

GESTIONE PROVE STUDENTI

In un database si vogliono organizzare le informazioni sulle prove sostenute dagli studenti di una scuola nelle diverse materie e con i diversi docenti : si devono quindi conservare i dati degli studenti, dei docenti, delle materie e delle prove in modo da rendere possibili le seguenti interrogazioni :

1)	L'elenco completo di tutte le prove sostenute da tutti gli studenti della scuola;
2)	L'elenco delle prove di uno studente;
3)	L'elenco delle prove effettuate da un docente in una classe;
4)	I voti di uno studente in una materia;
5)	I voti di una classe nella prova svolta in una materia in un determinato giorno;
6)	Le date ed il tipo delle prove effettuate da un docente in una materia;
7)	Le date ed il tipo di prove effettuate da un docente in una classe;
8)	La media dei voti di ogni alunno in ciascuna materia;
9)	L'elenco del numero degli alunni per ogni classe;
10)	L'elenco, per una data classe, del numero di prove diverse sostenute da ciascun studente;

Realizzare :

- ✧ uno schema concettuale della base di dati;
- ✧ uno schema logico della base di dati;
- ✧ la definizione delle relazioni della base di dati in linguaggio SQL;
- ✧ le precedenti interrogazioni espresse in linguaggio SQL.

➤ **Analisi dei dati**

Le entità che possono essere individuate per risolvere il problema sono :

Tipo prova	contiene la classificazione delle diverse prove che uno studente può sostenere;
Materia	contiene l'elenco delle materie oggetto delle verifiche;
Classe	contiene l'elenco di tutte le classi della scuola ;
Comune	Contiene informazioni sui Comuni di Italia
Docente	contiene informazioni anagrafiche dei docenti della scuola;
Studente	contiene informazioni anagrafiche degli studenti della scuola;

Per l'entità **Tipo prova** gli attributi, che servono a classificare le diverse prove sostenute dallo studente, sono : il codice e la descrizione del tipo di prova (orale, scritta, pratica, grafica, ecc.);

Per l'entità **Materia** gli attributi sono : il codice e la descrizione della materia;

Per l'entità **Classe** gli attributi sono : il codice e la sigla di tutte le classi della scuola;

Per l'entità **Comune** gli attributi riguardano le informazioni che riguardano il codice del comune, la denominazione, il CAP, la Provincia.

Per l'entità **Docente** gli attributi riguardano i dati anagrafici del docente e quindi, oltre al codice, Cognome, Nome, gli altri attributi possono essere : l'indirizzo, il numero di telefono;

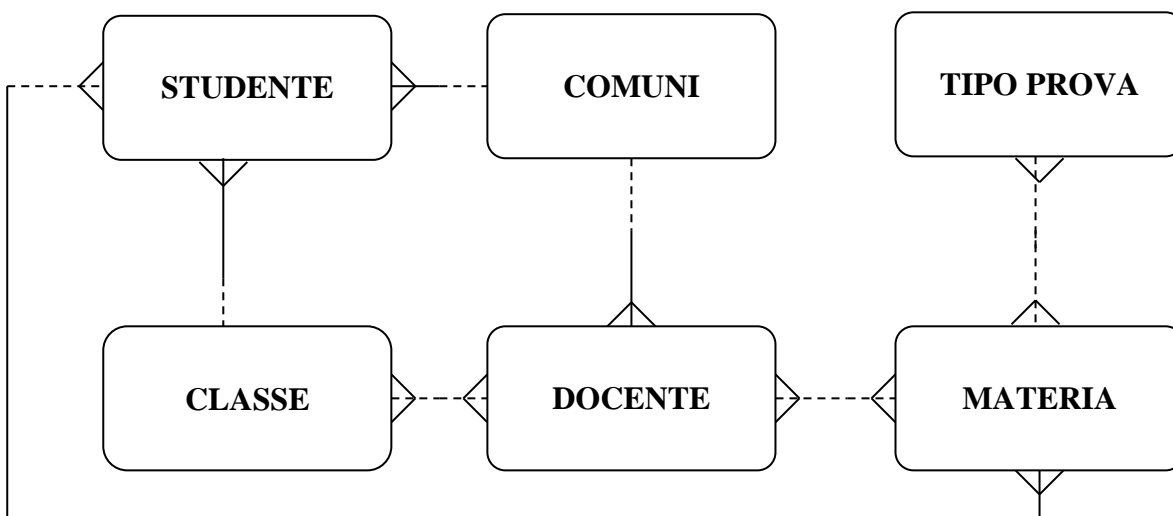
Per l'entità **Studente** gli attributi riguardano i dati anagrafici dello studente e quindi, oltre al codice, Cognome, Nome, gli altri attributi possono essere : l'indirizzo, il numero di telefono. Un altro attributo fondamentale per gestire lo studente nell'ambito scolastico è sicuramente la classe di appartenenza;

Associazioni fra le entità :

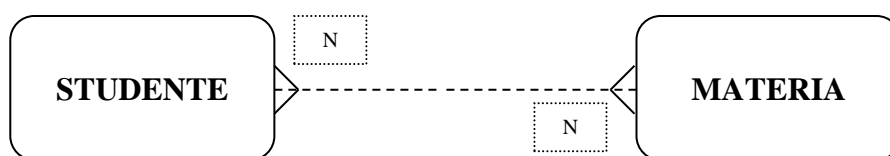
- ✦ Tra l'entità *Docente* e l'entità *Classe* per una scuola è prevista un'associazione molti-a-molti in quanto ogni docente può insegnare in una o più classi, ed ogni classe deve avere uno o più docenti.
- ✦ Tra l'entità *Docente* e l'entità *Materia* è prevista un'associazione molti-a-molti in quanto un docente può insegnare una o più materie, ed una materia deve essere insegnata da uno o più docenti.
- ✦ Tra l'entità *Studente* e l'entità *Classe* per una scuola è prevista un'associazione uno-a-molti in quanto ogni studente deve appartenere ad una classe ed ogni classe deve avere uno o più studenti.
- ✦ Tra l'entità *Studente* e l'entità *Materia* è prevista un'associazione molti-a-molti in quanto uno studente può essere verificato su una o più materie, ed una materia può essere oggetto di verifica di uno o più studenti.
- ✦ Tra l'entità *Tipo_Prova* e l'entità *Materia* vi è un'associazione multi-a-molti, perché per ogni materia ci possono essere più tipi di verifica e per ogni tipo di verifica ci possono essere più materie.

