

TUTORIAL

Versione 2026

Le informazioni contenute nel presente documento non hanno natura contrattuale e possono essere soggette a modifiche senza preavviso.

Il software descritto nel presente manuale viene venduto con un contratto di licenza. Il software può essere utilizzato, copiato o riprodotto in base ai termini e alle condizioni del contratto.

Il presente manuale non può essere copiato, riprodotto o trasmesso, in tutto o in parte, in nessuna forma, su nessun supporto e per nessuno scopo che si discosti dall'utilizzo da parte dell'acquirente, senza autorizzazione scritta di TEKLYNX Corporation SAS.

©2026 TEKLYNX Corporation SAS,

Tutti i diritti riversati.

Indice

Informazioni su questo manuale 5

Convenzioni tipografiche 5

Informazioni sul prodotto 5

Collegamento ai database 6

Qualche richiamo utile 6

Installazione dell' origine data ODBC 7

Per creare delle variabili Database: 7

Importazione dei dati 8

Active Query Builder 9

Guida introduttiva 9

Aggiunta di un oggetto alla query 10

Modifica delle proprietà di un oggetto 11

Unione di tabelle 11

Organizzazione dei campi output 12

Definizione dei criteri 13

Definizione di una query con parametri 13

La griglia Risultato della query 14

Database Manager 16

L'interfaccia Struttura del database 16

L'interfaccia Modifica database 18

L'interfaccia Query del database 20

L'interfaccia Stampa 22

Creazione di un nuovo modello di etichetta da stampare 22

Selezione di un modello di etichetta esistente 22

Selezione della stampante da utilizzare 23

Configurazione della stampante selezionata 23

Formula ... 24
L'origine dati Formula ... 24
Le funzioni ... 24
Gli operatori ... 25
Le funzioni di calcolo dei caratteri di controllo ... 26
Le funzioni di conversione ... 29
Funzioni di conversione ATA 2000 ... 35
Le funzioni sulla data e l'ora ... 37
Le funzioni logiche ... 45
Le funzioni matematiche ... 46
Le funzioni sui testi ... 49
Definizione delle proprietà dell'origine dati Formula ... 55
Esercitazione : Calcolo del ”Modulo” specifico ... 56
Per calcolare il peso ... 56
Per aggiungere il risultato del calcolo del peso: ... 57
Per calcolare il carattere di controllo: ... 57
Per calcolare il dato da codificare: ... 58
Per creare il codice a barre: ... 58
Installazione di Network ... 59
Descrizione funzionale ... 59
Procedura d’installazione ... 59
Operazioni preliminari ... 59
Descrizione di Network Manager ... 60
Installazione di Network License Manager ... 60
Configurazione ... 61
Per configurare User Manager ... 61
Avvio di License Service ... 62
Per avviare Service Controller ... 62
Installazione del software sulle workstation ... 62
Per installare il software su una workstation: ... 62

Informazioni su questo manuale

Convenzioni tipografiche

Nel presente manuale vengono utilizzate diverse convenzioni per distinguere vari tipi di informazioni:

- I termini dell'interfaccia, come i comandi, vengono visualizzati in **grassetto**.
- I tasti sono scritti in maiuscolo. Ad esempio: "Premere il tasto MAIUSC".
- Gli elenchi numerati indicano una procedura da seguire.
- Quando accanto a un paragrafo compare la congiunzione -oppure-, significa che è possibile scegliere una procedura alternativa per svolgere un determinato compito.
- Quando un comando di menu include dei sottomenu, il nome del menu seguito dal comando da selezionare compare in grassetto. Quindi, "Scegliere **File** > **Apri**" significa selezionare il menu **File**, quindi il comando **Apri**.

Informazioni sul prodotto

Alcune funzionalità descritte nel manuale potrebbero non essere disponibili per il prodotto.

Fare riferimento alle specifiche tecniche fornite con il prodotto per un elenco completo delle funzionalità specifiche del software.

Collegamento ai database

Qualche richiamo utile

In questo capitolo scoprirete quanto è potente il vostro software. Infatti, metteremo in relazione un'etichetta (il contenitore) con un database (il contenuto). A tal fine, ricorreremo ai collegamenti **ODBC** (Open Data Base Connectivity) e **OLE DB** (Object Linking and Embedding Database).

Il database consente di immagazzinare dei dati. Tutti i dati sono disposti sotto forma di tabelle a due dimensioni, che si chiama relazione. Ogni riga della tabella si chiama record. Un record serve per gestire un oggetto le cui proprietà sono disposte nelle diverse colonne della tabella sotto forma di campi.

Un database può contenere più tabelle. Per mettere in relazione le diverse tabelle di uno stesso database, è possibile ricorrere ai collegamenti. Vedremo più avanti in questo capitolo, tramite un esempio concreto, come creare dei collegamenti.

ODBC

Si tratta di un standard di collegamento che consente l'accesso ai database. Utilizzando ODBC è molto semplice mettere in relazione un'applicazione come il vostro software di etichettatura con un certo numero di database.

Il software fornisce più driver ODBC in modo da accedere ai database più comuni. Alcuni driver sono elencati qui sotto:

- Microsoft Access Driver (*.mdb)
- Microsoft Excel Driver (*.xls)
- Microsoft FoxPro Driver (*.dbf)
- ...

OLE DB

Si tratta di uno standard di connessione che consente di accedere a tutti gli standard di database e ai dati immagazzinati nei sistemi di messaggistica.

La prima fase, prima di accedere ai dati, consiste nell'installare l'origine dati richiesta.

Installazione dell' origine data ODBC

Questo primo esempio consentirà di installare una origine dati che fa riferimento a un database Access™.

Per installare l'origine dati:

1. Aprite l'etichetta num de haut en bas.tpl posta nella cartella InstallDir\Samples\Forms\Tutorial dell'applicazione.

   Si tratta di una sopracoperta per CD che dovrà contenere il titolo dell'opera e i brani registrati.

   I dati da inserire nell'etichetta si trovano in un database chiamato VOLUMES OF CLASSICAL MUSIC.MDB.
2. Scegliete Origine dati > Database > Crea/Modifica una query. Appare la finestra di dialogo Definizione di una query ODBC .
3. Fate clic su alla destra dell'elenco Selezionate una origine dati. La finestra di dialogo Amministratore origine dati ODBC.

   L'ODBC è un componente completamente indipendente dal software di etichettatura. Potete anche accedere all'Amministratore origine dati ODBC tramite il pannello di controllo di Windows: scegliete Start Impostazioni Pannello di controllo e fate clic sull'icona ODBC.

4. Dal riquadro DSN utente, fate clic su Aggiungi... poi selezionate il driver Microsoft Access Driver (*.mdb) quindi fate clic su Fine. Appare la finestra Installazione di ODBC Microsoft Access.
5. Inserite un nome nella zona Nome origine dati, per esempio Classical music source e se necessario date una breve descrizione.

   Ora sappiamo che una origine dati è associata a un database. La fase successiva consiste nel selezionare il database utilizzato per il nostro esempio, ossia VOLUMES OF CLASSICAL MUSIC.MDB.

Per creare delle variabili Database:

1. Sempre dalla finestra Installazione di ODBC Microsoft Access, fate clic su Seleziona nella zona Database.
2. Selezionate il database posto nella cartella d'installazione del vostro software di etichettaturaInstallDir\Samples\Forms\Tutorial VOLMUSIC.MDB poi fate clic su OK.

3. Chiudete poi la finestra Installazione di ODBC Microsoft Access facendo clic su OK poi uscite dall'Amministratore origine dati ODBC facendo clic su OK.

Importazione dei dati

Una volta installata l'origine dati, possiamo importare i dati immagazzinati nel database per inserirli nell'etichetta.

Per importare i dati:

1. Se la finestra di dialogo Definizione di una query non è visualizzata sullo schermo, scegliete Unione Crea/Modifica una query.
2. A partire dall'elenco Selezionate una origine dati, scegliete Classical music source quindi, selezionate la tabella volumi.
3. Fate clic su Seleziona tutto poi su OK.

   Questa azione consente di creare automaticamente tutte le variabili Database a partire dai campi del database. I ramo Formula dalla scheda Origine dati e aggiungete una Formula.
4. Salvate il documento.

   L'applicazione chiede se desiderate salvare la query.
5. Fate clic su Sì, selezionate InstallDir\Samples\Forms\Tutorial e inserite VOLMUSIC.CSQ come nome di query.

Active Query Builder

Active Query Builder è un componente per la creazione di query che consente di creare query SQL complesse tramite un'interfaccia di creazione visiva.

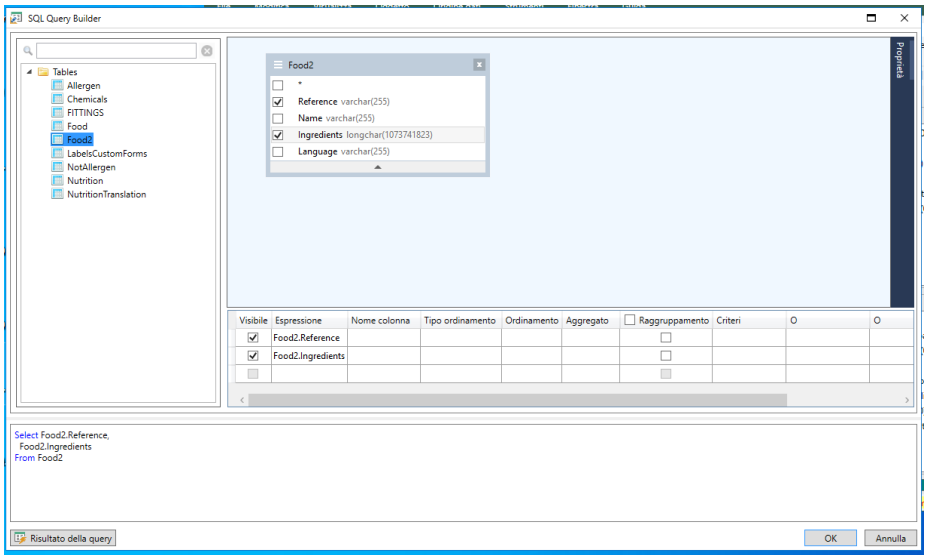
Per lavorare con Active Query Builder, è necessario possedere una conoscenza di base di SQL. Active Query Builder aiuta a scrivere correttamente il codice SQL risparmiando i dettagli tecnici, ma solo conoscendo i principi dei database SQL è possibile ottenere i risultati desiderati.

Guida introduttiva

Ecco come Active Query Builder appare all'apertura.

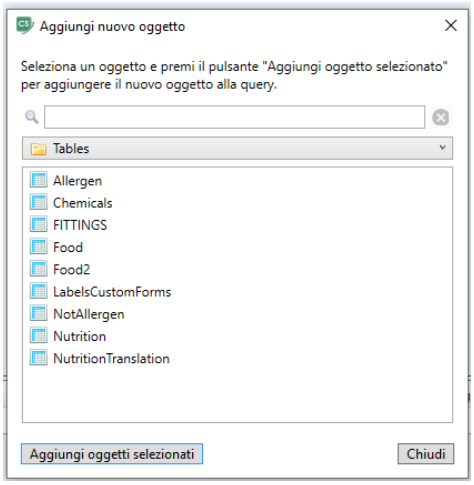
La finestra principale può essere divisa nelle seguenti sezioni:

- L'**Area di creazione della query** è l'area principale in cui viene mostrata la rappresentazione visiva della query. Quest'area consente di definire oggetti di database di origine e tabelle derivate, definire i collegamenti tra di loro e configurare le proprietà di tabelle e collegamenti.
- Il **Pannello colonne** si trova sotto l'area di creazione della query. Viene utilizzato per eseguire tutte le operazioni necessarie con le colonne di output delle query e le espressioni. Nel pannello è possibile definire gli alias di campo, dividere e raggruppare e definire i criteri.
- **L’area di visualizzazione della Tabella** si trova a sinistra. Qui è possibile cercare la propria query e individuarne rapidamente qualsiasi parte. Usare il campo Cerca nella parte superiore del riquadro per specificare la propria ricerca
- Il controllo della pagina sopra l'area di creazione delle query consente di passare dalla query principale alle sottoquery e viceversa.
- La piccola area nell'angolo dell'area di creazione delle query identificata con la lettera "Q" è il controllo di unione delle sottoquery. Qui è possibile aggiungere nuove sottoquery di unione ed eseguire tutte le operazioni necessarie.



Aggiunta di un oggetto alla query

Per aggiungere un oggetto alla query, fate clic con il tasto destro nell'area di creazione della query e selezionate **Aggiungi oggetto** nel menu a discesa.

