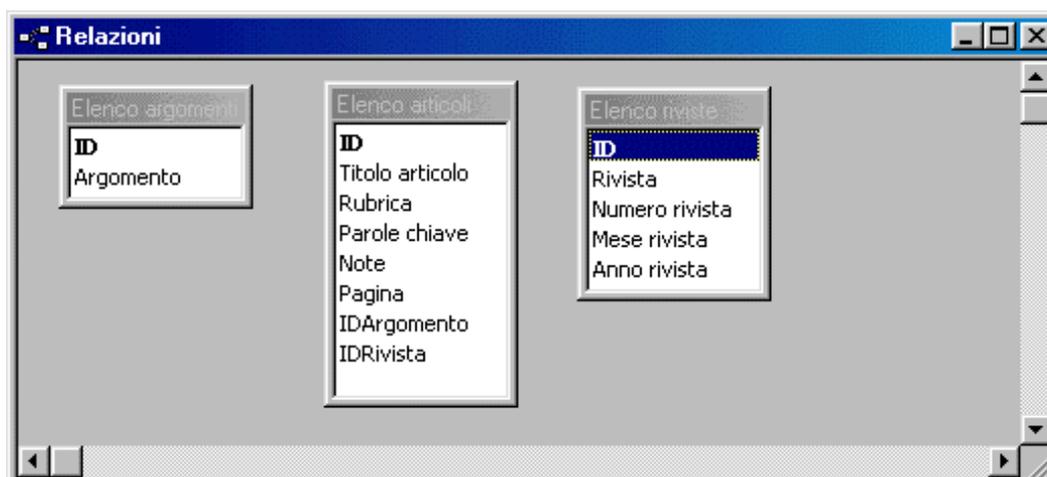
## Impostare le relazioni

Per inserire le relazioni tra le tabelle usare il bottone  oppure dal menu 'Strumenti' il comando 'Relazioni'

Dalla finestra seguente aggiungere le 3 tabelle prima create alla finestra delle relazioni:



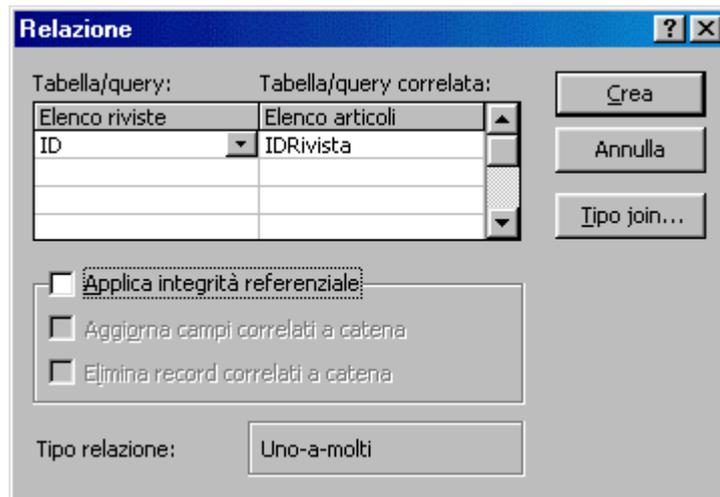
Apparirà la finestra delle relazioni:



### Creazione della relazione tra 'Elenco articoli' e 'Elenco riviste'

Il campo 'IdRivista' di 'Elenco articoli' sarà in relazione con il campo 'ID' di 'Elenco riviste'

1. Portare il mouse su IDRivista e trascinare su ID
2. Apparirà la finestra di impostazione della relazione:



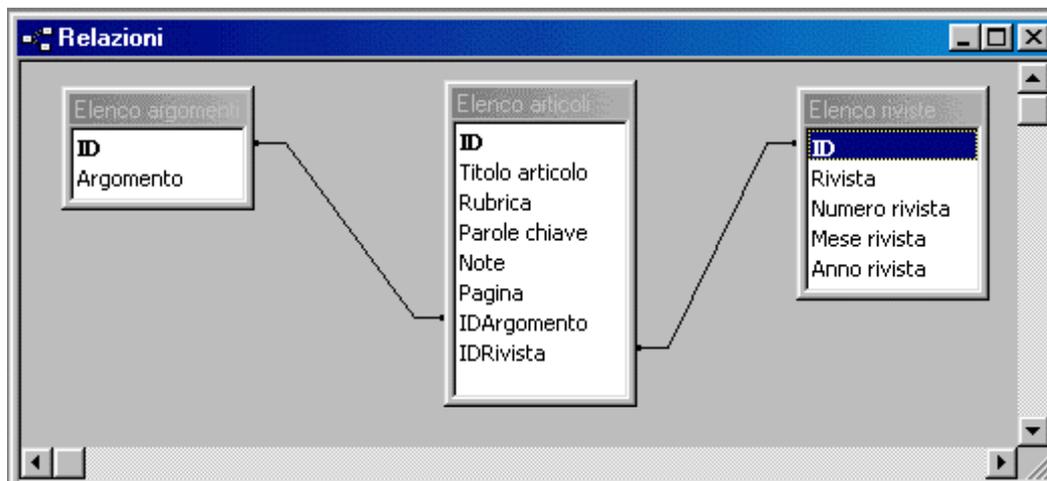
(come si vede la relazione viene automaticamente impostata come uno a molti)

3. fare click su Crea

## Creazione della relazione tra 'Elenco articoli' e 'Elenco argomenti'

Procedere come già visto mettendo in relazione il campo 'Idargomento' di 'Elenco articoli' con il campo 'ID' di 'Elenco argomenti'

La situazione dovrebbe essere quella di figura:



chiudere la finestra delle relazioni dal bottone di chiusura  e salvare le modifiche

## Importare i dati nelle tabelle

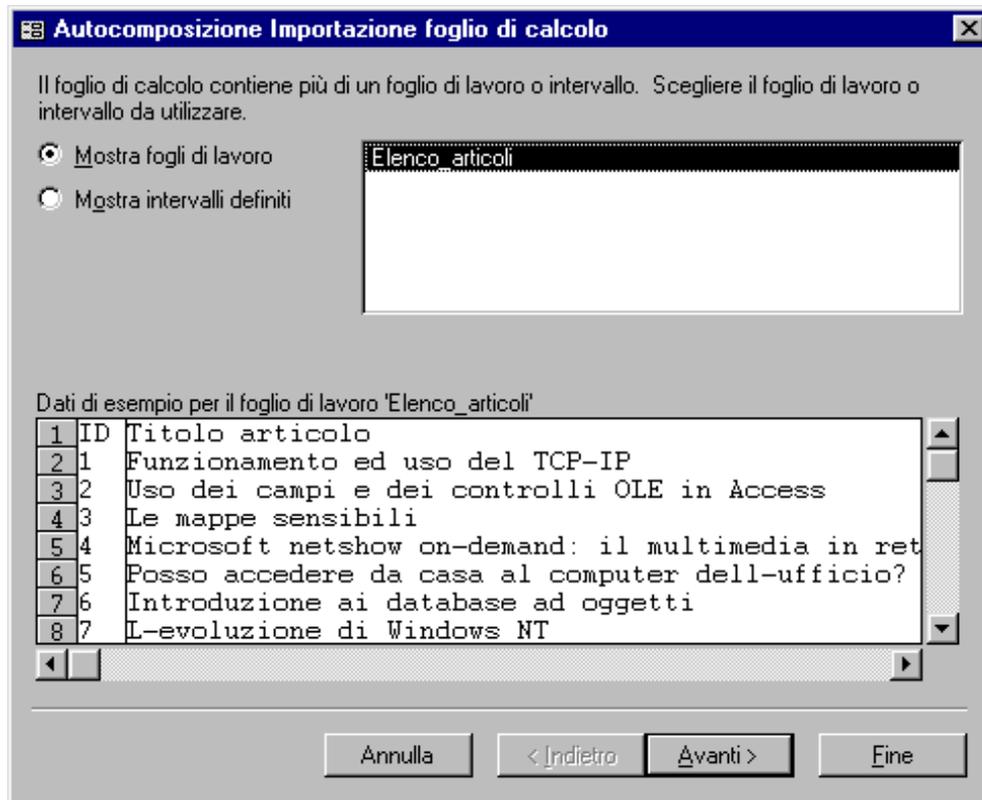
### Tabella 'Elenco articoli'

I dati verranno importati da una tabella Excel (file "[elenco articoli 2.xls](#)")

Dal menu file scegliere "Carica dati esterni" e poi "Importa", indicare il file anzidetto

## Wizard di importazione

Apparirà la finestra:



Fare click sul bottone Avanti

Nella schermata successiva fare click sul bottone "Intestazioni di colonne nella prima riga"

Fare click sul bottone Avanti

Nella terza schermata selezionare l'opzione "tabella esistente" e scegliere la tabella "elenco articoli" dal menu a tendina

Fare click sul bottone Avanti

Fare click sul bottone Fine

## Tabella 'Elenco argomenti'

I dati verranno importati da una tabella Excel (file "[elenco argomenti.xls](#)")

Dal menu file scegliere "Carica dati esterni" e poi "Importa", indicare il file anzidetto

