OpenCOBOL 1.1
[06FEB2009 Version]
Textpad Integration

5 April 2010

OpenCOBOL is copyrighted by:
Keisuke Nishida
Roger While

Distribution and documentation prepared by:
Gary Cutler
CutlerGL@gmail.com

Textpad is a product of Helios Software Solutions
http://www.textpad.com

Contents

Introduction

OpenCOBOL is a COBOL compiler and runtime environment. The OpenCOBOL compiler generates C code which is automatically compiled and linked by the GNU C compiler (gcc) and the GNU linker/loader (ld).

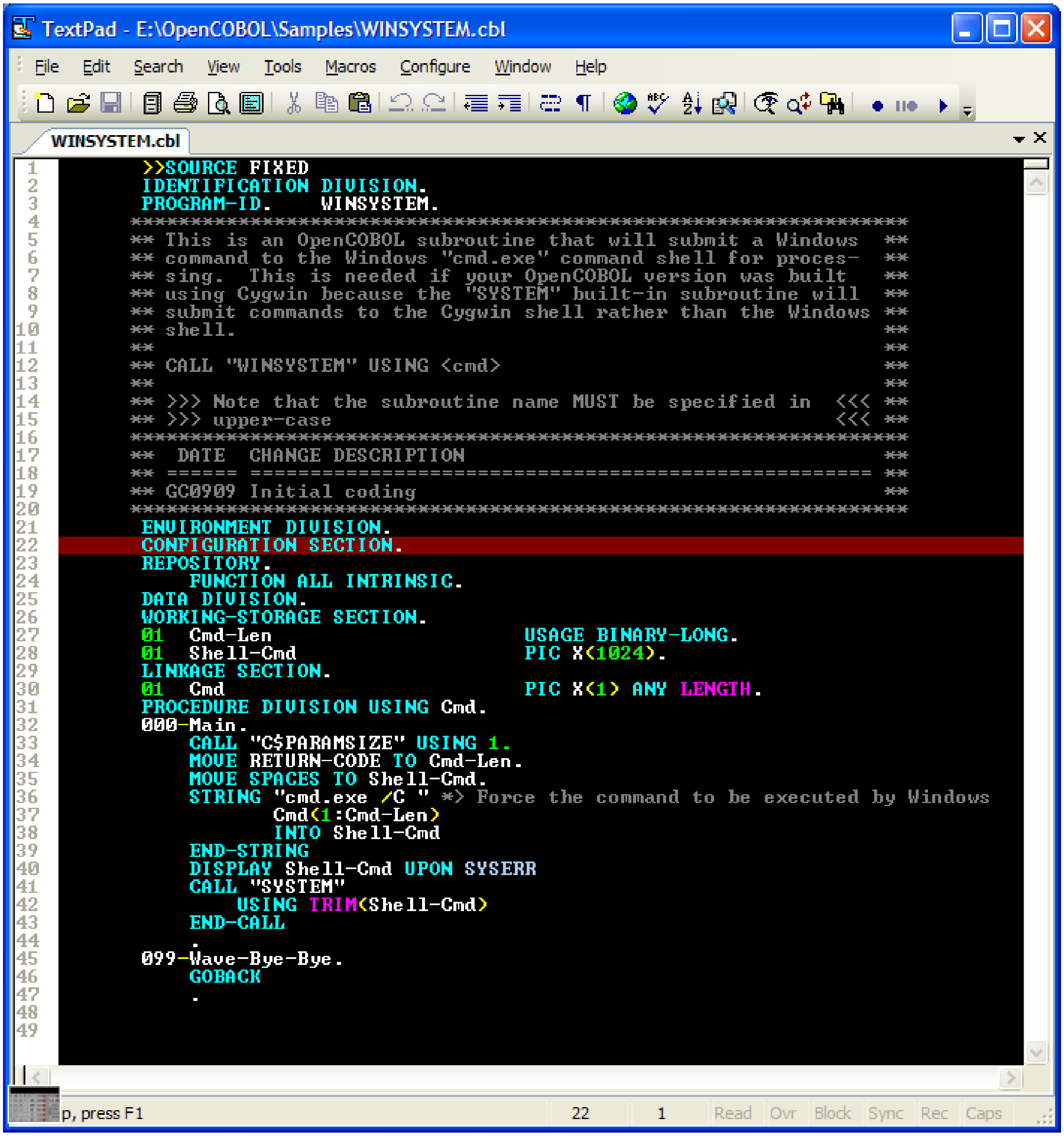
The Textpad editor, from Helios Software, is an amazingly powerful tool to use as an editor. With just a little bit of tweaking, you can turn Textpad into a powerful Integrated Development Environment (IDE) for OpenCOBOL.

