



<http://www.cbl.com/pdf/SELCOPYRefCardMVS.pdf>

SELCOPY Reference cards for MVS, VSE, CMS, iSeries, UNIX and PC platforms, may be downloaded from: <http://www.cbl.com/selcdoc.html>

### Syntax

<int>	Positive or negative integer value.
<expr_int>	Expression evaluating to a positive or negative <int> value using only the + (add) and - (subtract) operators. Note that * (multiply) and / (divide) are not supported. May include POS keywords, user vars (@ ptrs, etc.), SELCOPY keyword vars, <int> values and/or equated values.
<pos_int>	<int> or <expr_int> value equal to a position in the Work Area.
<len_int>	<int> or <expr_int> value equal to a valid length.
<name_str>	Case insensitive character string. <name_str> may not include special characters: " ' , = !
<char_str>	Character string. Enclose in " or ' if case sensitivity required or string contains blanks or special characters. (See <name_str>).
<hex_str>	Hex string specified as X'nn' where n is a base 16 digit (0-F).
<char>	<char_str> or <hex_str> of length 1 byte.

### Fields:

Reference	Syntax	Description
<field_p1p2>	[POS] <pos_int> , <pos_int>	Field starts in the 1st position and ends in the 2nd. Field defined by <field_p1p2> must be an ascending range of positions.
<field_pLENn>	POS <pos_int> LENGTH <len_int>	Field starts in position <pos_int> and is of length <len_int> bytes.
<field_nATp>	<len_int> {ATIFROM} <pos_int>	Field of <len_int> bytes in length starting at position <pos_int>.
<field_p1>	<pos_int>	Field starts in position <pos_int> with length inherited from another field in the operation or equal to the current LRECL value.
<field_str>	<char_str>   <hex_str>	Literal string, with length implied by the string length.

### Keyword Variables:

Name	Direct Assignment	Indirect Assignment
RETCODE	RETCODE=<int>   <expr_int>	Updated by SELCOPY to flag run-time warnings at end of job.
LINE	LINE=<int>	Incremented for each line of data written to SELCOPY's output listing, e.g. following a PRINT.
INCOUNT	Not Applicable.	Separate values maintained for each input file. The relevant INCOUNT value is incremented following the READ of a file.

### User Variables:

Name	Direct Assignment	Indirect Assignment
@	@=<int>   <expr_int>   {<field_p1p2>   <field_nATp>} TYPE={B C P}	On every logical field compare against a range of positions where PTR is omitted. If successful, @ is set equal to the position, relative to the start of the User Work Area, of the 1st occurrence of the logical field found within the range of positions. If the field is not found, @ is set to NULL (0).
@<name_str>	@<name_str>=.... (As for @.)	On every logical field compare against a range of positions where PTR=<name_str> is specified. @<name_str> is set in the same way as for @.
LRECL	LRECL=.... (As for @.)	LRECL is set to the length of data last read.
DIFF	Not Applicable.	On every logical field compare that is not a compare against a range of positions. If the compare is unsuccessful, DIFF is set equal to the position within the 1st field, relative to the start of the User Work Area, of the 1st difference found. If the fields match, DIFF is set to NULL (0).

