

**I.P.S.I.A. MORETTO BRESCIA**

**Tesi di maturità**

**Anno scolastico 2003/04**

**MAGAZZINO 2004**

**Studenti:**

**Pell simone  
pell stefano**



<b>Introduzione .....</b>	<b>3</b>
<b>Capitolo 1 – Specifiche del progetto.....</b>	<b>4</b>
<b>Capitolo 2 – Tecnologie utilizzate.....</b>	<b>18</b>
<b>I Database _____</b>	<b>18</b>
<b>Il linguaggio SQL _____</b>	<b>19</b>
<b>Microsoft Access _____</b>	<b>21</b>
<b>Borland Delphi 5 _____</b>	<b>22</b>
<b>Microsoft PowerPoint _____</b>	<b>26</b>
<b>Capitolo 3 – Struttura del database e importazione dati esistenti .....</b>	<b>27</b>
<b>Capitolo 4 – Programma Magazzino 2004 .....</b>	<b>32</b>
<b>Conclusione.....</b>	<b>36</b>

## **Introduzione**

La scuola da sempre per quanto riguarda la gestione dell'istituto è divisa in vari settori ognuno con un proprio compito (l'ufficio personale, l'ufficio contabilità, la segreteria didattica, l'ufficio magazzino).

Il nostro obiettivo è quello di realizzare un programma per la gestione dei prodotti del magazzino.

Questo programma era già stato sviluppato da degli esperti programmatori però aveva dei problemi.

Noi avendo una minima conoscenza e con la collaborazione del professor Cleto Azzani abbiamo risviluppato questo programma in Delphi risolvendo tutti i problemi.

L'applicazione permette la gestione del Magazzino (carichi, resi, scarichi e versamenti) e dell'Inventario consentendo l'immediata consultazione delle giacenze.

Permette inoltre la gestione delle stampe.

# Capitolo 1 – Specifiche del progetto

## Specifiche Tecniche del Programma Magazzino 2004

### Archivi Attuali

Il programma attuale è stato sviluppato in Delphi ed è basato su un DataBase nativo Interbase 6.01; la struttura del DataBase è sotto riportata; le tabelle fondamentali sono: ARTICOLI e MAGAZZINO:

```
CREATE TABLE ARTICOLI
(
  CODICE CHAR(8) NOT NULL,
  DESCRIZIONE CHAR(50),
  DESCRIZIONE2 CHAR(50),
  DESCRIZIONE3 CHAR(50),
  TIPOLOGIA CHAR(8),
  COD_FORNITORE CHAR(8),
  BARCODE CHAR(30),
  COSTO NUMERIC(15, 5),
  U_MISURA CHAR(4),
  POSIZIONE CHAR(10),
  DATA_ACQ TIMESTAMP NOT NULL,
  QUANTITA_CORR NUMERIC(15,2),
  TOT_CARICATO NUMERIC(15,2),
  TOT_SCARICATO NUMERIC(15,2),
  SCORTA_MIN NUMERIC(15,2),
  VALORI NUMERIC(15, 5),
  VAL_UNITA NUMERIC(15, 5),
  VAL_INIZIALE NUMERIC(15, 5),
  TOT_INIZIALE NUMERIC(15,2),
  PRIMARY KEY (CODICE) );
```

Nella tabella ARTICOLI sono archiviati i dati identificativi essenziali relativi a tutti i prodotti registrati a magazzino:

CODICE codice articolo di tipo alfanumerico 8 caratteri;

DESCRIZIONE Descrizione Articolo Alfanumerico 45 caratteri;

UNITA\_MIS Alfanumerico 3 caratteri;

CATALOGO Tipologia Merceologica Alfanumerico 2 caratteri;  
 POSIZIONE Collocazione della merce nel Magazzino Alfanumerico 4 Caratteri;  
 TOT\_CARICATO quantità totale merce caricata in magazzino e perciò normalmente acquistata dell'anno finanziario; TOT\_SCARICATO quantità totale merce scaricata ai reparti nell'anno finanziario;  
 SCORTA\_MIN scorta minima di cui va garantita la presenza;  
 VALORI Valore complessivo del lotto di quell'articolo presente;  
 VAL\_UNITA Valore unitario medio;  
 TOT\_INIZIALE Giacenze iniziali (ad inizio dell'anno finanziario corrente);  
 VAL\_INIZIALE Valore complessivo delle giacenze iniziali;  
 COD\_FORNITORE Codice dell'ultima ditta fornitrice di quell'articolo;  
 DAT\_ACQ Data dell'ultimo acquisto effettuato da quella ditta;  
 QUANT Quantità di prodotti acquistati da quella ditta;

```

CREATE TABLE MAGAZZINO
(
  DATA    TIMESTAMP NOT NULL,
  N_MOVIMENTO  INTEGER NOT NULL,
  COD_ARTICOLO CHAR(8),
  R_F CHAR(1),
  COD_RF CHAR(8),
  UTENTE CHAR(8),
  CAUSALE CHAR(8),
  QUANTITA NUMERIC(15,2),
  N_DOCUMENTO  INTEGER,
  DATA_DOCTIMESTAMP,
  N_RIF_ORDINE  INTEGER,
  COSTO_ACQ NUMERIC(15, 5),
  IVA CHAR(3) DEFAULT 20,
  PRIMARY KEY (DATA, N_MOVIMENTO) );
  
```

Nella tabella MAGAZZINO sono archiviati i dati relativi alle operazioni di CARICO e SCARICO effettuate in Magazzino:

DATA Data+Orario in cui è stato registrato il movimento ,  
 N\_MOVIMENTO Numero progressivo del movimento max. 6 cifre assegnato in automatico a partire da 1 all'inizio di ogni anno finanziario;

COD\_ARTICOLO Codice dell'articolo movimentato;

R\_F ?????