Procedere come già visto scegliendo di importare i dati nella tabella 'Elenco argomenti'

## Tabella 'Elenco riviste'

I dati verranno importati da una tabella Excel (file "[elenco riviste.xls](#)")

Dal menu file scegliere "Carica dati esterni" e poi "Importa", indicare il file anzidetto

Procedere come già visto scegliendo di importare i dati nella tabella 'Elenco riviste'

## Creare le query

### Realizzare la query "Elenco articoli con argomenti e riviste"

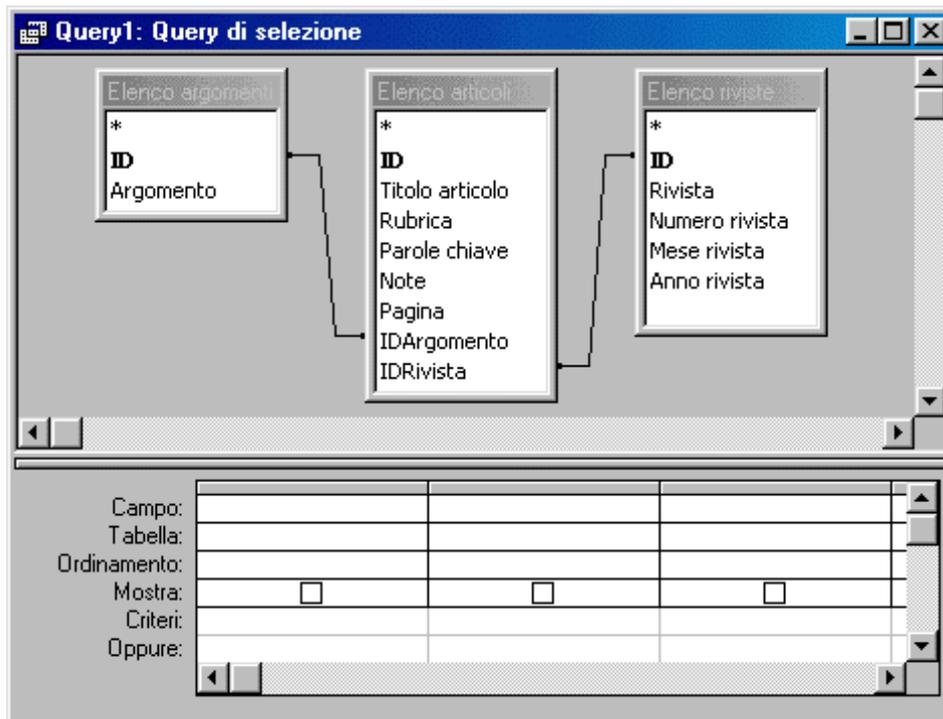
Dalla finestra del database scegliere la cartella 'query'

Dalla finestra seguente selezionare 'Visualizzazione struttura'



Nella finestra seguente selezionare le 3 tabelle e fare click sul bottone Aggiungi per inserirle tra quelle usate per la query. Chiudere la finestra dal bottone chiudi

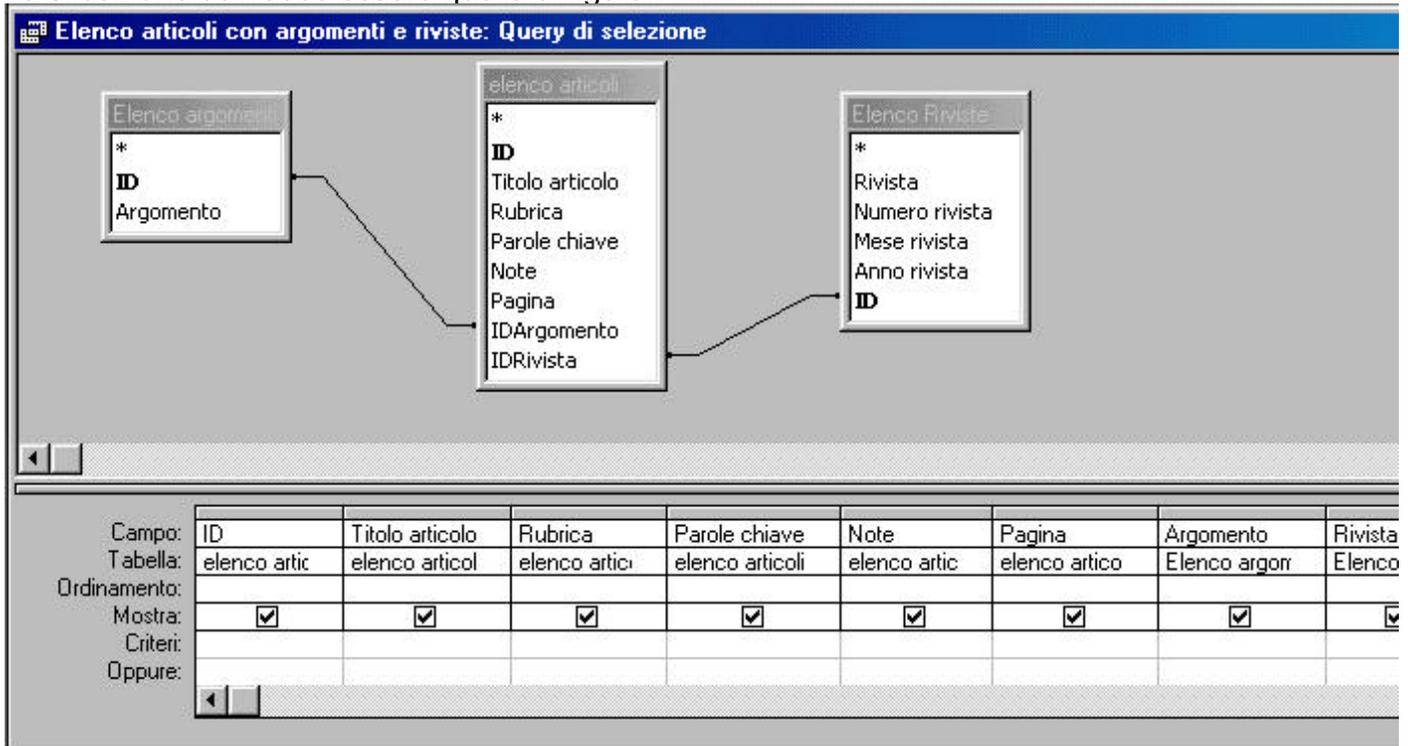
Apparirà la finestra della query in modalità struttura



Trascinare i campi dalle tabelle in alto nei corrispondenti campi della riga 'Campo' secondo quanto descritto nella tabella seguente:

Campo	Tabella	Mostra
ID	Elenco articoli	<input type="checkbox"/>
Titolo articolo	Elenco articoli	<input type="checkbox"/>
Rubrica	Elenco articoli	<input type="checkbox"/>
Parole chiave	Elenco articoli	<input type="checkbox"/>
Note	Elenco articoli	<input type="checkbox"/>
Pagina	Elenco articoli	<input type="checkbox"/>
Argomento	Elenco argomenti	<input type="checkbox"/>
Rivista	Elenco riviste	<input type="checkbox"/>
Numero rivista	Elenco riviste	<input type="checkbox"/>
Mese rivista	Elenco riviste	<input type="checkbox"/>
Anno rivista	Elenco riviste	<input type="checkbox"/>

La situazione dovrebbe essere quella di figura:

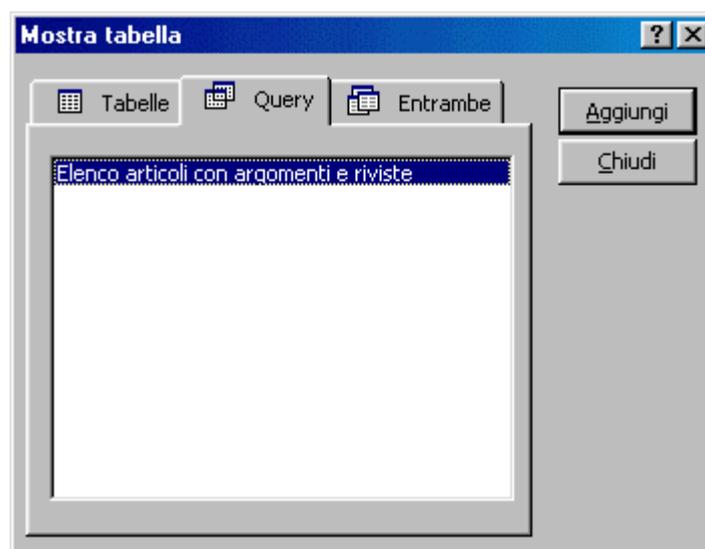


Ordinare in modo crescente per 'Rivista' e 'numero rivista' scegliendo la modalità dal menu a tendina del rigo 'Ordinamento'

## Realizzare la query "cerca argomento"

Procedere come già visto nell'esercizio 1 tenendo presenti le seguenti osservazioni:

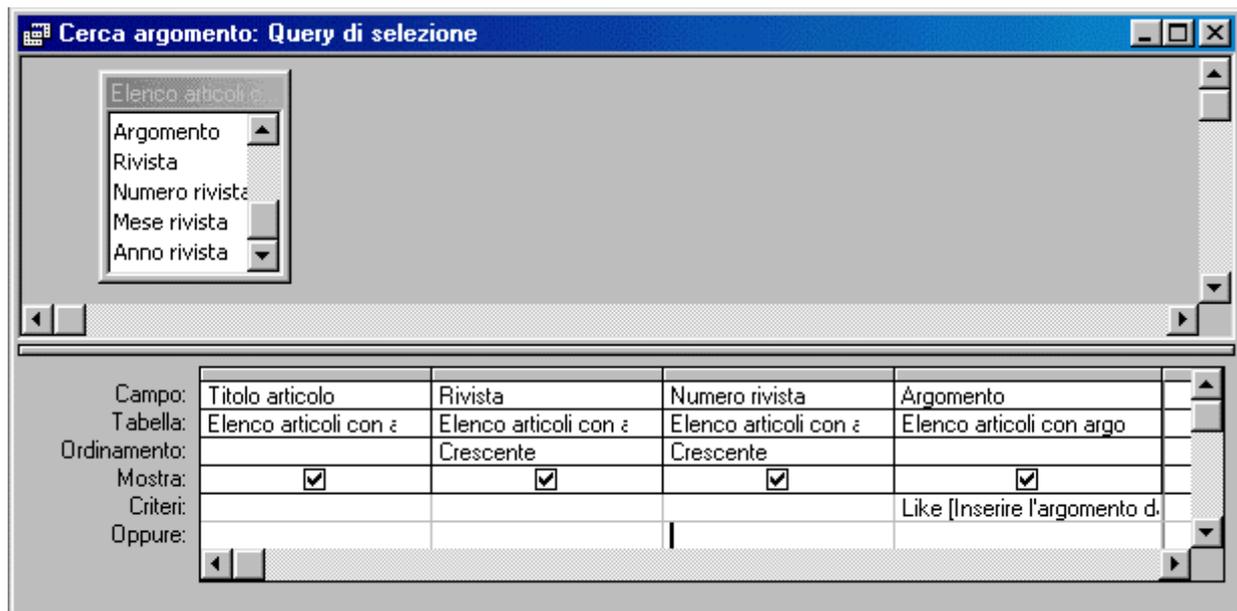
1. scegliere la query 'elenco articoli con argomenti e riviste' come origine dei dati (dalla finestra 'mostra tabella');



2. Inserire i campi 'Titolo articolo', 'Rivista', 'Numero rivista', 'Argomento'

Nel campo 'Criteri' è possibile inserire il criterio di ricerca esattamente come già visto

Il risultato dovrebbe essere:



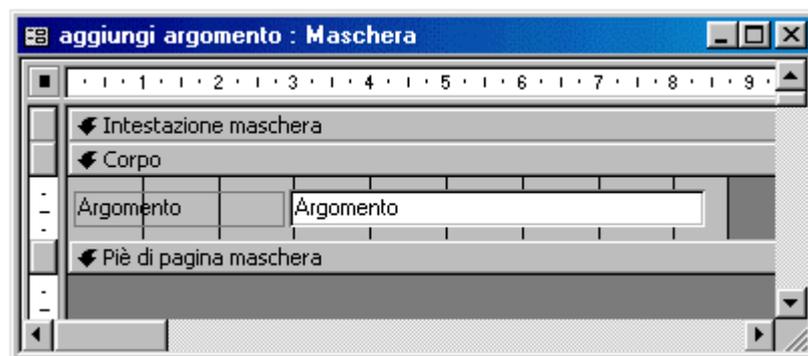
## Creare le maschere

### Realizzare la maschera "aggiungi argomento"

#### Descrizione

La maschera serve a creare la lista degli argomenti da collegare agli articoli.  
Essa mostra soltanto il campo argomento.  
Il campo ID viene gestito automaticamente da Access

La maschera dovrà avere il seguente aspetto:



#### Creazione della maschera

Procedere come visto precedentemente.

Scegliere 'autocomposizione maschera' e la tabella "elenco argomenti" come origine dei dati

Inserire nei campi selezionati soltanto 'argomento'

Scegliere il layout 'a colonne'

Scegliere lo stile 'standard'

Denominare la maschera "aggiungi argomento"

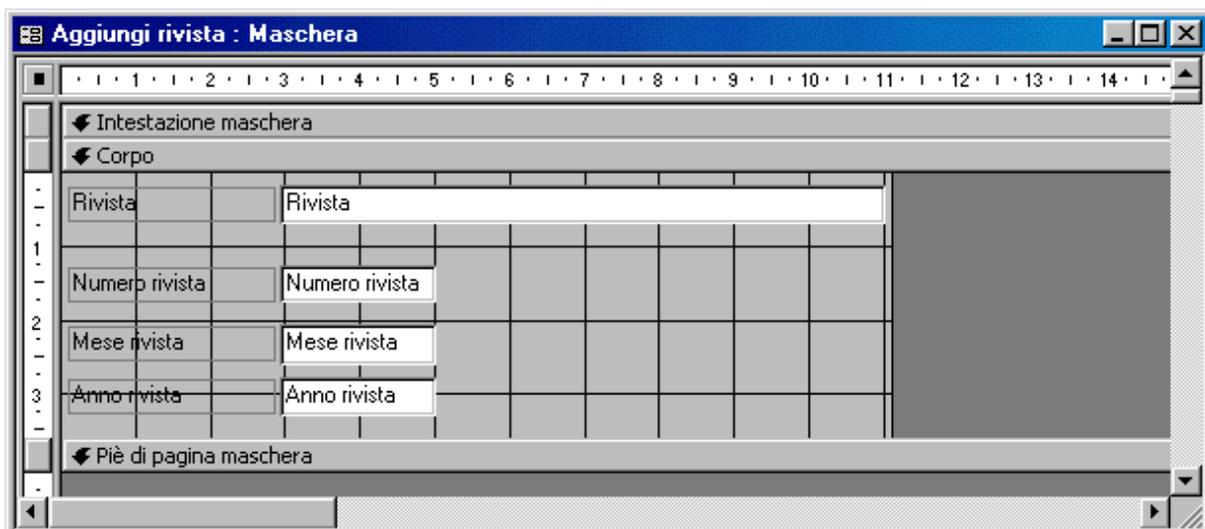
## Realizzare la maschera "aggiungi rivista"

### Descrizione

La maschera serve a creare la lista delle riviste da collegare agli articoli.  
Essa mostra soltanto i campi rivista, numero rivista, anno rivista, mese rivista.

Il campo ID viene gestito automaticamente da Access

La maschera dovrà avere il seguente aspetto:



### Creazione della maschera

Procedere come visto precedentemente

Scegliere 'autocomposizione maschera' e la tabella "elenco riviste" come origine dei dati

Inserire nei campi selezionati soltanto 'rivista', 'numero rivista', 'anno rivista', 'mese rivista'

Scegliere il layout 'a colonne'

Scegliere lo stile 'standard'

Denominare la maschera "aggiungi rivista"

## Realizzare la maschera "elenco articoli"

### Descrizione

La maschera serve ad introdurre gli articoli nell'archivio. Contemporaneamente essa permette di inserire i dati dell'articolo (titolo, pagina, ...), la rivista in cui si trova e l'argomento. Per questa ragione questa maschera interagisce anche con i dati delle tabelle 'elenco riviste' ed 'elenco argomenti'

La maschera dovrà avere il seguente aspetto finale:

Titolo articolo	Funzionamento ed uso del TCP-IP			IDArgomento	2
Rubrica	-----			IDRivista	1
Parole chiave	subnet mask, TCP-IP, internet, reti locali,				
Note	subnet mask, winIpCfg, DHCP, WINS,				
Pagina	330				
Argomento	Internet, reti, telematica				
Rivista	MC Microcomputer	169	GEN 1997		
	PC Professionale	54	GEN 1997		
	PC Professionale	68	DIC 1996		

## Creazione della prima versione della maschera

Procedere come visto precedentemente

Scegliere 'autocomposizione maschera' e la query "elenco articoli" come origine dei dati  
Inserire nei campi selezionati tutti tranne 'ID'

Scegliere il layout 'a colonne'

Scegliere lo stile 'standard'

Denominare la maschera "Introduci articoli"

## Miglioramenti alla maschera

## Descrizione

In questa situazione la maschera richiede l'introduzione dei riferimenti numerici alla rivista e all'argomento. Ovviamente tale procedura è tutt'altro che semplice

Per migliorare l'uso della maschera si introdurranno due controlli aggiuntivi: una casella combinata per l'argomento ed una casella di riepilogo per la rivista