In addition to providing a powerful text-editing solution capable of editing multiple files concurrently in a series of tabbed windows, Textpad can also perform syntax recognition against anything being edited and "colorize" that file according to the appropriate syntax parsing rules. A sample screenshot of this syntax highlighting at work is shown in Figure 1.

In addition to syntax highlighting, Textpad can be configured to provide any number of additional "tools" – Windows commands that can be executed against the OpenCOBOL program being edited in the current tab. This document will show you how to add such tools as:

- **Command Window** – opening a Windows command window (cmd.exe) with the folder in which the edited program resides being pre-selected as the current directory.
- **Compile OpenCOBOL Program** – running the OpenCOBOL compiler (cobc) against the current program.

Figure 1 - Textpad Screenshot Showing OpenCOBOL Syntax Highlighting



```
       >>SOURCE FIXED
       IDENTIFICATION DIVISION.
       PROGRAM-ID.    WINSYSTEM.
      *****************************************************************
      ** This is an OpenCOBOL subroutine that will submit a Windows  **
      ** command to the Windows "cmd.exe" command shell for proces-  **
      ** sing.  This is needed if your OpenCOBOL version was built   **
      ** using Cygwin because the "SYSTEM" built-in subroutine will  **
      ** submit commands to the Cygwin shell rather than the Windows **
      ** shell.                                                      **
      **                                                             **
      ** CALL "WINSYSTEM" USING <cmd>                                **
      **                                                             **
      ** >>> Note that the subroutine name MUST be specified in  <<< **
      ** >>> upper-case                                          <<< **
      *****************************************************************
      **  DATE  CHANGE DESCRIPTION                                   **
      ** ====== ==================================================== **
      ** GC0909 Initial coding                                       **
      *****************************************************************
       ENVIRONMENT DIVISION.
       CONFIGURATION SECTION.
       REPOSITORY.
           FUNCTION ALL INTRINSIC.
       DATA DIVISION.
       WORKING-STORAGE SECTION.
       01  Cmd-Len                            USAGE BINARY-LONG.
       01  Shell-Cmd                          PIC X(1024).
       LINKAGE SECTION.
       01  Cmd                                PIC X(1) ANY LENGTH.
       PROCEDURE DIVISION USING Cmd.
       000-Main.
           CALL "C$PARAMSIZE" USING 1.
           MOVE RETURN-CODE TO Cmd-Len.
           MOVE SPACES TO Shell-Cmd.
           STRING "cmd.exe /C " *> Force the command to be executed by Windows
                  Cmd(1:Cmd-Len)
                  INTO Shell-Cmd
           END-STRING
           DISPLAY Shell-Cmd UPON SYSERR
           CALL "SYSTEM"
               USING TRIM(Shell-Cmd)
           END-CALL
           .
       099-Wave-Bye-Bye.
           GOBACK
           .
```

Configuring Textpad for OpenCOBOL Syntax Highlighting

Once you have downloaded and installed Textpad (don't forget to pay for it – it's WELL WORTH THE NOMINAL PRICE) from the Textpad website (www.textpad.com), perform the following steps:

