

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
<i>tablename</i>	Specifies the name of the table to copy. To copy just the dictionary portion of a table, specify Dict. <i>tablename</i> . Specify DATA. <i>tablename</i> to copy the data portion only.
<i>location</i>	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 <i>tablename</i> exists.
<i>database</i>	Specifies the target database for the copy process. If database is null, the table is copied to the database where <i>tablename</i> exists.
<i>newtablename</i>	Specifies the new table name.
<i>lockflag</i>	If true (1), lock the table during the copy process.
<i>deleteflag</i>	If true (1), delete the original table after the table is copied to a new name or location.
<i>overwriteflag</i>	If true (1), and a table exists with the same name as <i>newtablename</i> , overwrite it. If the table does not exist, a new one is created.
<i>existflag</i>	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.
<i>status</i>	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)
<i>removerelational-indexing</i>	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
```