Sulla base dell'analisi effettuata si può disegnare il modello Entità/Relazioni :

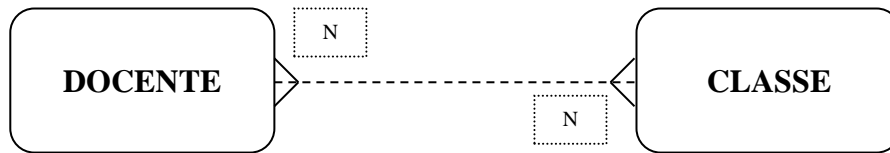
➤ **Modello E/R**



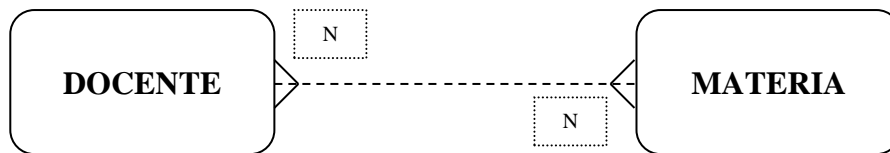
L'associazione molti-a-molti che sussiste tra le entità *Materia* e *Studente* deve essere scomposta in due associazioni uno-a-molti, per consentire la rappresentazione degli attributi dell'associazione. Per esempio nell'associazione fra l'entità *Studente* e l'entità *Materia* la data della prova ed il voto ottenuto dallo studente non sono attributi né dell'entità *Studente*, né dell'entità *Materia*. Nel modello E/R, quindi, l'associazione suddetta viene realizzata introducendo una terza entità *Prova*, avente come attributi la data ed il voto della verifica e inoltre, il codice dello studente ed il codice della materia, ovvero gli attributi chiave delle due entità considerate.



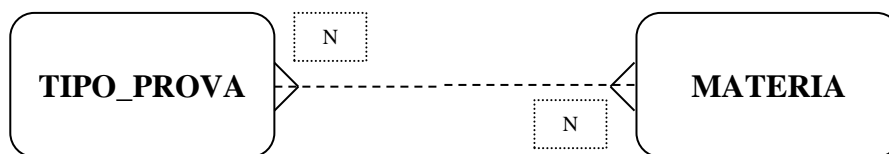
L'associazione multi-a-molti che sussiste tra le entità Docente e Classe deve essere scomposta in due associazioni uno-a-molti, per consentire la rappresentazione degli attributi dell'associazione. Per esempio nell'associazione fra l'entità *Docente* e l'entità *Classe* il numero delle ore di insegnamento del docente in una classe non è attributo né di *Docente*, né di *Classe*, ma dell'associazione. Nel modello E/R, quindi, l'associazione suddetta viene realizzata introducendo una terza entità *Insegna*, avente come attributi il numero delle ore di insegnamento ed, inoltre, il codice del docente ed il codice della classe, ovvero gli attributi chiave delle due entità considerate.



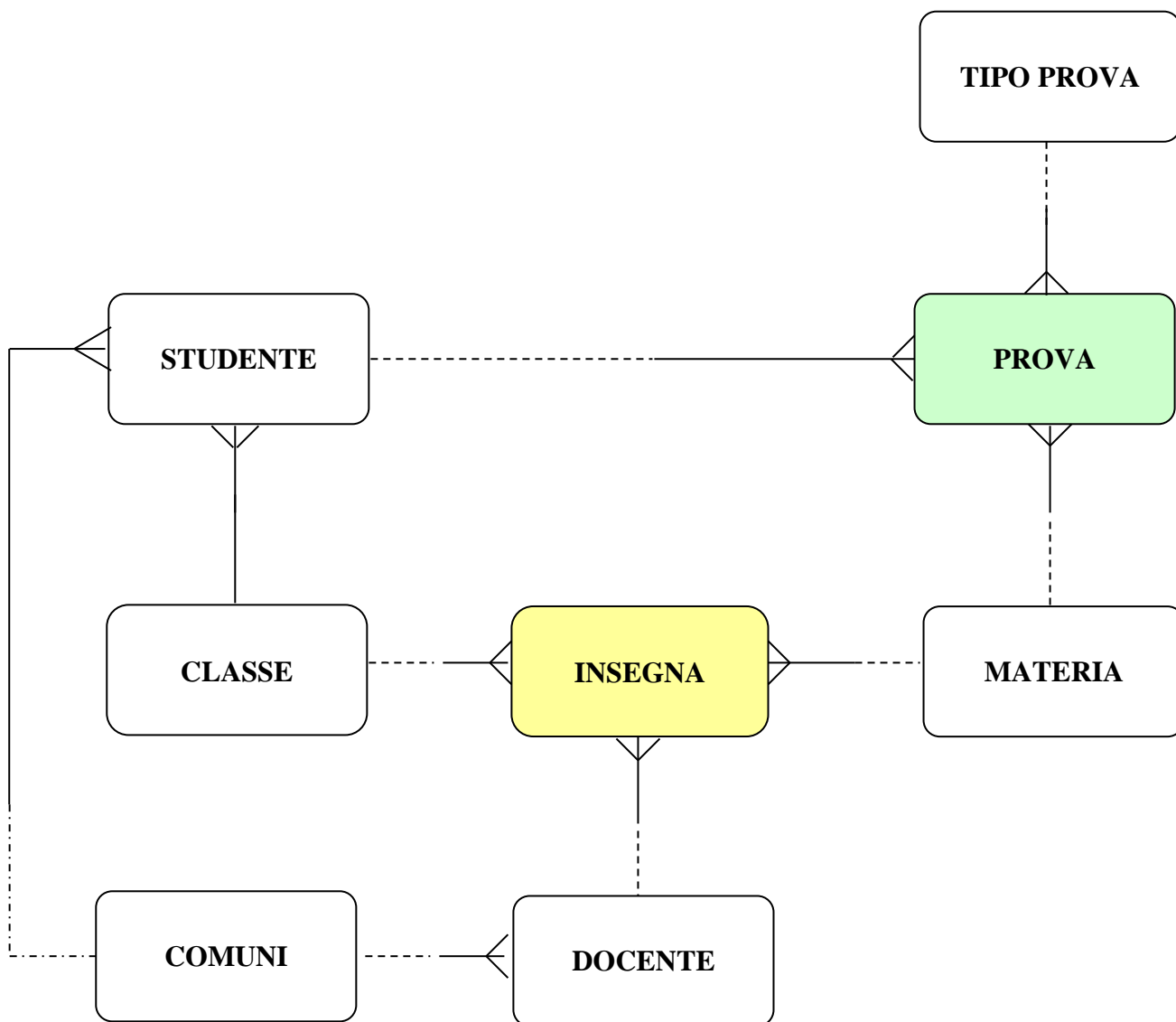
Fra le entità previste dall'analisi dei dati esistono l'entità *Docente* e l'entità *Materia* per le quali sussiste un'associazione multi-a-molti. Per realizzare tale associazione è consigliabile utilizzare l'entità *Insegna*, già utilizzata per realizzare l'associazione fra l'entità *Docente* e l'entità *Classe*, aggiungendo il codice della materia, ovvero l'attributo chiave dell'entità *Materia* (il codice del docente è già presente!).



Fra le entità previste dall'analisi dei dati esistono l'entità *Tipo_prova* e l'entità *Materia* per le quali sussiste un'associazione multi-a-molti. Per realizzare tale associazione è consigliabile utilizzare l'entità *Prova*, aggiungendo il codice dello studente, ovvero l'attributo chiave dell'entità *Studente*.



