

Alias_Table Subroutine

Description

Creates a temporary synonym (an alias) for an existing table. The permanent database definition is not altered.

Syntax

Alias_Table(*locationlist*, *database*, *tablename*, *aliasname*)

Parameters

The Alias_Table subroutine has the following parameters.

Parameter

Parameter	Description
locationlist	States the location of the existing table. Can be specified in either of the following ways: [location] [location, table_type] [location, table_type, control_table_path] volume_pointer_name Typically, this value will be the path to a directory containing Native Tables.
database	Database to which the table belongs. If null, the current database is used. Use the List_Tables command to determine the database for a specific table. Note that most of the system tables belong to the "GLOBAL" database.
tablename	Name of the table for which you are creating an alias. Use the form that is returned by List_Volume. If List_Volume shows that the table name is qualified (SALLY@TABLENAME, for example), you must also qualify the table name.
aliasname	The alias for the table. If Alias_Table is successful, the specified table will be accessible using the alias name.

Caution: Do not use the name of a system table for the alias name parameter. Use Get_Status() to check if Alias_Table failed.

Remarks

Note: All parameters are passed by reference, not by value.

See also

[List_Volume](#), [Attach_Table](#), [Copy_Table](#)

Example

```
/* The alias "BACKUP" is used for the CAR_PARTS table in the volume E:\DAT in the database INVENTORY. */  
  
Declare Subroutine Alias_Table, Set_Status, Msg  
Declare Function Get_Status  
  
Alias_Table ("E:\DAT", "INVENTORY", "CAR_PARTS", "BACKUP")  
if Get_Status() then  
    Msg(@window, "Unable to alias the CAR_PARTS table.")  
    Set_Status (0)  
end
```