The following acknowledgment has been reproduced from COBOL Edition, U.S. Department of Defense, at the request of the Conference on Data Systems Languages.

"Any organization interested in reproducing the COBOL report and specifications in whole or in part, using ideas taken from this report as the basis for an instruction manual or for any other purpose is free to do so. However, all such organizations are requested to reproduce this section as part of the introduction to the document. Those using a short passage, as in a book review, are requested to mention ‘COBOL’ in acknowledgment of the source, but need not quote this entire section.

“COBOL is an industry language and is not the property of any company or group of companies, or of any organization or group of organizations.

“No warranty, expressed or implied, is made by any contributor or by the COBOL Committee as to the accuracy and functioning of the programming system and language. Moreover, no responsibility is assumed by any contributor or by the committee, in connection therewith.

“Procedures have been established for the maintenance of COBOL. Inquiries concerning the procedures for proposing changes should be directed to the Executive Committee of the Conference on Data Systems Languages.

“The authors and copyright holders of the copyrighted material used herein


have specifically authorized the use of this material in whole or in part, in the COBOL specifications. Such authorization extends to the reproduction and use of COBOL specifications in programming manuals or similar publications."
CONTENTS

I. COBOL Character Set 1
II. COBOL Reserved Words 2
III. Complete COBOL Language Formats 12
    General Format for IDENTIFICATION DIVISION 13
    General Format for ENVIRONMENT DIVISION 13
    General Format for DATA DIVISION 20
    General Format for PROCEDURE DIVISION 35
IV. Function Names Available in Extensions to COBOL 85 74
V. New COBOL 9X Reserved Words 75
# I. COBOL Character Set

The following lists are in ascending order:

<table>
<thead>
<tr>
<th>EBCDIC</th>
<th>ASCII</th>
</tr>
</thead>
<tbody>
<tr>
<td>space</td>
<td>space</td>
</tr>
<tr>
<td>.</td>
<td>”</td>
</tr>
<tr>
<td>&lt;</td>
<td>$</td>
</tr>
<tr>
<td>(</td>
<td>’</td>
</tr>
<tr>
<td>+</td>
<td>(</td>
</tr>
<tr>
<td>$</td>
<td>)</td>
</tr>
</tbody>
</table>

| period, decimal point   | quotation mark       |
| less than              | dollar sign          |
| left parenthesis        | single quotation mark|
| plus symbol             | left parenthesis      |
| dollar sign             | right parenthesis    |
II. COBOL Reserved Words

Each COBOL compiler has a list of reserved words that:

1. Includes all entries in the ANS COBOL standard.
2. Includes additional entries not part of the standard but that are either VAX or IBM compiler extensions. These are called enhancements.

The following is based on the 1974 and 1985 American National Standard. You may find that your computer has additional reserved words. Diagnostic messages will print if you are using a reserved word incorrectly.
New reserved words that are not relevant for COBOL 74, but are relevant only for COBOL 85, are denoted with a single asterisk (*). COBOL 74 reserved words that are not reserved in the new standard are denoted with a double asterisk (**) Words in red are VAX COBOL 85 extensions. Words in blue are IBM COBOL 85 extensions. Boxed words are both VAX and IBM COBOL 85 extensions.

ACCEPT  CHARACTERS
ACCESS    CLASS *
ACTUAL    CLOCK-UNITS
ADD       CLOSE
ADVANCING  COBOL
AFTER     CODE
ALL       CODE-SET
ALLOWING  COLLATING
ALPHABET  COLUMN
ALPHABETIC COM-REG
ALPHABETIC-LOWER   COMMIT
ALPHABETIC-UPPER   COMMIT
ALPHANUMERIC   COMMON
ALPHANUMERIC-EDITED COMMUNICATION
ALSO       COMP
ALTER     COMP-1
ALTERNATE  COMP-2
AND       COMP-3
ANY       COMP-4
APPLY     COMP-5
ARE       COMP-6
AREA      COMPUTATIONAL
DATE-COMPILED
DATE-WRITTEN
DAY
DAY-OF-WEEK *
DB
DB-ACCESS-CONTROL-KEY
DB-CONDITION
DB-CURRENT-RECORD-ID
DB-CURRENT-RECORD-NAME
DB-exception
DB-key
DB-key
DB-RECORD-NAME
DB-SET-NAME
DB-STATUS
DEBUG-SUB
DEBUG-UWA
DE
DEBUG-CONTENTS
DEBUG-ITEM
DEBUG-LENGTH
DEBUG-LINE
DEBUG-NAME
DEBUG-NUMERIC-CONTENTS
DEBUG-SIZE
DEBUG-START
DEBUG-SUB
DEBUG-SUB-1
DEBUG-SUB-2
DEBUG-SUB-3
EJECT
ELSE
EMI
EMPTY
ENABLE
END
END-ACCEPT
END-ADD *
END-CALL *
END-COMMIT *
END-COMPUTE *
END-CONNECT
END-DELETE *
END-DISCONNECT
END-DIVIDE *
END-ERASE
END-EVALUATE *
END-FETCH
END-FIND
END-FREE
END-GET
END-IF *
ENDING
END-KEEP
END-MODIFY
END-MULTIPLY *
DEBUG-SUB-ITEM
DEBUG-SUB-N
DEBUG-SUM-NUM
DEBUGGING
DECIMAL-POINT
DECLARATIVES
DEFAULT
DELETE
DELIMITER
DEPENDING
DESCENDING
DESCRIPTOR
DESTINATION
DETAIL
DICTIONARY
DISABLE
DISCONNECT
DISP
DISPLAY
DISPLAY-1
DISPLAY-6
DISPLAY-7
DISPLAY-9
DIVIDE
DIVISION
DOES
DOWN
DUPPLICATE
DUPLICATES
DYNAMIC

END-OF-PAGE
END-PERFORM *
END-READ *
END-READY
END-RECEIVE *
END-RECONNECT
END-RETURN *
END-REWRITE *
END-ROLLBACK
END-SEARCH *
END-START *
END-STORE
END-STRING *
END-SUBTRACT *
END-UNSTRING *
END-WRITE *
ENTER
ENTRY
ENVIRONMENT
EOP
EQUAL
EQUALS
ERASE
ERROR
ESI
EVALUATE *
EVERY **
EXCEEDS
EXCEPTION
EXCLUSIVE
EXIT
EXOR
EXTEND
EXTERNAL *

FAILURE
FALSE *
FD
FETCH
FILE
FILE-CONTROL
FILE-LIMIT
FILE-LIMITS
FILLER
FINAL
FIND
FINISH
FIRST
FOOTING
FOR
FREE
FROM

GENERATE
GET
GIVING
GLOBAL *
GO
GOBACK
GREATER
GROUP

KEEP
KEY
LABEL
LAST
LD
LEADING
LEAVE
LEFT
LENGTH
LESS
LIMIT
LIMITS
LINAGE
LINAGE-COUNTER
LINE
LINE-COUNTER
LINES
LINKAGE
LOCALLY
LOCK
LOW-VALUE
LOW-VALUES
MATCH
MATCHES
MEMBER
MEMBERSHIP
MEMORY **
MERGE
HEADING

HIGH-VALUE

HIGH-VALUES

ID

IDENTIFICATION

IF

IN

INCLUDING

INDEX

INDEXED

INDICATE

INITIAL

INITIALIZE *

INITIATE

INPUT

INPUT-OUTPUT

INSERT

INSPECT

INSTALLATION

INTO

INVALID

I-O

I-O-CONTROL

IS

JUST

JUSTIFIED

KANJI
ON
ONLY
OPEN
OPTIONAL
OR
ORDER *
ORGANIZATION
OTHER *
OTHERS
OUTPUT
OVERFLOW
OWNER

PACKED-DECIMAL *
PADDING *
PAGE
PAGE-COUNTER
PARAGRAPH
PASSWORD
PERFORM
PF
PH
PIC
PICTURE
PLUS
POINTER
POSITION
POSITIVE
PRESENT
PRINTING