Il modello completo diventa :



L'associazione tra *Prova* e *TipoProva* è di tipo uno-a-molti, in quanto ogni tipologia di prova può essere associata ad una o più prove ed ogni prova deve essere di un solo tipo.

L'associazione tra *Studiante* e *Prova* è di tipo uno-a-molti, in quanto ogni studente può essere verificato con una o più prove ed ogni prova deve essere sostenuta da un solo studente.

L'associazione tra *Materia* e *Prova* è di tipo uno-a-molti, perché ogni materia può essere controllata con una o più prove ed ogni prova deve essere riferita ad una sola materia.

L'associazione tra *Docente* e *Insegna* è di tipo uno-a-molti, in quanto ogni docente può insegnare in una o più classi ed in una classe ci deve essere un docente per ogni materia.

L'associazione tra *Materia* e *Insegna* è di tipo uno-a-molti, in quanto ogni materia può essere insegnata in una o più classi ed in una classe ci deve essere una materia per ogni docente assegnato alla classe.

Nel modello E/R sono stati indicati i nomi delle entità, l'opzionalità od obbligatorietà delle associazioni.

Utilizzando le regole di derivazione si passa dal modello concettuale al modello logico, cioè alle tabelle, introducendo le chiavi esterne per rappresentare le associazioni.

Per realizzare l'associazione uno a molti fra l'entità *Prova* l'entità *TipoProva* si introduce, fra gli attributi della tabella *tbProve*, la chiave esterna **Cod_tipoprova** associata alla chiave primaria **Id_tipoProva** della **tbTipoProve**.

Per realizzare l'associazione uno a molti fra l'entità *Prova* e l'entità *Studente* si introduce, fra gli attributi della tabella *tbProve*, la chiave esterna **Cod_studente** associata alla chiave primaria **Id_studente** della *tbStudenti*.

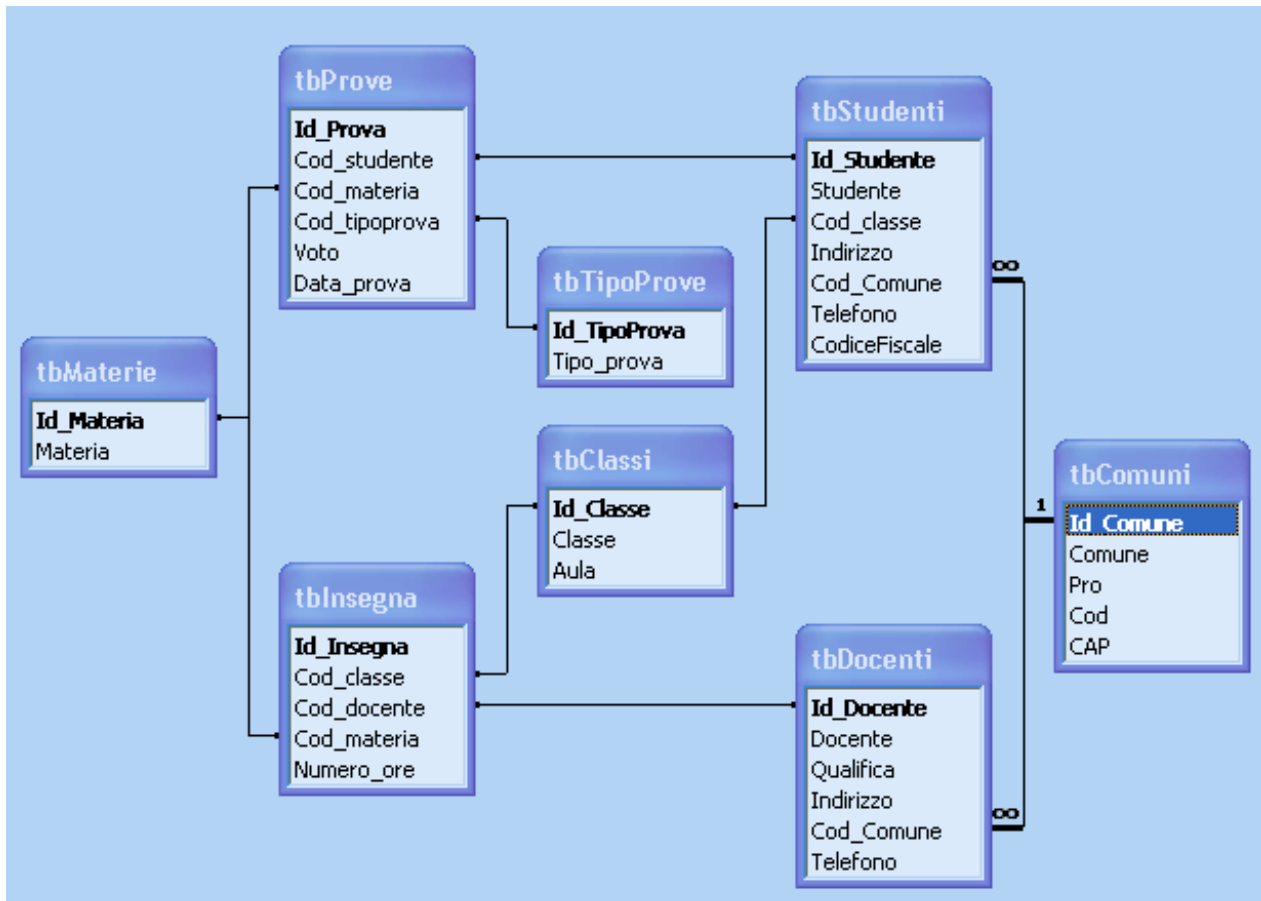
Per realizzare l'associazione uno a molti fra l'entità *Prova* e l'entità *Materia* si introduce, fra gli attributi della tabella *tbProve*, la chiave esterna **Cod_materia** associata alla chiave primaria **Id_materia** della *tbMaterie*.

Per realizzare l'associazione uno a molti fra l'entità *Studente* e l'entità *Classe* si introduce, fra gli attributi della tabella *tbStudenti*, la chiave esterna **Cod_classe** associata alla chiave primaria **Id_classe** della *tbClassi*.

Per realizzare l'associazione uno a molti fra l'entità *Insegna* e l'entità *Classe* si introduce, fra gli attributi della tabella *tbInsegna*, la chiave esterna **Cod_classe** associata alla chiave primaria **Id_classe** della *tbClassi*.

Per realizzare l'associazione uno a molti fra l'entità *Insegna* e l'entità *Materia* si introduce, fra gli attributi della tabella *tbInsegna*, la chiave esterna **Cod_materia** associata alla chiave primaria **Id_Materia** della *tbMaterie*.

Per realizzare l'associazione uno a molti fra l'entità *Insegna* e l'entità *Docente* si introduce, fra gli attributi della tabella *tbInsegna*, la chiave esterna **Cod_docente** associata alla chiave primaria **Id_Docente** della *tbDocenti*.



➤ **Schema logico – Le Tabelle**

Nel definire le tabelle e la struttura delle stesse, è consigliabile elencarle partendo dalle tabelle che non presentano chiavi esterne.