COD\_RF Codice del fornitore in Caso di CA o RF, Codice Reparto in caso di Operazione SC o VE;

CAUSALE Tipologia Movimento Alfanumerico 2 caratteri : CA Carico da Fornitore, RF Reso a Fornitore, SC Scarico verso Reparto, VE Versamento da Reparto a Magazzino, ST Storno per registrazione errata;

QUANTITA Quantità movimentata ;

N\_DOCUMENTO Numero Bolla o Fattura Accompagnatoria in caso di operazioni CA, RF, SC, VE;

DATA\_DOC Data del Documento;

N\_RIF\_ORDINE Riferimento Nostro Buono Ordine;

COSTO\_ACQ Costo Unitario della merce Scontato e al netto dell'IVA;

IVA Imposta sul Valore Aggiunto applicato alla merce applicata

TABELLA DELLE PASSWORD DEGLI UTENTI ABILITATI (Supervisore accede a tutto; User legge scrive sul Data Base; Backup esegue Backup e Restore)

```
CREATE TABLE P_W
```

```
(  
  UTENTE CHAR(50) NOT NULL,  
  P_WORD CHAR(10),  
  PRIMARY KEY (UTENTE)  
);
```

Tabella dei REPARTI, UFFICI e CENTRI DI COSTO

```
CREATE TABLE REPARTI
```

```
(  
  CODICE CHAR(8) NOT NULL,  
  NOME CHAR(50),  
  REFERENTE CHAR(50),  
  TELEFONO CHAR(15),  
  NOTE CHAR(50),  
  PRIMARY KEY (CODICE)
```

);

Tabella dei Tipi di MATERIALI

CREATE TABLE TIPOLOGIA

```
(  
  CODICE CHAR(8) NOT NULL,  
  DESCRIZIONE CHAR(50) NOT NULL,  
  PRIMARY KEY (CODICE)  
);
```

Nella tabella TIPOLOGIA sono archiviati i dati relativi alle tipologie merceologiche (CODICE della Categoria Alfanumerico 8 caratteri; DESCRIZIONE Descrizione della Categoria o Settore : Alfanumerico 50 caratteri)

TABELLA DEGLI UTENTI (Responsabili di reparto, ATA, Bidelli, etc.)

CREATE TABLE UTENTI

```
(  
  CODICE CHAR(8) NOT NULL,  
  DESCRIZIONE CHAR(50),  
  REPARTO CHAR(8),  
  PRIMARY KEY (CODICE)  
);
```

TABELLA FORNITORI

CREATE TABLE FORNITORI

```
(  
  CODICE CHAR(8) NOT NULL,  
  R_SOCIALE CHAR(50),  
  CONTATTO CHAR(50),  
  INDIRIZZO CHAR(50),
```

CAP CHAR(5),  
P\_IVA CHAR(30),  
CITTA CHAR(20),  
PROVINCIA CHAR(2),  
TELEFONO CHAR(15),  
CELLCHAR(15),  
FAX CHAR(15),  
E\_MAIL CHAR(25),  
PRIMARY KEY (CODICE)  
);

Procedure Standard

Procedura di CARICO da FORNITORE

Procedura di SCARICO a REPARTO

Il programma deve emettere il documento di Scarico in duplice copia

Procedura di VERSAMENTO da REPARTO a MAGAZZINO

Il programma deve emettere il documento di Scarico in duplice copia

Procedura di RESO MERCE da MAGAZZINO a FORNITORE

Il programma deve emettere la Bolla di Accompagnamento in duplice copia

Tutte le procedure devono prevedere una fase di compilazione di un prospetto relativo ai singoli articoli movimentati con possibilità per l'operatore di effettuare correzioni per errori materiali prodotti da inserzioni errate. Una volta che l'operatore avrà confermato la registrazione deve intendersi definitiva ed eventuali errori vanno corretti unicamente utilizzando la procedura di STORNO (correzione di errore contabile di registrazione).

### **Stampe**

Giornale di Magazzino (parziale o complessivo)

Articoli (scheda articolo completa + dettaglio movimenti di CARICO e/o SCARICO effettuati in quell'anno o in un periodo predeterminato)

Reparti (Anagrafica + dettaglio PRELIEVI e VERSAMENTI effettuati da quel reparto in quell'anno o in un periodo predeterminato)

Fornitori (Anagrafica + dettaglio CARICHI e RESI effettuati da quel reparto da quel fornitore in quell'anno o in un periodo predeterminato)

Giacenze e Scorte (segnalazione Materiali Sottoscorta);

Anagrafica Articoli suddivisi x Tipologia Merceologica

Ricerche e Visualizzazioni a Video

(come stampe – preview di stampa a video)

Chiusure a fine anno finanziario

La chiusura dell'anno finanziario deve comportare il trasferimento automatico della giacenza di un determinato articolo nel campo TOT\_INIZ della tabella scheda e del valore complessivo delle giacenze in VAL\_INIZ; devono essere inoltre annullati il Totale Carico e Scarico.

La numerazione dei Movimenti prevista dal Giornale di Magazzino deve ripartire da 1 nel successivo anno finanziario; presumo che ciò implichi un backup annuale delle operazioni effettuate per eventuali ispezioni e/o controlli futuri.

Esportazione delle Tabelle Dati in formato CSV Accesso al DataBase

Per consentire eventuali elaborazioni esterne dei dati estemporanee usando pacchetti Microsoft. proporrei formato testo CSV (delimiter “;”)

Operazioni da proteggere e log delle operazioni

Registrare l'identificativo dell'operatore che accede al programma e da quale postazione accede e l'orario a cui accede;

Interfacciamento con software “gestione impegni”

Vediamo se possibile condividere le tabelle FORNITORI e ARTICOLI con applicativo ACCESS (ho trovato ODBC driver x Interbase sul CDROM di Delphi 6).

Installazione dei Programmi

Preparazione di un CD di installazione che consenta di reinstallare il programma completamente.

Documentazione tecnica

Descrizione del tracciato record delle tabelle usate nel programma.

Descrizione sintetica delle procedure di Installazione su Server e su Client.

Password di accesso al DataBase.

## **Obiettivi 2004**

- Passare a DataBase ACCESS
- Rifare programma semplificando le procedure di utilizzo
- *Inserire tutte le esportazioni in Excel e relative stampe*
- Interfacciarsi con DataBase Ordini e DataBase Inventario
- *Gestione dei DataBase dei vari Anni e Gestire il Backup*
- Esportare in MySQL via ODBC

## **Specifiche Tecniche del Programma Magazzino**

### **Stampe**

#### **Giornale di Magazzino** A3 orizzontale

(parziale da DATA1 a DATA2 o complessivo dal 1/1/anno\_corrente a 31/12/anno\_corrente stampa in formato A3 orizzontale; intestazione in alto a sinistra; numerazione delle pagine spazio a SX x rilegatura.