REFERENCE-MODIFIER
REFERENCES
REGARDLESS
RELATIVE
RELEASE
RELOAD
REMAINDER
REMOVAL
RENAMES
REPLACE *
REPLACING
REPORT
REPORTING
REPORTS
REREAD
REKUN
RESERVE
RESET
RETAINING
RETRIEVAL
RETURN
RETURN-CODE
REVERSED
REWIND
REWRITE
RF
RH
RIGHT
RMS-Filename
RMS-STS
PRIOR
PROCEDURE
PROCEDURES
PROCEED
PROGRAM
PROGRAM-ID
PROTECTED
PURGE *

QUEUE
QUOTE
QUOTES

RANDOM
RD
READ
READERS
  READY
REALM
REALMS
RECEIVE
RECONNECT
RECORD
RECORD-NAME
RECORD-OVERFLOW
RECORDING
RECORDS
REDEFINES
REEL
REFERENCE *
This guide contains the composite language formats of the American National Standard COBOL. Shaded entries are those that are applicable to COBOL 85 only. Entries in blue are IBM extensions. Entries in red are VAX extensions. Entries with an * are both IBM and VAX extensions.
General Format for IDENTIFICATION DIVISION

{IDENTIFICATION DIVISION.}
{ID DIVISION.}

PROGRAM-ID. program-name [IS {COMMON INITIAL} PROGRAM].

[AUTHOR. [comment-entry] ... ]
[INSTALLATION. [comment-entry] ... ]
[DATE-WRITTEN. [comment-entry] ... ]
[DATE-COMPILED. [comment-entry] ... ]
[SECURITY. [comment-entry] ... ]

General Format for ENVIRONMENT DIVISION*

[ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. [computer-name [WITH DEBUGGING MODE]].]

[OBJECT-COMPUTER. [computer-name

[PROGRAM COLLATING SEQUENCE IS alphabet-name-1]
[SEGMENT-LIMIT IS segment-number]].]
[SPECIAL-NAMES. [[implementor-name-1
   IS mnemonic-name-1 [ON STATUS IS condition-name-1 [OFF STATUS IS condition-name-2]]
   IS mnemonic-name-2 [OFF STATUS IS condition-name-2 [ON STATUS IS condition-name-1]]]
   ON STATUS IS condition-name-1 [OFF STATUS IS condition-name-2]
   OFF STATUS IS condition-name-2 [ON STATUS IS condition-name-1]
   ALPHABET alphabet-name-1 IS
   [ASCII
   {EBCDIC/}
   STANDARD-1
   STANDARD-2
   NATIVE
   implementor-name-2
   {literal-1 [THROUGH literal-2]...}{ALSO literal-3}...
   ]...
   ]...]
   [SYMBOLIC CHARACTERS [{{symbolic-character-1}...IS{integer-1}...}...]
   ]...

*The ENVIRONMENT DIVISION, CONFIGURATION SECTION, and INPUT-OUTPUT SECTION entries are required for COBOL 74.
IN alphabet-name-2 ... CLASS class-name IS literal-4 [THROUGH literal-5] ... ... CURRENCY SIGN IS literal-6
DECIMAL-POINT IS COMMA.[[INPUT-OUTPUT SECTION.
FILE-CONTROL
{file-control-entry} ...
I-O-CONTROL.
[SAME RECORD SORT SORT-MERGE] AREA FOR file-name-1 {file-name-2} ...
... MULTIPLE FILE TAPE CONTAINS
{file-name-3 [POSITION integer-1]} ...
... .]]

---

General Format for FILE-CONTROL Entry

SEQUENTIAL FILE
SELECT [OPTIONAL] file-name-1
ASSIGN TO \{implementor-name-1\} ... \\
RESERVE integer-1 [AREA AREAS] \\
\[ORGANIZATION IS SEQUENTIAL] \\
\[BLOCK CONTAINS [smallest-block TO] blocksize \{RECORDS CHARACTERS\}\] \\
\[CODE-SET IS alpha-name\] \\
\[PADDING CHARACTER IS \{data-name-1\}\]  \\
\[RECORD DELIMITER IS \{STANDARD-1 \{implementor-name-2\}\}\] \[ACCESS MODE IS SEQUENTIAL\] \\
\[FILE STATUS IS data-name-2\]. \\
RELATIVE FILE \\
SELECT [OPTIONAL] file-name-1 \\
ASSIGN TO \{implementor-name-1\} ... \\
RESERVE integer-1 [AREA AREAS]
ORGANIZATION IS RELATIVE

BLOCK CONTAINS [smallest-block TO] blocksize {RECORDS CHARACTERS}

PASSWORD IS data-name

ACCESS MODE IS

{SEQUENTIAL [RELATIVE KEY IS data-name-1]}

{RANDOM}

{DYNAMIC RELATIVE KEY IS data-name-1}

FILE STATUS IS data-name-2.

INDEXED FILE

SELECT [OPTIONAL] file-name-1

ASSIGN TO {implementor-name-1} {literal-1} ...

RESERVE integer-1 {AREA AREAS}

ORGANIZATION IS INDEXED

BLOCK CONTAINS [smallest-block TO] blocksize {RECORDS CHARACTERS}

PASSWORD IS data-name
ACCESS MODE IS \{ SEQUENTIAL \ \\
\ \ \ \ \ \ \ \ \ \ \ \ \ \ RANDOM \ \\
\ \ \ \ \ \ \ \ \ \ \ \ \ \ DYNAMIC \} \)

RECORD KEY IS data-name-1
[ALTERNATE RECORD KEY IS data-name-2 [WITH DUPLICATES]] ...
[FILE STATUS IS data-name-3].

SORT OR MERGE FILE

SELECT file-name-1 ASSIGN TO \{ implementor-name-1 \} \ldots .

REPORT FILE

SELECT [OPTIONAL] file-name-1

ASSIGN TO \{ implementor-name-1 \} \ldots

RESERVE integer-1 [AREA ]

[ ]

[ ORGANIZATION IS SEQUENTIAL]

[ BLOCK CONTAINS ] blocksize \{ RECORDS CHARACTERS \}]

CODE-SET IS alpha-name}
General Format—I-O-CONTROL

[I-O-CONTROL.

AFFILI

{DEFERRED-WRITE
EXTENSION extend-amt
FILL-SIZE
LOCK-HOLDING
MASS-INSERT

{CONTIGUOUS
CONTIGUOUS-BEST-TRY
PREALLOCATION preall-amt
PRINT-CONTROL
WINDOW window-pointers

} ON {file-name} ...

...}
General Format for DATA DIVISION

[DATA DIVISION.
[SUB-SHEMA SECTION, [subschema-entry [keeplist] ...]]

FILE SECTION.
[file-description-entry
{record-description-entry} ... ] ...
[sort-merge-file-description-entry
{record-description-entry} ... ] ...
[report-file-description-entry] ... ]
General Format—Subschema Description

DB subschema-name WITHIN schema-name
  [FOR database-name] [THRU stream-name]

General Format—Keeplist Description

LD keeplist-name [LIMIT IS integer].
Sequential File Description Entry

FD file-name-1

[IS EXTERNAL]
[IS GLOBAL]

BLOCK CONTAINS [integer-1 TO integer-2 {RECORDS CHARACTERS}]

RECORD IS VARYING IN SIZE [FROM integer-4 TO integer-5 CHARACTERS]
[DEPENDING ON data-name-1]
CONTAINS integer-6 TO integer-7 CHARACTERS

LABEL {RECORD IS STANDARD} {RECORDS ARE OMITTED}

VALUE OF {implementor-name-1 IS data-name-2} ... 

DATA {RECORD IS} {RECORDS ARE} {data-name-3} ...