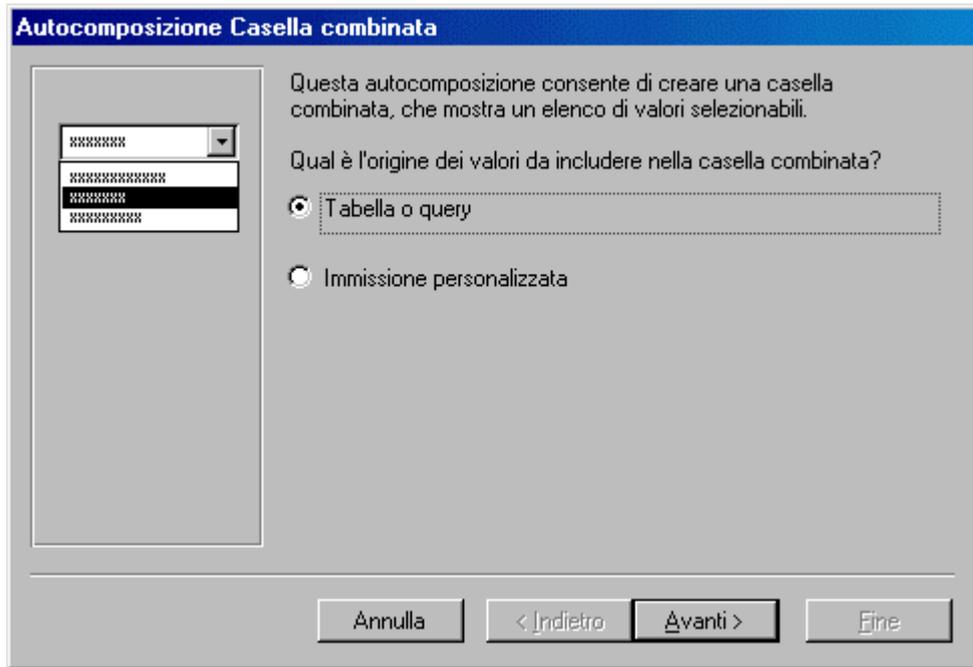
### Casella combinata per l'argomento

Per inserire la casella combinata usare il bottone  della palette degli strumenti:



dopo aver tracciato sulla maschera il controllo partirà l'autocomposizione.

Nella finestra seguente selezionare 'Tabella o query' come origine dei dati per il controllo che si sta creando



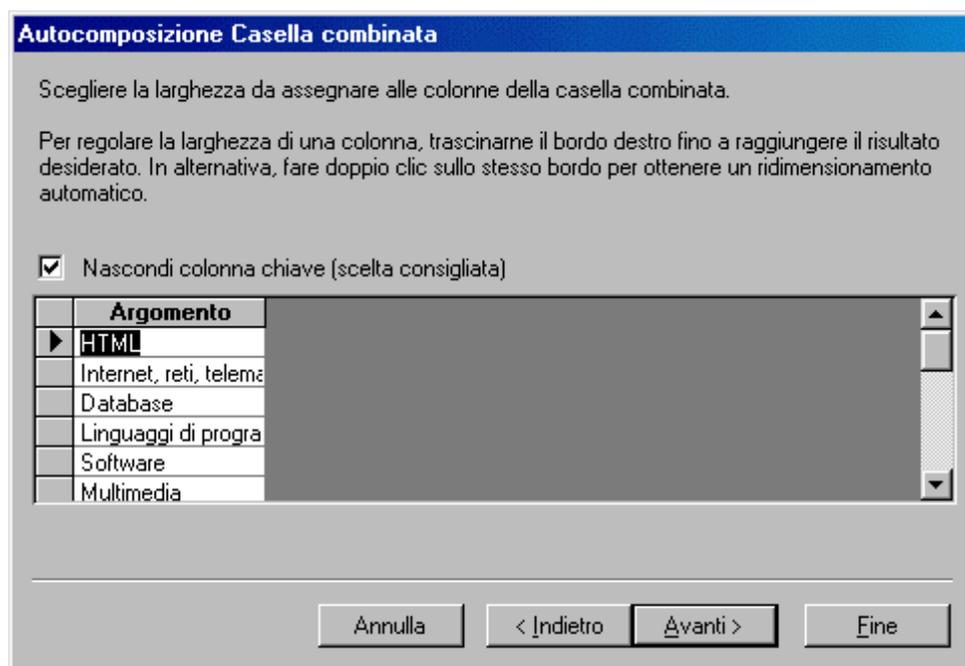
Nella finestra seguente selezionare la tabella 'elenco argomenti'



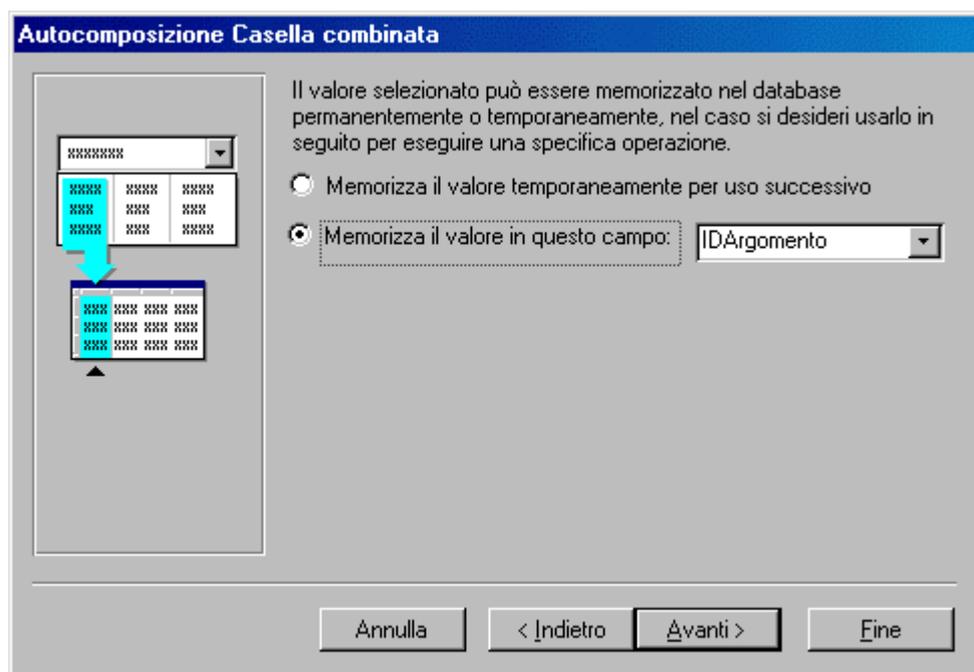
Nella finestra seguente selezionare i campi 'ID' e 'argomento' per adoperarli come origine dei dati della casella combinata



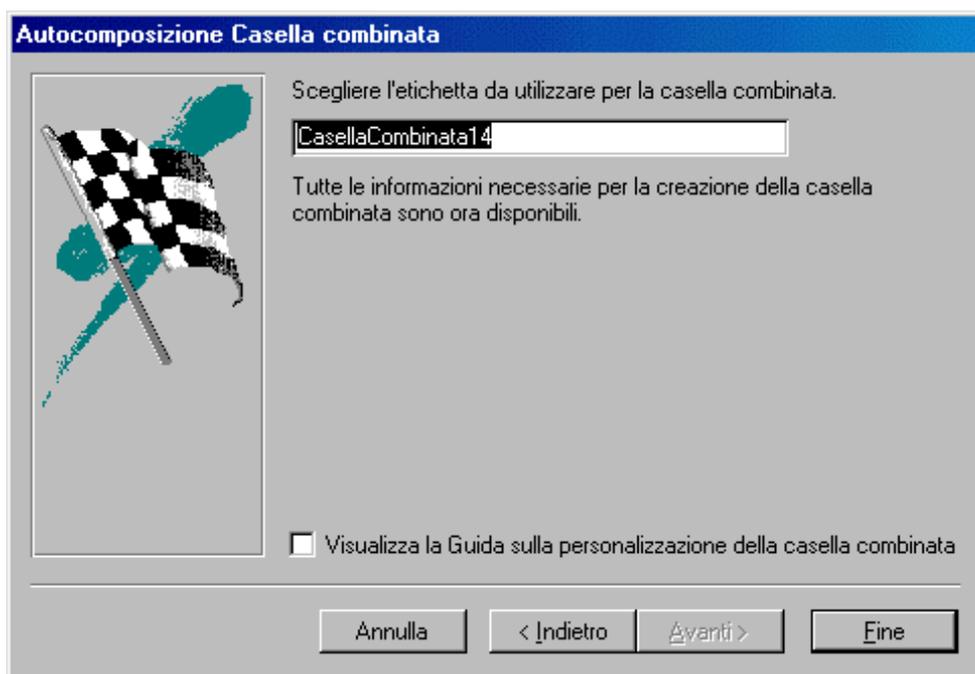
Nella finestra seguente assicurarsi che sia selezionata la casella 'Nascondi colonna chiave'



Nella finestra seguente bisogna selezionare in quale campo (di quelli presenti nella maschera) verranno immessi i valori selezionati da questa casella. Selezionare il campo 'IDArgomento'



Nella finestra seguente può essere scelto il nome da dare al bottone.



