

DATABASE

NORMALIZZAZIONE

TRATTO DA CAMAGNI-NIKOLASSY, CORSO DI INFORMATICA, VOL 2, HOEPLI

Informatica

La Normalizzazione

IN QUESTA UNITÀ IMPAREREMO...

- il concetto di dipendenza funzionale
- le motivazioni alla base della normalizzazione
- ad applicare le regole di normalizzazione

La Normalizzazione

- L'obiettivo della normalizzazione è quello di scomporre le tabelle in tabelle più piccole in modo da avere relazioni più strutturate e ridurre al minimo la ridondanza dei dati

NORMALIZZAZIONE

La **normalizzazione** è un processo che tende a eliminare la ripetizione dei dati (ridondanza) e a migliorarne la consistenza.

DB non normalizzato

- Un database non normalizzato presenta problemi di :
 - ▣ Ridondanza
 - ▣ Inefficienza
 - ▣ Complessità
 - ▣ Perdita di informazioni

RIDONDANZA

Si ha **ridondanza** dei dati ogni volta che vengono memorizzati inutilmente dei dati ripetuti.

Esempio

- Consideriamo la tabella che contiene i dati dei dipendenti di un'istituzione scolastica

Cognome	Nome	Funzione	Stipendio	Scuola
Rossi	Mario	Preside	50 €	ITIS G. Verdi
Verdi	Filippo	Docente	35 €	ITIS G. Verdi
Verdi	Filippo	Docente	35 €	ITIS P. Rossi
Gialli	Enrico	Docente	35 €	ITIS G. Verdi
Paperino	Paolino	Docente	30 €	ITIS G. Verdi
Paperino	Paolino	Docente	30 €	ITIS S. Gialli
Ali	Baba	Bidello	16 €	ITIS G. Verdi
Verdi	Giuseppe	Bidello	16 €	ITIS G. Verdi

In tabella abbiamo:

- **ridondanza**: lo stipendio di ciascun dipendente è **ripetuto** in tutte le tuple (record) relative;
- **anomalia di modifica**: se lo stipendio di un dipendente varia è necessario andare a modificarne il valore in tutte le tuple nelle quali è presente;
- **anomalia di cancellazione**: se un dipendente resta senza scuola, dobbiamo cancellarlo;
- **anomalia di inserimento**: un nuovo dipendente che non è assegnato ad alcuna scuola non può essere inserito.

Forma normale

- Le ultime tre anomalie sono dette di **aggiornamento**

Queste anomalie sono presenti perché abbiamo utilizzato un'unica relazione per rappresentare informazioni eterogenee mentre sarebbe stato opportuno suddividere la tabella in più relazioni. Le regole di **normalizzazione** ci permettono di passare "automaticamente" da una tabella di partenza generica a un insieme di tabelle esenti dalle anomalie sopra elencate.

- La normalizzazione si basa sul concetto di forma normale
 - ▣ Una relazione (tabella) è in una particolare forma normale se soddisfa un certo insieme di vincoli
- Le forme normali sono cinque, noi ne studieremo tre.

Dipendenze funzionali

- Il concetto di **dipendenza funzionale** è alla base delle prime tre forme normali
- Consideriamo il seguente esempio

ID_Anag	Cognome	Nome	Indirizzo	Località	CAP	Telefono
1	Rossi	Mario	Via Lombardia, 1	Como	22100	031.123123
2	Verdi	Filippo	Via Marche, 34	Milano	20100	02.321321
3	Gialli	Enrico	Viale Abruzzi, 4	Torino	10100	011.132132
4	Paperino	Paolino	Via Lazio, 34	Como	22100	031.121212
5	Ali	Baba	Via Puglia, 34	Milano	20100	02.332233
6	Verdi	Luca	Via Emilia, 32	Como	22100	031.112233