LINAGE IS {data-name-4 integer-8} LINES WITH FOOTING AT {data-name-5 integer-9}
LINES AT TOP {data-name-6 integer-10} [LINES AT BOTTOM {data-name-7 integer-11}]
CODE-SET IS alphabet-name-1.
[ACCESS MODE IS] SEQUENTIAL
[FILE STATUS IS file-status].

RELATIVE FILE
FD file-name-1
[IS EXTERNAL]
[IS GLOBAL]
BLOCK CONTAINS [integer-1 TO] integer-2 {RECORDS
CHARACTERS}
RECORD CONTAINS integer-3 CHARACTERS
IS VARYING IN SIZE [[FROM integer-4][TO integer-5] CHARACTERS]
[DEPENDING ON data-name-1]
CONTAINS integer-6 TO integer-7 CHARACTERS
LABEL {RECORD IS} {STANDARD}
VALUE OF {implementor-name-1 IS {data-name-2}
literal-1} ... }
DATA {RECORD IS } {data-name-3} ...

[ACCESS MODE IS] SEQUENTIAL [RELATIVE KEY IS rel-key]
{RANDOM
DYNAMIC} RELATIVE KEY IS rel-key
INDEXED FILE
FD file-name-1
[IS EXTERNAL]
[IS GLOBAL]
BLOCK CONTAINS [integer-1 TO integer-2 {RECORDS CHARACTERS}]
RECORD CONTAINS integer-3 CHARACTERS
IS VARYING IN SIZE [[FROM integer-4][TO integer-5] CHARACTERS]
DEPENDING ON data-name-1
CONTAINS integer-6 TO integer-7 CHARACTERS
LABEL
RECORD IS {STANDARD}
RECORDS ARE {OMITTED}
VALUE OF {implementor-name-1 IS {data-name-2}} ...
DATA
RECORD IS {RECORDS ARE} (data-name-3) ...
[ACCESS MODE IS] {SEQUENTIAL}
{RANDOM}
{DYNAMIC}
RECORD KEY IS rec-key
ALTERNATE RECORD KEY IS alt-key [WITH DUPLICATES] ...
SORT-MERGE FILE

SD file-name-1

[RECORD

CONTAINS integer-1 CHARACTERS
IS VARYING IN SIZE [[FROM integer-2] [TO integer-3] CHARACTERS]
[DEPENDING ON data-name-1]
CONTAINS integer-4 TO integer-5 CHARACTERS]

DATA {RECORD IS
RECORDS ARE} {data-name-2} ...]

REPORT FILE

FD file-name-1

[IS EXTERNAL]
[IS GLOBAL]

[BLOCK CONTAINS [integer-1 TO] integer-2 {RECORDS
CHARACTERS}]

RECORD {CONTAINS integer-3 CHARACTERS
IS VARYING IN SIZE [[FROM integer-4][TO integer-5] CHARACTERS]
[DEPENDING ON data-name-1]
CONTAINS integer-6 TO integer-7 CHARACTERS]}

LABEL {RECORD IS
RECORDS ARE} {STANDARD
OMITTED}

VALUE OF {implementor-name-1 IS {data-name-2}} ...]
General Format for Data Description Entry

FORMAT 1
level-number [data-name-1]
FILLER
[REDEFINES data-name-2]
[IS EXTERNAL]
[IS GLOBAL]
[PICTURE PIC] IS character-string
[USAGE IS]

\[
\begin{bmatrix}
\text{BINARY} \\
\text{COMP-1*} \\
\text{COMPUTATIONAL-1*} \\
\text{COMP-2*} \\
\text{COMPUTATIONAL-2*} \\
\text{COMP-3*} \\
\text{COMPUTATIONAL-3*} \\
\text{DISPLAY} \\
\text{DISPLAY-1*} \\
\text{INDEX} \\
\text{PACKED-DECIMAL} \\
\text{POINTER*}
\end{bmatrix}
\]

[SIGN IS] \{LEADING TRAILING\} [SEPARATE CHARACTER]

OCCURS integer-2 TIMES

\[
\begin{bmatrix}
\text{ASCENDING} \\
\text{DESCENDING} \\
\text{INDEXED BY \{index-name-1\} \ldots}
\end{bmatrix}
\]

OCCURS integer-1 TO integer-2 TIMES DEPENDING ON data-name-4

\[
\begin{bmatrix}
\text{ASCENDING} \\
\text{DESCENDING} \\
\text{INDEXED BY \{index-name-1\} \ldots}
\end{bmatrix}
\]
SYNCHRONIZED
SYNC
JUSTIFIED
JUST
BLANK WHEN ZERO
ZEROES
ZEROS

VALUE IS

\[
\begin{cases}
\text{literal-1} \\
\text{EXTERNAL external-name} \\
\text{REFERENCE data-name} \\
\text{NULL} \\
\text{NULLS}
\end{cases}
\]

FORMAT 2

66 data-name-1 RENAMES data-name-2 
\[
\begin{cases}
\text{THROUGH} \\
\text{THRU}
\end{cases}
\]
data-name-3.
FORMAT 3

88 condition-name-1 {
   VALUE IS VALUES ARE
   
   literal-1
      \begin{align*}
         \text{EXTERNAL} & \text{ external-name} \\
         \text{REFERENCE} & \text{ data-name} \\
         \text{low-val}
      \end{align*}
   
   \begin{align*}
      \text{THROUGH} & \text{ external-name} \\
      \text{THRU} & \text{ data-name} \\
      \text{high-val}
   \end{align*}
   
   \cdots
   
   literal-2
      \begin{align*}
         \text{EXTERNAL} & \text{ external-name} \\
         \text{REFERENCE} & \text{ data-name} \\
      \end{align*}
   
   \begin{align*}
      \text{THROUGH} & \text{ external-name} \\
      \text{THRU} & \text{ data-name} \\
      \text{high-val}
   \end{align*}
\}
FORMAT 1
CD cd-name-1

FOR [INITIAL] INPUT

[[SYMBOLIC QUEUE IS data-name-1]
SYMBOLIC SUB-QUEUE-1 IS data-name-2
SYMBOLIC SUB-QUEUE-2 IS data-name-3
SYMBOLIC SUB-QUEUE-3 IS data-name-4
MESSAGE DATE IS data-name-5
MESSAGE TIME IS data-name-6
SYMBOLIC SOURCE IS data-name-7
TEXT LENGTH IS data-name-8
END KEY IS data-name-9
STATUS KEY IS data-name-10
MESSAGE COUNT IS data-name-11]]
[data-name-1, data-name-2, data-name-3, data-name-4, data-name-5, data-name-6, data-name-7, data-name-8, data-name-9, data-name-10, data-name-11]
FORMAT 2

CD cd-name-1 FOR OUTPUT

[DESTINATION COUNT IS data-name-1]
[TEXT LENGTH IS data-name-2]
[STATUS KEY IS data-name-3]
[DESTINATION TABLE OCCURS integer-1 TIMES
  [INDEXED BY {index-name-1} ... ]]
[ERROR KEY IS data-name-4]
[SYMBOLIC DESTINATION IS data-name-5].

FORMAT 3

CD cd-name-1

FOR [INITIAL] I-O

[[MESSAGE DATE IS data-name-1]
  [MESSAGE TIME IS data-name-2]
  [SYMBOLIC TERMINAL IS data-name-3]
  [TEXT LENGTH IS data-name-4]
  [END KEY IS data-name-5]
  [STATUS KEY IS data-name-6]]
[data-name-1, data-name-2, data-name-3,
  data-name-4, data-name-5, data-name-6]

General Format for Report Description Entry

RD report-name-1
[IS GLOBAL]
[CODE literal-1]
CONTROL IS \{\text{data-name-1} \ldots \}\)
\{\text{FINAL data-name-1} \ldots \}\)

PAGE \{\text{LIMIT IS} \\text{LIMITS ARE}\} \text{integer-1} \{\text{LINE LINES} [\text{HEADING} \text{integer-2}]\)
\{\text{FIRST DETAIL} \text{integer-3} [\text{LAST DETAIL} \text{integer-4}]\)
\{\text{FOOTING} \text{integer-5}\).