Fare click su Fine per completare la procedura

## Casella di riepilogo per la rivista

Per inserire la casella di riepilogo usare il bottone  della palette degli strumenti

dopo aver tracciato sulla maschera il controllo partirà l'autocomposizione (procedere come già visto):

- scegliere come origine dei dati la tabella 'elenco riviste'

- inserire tutti i campi tra quelli selezionati
- accertarsi che sia selezionata la casella 'Nascondi colonna chiave'
- Memorizzare il valore nel campo "IDRivista"

## Bottoni 'aggiungi argomento', 'aggiungi rivista', 'aggiorna'

Per permettere l'aggiunta di un argomento o di una rivista alle liste, durante l'immissione degli articoli, si introdurranno dei bottoni per aprire le maschere 'Aggiungi argomenti' e 'Aggiungi rivista'

Procedere come già visto nell'esercizio 1

Il bottone aggiorna serve ad aggiornare la lista delle riviste e degli argomenti

Nell'autocomposizione del bottone scegliere 'operazioni su maschere' e poi 'Aggiorna dati maschera'

La maschera dovrebbe avere l'aspetto finale seguente

The screenshot shows a window titled "elenco articoli" with the following fields and values:

- Titolo articolo: Funzionamento ed uso del TCP-IP
- Rubrica: .....
- Parole chiave: subnet mask, TCP-IP, internet, reti locali,
- Note: subnet mask, winIpCfg, DHCP, WINS,
- Pagina: 330
- Argomento: Internet, reti, telematica
- Rivista: MC Microcomputer 169 GEN 1997

Additional fields on the right:

- IDArgomento: 2
- IDRivista: 1

Buttons: Aggiorna, Aggiungi argomento, Aggiungi rivista

Record: 1 di 26

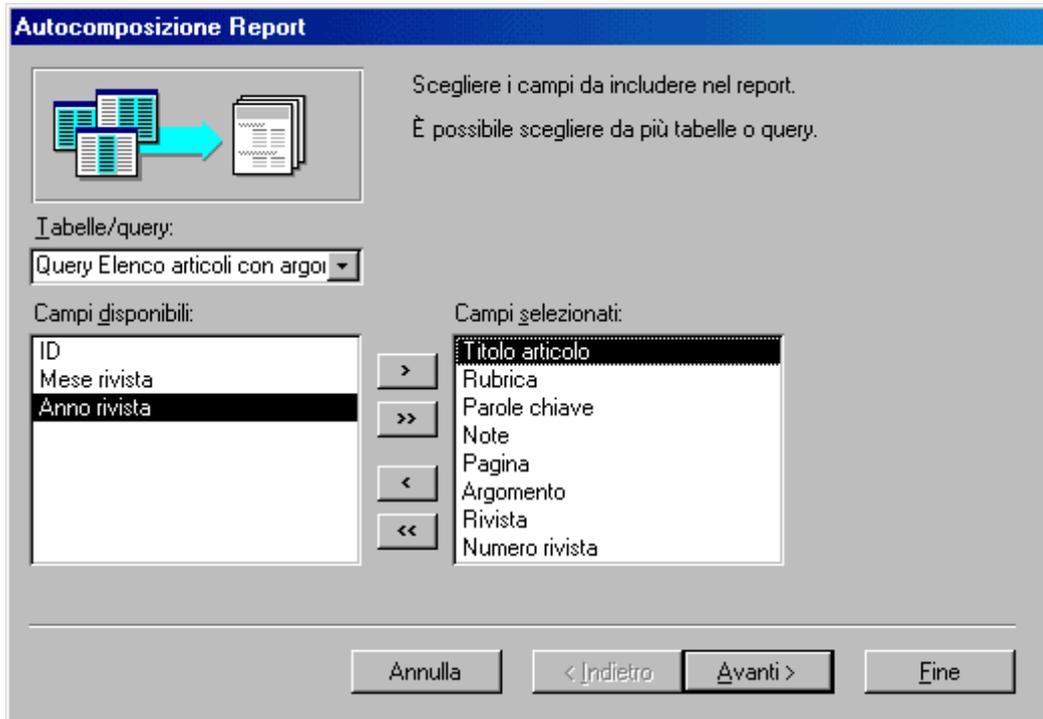
## Report basato sulle relazioni

Dalla finestra del database scegliere la cartella 'Report'

Fare click sul bottone Nuovo

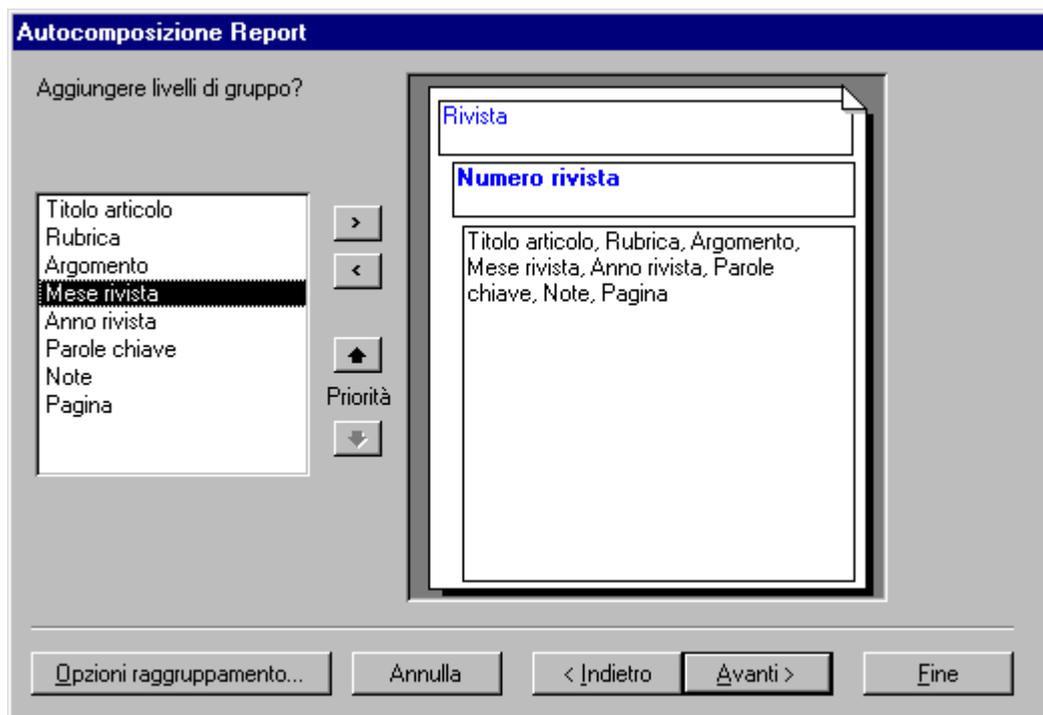
Nella finestra 'Nuovo report' scegliere 'Autocomposizione report' e poi selezionare la tabella 'elenco articoli con argomenti e riviste' dal menu a tendina in basso

Nella prima delle finestre 'Autocomposizione report' spostare tutti i record tranne 'ID', 'mese rivista' e 'anno rivista' nella sezione 'campi selezionati'



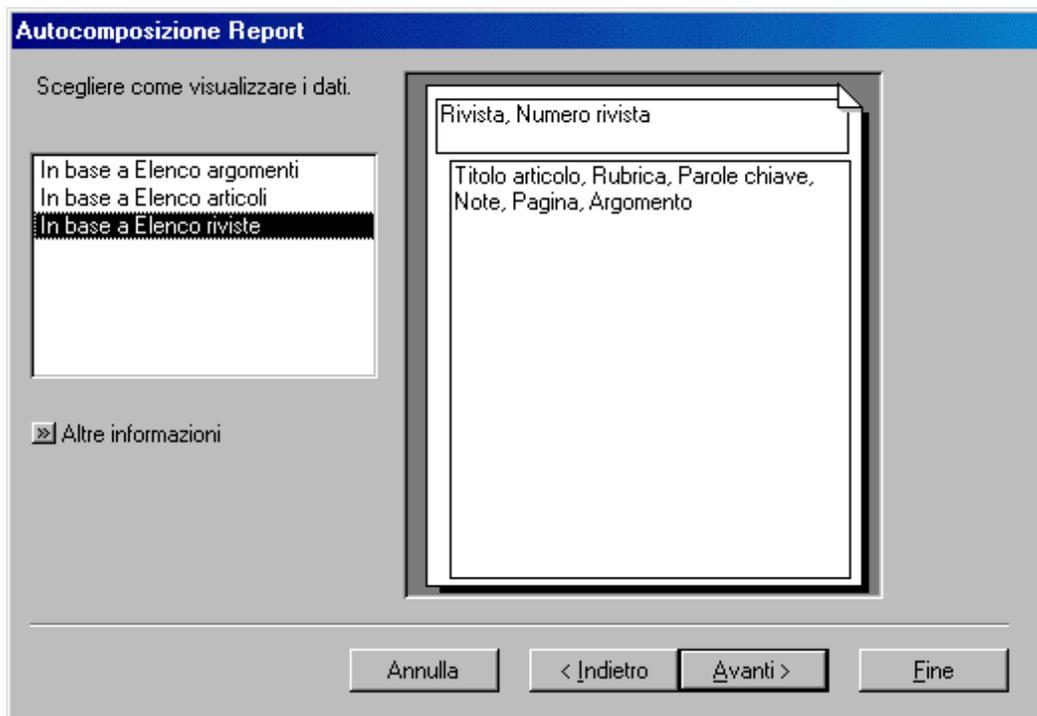
Fare click sul bottone Avanti

Nella finestra seguente raggruppare secondo 'Rivista' e 'Numero rivista'