Voci obbligatorie da riportare ::

MOVIMENTO – Numero del Movimento max 4 ch.

DATA\_MOV – Data del Movimento

TIPO\_MOV – Tipo del Movimento : CA – SC – VE – RF – ST

QUANTITA –

VALORI – (positivo x un CA o VE; negativo x uno SC o RF)

COD\_ART –

DESCRIZIONE –

PROVENIENZA – Ragione Sociale del Fornitore

DESTINAZIONE – Laboratorio o Utente o Ufficio

DOCUMENTO – BOLLA in caso carico – PRELIEVO in caso di scarico

N\_DOCUM –

DATA\_DOC –

N\_ORDINE solo in caso di carico

NB E' possibile

**Scheda Articolo** A4 verticale

(scheda articolo completa di dettaglio movimenti di CARICO e/o SCARICO effettuati in quell'anno o in un periodo predeterminato a partire dalla giacenza al 1/1/anno\_corrente)

Voci da riportare :

MOVIMENTO - Numero del Movimento max 4 ch.

DATA\_MOV - Data del Movimento

TIPO\_MOV - Tipo del Movimento : CA - SC - VE - RF - ST

QUANTITA -

VALORI - (positivo x un CA o VE; negativo x uno SC o RF)

PROGRESSIVO - (somma progressiva Valori)

PROVEN/DESTIN - (soggetto o Laboratorio in caso di SC o Fornitore in caso di CA)

DOCUMENTO - BOLLA in caso carico - PRELIEVO in caso di scarico

N\_DOCUM -

DATA\_DOC -

**Scheda Reparti** A4 verticale

(Anagrafica + dettaglio PRELIEVI e VERSAMENTI effettuati da quel reparto in quell'anno o in un periodo predeterminato)

Voci da riportare :

MOVIMENTO - Numero del Movimento max 5 ch.

DATA\_MOV - Data del Movimento

TIPO\_MOV - Tipo del Movimento : SC - VE

QUANTITA -

VALORI - (positivo x un VE; negativo x uno SC)

COD\_ART -

DESCRIZIONE -

UTENTE - (chi ha effettuato l'operazione)

DOCUMENTO - VERSAMENTO in caso VE - PRELIEVO in caso di SC

N\_DOCUM -

DATA\_DOC -

**Scheda Fornitori** A4 verticale

(Anagrafica + dettaglio CARICHI e RESI effettuati presso quel fornitore in quell'anno o in un periodo predeterminato)

Voci da riportare :

MOVIMENTO - Numero del Movimento max 5 ch.

DATA\_MOV - Data del Movimento

TIPO\_MOV - Tipo del Movimento : CA - RF

QUANTITA -

VALORI - (positivo x un CA; negativo x un RF)

PROGRESSIVO - (somma progressiva Valori)

COD\_ART -

DESCRIZIONE -

DESCRIZIONE2 -

DOCUMENTO - BOLLA o FATTURA di ACQUISTO in caso CA o di RESO in caso di RF

N\_DOCUM -

DATA\_DOC -

N\_ORDINE solo in caso di carico

**Giacenze e Scorte** A4 verticale

(segnalazione Giacenze per tipologia merceologica con indicazione del Sottoscorta in ordine di COD\_ART o di DESCRIZIONE);

Voci da riportare :

COD\_ART -

DESCRIZIONE -

DESCRIZIONE2 -

GIACENZA INIZIALE (al 1/1/anno\_corrente)

TOTALE CARICHI

TOTALE SCARICHI

GIACENZA ATTUALE (alla data di stampa)  
SEGNALAZIONE SOTTOSCORTA  
CODICE DITTA (riferito ultimo acquisto)  
QUANTITA (riferito ultimo acquisto)  
COSTO (riferito ultimo acquisto)  
DATA (riferito ultimo acquisto)

**Anagrafica Articoli** A4 verticale

(stampa suddivisi per catalogo in ordine COD\_ART o di DESCRIZIONE);

Voci da riportare :

COD\_ART -  
DESCRIZIONE -  
DESCRIZIONE2 -  
DESCRIZIONE3 -  
GIACENZA ATTUALE (alla data di stampa)

**NOTE:**

In tutte le maschere ove figura il Data-Entry di una data è opportuno usare l'oggetto calendario;

Quando effettuo una operazione di carico o di scarico se non sono presenti tutti i dati compreso il riconoscimento del codice fornitore corretto e codice articolo corretto l'operazione aggiungi non deve essere consentita;

Sulla scheda articoli deve essere indicata chiaramente : Giacenza iniziale, Totale Carichi, Totale Scarichi e Giacenza Attuale.

SCARICHI da Magazzino verso reparto:

Lasciare libera la introduzione del numero documento e della data del documento

Non deve recuperare operazioni vecchie con lo stesso numero !!!!

Nel selezionare un articolo devo avere la indicazione della giacenza x verificare se lo scarico che sto effettuando è consentito o no;

Le operazioni di carico e di carico e scarico dopo la pressione di AGGIUNGI devo numerarle progressivamente 1, 2, 3, 4, 5 e elencarle in ordine progressivo di entrata.

La descrizione in fase di scarico è in alcuni casi insufficiente (troncamento caratteri)

URGENTE introdurre la numerazione da 1 in poi dal 1/1/anno\_corrente (risolvere il problema degli anni passati).

```
SET SQL DIALECT 1;
```

```
/* CREATE DATABASE 'D:\DB-090404\MAGAZ.GDB' PAGE_SIZE 4096
```

```
  DEFAULT CHARACTER SET */
```

```
/* Domain definitions */
```

```
CREATE DOMAIN TIPO_CODICE AS CHAR(8);
```

```
CREATE DOMAIN TIPO_QUANTITA AS NUMERIC(15, 2);
```

```
/* Table: ARTICOLI, Owner: SYSDBA */
```

```
CREATE TABLE ARTICOLI
```

```
(  
  CODICE      TIPO_CODICE NOT NULL,  
  DESCRIZIONE CHAR(50),  
  DESCRIZIONE2 CHAR(50),  
  DESCRIZIONE3 CHAR(50),  
  TIPOLOGIA TIPO_CODICE,  
  COD_FORNITORE TIPO_CODICE,  
  BARCODE CHAR(30),  
  COSTO      NUMERIC(15, 5),  
  U_MISURA CHAR(4),  
  POSIZIONE  CHAR(10),  
  DATA_ACQ TIMESTAMP NOT NULL,  
  QUANTITA_CORR TIPO_QUANTITA,  
  TOT_CARICATO TIPO_QUANTITA,  
  TOT_SCARICATO TIPO_QUANTITA,  
  SCORTA_MIN  TIPO_QUANTITA,  
  VALORI     NUMERIC(15, 5),  
  VAL_UNITA  NUMERIC(15, 5),  
  VAL_INIZIALE NUMERIC(15, 5),  
  TOT_INIZIALE TIPO_QUANTITA,
```