General Format for Report Group Description Entry

FORMAT 1
01 \{\text{data-name-1}\)
\{\text{LINE NUMBER IS} \text{integer-1} \{\text{ON NEXT PAGE}\}[\text{PLUS} \text{integer-2}]\)
\{\text{NEXT GROUP IS} \text{integer-3} [\text{PLUS} \text{integer-4}]\)
\{\text{NEXT PAGE}\}.
TYPE IS

REPORT HEADING
RH
PAGE HEADING
PH
CONTROL HEADING
CH
DETAIL
DE
CONTROL FOOTING
CF
PAGE FOOTING
PF
REPORT FOOTING
RF

[[USAGE IS] DISPLAY].

FORMAT 2
level-number [data-name-1]

LINE NUMBER IS {integer-1 [ON NEXT PAGE]}

[[USAGE IS] DISPLAY].
FORMAT 3
level-number [data-name-1]

\[
\begin{align*}
\{ \text{PICTURE} \} & \quad \text{IS character-string} \\
\{ \text{PIC} \} & \\
\{ \text{USAGE IS} \} & \quad \text{DISPLAY} \\
\{ \text{SIGN IS} \} & \{ \text{LEADING} \} \quad \{ \text{TRAILING} \} \quad \{ \text{SEPARATE CHARACTER} \} \\
\{ \text{JUSTIFIED} \} & \{ \text{JUST} \} \quad \text{RIGHT} \\
\{ \text{BLANK WHEN ZERO} \} & \\
\{ \text{LINE NUMBER IS} \} & \{ \text{integer-1} \} \quad \{ \text{ON NEXT PAGE} \} \} \\
\{ \text{PLUS integer-2} \} & \\
\{ \text{COLUMN NUMBER IS} \} & \{ \text{integer-3} \} \\
\{ \text{SOURCE IS} \} & \{ \text{identifier-1} \} \\
\{ \text{VALUE IS} \} & \{ \text{literal-1} \} \\
\{ \text{SUM} \} & \{ \text{identifier-2} \} \quad \{ \text{UPON} \} \{ \text{data-name-2} \} \quad \{ \text{...} \} \} \quad \{ \text{...} \} \\
\{ \text{RESET ON} \} & \{ \text{data-name-3} \} \\
\{ \text{FINAL} \} & \\
\{ \text{GROUP INDICATE} \}. \\
\end{align*}
\]
General Format for PROCEDURE DIVISION

FORMAT 1

[PROCEDURE DIVISION [USING {data-name-1} ... ] [GIVING identifier-1].
DECLARATIVES.
{section-name SECTION [segment-number].
  USE statement.
  [paragraph-name.
    [sentence] ... ] ... } ... 
END DECLARATIVES.]
{section-name SECTION [segment-number].
  [paragraph-name.
    [sentence] ... ] ... } ... ]

FORMAT 2

[PROCEDURE DIVISION [USING {data-name-1} ... ] [GIVING identifier-1].
[paragraph-name.
  [sentence] ... } ... ]

General Format for COBOL Verbs

ACCEPT identifier-1 [FROM mnemonic-name-1]
  [AT END imperative statement-1]
  [NOT AT END imperative statement-2]
  [END-ACCEPT]
WITH CONVERSION
REVERSED
WITH NO ECHO
DEFAULT IS \{def-src-lit
                       def-src-item
                              CURRENT VALUE\}
CONTROL KEY IN key-dest-item

\{[ON EXCEPTION stment] [NOT ON EXCEPTION stment2] \}
\{[AT END stment] [NOT AT END stment2] \}
END-ACCEPT

ACCEPT CONTROL KEY IN key-dest-item

FROM LINE NUMBER \{line-num
                    \{line-id [PLUS [plus-num]]\}
                    \{PLUS [plus-num]\}\}
FROM COLUMN NUMBER \{column-num
                     \{column-id [PLUS [plus-num]]\}
                     \{PLUS [plus-num]\}\}
ERASE [TO END OF] \{SCREEN
                    \{LINE\}\}
WITH BELL

\{[ON EXCEPTION stment] [NOT ON EXCEPTION stment2] \}
\{[AT END stment] [NOT AT END stment2] \}
ACCEPT cd-name-1 MESSAGE COUNT

ADD {identifier-1} \ literal-1 \ ... \ TO {identifier-2 [ROUNDED]} \ ...

[ON SIZE ERROR imperative-statement-1]
[NOT ON SIZE ERROR imperative-statement-2]
[END-ADD]

ADD {identifier-1} \ literal-1 \ ... \ TO {identifier-2} \ literal-2 \ ...

GIVING {identifier-3 [ROUNDED]} \ ...
[ON SIZE ERROR imperative-statement-1]
[NOT ON SIZE ERROR imperative-statement-2]
[END-ADD]

ADD {CORRESPONDING} \ identifier-1 \ TO \ identifier-2 [ROUNDED]

[ON SIZE ERROR imperative-statement-1]
[NOT ON SIZE ERROR imperative-statement-2]
[END-ADD]

ALTER {procedure-name-1 \ TO \ [PROCEED \ TO \ procedure-name-2] \ ...}

CALL {identifier-1} \ [USING \ {BY REFERENCE \ identifier-2 \ ... \} \ ...
]
CALL \{identifier-1\} \{literal-1\}

[ON OVERFLOW imperative-statement-1 [END-CALL]]

USING \{identifier-2\} ...

[BY REFERENCE]

ADDRESS OF

ADDRESS OF

LENGTH OF

BY CONTENT

BY DESCRIPTOR

BY VALUE

OMITTED

[BY REFERENCE]

[BY CONTENT]

[BY DESCRIPTOR]

[BY VALUE]

OMITTED

[ON EXCEPTION imperative-statement-1]

[NOT ON EXCEPTION imperative-statement-2]

[ON OVERFLOW imperative-statement-2]

[NOT ON OVERFLOW imperative-statement-2]

[END-CALL]

CANCEL \{identifier-1\} ...

[END-CALL]
SW CLOSE 

\[
\text{file-name-1} \left\{ \begin{array}{c}
\text{REEL} \\
\text{UNIT} \\
\text{FOR REMOVAL}
\end{array} \right\} \left\{ \begin{array}{c}
\text{NO REWIND} \\
\text{LOCK}
\end{array} \right\} \ldots
\]

RI CLOSE \{file-name-1 [WITH LOCK]\} ...  

\begin{enumerate}
\item \text{COMMIT [RETAINING]}
\item \text{ON ERROR stment}
\item \text{NOT ON ERROR stment2}
\item \text{END-COMMIT}
\end{enumerate}

\begin{enumerate}
\item \text{COMPUTE \{identifier-1 [ROUNDED]\} ... \{=\} arithmetic-expression-1}
\item \text{ON SIZE ERROR imperative-statement-1]
\item \text{NOT ON SIZE ERROR imperative-statement-2]}
\item \text{END-COMPUTE}
\end{enumerate}

\begin{enumerate}
\item \text{CONNECT [record-name] TO \{\{set-name\} ... \}}
\item \text{REALM RECORD \{set-name\} ...}
\item \text{set-name \{set-name\} ...}
\item \text{CURRENCY}
\item \text{ON ERROR stment [NOT ON ERROR stment2]}
\item \text{END-CONNECT}
\end{enumerate}
CONTINUE
DELETE file-name-1 RECORD
  [INVALID KEY imperative-statement-1]
  [NOT INVALID KEY imperative-statement-2]
  [END-DELETE]
DISABLE {INPUT [TERMINAL]
  I-O TERMINAL
  OUTPUT} cd-name-1
DISCONNECT [record-name] FROM {set-name} ...
  ALL
  [ON ERROR stment]
  [NOT ON ERROR stment2]
  [END-DISCONNECT]
DISPLAY {identifier-1} ... [UPON {CONSOLE
  SYSOUT
  mnemonic-name-1}] [WITH NO ADVANCING]
DISPLAY \{src-item