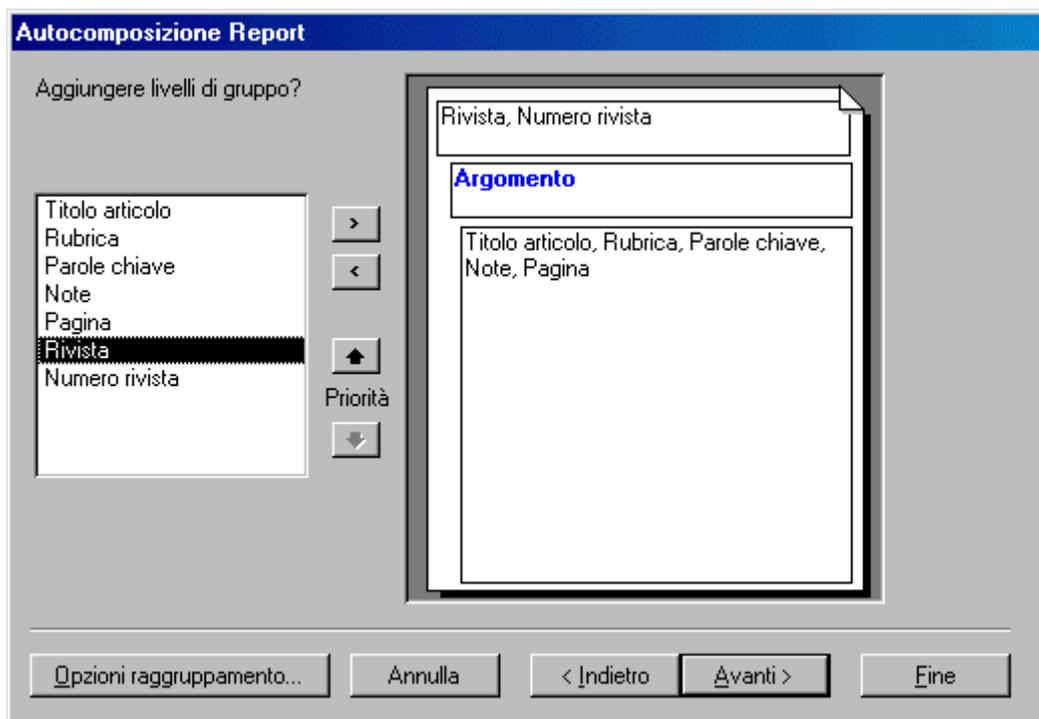
Fare click sul bottone Avanti

Nella finestra seguente scegliere di raggruppare i dati 'in base ad elenco riviste'



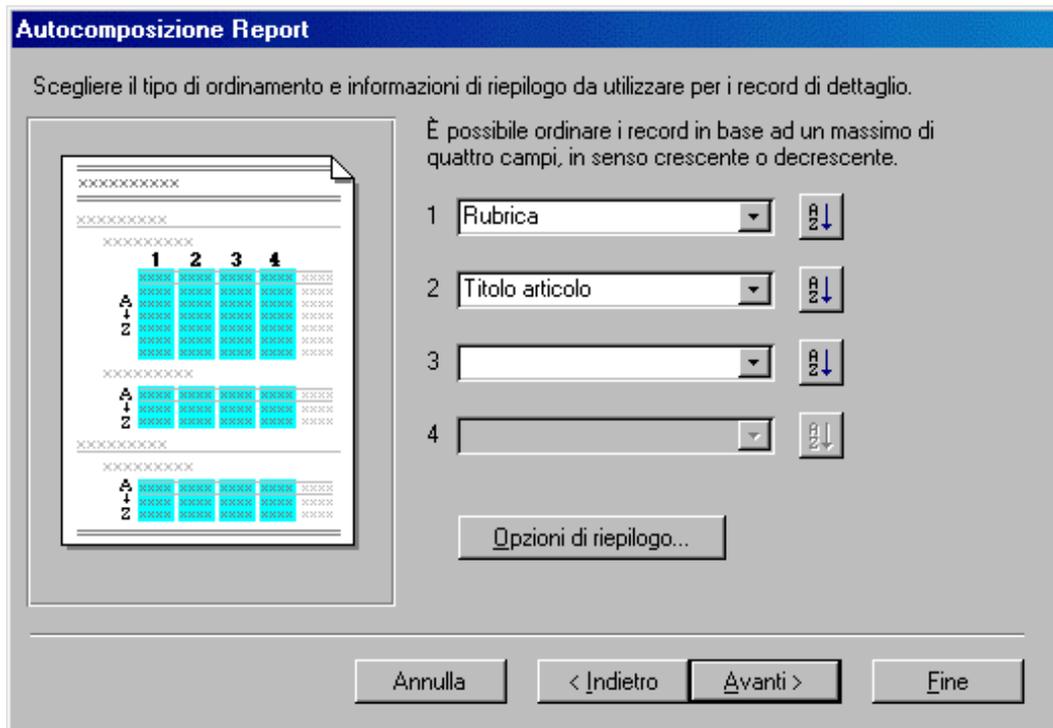
Fare click sul bottone Avanti

Nella finestra seguente scegliere 'argomento' e fare click sulla freccia di spostamento a destra per raggruppare i dati in base a questo campo



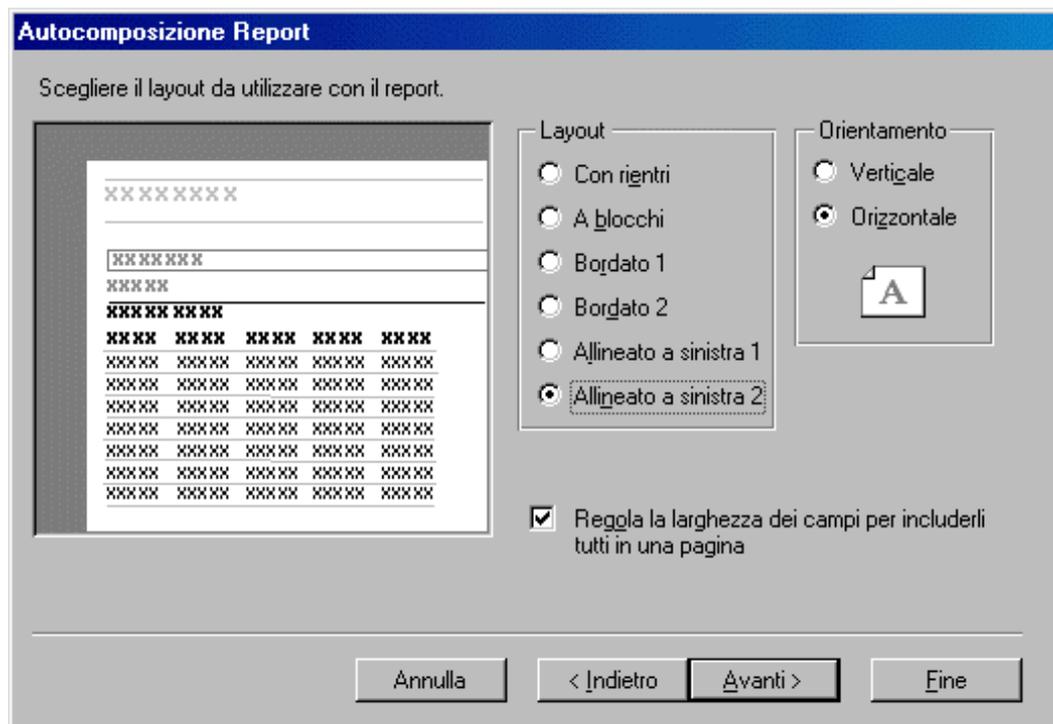
Fare click sul bottone Avanti

Nella finestra seguente scegliere di ordinare i dati in base a 'Rubrica' e a 'Titolo articolo' scegliendo i rispettivi campi dai menu a tendina