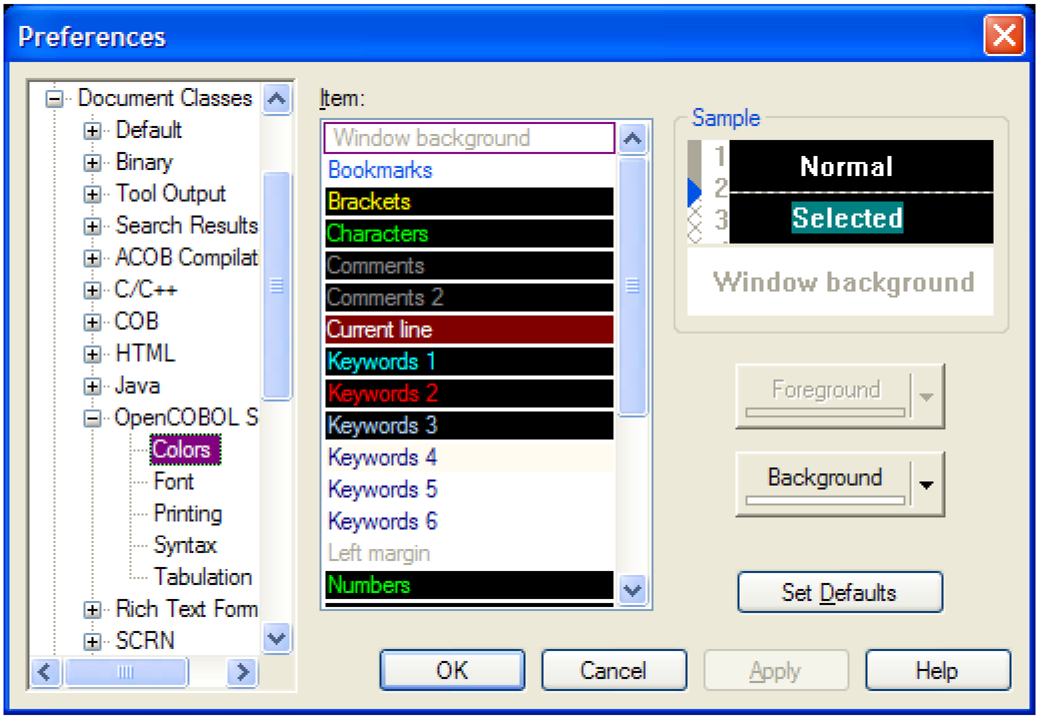
1. Open Textpad and execute its "**Configure** / **Preferences** / **Folders**" command. Identify the folder associated with "Syntax".