Dipendenza funzionale

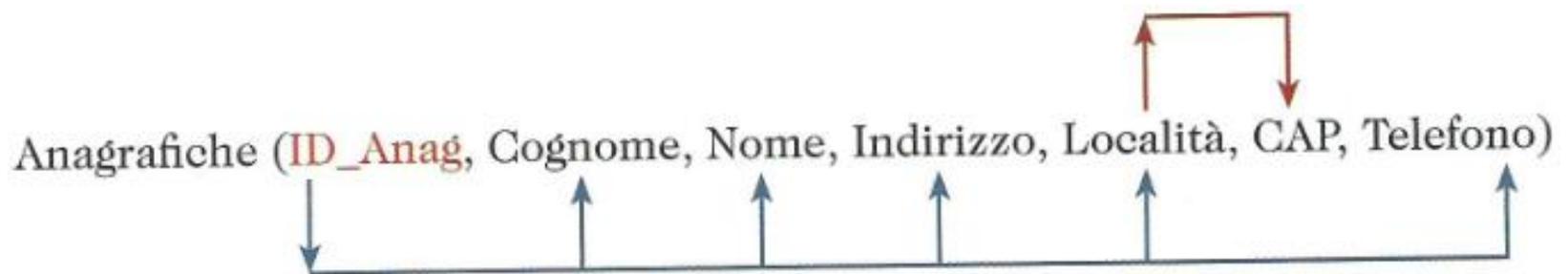
- I valori contenuti nelle colonne *Cognome*, *Nome*, *Indirizzo* e *Telefono* sono associati ad un solo valore della colonna *ID_Anag*
- Si dice che *Cognome*, *Nome*, *Indirizzo* e *Telefono* sono **funzionalmente dipendenti** dalla colonna *ID_Anag* che prende il nome di **determinante** per tali colonne

ID_Anag \longrightarrow *Nome, Cognome, Indirizzo, Telefono*

- Lo stesso non vale per le colonne *Località* e *CAP*, in quanto a valori diversi della colonna *ID_Anag* corrispondono valori ripetuti nelle colonne *Località* e *CAP*

Dipendenza funzionale

- In definitiva nel caso della tabella, le dipendenze funzionali sono:



DIPENDENZA FUNZIONALE

Una colonna Y di una tabella R viene detta **funzionalmente dipendente** dalla colonna X di R se ogni valore di X in R viene associato con un solo valore di Y. La colonna X è detta **determinante** per la colonna Y.

Dipendenza funzionale

- Quindi dire che la colonna Y è funzionalmente dipendente da X è come dire che i valori della colonna X identificano i valori della colonna Y
- Se la colonna X è una chiave primaria, allora tutte le colonne nella tabella R devono essere funzionalmente dipendenti da X.

DIPENDENZA FUNZIONALE: NOTAZIONE

Una notazione breve per descrivere una dipendenza funzionale è: $R.x \rightarrow R.y$ che può essere letta come "nella tabella relazionale R la colonna x determina funzionalmente i valori della colonna y".

Normalizzazione

- La normalizzazione è un processo che serve a rimuovere i dati ridondanti dalle tabelle decomponendole in tabelle più piccole
- Lo scopo è quello di avere solo chiavi primarie a sinistra delle colonne dipendenti funzionalmente

Prima forma normale (1FN)

RELAZIONE IN PRIMA FORMA NORMALE

Una relazione si dice in prima forma normale (1NF) se e solo se tutti i suoi attributi sono valori atomici: ciò implica che né gli attributi né i valori da questi assunti possono essere scomposti ulteriormente.

- Esempio: la seguente tabella non è in 1FN

Codice Fiscale	Generalità
LBRRSS79Y12T344A	Rossi Alberto

- Generalità è un campo composto, bisogna scomporre tale attributo in due campi:

Codice Fiscale	Cognome	Nome
LBRRSS97Y12T344A	Rossi	Alberto

Prima forma normale (1FN)

□ Ulteriore esempio

Codice Fiscale	Cognome	Nome	Residenza
LBRRSS97Y12T344A	Rossi	Alberto	Via Roma, 22; 22100 Como
GNNBNCT99A11L611B	Bianchi	Gianni	Piazza Lodi, 12; 80010 Napoli



Codice Fiscale	Cognome	Nome	Indirizzo	CAP	Città
LBRRSS97Y12T344A	Rossi	Alberto	Via Roma, 22	22100	Como
GNNBNCT99A11L611B	Bianchi	Gianni	Piazza Lodi, 12	80010	Napoli

Scomposizione Indirizzo

Anche la tabella ottenuta non è ancora in prima forma normale: infatti l'attributo Indirizzo è composto a sua volta da tre possibili attributi: TipoVia, NomeVia, Civico. Quindi sarebbe da scomporre in questo modo:

Codice Fiscale	Cognome	Nome	TipoVia	NomeVia	Civico	CAP	Città
LBRRSS97Y12T344A	Rossi	Alberto	Via	Roma	22	22100	Como
GNNBNCT99A11L611B	Bianchi	Gianni	Piazza	Lodi	12	80010	Napoli

Anche se formalmente corretta, nella pratica la scomposizione dell'indirizzo nelle sue tre componenti **non viene mai effettuata**, tranne in applicazioni particolari nelle quali è importante isolare i singoli elementi.

