ReadNext Statement

Description

Extracts the next key from the list of keys, until the list is exhausted.

Syntax

ReadNext *variable* [, *value*] [USING *cursorvar*] Then | Else *statements*

The ReadNext statement has the following fcparameters.

Parameter	Description
variable	Is assigned the row key from the selected list.
value	When a multi-valued sort list is being used, you can use the optional variable value. It returns the position of variable in the multi-valued field.
cursorvar	A variable containing the cursor variable set for this table in a Select...By statement.
Then	Executed if the ReadNext is successful.
Else	The statement(s) following Else are executed if a key cannot be read from the selected list, for example, when the list has been exhausted. The Status() function indicates the reason for the false branch, and the system variable @FILE_ERROR contains details about the nature of the error.

Note: A ReadNext may return a null record key before the active list is exhausted. Testing for the null key is not a reliable test for list exhaustion.

Use ReadNext to read the keys from a selected list. The list of keys can be from a BASIC+ Select statement. A Select statement is used in the BASIC+ program to create a list of record keys. For more information, refer to Read.

Caution: If you exit a ReadNext loop before exhausting the select list, you must clear the select list using ClearSelect. Otherwise, your results may be unpredictable.

Example: Processing the CUSTOMERS table.

```
declare function Set_FSError
open "CUSTOMERS" To customers_table else
status = Set_FSError()
return
end
select customers_table

Done = 0
loop
ReadNext @ID else Done = 1
Until Done Do
read @RECORD From customers_table, @ID else
status = Set_FSError()
return
end
* processing logic here ...
GoSub PROCESS
Repeat
return 0
PROCESS:
   /* process the row */
return
return
```

See also

Read, Select