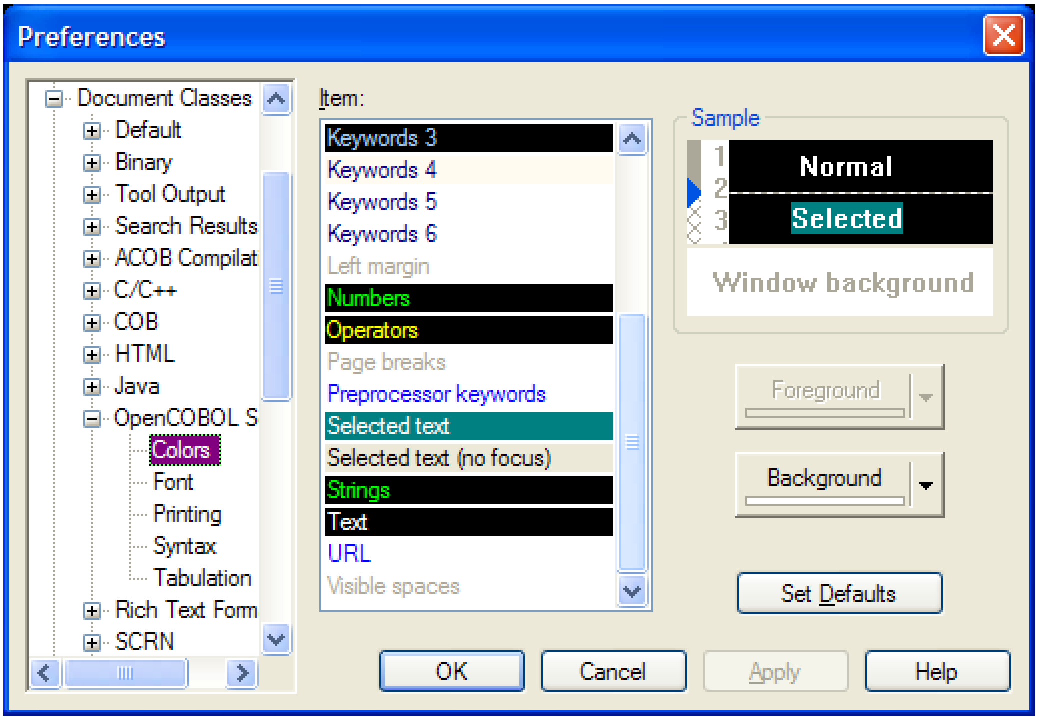
2. Use Windows Explorer to locate and open the folder identified in #1 above and create a new (empty) text file named "opencobol.syn".
3. Back in Textpad, create a new document class specifically for OpenCOBOL ("**OpenCOBOL Source**"). To do this…
   a. Execute the "**Configure / New Document Class…**" command.
   b. Specify "**OpenCOBOL Source**" as the document class name, then press the "Next" button.
   c. Specify "***.cbl,*.cpy**" as the two extensions that will be associated with this document class, then press the "Next" button.
   d. Click the "Enable syntax highlighting" switch and select "opencobol.syn" from the pull-down list of available syntax files, then press the "Next" button.
   e. Click the "Finish" button to register the new document class with Textpad.
   f. Associate this new document class with ".cbl" and ".cpy" extensions.
4. Now you need to actually define the OpenCOBOL syntax highlighting rules by populating the "opencobol.sys" file (that is empty right now) with the values found in the "
5. " section of this document, as follows:
   a. Execute the "**Configure / Preferences** " Textpad menu command.
   b. Open the "Document Classes" list by clicking on the "+" icon.
   c. Open the "OpenCOBOL Source" document class by clicking on its "+" icon.
   d. Single-click the "Syntax" element under the document class.
   e. In the "Other characters in words" field, specify the characters "**-_**" (hyphen and underscore) to indicate that both of these characters may be used when forming words in OpenCOBOL.
   f. The "Syntax definition file" should have "opencobol.syn" in it. Click the "Open" button next to it, then click the "OK" button. The contents of that file (there aren't any yet) will be displayed.
   g. Select the opencobol.syn file contents found in the "The opencobol.syn File" section of this document, copy them to the clipboard and paste them into the opencobol.syn file in Textpad.
   h. Close Textpad, specifying "Save" when prompted by "Save changes to opencobol.syn?".

At this point, you've "taught" OpenCOBOL about a new class of documents and you've defined how it can parse those documents to recognize syntactical elements in those documents. There are two tasks remaining to complete the job of fully "training" Textpad.

6. Next, you need to tell Windows that Textpad should be used as the default program with which "*.cbl" and "*.cpy" programs should be opened when double-clicked from Windows Explorer. This can (conveniently) accomplished from within Textpad itself, as follows:
    a. Start Textpad (if it isn't already running) and execute the "**Configure** / **Preferences**" command.
    b. Select "Associated Files" from the list.
    c. In the "New Extension" field, specify "**.cbl**" (don't use an asterisk); in the "Description" field, specify "**OpenCOBOL Program**"; click the "Add" button.
    d. Repeat c for an extension of "**.cpy**" and a description of "**OpenCOBOL Copybook**".
    e. Click "OK".
    f. Click the "New document" tool on the Textpad toolbar (it looks like a blank sheet of paper) and paste in the sample program found in this document in the "A Sample OpenCOBOL Program" section.
    g. Close Textpad and save the sample OpenCOBOL program to the location of your choice.
    h. Test your file association by double-clicking the sample OpenCOBOL program file.
7. Finally, you need to specify the colors to use for syntax highlighting. You'll find this MUCH easier if you are actually editing an OpenCOBOL program – why not use the one you created in step 6.g above?
    a. Once again, execute Textpad's "**Configure** / **Preferences**" command.
    b. Open the "Document Classes" list by clicking on the "+" icon.
    c. Open the "OpenCOBOL Source" document class by clicking on its "+" icon.
    d. Single-click the "Colors" element under the document class.
    e. Specify colors you wish to use for the various syntactical elements of OpenCOBOL source code by single-clicking an element,
    f. See Figure 2 for examples of how I set up Textpad to produce the screenshot shown in Figure 1.