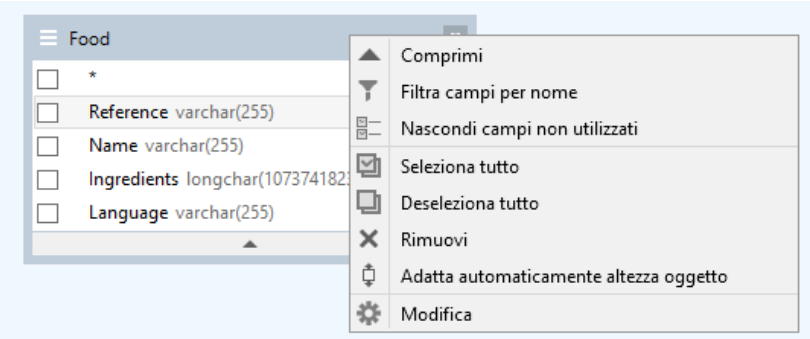


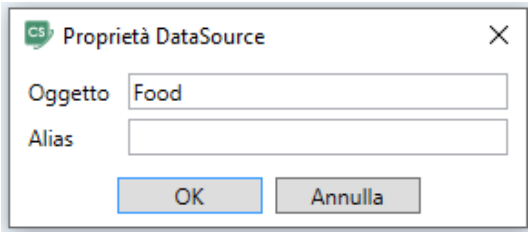
La finestra **Aggiungi nuovo oggetto** consente di aggiungere più oggetti contemporaneamente. Gli elementi della tabella sono elencati nella casella combinata. Il filtro **Mostra tutti gli oggetti** mostrerà tutti gli oggetti disponibili per la selezione. Il **campo Cerca** nella parte superiore della finestra è utilizzato per cercare un oggetto necessario. È possibile selezionare uno o più oggetti tenendo premuto il **tasto CTRL** quindi premere il pulsante **Aggiungi oggetti selezionati** per aggiungere questi oggetti alla query. È possibile ripetere questa operazione diverse volte. Dopo aver completato l'aggiunta di oggetti, fare clic sul pulsante **Chiudi** per nascondere questa finestra.

Per rimuovere un oggetto dalla query, fare clic sul pulsante **Chiudi** nell'intestazione dell'oggetto.

Modifica delle proprietà di un oggetto

È possibile modificare le proprietà di ciascun oggetto aggiunto alla query facendo clic con il tasto destro sull'oggetto e selezionando l'elemento **Modifica...** dall'elenco a discesa oppure facendo doppio clic sul nome dell'oggetto.

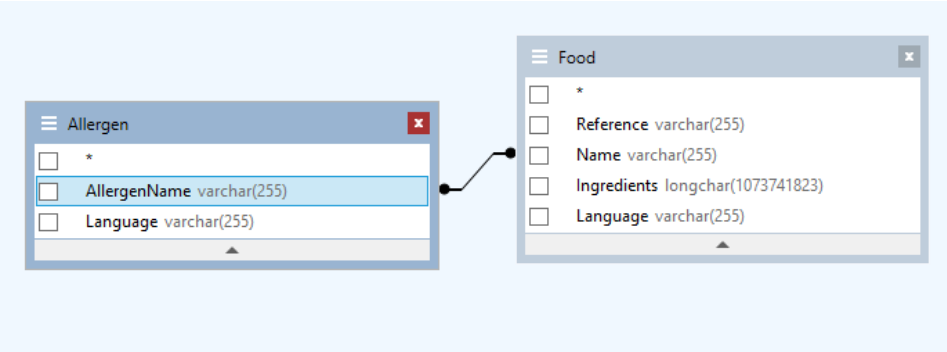




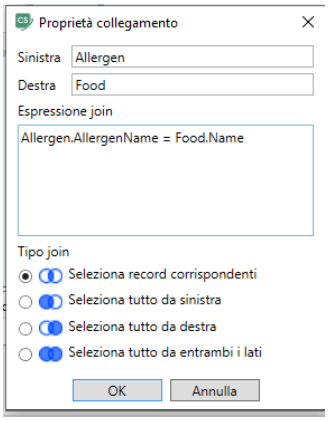
La finestra di dialogo **Proprietà del Datasource** può variare da server a server, ma almeno le proprietà Alias sono uguali per tutti i server di database.

Unione di tabelle

Per creare un collegamento tra due oggetti (per unirli) è necessario selezionare il campo tramite il quale collegare un oggetto a un altro e trascinarlo nel campo corrispondente dell'altro oggetto. Al termine dell'operazione, compare una riga che unisce i campi collegati.



Il tipo di collegamento predefinito è COLLEGAMENTO INTERNO, che fa sì che solo gli elementi corrispondenti di entrambe le tabelle vengano inclusi nel set di dati risultante. Per definire altri tipi di collegamenti è necessario fare clic con il tasto destro sul collegamento e selezionare l'elemento **Modifica...** nel menu a discesa o fare semplicemente doppio clic per aprire la finestra di dialogo **Proprietà del link**. Questa finestra di dialogo consente di definire il tipo di collegamento e altre proprietà di collegamento.



Per rimuovere un collegamento tra oggetti, fare clic con il pulsante destro del mouse sulla linea di collegamento e selezionare l'opzione **Rimuovi** nel menu a discesa, oppure selezionare la linea di collegamento e premere il pulsante **Elimina**.

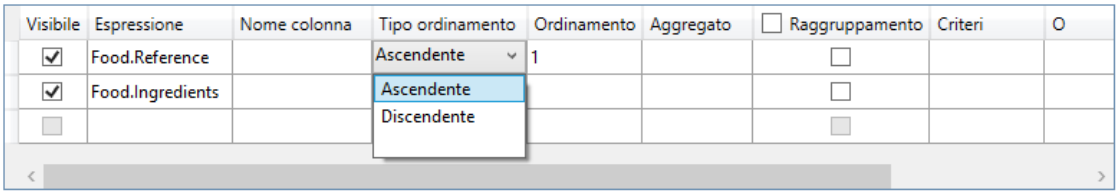
Organizzazione dei campi output

L'ordinamento dei campi output di una query può essere impostato utilizzando le colonne **Tipo di ordinamento** e **Criterio di ordinamento**.

La colonna **Tipo di ordinamento** consente di indicare il modo in cui i campi saranno organizzati: ordine **Ascendente** o**Discendente**.

La colonna **Criterio di ordinamento** consente di scegliere l'ordine in cui organizzare i campi.

Per disabilitare l'ordinamento per alcuni campi è necessario deselezionare la colonna **Tipo di ordinamento** per quel campo



Definizione dei criteri

Per definire i criteri dell'espressione riportati nel **Pannello colonne** è necessario utilizzare la colonna **Criteri**.

In questa colonna va scritto il criterio omettendo l'espressione. Per ottenere il seguente criterio in una query

> **WHERE** (campo >= 10) **AND** (campo <= 20)
> bisogna scrivere
> >= 10 **AND** <= 20 nella colonna **Criteri**.

È possibile inoltre usare l'**elenco a discesa** nella colonna **Criteri** per incollare espressioni e operatori comuni nel campo di modifica. Per la creazione di criteri avanzati, utilizzare il pulsante **Editore delle espressioni** nella colonna **Criteri**. Viene visualizzata la finestra di dialogo **Editor delle espressioni**.

È possibile indicare diversi criteri per una singola espressione utilizzando le colonne **O...**. Questi criteri verranno concatenati nella query utilizzando l'operatore **O**.

Definizione di una query con parametri

Query Builder consente di creare una query con parametri in cui il valore del parametro è contenuto in una variabile.

Nota: è necessario aver creato prima una variabile.

1. Trascinate la tabella nella quale verrà eseguita la query.
2. Selezionate i campi al quali applicare i criteri.

3. Nella colonna **Criterio** o nel campo di modifica del formato SQL, specificate la variabile da utilizzare come oggetto per un criterio di ricerca.

   Ad esempio: per cercare il valore della variabile Var0 creata in precedenza:

   - **In SQL:**

     SELECT [Tabella].*
     FROM [Tabella].*
     WHERE [Tabella].campo=
     APPLICAZIONE.DOCUMENTO.Var0

   - **Colonna criterio**
     = APPLICAZIONE.DOCUMENTO.Var0

4. Fare clic sul pulsante **Risultato della query** per visualizzare il risultato della query.

La griglia Risultato della query

Per accedere alla griglia **Risultato della query**, fate clic sul pulsante  dalla finestra di dialogo **Definizione di una query,** nella barra degli strumenti **Consultazione in database di unione** o dal menu **Origine dati > Database**.

Questa griglia consente di visualizzare il risultato di una query o di ricercare un termine particolare e tutte le sue occorrenze e infine di stampare le etichette corrispondenti.

Nota: Il tag **"<Dati binari>"** viene mostrato nella finestra di dialogo Risultato della query per i campi del database contenenti il **tipo di dati BLOB** (ad esempio, file immagini).

La griglia **Risultato di una query** contiene :

- **delle funzioni di ricerca**

  Campo di ricerca che consente di immettere il campo sul quale verrà effettuata la ricerca

  Valore di ricerca che consente di immettere il valore ricercato

  Ricercare il valore dovunque nel campo o all'inizio del campo .

- **delle funzioni di spostamento nei record del risultato della ricerca**

  Primo record 

  Record precedente 

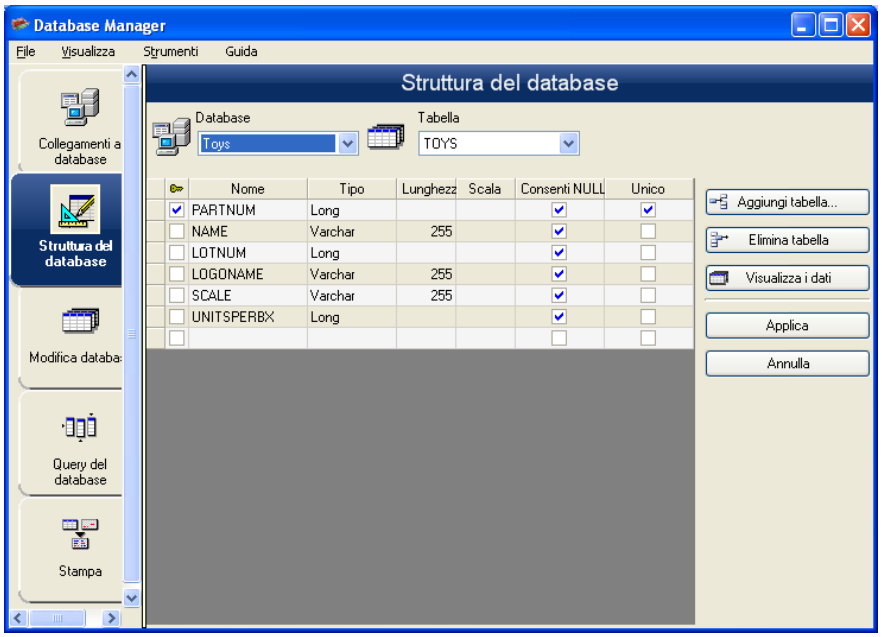
  Record seguente 

  Ultimo record 

- **La griglia di risultato**
  Visualizza le occorrenze del risultato della query.
- **Ripeti query**
  Eseguire nuovamente la richiesta e aggiornare la griglia.

Database Manager

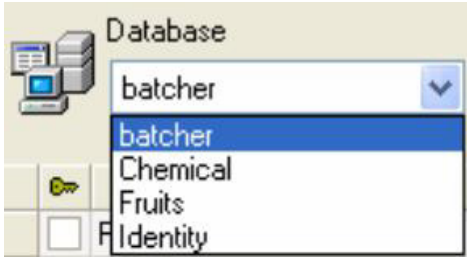
L'interfaccia Struttura del database



Interfaccia nella quale si gestisce la struttura del file di database: aggiunta / modifica / eliminazione di tabelle, campi...

Scelta di un database nella lista dei collegamenti

1. Fate clic sull'elenco a discesa **Database**.
2. Fate clic sul valore desiderato.

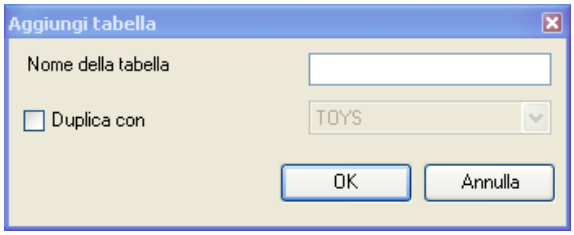


Scelta di una tabella in un database

1. Fate clic sull'elenco a discesa **Tabella**.
2. Fate clic sul valore desiderato.

Aggiunta di una tabella nel database attivo

1. Fate clic sul pulsante **Aggiungi tabella**.
2. Inserite il nome della nuova tabella.
3. Fate clic sul pulsante **OK**.

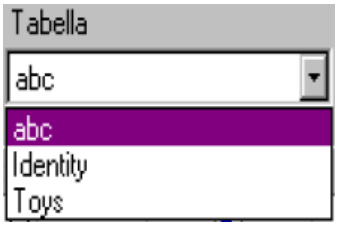


È anche possibile copiare la struttura della tabella su una tabella già esistente nel database selezionato. Per fare questo:

1. Fate clic nella casella di controllo **Duplica con.**
2. Fate clic nell'elenco a discesa.
3. Fate clic sul valore desiderato.
4. Fate clic sul pulsante **OK**.