PRIMARY KEY (CODICE)

);

/\* Table: CAUSALE, Owner: SYSDBA \*/

CREATE TABLE CAUSALE

(

CODICE TIPO\_CODICE NOT NULL,  
DESCRIZIONE CHAR(50) NOT NULL,  
PRIMARY KEY (CODICE)

);

/\* Table: FORNITORI, Owner: SYSDBA \*/

CREATE TABLE FORNITORI

(

CODICE TIPO\_CODICE NOT NULL,  
R\_SOCIALE CHAR(50),  
CONTATTO CHAR(50),  
INDIRIZZO CHAR(50),  
CAP CHAR(5),  
P\_IVA CHAR(30),  
CITTA CHAR(20),  
PROVINCIA CHAR(2),  
TELEFONO CHAR(15),  
CELL CHAR(15),  
FAX CHAR(15),  
E\_MAIL CHAR(25),  
PRIMARY KEY (CODICE)

);

/\* Table: MAGAZZINO, Owner: SYSDBA \*/

CREATE TABLE MAGAZZINO

(

```

DATA      TIMESTAMP NOT NULL,
N_MOVIMENTO  INTEGER NOT NULL,
COD_ARTICOLO TIPO_CODICE,
R_F CHAR(1),
COD_RF  TIPO_CODICE,
UTENTE  TIPO_CODICE,
CAUSALE TIPO_CODICE,
QUANTITA TIPO_QUANTITA,
N_DOCUMENTO  INTEGER,
DATA_DOCTIMESTAMP,
N_RIF_ORDINE  INTEGER,
COSTO_ACQ     NUMERIC(15, 5),
IVA CHAR(3) DEFAULT 20,
PRIMARY KEY (DATA, N_MOVIMENTO)
);

```

```

/* Table: P_W, Owner: SYSDBA */

```

```

CREATE TABLE P_W
(
  UTENTE  CHAR(50) NOT NULL,
  P_WORD  CHAR(10),
  PRIMARY KEY (UTENTE)
);

```

```

/* Table: REPARTI, Owner: SYSDBA */

```

```

CREATE TABLE REPARTI
(
  CODICE  TIPO_CODICE NOT NULL,
  NOME    CHAR(50),
  REFERENTE  CHAR(50),
  TELEFONO CHAR(15),
  NOTE    CHAR(50),
  PRIMARY KEY (CODICE)
);

```

```
);
```

```
/* Table: TIPOLOGIA, Owner: SYSDBA */
```

```
CREATE TABLE TIPOLOGIA
```

```
(
```

```
  CODICE    TIPO_CODICE NOT NULL,  
  DESCRIZIONE CHAR(50) NOT NULL,  
  PRIMARY KEY (CODICE)
```

```
);
```

```
/* Table: UTENTI, Owner: SYSDBA */
```

```
CREATE TABLE UTENTI
```

```
(
```

```
  CODICE    TIPO_CODICE NOT NULL,  
  DESCRIZIONE CHAR(50),  
  REPARTO  TIPO_CODICE,  
  PRIMARY KEY (CODICE)
```

```
);
```

```
CREATE GENERATOR GEN_SUBIDX_MAG;
```

```
/* Grant Roles for this database */
```

```
/* Grant permissions for this database */
```

## Capitolo 2 – Tecnologie utilizzate

### I Database

#### Che cosa è un database?

Un database è un insieme organizzato d'informazioni. L'organizzazione è l'elemento fondamentale e richiede una fase di progettazione del db. Si crea un database nel momento in cui si raccolgono e si schedano dati per uno scopo specifico. Nella gestione dei database vengono utilizzati database operativi e database analitici.

I database operativi vengono utilizzati per raccogliere, modificare e mantenere dati su base aggiornata. I tipi di dati raccolti sono dinamici cioè cambiano costantemente e riflettono informazioni sempre aggiornate. Un database analitico raccoglie e traccia dati storici che sono legati ad un preciso momento. I tipi di dati raccolti sono statici cioè non vengono mai modificati. Le informazioni ottenute da un database analitico riflettono un momento particolare in cui i dati sono stati raccolti e non sono aggiornati. Esistono vari tipi di modelli di database. Il modello più diffuso dei database è il db relazionale, si tratta di db la cui organizzazione segue un modello teorico sviluppato fin dagli anni '70 dai ricercatori Boyce e Codd presso i laboratori Ibm. Si tratta del modello di db più diffuso.

Nella mia tesina ho utilizzato un database di tipo operativo. Questo database mi è servito per fare delle tabelle; utilizzando i comandi sql, il programma effettua delle "query" (interrogazioni) per cercare nel database le varie informazioni (ricerca dei destinatari, degli impegni, del tipo impegno e tabella luoghi). Un database è "un raccogliitore dei dati" strutturato in maniera particolare: ci sono varie tabelle in relazione ed ogni tabella è costituita da un insieme di righe (record) e colonne (campi). I campi corrispondono ai vari pezzetti di informazione che si vogliono memorizzare (ad esempio una tabella clienti avrà i campi nome, cognome, telefono). I record corrispondono alle varie entità di cui si vogliono memorizzare le informazioni (ad esempio in una tabella clienti ogni record corrisponderà ad un cliente). Ogni database viene studiato e strutturato in base alle esigenze del programma; ad ogni tabella viene assegnato un indice primario (chiamato primary key) allo scopo di evitare inserimenti doppi dei record e di indirizzare velocemente le interrogazioni fatte; tutto ciò significa che un errato studio di database può compromettere seriamente il buon funzionamento del programma che lo usa rendendo necessario la loro riscrittura (sia del database che del programma).