### POS Keywords:

Name	Description
CBLNAME	CBLNAME MODULE loaded by SELCOPY.
DATE	Today's ISO date within SELCOPY's DATE block. ("yy/mm/dd")
DSN	fileid of the last fname processed.
FNAME	fname string belonging to the last file processed.
FSIZE	4 byte binary field containing the size (#records) at open time of the last file processed.
FT	SELCOPY's internal File Table Control Block for last file processed.
HEAD	SELCOPY output listing header.
PARM	List of parameters passed to the SELCOPY job.
PCB	Program Communication Block for last IMS file processed.
RETCODE	4 byte binary field containing SELCOPY's return code.
RETSYS RETXV	4 byte binary field containing the system return code following the last XV operation.
RETVSAM	4 byte binary field (x'00' + 3 byte VSAM RPL Error Feedback).
RPL	VSAM, DB2, ADABAS, IMS Request Parameter List.
SEG	Segment name of last IMS record processed. (POS PCB+20).
SQLCA	DB2 SQL Communications Area Control Block.
SQLDA	DB2 SQL Descriptor Area Control Block.
SQLMA	DB2 formatted SQL Message Area.
STATUS	Status code of last IMS record processed. (POS PCB+10).
UXADIFF	4 byte binary field containing the absolute address of the byte referenced by POS DIFF. (See User Variables)
UXATPTR	4 byte binary field containing the absolute address of the byte referenced by POS @. (See User Variables)
UXINCNT	4 byte binary field containing a count of the number of records processed from the primary input file.
UXLINE	4 byte binary field containing the number of lines already written to the current page of the SELCOPY listing.
UXLRECL	4 byte binary field containing the current value of LRECL.
UXPGNO PGNO	4 byte packed decimal field containing the page number of the current page of the SELCOPY listing output.
UXREPLYL	4 byte binary field containing the length of the data received from the last LOG operation with parameter REPLY.

### Redundant Legacy Syntax:

- / (Dot slash). Specified in column 1 before any SELCOPY operation. These characters are ignored by SELCOPY.
- NOW (Synonym N). Precedes any SELCOPY operation keyword to indicate that the operation is executed unconditionally. This is default and it is recommended that NOW is omitted.
- FILE= (Synonym F=) preceding fname on any I/O statement is recommended to be omitted.

### Job Environment

- \* (Asterisk) Specified in any column to define start of comment data.
- > (Asterisk Greater-Than) Specified in any column to define comment data also written to the summary.
- < (Asterisk Less-Than) Specified in column 1 to define the whole record as comment, ignoring command separator characters.
- \ (Backslash) Specified as the last character on the record to define control statement continuation on the following record.
- ! (Exclamation mark) Default control statement separation character. OPTION SEP=<char>OFF overrides.

EQU <name\_str> { <int> | <expr\_int> | <char\_str> | <hex\_str> } ...

### OPTION

[ASA0INOASAO] [ABTRAP=ON/OFF] [CONTMAX=<int>] [DATAWIDTH=<int>] [FILL=<char>] [HEAD=<char\_str>INO] [NOPTCLIPRTCTL] [NOPRINT] [NOPTOT] [PAGEDEPTH=<int>] [PAGEWIDTH=<int>] [PASS=<hex\_str>] [RANGE=<char\_str>...] [RDWINORDW] [REPORT] [SEP=<char>OFF] [SITE=<char\_str>] [SSN=<name\_str>] [WORKLEN=<int>]

REPORT [parm1] [parm2] [parm3] ...

### Run-Time Environment

### Logic Flow:

- IF condition \* AND has higher priority than OR, i.e.
- AND condition \* IF A !AND B !OR C !AND D
- OR condition \* is equivalent to: IF (A & B) | (C & D)
- THEN clause
- [ELSE clause] \* THEN IF and ELSE IF allows nesting of further
- [THEN clause] \* IF/AND/OR conditions.

clause is a SELCOPY run-time operation (can be IF).

condition is one of:

1. field1 <op> field2 [FILL=<char>]
2. range <op> field2 [PTR=@<name\_str>] [STEP=<int>]
3. {field1 TYPE=BIP | <expr\_int>} <op> {field2 TYPE=BIP | <expr\_int>}
4. {EOFIDIRIDATA} {INCOUNTLINEIRETCODE <op> {field2 TYPE=B | <expr\_int>}} [ [FILE=<name\_str> ]

Where <op> is one of the following relational or bitwise operators:

- = Equals. (Default if operator expected but not specified.)
- ≠ Not Equal.
- > Greater Than.
- >= Greater Than or Equal.
- < Less Than.
- <= Less Than or Equal.
- ONES True if all bits tested are set ON.
- ZEROS True if all bits tested are set OFF.
- MIXED True if, of all the bits tested, at least one is ON and another OFF.