Eliminazione di una tabella nel database attivo

1. Fate clic nell'elenco a discesa **Tabella**.
2. Fate clic sul valore desiderato.



3. Fate clic sul pulsante **Elimina la tabella**.

Visualizzare / nascondere i dati della tabella attiva

1. Fate clic sul pulsante Visualizza i dati.

Determinazione di un campo come campo chiave

1. Fate clic nella casella di controllo del campo desiderato.
2. Fate clic sul pulsante **Applica**.

Determinazione del tipo di contenuto di un campo

1. Fate clic nella casella **Tipo** del campo desiderato.
2. Fate clic sul pulsante dell'elenco a discesa.
3. Fate clic sul valore desiderato.
4. Fate clic sul pulsante **Applica**.

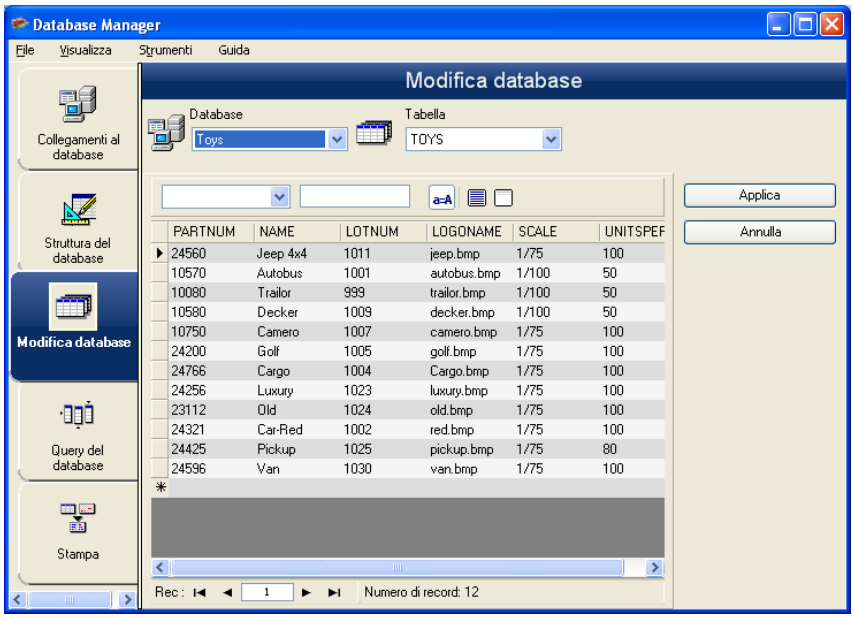
Determinazione della grandezza massima di un campo

1. Fate clic nella casella Lunghezza del campo desiderato.
2. Inserite il valore desiderato.
3. Fate clic sul pulsante **Applica**.

Accettazione di un campo vuoto

1. Fate clic nella casella di controllo **Consenti Null** del campo desiderato.
2. Fate clic sul pulsante **Applica**.

L'interfaccia Modifica database



Interfaccia nella quale si gestisce il contenuto del file di database: aggiunta / modifica / eliminazione di dati.
Queste operazioni dipendono dal tipo di database. Non è possibile modificare i record dei file di Excel.

Selezione dei record in base al loro contenuto

1. Fate clic sul pulsante dell'elenco a discesa.
2. Fate clic sul valore desiderato.
3. Fate clic nel campo di immissione.
4. Inserite il valore desiderato nel campo di immissione.

Selezione di tutti i record identici

1. Fate clic sul pulsante dell'elenco a discesa.
2. Fate clic sul valore desiderato.
3. Fate clic nel campo di immissione.
4. Inserite il valore desiderato nel campo di immissione.
5. Fate clic sul pulsante **Seleziona tutto**.

Selezione di un record identico

Per selezionare un record: Primo, Precedente, Prossimo, Successivo, fate clic sul pulsante 1, 2, 3 o 4 dello strumento di ricerca.



Creazione di un nuovo record

1. Fate clic in un campo della riga di immissione, indicata con un asterisco.
2. Inserite i valori desiderati nei campi corrispondenti.
3. Fate clic sul pulsante **Applica**.

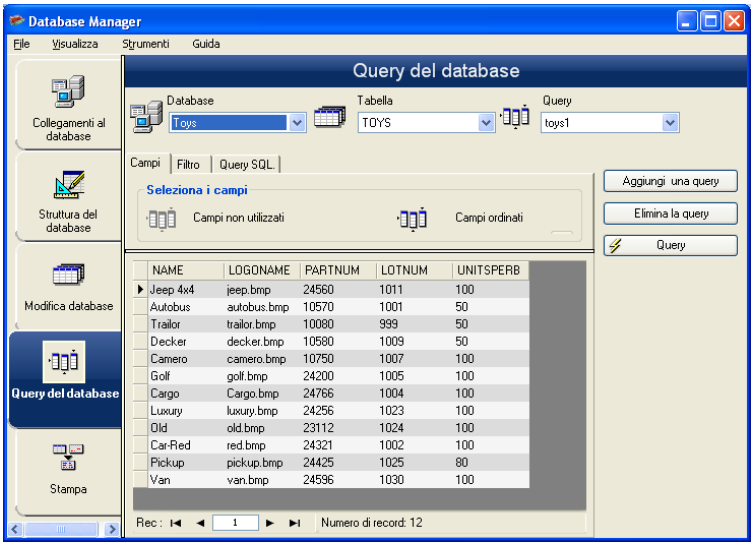
Duplicare un record

1. Fare clic o doppio clic su una riga o un gruppo di righe per evidenziare i record selezionati.
2. Fare clic con il pulsante destro del mouse per visualizzare il menu di contesto.
3. Selezionare **Duplica i record** selezionati.

Eliminazione di un record

1. Fate clic nella casella del cursore del record del campo desiderato.
2. Fate clic con il pulsante destro nella casella del cursore del record del campo desiderato.
3. Fate clic sul menu di scelta rapida Elimina record selezionato.

L’interfaccia Query del database



Interfaccia dove si creano e si applicano diversi filtri.

Aggiunta di una query

1. Fate clic sul pulsante **Aggiungi una query**.
2. Inserite il nome della query.
3. Fate clic sul pulsante **OK**.

Selezionare / deselezionare uno o più campi

Per selezionare o deselezionare uno o più campi:

1. Fate clic sul pulsante  .
2. Fate clic sul pulsante Query.

Modifica dell’ordine dei campi selezionati

1. Fate clic sul campo desiderato nella finestra **Campi ordinati**.
2. Fate clic sul pulsante "freccia su" o "freccia giù" fino al valore desiderato.

Creazione di un filtro tramite dati predefiniti

1. Fate clic sul pulsante Aggiungi la riga ().
2. Fate clic nel campo Campo.
3. Fate clic sul pulsante dell’elenco a discesa.

4. Fate clic sul valore desiderato.
5. Fate clic nel campo Operatore.
6. Fate clic sul pulsante dell’elenco a discesa.
7. Fate clic sul valore desiderato.
8. Fate clic nel campo Valore.
9. Inserite il valore desiderato.
10. Fate clic sul pulsante **Query**.

Applicazione di un operatore logico a più filtri

È necessario creare diversi filtri.

1. Fate clic nel campo **Logico**.
2. Fate clic sul pulsante dell’elenco a discesa.
3. Fate clic sul valore desiderato.
4. Fate clic sul pulsante **Query** per applicare e visualizzare le modifiche.

Ordinamento dell’elenco dei filtri

È necessario creare diversi filtri.

1. Fate clic sulla casella del cursore del campo desiderato.
2. Fate clic sul pulsante "freccia su" o "freccia giù" fino al valore desiderato.

Eliminazione di un filtro

È necessario creare almeno un filtro.

1. Fate clic sulla casella del cursore del campo desiderato.
2. Fate clic sul pulsante **Elimina la riga**.

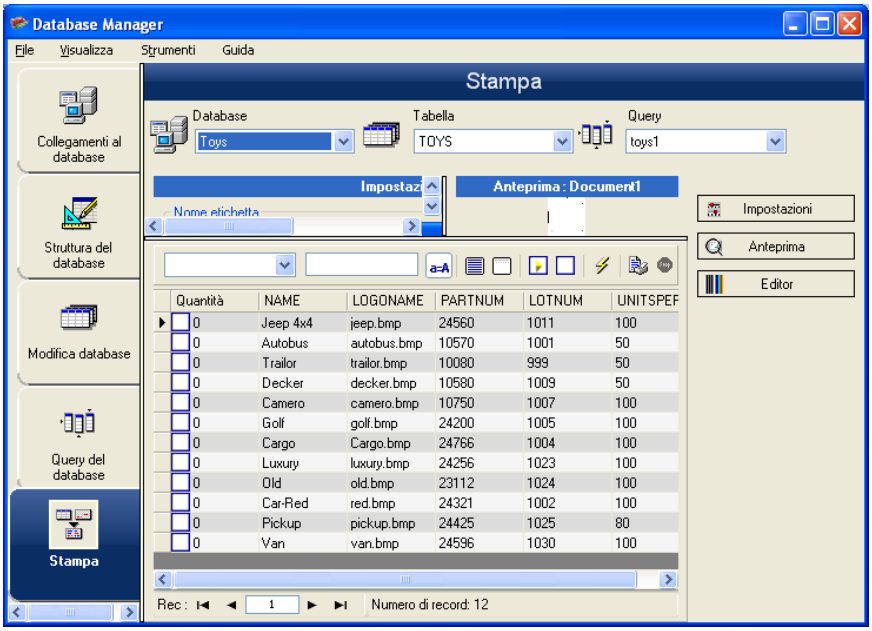
Modifica di un filtro in linguaggio SQL

È necessario creare almeno un filtro.

1. Fate clic nella casella di controllo **Modifica la query in linguaggio SQL** per attivare la query in linguaggio SQL e modificarla manualmente.
2. Fate clic sul pulsante **Query**.

Nota : È possibile scrivere il filtro direttamente in linguaggio SQL senza averlo creato prima nella scheda "Filtro".

L’interfaccia Stampa



Interfaccia in cui si seleziona il file di stampa, la stampante dedicata e i diversi parametri prima di lanciare la stampa.

Creazione di un nuovo modello di etichetta da stampare

1. Fate clic sul pulsante Creazione guidata etichette.
2. Seguite le istruzioni della procedure guidata.

Nota: La creazione di un’etichetta in relazione con il database consente di definire esattamente gli elementi necessari alla disposizione dei diversi campi del database.

Selezione di un modello di etichetta esistente

1. Fate clic sul pulsante Apri un’etichetta esistente.
2. Selezionate un file .lab.
3. Fate clic sul pulsante **OK**.

Nota: I pulsanti di opzione "Campo" nei gruppi di opzioni "Nome etichetta" e "Nome stampante" consentono di scegliere l’etichetta o la stampante desiderata quando queste sono definite in un campo del database attivo.

Selezione della stampante da utilizzare

Dev’essere selezionato un modello di etichetta.

1. Fate clic sul pulsante **Aggiungi o elimina una stampante** ().
2. Selezionate la stampante desiderata.
3. Fate clic sul pulsante **Conferma**.

Nota: L’ultima stampante usata è selezionata per default.

Configurazione della stampante selezionata

Dev’essere selezionato un modello di etichetta.

Dev’essere selezionata una stampante.

1. Fate clic sul pulsante Impostazioni della stampante ().
2. Eseguite le impostazioni desiderate.
3. Fate clic sul pulsante **OK**.

Formula

L'origine dati Formula

Comando **Origine dati > Formula > Aggiungi**

L'origine dati **Formula** contiene un elenco di origini dati che sono state create dall'utente. Tali origini dati vengono popolate da combinazioni di operatori, costanti, variabili, variabili di controllo, formule e funzioni. I dati possono essere numerici o alfanumerici.

Per effettuare un calcolo all'interno di un documento, è necessario creare una origine dati Formula.

Per questa origine dati esiste una finestra di dialogo specifica che consente di definire le funzioni necessarie alla formula data.

Le funzioni

Nota: Le impostazioni regionali del sistema Windows potrebbero influire sull'output di alcune formule.

Le funzioni sono delle formule predefinite che eseguono dei calcoli utilizzando dei valori particolari chiamati argomenti, in un certo ordine chiamato sintassi.

Le funzioni sono utilizzate per restituire un valore di tipo numerico, stringa di caratteri o logica derivante da un calcolo o da una operazione.

Esistono gruppi di funzioni nella definizione delle formule:

- Funzioni di calcolo dei caratteri di controllo (check digits)
- Funzioni di conversione
- Funzioni di conversione ATA 2000
- Funzioni sulla data e l'ora
- Funzioni logiche
- Funzioni matematiche
- Funzioni sulle stringhe di caratteri

Gli operatori

Il programma comprende gli operatori aritmetici, comparativi, di concatenazione e logici.

