Transfer Statement

Description

Moves the contents of one variable into another variable, and clears the original variable.

Syntax

Transfer *variable1 To variable2*

Parameters

The Transfer statement accepts arguments for the following parameters.

Parameter	Description
variable1	The source variable. After the execution of a Transfer statement, variable1 is null.
variable2	The target. After a transfer, variable2 contains the data originally represented by variable1. It is not necessary to initialize the target variable before transferring to it.

Remarks

The Transfer statement moves the value of a variable by reference. Instead of copying the data represented by a variable, Transfer copies the pointer to that data. The effect is a very quick reassignment, with no loss of memory.

If the contents of a variable are copied using an assignment operator (for example, A = B), BASIC+ makes a copy of the value in the source variable (B), and loads it into the location represented by the target variable (A). The assignment operator is a transfer by value. This consumes both time and memory, and thus direct assignment is less efficient than using the Transfer statement.

Example

```
/* The program processes each record in the CUSTOMER file.
After calling the external program RECORD_STATS,
the program restores the values of @RECORD and @ID by transferring them from temporary variables.
The local subroutine FURTHER_PROCESSING is not shown. */

Open "TEST" To CUSTOMER_TABLE Else
  status = Set_FSError()
  Return
End
Select CUSTOMER_TABLE
Done = 0
Loop
  ReadNext @ID Else Done = 1
Until Done
  Read @RECORD From CUSTOMER_TABLE, @ID Then
    * save contents before calling routine
    * in case RECORD_STATS changes values
    SAVE_RECORD = @RECORD
    SAVE_ID = @ID
    Call RECORD_STATS
    * restore contents
    Transfer SAVE_RECORD To @RECORD
    Transfer SAVE_ID To @ID
    GoSub Further_Processing
  End
Repeat
```