Figure 2 - Sample Textpad Color Settings for "OpenCOBOL Source" Colors





Adding Some Useful OpenCOBOL Tools

Two tools you will find useful for OpenCOBOL programs are:

- A tool to open a Windows command shell window in the directory where the file currently being edited resides. Actually, this may be useful for ANY type of file!
- A tool to compile an OpenCOBOL program.

To add a command window tool:

1. Start Textpad (if it isn't already running) and execute the "**Configure** / **Preferences**" command.
2. Open the Tools list by pressing the "+" icon.
3. Click the "Tools" entry on the list.
4. Click the "Add" button and select the "DOS Command" choice.
5. Enter the command "**start "Textpad Command Window" cmd.exe**" in the DOS Command window. Press the OK button.
6. Press the 'Apply" button.
7. Single-click the new tools entry in the list in the middle of the preferences window. You may now change the tools caption to whatever you wish – "**Command Window**", for example. Click the "Apply" button twice – once to close the caption edit and a second time to actually rename the tool.
8. Now single-click the new tool in the list of tools on the left part of the preference screen (under the "Tools" header) and make the following changes to its definition:
    - Make sure "**$FileDir**" is in the "Initial folder" field
    - Make sure only the "Close DOS window on exit" option is checked
9. Click the "Apply" button.
10. Click the "OK" button.

To add an OpenCOBOL compiler tool:

1. Start Textpad (if it isn't already running) and execute the "**Configure** / **Preferences**" command.
2. Open the Tools list by pressing the "+" icon.
3. Click the "Tools" entry on the list.
4. Click the "Add" button and select the "Program" choice.
5. Use the "Select File" dialog that appears to locate the folder where your OpenCOBOL compiler resides (probably C:\OpenCOBOL\bin). Select the "cobc" program and press the OK button.
6. Press the 'Apply" button.

7. Single-click the new tools entry in the list in the middle of the preferences window. You may now change the tools caption to whatever you wish – “**Command Window**”, for example. Click the “Apply” button twice – once to close the caption edit and a second time to actually rename the tool.
8. Now single-click the new tool in the list of tools on the left part of the preference screen (under the “Tools” header) and make the following changes to its definition:

    - Edit the “Parameters” field to include the command-line arguments you would most-frequently use with the cobc command, using the string “**$File**” at the point where you would specify the program source file name.
    - Make sure “**$FileDir**” is in the “Initial folder” field
    - Make sure only the “Prompt for parameters” and “Capture output” options are checked
    - Insert the following string (without the leading and trailing quote characters) into the “Regular expression to match output” field:

      “**^\(.+\):\([0-9]+\):**”
    - Set the “File” pull-down to a value of “1”, the “Line” pull-down to a value of “2” and the “Column” pull-down to an empty value.
9. Click the “Apply” button.
10. Click the “OK” button.

When you use this tool, you’ll be prompted for any parameters to supply the compiler (the ones you specified in #8 above will already be present – you may edit them as you require for the compilation). A “Tool Output” window will be created and any compilation warning and/or error messages will be displayed there. If you then double-click on one of the error/warning messages, you’ll be taken back to the program window and the current line will be positioned to the line containing the error/warning!