Seconda forma normale (2FN)

Partiamo dalla situazione seguente:

una società commerciale acquista articoli da un certo numero di fornitori distribuiti sul territorio europeo. Ogni fornitore è posizionato in una città. Una città può avere più di un fornitore e a ogni città è associato il codice dello stato di appartenenza. Ogni fornitore può fornire più articoli. La società crea una semplice tabella per salvare queste informazioni che possono essere espresse in notazione relazionale come

Tabella1 (ID_Fornitore, Stato, Città, ID_Articolo, Qta)

ID_Fornitore	è il numero identificativo del fornitore (la sua chiave primaria);
Stato	è il codice di stato a cui appartiene la Città;
Città	è il nome della città dove il fornitore risiede;
ID_Articolo	è il numero di articoli forniti;
Qta	è il numero di articoli forniti.

- Per identificare univocamente la quantità di un certo prodotto fornita da un dato fornitore si usa la chiave primaria composta ID_Fornitore+ID_Articolo

Seconda forma normale (2FN)

Tabella1

ID_Fornitore	Stato	Città	ID_Articolo	Qtà
S1	20	Roma	P1	300
S1	20	Roma	P2	400
S1	20	Roma	P3	200
S1	20	Roma	P4	150
S1	20	Roma	P5	350
S1	20	Roma	P6	500
S2	10	Londra	P1	400
S2	10	Londra	P2	300
S3	30	Parigi	P2	200
S4	40	Budapest	P2	100
S4	40	Budapest	P4	150
S4	40	Budapest	P5	250

- La Tabella1 è in 1FN, però contiene dati ridondanti
 - Città e Stato sono ripetute per ogni articolo

Seconda forma normale (2FN)

- La ridondanza provoca anomalie di aggiornamento

Inserimento	L'informazione che un nuovo fornitore (S5) sia residente in una particolare città (Atene) non può essere aggiunta finché non fornisce un articolo.
Cancellazione	Se una riga viene cancellata, allora non solo le informazioni sulla quantità e sull'articolo vengono perdute, ma anche quelle sul fornitore.
Modifica	Se il fornitore S1 si trasferisce da Londra a New York, allora sei righe devono essere modificate con questa nuova informazione.

Seconda forma normale (2FN)



TABELLA IN SECONDA FORMA NORMALE

Una tabella in **seconda forma normale** è una tabella in 1NF in cui tutte le colonne non chiave sono completamente dipendenti dalla intera chiave primaria. Ovvero non sono ammesse colonne che dipendono funzionalmente solo da una parte della **chiave primaria composta**.

- La Tabella1 è in 1FN ma non in 2FN perché le colonne *Stato* e *Città* sono **funzionalmente dipendenti** solo dalla colonna *ID_Fornitore* della chiave primaria composta (*ID_Fornitore*+*ID_Articolo*)
- Le dipendenze funzionali nella Tabella1 sono:

$ID_Fornitore \rightarrow Città, Stato$

$Città \rightarrow Stato$

$(ID_Fornitore, ID_Articolo) \rightarrow Qta$

Seconda forma normale (2FN)

- Una Tabella può essere trasformata da 1FN a 2FN seguendo i seguenti passi:
 - 1 Identificare ogni determinante diverso dalla chiave composta e le colonne che determina.
 - 2 Creare e nominare una nuova tabella per ogni determinante e le colonne chiave che determina.
 - 3 Spostare le colonne così individuate dalla tabella originale alla nuova tabella. Il determinante diventa la nuova chiave primaria della nuova tabella.
 - 4 Cancellare le colonne che sono state spostate dalla tabella originale eccetto i determinanti che servono come chiave esterna.
 - 5 La tabella originale può essere rinominata per mantenere un significato semantico.