AT LINE NUMBER
\\{line-num
\\\{line-id \PLUS \{plus-num\}\}
\PLUS \{plus-num\}\}

AT COLUMN NUMBER
\\{column-num
\\\{column-id \PLUS \{plus-num\}\}
\PLUS \{plus-num\}\}

ERASE [TO END OF] \{SCREEN
LINE \}\}

WITH BELL
UNDERLINED
BOLD
WITH BLINKING
REVERSED
WITH CONVERSION

\[WITH\ NO ADVANCING\]

DIVIDE \{identifier-1\} \ INTO \{identifier-2\} \ GIVING identifier-3 \[ROUNDED\]

REMAINDER identifier-4
\[ON\ SIZE\ ERROR\ imperative-statement-1\]
DIVIDE {identifier-1} \{literal-1\} \text{BY} \{identifier-2\} \{literal-2\} \text{GIVING} \{identifier-3\} \text{[ROUNDED]} \text{REMAINDER} identifier-4 \text{[ON SIZE ERROR imperative-statement-1]} \text{NOT ON SIZE ERROR imperative-statement-2} \text{END-DIVIDE} \text{[ON SIZE ERROR imperative-statement-1]} \text{NOT ON SIZE ERROR imperative-statement-2} \text{END-DIVIDE} \text{DIVIDE} \{identifier-1\} \{\text{INTO}\} \{identifier-2\} \{\text{ROUNDED}\} \text{GIVING} \{identifier-3\} \text{[ROUNDED]} \text{[ON SIZE ERROR imperative-statement-1]} \text{NOT ON SIZE ERROR imperative-statement-2} \text{END-DIVIDE} \text{DIVIDE} \{identifier-1\} \{\text{INTO}\} \{identifier-2\} \{\text{ROUNDED}\} \text{GIVING} \{identifier-3\} \text{[ROUNDED]} \text{[ON SIZE ERROR imperative-statement-1]} \text{NOT ON SIZE ERROR imperative-statement-2} \text{END-DIVIDE} \text{DIVIDE} \{identifier-1\} \{\text{INTO}\} \{identifier-2\} \{\text{ROUNDED}\} \text{GIVING} \{identifier-3\} \text{[ROUNDED]} \text{[ON SIZE ERROR imperative-statement-1]}
[NOT ON SIZE ERROR imperative-statement-2]
[END-DIVIDE]

ENABLE

\[
\begin{align*}
\text{INPUT [TERMINAL]} & \quad \text{cd-name-1} \\
\text{I-O TERMINAL} & \\
\text{OUTPUT} & 
\end{align*}
\]

ENTRY literal USING identifier-1 ...

ERASE [ALL [record-name]]

[ON ERROR stment]

[NOT ON ERROR stment2]

[END-ERASE]

EVALUATE

\[
\begin{align*}
\text{identifier-1} & \quad \text{literal-1} \\
\text{expression-1} & \quad \text{TRUE} \\
\text{FALSE} & \\
\text{also} & \quad \text{identifier-2} \\
\text{literal-2} & \quad \text{expression-2} \\
\text{TRUE} & \\
\text{FALSE} & \\
\text{...} & 
\end{align*}
\]

\{ WHEN

\[
\begin{align*}
\text{ANY} & \quad \text{condition-1} \\
\text{TRUE} & \\
\text{FALSE} & \\
\text{NOT} & \quad \text{identifier-3} \\
\text{literal-3} & \quad \text{arithmetic-expression-1} \\
\text{THROUGH} & \quad \text{THRU} & \quad \text{identifier-4} \\
\text{litera-4} & \quad \text{arithmetic-expression-2} & 
\end{align*}
\]

\}
[ALSO
(ANY
condition-2
TRUE
FALSE

\{NOT\}\{identifier-5
literal-5
arithmetic-expression-3\}\{\begin{array}{c}
\text{THROUGH} \\
\text{THRU}
\end{array}\}\{identifier-6
literal-6
arithmetic-expression-4\}\}\}\}

\begin{array}{l}
\text{imperative-statement-1} \\
\text{imperative-statement-2}
\end{array}

[END-EVALUATE]
EXIT
EXIT PROGRAM

FETCH database-record
[FOR UPDATE]

\begin{array}{c}
\text{RETLAINING}\{\begin{array}{c}
\text{REALM} \\
\text{RECORD}
\end{array}\{\begin{array}{c}
\text{SET} [set-name] \\
\{set-name\} \\
\{set-name\} \\
\{set-name\}
\end{array}\}\text{CURRENCY}\}
\end{array}

\begin{array}{l}
\{\\begin{array}{c}
\text{AT END stment}\|\text{NOT AT END stment2}
\end{array}\}
\{\\begin{array}{c}
\text{ON ERROR stment}\|\text{NOT ON ERROR stment2}
\end{array}\}
\end{array}

[END-FETCH]
FIND database-record

[FOR UPDATE]

RETAINING

RECORD

{set-name} ...

CURRENCY

[CURLY BRACKET EXPRESSIONS]

{[AT END stmt] [NOT AT END stmt2]}

{[ON ERROR stmt] [NOT ON ERROR stmt2]}

END-FIND

FIND ALL keeplist-name [record-name] [WITHIN {realm-name}] [set-name]

[FOR UPDATE]

USING {rec-key} ...

WHERE {bool-expres}

{[AT END stmt] [NOT AT END stmt2]}

{[ON ERROR stmt] [NOT ON ERROR stmt2]}

END-FIND

FREE {database-key-id

ALL [FROM {keeplist-name} ...] }

[FOR UPDATE]

ON ERROR stmt]

NOT ON ERROR stmt2]

END-FREE

END-FIND

FREE {database-key-id

ALL [FROM {keeplist-name} ...] }

[FOR UPDATE]

ON ERROR stmt]

NOT ON ERROR stmt2]

END-FREE
GENERATE {data-name-1
report-name-1}

GET record-name
{record-item} ... [ON ERROR stmt1]
[NOT ON ERROR stmt2]
END-GET

GO TO [procedure-name-1]
GO TO {procedure-name-1} ... DEPENDING ON identifier-1

IF condition-1 THEN {statement-1} ...
ELSE {statement-2} ...
NEXT SENTENCE
END-IF

INITIALIZE {identifier-1} ...

REPLACING { ALPHABETIC
ALPHANUMERIC
NUMERIC
ALPHANUMERIC-EDITED
NUMERIC-EDITED
BBCS
EGCS
}
DATA BY {identifier-2}
literal-1} ...
INITIATE \{report-name-1\} ...

INSPECT \textit{identifier-1} TALLYING
\[
\{ \text{identifier-2} \text{ FOR } \{ \text{CHARACTERS } \{ \text{BEFORE} \text{ AFTER } \} \text{ INITIAL } \{ \text{identifier-4} \text{ literal-2 } \} \ldots \} \ldots \}
\]

INSPECT \textit{identifier-1} REPLACING
\[
\{ \text{ALL} \text{ LEADING} \} \{ \text{identifier-3} \} \text{ BY } \{ \text{CHARACTERS } \{ \text{BEFORE} \text{ AFTER } \} \text{ INITIAL } \{ \text{identifier-4} \text{ literal-2 } \} \ldots \} \ldots \}
\]