The opencobol.syn File

The following is a complete listing of the opencobol.syn file, appropriate for use with the latest version of Textpad.

```
; Syntax file for OpenCOBOL V1.1 [06FEB2009]

C=1

[Syntax]
Namespace1 = 6
IgnoreCase = Yes
InitKeyWordChars = ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
KeyWordChars = ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789-
BracketChars=()
OperatorChars=><=+/*-
PreprocStart =
SyntaxStart =
SyntaxEnd =
CommentStart =
CommentEnd =
CommentStartAlt =
CommentEndAlt =
SingleComment = *
SingleCommentCol = 7
SingleCommentAlt = *>
SingleCommentColAlt = 0
StringStart = "
StringEnd = "
StringAlt =
StringEsc =
CharStart = '
CharEnd = '
CharEsc =

[Preprocessor keywords]

[Keywords 1]

ACCEPT
ACCESS
ADD
ADDRESS
ADVANCING
AFTER
ALL
ALLOCATE
ALPHABET
ALPHABETIC
ALPHABETIC-LOWER
ALPHABETIC-UPPER
ALPHANUMERIC
ALPHANUMERIC-EDITED
ALSO
ALTER
ALTERNATE
AND
ANY
ARE
AREA
AREAS
ARGUMENT-NUMBER
ARGUMENT-VALUE
AS
ASCENDING
ASSIGN
AT
AUTO
AUTO-SKIP
AUTOMATIC
AUTOTERMINATE
BACKGROUND-COLOR
BASED
BEEP
BEFORE
BELL
BINARY
BINARY-C-LONG
BINARY-CHAR
BINARY-DOUBLE
BINARY-LONG
BINARY-SHORT
BLANK
BLINK
BLOCK
BOTTOM
BY
BYTE-LENGTH
CALL
```

```
CANCEL
CF
CH
CHAINING
CHARACTER
CHARACTERS
CLASS
CLOSE
CODE
CODE-SET
COL
COLLATING
COLS
COLUMN
COLUMNS
COMMA
COMMAND-LINE
COMMIT
COMMON
COMP
COMP-1
COMP-2
COMP-3
COMP-4
COMP-5
COMP-X
COMPUTATIONAL
COMPUTATIONAL-1
COMPUTATIONAL-2
COMPUTATIONAL-3
COMPUTATIONAL-4
COMPUTATIONAL-5
COMPUTATIONAL-X
COMPUTE
CONFIGURATION
CONSTANT
CONTAINS
CONTENT
CONTINUE
CONTROL
CONTROLS
CONVERTING
COPY
CORR
CORRESPONDING
COUNT
CRT
CURRENCY
CURSOR
CYCLE
DATA
DATE
DAY
DAY-OF-WEEK
DE
DEBUGGING
DECIMAL-POINT
DECLARATIVES
DEFAULT
DELETE
DELIMITED
DELIMITER
DEPENDING
DESCENDING
DETAIL
DISK
DISPLAY
DIVIDE
DIVISION
DOWN
DUPLICATES
```

```
DYNAMIC
EBCDIC
ELSE
END
END-ACCEPT
END-ADD
END-CALL
END-COMPUTE
END-DELETE
END-DISPLAY
END-DIVIDE
END-EVALUATE
END-IF
END-MULTIPLY
END-OF-PAGE
END-PERFORM
END-READ
END-RETURN
END-REWRITE
END-SEARCH
END-START
END-STRING
END-SUBTRACT
END-UNSTRING
END-WRITE
ENTRY
ENVIRONMENT
ENVIRONMENT-NAME
ENVIRONMENT-VALUE
EOL
EOP
EOS
EQUAL
EQUALS
ERASE
ERROR
ESCAPE
EVALUATE
EXCEPTION
EXCLUSIVE
EXIT
EXTEND
EXTERNAL
FALSE
FD
FILE
FILE-CONTROL
FILE-ID
FILLER
FINAL
FIRST
FLOAT-LONG
FLOAT-SHORT
FOOTING
FOR
FOREGROUND-COLOR
FOREVER
FREE
FROM
FULL
FUNCTION
FUNCTION-ID
GENERATE
GIVING
GLOBAL
GO
GOBACK
GREATER
GROUP
HEADING
HIGH-VALUE
```

```
HIGH-VALUES
HIGHLIGHT
I-O
I-O-CONTROL
ID
IDENTIFICATION
IF
IGNORE
IGNORING
IN
INDEX
INDEXED
INDICATE
INITIAL
INITIALIZE
INITIALIZED
INITIATE
INPUT
INPUT-OUTPUT
INSPECT
INTO
INTRINSIC