Gli operatori aritmetici

Operatore	Usato per
*	Moltiplicare due numeri
+	Aggiungere dei numeri
-	Sottrarre un numero da un altro o assegnare un valore negativo a un operando
/	Dividere un numero per un altro
^	Elevare un numero alla potenza dell'esponente
%	Modulo

Nota: Alcune formule non restituiscono un errore quando normalmente ci si aspetterebbe che lo facciano. Alcuni esempi sono: 1/0, 0/0, 0,0/0,0, 1/000 e 0^(-2). Questo è intenzionale e per evitare di bloccare il processo di stampa. L'utente deve definire il desiderato comportamento aggiungendo controlli aggiuntivi, se necessario.

Gli operatori di confronto

Operatore	Significato
<	Inferiore a
<=	Inferiore o uguale a
>	Superiore a
>=	Superiore o uguale a
=	Uguale
<>	Diverso

L'operatore di concatenazione

Serve a combinare due stringhe.

Operatore	Significato
&	Concatenazione di due stringhe

L'operatore logico

(vedere anche le funzioni logiche)

Operatore	Significato
!	Non logico

Le funzioni di calcolo dei caratteri di controllo

Nota: Le impostazioni regionali del sistema Windows potrebbero influire sull'output di alcune formule.

addmodulo10 (*stringa*): restituisce il valore intero con il carattere di controllo modulo 10 aggiunto alla fine.

addmodulo10_212(stringa) : restituisce un carattere di controllo modulo 10_212 (una variante del carattere di controllo standard modulo 10), e include la stringa testo corrente e la checksum alla fine della stringa.

addmodulo43 (*stringa*): restituisce il valore intero con il carattere di controllo modulo 43 e include la stringa testo corrente e il carattere di controllo alla fine della stringa.

bimodulo11 (*stringa*): restituisce 2 caratteri di controllo modulo 11 (code 11).

canadacustomscd (*stringa*): restituisce il carattere di controllo del Canada Customs Standard.

Check103*(«stringa»)*: Questa funzione calcola Mod103.

check128 (*stringa*): restituisce il carattere di controllo del codice 128 e EAN 128.

checkPZN(«stringa»): calcola il carattere di controllo per il codice a barre PZN. Il codice PZN si basa sul Codice 39 utilizzato per i prodotti farmaceutici in Germania.

CheckSum2Mod47*(«stringa»)*: Questa funzione calcola Mod47.

CheckSumMod10*(«stringa»)*: Questa funzione calcola Mod10.

CheckSumMod10_Codabar*(«stringa»)*: Questa funzione calcola una somma di controllo specifica Mod10per Codabar.

CheckSumMod10_MSI*(«stringa»)*: Questa funzione calcola la somma di controllo Mod10per il codice a barre MSI.

CheckSumMod11_3Suisse*(«stringa»)*: Questa funzione calcola la somma di controllo Mod10per 3Suisses.

CheckSumMod34*(«stringa»)*: Questa funzione calcola Mod34.

checkupce (*stringa*): restituisce il carattere di controllo per UPC-E (6 caratteri obbligatori).

CRC16(«stringa»): restituisce il valore CRC-16.

CRC16("123456789") = 47933
"0x" & hex(CRC16("123456789"), 4) = 0xBB3D

modulo10 (*valore*): restituisce un carattere di controllo modulo 10 del codice 2/5 intrecciato, EAN13, EAN 8, EAN 128 K-Mart, VPCA.

modulo10_212(stringa) restituisce una variante del carattere di controllo modulo 10

modulo10IBM(stringa) : restituisce un carattere di controllo IBM modulo 10 (una variante del carattere di controllo standard modulo 10)

modulo10UPS(stringa) : restituisce un carattere di controllo UPS modulo 10 (una variante del carattere di controllo standard modulo 10)

modulo11 (*stringa*): restituisce un carattere di controllo modulo 11 (codice 11).

modulo11IBM(stringa) : restituisce un carattere di controllo modulo 11 IBM (una variante del carattere di controllo standard modulo 11)

modulo24 (*stringa*): restituisce un carattere di controllo modulo 24 (codice PSA).

modulo32 (*valore*): restituisce un carattere di controllo modulo 32 (industria farmaceutica Italia).

modulo43 (*stringa*): restituisce un carattere di controllo modulo 43 (codice 39).

modulo47 (*stringa*): restituisce due caratteri di controllo modulo 47 (codice 93).

plessey (*stringa*): restituisce due caratteri di controllo (codice Plessey).

pricecd (*stringa*): restituisce 4 caratteri di controllo UPC Random Price.

pricecd5 (*stringa*): restituisce 5 caratteri di controllo UPC Random Price.

StringToExt39*(«stringa»)*: Questa funzione prepara la stringa da utilizzare nel codice a barre Extended 39.

UPSCheckDigit («stringa»): restituisce i caratteri di controllo UPS. Il metodo si basa sull'inserimento di una stringa. La parte restante del numero di tracciabilità è lasciato all'utente.

Viene utilizzata una sequenza di 15 caratteri, da cui si ottiene il numero di controllo

1 - I primi due caratteri devono essere "1Z".

2 - I successivi 6 caratteri devono corrispondere al numero di account UPS "XXXXXX".

3- I 2 caratteri successivi indicano il tipo di servizio:

- "01" per Next Day Air (spedizioni con consegna garantita il giorno lavorativo successivo).

- "02" per Second Day Air (spedizioni con consegna garantita il secondo giorno lavorativo).

- "03" per Ground (spedizioni via terra).

4- I 5 caratteri successivi si riferiscono al nostro numero di fattura (le nostre fatture sono di 6 cifre; viene eliminata la prima cifra; ad esempio la fattura 123456 genera il codice 23456).

5- Le ultime 2 cifre sono il numero di imballo con aggiunta dello zero. Es: imballo 1 dà "01", imballo 2 dà "02".

6- L'ultimo carattere è quello di controllo.

Nota: la sequenza appena descritta è composta da 17 caratteri. Di questi solo 15 sono necessari per calcolare la cifra di controllo. Eliminare quindi "1Z", quindi utilizzare gli ultimi 15 caratteri del metodo.

Le funzioni di conversione

Nota: Le impostazioni regionali del sistema Windows potrebbero influire sull'output di alcune formule.

AI253 (stringa): Funzione specifica per la preparazione della stringa per l'ID dell'applicazione 253.

AI8003 (stringa): Funzione specifica per la preparazione della stringa per l'ID dell'applicazione 8003.

ASCII (stringa) : restituisce il codice ASCII del primo carattere dell'argomento stringa.

Esempio :
ascii("A") = 65

char (valore)

- er valori da 128 a 255: indica il valore che corrisponde all'argomento integer nella tabella ASCII.
- Per tutti i valori negativi, i valori tra 0 e 127 e maggiori di 255: restituisce il carattere Unicode che corrisponde all'argomento integer.

Esempio:
char (65) = "A"

Nota: l'applicazione supporta i caratteri ASCII estesi.

CodabarData(«data», «start», «stop»): regola i dati per CodabarData.

Code128CData(«stringa»): regola i dati per il codice 128 C.

Currencytoeuro (valore) : converte la valuta corrente in Euro.

DatabarData(«data», «min», «max», «hasComposite»): regola i dati per DatabarData.

DBCSToUnicode(«stringa»,«codepage»): La formula contiene 2 parametri – I dati «stringa» per convertire, e «codepage» usata. Il parametro «codepage» può essere il nome di una lingua in inglese (Thai, Japanese, ChineseGBK, Korean, ChineseBig5, European, Eastern, Cyrillic, Greek, Turkish, Hebrew, Arabic, Baltic, Vietnamese, UTF-8, ACP) o un valore numerico (ricerca identificatori di pagina di codice su Internet).

Esempio:

DBCSToUnicode(unicodetoDBCS("傍傍傍傍", "UTF-8"), "UTF-8") =傍傍傍傍

Questa funzione è necessaria per i dati che provengono da fonti non-Unicode di file o dati (Database ad esempio).

Note: funzione inversa = UnicodeToDBCS.

dollar () : modifica un campo per rappresentare un prezzo.

Esempio:

Se un campo chiamato PRICE visualizza un valore 199, allora:

dollar(PRICE) produrrà $199,00.

Ean128Data («stringa_1»,«stringa_2»): Adjust "string_1" data for EAN 128 barcode according to template "string_2"

Eurotocurrency (valore) : converte l'Euro in valuta corrente.

Nota : il numero di decimali da visualizzare dev'essere definito nella scheda Output della variabile Formula, selezionando la casella di controllo Visualizza i decimali. D'altra parte, il tasso di conversione utilizzato è quello definito nella scheda Altre della finestra di dialogo Opzioni.

FileToData(«fileName», «errorData», «maxSize»): restituisce i dati letti dal file.

fixed (valore , nb_decimali, non_sep) restituisce una stringa di caratteri formattata con un numero di decimali uguale a *nb_decimali* e con o senza separatore delle migliaia.

Esempi:

fixed (1234.5678, 3, TRUE) = "1234.568"

fixed (1234.5678, 3, FALSE) = "1,234.567"

Nota : il separatore delle migliaia da visualizzare dev'essere definito nella scheda Output della variabile Formula, selezionando la casella di controllo Visualizza i decimali.

FormatDate(«date», «dateFormat», «localeID»): converte «date» in una stringa formattata in base a «dateFormat».

Valore restituito	stringa	un valore di data in formato di stringa, formattato secondo un formato.
date	stringa	Valore di data in formato di stringa.
dateFormat	stringa / numero	Formato del valore data di base. - I numeri codificano e formati predefiniti (per es. 1 dd/mm, 2 dd mmmm ecc. I valori possibili sono disponibili qui. - 0 o valori negativi indicano l'utilizzo del formato di data corrente - Il valore stringa codifica direttamente il formato (per es. "yyyy/mm/dd")
localeID	numero	Facoltativo (impostazione predefinita: 0, saranno utilizzate le impostazioni locali del sistema). Questo parametro specifica le impostazioni locali del sistema utilizzate per visualizzare il valore della data.

FormatMoney(«amount», «use separation characters(T/F)», «price characters», «force leading zaro (T/F)», «location for price character», «# of decimal places») Questa funzione consente di aggiungere simbolo della valuta, punti, forzare lo zero iniziale, utilizzare caratteri di separazione.

Esempi:

FormatMoney("123456.234", "T","$","F",0,2) = $123,456.23

FormatMoney("1234","F","$","T",8,2) = $1234.00

FormatMoney("0.234","F","$","F",0,2) = $.23

FormatMoney("0.234","F","$","T",0,2) = $0.23

Nota: il posizionamento del carattere prezzo è responsabile della posizione del carattere prezzo; tra il carattere prezzo e l'importo vengono aggiunti degli spazi. Questo valore viene ignorato se inferiore alla lunghezza della stringa del risultato.

FormatNumber(number): Questa funzione consente di formattare un campo numerico in cui il simbolo cancelletto (#) indica di visualizzare solo se esiste un valore e zero (0) indica di visualizzare sempre.

Esempi:

FormatNumber(123.45, "US$ #,###,###.00") = US$ 123.45

FormatNumber(123.45, "US$ 0,000,000.00") = US$ 0,000,123.45

FormatNumber(.45, "#,##0.00") = 0.45

FormatNumber(.45, "#,###.00") = 45

FormatNumber(7188302335, "(###) ###-####") = (718) 830-2335

FormatNumber(123.45, "00.00") = 23.45

FormatNumber(123.567, "###,##0.00") = 123.57

GetEnv(name): Restituisce il valore «name» (stringa) della variabile di ambiente.

Esempio

GetEnv("OS") = " Windows_NT".

Nota: la funzione utilizza le variabili di ambiente definite dall'utente

GS1AIData(«AIName», «AIData», «isLastAI»): regola i dati per GS1 AI.

GS1HRData(«AIName», «AIData», «isLastAI»): Adjusts data for GS1 Human readable.

GS1Norm(«AIName», «AIData», «isLastAI»): Adjusts data for GS1 Norm.

int (valore) restituisce il maggiore degli interi inferiore o uguali all'argomento *valore*.

Esempi:

int (-5.863) = -6

int (5.863) = 5

LmChar («integer») : Restituisce il carattere corrispondente al numero intero nella tabella ANSI. Questa tabella non dipende dal sistema delle impostazioni locali.
Il valore è compreso tra 0 e 255.

MaxiCodeData(«Mode»,«PostalCode»,«CountryCode»,«ClassOfService»,«TrackingNumber»,«UpsShipperNumber»,«JulianDate»,«ShipmentID»,«PackageNumber»,«TotalPackages»,«PackageWeight»,«AdressValidation»,«ShipToAdress»,«ShipToCity»,«ShipToState»): crea i dati Maxicode da «stringa» di input.

text (valore, formato) restituisce l'argomento *valore* scritto nel modo imposto dall'argomento *formato* che corrisponde al formato descritto durante la spiegazione della variabile Modulo.