INSPECT \textit{identifier-1} TALLYING
\[
\{ \text{identifier-2} \text{ FOR } \{ \text{CHARACTERS } \{ \text{BEFORE} \text{ AFTER } \} \text{ INITIAL } \{ \text{identifier-4} \text{ literal-2 } \} \ldots \} \ldots \}
\]
REPLACING

\[
\begin{align*}
\text{CHARACTERS} & \quad \text{BY} \quad \{ \text{identifier-5} \} \quad \{ \text{BEFORE} \} \quad \{ \text{AFTER} \} \quad \text{INITIAL} \quad \{ \text{identifier-4} \} \quad \text{...} \\
\text{ALL} & \quad \{ \text{identifier-3} \} \quad \text{BY} \quad \{ \text{identifier-5} \} \quad \{ \text{BEFORE} \} \quad \{ \text{AFTER} \} \quad \text{INITIAL} \quad \{ \text{identifier-4} \} \quad \text{...} \quad \text{...} \\
\text{LEADING} & \quad \{ \text{identifier-1} \} \quad \text{BY} \quad \{ \text{identifier-3} \} \quad \{ \text{BEFORE} \} \quad \{ \text{AFTER} \} \quad \text{INITIAL} \quad \{ \text{identifier-4} \} \quad \text{...} \quad \text{...} \\
\text{FIRST} & \quad \{ \text{identifier-3} \} \quad \text{BY} \quad \{ \text{identifier-5} \} \quad \{ \text{BEFORE} \} \quad \{ \text{AFTER} \} \quad \text{INITIAL} \quad \{ \text{identifier-4} \} \quad \text{...} \quad \text{...}
\end{align*}
\]

INSPECT \text{identifier-1 CONVERTING} \{ \text{identifier-6} \} \text{ TO } \{ \text{identifier-7} \}

\[
\begin{align*}
\{ \text{BEFORE} \} \quad \text{INITIAL} \quad \{ \text{identifier-4} \} & \quad \text{...} \\
\{ \text{AFTER} \} & \quad \text{...}
\end{align*}
\]

KEEP \{ \text{database-key-id} \} \text{ USING destination-keeplist}

\[
\begin{align*}
\text{ON ERROR imperative statement-1} \\
\text{NOT ON ERROR imperative statement-2} \\
\text{END-KEEP}
\end{align*}
\]

MERGE \text{file-name-1} \text{ ON } \{ \text{ASCENDING} \} \text{ DESCENDING KEY } \{ \text{data-name-1} \} \text{...} \quad \text{...}

\[
\begin{align*}
\text{COLLECTING SEQUENCE IS } \text{alphabet-name-1} \\
\text{USING } \text{file-name-2} \{ \text{file-name-3} \} \text{...}
\end{align*}
\]

\[
\begin{align*}
\text{OUTPUT PROCEDURE IS } & \text{procedure-name-1} \quad \{ \text{THROUGH} \} \quad \text{procedure-name-2} \\
\text{GIVING } & \{ \text{file-name-4} \} \text{...}
\end{align*}
\]
MODIFY [record-name
(record-item} ... ]

[RETAINING [REALM
RECORD
[SET [set-name] ...
{set-name} ... ]]
CURRENCY]

[ON ERROR stment]
[NOT ON ERROR stment2]
[END-MODIFY]

MOVE {identifier-1} TO {identifier-2} ...

MOVE {CORRESPONDING CORR} identifier-1 TO identifier-2

MULTIPLY {identifier-1} BY {identifier-2 [ROUNDED]} ...

[ON SIZE ERROR imperative-statement-1]
[NOT ON SIZE ERROR imperative-statement-2]
[END-MULTIPLY]

MULTIPLY {identifier-1} BY {identifier-2}
GIVING \{\text{identifier-3 [ROUNDED]}\} ... \\
[\text{ON SIZE ERROR} \text{ imperative-statement-1}] \\
[\text{NOT ON SIZE ERROR} \text{ imperative-statement-2}] \\
[\text{END-MULTIPLY}] \\

\text{S OPEN} \\
S \text{ OPEN} \\
\begin{align*}
\{ & \text{INPUT \{file-name-1 [WITH NO REWIND]\} ...} \\
& \text{OUTPUT \{file-name-2 [WITH NO REWIND]\} ...} \\
& \begin{cases}
\begin{align*}
\text{ALLOWING} \\
& \{ \text{NO OTHERS} \\
& \text{READERS} \\
& \text{WRITERS} \\
& \text{UPDATERS} \\
& \text{ALL} \\
\end{cases}
\end{align*}
\end{cases}
\begin{align*}
& \text{I-O \{file-name-3\} ...} \\
& \text{EXTEND \{file-name-4\} ...}
\end{align*}
\end{align*}

\text{RI OPEN} \\
\text{RI OPEN} \\
\begin{align*}
\{ & \text{INPUT \{file-name-1\} ...} \\
& \text{OUTPUT \{file-name-2\} ...} \\
& \text{ALLOWING} \\
& \text{\begin{cases}
\begin{align*}
& \{ \text{NO OTHERS} \\
& \text{READERS} \\
& \text{WRITERS} \\
& \text{UPDATERS} \\
& \text{ALL} \\
\end{cases}
\end{align*}
\end{cases}
\end{align*}
\begin{align*}
& \text{I-O \{file-name-3\} ...} \\
& \text{EXTEND \{file-name-4\} ...}
\end{align*}
\end{align*}
W OPEN {OUTPUT {file-name-1 [WITH NO REWIND] ... } ... }

EXTEND {file-name-2} ...

PERFORM [procedure-name-1 {THROUGH THRU} procedure-name-2]

[imperative-statement-1 END-PERFORM]

PERFORM [procedure-name-1 {THROUGH THRU} procedure-name-2]

{identifier-1} TIMES [imperative-statement-1 END-PERFORM]

PERFORM [procedure-name-1 {THROUGH THRU} procedure-name-2]

[WITH TEST {BEFORE AFTER} UNTIL condition-1]

[imperative-statement-1 END-PERFORM]

PERFORM [procedure-name-1 {THROUGH THRU} procedure-name-2]

[WITH TEST {BEFORE AFTER}]

VARYING {identifier-2 index-name-1} FROM {identifier-3 index-name-2 literal-1}
BY \{identifier-4\} UNTIL condition-1

[AFTER \{identifier-5\} FROM \{identifier-6\}]

BY \{identifier-7\} UNTIL condition-2] ...
RECEIVE cd-name-1 \{MESSAGE\} INTO identifier-1

[NO DATA imperative-statement-1]
[WITH DATA imperative-statement-2]
[END-RECEIVE]

RECONNECT [record-name] WITHIN \{set-name} ... \}

RETAINING \{REALM RECORD \{SET [set-name] ...\}\} CURRENCY

[ON ERROR stment]
[NOT ON ERROR stment2]
[END-RECONNECT]

RELEASE record-name-1 [FROM identifier-1]
RETURN file-name-1 RECORD [INTO identifier-1]
AT END imperative-statement-1
[NOT AT END imperative-statement-2]
[END-RETURN]

S REWRITE record-name-1 [FROM identifier-1]
RI REWRITE record-name-1 [FROM identifier-1]
[ALLOWING NO OTHERS]
[INVALID KEY imperative-statement-1]
[NOT INVALID KEY imperative-statement-2]
END-REWRITE

ROLLBACK
[ON ERROR stment]
[NOT ON ERROR stment2]
END-ROLLBACK

SEARCH identifier-1 [VARYING {identifier-2 index-name-1}]
[AT END imperative-statement-1]
{WHEN condition-1 {imperative-statement-2bildense}{NEXT SENTENCE}} ...
[END-SEARCH]

SEARCH ALL identifier-1 [AT END imperative-statement-1]

WHEN
{data-name-1 {IS EQUAL TO} identifier-3
{IS =} literal-1
{arithmetic-expression-1}}
condition-name-1

AND
{data-name-2 {IS EQUAL TO} identifier-4
{IS =} literal-2
{arithmetic-expression-2}}]
condition-name-2