INVALID
IS
JUST
JUSTIFIED
KEY
LABEL
LAST
LEADING
LEFT
LENGTH
LESS
LIMIT
LIMITS
LINAGE
LINAGE-COUNTER
LINE
LINES
LINKAGE
LOCAL-STORAGE
LOCALE
LOCK
LOW-VALUE
LOW-VALUES
LOWLIGHT
MANUAL
MEMORY
MERGE
MINUS
MODE
MOVE
MULTIPLE
MULTIPLY
NATIONAL
NATIONAL-EDITED
NATIVE
NEGATIVE
NEXT
NO
NOT
NULL
NULLS
NUMBER
NUMBERS
NUMERIC
NUMERIC-EDITED
OBJECT-COMPUTER
OCCURS
OF
OFF
```

```
OMITTED
ON
ONLY
OPEN
OPTIONAL
OR
ORDER
ORGANIZATION
OTHER
OUTPUT
OVERFLOW
OVERLINE
PACKED-DECIMAL
PADDING
PAGE
PARAGRAPH
PERFORM
PF
PH
PIC
PICTURE
PLUS
POINTER
POSITION
POSITIVE
PRESENT
PREVIOUS
PRINTER
PRINTING
PROCEDURE
PROCEDURE-POINTER
PROCEDURES
PROCEED
PROGRAM
PROGRAM-ID
PROGRAM-POINTER
PROMPT
QUOTE
QUOTES
RANDOM
RD
READ
RECORD
RECORDING
RECORDS
RECURSIVE
REDEFINES
REEL
REFERENCE
RELATIVE
RELEASE
REMAINDER
REMOVAL
RENAMES
REPLACING
REPORT
REPORTING
REPORTS
REPOSITORY
REQUIRED
RESERVE
RETURN
RETURNING
REVERSE-VIDEO
REWIND
REWRITE
RF
RH
RIGHT
ROLLBACK
ROUNDED
```

```
RUN
SAME
SCREEN
SCROLL
SD
SEARCH
SECTION
SECURE
SEGMENT-LIMIT
SELECT
SENTENCE
SEPARATE
SEQUENCE
SEQUENTIAL
SET
SHARING
SIGN
SIGNED
SIGNED-INT
SIGNED-LONG
SIGNED-SHORT
SIZE
SORT
SORT-MERGE
SOURCE
SOURCE-COMPUTER
SPACE
SPACES
SPECIAL-NAMES
STANDARD
STANDARD-1
STANDARD-2
START
STATUS
STOP
STRING
SUBTRACT
SUM
SUPPRESS
SYMBOLIC
SYNC
SYNCHRONIZED
TALLYING
TAPE
TERMINATE
TEST
THAN
THEN
THROUGH
THRU
TIME
TIMES
TO
TOP
TRAILING
TRANSFORM
TRUE
TYPE
UNDERLINE
UNIT
UNLOCK
UNSIGNED
UNSIGNED-INT
UNSIGNED-LONG
UNSIGNED-SHORT
UNSTRING
UNTIL
UP
UPDATE
UPON
USAGE
```

```
USE
USING
VALUE
VALUES
VARYING
WAIT
WHEN
WITH
WORDS
WORKING-STORAGE
WRITE
YYYYDDD
YYYYMMDD
ZERO
ZEROES
ZEROS

[Keywords 2]
ABS
ACOS
ANNUITY
ASIN
ATAN
BYTE-LENGTH
CHAR
COMBINED-DATETIME
CONCATENATE
COS
CURRENT-DATE
DATE-OF-INTEGER
DATE-TO-YYYYMMDD
DAY-OF-INTEGER
DAY-TO-YYYYDDD
E
EXCEPTION-FILE
EXCEPTION-LOCATION
EXCEPTION-STATEMENT
EXCEPTION-STATUS
EXP
EXP10
FACTORIAL
FRACTION-PART
INTEGER
INTEGER-OF-DATE
INTEGER-OF-DAY
INTEGER-PART
LENGTH
LOCALE-DATE
LOCALE-TIME
LOCALE-TIME-FROM-SECONDS
LOG
LOG10
LOWER-CASE
MAX
MEAN
MEDIAN
MIDRANGE
MIN
MOD
NUMVAL
NUMVAL-C
ORD
ORD-MAX
ORD-MIN
PI
PRESENT-VALUE
RANDOM
RANGE
REM
REVERSE
SECONDS-FROM-FORMATTED-TIME
```

```
SECONDS-PAST-MIDNIGHT
SIGN
SIN
SQRT
STANDARD-DEVIATION
STORED-CHAR-LENGTH
SUBSTITUTE
SUBSTITUTE-CASE
SUM
TAN
TEST-DATE-YYYYMMDD
TEST-DAY-YYYYDDD
TRIM
UPPER-CASE
VARIANCE
WHEN-COMPILED
YEAR-TO-YYYY

[Keywords 5]
C01
C02
C03
C04
C05
C06
C07
C08
C09
C10
C11
C12
CONSOLE
FORMFEED
PRINTER
SWITCH-1
SWITCH-2
SWITCH-3
SWITCH-4
SWITCH-5
SWITCH-6
SWITCH-7
SWITCH-8
SYSERR
SYSIN
SYSIPT
SYSLIST
SYSLST
SYSOUT

[Keywords 4]

[Keywords 6]

[Keywords 7]
```