## Il linguaggio SQL

(Structured Query Language) •L'SQL è un "linguaggio" che permette di effettuare interrogazioni su un RDBMS (Relational DataBase Management System). Lo standard di riferimento prende il nome di ANSI SQL (SQL standardizzato della American National Standard Institute). Sebbene i produttori software affermino che il loro linguaggio è "ANSI Compliant" (compatibile con l'ANSI SQL) nella realtà i vari linguaggi presentano differenze fra di loro.

Il linguaggio SQL (Structured Query Language) è un linguaggio strutturato che viene utilizzato quando si ha l'esigenza di interrogare, gestire e inserire dei dati in Database. Questo potente linguaggio ci consente, come citato sopra, di aver accesso ai dati contenuti in un Database relazionale (Oracle, Sybase, informix, Microsoft SQL server, access, ecc.) e descrivere le informazioni che, chi utilizza il programma, intende visualizzare. Il linguaggio SQL ci permette anche di sfruttare un Database consentendone l'elaborazione. Come nella maggior parte dei linguaggi anche questo presenta alcune varianti a seconda degli standard presenti; dopo attente considerazioni abbiamo deciso che gli elementi da noi utilizzati sarebbero dovuti essere il più "standard" possibile per essere riconosciuti dalla maggior parte dei computer, quindi anche in questo caso abbiamo adottato un linguaggio molto diffuso nel mondo "ANSI" (American National Standard Institute).

Nella fattispecie in un database relazionale i dati vengono memorizzati in tabelle; ogni riga viene chiamata "record" e ogni colonna viene chiamata "campo". In primo luogo dobbiamo dire che ogni campo può contenere al suo interno informazioni di diverso tipo (stringhe, numeri, date e ore, ecc.); le operazioni più comuni eseguite sulle tabelle sono: SELECT (per selezionare degli elementi), INSERT (per inserire un nuovo record), UPDATE (per aggiornare un record), DELETE (per eliminare un record).

Per esempio se in una tabella chiamata "Anagrafica" vengono riportate le colonne: Nome, Cognome, Indirizzo, data di nascita possiamo, tramite l'istruzione select, prendere in considerazione solo alcune colonne:

```
SELECT Nome, Cognome  
FROM Anagrafica;
```

Dopo aver digitato questo comando otterremo un'altra tabella generata dal programma che contiene solo le colonne Nome e Cognome. Per quanto riguarda i nomi delle colonne non è possibile utilizzare il carattere "Spazio" al loro interno, In quanto non sopportato dall'SQL. Analizzando le due righe di codice scritto sopra a scopo di illustrativo notiamo che alla fine di ogni istruzione deve essere presente un (;) per segnalare la fine del comando stesso. Esistono anche altre istruzioni sopra citate, per esempio INSERT che ci consente di aggiungere alla nostra tabella un altro record (riga), oppure UPDATE che ci permette di aggiornare il contenuto di un record o una colonna, o DELETE che abilita ad eliminare un record o il suo contenuto.

Per poter mettere in relazione dei valori è necessario avere a disposizione della simbologia che ci permetta di compiere dei confronti. E quindi useremo "**OPERATORI RELAZIONALE**":

Gli operatori relazionali vengono utilizzati assieme alla clausola WHERE (DOVE) per specificare che solamente alcune righe (dette anche "record" del DBMS) della tabella verranno visualizzate in base al criterio specificato nell'istruzione Where.

=	UGUALE
<>	DIVERSO
<	"MINORE DI..."
>	"MAGGIORE DI..."
<=	"MINORE O UGUALE A ..."
>=	"MAGGIORE O UGUALE A..."

## Microsoft Access

Microsoft Access è un gestore di DATABASE (DBMS: DATABASE MANAGEMENT SYSTEM) relazionali. Oltre al motore del database (quella parte cioè del programma che gestisce la memorizzazione, ricerca , ecc..dei dati). Access include vari strumenti per la manipolazione dei dati, tra cui un vero e proprio linguaggio di programmazione derivato da Visual Basic (si chiama Visual Basic for Application ed è presente in tutto il pacchetto office).

### OGGETTI DI UN DATABASE

Un database di Access è costituito dai seguenti elementi:

- **TABELLE:** sono il cuore del database e contengono tutti i dati memorizzati; tutti gli oggetti successivi non possono memorizzare alcun dato, ma si limitano a “Pescare” i dati delle tabelle.

Buona parte della progettazione del database consiste nel progettare le tabelle nel modo corretto. Si possono anche usare per la gestione dei dati ma sono molto limitate graficamente.

- **QUERY:** si utilizzano per interrogare le tabelle, cioè per selezionare e combinare i dati nel modo opportuno.
- **MASCHERE:** si utilizzano per presentare i dati a video in modo significativo e comodo da gestire. Ad esempio permettono di vedere eventuali elementi multimediali (come immagini) memorizzate nel db, cosa non possibile direttamente nelle tabelle.
- **REPORT:** si utilizzano per presentare i dati in stampa; come per le maschere consentono di stampare i dati con una grafica impossibile da ottenere nelle tabelle.
- **MACRO:** consentono di creare dei programmi per la gestione dei dati senza bisogno di programmare (senza cioè scrivere direttamente le istruzioni).

Naturalmente non consentono di fare tutto quello che è possibile fare usando direttamente il linguaggio VBA (Visual Basic for Application), ma possono essere un buono inizio per i principianti e un buon aiuto per chi è un po' più esperto.

In Access le tabelle possono essere visualizzate in due modi possibili:

- **VISUALIZZAZIONE FOGLIO DATI:** permette di visualizzare e gestire i dati senza usufruire di grafica particolare; i dati sono mostrati in una tabella simile ai fogli di Excel.

- **VISUALIZZAZIONE STRUTTURA:** permette di definire i campi costituenti la tabella.

## **Borland Delphi 5**

Borland Delphi 5 è il linguaggio di programmazione che noi abbiamo adottato per gestire il database del magazzino dell'istituto I.P.S.I.A. MORETTO.

Il **Delphi** nasce come evoluzione del "Borland Turbo Pascal". Il **Pascal** è un linguaggio ad alto livello che è stato sviluppato alla fine degli anni sessanta dal professor Niklaus Wirth a Zurigo. Il suo scopo era quello di realizzare un linguaggio comprendente un piccolo numero di concetti fondamentali di programmazione sufficienti ad insegnarla in forma di disciplina logica e sistematica; nello stesso tempo voleva un linguaggio che fosse facile ed efficiente da implementare sulla maggior parte dei calcolatori. Il suo successo nel conseguire questo obiettivo può essere misurato dalla rapida diffusione dell'uso del PASCAL sia come linguaggio usato nella didattica dei principi di programmazione (si usa tutt'oggi nelle università), sia come linguaggio effettivo per scrivere software di base ed applicativo.