Esempi:

text(012.345, "######,(###)") = 12,(345)

text(1012.345, "######,(###)") = 1012,(345)

trunc (valore) restituisce la parte intera dell'argomento *valore* .

Esempio:

trunc (123.45) = 123

UnicodetoDBCS(Dati, Tabella dei codici): La formula contiene 2 parametri – I dati da convertire, e la tabella dei codici usata. Il parametro della tabella dei codici può contenere uno dei valori seguenti:

Thai
Japanese
Chinese GBK
Korean
Chinese Big5
European eastern
Greek
Turkish
Hebrew
Arabic
Baltic
Vietnamese

Nota: La formula non distingue tra maiuscole e minuscole.

value (stringa) restituisce il valore numerico di una stringa di caratteri.

Esempio:

value("123") = 123

value("16:48:00")-value("12:00:00") uguale "16:48:00"-"12:00:00" uguale 0,00, il numero di serie equivale a 4 ore e 48 minuti.

Nota: generalmente non è necessario utilizzare la funzione value in una formula, perché l'applicazione converte automaticamente il testo in numeri se occorre.

ValueEx(«stringa»): converte «stringa» in un valore numerico.

VoiceCode(«stringa», «stringa», «stringa»): VoiceCode è un codice numerico a 4 cifre creato utilizzando il codice GTIN, il Lotto e la Data opzionale da un PTI.

Return value	stringa	Calcola un valore VoiceCode.

GTIN	stringa	Codice numerico GTIN a 14 cifre. Sono ammessi solo numeri. Se viene inserito un valore non numerico, sarà restituito un errore. A sinistra sono ammesse 14 cifre se la lunghezza dei dati è superiore a 14. A sinistra vengono aggiunti gli '0' nel caso in cui la lunghezza dei dati sia inferiore a 14.
LotNumber	stringa	Fino a 20 simboli alfanumerici. A sinistra sono ammessi 20 simboli se la lunghezza dei dati è superiore a 20.
Date	stringa	Esiste un parametro opzionale. 6 cifre che costituiscono la data nel formato AAMMGG. La formula restituisce un errore in caso di lunghezza errata, valore non numerico o data non corretta.

L'operazione è la seguente:

1. Calcolo del PlainText.

   a. Il PlainText è un codice GTIN a 14 cifre aggiunto al Codice lotto e alla Data (se presenti).

   b. Non comprende i prefissi o le parentesi dell'ID applicazione.

   c. Non esistono spazi tra i campi GTIN, Lotto e Data.

   d. Se la data è presente, viene rappresentata nel formato AAMMGG Date senza stringa e caratteri '/'.

2. Calcolo dell'hash CRC-16 ANSI dei byte ASCII in PlainText utilizzando il CRC-16 ANSI standard hash con polinominale di X16 + X15 + X2 + 1
   Vedere la funzione CRC16.
3. Calcolo del VoiceCode dall'hash utilizzando le 4 cifre meno importanti in formato decimale (mod. hash 10000).

   Esempio

   GTIN = (01) 10850510002011
   Lotto = (10) 46587443HG234
   PlainText = 10850510002011146587443HG234
   Hash CRC-16 = 26359
   VoiceCode = 6359
   Cifre grandi = 59
   Cifre piccole = 63

Esempi:

VoiceCode("10850510002011", "46587443HG234") = 6359
VoiceCode("65457886676767", "2", "100126") = 5836

Funzioni di conversione ATA 2000

Nota: Le impostazioni regionali del sistema Windows potrebbero influire sull'output di alcune formule.

Bin2Hex(«valore», «pad»)

Converte un valore binario in un valore di output esadecimale. Il parametro 'pad' viene usato per definire la lunghezza dell'intero output aggiungendo 0 caratteri a sinistra del risultato.

Esempio:
Bin2Hex("0010100111101001101",0) restituisce 14F4D
Bin2Hex("0010100111101001101",0) restituisce 00014F4D

CRC16_CCITT(«string»)

Calcola il checksum CRC16 CCITT.
Questo viene usato nella regolazione ATA2000 per il calcolo TOC per CRC.

Esempio:
CRC16_CCITT ("A") restituisce B915
CRC16_CCITT ("AB") restituisce 4B74
CRC16_CCITT ("ABC") restituisce F508

CRC32_CCITT(«string»)

Calcola il checksum CRC32 CCITT.
Questo viene usato nella regolazione ATA2000 per il calcolo TOC per CRC.

Esempio:
CRC32_CCITT ("A") restituisce D3D99E8B
CRC32_CCITT ("AB") restituisce 30694C07
CRC32_CCITT ("123456789") restituisce CBF43926

Dec2Bin(«valore», «pad»)

Converte un valore decimale in un valore di output binario. Il parametro ‘pad’ viene usato per definire la lunghezza dell'intero output aggiungendo 0 caratteri a sinistra del risultato.

Esempio:
Dec2Bin("5", 0) restituisce 101
Dec2Bin("5", 4) restituisce 0101

Dec2Hex(«valore», «pad»)

Converte un valore decimale in un valore di output esadecimale. Il parametro ‘pad’ viene usato per definire la lunghezza dell'intero output aggiungendo 0 caratteri a sinistra del risultato.

Esempio:
Dec2Hex(510, 0) restituisce 1FE
Dec2Hex(510, 4) restituisce 01FE

String2Hex(«valore», «pad»)

Converte im valore di stringa in un valore di output esadecimale. Il parametro ‘pad’ è usato per definire l'allineamento in Byte per l'output aggiungendo 0 caratteri a destra del risultato.
In RFTAG, i dati sono molto spesso codificati secondo il sistema esadecimale con un allineamento delle parole (2 Byte).

Esempio:
String2Hex("A", 0) restituisce 41 (**A** è il codice AINSI **65** nel sistema decimale che è **41** nel sistema esadecimale)
String2Hex("ABC", 2) restituisce 414243
String2Hex("ABC", 4) restituisce 41424300

Ecco un esempio di record RFID ATA2000 Birth:

String2Hex("MFR S0671*SEQ M37GXB92*PNO PQ7VZ4*PDT CONTROLLER*ICC 456789*UIC 2*DMF 20160701*UNT KG*HAZ UN0003*ESD 1*LOT 123456*CNT FR*PMLPML123456789", 2) restituisce
4D46522053303637312A534551204D333747584239322A504E4F20505137565A342A50445420434F4E54524F4C4C45522A49

String6BitsEncoding(«valore», «pad»)

Converte un valore di stringa in un valore di output con codifica ASCII a 6 bit.
Il parametro ‘pad’ viene usato per definire l'allineamento in Byte per l'output aggiungendo 0 caratteri a

destra del risultato.
In ATA2000 i dati possono essere codificati a 6 bit (invece che a 8 bit) per scopi di compressione.

Esempio:
String6BitsEncoding("11", 0) restituisce C710

VDAString6BitsEncoding(«stringa»)
Converte un valore di stringa in un valore di uscita con codifica ASCII a 6 bit secondo i requisiti dell'Associazione tedesca dell'industria automobilistica (VDA).
In ATA2000 i dati possono essere codificati in 6 bit (invece che in 8 bit) a scopo di compressione.
La tabella utilizzata per la conversione è disponibile **qui**. Per ulteriori dettagli, consultare le indicazioni specifiche

VDA5500 e VDA4994.

Esempio:
VDAString6BitsEncoding("11") restituisce C71860820820

Le funzioni sulla data e l'ora

La variabile data qui di seguito rappresenta la data e l'ora di sistema e non la variabile data definita dal programma.

Nota: Le impostazioni regionali del sistema Windows potrebbero influire sull'output di alcune formule.

BestBefore(date , dateFormat , offset , offsetUnit , changeMonth , outputFormat, localeID)
Consente di calcolare una data di scadenza in base al valore immesso tramite tastiera, non in base alla data corrente.

Valore restituito	stringa	Un valore data in formato stringa, calcolato per una data di base e un offset.
date	stringa	Valore data di base in formato stringa
dateFormat	stringa / numero	Formato del valore data di base. - I numeri codificano e formati predefiniti (per es. 1 dd/mm, 2 dd mmmm ecc. I valori possibili sono disponibili qui.

		- 0 o valori negativi indicano l'utilizzo del formato di data corrente - Il valore stringa codifica direttamente il formato (per es. "yyyy/mm/dd")
offset	numero	Numero di unità di data da aggiungere alla data di base.
offsetUnit	stringa / numero	Significato del parametro offset. - **1** o **"d"** o **"D"** significa giorno - **2** o **"m"** o **"M"** significa mese - **3** o **"y"** o **"Y"** significa anno
changeMonth	numero	Facoltativo (impostazione predefinita: True) - **0** significa **False** - **1** significa **True** Se la data calcolata non esiste nel mese (per es. 30 febbraio) **True** restituisce il primo giorno del mese successivo (il 1° marzo) **False** restituisce l'ultimo giorno del mese corrente (il 28 febbraio)
outputFormat	stringa / numero	Facoltativo (impostazione predefinita: uguale a dateFormat) Questo parametro specifica il formato del risultato della data calcolata. I valori possibili sono gli stessi di dateFormat.
localeID	numero	Facoltativo (impostazione predefinita: 0, saranno utilizzate le impostazioni locali del sistema). Questo parametro specifica le impostazioni locali del sistema utilizzate per visualizzare il valore della data.

Esempio:

BestBefore («date» , «dateFormat» , «offset» , «offsetUnit» , «changeMonth» , «outputFormat»)

BestBefore("14/06/2012" ,"dd/mm/yyyy","18","m", 1, "dd/mm/yyyy") will return 14/12/2013.

Esempio: Formula with a date data source.

Create a date variable name date0 with the date of the day: 26/06/2012 for this example with dd/mm/yy format.

BestBefore(date0 ,"dd/mm/yy","18","m", 1, "dd/mm/yyyy") will return 14/12/2013.

CFIA_Month(«data»):

Restituisce il nome del mese utilizzato dalla Canadian Food Inspection Agency (CFIA): GEN., FEB., MAR., APR., MAG., GIU., LUG., AGO., SET., OTT., NOV., DIC.

Esempio:

CFIA_Month ("01/03/2021") restituisce MAR.

CFIA_Month (today()) restituisce il nome del mese in base alla data odierna, se oggi è il 01/01/2021 - il valore restituito è GEN.
CFIA_Month ("01/06") restituisce GIU.

Nota: il formato dell'argomento "data" dipende dalle impostazioni di ambiente applicate al sistema.

DateOffset(date , offset, offsetUnit , changeMonth)

Valore restituito	data	Aggiunge un intervallo di data alla data di base specificata.
date	data	Valore data di base.
offset	numero	Numero di unità di data da aggiungere alla data di base.
offsetUnit	stringa / numero	Significato del parametro offset. - **1** o **"d"** o **"D"** significa giorno - **2** o **"m"** o **"M"** significa mese - **3** o **"y"** o **"Y"** significa anno
changeMonth	numero	Facoltativo (impostazione predefinita: True) - **0** significa **False** - **1** significa **True** Se la data calcolata non esiste nel mese (per es. 30 febbraio) **True** restituisce il primo giorno del mese successivo (il 1° marzo) **False** restituisce l'ultimo giorno del mese corrente (il 28 febbraio)

Esempio:
DateOffset("26/06/2012",1,"d",1) will return 27/06/2012
-OR-
DateOffset(today(),1,"D", 0)

Esempio: Formula with a date data source.
Create a date variable name Date0 with the date of the day. : 26/06/2012 for this example with dd/mm/yy format. You should ensure that date variable is in default system format (otherwise you can use DateValue() function to get the date value from the date variable).
DateOffset(Date0,1,"D", 0) will return 27/06/2012

DateValue(formattedDate , format)

Valore restituito	data	Restituisce il valore data del parametro *formattedDate.*
formattedDate	stringa	Una data in formato testo, da convertire in un valore data.
format	stringa	Facoltativo (impostazione predefinita: il formato di data del sistema) se specificato, la stringa di formato, ovvero "dd/mm/yy"

Esempio:

DateValue(22062012,"ddmmyyyy") will return 22/06/2012.

day (*data*) restituisce il giorno del mese dell'argomento *data*.

FiscalDate(«fiscalStartDate», «outputFormat»)

Consente di calcolare la data d'inizio dell'anno fiscale (yyyy.mm.dd).

