Copy_Table Subroutine

Description

Copies a native table, the dictionary for the table, and indexing information if it exists, to a new name or location. The source and target table must be the same filing system type.

Syntax

Copy_Table *(tablename, location, database, newtablename, lockflag, deleteflag, overwriteflag, existflag, status, removerelationalindexing)*

Parameters

The Copy_Table subroutine has the following parameters.

Parameter	Description
tablename	Specifies the name of the table to copy. To copy just the dictionary portion of a table, specify Dict.tablename. Specify DATA. tablename to copy the data portion only.
location	The target volume for the copy process. Specify the location argument where the table will reside. If location is null, the table is copied to the location where *tablename* exists.
database	Specifies the target database for the copy process. If database is null, the table is copied to the database where *tablename* exists.
newtablename	Specifies the new table name.
lockflag	If true (1), lock the table during the copy process.
deleteflag	If true (1), delete the original table after the table is copied to a new name or location.
overwriteflag	If true (1), and a table exists with the same name as *newtablename*, overwrite it. If the table does not exist, a new one is created.
existflag	If true (1), copy the source table over the destination table, only if the destination table exists. In other words, copy the table, only when it is a replacement operation.
status	If true (1) on return, then the table was copied successfully. Note that while this does indicate success or failure, the Get_Status() function is the preferred method for obtaining status information. (Optional)
removerelation al-indexing	If true (1), any relational indexes will be removed from the destination table only. Cross reference and Btree indexes are however, left intact. (Optional)

Remarks

The Copy_Table command will copy a table only. It will not attach the table to the current session or add the table to the Database Definition. To attach the table refer to the Attach_Table command. To define the database refer to the Define_Database command.

See also

Delete_Index, Get_Status, Attach_Table, Alias_Table, Rename_Table, Copy Table Dialog

Example

```
/* Copies CAR_PARTS_BAK to CAR_PARTS in volume E:\DAT, database INVENTORY.
No locking is performed. The original table is deleted after the copy is performed.
If a table with the same name exists, it is overwritten. */

declare subroutine Set_Status, Copy_Table
declare function Get_Status

status = ""
Set_Status(0)
Copy_Table("CAR_PARTS_BAK", "E:\DAT", "INVENTORY", "CAR_PARTS", "0", "1", "1", "0",status,"0")

If Get_Status(ErrCodes) Then
  GoSub ErrorHandling
End
```