Oggi le varie versioni di Delphi sono disponibili in differenti edizioni:

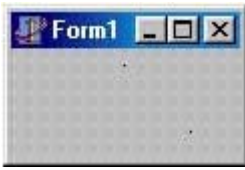
- Le "**Standard**" sono, come dice il nome stesso, il livello base di Delphi per la realizzazione di applicazioni semplici che non richiedono l'utilizzo di database e l'accesso ad Internet, perlopiù indirizzate a studenti, hobbisti, principianti.
- Le "**Developer**" o "**Professional**" sono rivolte a coloro che utilizzano il Delphi per lavoro e che necessitano di supporto per database e internet.
- Le "**Enterprise**", come dice il nome, sono rivolte alle aziende che necessitano del supporto per il controllo di versione, accesso a database sever, internet, internazionalizzazione del software.

Il Delphi é "object oriented" (cioè orientato agli oggetti) quindi la stesura del codice sorgente non è l'unica operazione da fare ma è accompagnata da un'impostazione grafica composta da finestre, bottoni, caselle di testo e molti altri oggetti.

Tra i vari componenti possiamo vederne alcuni:

- **FORM:** è un interfaccia grafica tipica di ogni applicazione basata su sistemi operativi di Windows. Essa ha la funzione di permettere all'utente di interagire con l'applicativo e quindi impartirgli degli "ordini" tramite l'uso dei vari oggetti "visuali"

che il programmatore include al suo interno, allo scopo di soddisfare le specifiche del programma.



- **BUTTON:** è uno degli oggetti standard che permette di interagire con il programma. La sua funzione principale è quella di far eseguire determinate operazioni alla sua pressione, che vengono stabilite dal programmatore tramite la scrittura di istruzioni particolari che vengono eseguite in sequenza.



- **LABEL:** è un' "etichetta" che non può essere modificata direttamente dall'utente che ha uno scopo principalmente indicativo (non può essere usata per fare una descrizione su più righe).



- **EDIT:** è una casella di testo sviluppata su un'unica riga che permette all'utente di inserire stringhe di testo allo scopo di fornire dei dati al programma. Questo può servire , per esempio, quando l'utente deve effettuare una ricerca o quando deve inserire una stringa richiesta dal programma stesso (per esempio una password).



- **MEMO:** è un oggetto con le stesse funzioni della Edit ma con la differenza che il testo immesso può essere distribuito su più righe. Per esempio lo si utilizza per l' inserimento di commenti o appunti.



- **CHECK BOX:** è un oggetto che ci permette di selezionare o meno un'opzione.



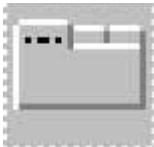
- **RADIO BUTTON:** è anch'esso un oggetto grafico, ma al contrario della check box ci consente di selezionare solo una delle varie opzioni.



- **COMBO BOX:** è una casella combinata che ci consente di selezionare una delle voci contenute al suo interno. Questo strumento ci permette di mantenere una visione globale delle possibili decisioni da prendere.



- **PAGE CONTROL:** è un componente che permette di distribuire i vari controlli (button, edit, label, ...) su più pagine attivabili cliccando sulle relative "linguette".



- **PANEL:** è un oggetto che ha lo scopo di "abbellire" il nostro programma; ci permette inoltre di inserire al suo interno degli oggetti, dividendoli magari per categoria, in modo da ottenere un'estetica il più intuitiva possibile.



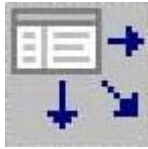
- **ADO QUERY:** è un oggetto non visuale che permette di eseguire qualsiasi operazione su un database di tipo Access. Per svolgere il suo compito utilizza il linguaggio SQL. Le operazioni solitamente più eseguite sono: ricerche (in Sql vengono chiamate select), inserimenti di nuovi dati (insert), modifiche a dati esistenti (update), eliminazione di dati (delete).



- **ADO CONNECTION:** è un oggetto non visuale e permette all'oggetto precedente di comunicare con il database fisico. Assieme ad ADO Query questo piccolo oggetto rappresenta uno degli oggetti più importanti del Delphi.



- **DATA SOURCE:** è lo strumento che permette di far leggere le informazioni contenute in un database Msaccess (.mdb) all'intero sistema.



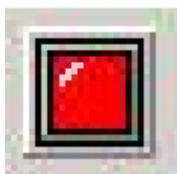
- **TIMER:** è un componente non visuale (che non appare a run-time ma svolge compiti specifici e a design-time sono rappresentati da un'icona sul form) che serve a generare degli eventi dopo un tempo che può essere stabilito dal programmatore.



- **LOG FILE:** è un oggetto non visuale che permette di registrare in un file di testo (log file) gli eventi più importanti di quel specifico programma.  
Con questo riusciamo a risalire ad eventuali problemi o solamente ad avere un resoconto del lavoro svolto.



- **HHALED:** è un oggetto che simula il funzionamento di un diodo led



## **Microsoft PowerPoint**

### **Cosa è PowerPoint?**

E' un programma di grafica di presentazione il cui obiettivo riguarda l'ambito della comunicazione. Consente di assemblare in modo rapido e intuitivo testi, colori, immagini, disegni, forme per creare presentazioni/messaggi efficaci.

Una presentazione (lezione di un docente, presentazione aziendale o quant'altro) è generalmente composta da più lucidi/diapositive. Scopo di PowerPoint è rendere più semplice possibile la creazione di sequenze di pagine (dette appunto lucidi o diapositive) da sottoporre a un pubblico per esporre un concetto, una lezione, un servizio...

Una presentazione può essere visualizzata direttamente sul Monitor del computer o proiettata, attraverso l'uscita Video del computer, su un grande schermo attraverso VideoProiettori.