Esempi (se la data corrente è 2009.11.24):

FiscalDate("2009.01.01", 1) = 09

FiscalDate("2009.01.01", "yyyy") = 2009

FiscalDate("2009.01.01", 3) = 47

FiscalDate("2009.01.01", "day") = 328

hour (*data*) restituisce l'ora dell'argomento *data*.

minute (*data*) restituisce i minuti dell'argomento *data*.

month (*data*) restituisce il mese dell'argomento *data*.

now ()restituisce la data e l'ora correnti.

second (): restituisce il secondo dell'argomento data.

shiftcode(items)

Lo Shift Code è utilizzato come sorgente dati per un campo dell'etichetta che si deve modificare in base all'ora corrente. Un turno (shift) rappresenta un periodo di tempo definito e denominato.

valore restituito	stringa	Restituisce il valore Shift Code corrente.

item (elementi)	stringa	Elementi shift code nel seguente formato: OraInizio:MinInizio-OraFine:MinFine-valore|…| OraInizio:MinInizio-OraFine:MinFine-valore

Ad esempio, i turni si potrebbero definire nel seguente modo:

- dalle 7:00 alle 15:00 = Giorno
- dalle 15:00 alle 23:00 = Sera
- dalle 23:00 alle 7:00 = Notte

Nota:
1. Per definire i valori temporali viene utilizzato un orologio di 24 ore. 0:00 è mezzanotte; 12:00 è mezzogiorno.
2. In caso di sovrapposizioni di tempo immesso, viene restituito il primo valore accettabile.
3. Se non esiste un turno per il periodo corrente, viene restituita una stringa vuota.

Esempio (se sono le 16:00):
shiftcode("7:00-15:00-Giorno|15:00-23:00-Sera|23:00-7:00-Notte") ="Sera"

SpecificDateFormat("[date format]", "+/[offset data][date interval]")

La funzione SpecificDateFormat consente di personalizzare la data impostando uno specifico intervallo di date e utilizzare la data personalizzata in un'espressione con formula. La funzione SpecificDateFormat determina la data attuale sulla base del clock di sistema, utilizzando il formato di data predefinito o un formato diverso a scelta. Se utilizzata in un'espressione, la funzione SpecificDateFormat deve essere trascritta utilizzando i seguenti parametri: SpecificDateFormat ("[date format]", "+/ [offset data][date interval]")

Per esempio: SpecificDateFormat("mmmm","+2M")

- [date format] Specifica il formato di data da utilizzare. Questo parametro deve essere riportato tra parentesi. La ***tabella*** elenca i formati di data validi.

Impostazione data	Output	Descrizione
D e d	1	data del mese (senza zero iniziale)
DD	01	data del mese a 2 caratteri (con zero iniziale)
dd	1	data del mese a 2 caratteri (con spazio iniziale)
DDD	224	giorni dall'inizio dell'anno (con zero iniziale)

ddd	224	giorni dall'inizio dell'anno (con spazio iniziale)
DDDDD e ddddd	33482	giorni dal 1/1/1900 (da 5 a 9 caratteri disponibili)
W e w	1	giorno della settimana (Domenica=1, Sabato=7)
WW	34	settimane dall'inizio dell'anno (con zero iniziale)
ww	34	settimane dall'inizio dell'anno (con spazio iniziale)
WWW e www	783	settimane dal 1/1/1900 (ultime 3 cifre)
WWWW e wwww	4783	settimane dal 1/1/1900 (da 4 a 9 caratteri disponibili)
WWWWWWWWW	000004783	settimane dal 1/1/1900 (con zero iniziale)
wwwwwwwww	4783	settimane da 1/1/1900 (con spazio iniziale)
M e m	9	numero del mese (senza zero iniziale)
MM	09	numero del mese a 2 caratteri (con zero iniziale)
mm	9	numero del mese a 2 caratteri (con spazio iniziale)
MMM	009	numero del mese a 3 caratteri (con zero iniziale)
mmm	9	numero del mese a 3 caratteri (con spazio iniziale)
MMMM e mmmm	1101	mesi dal 1/1/1900
MMMMM	01101	mesi a 5 caratteri dal 1900 (con zero iniziale)
mmmmm	1101	mesi a 5 caratteri dal 1900 (con spazio iniziale) (da 5 a 9 caratteri disponibili)
MMMMMMMMM	000001101	mesi a 9 caratteri dal 1900 (con zero iniziale)
mmmmmmmmm	1101	mesi a 9 caratteri dal 1900 (con spazio iniziale)
YY e yy	91	anno a 2 caratteri
Y e y	1991	anno per intero
YYY e yyy	991	ultime 3 caratteri dell'anno (da 3 a 9 caratteri disponibili)
YYYYYYYYY	000001991	anno per intero (con zero iniziale)
yyyyyyyyy	1991	anno per intero (con spazio iniziale)

- +/[offset data] Indica il valore di offset della data. L'intervallo deve essere preceduto dal segno più (+) o dal segno meno (), a seconda che si voglia aggiungere o sottrarre dalla data attuale.
- [date interval] L'intervallo date deve contenere d per giorni, w per settimane, m per mesi o y per anni.

La tabella seguente illustra alcuni esempi della funzione SpecificDateFormat utilizzata in espressioni con formati diversi. (Nota: in questi esempi si suppone che la data attuale sia il 29 luglio 2013)

Espressione	Risultato
SpecificDateFormat("mm dd yy" , "+1y")	7 29 14 (1 anno dalla data attuale)
SpecificDateFormat("ww/m/yyyy" , "30w)	1/12/2012 (30 settimane prima della data attuale)
SpecificDateFormat("ddddd / wwww / mmmm" , "")	41484/5928/1363 (giorni, settimane e mesi dal 1900)

TimeOffset(FormatDate(Now(), "mm/dd/yyyy hh:nn:ss"),"time interval +/offset time")

La funzione TimeOffset consente di personalizzare l'ora impostando uno specifico intervallo di ore e utilizzare l'ora personalizzata in un'espressione con formula. La funzione TimeOffset determina l'ora attuale sulla base del clock di sistema, utilizzando il formato di ora predefinito.

Se utilizzata in un'espressione, la funzione TimeOffset deve essere trascritta utilizzando i seguenti parametri:

TimeOffset(FormatDate(Now(), "mm/dd/yyyy hh:nn:ss"),"time interval +/ offset time")

Per esempio: TimeOffset(FormatDate(Now(), "mm/dd/yyyy hh:nn:ss"),"h+5")

- È possibile utilizzare solo il formato predefinito "mm/dd/yyyy hh:nn:ss". Per immettere un formato diverso, è necessario includere la funzione TimeOffset nella funzione DateValue, quindi includerla nella funzione FormatDate, dove è possibile specificare un altro formato di ora.
- "+/offset time" Indica il valore di offset dell'ora. L'intervallo deve essere preceduto dal segno più (+) o dal segno meno (), a seconda che si voglia aggiungere o sottrarre dall'ora attuale.
- "time interval" L'intervallo di tempo è indicato da s per secondi, m per minuti o h per ore.

La tabella seguente illustra alcuni esempi della funzione TimeOffset utilizzata in espressioni con formati diversi. (Nota: in questi esempi si suppone che l'ora attuale sia 16:12:10 del 30 luglio 2013)

Espressione	Risultato
TimeOffset(FormatDate(Now(), "mm/dd/yyyy hh:nn:ss"),"h+5")	07/30/2013 21:12:10 (5 ore dall'ora attuale)
FormatDate(DateValue(TimeOffset(FormatDate(Now(), "mm/dd/yyyy hh:nn:ss"), "n+10"), "mm/dd/yyyy hh:nn:ss"), "hhnn")	1622 (10 minuti dall'ora attuale)

today ()restituisce la data corrente.

Week *(data)* restituisce il numero della settimana dell'argomento *data*.

weekday *(data)* restituisce il giorno della settimana dell'argomento *data*.

Nota : domenica è considerata come il primo giorno della settimana.

Esempio:
ossia "martedì 20 novembre 1999", la data corrente, si ha weekday(now()) = 3

WeekISO8601(Date, DateFormat)

Consente di creare una formula che supporta la norma ISO 8601. ISO 8601 è uno standard internazionale che riguarda la conversione di data e ora.

Valore restituito	data	Restituisce la settimana relativa alla data specificata
Date	stringa	Una data in formato testo, da convertire in un valore data.
DateFormat	stringa	Formato del valore data di base. - I numeri codificano formati predefiniti. (ad es. 1 gg/mm/aa, 2 gg/mm/aaaa ecc. - 0 o valori negativi indicano l'utilizzo del formato di data del sistema corrente - Il valore stringa codifica il formato direttamente. (ad es. "aaaa/mm/gg")

Esempi :

WeekISO8601(" 03/01/2010 ",0) = 53
WeekISO8601(" 02/01/2011 "," dd/mm/yyyy ") = 52
WeekISO8601(" 01/01/2011 ", " dd/mm/yyyy ") = 52

year (data) restituisce l'anno dell'argomento data.

Esempio:
minute (now ())restituisce i minuti dell'ora corrente.
year (today ()) restituisce l'anno della data corrente.

Le funzioni logiche

Le funzioni logiche consentono di controllare se si sono verificate o no una o più condizioni.

Nota: Le impostazioni regionali del sistema Windows potrebbero influire sull'output di alcune formule.

Nota: TRUE equivale a 1 e FALSE equivale a 0.

and (*espr_1*, *espr_2*) restituisce TRUE se i due argomenti sono veri, FALSE se almeno un argomento è falso. Gli argomenti devono essere valutati a valori logici.

Esempio :
and(exact("stringa","stringa"),exact("stringa","stringa")) = 0
and(exact("stringa"," stringa"),exact("stringa"," stringa")) = 1

exact (*stringa_1*, *stringa_2*) restituisce TRUE se le due stringhe sono identiche, altrimenti FALSE. Questa funzione rispetta le maiuscole e le minuscole.

Esempio :
exact("software","software") = 1
exact("software","software") = 0

if (*espr*, *Val_se_vero*, *Val_se_falso*) restituisce l'argomento *Val_se_vero* sei *Espr* è vera e l'argomento *Val_se_falso* se *Espr* è falsa.

Esempio :
if(exact("stringa", "stringa"), "vero", "falso")) = falso
if(exact("stringa", "stringa"), "vero", "falso")) = vero

not (*logico*) restituisce l'opposto dell'argomento *logico*.

Esempio : not(exact("stringa", "stringa")) = 1
not(exact("stringa", "stringa")) = 0
not(False) = 1 o not(O) = 1
not(True) = 0 o not(1) =0
not(1+1=2) = 0

o (*espr_1*, *espr_2*) restituisce TRUE se uno dei due argomenti è vero e FALSE se i due argomenti sono falsi. Gli argomenti devono essere valutati a valori logici.

Esempio :

or(exact("stringa", "stringa"),exact("stringa ", "stringa")) = 0

or(exact("stringa", "stringa"),exact("stringa ", "stringa")) = 1

or(true,true) = 1 o or(1,1) = 1

or(true,false) = 1 o or(1,0) = 1

or(false,false)= 0 o or(0,0) = 0

Le funzioni matematiche

Nota: Le impostazioni regionali del sistema Windows potrebbero influire sull'output di alcune formule.

Abs*(data)*: Questa funzione dà come risultato il valore assoluto (positivo) dei dati. Consente l'uso di lettere dopo i numeri.

Esempi

Abs(-5) = 5

Abs(5) = 5

base10tobaseX(*stringa_1,stringa_2*) converte *stringa_2* di base 10 in base *stringa_1*

Esempi

Se il campo chiamato Base 16 contiene la stringa "0123456789ABCDEF"

BASE10TOBASEX(Base16, 12) dà C

BASE10TOBASEX(Base16,10) dà A

BASE10TOBASEX("012345","9") dà 13

baseXtobase10(*stringa_1,stringa_2*) converte *stringa_2* di base *stringa_1* in base 10.

Esempi

Se il campo chiamato Base 16 contiene la stringa "0123456789ABCDEF"

BASEXTOBASE10(Base16, "E") dà 14

BASEXTOBASE10(Base16,10) dà A

BASEXTOBASE10("012345","9") dà 13

Ceil*(data)*: Questa funzione arrotonda i dati al numero intero successivo. Consente l'uso di lettere dopo i numeri.

Esempi
Ceil(3.234) = 4
Ceil(7.328) = 8

Decimals*(data1, data2)*: Questa funzione utilizza data2 posizioni decimali in data1. Consente l'uso di lettere dopo i numeri.

Esempi
Decimals(4, 2) = 4.00
Decimals(3.524, 1) = 3.5

eval_add(«string»,«string»): restituisce il totale dei parametri.

Esempi
eval_add(5,5)=10

eval_div(«string»,«string»): restituisce la divisione dei parametri.

Esempi
eval_div(20,2)=10

eval_mult(«string»,«string»): restituisce la moltiplicazione dei parametri.

Esempi
eval_mult(5,2)=10

eval_sub(«string»,«string»): restituisce la differenza dei parametri.

Esempi
eval_sub(20,10)=10

Floor*(data)*: Questa funzione arrotonda i dati al numero intero successivo. Consente l'uso di lettere dopo i numeri.

