

# Cobbler

## Overview

Cobbler is a desktop application for text editing of COBOL program text files. The tool offers basic light editing features and syntax highlighting. Cobbler was written by [Joshua Horvath](#).



## Quick Start

Open the Cobbler.jar file to start the application. On first launch of the Cobbler application a new Cobbler folder is created at the root level of the current user's user folder. This folder is used to store application settings such as what display theme to load and locations of recently opened files.

## Open Files

To open a COBOL text file:

1. From the **File** menu select the **Open...** menu option.
  - a. If there are unsaved changes, a confirmation dialog will be presented allowing a chance to stop the current operation and save the current document.
    - i. This unsaved changes check takes place for all operations were a file is opened, created, closed, or the application is shut down.
2. In the **Select a COBOL file** dialog, navigate to and select a COBOL text file.
  - a. The file suffix filter is available, and on by default, to only allow COBOL files to be selected.
    - i. The current COBOL file suffixes supported by this filter are:
      1. .cob, .cbl, .cpy, .pco, .fd, .sel, .ws
        - a. These suffixes will be recognized with all lowercase or uppercase versions.
    - ii. The filter may be turned off to allow other types of text files to be opened.
3. Click the **Open** button.
4. The contents of the text file are displayed in the main text pane of the application window.

## To open a recent file:

Files that have been opened in Cobbler in the past can be reopened under the Recent Files menu.

1. From the **File** menu select the **Recent Files** submenu.
  - a. Files that have been opened are listed in the Recent Files menu with full file paths displayed.
2. Select a file path from the submenu to open the file and display the contents in the Cobbler application.
  - a. If the file was moved, renamed, or deleted, an error will be displayed.

## Save Files

1. From the **File** menu select the **Save** or **Save As...** menu option.
  - a. If the **Save** option was selected, and file exists on the disk, then the changes will be written to the file and the operation ends. If the **Save** option was selected and the file does not exist on disk, then the **Save As** dialog is displayed.
2. In the **Save As** dialog, select a location to save the file, and edit the name of the file as needed, and click the **Save** button.
  - a. If a file suffix is not added by hand, the file suffix “.cob” will automatically be appended.

## Edit Menu

The Edit menu features contain controls that are standard for a text editor including:

- Undo
- Redo
- Select All
- Copy
- Cut
- Paste

## Utilities Menu

The Utilities menu contains features that assist in the editing of text files and changing application settings.

- Go to line...
- Find...

- Replace...
- Add Line Numbers
- Remove Line Numbers
- Settings...

## Go to Line

Opens a dialog that allows quickly moving to a specific line number in the text file.

## Find and Replace Dialogs

Dialogs for finding and replacing text within the open text file are available under the Utilities menu. Results, such as find and replace counts, will appear in the status bar at the bottom of the application window. Closing the find or replace dialog will reset the status bar contents. Additionally, a find and replace toolbar is available that sits at the bottom of the main application window. This toolbar can be accessed from the Find/Replace Bar menu option.

## Add Line Numbers

Automatically add line numbers to COBOL files. Line numbers are incremented by ten. If the file already has line numbers, then the file will be renumbered. If the first six characters in each line are not either all digits (in the renumbering case) or all word spaces, then Cobble may have trouble adding the line numbers without breaking changes. In this case a warning confirmation will be displayed, and Cobble will do its best to add the line numbers without making breaking changes. COBOL code should not be wholly overwritten with numbers, in worst case statement code will be pushed to the right and may require minor editing. Note that tab characters may cause this behavior.

## Remove Line Numbers

Automatically remove hard coded line numbers from COBOL files. The first six columns of COBOL code are reserved for line numbers but are rarely needed in modern COBOL code. If Cobble thinks it may have trouble removing line numbers from a file, a warning confirmation dialog will be presented. Also in this case, if Cobble encounters a line where it finds anything other than digit characters in the first six columns, that line will be skipped. If after removing line numbers the number of lines skipped is greater than zero, then a pop-up dialog will be displayed stating how many lines were skipped.

## Application Settings

From the Utilities menu exists a menu option to open the Settings dialog. In this dialog the following options exist:

- A menu where the user can select a visual theme for the cobbler application, including a dark mode.
- A menu that allows user to specify the number of recent files to track and display in the Recent Files menu.
- A checkbox that gives an option to clear the recent files menu.
- A checkbox to enable or disable the spell checker.
- A checkbox to show or hide invisible characters.

Changes made in the Settings dialog take affect when the **Save** button is clicked.

## New Document Template

Because COBOL is a verbose programing language, a feature that can be helpful is to autogenerate a basic file structure without having to manual type some of the boilerplate header information.

1. From the **File** menu select the **New Document Template** menu option.
2. A new hello world COBOL program is displayed in the main text pane of the application window.
  - a. The current user name is displayed as the Author.
  - b. The current date, in the form MM/DD/YYYY is displayed as Date Written.

## Text Autocomplete

Text autocomplete is available through the key command, control + spacebar. On typing this key combination, a popup menu is displayed. This popup menu is populated with all the COBOL keywords and intrinsic functions stored inside of Cobbler. The popup menu is filtered by what characters are typed to the immediate left of the curser location in the editor.

## Spell Checker

Comments within COBOL code can be spell checked. When a word appears within a code comment that is not recognized, then a squiggle line appears under that word. The comment spell checker can be turned off and on in the settings dialog.