A Sample OpenCOBOL Program

```
       >>SOURCE FORMAT FIXED
       IDENTIFICATION DIVISION.
      *****************************************************************
      ** This sample OpenCOBOL program serves as a good tool to use  **
      ** when experimenting with colorizing Textpad syntax high-     **
      ** lighting.                                                   **
      *****************************************************************
       PROGRAM-ID. demosyntax.
       ENVIRONMENT DIVISION.
       CONFIGURATION SECTION.
       REPOSITORY.
           FUNCTION ALL INTRINSIC. *> Eliminate need for "FUNCTION" before functions
       DATA DIVISION.
       WORKING-STORAGE SECTION.
       01  Account-Type                 PIC X(1).
           88 Interest-Bearing-Checking VALUE 'c', 'C'.
           88 Statement-Savings         VALUE 's', 'S'.
           88 Platinum-Savings          VALUE 'p', 'P'.
       01  ADB-Char                     PIC X(10).
       01  Ave-Daily-Balance            PIC 9(7)V99.
       01  Formatted-Amount             PIC Z(6)9.99.
       01  Interest-Amount              PIC 9(7)V99.
       PROCEDURE DIVISION.
       000-Main.
           PERFORM UNTIL EXIT *> 12MAR2010 OpenCOBOL 1.1
      *    PERFORM FOREVER    *> 06FEB2009 OpenCOBOL 1.1
             DISPLAY
                 "Enter Account Type (c,s,p,other): "
                 WITH NO ADVANCING
             END-DISPLAY
             ACCEPT
                 Account-Type
             END-ACCEPT
             IF Account-Type = SPACES
               STOP RUN
             END-IF
             DISPLAY
                 "Enter Ave Daily Balance (nnnnnnn.nn): "
                 WITH NO ADVANCING
             END-DISPLAY
             ACCEPT
                 ADB-Char
             END-ACCEPT
             >>SOURCE FORMAT FREE
             MOVE NUMVAL(ADB-Char) TO Ave-Daily-Balance
             EVALUATE TRUE ALSO Ave-Daily-Balance
               WHEN Interest-Bearing-Checking ALSO 0.00 THRU 999.99
                 *> No Interest
                 MOVE 0 TO Interest-Amount
               WHEN Interest-Bearing-Checking ALSO 1000.00 THRU 1499.99
                 *> 0.5% Interest
                 COMPUTE Interest-Amount ROUNDED = 0.005 * Ave-Daily-Balance
               WHEN Interest-Bearing-Checking ALSO ANY
                 *> 1.0% Interest
                 COMPUTE Interest-Amount ROUNDED = 0.010 * Ave-Daily-Balance
               WHEN Statement-Savings ALSO 0.00 THRU 10000.00
                 *> 1.0% Interest
                 COMPUTE Interest-Amount ROUNDED = 0.010 * Ave-Daily-Balance
               WHEN Statement-Savings ALSO ANY
                 *> 1.0% Interest on 1st $10000 PLUS 1.5% on amount over $10000
                 COMPUTE Interest-Amount ROUNDED = 0.010 * 10000
                                                 + 0.015 * (Ave-Daily-Balance - 10000)
               WHEN Platinum-Savings ALSO ANY
                 *> 2% Interest
                 COMPUTE Interest-Amount ROUNDED = 0.020 * Ave-Daily-Balance
               WHEN OTHER
                 *> No Interest
                 MOVE 0 TO Interest-Amount
             END-EVALUATE
             >>SOURCE FORMAT FIXED
             MOVE Interest-Amount TO Formatted-Amount
             DISPLAY "Accrued Interest = " Formatted-Amount
           END-PERFORM
           .
```