...
\begin{verbatim}
\{imperative-statement-2\}
\} \{NEXT SENTENCE\}
\{END-SEARCH\}
SEND cd-name-1 FROM identifier-1
SEND cd-name-1 [FROM identifier-1]
\{WITH identifier-2\}
 \{WITH \texttt{ESI}\}
 \{WITH \texttt{EMI}\}
 \{WITH \texttt{EGI}\}
\{BEFORE\} ADVANCING \{identifier-3\} \{integer-1\} \{LINE\} \{LINES\}
\{AFTER\} \{mnemonic-name-1\} \{PAGE\}
\{REPLACING\} LINE
\begin{verbatim}
SET \{index-name-1\} \{identifier-1\} \ldots \{index-name-2\} \{identifier-2\} \{integer-1\}
\end{verbatim}
\begin{verbatim}
SET \{index-name-3\} \{UP BY\} \{DOWN\} \{integer-2\}
\end{verbatim}
\begin{verbatim}
SET \{mntemonic-name-1\} \ldots \{ON\} \{OFF\} \ldots
\end{verbatim}
\end{verbatim}
SET \{condition-name-1\} ... TO TRUE
SET pointer-id TO REFERENCE OF identifier
SET status-code-id TO \{SUCCESS, FAILURE\}

SET \{identifier
ADDRESS OF identifier\} TO \{identifier
ADDRESS OF identifier
NULL
NULLS\}

SORT file-name-1 \{ON ASCENDING DESCENDING\} KEY \{data-name-1\} ... \} ...

\[\text{WITH DUPLICATES IN ORDER}\]
\[\text{COLLATING SEQUENCE IS alphabet-name-1}\]
\{INPUT PROCEDURE IS procedure-name-1 \{THROUGH
THRU\} procedure-name-2\}
USING \{file-name-2\} ...

\{OUTPUT PROCEDURE IS procedure-name-3 \{THROUGH
THRU\} procedure-name-4\}
GIVING \{file-name-3\} ...
START file-name-1

KEY

{ IS EQUAL TO
  IS =
  IS GREATER THAN
  IS >
  IS NOT LESS THAN
  IS NOT <
  IS GREATER THAN OR EQUAL TO
  IS >=
}
data-name-1

REGARDLESS OF LOCK
ALLOWING
{ UPDATERS
  READERS
  NO OTHERS
}

[ INVALID KEY imperative-statement-1]

[ NOT INVALID KEY imperative-statement-2]

END-START

STOP { RUN literal-1 }

STORE record-name [ { NEXT TO DBKEY } WITHIN { realm-name } ... ]

RETAINING

[ REALM
  RECORD
  SET [ set-name ] ...
  { set-name } ...
]

CURRENCY
[ON ERROR statement]
[NOT ON ERROR statement]
[END-STORE]

STRING \{ identifier-1 \}
\{ literal-1 \}
... DELIMITED BY \{ identifier-2 \}
\{ literal-2 \}

INTO identifier-3
[WITH POINTER identifier-4]
[ON OVERFLOW imperative-statement-1]
[NOT ON OVERFLOW imperative-statement-2]
[END-STRING]

SUBTRACT \{ identifier-1 \}
\{ literal-1 \}
... FROM \{ identifier-3 \ [ROUNDED] \}
... [ON SIZE ERROR imperative-statement-1]
[NOT ON SIZE ERROR imperative-statement-2]
[END-SUBTRACT]

SUBTRACT \{ identifier-1 \}
\{ literal-1 \}
... FROM \{ identifier-2 \}
\{ literal-2 \}
GIVING \{ identifier-3 \ [ROUNDED] \}
... [ON SIZE ERROR imperative-statement-1]
[NOT ON SIZE ERROR imperative-statement-2]
[END-SUBTRACT]
SUBTRACT \{CORRESPONDING\} identifier-1 FROM identifier-2 [ROUNDED]
[ON SIZE ERROR imperative-statement-1]
[NOT ON SIZE ERROR imperative-statement-2]
[END-SUBTRACT]
SUPPRESS PRINTING
TERMINATE \{report-name-1\} ...
UNLOCK file-name \{RECORD ALL RECORDS\}
UNSTRING identifier-1
\[DELIMITED BY [ALL] \{identifier-2\} \[OR [ALL] \{identifier-3\}\] ... \]
INTO \{identifier-4 \{DELIMITER IN identifier-5\} \{COUNT IN identifier-6\}\} ... 
[WITH POINTER identifier-7]
[TALLYING IN identifier-8]
[ON OVERFLOW imperative-statement-1]
[NOT ON OVERFLOW imperative-statement-2]
[END-UNSTRING]
USE [GLOBAL] AFTER STANDARD \{EXCEPTION ERROR\} PROCEDURE ON \{file-name-1\} ...
INPUT OUTPUT I-O EXTEND
USE [GLOBAL] AFTER STANDARD {BEGINNING
END

FILE
REEL
UNIT
LABEL
PROCEDURE
ON
INPUT
OUTPUT
I-O
EXTEND

USE [GLOBAL] BEFORE REPORTING identifier-1

USE FOR DEBUGGING ON
cd-name-1
{ ALL REFERENCES OF } identifier-1
file-name-1
procedure-name-1
ALL PROCEDURES

USE [GLOBAL] FOR DB-EXCEPTION
ON
{DBM$exception-condition} ...

S WRITE record-name-1 [FROM identifier-1]
ALL ALLOWING NO OTHERS
General Format for Copy and Replace Statements

COPY text-name-1 \{ OF library-name-1 \}
COPY record-name FROM DICTIONARY

REPLACE {==pseudo-text-1==} BY {==pseudo-text-2==} ...

REPLACE OFF
General Format for Conditions

RELATION CONDITION

\[
\begin{align*}
\text{identifier-1} & \quad \text{literal-1} \\
\text{arithmetic-expression-1} & \quad \text{index-name-1} \\
\end{align*}
\]

\[
\begin{align*}
\text{IS} & \quad \text{NOT} \quad \text{GREATER \ THAN} \\
\text{IS} & \quad \text{NOT} \quad > \\
\text{IS} & \quad \text{NOT} \quad \text{LESS \ THAN} \\
\text{IS} & \quad \text{NOT} \quad < \\
\text{IS} & \quad \text{NOT} \quad \text{EQUAL \ TO} \\
\text{IS} & \quad \text{NOT} \quad = \\
\text{IS \ GREATER \ THAN \ OR \ EQUAL \ TO} \\
\text{IS} & \quad \geq \\
\text{IS \ LESS \ THAN \ OR \ EQUAL \ TO} \\
\text{IS} & \quad \leq
\end{align*}
\]

CLASS CONDITION

\[
\begin{align*}
\text{identifier-1} \quad \text{IS} \quad \text{NOT} \\
\end{align*}
\]

\[
\begin{align*}
\text{NUMERIC} \\
\text{ALPHABETIC} \\
\text{ALPHABETIC-LOWER} \\
\text{ALPHABETIC-UPPER} \\
\text{class-name}
\end{align*}
\]

CONDITION-NAME CONDITION

\[
\begin{align*}
\text{condition-name-1}
\end{align*}
\]
CURRENCY INDICATOR ACCESS

CURRENT WITHIN \{record-name
set-name
realm-name\}\]

KEEPLIST ACCESS

\{OFFSET integer-exp
FIRST LAST\} WITHIN keeplist-name

SWITCH-STATUS CONDITION

condition-name-1

SIGN CONDITION

arithmetic-expression-1 IS [NOT]\{POSITIVE
NEGATIVE
ZERO\}

TENANCY CONDITION

[NOT] [set-name]\{OWNER
MEMBER
TENANT\}

DATABASE KEY CONDITION

database-key IS [NOT]\{ALSO database-key
NULL
WITHIN keeplist-name\}
SUCCESS/FAILURE CONDITION

status-code-id IS \{SUCCESS\, FAILURE\}

NEGATED CONDITION

NOT condition-1

COMBINED CONDITION

condition-1 \{ AND \, OR \} condition-2 \ldots

ABBREVIATED COMBINED RELATION CONDITION

relation-condition \{ AND \, OR \} \{ NOT \, [relational-operator] \, object \} \ldots