### **Cosa posso fare con PowerPoint?**

- Digitare testo direttamente all'interno di una presentazione oppure scrivere il testo utilizzando un altro programma e poi importarlo.
- Definire un particolare layout per tutte le diapositive di una presentazione oppure personalizzare ognuna singolarmente.
- Organizzare la sequenza delle diapositive e controllare le transizioni da una all'altra premendo un semplice tasto.
- Inserire oggetti grafici, effetti sonori, filmati.
- Stampare su carta o salvare in formato HTML (per Internet ) la presentazione.

## Capitolo 3 – Struttura del database e importazione dati esistenti

Dato che la nostra conoscenza dei database è soprattutto orientata nell'ambiente Access e che il database del magazzino è un database realizzato in Interbase, abbiamo dovuto realizzare un piccolo programma che lo converta in un database in Access.

Per travasare i dati abbiamo dovuto creare la struttura del nuovo database (direttamente con Access) che deve avere le stesse identiche tabelle e gli stessi identici campi del vecchio database.

Esaminiamo ora la struttura del database "DBMagazzino.mdb".

Il database è composto da otto tabelle: ARTICOLI, CAUSALE, FORNITORI, MAGAZZINO, P\_W, REPARTI, TIPOLOGIA, UTENTI.

Ora analizzeremo ogni tabella specificando la struttura e la funzione di ogni campo.

### Tabella ARTICOLI

CODICE	DESCRIZIONE	DESCRIZIONE2	DESCRIZIONE3	TIPOLOGIA	COD_FORNITORE	BARCODE	COSTO

U_MISURA	POSIZIONE	DATA_ACQ	QUANTITA_CORR	TOT_CARICATO	TOT_SCARICATO	SCORTA_MIN

VALORI	VAL_UNITA	VAL_INIZIALE	TOT_INIZIALE

In questa tabella sono archiviati i dati identificativi essenziali relativi a tutti i prodotti registrati dal magazzino.

Essa è suddivisa in diciannove colonne denominate campi.

Il campo CODICE identifica in modo univoco ciascun articolo.

Il campo DESCRIZIONE contiene informazioni relative all'articolo, questo campo è obbligatorio.

I campi DESCRIZIONE2 e DESCRIZIONE3 contengono informazioni aggiuntive e non sono obbligatori.

Il campo TIPOLOGIA identifica attraverso una lettera il settore di competenza di quell'articolo (es. 5080 INTEGRATO NUTCHIP NUTD.1 DEA **C** (Articolo elettronico), 1002 BUSTE GIALLE 18X24 **V** (Articolo di cancelleria)).

Il campo COD\_FORNITORE contiene in modo univoco il codice della azienda che fornisce il materiale.

Il campo BARCODE contiene il codice a barre dell'articolo.

Il campo COSTO contiene il prezzo dell'articolo.

Il campo U\_MISURA definisce l'unità di misura adottato per quantificare l'articolo.

Il campo POSIZIONE è una sigla che identifica la posizione di quell'articolo all'interno del magazzino.

Il campo DATA\_ACQ specifica il giorno, il mese e l'anno in cui è stato acquistato il prodotto.

I campi QUANTITA\_CORRENTE, TOT\_CARICATO, TOT\_SCARICATO, SCORYA\_MIN riportano rispettivamente la quantità di un determinato articolo presente nel magazzino, il totale del materiale che viene acquistato e scaricato durante l'anno e la quantità minima che il magazzino deve mantenere.

Il campo VALORI contiene il valore patrimoniale della merce (riferita allo specifico articolo) presente nel magazzino.

Il campo VAL\_UNITA è il valore medio di un singolo articolo (VALORI/GIACENZA).

Il campo VAL\_INIZIALE contiene il valore patrimoniale della merce (riferita allo specifico articolo) presente nel magazzino al 1° gennaio.

Il campo TOT\_INIZIALE è la quantità della merce riferita a uno specifico articolo presente al 1 gennaio.

## **Tabella FORNITORI**

<b>CODICE</b>	<b>R_SOCIALE</b>	<b>CONTATTO</b>	<b>INDIRIZZO</b>	<b>CAP</b>	<b>P_IVA</b>

CITTA	PROVINCIA	TELEFONO	CELL	FAX	E_MAIL

In questa tabella sono inseriti i dati anagrafici dei fornitori del materiale scolastico.

### Tabella MAGAZZINO

N_MOVIMENTO	DATA	COD_ARTICOLO	COD_RF	UTENTE	CAUSALE

QUANTITA	N_DOCUMENTO	DATA_DOC	RIFERIMENTO	COSTO_ACQ	IVA

In questa tabella sono archiviate tutte le operazioni che si fanno nel magazzino.

E' suddivisa in 11 campi.

Il campo N\_MOVIMENTO è il numero progressivo del movimento con lunghezza massima di sei cifre assegnato in automatico a partire da 1 all'inizio di ogni anno finanziario.

Il campo DATA riporta la data in cui è stato registrato il movimento.

Il campo COD\_RF identifica il codice del fornitore in caso di un carico o di reso e il codice del reparto in caso di uno scarico o di un versamento mentre in caso di uno storno contiene il COD\_RF del movimento stornato.

Il campo UTENTE contiene il nome della persona che ha effettuato il movimento.

Il campo CAUSALE riporta la sigla del movimento che è stato fatto: CA Carico da Fornitore, RF Reso a Fornitore, SC Scarico verso Reparto, VE Versamento da Reparto a Magazzino, SC Storno di un Carico, SR Storno di un Reso, SS Storno di uno Scarico, SV Storno di un Versamento).

Il campo QUANTITA contiene la quantità di merce movimentata (es. 10 integrati NE555).

Il campo N\_DOCUMENTO contiene il Numero della Bolla o Fattura Accompagnatoria in caso di operazioni CA,RF,SC,VE.

Il campo DATA\_DOC identifica la Data del Documento.

Il campo RIFERIMENTO contiene nel caso di un carico i riferimenti al Buono d'Ordine, nel caso di un reso contiene il numero della bolla o fattura che ha comportato il carico della merce, nel caso di uno scarico o un versamento è vuoto mentre nel caso di uno storno contiene la sigla del movimento stornato.

Il campo COSTO\_ACQ è il costo Unitario della Merce Scontato e al netto dell'iva.

Il campo IVA contiene l'Imposta sul Valore Aggiunto della merce applicata.

### **Tabella P\_W**

<b>UTENTE</b>	<b>P_WORD</b>

In questa tabella sono inseriti gli utenti che possono accedere al database e le rispettive password.