In successione si definiscono le tabelle che presentano chiavi esterne. Si evidenzia con il grassetto le chiavi primarie e in corsivo le chiavi esterne. Per ogni tabella si è usato come chiave primaria un campo contatore, non informativo, utile per la realizzazione delle associazioni e soprattutto molto utili nel caso di applicativi realizzati in linguaggi di tipo visuale, come ad es. Visual Basic, per realizzare in modo agevole i collegamenti con le tabelle mediante componenti tipo combobox o datacombo.

Nome Tabella	Nome dei campi
TbTipoProve	Id_tipoProva , Tipo_prova
TbMaterie	Id_materia , Materia
TbClassi	Id_classe , Classe, Aula
TBComuni	Id_Comune , Comune, Pro, Cod, CAP
TbStudenti	Id_studente , Studente, <i>Cod_classe</i> , Indirizzo, <i>Cod_Comune</i> , Telefono
TbDocenti	Id_docente , Docente, Indirizzo, <i>Cod_Comune</i> , Telefono
TbInsegna	Id_Insegna, <i>Cod_classe</i> , <i>Cod_docente</i> , <i>Cod_materia</i> , Numero_ore
TBProve	Id_prova, <i>Cod_studente</i> , <i>Cod_materia</i> , <i>Cod_tipoprova</i> , Voto, Data_prova

➤ Schema logico - Descrizione dei dati

Tabella	Campo	Chiave	Tipo	Dim.	Descrizione
tbTipoProve	Id_tipoProva	Primaria	Numerico	4	
	Tipo_prova		Testo	15	orale,scritto,ecc.
tbMaterie	Id_materia	Primaria	Numerico	4	
	Materia		Testo	20	italiano,storia,ecc.
tbClassi	Id_classe	Primaria	Numerico	4	
	Classe		Testo	5	sigla della classe
	Aula		Testo	5	
tbComuni	Id_Comune	Primaria	Numerico	4	
	Des		Testo	30	Descriz. Comune
	Pro		Testo	2	Provincia
	Cod		Testo	4	Per Calcolo C.F.
	CAP		Testo	5	
tbDocenti	Id_docente	Primaria	Numerico	4	
	Docente		Testo	30	cognome e nome
	Indirizzo		Testo	30	
	Residenza		Testo	30	
	Telefono		Testo	15	
	Cod_Comune	Esterna	Numerico	4	Collegata alla TbComuni
tbInsegna	Id_insegna	Primaria	Numerico	4	
	Cod_classe	Esterna	Numerico	4	alla tbClassi
	Cod_docente	Esterna	Numerico	4	alla tbDocenti
	Cod_materia	Esterna	Numerico	4	alla tbMaterie
	Numero_ore		Numerico	2	
tbStudenti	Id_studente	Primaria	Numerico	4	
	Studente		Testo	30	cognome e nome
	Cod_classe	Esterna	Numerico	4	Collegata alla tbClassi
	Indirizzo		Testo	30	
	Residenza		Testo	30	
	Telefono		Testo	15	
	Cod_Comune	Esterna	Numerico	4	Collegata alla TbComuni
Codice_Fiscale		Testo	16		
tbProve	Id_prova	Primaria	Numerico	4	
	Cod_studente	Esterna	Numerico	4	alla tbStudenti
	Cod_materia	Esterna	Numerico	4	alla tbMaterie
	Cod_tipoprova	Esterna	Numerico	4	alla tbTipoprove
	Voto		Numerico	2	
	Data_prova		Data/ora	8	

Nota: i campi obbligatori sono i campi che si ritengono indispensabili per la risoluzione del problema.

➤ **Analisi del problema**

La soluzione adottata per la risoluzione del problema prevede l'utilizzo del relativo programma da parte di una qualsiasi scuola di primo o secondo grado in quanto non esiste nessun vincolo specifico. Potendosi adattare a qualsiasi istituto si deve prevedere anche la possibilità di tipi di verifica di tipo grafico sperimentale che possono essere effettuate solo in determinati istituti. Il problema comprende, quindi, la gestione di otto tabelle, individuate dalle relative entità definite nel modello E/R.

Le tabelle *tbTipoProve*, *tbMaterie*, *TbClassi*, *tcComuni* sono le **tabelle di controllo**, dette anche tabelle primarie, che vengono create nella fase iniziale di creazione ed avviamento del database e che, nel tempo, subiranno poche modifiche. Il programma di creazione di tali tabelle deve, comunque, prevedere la possibilità di aggiornamento dei dati.

Per quanto riguarda le tabelle *TbDocenti* e *TbStudenti*, la loro gestione deve essere realizzata mediante un tipico programma di aggiornamento che prevede le funzioni di inserimento, cancellazione e modifica dei record. Tali tabelle contengono fondamentalmente i dati anagrafici sia dei docenti che degli studenti in cui è stato aggiunto un ulteriore attributo (chiave esterna) per associare lo studente alla rispettiva classe di appartenenza. Queste tabelle possono anche essere create all'avvio della procedura ma è evidente che, per la natura dei dati in esse contenuti, potrebbe essere abbastanza frequente il loro aggiornamento (inserimento nuovi docenti, studenti che cambiano scuola o classe, ecc.).

Tutte le tabelle precedentemente trattate costituiscono, comunque, la fonte dei dati per altre due tabelle: la tabella *TbProve*, che serve per gestire tutte le prove sostenute da un qualsiasi studente in una qualsiasi materia, e la tabella *TbInsegna*, che serve a gestire le materie insegnate da un docente in una certa classe.

La gestione delle due tabelle in esame rappresenta il nucleo fondamentale della procedura che risolve il problema proposto. La tabella *TbInsegna*, contenente l'abbinamento del docente alla classe ed alla materia insegnata in quella classe, deve essere creata all'avvio della procedura o all'avvio dell'anno scolastico ma può subire aggiornamenti più o meno frequenti a causa di variazioni di docenti o al passaggio di docenti da una classe ad un'altra. La tabella *TbProve* è sicuramente una tabella che all'inizio dell'anno scolastico non presenta dati e che successivamente sarà sottoposta soprattutto ad operazioni di inserimento (ogni volta che ad uno studente viene assegnato un voto) e le cui modifiche si limitano a correggere eventuali errori nei dati precedentemente inseriti.

La tabella *TbProve* non presenta aspetti particolari in quanto viene usata semplicemente per semplificare l'associazione multi-a-molti che sussiste fra la tabella *TbStudenti* e la tabella *TbMaterie*. Oltre alle chiavi esterne collegate alle chiavi primarie delle due tabelle in esame, la tabella contiene come attributi il voto, la data della prova e la chiave esterna alla tabella *TbTipoProve* in modo da classificare anche il tipo prova a cui è stato sottoposto lo studente. È da notare che con questi dati è sempre possibile conoscere tutti i voti di uno studente in una singola materia, indipendentemente dal docente che ha attribuito il voto. Nel caso in cui, infatti, durante il corso dell'anno scolastico, per un qualsiasi motivo dovesse cambiare il docente di una materia, non si avrebbero variazioni sui precedenti voti attribuiti agli studenti dal docente precedente.