DATABASE KEY IDENTIFIER ACCESS

database-key-identifier

DATABASE SET OWNER ACCESS

OWNER WITHIN set-name
RECORD SEARCH ACCESS

\{FIRST
  LAST
  NEXT
  PRIOR
  ANY
  DUPLICATE
  [RELATIVE\ int-exp] \}

\[record-name] \{WITHIN \{realm-name\}\} \{USING \{record-key\} \...\} \{WHERE \{boolean-expression\}\}

boolean-express:
\{boolean-alt | OR boolean-alt| \...\} \}

boolean-alt:
\{simple-boolean-relation | AND simple-boolean-relation| \...\} \}

simple-boolean-relation:
\{boolean-condition \}
\{NOT boolean-expression\} \}
boolean-condition:

\[
\begin{aligned}
&\text{IS} \begin{cases}
\text{GREATER THAN OR EQUAL TO} \\
\text{LESS THAN OR EQUAL TO}
\end{cases} \\
&\text{IS [NOT]} \begin{cases}
\text{EQUAL TO} \\
\text{GREATER THAN} \\
\text{LESS THAN}
\end{cases} \\
&\text{DOES [NOT]} \begin{cases}
\text{CONTAIN} \\
\text{CONTAINS} \\
\text{MATCH} \\
\text{MATCHES}
\end{cases}
\end{aligned}
\]

DATABASE ON ERROR CONDITION

[NOT] ON ERROR imperative statement
QUALIFICATION

FORMAT 1
{\{ data-name-1 \} \{ condition-name \} } {\{ IN \} \{ data-name-2 \} ... } {\{ IN \} \{ file-name \} } {\{ IN \} \{ cd-name \} }

FORMAT 2
paragraph-name {\{ IN \} \{ section-name \} }

FORMAT 3
text-name {\{ IN \} \{ library-name \} }
FORMAT 4
LINAGE-COUNTER \{IN \ OF \} report-name

FORMAT 5
\{PAGE-COUNTER \ OF \} \{IN \ OF \} report-name

FORMAT 6
\{IN \ OF \} \{IN \ OF \} report-name
\{IN \ OF \} report-name

Miscellaneous Formats

SUBSCRIPTING
\{condition-name-1\} ( \{integer-1 \data-name-2 \{\{ \pm \ integer-2\} \index-name-2 \{\{ \pm \ integer-3\} \arithmetic-expression \} \ldots \} \data-name-1\} \}
REFERENCE MODIFICATION

data-name-1 (leftmost-character-position:[length])

IDENTIFIER

data-name-1 [{IN} data-name-2] ... [{IN} {cd-name file-name report-name}]

[({subscript} ... )] [(leftmost-character-position: [length])]

---

**General Format for Nested Source Programs**

**IDENTIFICATION DIVISION.**
**PROGRAM-ID.** program-name-1 [IS INITIAL PROGRAM].
**ENVIRONMENT DIVISION.** environment-division-content
**DATA DIVISION.** data-division-content
**PROCEDURE DIVISION.** procedure-division-content
[[nested-source-program] ... ]
**END PROGRAM** program-name-1.
General Format for Nested-Source-Program

IDENTIFICATION DIVISION.

PROGRAM-ID. program-name-2 [IS {COMMON INITIAL} PROGRAM].

[ENVIRONMENT DIVISION. environment-division-content]

[DATA DIVISION. data-division-content]

[PROCEDURE DIVISION. procedure-division-content]

[nested-source-program] ...

END PROGRAM program-name-2.

General Format for a Sequence of Source Programs

{IDENTIFICATION DIVISION.

PROGRAM-ID. program-name-3 [IS INITIAL PROGRAM].

ENVIRONMENT DIVISION. environment-division-content]

DATA DIVISION. data-division-content]

PROCEDURE DIVISION. procedure-division-content]

[nested-source-program] ...

END PROGRAM program-name-3.} ...

IDENTIFICATION DIVISION.

PROGRAM-ID. program-name-4 [IS INITIAL PROGRAM].

ENVIRONMENT DIVISION. environment-division-content]
IV. FUNCTION NAMES AVAILABLE IN EXTENSIONS TO COBOL 85

<table>
<thead>
<tr>
<th>Function</th>
<th>Data Type</th>
<th>Other Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABS</td>
<td>INTEGER</td>
<td>ORD</td>
</tr>
<tr>
<td>ACOS</td>
<td>INTEGER-OF-DATE</td>
<td>ORD-MAX</td>
</tr>
<tr>
<td>ANNUITY</td>
<td>INTEGER-OF-DAY</td>
<td>ORD-MIN</td>
</tr>
<tr>
<td>ASIN</td>
<td>INTEGER-PART</td>
<td>PI</td>
</tr>
<tr>
<td>ATAN</td>
<td>LENGTH</td>
<td>PRESENT-VALUE</td>
</tr>
<tr>
<td>CHAR</td>
<td>LENGTH-AN</td>
<td>RANDOM</td>
</tr>
<tr>
<td>CHAR-NATIONAL</td>
<td>LOG</td>
<td>RANGE</td>
</tr>
<tr>
<td>COS</td>
<td>LOG10</td>
<td>REM</td>
</tr>
<tr>
<td>CURRENT-DATE</td>
<td>LOWER-CASE</td>
<td>REVERSE</td>
</tr>
<tr>
<td>DATE-OF-INTEGER</td>
<td>MAX</td>
<td>SIGN</td>
</tr>
<tr>
<td>DAY-OF-INTEGER</td>
<td>MEAN</td>
<td>SIN</td>
</tr>
<tr>
<td>DISPLAY-OF</td>
<td>MEDIAN</td>
<td>SQRT</td>
</tr>
<tr>
<td>EXCEPTION-FILE</td>
<td>MIDRANGE</td>
<td>STANDARD-DEVIAITION</td>
</tr>
<tr>
<td>EXCEPTION-LOCATION</td>
<td>MIN</td>
<td>SUM</td>
</tr>
<tr>
<td>EXCEPTION-STATEMENT</td>
<td>MOD</td>
<td>TAN</td>
</tr>
<tr>
<td>EXCEPTION-STATUS</td>
<td>NATIONAL-OF</td>
<td>UPPER-CASE</td>
</tr>
<tr>
<td>EXP</td>
<td>NUMVAL</td>
<td>VARIANCE</td>
</tr>
<tr>
<td>FACTORIAL</td>
<td>NUMVAL-C</td>
<td>WHEN-COMPiled</td>
</tr>
<tr>
<td>FRACTION-PART</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## V. NEW COBOL 9X RESERVED WORDS

<table>
<thead>
<tr>
<th>ALIGN</th>
<th>INHERITS</th>
<th>PROPERTY</th>
</tr>
</thead>
<tbody>
<tr>
<td>B-AND</td>
<td>INTERFACE</td>
<td>RAISE</td>
</tr>
<tr>
<td>B-NOT</td>
<td>INTERFACE-ID</td>
<td>REPOSITORY</td>
</tr>
<tr>
<td>B-OR</td>
<td>INVARIANT</td>
<td>RESERVED</td>
</tr>
<tr>
<td>B-XOR</td>
<td>INVOKE</td>
<td>RETURNING</td>
</tr>
<tr>
<td>CLASS-ID</td>
<td>METHOD</td>
<td>REUSES</td>
</tr>
<tr>
<td>CONFORMING</td>
<td>METHOD-ID</td>
<td>SELF</td>
</tr>
<tr>
<td>END-INVoke</td>
<td>NATIONAL</td>
<td>SUPER</td>
</tr>
<tr>
<td>EXCEPTION-OBJECT</td>
<td>NATIONAL-EDITED</td>
<td>SYSTEM-OBJECT</td>
</tr>
<tr>
<td>FACTORY</td>
<td>OBJECT</td>
<td>UNIVERSAL</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>OVERRIDE</td>
<td></td>
</tr>
</tbody>
</table>