Esempi
Floor(3.234)= 3
Floor(7.328)= 7

hex(*val_1,val_2*) converte il numero decimale *val_1* in formato esadecimale con larghezza totale *val_*

Esempi
hex(2,8) = 00000002

int (*valore*) restituisce il maggiore degli interi inferiore o uguali all'argomento *valore*.

Esempi :
int (-5.863) = -6
int (5.863) = 5

max*(data1, data2)*: Questa funzione visualizza il valore più alto. Consente l'uso di lettere dopo i numeri.

Esempi
Max(5, 12) = 12

min*(data1, data2)*: Questa funzione visualizza il valore più basso. Consente l'uso di lettere dopo i numeri.

Esempi
Min(5, 12) = 5

mod (*val_1, val_2*) restituisce il resto della divisione dell'argomento *val_1* per l'argomento *val_2*. Il risultato è dello stesso segno del divisore.

Esempi:
mod (7,2) = 1
mod (-7,2) = 1
mod (7,-2) = -1
mod (-7,-2) = -1

quotient (*val_1, val_2*) restituisce l'intero risultato della divisione dell'argomento *val_1* per l'argomento *val_2*.

Esempi:
quotient (10,2) = 5

round (*val_1, val_2*) restituisce l'argomento *val_1* arrotondato al numero di cifre indicato dall'argomento *val_2*.

Se *val_2* è maggiore di 0, *val_1* è arrotondato al numero di decimali indicato.

Se *val_2* è uguale a 0, *val_1* è arrotondato al numero intero più vicino.

Se *val_2* è minore di 0, *val_1* è arrotondato a sinistra della virgola.

Esempi:
round (4.25,1) = 4.3
round (1.449, 1) = 1.4
round (42.6,-1) = 40

Le funzioni sui testi

Una stringa di caratteri è assimilabile a una tabella in cui ogni casella contiene un carattere. È definita dalla sua lunghezza (numero totale di caratteri della stringa, spazi compresi). La posizione di un carattere in una stringa corrisponde al suo posto nella tabella ossia il primo carattere ha la posizione 1. Esempio: Position 3 corrisponde al terzo carattere della stringa.

Nota: Le impostazioni regionali del sistema Windows potrebbero influire sull'output di alcune formule.

cyclebasex *( )* consente il conteggio in qualsiasi sistema di conteggio di un database. Il sistema di numerazione dev'essere definito all'interno dell'espressione collegata. Devono anche essere specificati il valore di partenza, il valore di ogni incremento e il numero di copie di ogni numero. Tutti questi valori possono essere collegati ad altri campi dell'etichetta ma i nomi dei campi non devono apparire tra virgolette.

Esempio:
Se il campo chiamato Base16 contiene la stringa di caratteri 0123456789ABCDEF, allora
cyclebasex(base16, « 8 », 1 ,1) = 8,9,A,B
cyclebasex(base16, « F », -1,1) = F,E,D,C,B,A 9,8,7
cyclebasex(base16, « B0 », 1,1) = B0, B1, B2
cyclebasex(«012345 », « 4 »,1,2) = 4,4,5,5,10,10,11,11

cyclechar *()* crea una serie personalizzata di caratteri per un ciclo completo.

Esempio:

cyclechar(« A », « C ») = A B C A B C A B C

cyclechar(« A », « C », 1,2) = A A B B C C A A B B

cyclenumber ( ) consente di configurare la vostra sequenza di numeri, in opposizione col la progressione normale delle cifre o delle lettere (0,1,2… o A,B,C).

Esempio:

cyclenumber(1,3) produrrà delle etichette sequenziate 1 2 3 1 2 3 1 2 3...

cyclenumber(1,3,1,2) produrrà delle etichette sequenziate 1 1 2 2 3 3 1 1 2 2 3 3...

cyclestring ( ) consente di creare un gruppo di parole o caratteri passando da un ciclo completo come un campo d'incremento. La stringa intera dev'essere racchiusa tra virgolette (« ») e ogni parola o gruppo di caratteri dev'essere separato dagli altri da un punto e virgola (;)

Esempio:

cyclestring(« Mon ; Tue ; Wed ; Thu ; Fri ; Sat ; Sun ») = Mon Tue Wed Thu Fri Sat Sun

L'esempio seguente è per la stampa di etichette di reparto utilizzando tutte le lettere dell'alfabeto eccetto O e I.

cyclestring(« A;B;C;D;E;F;G;H;J;K;L;M;N;P;Q;R;ST;U;V;W;X;Y;Z »)

exact (*stringa_1, stringa_2*) restituisce TRUE se le due stringhe sono identiche, altrimenti FALSE.

Esempio :

exact("software","software") = 1

exact("software","software") = 0

Extract(«string», «sep», «pos») sposta di nuovo la sotto stringa dalla stringa carattere «string» nella posizione «pos» specificata contenente dati separati dalla stringa «sep».

Esempio:

Extract("1;2;3;4", ";", 3) = 3

find (*stringa, chiave, partenza*) restituisce la posizione della prima occorrenza dell'argomento *chiave* nell'argomento *stringa*. La ricerca, nell'argomento *stringa*, inizia a partire dalla posizione data dall'argomento *partenza* (*partenza* >= 1). La funzione restituisce zero se non si trova nessuna occorrenza dell'argomento *chiave*. Si distingue tra maiuscole e minuscole.

Esempio :

find("Peter McPeepert","P",1) = 1

find("Peter McPeepert","P",1) = 12

left (*stringa*, *nb_car*) restituisce la stringa di caratteri estratta dall'argomento *stringa*. Questa stringa inizia alla posizione 1 dell'argomento *stringa* e ha una lunghezza uguale all'argomento *nb_car*.

Esempio :

left("Peter McPeepert",1) = P

left("Peter McPeepert ",5) = Peter

len (*stringa*) restituisce la lunghezza dell'argomento *stringa*. Gli spazi sono contati come caratteri.

Esempio :

len("Parigi, New York") = 15

len("") = 0

len(" ") = 1

lower (*stringa*) converte tutte le lettere maiuscole di una stringa di testo in lettere minuscole.

Esempio :

lower("Parigi, New York") = parigi, new york

LTrim*(«string»)*: Questa funzione rimuove automaticamente gli spazi iniziali o finali dei dati sul lato sinistro.

Esempio:

LTrim(" No."): No

mid (*stringa*, *partenza*, *nb_car*) restituisce la stringa di caratteri estratta dall'argomento *stringa*. Questa stringa inizia dalla posizione corrispondente al valore dell'argomento *partenza* (*partenza* >=1) e ha una lunghezza uguale all'argomento *nb_car*.

Esempio :

mid("Parigi, New York",8,8) = New York

Output (variabile) : restituisce il valore formattato della variabile.

Esempio :

ossia il Contatore0 il cui prefisso è "numero :". Il risultato della formula

compteur0&output(compteur0) è :
1 numero : 001

Nota : la formattazione della variabile è definito nella scheda **Output.**

pad () aggiunge dei caratteri del lato sinistro del campo per dare una lunghezza predefinita a tutta la voce. È possibile selezionare qualsiasi carattere come carattere di riempimento.

Esempio:
Se un campo chiamato GREETING visualizza un valore HELLO, allora
pad(GREETING,8) = 000HELLO
pad(5,3) = 005
pad(”Nine”,6,a) = aaNine

replace (*stringa*, *partenza*, *nb_car*, *nvl_stringa*) restituisce l'argomento *stringa* trasformato. Un numero (uguale all'argomento *nb_car*) di caratteri dalla posizione definita dall'argomento *partenza* è stato sostituito dall'argomento *nvl_stringa*.

Esempio :
replace("Parigi, New York",8,8,"Singapore") = Parigi, Singapore

ReplaceString(«string», «old_string», «new_string») sostituisce tutte le occorrenze di una «old_string» specificate nella stringa carattere «string» con un'altra «new_string» specificata.

Esempio :
ReplaceString("abc12def12", "12", "") = abcdef

rept (*stringa*, *nb_car*) restituisce la stringa di caratteri dove l'argomento *stringa* è ripetuto un numero di volte corrispondente all'argomento *nb_car*.

Esempio :
rept("Ah Parigi ! ",2) = Ah Parigi ! Ah Parigi !

right (*stringa*, *nb_car*) restituisce la stringa di caratteri composta degli ultimi caratteri dell'argomento *stringa* e avente una lunghezza uguale all'argomento *nb_car*.

Esempio :
right("Buono d'ordine",8) = comando

RTrim *(«string»)*: Questa funzione rimuove automaticamente gli spazi iniziali o finali dei dati sul lato destro.

Esempio:
RTrim("Part ") :Part

search (*stringa, chiave, partenza*) restituisce la posizione della prima occorrenza dell'argomento *chiave* nell'argomento *stringa*. La ricerca inizia dalla posizione definita dall'argomento *partenza* (*partenza* >= 1). La funzione restituisce zero se non si trova nessuna occorrenza dell'argomento *chiave*.

Esempio :
search("Buono d'ordine","comando,1) = 8
search("Buono d'ordine","c",1) = 8

Sort(«elenco», «separatore», «direzione»): Restituisce un elenco ordinato di elementi. L'ordinamento avviene tramite confronto di stringhe. L'elenco è un argomento stringa contenente più elementi con un separatore come la virgola o il punto. La direzione è un numero intero opzionale che specifica la direzione dell'ordinamento, 0 = ascendente, 1 = discendente. Se la direzione non è specificata, il valore predefinito è ascendente.

Esempi:
Sort("3;5;6;1", ";") = "1;3;5;6"
Sort("one;two;three", ";", 1) =“two;three;one”

StrAfter*(«data»,«start after», «length»)*: Questa funzione dà come risultato una stringa lunga esattamente un certo numero di caratteri dopo un inizio specificato dopo il carattere.

Esempi:
StrAfter("1234-5678", '-', 3)= Takes the 3 characters after the dash (567)
StrAfter("1234-5678", '-')= Takes all characters after the dash (5678)

StrBefore*(«data»,«start before», «length»)*: Questa funzione dà come risultato una stringa lunga esattamente un certo numero di caratteri prima di un inizio specificato prima del carattere.

Esempi:
StrBefore("1234-5678", '-', 2)= Takes the 2 characters immediately before the dash (34)
StrBefore("1234-5678", '-')= Takes all characters before the dash (1234)

SuppressBlankRows («stringa»): restituisce una stringa con righe vuote saltate. Consente di creare un oggetto e posizionare all'interno i campi che si desidera e sopprimere quelli che sono vuoti.

Esempio:

SuppressBlankRows({Var0} & char(10) & {Var1} & char(10) & {Var2}) (Var0, Var1 e Var2 sono variabili, char(10) è "\n" simboli che stanno per una nuova riga)

Variabili/Valore	Formula	mostra
Var0 = "Doris" Var1 = "Bull Run Ranch" Var2 = "Aurora"	*SuppressBlankRows({Var0} & char(10) & {Var1} & char(10) & {Var2})*	Doris Bull Run Ranch Aurora
Var0 = "Craig" Var1 = "" Var2 = "Chicago"	*SuppressBlankRows({Var0} & char(10) & {Var1} & char(10) & {Var2})*	Craig Chicago
Var0 = "Craig" Var1 = "" Var2 = "Chicago"	*{Var0} & char(10) & {Var1} & char(10) & {Var2}*	Craig Chicago

SuppressEmpty(«elenco», «separatore»): rimuove le voci vuote da un elenco di elementi. L'elenco è un argomento stringa contenente più elementi con un separatore come la virgola o il punto.

Esempio:

SuppressEmpty("3;;;1", ";") = "3;1"

trim(stringa) restituisce l'argomento *stringa* trasformato. Tutti gli spazi all'inizio o alla fine della stringa vengono eliminati. Il numero di spazi compresi tra le due parole è ridotto a uno.

Esempio :

trim(" Buono d'ordine ") = Buono d'ordine

trimall(*stringa*) restituisce l'argomento *stringa* trasformato. Tutti gli spazi incontrati sono eliminati.

Esempio :

trimall("Parigi / New York / Roma") = Parigi/NewYork/Roma

Unique(«elenco», «separatore», «comparatore»): Rimuove le voci duplicate da un elenco di elementi. L'elenco è un argomento stringa contenente più elementi con un separatore come la virgola o il punto. Il comparatore opzionale indica se il confronto è sensibile alle maiuscole o meno, 0 = sensibile alle maiuscole, 1 = ignora le maiuscole. Se il comparatore non viene specificato, l'impostazione predefinita è sensibile alle maiuscole.

Esempi:
Unique("3;5;3;1", ";") = "3;5;1"
Unique("one;ONE;two", ";", 1) = “one;two”

upper (*stringa*) restituisce l'argomento *stringa* convertito in maiuscole.

Esempio :
upper("Buono d'ordine") = BUONO D'ORDINE

ztrim *( )* elimina tutti i zeri dalla sinistra dei soli campi numerici.