Maggiori considerazioni occorrono, invece, per giustificare la gestione della tabella *TbInsegna*. La prima funzione della suddetta tabella è quella di semplificare l'associazione multi-a-molti fra la tabella *TbClassi* e la tabella *TbDocenti* in quanto, come già specificato nella definizione delle relazioni fra le entità, un docente può insegnare in più classi ed una classe ha più docenti.

Per questo motivo la tabella contiene, come chiavi esterne, le chiavi primarie delle due tabelle interessate all'associazione.

D'altra parte, anche l'associazione fra le tabelle *TbDocenti* e la tabella *TbMaterie* è un'associazione multi-a-molti, in quanto un docente può insegnare più materie (per es. Italiano e Storia) ed una stessa materia, ovviamente, può essere insegnata da più docenti.

Nell'analisi dei dati l'associazione fra le due precedenti tabelle è stata effettuata utilizzando la stessa tabella *TbInsegna* (introducendo come ulteriore chiave esterna la chiave primaria della tabella *TbMaterie*) e non una terza tabella che risolvesse tale associazione. Questa soluzione presenta il vantaggio di ridurre il numero di tabelle da gestire e da collegare senza, comunque, perdere la possibilità di effettuare interrogazioni di vario tipo sulla base di dati a disposizione.

Con una struttura così definita la tabella *TbInsegna* consente di gestire una qualsiasi situazione legata alle possibili combinazioni fra docenti, classi e materie, non ponendo alcun vincolo

particolare, ma offrendo la possibilità di rendere la procedura molto flessibile e quindi adattabile a qualsiasi esigenza.

E' sufficiente notare che, con quest'unica tabella, è possibile gestire, per es., il caso in cui uno stesso docente può insegnare più materie nella stessa classe o in più classi, ma potrebbe insegnare un'ulteriore materia in un'altra classe (un docente può insegnare Italiano e Storia in due classi e Geografia in una terza classe).

In definitiva, la tabella *TbInsegna*, consente di associare al docente di una classe la materia insegnata in quella classe.

L'ultima considerazione riguarda una possibile variazione da apportare alla definizione della struttura delle tabelle, per migliorare le prestazioni della procedura da implementare.

Nella tabella *TbProve* è stata inserita, fra gli attributi, la data in cui lo studente ha sostenuto la prova. In questo modo è solo possibile distinguere i voti per data ma non è possibile distinguerli in base alla suddivisione dell'anno scolastico in trimestri o quadrimestri.

Per risolvere questo problema si potrebbe, per es., aggiungere nella stessa tabella *TbProve* un altro attributo che consente di gestire questo dato, ovvero la fase, rispetto all'anno scolastico, in cui il voto è stato attribuito allo studente.

➤ **Applicativo utilizzato per la soluzione proposta**

L'applicativo usato per sviluppare il DataBase relazionale è Microsoft Access. L'ambiente di sviluppo fornito da "Access" è di tipo visuale e permette attraverso le tabelle la creazione del database e mediante maschere e report la realizzazione di semplici interfacce. E' possibile aggiungere, se necessario, del codice Visual Basic. Il linguaggio SQL sarà invece lo strumento che ci consentirà di rispondere alle query proposte dal problema.

Creazione Data Base

```
CREATE DATABASE DbStudenti
```

Creazione delle Tabelle

```
CREATE TABLE TbTipo_Prova  
(Id_TipoProva COUNTER PRIMARY KEY,  
Tipo_Prova CHAR(20) );
```

```
CREATE TABLE TbMaterie  
(Id_Materia COUNTER PRIMARY KEY,  
Materia CHAR(20) );
```

```
CREATE TABLE TbClassi  
(Id_Classe COUNTER PRIMARY KEY,  
Classe CHAR(10),  
Aula SmallInt );
```

```
CREATE TABLE TbComuni  
(Id_Comune COUNTER PRIMARY KEY,  
Comune CHAR(20),  
Pro CHAR(2),  
Cod CHAR(4),  
CAP Char(5));
```

```
CREATE TABLE ...
```

➤ **QUERY**

Vengono ora mostrate le interrogazioni da effettuare sul database per estrarre le informazioni necessarie a soddisfare ogni richiesta effettuata ed inoltre vengono mostrate le possibili tabelle ottenute.

Per ogni query in cui è prevista una join fra due o più tabelle è presentata la versione con la INNER JOIN (tipica di Access) che sicuramente presenta difficoltà maggiori rispetto alla versione con la clausola WHERE (è consigliabile provarne la conversione).

1) L'elenco completo di tutte le prove sostenute da tutti gli studenti della scuola:

Tabelle da collegare : tbDocenti, tbClassi, tbStudenti, tbMaterie, tbProve, tbTipoProve, tbInsegna

Comando SQL così come lo imposterebbe il DBMS di ACCESS:

```
SELECT [tbDocenti].[Docente], [tbClassi].[Classe], [tbStudenti].[Studente],
       [tbMaterie].[Materia], [tbProve].[Voto], [tbTipoProve].[Tipo_prova]
FROM tbTipoProve INNER JOIN
     (tbMaterie INNER JOIN
      (tbStudenti INNER JOIN
       (tbProve INNER JOIN
        (tbClassi INNER JOIN
         (tbDocenti INNER JOIN
          tbInsegna ON
           [tbDocenti].[Codice]=[tbInsegna].[Cod_docente]) ON
            [tbClassi].[Codice]=[tbInsegna].[Cod_classe]) ON
             [tbProve].[Cod_materia]=[tbInsegna].[Cod_materia]) ON
              [tbStudenti].[Codice]=[tbProve].[Cod_studente]) ON
               [tbMaterie].[Codice]=[tbProve].[Cod_materia]) ON
                [tbTipoProve].[Codice]=[tbProve].[Cod_tipoprova]
        WHERE ([tbInsegna.Cod_classe]=[tbStudenti].[Cod_classe])
        ORDER BY [tbClassi].[Classe], [tbDocenti].[Docente], [tbStudenti].[Studente],
                 [tbMaterie].[Materia];
```

Nota : la condizione espressa nella clausola WHERE è indispensabile in quanto la classe dello studente deve coincidere con la classe abbinata al docente per la materia in esame, altrimenti verrebbero prese in considerazione tutte le classe del docente.