**Logic Flow (Continued):**

**GOTO** <name\_str> | **GET** | **EOJ** | **CANCEL**  
**DO** <name\_str> / **RETURN**  
**CALL** <name\_str> [**ENTRY**=<int>] [**ENTRYEOJ**=<int>]  
 [parm1] [parm2] ... [parm16]

**Input/Output:**

**OPEN** fname \* Not normally required.  
**CLOSE** fname \* Not normally required.

**READ** {fname|CARDIDUMMY|dbname} [#]<int>]  
 | [ [ fname ] DSN=]fileid

[ Direct Read Params | Multiple Input Params |  
 VSAM Params | DB2 Params | IMS Params | ADABAS Params ]

[**BLKSIZE**=<int>] [**DEFER**] [**INTO**=<field\_p1>] [**LRECL**=<int>]  
 [**RDWINORDW**] [**RECFM**=F|FB|V|VBIU]  
 [**TRUNCINOTRUNC**] [**UPD**] [**WORKLEN**=<int>] [**WTO**]

Direct Read Params:

{**KEY**=**STARTKEY**=**IKGE**=**STARTKGE**=} field [**KEYPOS**=<int>]  
 [**KEYLEN**=<len\_int>]  
 | {**REC**=**STARTREC**=} {field [TYPE=BICIP] | <int>}  
 | {**RBA**=**STARTRBA**=} {field [TYPE=BICIP] | <int>}

Multiple Input Params:

**DIR** | **DIRDATA**

Use the **FLAG EOMEMB** statement to cause early end of DATA for the current DIRDATA input member.

VSAM Params:

**VSAMIKSDSIESDSIRRDSSIAM** [**PASS**=<field\_str>] [**FWDIBWD**] [**UPD**]

DB2 Params:

{**SQL**=select\_clause | **TABLE**=table\_name [**WHERE**=where\_clause]  
 [ [**UPD**=update\_clause | **SORT**=sort\_clause ] ] [**FMT**=col\_list] [**NODUP**]}  
 [ [**NULLS**] [**VLEN**] [**PF**] ] [**CHAR** [**SEP**] ] [**DB2**] [**SSN**=<char\_str>]

IMS Params:

**IMS** | **DL1** [**SEG**=seg\_name [**SEARCH**=fld\_name <op> field]  
 | **SSA**=ssa\_clause ]

ADABAS Params:

**ADABAS FORMAT**={fmt\_buff | ALL} [**PASS**=field] [**CIPHER**=field] [**EXCL**]  
 [ **SEQ**=desc\_name val\_buff | **SEQ**=desc\_name {**KEY**=**STARTKEY**=}field]  
 | {**ISN**=**STARTISN**=} {field [TYPE=BICIP] | <int> } ]

**CAT** fname \* Must follow a READ or another CAT operation.

**UPDATE** fname [**FROM**=field]

**DELETE** fname

**WRITE** {fname|DUMMY} | [ [ fname ] DSN=]fileid

[ VSAM Params ]

[**APPEND**] [**BLKSIZE**=<int>] [**DEFER**] [**FILL**=<char>]  
 [**FROM**=field] [**LRECL**=<int>] [**NEWBLK**]  
 [**RECFM**=F|FB|V|VBIU] [**TRUNCINOTRUNC**] [**WTO**]

VSAM Params:

**VSAMIKSDSIESDSIRRDSSIAM** [**PASS**=<field\_str>] [**REUSE**]

**INSERT** {fname|dbname} [#]<int>] [**FROM**=field]

[ VSAM Params | DB2 Params | IMS Params | ADABAS Params ]

VSAM Params:

**VSAMIKSDSIESDSIRRDSSIAM** [**PASS**=<field\_str>]

DB2 Params:

**TABLE**=table\_name [**FMT**=col\_list] [**DB2**] [**SSN**=<char\_str>]

IMS Params:

**IMS** | **DL1** {**SEG**=seg\_name | **SSA**=ssa\_clause }

ADABAS Params:

**ADABAS FORMAT**={fmt\_buff | ALL} [**PASS**=field] [**CIPHER**=field] [**EXCL**]

**PRINT** [**FROM**=field | **LENGTH**=<len\_int>]  
 [**PAGEDEPTH**=<int>] [**PAGEWIDTH**=<int>] [**DATAWIDTH**=<int>]  
 [TYPE=BICIDIHIMPINIS]

**PUNCH** [**FROM**=field]

**LOG** [**FROM**=field | **LENGTH**=<len\_int>]  
 [**REPLY**=field] [**CLEAR**] [TYPE=BICIDIHIMPINIS]

**PLOG** [**FROM**=field] \* Both PRINT and LOG.

**DUMMY** \* Equivalent to WRITE DUMMY (not READ DUMMY.)

**Data Assignment:**

**MOVE** field1 TO field2 [**FILL**=<char>]

[**MOD**] **POS** field1 [<op>|=] field2 [**FILL**=<char>]

Where <op> is one of the following bitwise operators:

- OR** Set ON if either bit is ON.  
Set OFF if both bits are OFF.
- XOR** Set ON if one bit is ON and the other bit is OFF.  
Set OFF if both bits are ON or both bits are OFF.
- AND** Set ON if both bits are ON.  
Set OFF if either bit is OFF.

**GENERATE** field1 { [TYPE=BICIPIZ] **RANGE**=int1,int2  
 | TYPE=C [**RANGE**=<field\_str> ] }  
 [ **BASE**=<char\_str>|<hex\_str> ]

**Arithmetic:**

**ADD** {field1<expr\_int>} TO {field2<expr\_int>} [**INTO** field3] [TYPE=BIP]  
**SUB** {field1<expr\_int>} FROM {field2<expr\_int>} [**INTO** field3] [TYPE=BIP]  
**MULT** {field1<expr\_int>} BY {field2<expr\_int>} [**INTO** field3] [TYPE=BIP]  
**DIV** {field1<expr\_int>} BY {field2<expr\_int>} [**INTO** field3] [**REM** <field4>]  
 [TYPE=BIP]

**Data Conversion:**

**CVxx** field1 [**TO** field2] [**FORMAT**=<char\_str>]

<b>CVAE</b>	ASCII	→	EBCDIC
<b>CVEA</b>	EBCDIC	→	ASCII
<b>CVCB</b>	Character/Zoned Dec	→	Binary
<b>CVCP</b>	Character/Zoned Dec	→	Packed Decimal
<b>CVCF</b>	Character/Zoned Dec	→	Floating Point
<b>CVCH</b>	Character	→	Hexadecimal
<b>CVBC</b>	Binary	→	Character/Zoned Dec
<b>CVBP</b>	Binary	→	Packed Decimal
<b>CVPC</b>	Packed Decimal	→	Character/Zoned Dec
<b>CVPB</b>	Packed Decimal	→	Binary
<b>CVFC</b>	Floating Point	→	Character/Zoned Dec
<b>CVHC</b>	Hexadecimal	→	Character

**TRAN** field1 {field2 field3 | **TAB**=field4  
 | {UPPER|LOWER|PRINT|ASCII|EBCDIC} } [**INTO** field5]

**UPPER** field1 [**INTO** field2]  
**LOWER** field1 [**INTO** field2]

**COMPRESS** field1 TO field2  
**EXPAND** field1 TO field2

**System Commands:**

**DB2 SQL**=field [**SSN**=<char\_str>]

**XV** {**FETCH**|**GET**|**NEXT**} field1 **INTO** field2  
**XV** **SET** field1 **FROM** field2

**Common Parameters:**

- STOPAFT**=<int> Specifies the maximum number of times a control statement may be executed in the current job run.
- TIMES**=<int> Specifies the number of times that a statement should be executed before processing passes to the next statement.



**Compute (Bridgend) Ltd**  
 8 Merthyr Mawr Road, Bridgend, UK, CF31 3NH

Tel: +44 (1656) 652222  
[support@cbl.com](mailto:support@cbl.com)