Esempio 2FN

- La tabella1 trasformata in 2FN diventa:

Fornitori		
ID_Fornitore(pk)	Stato	Città
S1	20	Roma
S2	10	Londra
S3	30	Parigi
S4	40	Budapest

La colonna ID_Fornitore diventa la chiave primaria della nuova tabella, mentre nella Tabella1 ID_Fornitore resta un componente della chiave primaria e chiave esterna per l'associazione con la Tabella Fornitori.

Tabella 1		
ID_Articolo(pk)	ID_Fornitore(pk)(fk)	Qtà
P1	S1	300
P2	S1	400
P3	S1	200
P4	S1	150
P5	S1	350
P6	S1	500
P1	S2	400
P2	S2	300
P2	S3	200
P2	S4	100
P4	S4	150
P5	S4	250

Esempio2 – 2FN

- Database scuola con chiave primaria composta da *Matricola+CodiceMateria*

Matricola	Codice Materia	Alunno	Voto Materia
1234	Info1	Rossi	6
1234	Elettro2	Rossi	7
1235	Elettro2	Verdi	8

Il database qui sopra si trova in 1NF ma non in 2NF in quanto il campo “Alunno” non dipende dall’intera chiave ma solo da una parte di essa (“Codice Matricola”). Per rendere il nostro database 2NF dovremo scomporlo in due tabelle:

Matricola	Codice Materia	Voto Materia
1234	Info1	6
1234	Elettro2	7
1235	Elettro2	8

Matricola	Alunno
1234	Rossi
1235	Verdi

Esempio3 – 2FN

Abbiamo una relazione che rappresenta delle scrivanie da ufficio prodotte da un'azienda industriale, la chiave primaria è data da TipoTavolo e TipoLegno.

ModelloTavolo	TipoLegno	TipoFinitura
Marina	Mogano	Lisciatura
Marina	Ciliegio	Levigatura
President	Noce	Lucidatura
President	Mogano	Lisciatura
Top	Noce	Lucidatura

TipoFinitura è dipendente funzionalmente da un sottoinsieme della chiave primaria, ovvero da TipoLegno, dunque la tabella non è in 2NF.

È possibile normalizzare la relazione trasformandola in due tabelle:

ModelloTavolo	TipoLegno
Marina	Mogano
Marina	Ciliegio
President	Noce
President	Mogano
Top	Noce

TipoLegno	TipoFinitura
Mogano	Lisciatura
Ciliegio	Levigatura
Noce	Lucidatura

Osservazione su 2FN

- OSS.: Presupposto per verificare se il DB è in 2FN è che sia presente una chiave primaria composta

Naturalmente in un tabella dove la **chiave primaria** è **semplice** non ci sono problemi inerenti alla seconda forma normale.

Terza forma normale (3FN)

TERZA FORMA NORMALE

La **terza forma normale** richiede che la tabella sia già in seconda forma normale e che tutte le colonne in una tabella relazionale siano dipendenti solamente dalla chiave primaria.

Terza forma normale (3FN)

- Riprendiamo l'esempio precedente (Fornitori)

Fornitori		
ID_Fornitore(pk)	Stato	Città
S1	20	Roma
S2	10	Londra
S3	30	Parigi
S4	40	Budapest

Tabella 1		
ID_Articolo(pk)	ID_Fornitore(pk)(fk)	Qtà
P1	S1	300
P2	S1	400
P3	S1	200
P4	S1	150
P5	S1	350
P6	S1	500
P1	S2	400
P2	S2	300
P2	S3	200
P2	S4	100
P4	S4	150
P5	S4	250

- Le tabelle in 2FN contengono ancora anomalie di aggiornamento

insert	L'inserimento di una nuova Città non può essere effettuato finché non c'è un fornitore in quella Città.
delete	Cancellare qualsiasi riga dalla tabella Fornitori cancella l'informazione sulla città oltre all'associazione tra Città e Fornitore.

Terza forma normale (3FN)

- La *Tabella1* è già in 3FN, perché la colonna non chiave *Qtà* è completamente dipendente dalla chiave primaria (*ID_Articolo+ID_Fornitore*)
- La tabella *Fornitori* è in 2FN, ma non in 3FN perché la colonna *Stato* non dipende dalla chiave *ID_Fornitore*, ma dalla colonna *Città*
- È presente una **dipendenza transitiva**

Dipendenza transitiva

DIPENDENZA TRANSITIVA

Una **dipendenza transitiva** si ha quando una colonna è un determinante di un'altra colonna che a sua volta è un determinante per altre colonne.