Comando SQL con i connettivi logici:

```
SELECT tbDocenti.Docente,tbClassi.Classe,tbStudenti.Studente,
       tbMaterie.Materia,tbProve.Voto,tbTipoProve.Tipo_Prova
FROM   tbTipoProve,tbMaterie,tbStudenti,tbProve,tbClassi,tbDocenti,tbInsegna
WHERE  tbDocente.Codice=tbInsegna.Cod_Docente
      AND tbClassi.Codice=tbInsegna.Cod_Classe
      AND tbProve.Cod_Materia=tbInsegna.Cod_Materia
      AND tbStudenti.Codice=tbProve.Cod_studente
      AND tbMaterie.Codice=tbProve.Cod_material
      AND tbTipoProve.Codice=tbProve.Cod_tipoprova
      AND tbInsegna.Cod_Classe=tbStudenti.Cod_Classe
ORDER BY tbClassi.Classe,tbDocenti.Docente,tbStudenti.Studente,
         TbMaterie.Materia;
```

Docente	Classe	Studente	Materia	Voto	Tipo_prova
MALVANI COSIMO	3CM	GRIMALDI LUANA	ITALIANO	6	SCRITTO
MALVANI COSIMO	3CM	SCORRANO SILVIA	ITALIANO	8	ORALE
FERRI ENEA	4CM	CARAMIA DONATO	INFORMATICA	7,5	SCRITTO
FERRI ENEA	4CM	SOLARE ANNA	INFORMATICA	9	ORALE
LAMBIASE NICOLA	4CM	GIOVE ANTONIO	EC.AZIENDALE	5	SCRITTO

Docente	Classe	Studente	Materia	Voto	Tipo_prova
LAMBIASE NICOLA	4CM	RUSSO ANDREA	EC.AZIENDALE	8	ORALE
MINEO MARINA	4CM	DELLI NOCI ORESTE	STORIA	5	ORALE
MINEO MARINA	4CM	GUACCI ALESSANDRA	ITALIANO	7	SCRITTO
MINEO MARINA	4CM	GUACCI ALESSANDRA	STORIA	4	ORALE
ARPINO PATRIZIA	5CM	PIPOLA COSIMINA	INGLESE	7	ORALE
ARPINO PATRIZIA	5CM	VENTRUTI BARBARA	INGLESE	6,5	SCRITTO
ARPINO PATRIZIA	5CM	VINCI DONATELLO	INGLESE	6	ORALE
CARELLA M.ANTONIA	5CM	RUSSO ELENIA	EC.AZIENDALE	9	SCRITTO
CARELLA M.ANTONIA	5CM	SBLENDORIO GABRIELE	EC.AZIENDALE	5	ORALE
FERRI ENEA	5CM	AMOROSI PIERLUIGI	INFORMATICA	6	SCRITTO
FERRI ENEA	5CM	CAIAZZO VALERIA	INFORMATICA	6	ORALE
FERRI ENEA	5CM	COSIMO GIUSEPPE	INFORMATICA	6	ORALE
FERRI ENEA	5CM	RUSSO ELENIA	INFORMATICA	5	SCRITTO
SVELTO ANGELO	5CM	CERTA ALESSANDRA	EC.POLITICA	7	ORALE
SVELTO ANGELO	5CM	CERTA ALESSANDRA	DIRITTO	9	ORALE

2) L'elenco delle prove di uno studente:

Tabelle da collegare : tbStudenti, tbClassi, tbMaterie, tbProve, tbTipoProve

```
SELECT [tbStudenti].[Studente], [tbClassi].[Classe], [tbMaterie].[Materia],
       [tbProve].[Voto], [tbTipoProve].[Tipo_prova], [tbProve].[Data_prova]
FROM tbClassi INNER JOIN (tbTipoProve INNER JOIN (tbMaterie INNER JOIN
      (tbStudenti INNER JOIN tbProve ON
        [tbStudenti].[Codice]=[tbProve].[Cod_studente]) ON
        [tbMaterie].[Codice]=[tbProve].[Cod_materia]) ON
        [tbTipoProve].[Codice]=[tbProve].[Cod_tipoprova]) ON
        [tbClassi].[Codice]=[tbStudenti].[Cod_classe]
WHERE ([tbStudenti].[Studente]=[Studente?])
ORDER BY [tbProve].[Data_prova];
```

Studente	Classe	Materia	Voto	Tipo_prova	Data_prova
RUSSO ELENIA	5CM	INFORMATICA	7,5	SCRITTO	01/10/2004
RUSSO ELENIA	5CM	ITALIANO	6	ORALE	15/10/2004
RUSSO ELENIA	5CM	STORIA	8	ORALE	20/10/2004
RUSSO ELENIA	5CM	DIRITTO	6	ORALE	30/10/2004
RUSSO ELENIA	5CM	EC.AZIENDALE	7	SCRITTO	16/11/2004

Nota : Studente? è un parametro di input.

3) L'elenco delle prove effettuate da un docente in una classe:

Tabelle da collegare : tbDocenti, tbClassi, tbStudenti, tbMaterie, tbProve, tbInsegna

```
SELECT [tbDocenti].[Docente], [tbClassi].[Classe], [tbStudenti].[Studente],
       [tbMaterie].[Materia], [tbProve].[Voto]
FROM tbMaterie INNER JOIN (tbStudenti INNER JOIN (tbProve INNER JOIN
      (tbClassi INNER JOIN (tbDocenti INNER JOIN tbInsegna ON
        [tbDocenti].[Codice]=[tbInsegna].[Cod_docente]) ON
        [tbClassi].[Codice]=[tbInsegna].[Cod_classe]) ON
        [tbProve].[Cod_materia]=[tbInsegna].[Cod_materia]) ON
        [tbStudenti].[Codice]=[tbProve].[Cod_studente]) ON
        [tbMaterie].[Codice]=[tbProve].[Cod_materia]
WHERE ([tbDocenti].[Docente]=[Docente?] And
```

```
[tbClassi].[Classe]=[Classe?] And
[tbInsegna.Cod_classe]=[tbStudenti].[Cod_classe])
ORDER BY [tbMaterie].[Materia], [tbStudenti].[Studente];
```

Docente	Classe	Studente	Materia	Voto
MINEO MARINA	5CM	AMOROSI PIERLUIGI	ITALIANO	8
MINEO MARINA	5CM	BASILE ROBERTA	ITALIANO	6
MINEO MARINA	5CM	CERTA ALESSANDRA	ITALIANO	5
MINEO MARINA	5CM	COSIMO GIUSEPPE	STORIA	4
MINEO MARINA	5CM	GUANADIO DESIRE'	STORIA	6
MINEO MARINA	5CM	RUSSO ELENIA	STORIA	6
MINEO MARINA	5CM	SBLENDORIO GABRIELE	STORIA	4
MINEO MARINA	5CM	VINCI DONATELLO	STORIA	5

Nota : Docente? e Classe? sono parametri di input.

4) I voti di uno studente in una materia:

Tabelle da collegare : tbStudenti, tbClassi, tbMaterie, tbProve

```
SELECT [tbStudenti].[Studente], [tbClassi].[Classe], [tbMaterie].[Materia],
[tbProve].[Voto], [tbProve].[Data_prova]
FROM tbClassi INNER JOIN (tbTipoProve INNER JOIN (tbMaterie INNER JOIN
(tbStudenti INNER JOIN tbProve ON
[tbStudenti].[Codice]=[tbProve].[Cod_studente]) ON
[tbMaterie].[Codice]=[tbProve].[Cod_materia]) ON
[tbTipoProve].[Codice]=[tbProve].[Cod_tipoprova]) ON
[tbClassi].[Codice]=[tbStudenti].[Cod_classe]
WHERE ([tbStudenti].[Studente])=[Studente?] And
[tbMaterie].[Materia]=[Materia?])
ORDER BY [tbProve].[Data_prova];
```

Studente	Classe	Materia	Voto	Data_prova
RIZZO TANIA	5CM	STORIA	4	20/10/2003
RIZZO TANIA	5CM	STORIA	3	30/10/2003
RIZZO TANIA	5CM	STORIA	5	16/11/2003

Nota : Studente? e Materia? sono parametri di input.