### **Tabella REPARTI**

<b>CODICE</b>	<b>NOME</b>	<b>REFERENTE</b>	<b>TELEFONO</b>	<b>NOTE</b>

In questa tabella sono inseriti tutti i reparti della scuola.

Il campo CODICE identifica il Codice del reparto.

IL campo NOME contiene il nome del reparto.

Il campo REFERENTE contiene la persona che può effettuare un movimento per quel reparto (es. il professor Azzani può fare uno scarico per il reparto manutenzioni).

Il campo TELEFONO contiene i numeri telefonici interni all'istituto dei vari reparti.

Il campo NOTE contiene delle informazioni aggiuntive.

### **Tabella TIPOLOGIA**

<b>CODICE</b>	<b>DESCRIZIONE</b>

In questa tabella sono archiviati i dati relativi alle tipologie merceologiche; nel campo CODICE sono riportati i codici della Categoria mentre nel campo DESCRIZIONE la descrizione della Categoria o Settore.

### **Tabella UTENTI**

<b>CODICE</b>	<b>DESCRIZIONE</b>	<b>REPARTO</b>

In questa tabella sono archiviati i dati relativi agli utenti ; nei tre campi sono riportati i codici, i nomi e cognomi e i reparti delle persone autorizzate a richiedere del materiale dal magazzino.

# Capitolo 4 – Programma Magazzino 2004

All' avvio del programma ci si trova di fronte alla finestra gestione articoli.

Per inserire un nuovo articolo attivare la modifica cliccando sul pulsante

Per visualizzare un articolo cliccare sull'edit

L'articolo può essere inventariato mettendo il flag

E' possibile modificare il fornitore cliccando sull'edit

E' possibile modificare la tipologia cliccando sull'edit

Cliccare per salvare i dati su disco

In questa finestra è possibile visualizzare tutti gli articoli presenti nella tabella articoli del database Magazzino2004 e verificarne la quantità, quando la giacenza (cioè la quantità corrente è inferiore alla scorte minima lo sfondo delle due caselle di testo è rosso mentre se è superiore lo sfondo è verde.

E' possibile anche inserire un nuovo articolo selezionando il fornitore da una finestra che si apre cliccando sulla casella 'fornitore' e selezionando la tipologia da un'altra finestra che si apre cliccando sulla casella 'tipologia'.

Per salvare i dati bisogna cliccare sul pulsante 'salva su disco'.

Aprendo la finestra gestione fornitori appare questa videata.

E' possibile selezionare se è un carico o un reso

Cliccando è possibile vedere tutti i movimenti

Mettendo il flag il programma gli assegna il n° di inventario

I.P.S.I.A. MORETTO MAGAZZINO 2004

GESTIONE ARTICOLI | GESTIONE FORNITORI | GESTIONE REPARTI | GESTIONE STORNI | ESPORTAZIONI E STAMPE

CA-CARICO DA FORNITORE  RF-RESO A FORNITORE

INSERISCI NUOVO CARICO

NUMERO MOVIMENTO  
3956

NUMERO DOCUMENTO: 24567 | DATA DOCUMENTO: 28/05/2004 | RIFERIMENTO: 43435

FORNITORE: ARGO | ARGO SRL VIA G. NICASTRO, 51 RAGUSA

CODICE ARTICOLO: 1002 | BUSTE GIALLE cm 18X24 | QUANTITA' CORRENTE: 2855.000

QUANTITA: 10 | COSTO: 0,17 | IVA: 20 % | INVENTARIO

AGGIUNGI CARICO

18/06/2004 10.51.35

Cliccare per salvare i dati su disco

In questa finestra è possibile inserire un carico da fornitore e un reso a fornitore; anche qua è possibile inserire il fornitore dell'articolo cliccando sulla casella 'fornitore' e selezionare l'articolo da caricare cliccando sulla casella 'codice articolo'.

Cliccando sul pulsante 'aggiungi carico' il programma controlla se si sono inseriti tutti i campi obbligatori che sono le date, il fornitore, il codice articolo, il costo e il numero documento.

La finestra gestione reparti si presenta così

E' possibile selezionare se è  
Uno scarico o un versamento

Cliccando è possibile vedere  
tutti i movimenti

NUMERO MOVIMENTO  
3956

NUMERO DOCUMENTO: 34324  
DATA DOCUMENTO: 28/05/2004  
RIFERIMENTO: 434  
CODICE REPARTO: LE3-4

CODICE UTENTE: AZZANI  
UTENTE: AZZANI CLETO  
REPARTO: LAB. ELETTR. 3-4

CODICE ARTICOLO: INT01  
INTEGRATO LIN LM555-NE555  
QUANTITA' CORRENTE: 43.000

QUANTITA: 5  
COSTO: 0,14000  
INVENTARIO:

AGGIUNGI LO SCARICO

18/06/2004 11.00.45

Cliccare per salvare i dati su disco

Questa finestra differenzia dall'altra dal fatto che consente di inserire uno scarico e un versamento.

La finestra gestione storni si presenta nel seguente modo

Cliccando è possibile selezionare il movimento da stornare

I.P.S.I.A. MORETTO MAGAZZINO 2004

GESTIONE ARTICOLI | GESTIONE FORNITORI | GESTIONE REPARTI | GESTIONE STORNI | ESPORTAZIONI E STAMPE

GESTIONE DEGLI STORNI  
Correzione di Operazioni Contabili Errate

STORNO SCARICO A REPARTO

DATA REGISTRAZIONE  MOVIMENTO DA STORNARE

DATA DOCUMENTO  NUMERO DOCUMENTO  RIFERIMENTO

UTENTE

CODICE ARTICOLO

QUANTITA  COSTO  IVA  %

18/06/2004 11.10.43

Cliccare per salvare i dati su disco

In questa finestra è possibile stornare i movimenti errati.

## **Conclusione**

La stesura di questa tesina ci ha permesso di affrontare la soluzione di un problema reale e complesso con tutti i risvolti legati alle norme di contabilità, obblighi di legge relativi alle registrazioni, ecc e di approfondire la conoscenza sui DataBase, sull'interrogazione dei DataBase con il linguaggio SQL attraverso Access e sull'elaborazione dei dati.

Abbiamo utilizzato il linguaggio Delphi che già avevamo studiato negli anni passati e abbiamo potuto approfondire di molto il nostro bagaglio tecnico.