Esempio:
Se un campo chiamato WEIGHT visualizza un valore 000200, allora
ztrim(weight) = 200

Definizione delle proprietà dell'origine dati Formula

Comando: **Origine dati > Formula > Proprietà...**

1. Immettere la formula direttamente nella casella editing.
   -o-
   Selezionate con il mouse gli elementi desiderati quindi fate clic su **Inserisci**.

2. Fare clic su **OK**.

Suggerimento: È possibile inserire un elemento selezionandolo con un doppio clic.

Nota: Se il nome di una variabile utilizzata nella formula contiene uno dei seguenti caratteri &+-*/<>=^%,!\", deve essere racchiusa tra parentesi {}.

Note : Anteprima dinamica che rappresenta il risultato del calcolo della formula corrente, compresa la formattazione definita nella pagina Output. In caso di errore, l'anteprima viene visualizzata in rosso. Se il valore ottenuto è troncato, è necessario modificare la lunghezza massima specificata nella scheda **Output.**

Esercitazione : Calcolo del ”Modulo” specifico

Il vostro software integra delle funzioni di calcolo di modulo per l’integrazione di un carattere di controllo nei dati codice a barre.

Tuttavia può accadere che il calcolo del carattere di controllo sia specifico e che, di conseguenza, le funzioni integrate nel software non rispondano alle vostre esigenze.

Allora spetta a voi creare la funzione. Vedremo in questa sequenza che è perfettamente possibile realizzare qualsiasi calcolo di carattere di controllo grazie alle formule.

L’esempio presentato qui mostra il calcolo di un carattere di controllo per un codice a barre 2/5 interleaved.

Metodo di calcolo per un carattere di controllo

Il metodo consiste nel moltiplicare il primo carattere del dato per 1, il secondo per 2, il terzo per 1, ecc.

- Aprite l’etichetta chiamata CHECK01.LAB POSTA NELLA CARTELLA TUTORIAL.

  Il calcolo del carattere di controllo dev’essere eseguito sul dato LOTNUMBER.

Per calcolare il peso

1. Fate clic sul ramo Formula dalla scheda Origine dati e aggiungete una Formula.
2. Immettete la formula seguente:

   mid(LOT NUMBER,1,1)&
   mid(LOT NUMBER,2,1)*2 &
   mid(LOT NUMBER,3,1)&
   mid(LOT NUMBER,4,1)*2 &
   mid(LOT NUMBER,5,1)

   Il risultato è 2120103; infatti abbiamo:
   2*1 = 2
   6*2 = 12
   0*1 = 0
   5*2 = 10

3*1 = 3

La concatenazione dei risultati dà 2120103.

3. Denominate la formula WEIGHTED.

Per aggiungere il risultato del calcolo del peso:

La fase seguente consiste nel sommare le cifre risultanti dalla formula precedente.

1. Create una seconda Formula che chiamerete SUM.
2. All'interno della zona di immissione, inserite l'espressione seguente:

```
mid(WEIGHTED,1,1)+mid(WEIGHTED,2,1)+
mid(WEIGHTED,3,1)+mid(WEIGHTED,4,1)+
mid(WEIGHTED,5,1)+mid(WEIGHTED,6,1)+
mid(WEIGHTED,7,1)
```

Il risultato è 9; infatti abbiamo:
2+1+2+0+1+0+3= 9.

Per calcolare il carattere di controllo:

È con l'aiuto del risultato precedente che calcoleremo il valore del carattere di controllo.

1. Create una terza Formula che chiamerete CHECK DIGIT.
2. All'interno della zona di immissione, inserite l'espressione seguente:

```
if((SUM % 10)>0,10-SUM % 10,0)
```

Il risultato è 1; infatti abbiamo:
sum % 10 = 9 (% = modulo).

Dato che 9 è superiore a 0, effettuiamo la sottrazione:
10- 9 = 1.

Nota: Se il risultato fosse stato uguale a 0, il carattere di controllo avrebbe avuto come valore 0.

Per calcolare il dato da codificare:

La creazione del codice a barre deve comprendere il dato da codificare, ossia il valore della variabile LOT NUMBER concatenato al valore del carattere di controllo (CHECKDIGIT).

1. Create una quarta Formula che chiamerete DATA.
2. All'interno della zona di immissione, inserite l'espressione seguente:

   LOT NUMBER & CHECK DIGIT.

Per creare il codice a barre:

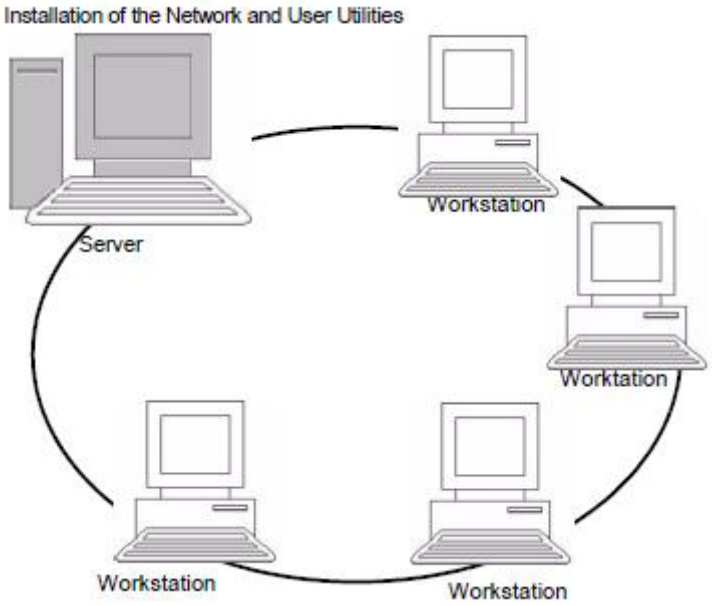
1. Fate clic sullo strumento Codice a barre posto nella barra degli strumenti di creazione, quindi selezionate la formula DATA e trascinatela nell'etichetta.
2. Selezionate il codice a barre e, con l'aiuto del menu di scelta rapida, scegliete Proprietà dell'oggetto.
3. L'etichetta è ora terminata. Troverete l'etichetta modello in TUTORIAL\CHECK.TPL.

Installazione di Network

Descrizione funzionale

I pacchetti di rete (multi-utente) consentono di controllare tramite la rete l'accesso alle licenze del software di etichettatura. Questa utility consente l'accesso al software di creazione delle etichette da parte di più utenti, nello stesso momento e da qualsiasi punto della rete.

Per utilizzare una versione di rete/multiutente del software di etichettatura, è necessario installare **Network License Manager** sul server o su una postazione che fungerà da server, quindi installare il software di etichettatura su ogni workstation.



Procedura d'installazione

Operazioni preliminari

Prima di procedere all'installazione del software, l'amministratore di rete dovrà definire la struttura della rete per il gruppo di utenti, in particolare:

- Definire i server delle licenze su cui saranno installati **Network License Manager** e la chiave hardware (dongle).
- Definire le workstation, ossia le postazioni client che utilizzeranno il software di etichettatura

Descrizione di Network Manager

Network License Manager consente di utilizzare la configurazione di rete del software di etichettatura.

Network Manager include:

- Network License Manager (License Service).
- **Network Settings Wizard**: Network Settings Wizard è una procedura guidata per configurare la rete.
- **User Manager**: User Manager è installato con **Network License Manager** per consentire di definire i diritti di accesso al software di etichettatura in un ambiente di rete.

Installazione di Network License Manager

Prima di installare il software di etichettatura su tutte le workstation che lo utilizzeranno, è necessario installare l'utility **License Service** sul server per configurare la rete.

Per installare **Network License Manager** sul server.

1. Inserire il DVD di installazione nell'apposita unità.
   Verrà visualizzata la finestra di installazione.

   Se il DVD non si avvia automaticamente:
   Scegliere **Start > Esegui**, quindi digitare la lettera dell'unità DVD, seguita dal file index.hta (per esempio, D:\index.hta).
2. Selezionare **Network License Manager**, che include **License Service** e **User Manager**. Fare clic sul pulsante **Installa**.
3. Seguire le istruzioni visualizzate sullo schermo.

4. Per definire le impostazioni della configurazione di rete, avviare sul server **Network Settings Wizard**. Per impostazione predefinita, se non si apportano modifiche alla configurazione, ciascuna workstation avrà le proprie impostazioni.

Configurazione

Tutti gli strumenti necessari per configurare la versione di rete sono disponibili dalla **Network toolbar**, alla quale è possibile accedere dalla barra delle applicazioni di Windows (Systray).

Dalla barra delle applicazioni di Windows, fare doppio clic sull'icona per visualizzare la **Network toolbar**.

Network Settings Wizard consente di definire le impostazioni per versione di rete utilizzata.

1. Per avviare **Network Settings Wizard**, fare clic sull'icona .
2. Selezionare una modalità di impostazione: Generica, Per utente o Per stazione.

    - **Generica**: tutti gli utenti utilizzeranno le stesse impostazioni per tutte le workstation. (user.ini).
    - **Per utente**: ogni utente potrà accedere alle proprie impostazioni su qualsiasi workstation. (user name.ini).
    - **Per stazione**: ogni workstation ha le proprie impostazioni (station.ini)

3. Indicare dove si desidera memorizzare queste impostazioni. Se si desidera condividere queste impostazioni tra le diverse workstation, indicare un percorso di rete accessibile a tutte le workstation (per esempio:TKDongle).
4. Indicare dove si desidera memorizzare i dati condivisi (variabili, elenchi, registro di stampa, ecc...).
   Accertarsi che tutti gli utenti abbiano accesso a queste cartelle.

Per configurare User Manager

La definizione dei diritti di accesso di rete per tutti gli utenti del software di etichettatura va effettuata durante l'installazione (consultare la guida in linea di User Manager). Fare clic sull'icona **User Manager** nella barra degli strumenti di **Rete**.

Avvio di License Service

Prima di installare il software etichettatura su tutte le workstation, è necessario verificare che **License Service** sia in esecuzione.

Il License Service è installato come servizio. Questo servizio, denominato **TkxWebLicenseServer**, viene abilitato automaticamente all'avvio del server.

Per avviare Service Controller

Fare clic sull'icona  visualizzata sulla barra degli strumenti di Rete

-oppurefare-

doppio sul file TkxWebLicenseServerController.exe.

-oppurefare-

Fare clic con il tasto destro del mouse sull'icona della **barra di Rete** dalla barra delle applicazioni di Windows e selezionare **Controller del License Service**.

Installazione del software sulle workstation

Il software di etichettatura dovrà essere installato su tutte le workstation che lo utilizzeranno.

Per installare il software su una workstation:

1. Inserire il DVD di installazione nell'apposita unità.
   Verrà visualizzata la finestra di installazione.

   Se il DVD non si avvia automaticamente: Scegliere **Start > Esegui**, quindi digitare la lettera dell'unità DVD, seguita da INDEX.HTA (per esempio, D:\index.hta).
2. Scegliere il prodotto da installare, quindi fare clic sul pulsante Installa e seguire le istruzioni visualizzate sullo schermo.
3. Avviare il software di etichettatura. Viene visualizzato il License Manager. Fare clic su **Prova** per avviare il software. Dal menu Strumenti, selezionare **Amministrazione rete**.
   - oppure -

Dal menu Start di Windows, selezionare il collegamento **Amministrazione di rete** nel gruppo software di etichettatura.

4. Attivare **Utilizza la licenza di rete**.
5. Selezionare il **Tipo di licenza** di rete.

   - La **licenza per cartelle condivise** utilizza la funzione di condivisione dei file di Windows per comunicare tra il software e il License Manager.

   - La **licenza web** utilizza la comunicazione http/https tra il software e il License Manager.
6. Specificare la **porta del server** se si è selezionato il tipo di **licenza web**.
7. Fare clic su **Modifica** per selezionare il server sul quale è installato License Manager e la chiave hardware di protezione.

   - oppure -

   Fare clic su **Sfoglia** per avviare una ricerca automatica del server su cui è installato **License Service**. Se la configurazione di rete è già stata eseguita in precedenza, appare un messaggio in cui si chiede se si desidera utilizzare la configurazione di rete attuale.
8. Per modificare o configurare le impostazioni di rete, fare clic sul pulsante **Network Settings Wizard**.
9. Fare clic su **OK**.
10. Riavviare il programma.



France
+33 (0) 562 601 080

Germany
+49 (0) 2103 2526 0

Singapore
+65 6908 0960

United States
+1 (414) 837 4800

Copyright 2026 TEKLYNX Corporation SAS. All rights reserved. LABEL MATRIX, LABELVIEW, CODESOFT, LABEL ARCHIVE, SENTINEL, PRINT MODULE, BACKTRACK, TEKLYNX CENTRAL, TEKLYNX, and Barcode Better are trademarks or registered trademarks of TEKLYNX Corporation SAS or its affiliated companies. All other brands and product names are trademarks and/or copyrights of their respective owners.

www.teklynx.com