5) I voti di una classe nella prova svolta in una materia in un determinato giorno:

Tabelle da collegare : tbStudenti, tbClassi, tbMaterie, tbProve, tbTipoProve

Comando SQL in ACCESS:

```
SELECT [tbStudenti].[Studente], [tbClassi].[Classe], [tbMaterie].[Materia],
[tbProve].[Voto], [tbTipoProve].[Tipo_prova], [tbProve].[Data_prova]
FROM tbTipoProve INNER JOIN (tbClassi INNER JOIN (tbStudenti INNER JOIN
(tbMaterie INNER JOIN tbProve ON
[tbMaterie].[Codice]=[tbProve].[Cod_materia]) ON
[tbStudenti].[Codice]=[tbProve].[Cod_studente]) ON
[tbClassi].[Codice]=[tbStudenti].[Cod_classe]) ON
[tbTipoProve].[Codice]=[tbProve].[Cod_tipoprova]
WHERE ([tbMaterie].[Materia]=[Materia?] And
```

```
[tbProve].[Data_prova]=[Dataprova?] And
[tbClassi].[Classe]=[Classe?])
ORDER BY [tbStudenti].[Studente];
```

Comando SQL equivalente:

```
SELECT tbStudenti.Studente, tbClassi.Classe, tbMaterie.Materia,
       tbProve.Voto, tbTipoProve.Tipo_prova, tbProve.Data_prova
FROM tbTipoProve , tbClassi, tbStudenti, tbMaterie, tbProve
WHERE tbMaterie.Materia=[Materia?]
AND   tbProve.Data_prova=[Dataprova?]
AND   tbClassi.Classe=[Classe?]
AND   tbMaterie.Codice=tbProve.Cod_material
AND   tbStudenti.Codice=tbProve.Cod_studente
AND   tbClassi.Codice=tbStudenti.Cod_classe
AND   tbTipoProve.Codice=tbProve.Cod_tipoprova
ORDER BY [tbStudenti].[Studente];
```

Studente	Classe	Materia	Voto	Tipo_prova	Data_prova
CAIAZZO VALERIA	5CM	STORIA	4	SCRITTO	15/11/2003
CERTA ALESANDRA	5CM	STORIA	6	SCRITTO	15/11/2003
COSIMO GIUSEPPE	5CM	STORIA	5	SCRITTO	15/11/2003

Nota : Materia?, Dataprova? e Classe? sono parametri di input.

6) Le date ed il tipo delle prove effettuate da un docente in una materia:

Tabelle da collegare : tbDocenti, tbClassi, tbMaterie, tbProve, tbTipoProve, tbInsegna

```
SELECT [tbDocenti].[Docente], [tbClassi].[Classe], [tbTipoProve].[Tipo_prova],
       [tbMaterie].[Materia], [tbProve].[Data_prova]
FROM tbTipoProve INNER JOIN (tbMaterie INNER JOIN (tbStudenti INNER JOIN
      (tbProve INNER JOIN (tbClassi INNER JOIN (tbDocenti INNER JOIN
      tbInsegna ON
      [tbDocenti].[Codice]=[tbInsegna].[Cod_docente]) ON
      [tbClassi].[Codice]=[tbInsegna].[Cod_classe]) ON
      [tbProve].[Cod_material]=[tbInsegna].[Cod_material]) ON
      [tbStudenti].[Codice]=[tbProve].[Cod_studente]) ON
      [tbMaterie].[Codice]=[tbProve].[Cod_material]) ON
      [tbTipoProve].[Codice]=[tbProve].[Cod_tipoprova]
WHERE ([tbDocenti].[Docente]=[Docente?]) And
      [tbMaterie].[Materia]=[Materia?]) And
      [tbInsegna.Cod_classe]=[tbStudenti].[Cod_classe])
ORDER BY [tbClassi].[Classe], [tbProve].[Data_prova];
```

Docente	Classe	Tipo_prova	Materia	Data_prova
FERRI ENEA	4CM	SCRITTO	INFORMATICA	15/11/2003
FERRI ENEA	4CM	SCRITTO	INFORMATICA	15/11/2003
FERRI ENEA	4CM	ORALE	INFORMATICA	20/10/2003
FERRI ENEA	5CM	ORALE	INFORMATICA	30/10/2003
FERRI ENEA	5CM	SCRITTO	INFORMATICA	15/11/2003
FERRI ENEA	5CM	SCRITTO	INFORMATICA	15/11/2003
FERRI ENEA	5CM	SCRITTO	INFORMATICA	15/11/2003

Nota : Docente? e Materia? sono parametri di input.

7) Le date ed il tipo di prove effettuate da un docente in una classe:

Tabelle da collegare : tbDocenti, tbClassi, tbStudenti, tbMaterie, tbProve, tbTipoProve, tbInsegna

Comando SQL in Access:

```
SELECT [tbDocenti].[Docente], [tbClassi].[Classe], [tbTipoProve].[Tipo_prova],
       [tbMaterie].[Materia], [tbProve].[Data_prova]
FROM tbTipoProve INNER JOIN
      (tbMaterie INNER JOIN
        (tbStudenti INNER JOIN
          (tbProve INNER JOIN
            (tbClassi INNER JOIN
              (tbDocenti INNER JOIN
                tbInsegna ON
                [tbDocenti].[Codice]=[tbInsegna].[Cod_docente])
              ON
              [tbClassi].[Codice]=[tbInsegna].[Cod_classe])
            ON
            [tbProve].[Cod_materia]=[tbInsegna].[Cod_materia])
          ON
          [tbStudenti].[Codice]=[tbProve].[Cod_studente])
        ON
        [tbMaterie].[Codice]=[tbProve].[Cod_materia])
      ON
      [tbTipoProve].[Codice]=[tbProve].[Cod_tipoprova]
WHERE ([tbDocenti].[Docente]=[Docente?] And [tbClassi].[Classe]=[Classe?]
       And ([tbInsegna.Cod_classe]=[tbStudenti].[Cod_classe]))
ORDER BY [tbClassi].[Classe], [tbProve].[Data_prova], [tbMaterie].[Materia];
```