In simboli:

se $R.X \rightarrow R.Y \rightarrow R.Z$, $R.Z$ è in dipendenza transitiva rispetto a $R.X$.

- Nel caso dell'esempio precedente si ha:

Fornitori.ID_Fornitore \rightarrow Fornitori.Città

Fornitori.Città \rightarrow Fornitori.Stato

Terza forma normale (3FN)

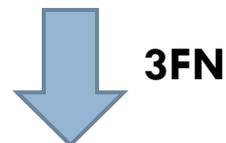
□ La procedura per trasformare una tabella in 3FN è:

- 1 Identificare qualsiasi determinante, oltre alla chiave primaria, e le colonne che determina.
- 2 Creare e nominare una nuova tabella per ogni determinante e le colonne univoche che determina.
- 3 Spostare le colonne del determinante dalla tabella originale nella nuova tabella. Il determinante diventa la chiave primaria della nuova tabella.
- 4 Cancellare le colonne che sono state spostate dalla tabella originale eccetto il determinante che serve come chiave esterna.

Esempio 1 – 3FN

- La tabella Fornitori trasformata in 3FN diventa:

Fornitori		
ID_Fornitore(pk)	Stato	Città
S1	20	Roma
S2	10	Londra
S3	30	Parigi
S4	40	Budapest



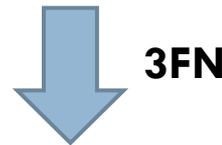
Id_Fornitore()	Città(fk)
S1	Roma
S2	Londra
S3	Parigi
S4	Budapest

Città(pk)	Stato
Roma	20
Londra	10
Parigi	30
Budapest	20

Esempio2 – 3FN

Supponiamo di avere una base dati di una palestra che associ il codice fiscale dell'iscritto al corso frequentato e all'istruttore di riferimento. Si supponga che il nostro DB abbia un'unica chiave primaria ("Codice Fiscale") e sia così strutturato:

Codice Fiscale	Corso	Istruttore
LBRSS93Y12T344A	Body building	Marco
GNNBNCT96A11L611B	Body building	Marco
LBRMNN96E64A112A	Body building	Marco
GLSTMT99U66P109B	Fitness	Federica



Codice Fiscale	Corso
LBRSS93Y12T344A	Body building
GNNBNCT96A11L611B	Body building
LBRMNN96E64A112A	Body building
GLSTMT99U66P109B	Fitness

Corso	Istruttore
Body building	Marco
Fitness	Federica

Vantaggi 3FN

Il vantaggio di avere una tabella in 3NF è che **elimina la ripetizione dei dati** in modo da ridurre l'ingombro sulla memoria di massa per il salvataggio delle tabelle ed eliminare le anomalie da aggiornamento.

In conclusione

RELAZIONE IN PRIMA FORMA NORMALE

Una relazione si dice in prima forma normale (1NF) se e solo se tutti i suoi attributi sono valori atomici: ciò implica che né gli attributi né i valori da questi assunti possono essere scomposti ulteriormente.



TABELLA IN SECONDA FORMA NORMALE

Una tabella in **seconda forma normale** è una tabella in 1NF in cui tutte le colonne non chiave sono completamente dipendenti dalla intera chiave primaria. Ovvero non sono ammesse colonne che dipendono funzionalmente solo da una parte della **chiave primaria composta**.

- **Precondizione: presenza di una chiave primaria composta**

TERZA FORMA NORMALE

La **terza forma normale** richiede che la tabella sia già in seconda forma normale e che tutte le colonne in una tabella relazionale siano dipendenti solamente dalla chiave primaria.

Esempio

- **Problema:** definire la struttura dati di un archivio che deve memorizzare le fatture commerciali
- Ipotizziamo di realizzare un'unica tabella con i seguenti campi:
 - ▣ NrFattura, Data, CodiceCli, RagioneSociale, Indirizzo, ID_Agente, NomeAgente, NrItem, Descrizione, Quantità, PrezzoUnitario
 - ▣ NB: gli ultimi quattro sono ripetuti più volte

Esempio, 1FN

- Separiamo ora dalla tabella principale i gruppi di campi che si ripetono, altrimenti avremo delle celle della tabella con più valori, e quindi non in 1FN
- Creiamo una nuova tabella **RigheFatt**, le cui righe costituiscono le righe della fattura