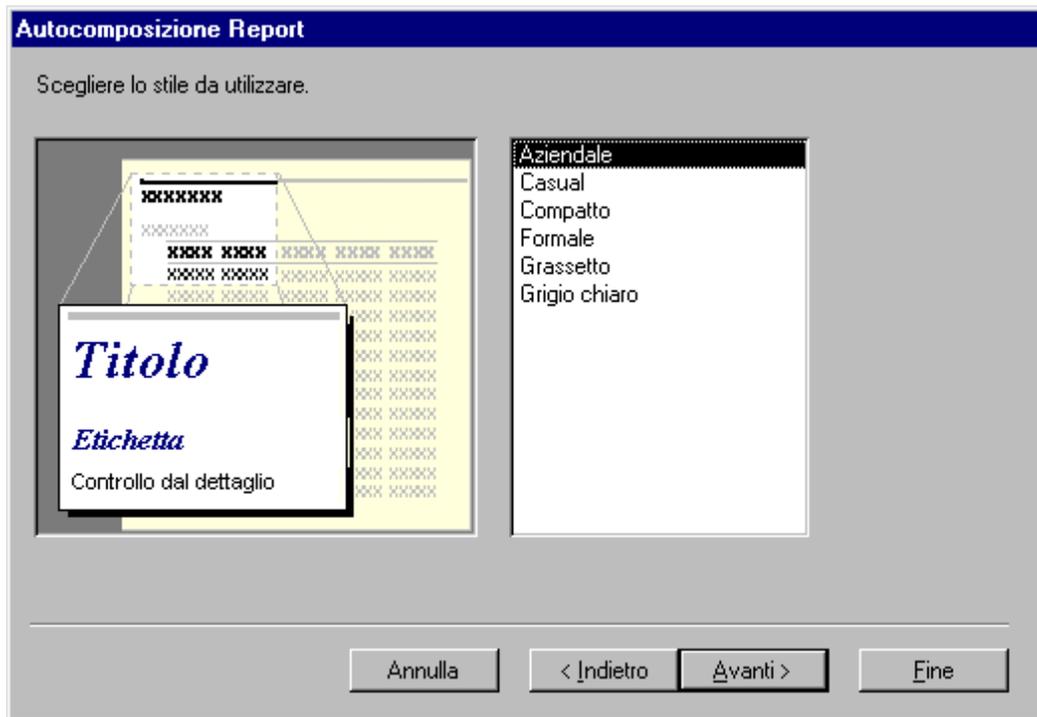
Fare click sul bottone Avanti

Nella finestra seguente scegliere il layout 'Allineato a sinistra 1', orientamento 'orizzontale', selezionare 'Regola la larghezza...'



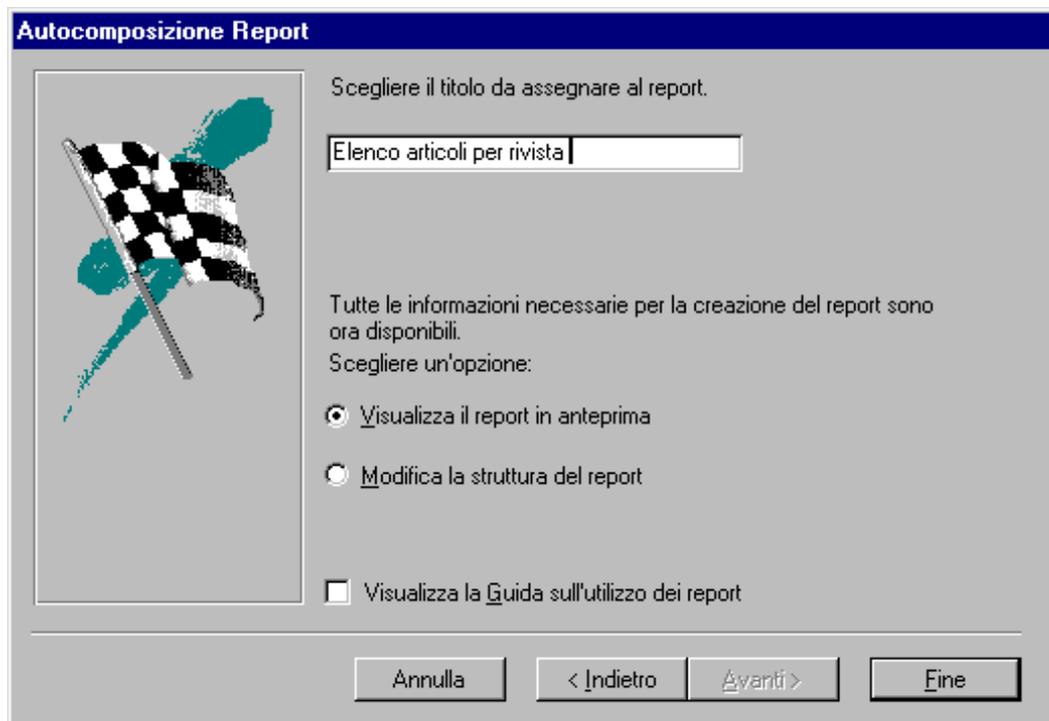
Fare click sul bottone Avanti

Dalla finestra seguente scegliere lo stile 'Aziendale' dalla lista



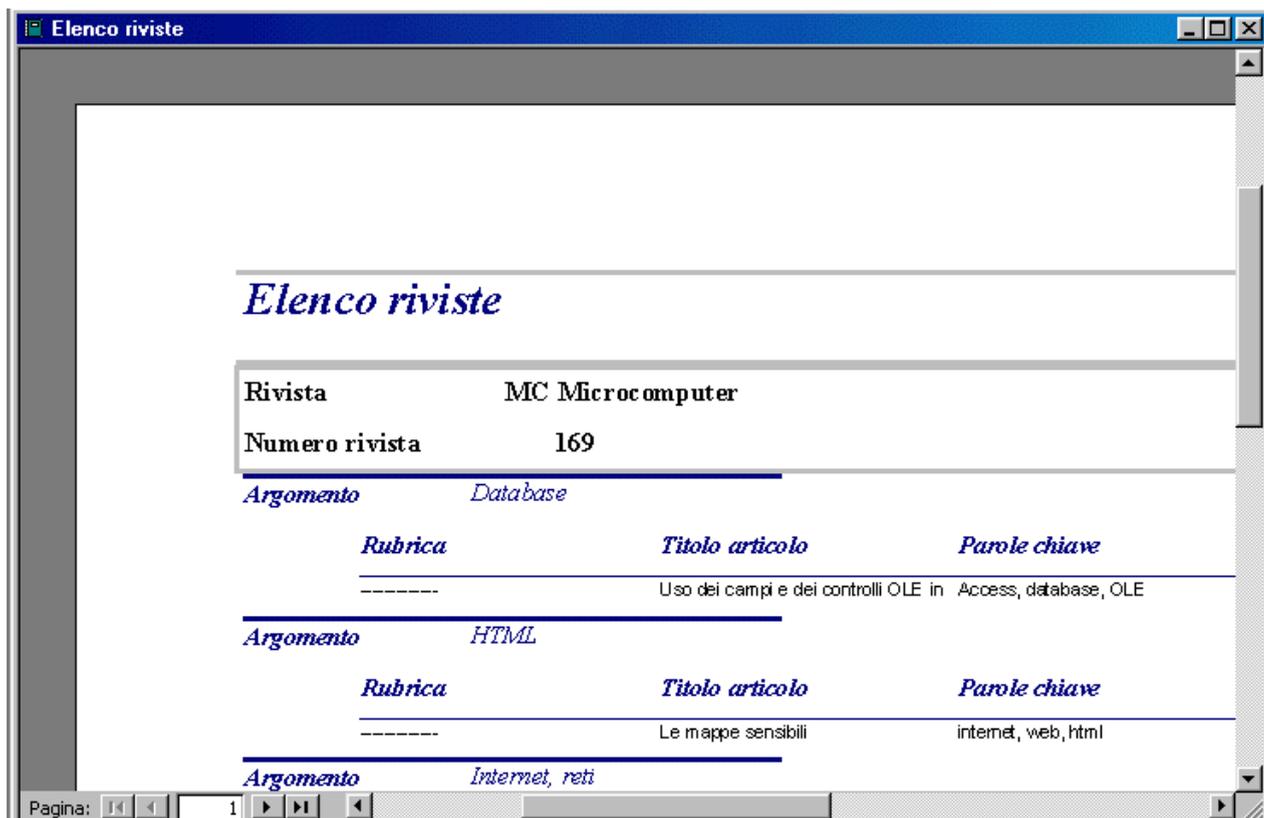
Fare click sul bottone Avanti

Nell'ultima finestra scegliere il nome da dare al report ('Elenco articoli')



Fare click sul bottone Fine

Apparirà l'anteprima del report:

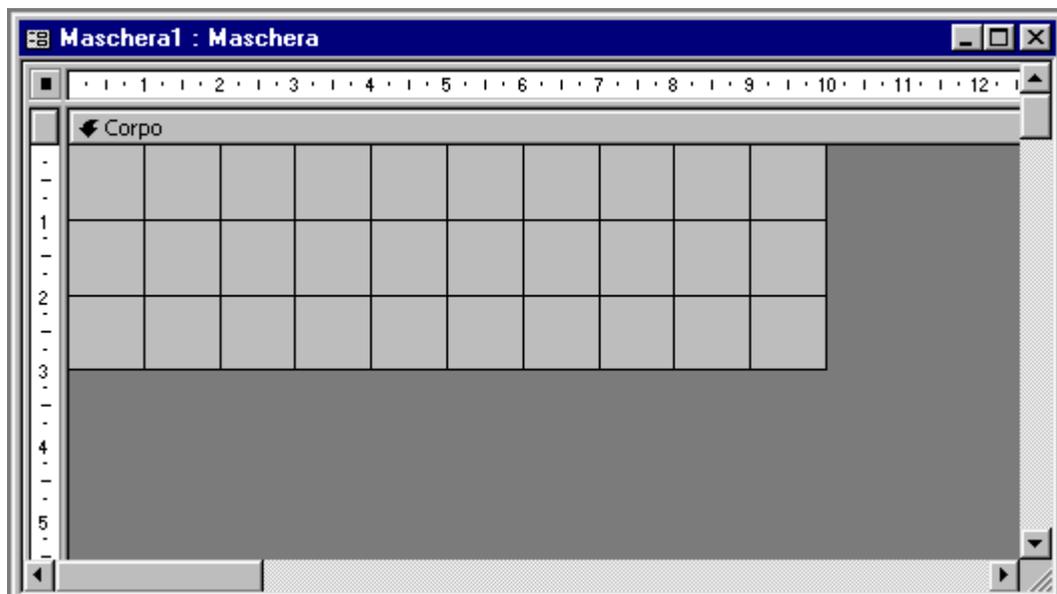


## Creare la maschera 'menu'

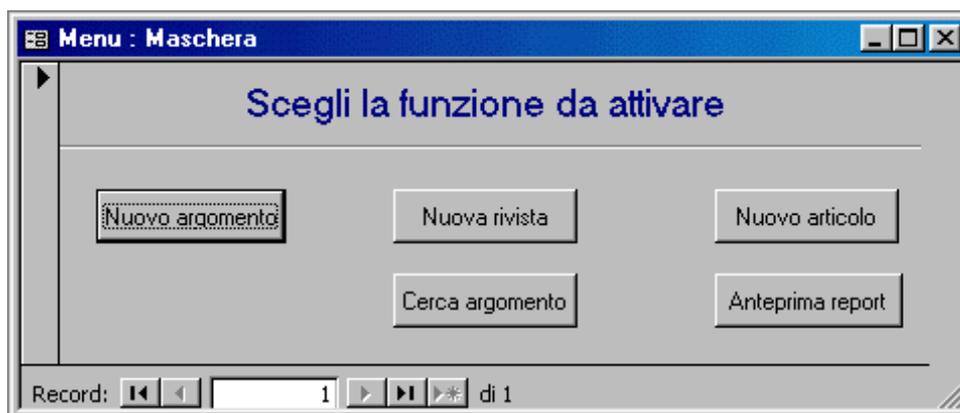
Dalla finestra del database scegliere la cartella 'maschere' e poi il bottone nuovo

Scegliere 'Visualizzazione struttura' e nessuna origine dei dati

Apparirà la maschera in modalità struttura



La maschera finita avrà il seguente aspetto:



Inserire il titolo con il bottone , colorare il testo e scegliere le dimensioni del carattere.

Inserire la linea orizzontale con il bottone , impostare lo stile tridimensionale con il menu degli effetti 3D:



I bottoni 'Nuovo argomento', 'nuova rivista', 'Nuovo articolo' aprono rispettivamente le maschere: 'aggiungi argomento', 'aggiungi rivista' e 'introduci articoli'. Procedere [come visto](#) a proposito dell'esercizio 1

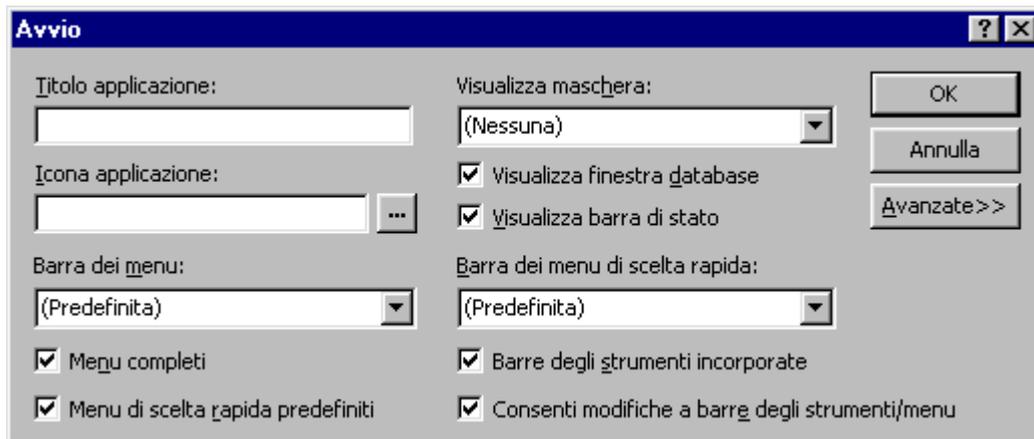
Il bottone 'cerca argomento' avvia la query omonima come già visto in esercizio 1

Il bottone 'Anteprima report' apre il report come visto nell'esercizio 1

Salvare la finestra (menu File, Salva) e scegliere il nome "Menu"

## Opzioni di avvio

Per mostrare la maschera menu all'apertura del file bisogna scegliere la finestra "menu" dal menu a tendina "Visualizza maschera"



Impostare il titolo dell'applicazione ("Elenco articoli delle riviste di informatica")

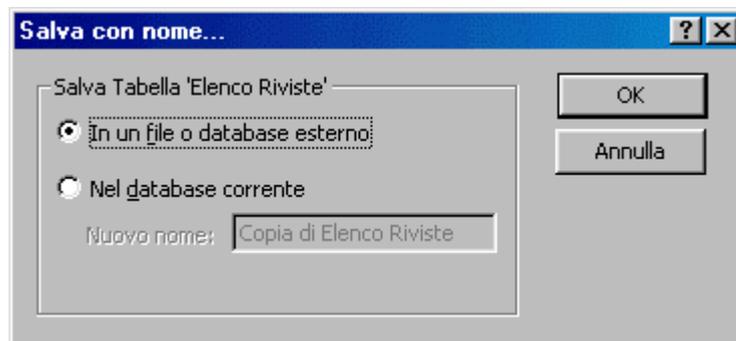
## Esportazione di tabelle e dati

### Per esportare i dati di una tabella in un file Excel

Selezionare con un click la tabella dalla finestra principale del database

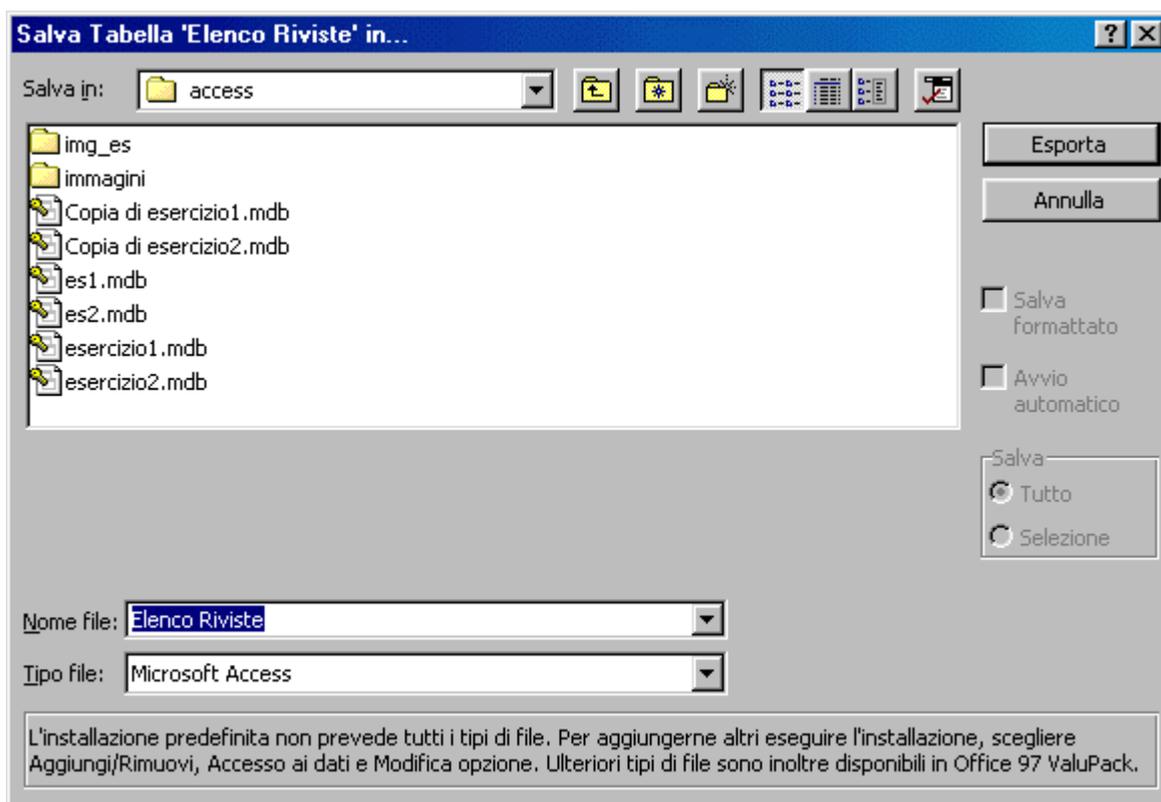
Dal menu 'file' scegliere 'Salva con nome/esporta...'

Apparirà la finestra seguente:



Selezionare 'In un file o database esterno'

Apparirà quindi la finestra:



Dal menu a tendina 'Tipo File' scegliere 'Microsoft Excel 97' e fare click sul bottone 'Esporta'