ComandoSQL:

```
SELECT tbDocenti.Docente, tbClassi.Classe, tbTipoProve.Tipo_prova,
       tbMaterie.Materia, tbProve.Data_prova
FROM   tbTipoProve, tbMaterie , tbStudenti , tbProve, tbClassi, tbDocenti, tbInsegna
WHERE  tbDocenti.Codice=tbInsegna.Cod_docente
AND    tbClassi.Codice=tbInsegna.Cod_classe
AND    tbProve.Cod_materia=tbInsegna.Cod_materia
AND    tbStudenti.Codice=tbProve.Cod_studente
AND    tbMaterie.Codice=tbProve.Cod_materia
AND    tbTipoProve.Codice=tbProve.Cod_tipoprova
AND    tbDocenti.Docente=[Docente?] And tbClassi.Classe=[Classe?]
AND    tbInsegna.Cod_classe=tbStudenti.Cod_classe
ORDER BY tbClassi.Classe, tbProve.Data_prova, tbMaterie.Materia;
```

Docente	Classe	Tipo_prova	Materia	Data_prova
MINEO MARINA	5CM	ORALE	ITALIANO	15/10/2003
MINEO MARINA	5CM	ORALE	ITALIANO	15/10/2003
MINEO MARINA	5CM	ORALE	STORIA	20/10/2003
MINEO MARINA	5CM	SCRITTO	STORIA	30/10/2003
MINEO MARINA	5CM	SCRITTO	STORIA	15/11/2003
MINEO MARINA	5CM	SCRITTO	STORIA	15/11/2003
MINEO MARINA	5CM	SCRITTO	STORIA	15/11/2003
MINEO MARINA	5CM	ORALE	ITALIANO	17/11/2003

Nota : Docente? e Classe? sono parametri di input

8) La media dei voti di ogni alunno in ciascuna materia;

Tabelle da collegare : tbStudenti, tbClassi, tbMaterie, tbProve,

```
SELECT [tbClassi].[Classe], [tbStudenti].[Studente], [tbMaterie].[Materia],
      Avg([tbProve].[Voto]) AS Media_Voti
FROM tbClassi INNER JOIN (tbMaterie INNER JOIN (tbStudenti INNER JOIN
      tbProve ON [tbStudenti].[Codice]=[tbProve].[Cod_studente]) ON
      [tbMaterie].[Codice]=[tbProve].[Cod_materia]) ON
      [tbClassi].[Codice]=[tbStudenti].[Cod_classe]
GROUP BY [tbStudenti].[Studente], [tbMaterie].[Materia],
         [tbProve].[Cod_studente], [tbProve].[Cod_materia], [tbClassi].[Classe]
ORDER BY [tbClassi].[Classe], [tbStudenti].[Studente];
```

Classe	Studente	Materia	Media_Voti
3CM	GRIMALDI GIOVANNA	ITALIANO	7
4CM	MAGGIO DANILA	ITALIANO	6
4CM	MAGGIO DANILA	STORIA	5
4CM	SOLARE ANNA	ITALIANO	7
4CM	SOLARE ANNA	STORIA	7,5
5CM	AMOROSI PIERLUIGI	ITALIANO	6
5CM	AMOROSI PIERLUIGI	DIRITTO	8
5CM	AMOROSI PIERLUIGI	STORIA	4
5CM	CAROLI FABIO	STORIA	4
5CM	CAROLI FABIO	INFORMATICA	7
5CM	CAROLI FABIO	ITALIANO	5
5CM	ROSA DAVIDE	INFORMATICA	3
5CM	ROSA DAVIDE	STORIA	6
5CM	ROSA DAVIDE	DIRITTO	6,5
5CM	ROSA DAVIDE	ED.FISICA	6

9) L'elenco del numero degli alunni per ogni classe;

Tabelle da collegare : tbStudenti, tbClassi

Comando SQL in Access:

```
SELECT [tbClassi].[Classe], COUNT(*) AS Numero_alunni
FROM tbClassi INNER JOIN tbStudenti ON
      [tbClassi].[Codice]=[tbStudenti].[Cod_classe]
GROUP BY [tbClassi].[Classe];
```

Comando SQL:

```
SELECT tbClassi.Classe, COUNT(*) AS Numero_alunni
FROM tbClassi, tbStudenti
WHERE tbClassi.Codice=tbStudenti.Cod_classe
GROUP BY tbClassi.Classe;
```

Classe	Numero_alunni
3CM	24
4CM	22
5CM	22

10) L'elenco, per una data classe, del numero di prove diverse sostenute da ciascun studente;

Tabelle da collegare : tbStudenti, tbClassi, tbMaterie, tbProve, tbTipoProve

Comando SQL in Access:

```
SELECT [tbClassi].[Classe], [tbStudenti].[Studente], [tbMaterie].[Materia],
      [tbTipoProve].[Tipo_prova], Count(*) AS Numero_prove
FROM tbTipoProve INNER JOIN (tbClassi INNER JOIN (tbMaterie INNER JOIN
      (tbStudenti INNER JOIN tbProve ON
      [tbStudenti].[Codice]=[tbProve].[Cod_studente]) ON
      [tbMaterie].[Codice]=[tbProve].[Cod_materia]) ON
      [tbClassi].[Codice]=[tbStudenti].[Cod_classe]) ON
      [tbTipoProve].[Codice]=[tbProve].[Cod_tipoprova]
WHERE [tbClassi].[Classe]=[Classe?]
GROUP BY [tbClassi].[Classe], [tbStudenti].[Studente], [tbMaterie].[Materia],
      [tbTipoProve].[Tipo_prova]
ORDER BY [tbStudenti].[Studente];
```

Comando SQL:

```
SELECT tbClassi.Classe, tbStudenti.Studente, tbMaterie.Materia,
      tbTipoProve.Tipo_prova, Count(*) AS Numero_prove
FROM tbTipoProve, tbClassi, tbMaterie, tbStudenti, tbProve
WHERE tbClassi.Classe=[Classe?]
AND   tbStudenti.Codice=tbProve.Cod_studente
AND   tbMaterie.Codice=tbProve.Cod_material
AND   tbClassi.Codice=tbStudenti.Cod_classe
AND   tbTipoProve.Codice=tbProve.Cod_tipoprova
GROUP BY tbClassi.Classe, tbStudenti.Studente, tbMaterie.Materia,
      tbTipoProve.Tipo_prova
ORDER BY tbStudenti.Studente;
```

Classe	Studente	Materia	Tipo_prova	Numero_prove
5CM	AMOROSI PIERLUIGI	INFORMATICA	SCRITTO	1
5CM	CAIAZZO VALERIA	ITALIANO	ORALE	1
5CM	CERTA ALESSANDRA	STORIA	SCRITTO	1
5CM	CERTA ALESSANDRA	INFORMATICA	ORALE	1
5CM	CERTA ALESSANDRA	INFORMATICA	PRATICO	1
5CM	COSIMO GIUSEPPE	STORIA	ORALE	1
5CM	RUSSO ELENIA	INFORMATICA	SCRITTO	1
5CM	RUSSO ELENIA	ITALIANO	ORALE	1
5CM	RUSSO ELENIA	STORIA	ORALE	1
5CM	RUSSO ELENIA	DIRITTO	ORALE	2
5CM	SBLENDORIO GABRIELE	INFORMATICA	ORALE	1
5CM	VINCI DONATELLO	ITALIANO	ORALE	1

Nota : Classe? è un parametro di input