Fatture(NrFattura(pk), Data(pk), CodiceCli, RagioneSociale, Indirizzo, ID_Agente, NomeAgente, NrItem(pk), Descrizione, Quantità, PrezzoUnitario)



RigheFatt(NumeroFat(pk)(fk), Data(pk)(fk), NrItem(pk), Descrizione, Quantità, PrezzoUnitario)

Fatture(NrFattura(pk), Data(pk), CodiceCli, RagioneSociale, Indirizzo, ID_Agente, NomeAgente)

Esempio, 2FN

- Occorre rimuovere le dipendenze parziali
- Creiamo delle tabelle separate con i dati funzionalmente dipendenti e i loro determinanti

Fatture(NrFattura(pk), Data(pk), CodiceCli, RagioneSociale, Indirizzo, ID_Agente, NomeAgente)

RigheFatt(NumeroFat(pk)(fk), Data(pk)(fk), NrItem(pk), Descrizione, Quantità, PrezzoUnitario)



2FN

RigheFatt(NumeroFat(pk)(fk), Data(pk)(fk), NrItem(pk), Quantità, PrezzoUnitario)

ITEM(NrItem(pk)(fk), Descrizione)

Fatture(NrFattura(pk), Data(pk), CodiceCli, RagioneSociale, Indirizzo, ID_Agente, NomeAgente)

Esempio, 3FN

- Rimuoviamo ora le dipendenze transitive e creiamo una tabella separata per ogni dipendenza funzionale

RigheFatt(NumeroFat(pk) (fk), Data(pk) (fk), NrItem(pk), Quantità, PrezzoUnitario)

ITEM(NrItem(pk) (fk), Descrizione)

Fatture(NrFattura(pk), Data(pk), CodiceCli, RagioneSociale, Indirizzo, ID_Agente, NomeAgente)



RigheFatt(NumeroFat(pk) (fk), Data(pk) (fk), NrItem(pk), Quantità, PrezzoUnitario)

ITEM(NrItem(pk) (fk), Descrizione)

Fatture(NrFattura(pk), Data(pk), CodiceCli, ID_Agente)

Clienti(CodiceCli(pk), RagioneSociale, Indirizzo)

Agenti(ID_Agente(pk), NomeAgente)

Esercizi

Normalizza le seguenti relazioni complesse, indicando per ogni passo i benefici ottenuti e le anomalie ancora presenti.

- 1 Anagrafica (Nome, Cognome, Indirizzo, CAP, Citta, Provincia, Regione, Telefono, Nazione)
- 2 Interrogazioni (Materia, Voto, Data, Nome, Cognome, Classe, Sezione)
- 3 Professori (Cognome, Nome, Indirizzo, Citta, CAP, Classe, Sezione, Materia)
- 4 CDMusicali (Gruppo, Titolo, Canzone1, Canzone2, ... CanzoneN, Genere, Lingua)
- 5 Fatture (Cliente, Indirizzo, NrItem, NumeroFat, Data, Descrizione, Quantità, Prezzo Unitario)
- 6 Animali (CodAnagrafico, Razza, NomeProprio, Sesso, DataNascita, LuogoNascita, Nazione)
- 7 Dipendenti (Matricola, Nominativo, Stipendio, Progetto, CodBilancio, Funzione, Sede)
- 8 Fornitori (Fornitore, Indirizzo, NumeroFat, Data, Descrizione, Quantità, Prezzo Unitario)
- 9 Alunni (Cognome, Nome, Indirizzo, Citta, CAP, Classe, Sezione, Specializzazione)
- 10 Gite (Classe, Sezione, Specializzazione, Periodo, Destinazione, Accompagnatore)
- 11 Automobili (Targa, Modello, CasaAutomob, Anno Immatric, Colore, Cilindrata, Categoria)
- 12 Francobolli (Nazione, Tematica, NumCatalogo, ValoreNuovo, ValoreUsato, Serie, Periodo)
- 13 Contravvenzioni (codMulta, codAgente, costoMulta, targa, codModello, descModello, codMarca, descMarca, nomeAgente, data, ora, descMulta)
- 14 Giocatori (Nazione, Squadra, Ruolo, Cognome, Nome, CittaSquadra, CittaNascita, Numero)
- 15 Libri (Autore1, Autore, ... AutoreN, Editore, Collana, Genere, AnnoPubblicaz, Prezzo, Nazione, Lingua, Titolo, ISBN)