## Notes and Current Issues:

- Keywords are recognized in lowercase, uppercase, and title case.
  - Recognized:
    - DISPLAY
    - display
    - Display
  - Not recognized:
    - disPLAY
- Syntax highlighting is not complete.
  - COBOL keywords containing a word space may not highlight correctly in the second word.
  - Sometimes a word may not highlight correctly if there is not a word space after the keyword.
  - Keywords containing a hyphen may not highlight.
- The feature checking for unsaved changes is currently a little buggy sometimes.
- Status bar at bottom of window main sometimes temporarily disappear when using the Find/Replace toolbar.

# Appendix I

## COBOL Reserved Words

accept	binary-short [200x]	computational-2	db-key
access	bit	computational-3	db-record-name
add	bits	computational-4	db-set-name
address	blank	computational-5	db-status
advancing	blink [xopen]	computational-6	db-uwa
after	blinking	computational-x	dbcs
all	block	compute	dbkey
allowing	bold	concurrent	de
alphabet	boolean	configuration	debug-contents
alphabetic	bottom	connect	debug-item
alphabetic-lower	by	contain	debug-length
alphabetic-upper		contains	debug-line
alphanumeric	call	content	debug-name
alphanumeric-	cancel	continue	debug-numeric-
edited	cd	control	contents
also	cf	controls	debug-size
alter	ch	conversion	debug-start
alternate	changed	converting	debug-sub
and	character	copy	debug-sub-1
any	characters	core-index	debug-sub-2
apply	class	corr	debug-sub-3
are	clock-units	corresponding	debug-sub-item
area	close	count	debug-sub-n
areas	cobol	crt	debug-sub-num
ascending	code	currency	debugging
assign	code-set	current	decimal-point
at	col [200x]	cursor	declaratives
author	collating		default
auto [xopen]	column	data	delete
automatic	comma	date	delimited
autoterminate	commit	date-compiled	delimiter
	common	date-written	dependency
background-color	communication	day	depending
[xopen]	comp	day-of-week	descending
batch	comp-1	db	descriptor
before	comp-2	db-access-	destination
beginning	comp-3	control-key	detail
bell [xopen]	comp-4	db-condition	dictionary
binary	comp-5	db-current-record-	disable
binary-char [200x]	comp-6	id	disconnect
binary-double	comp-x	db-current-record-	disp
[200x]	computational	name	display
binary-long [200x]	computational-1	db-exception	display-1

display-6	end-search	foreground-color	key
display-7	end-start	[xopen]	
display-9	end-store	free	label
divide	end-string	from	last
division	end-subtract	full [xopen]	ld
does	end-unstring	function	leading
down	end-write		left
duplicate	ending	generate	length
duplicates	enter	get	less
	entry	giving	limit
echo	environment	global	limits
editing	eol [xopen]	go	linage
egi	eop	goback	linage-counter
eject	eos [xopen]	greater	line
else	equal	group	line-counter
emi	equals		lines
empty	erase [xopen]	heading	linkage
enable	error	high-value	locally
end	esi	high-values	lock
end-accept	evaluate	highlight [xopen]	lock-holding
end-add	every		low-value
end-call	examine	i-o	low-values
end-commit	exceeds	i-o-control	lowlight [xopen]
end-compute	exception	id	
end-connect	exclusive	ident	manual
end-delete	exhibit	identification	match
end-disconnect	exit	if	matches
end-divide	exor	in	member
end-erase	extend	including	membership
end-evaluate	external	index	memory
end-fetch		indexed	merge
end-find	failure	indicate	message
end-finish	false	initial	mode
end-free	fd	initialize	modify
end-get	fetch	initiate	modules
end-if	file	input	move
end-keep	file-control	input-output	multiple
end-modify	filler	inspect	multiply
end-multiply	final	installation	
end-of-page	find	into	named
end-perform	finish	invalid	native
end-read	first	is	negative
end-ready	float-extended		next
end-receive	[200x]	just	no
end-reconnect	float-long [200x]	justified	non-null
end-return	float-short [200x]		not
end-rewrite	footing	kanji	note
end-rollback	for	keep	null

number		rewrite	standard-1
numeric	random	rf	standard-2
numeric-edited	rd	rh	start
	read	right	status
object-computer	readers	rms-current-	stop
occurs	ready	filename	store
of	realm	rms-current-sts	stream
off	realms	rms-current-stv	string
offset	receive	rms-filename	sub-queue-1
omitted	reconnect	rms-sts	sub-queue-2
on	record	rms-stv	sub-queue-3
only	record-name	rollback	sub-schema
open	record-overflow	rounded	subtract
optional	recording	run	success
options [200x]	records		sum
or	redefines	same	suppress
order	reel	screen [xopen]	symbolic
otherwise	reference	sd	sync
	reference-modifier	search	synchronized
packed-decimal	references	section	
padding	regardless	secure [xopen]	table
page	relative	security	tallying
page-counter	release	segment	tape
password	reload	segment-limit	tenant
perform	remainder	select	terminal
pf	remarks	send	terminate
ph	removal	sentence	test
pic	renames	separate	text
picture	reorg-criteria	sequence	than
plus	replace	sequence-number	then
pointer	replacing	sequential	through
position	report	service	thru
positioning	reporting	set	time
positive	reports	sets	times
previous	required [xopen]	sign	to
printing	rerun	signed [200x]	top
prior	reserve	size	trace
procedure	reset	skip1	trailing
procedures	retaining	skip2	transform
proceed	retrieval	skip3	true
program	return	sort	type
program-id	return-code	sort-merge	
protected	[xopen]	source	underline [xopen]
purge	returning	source-computer	underlined
	reverse-video	space	unequal
queue	[xopen]	spaces	unit
quote	reversed	special-names	unlock
quotes	rewind	standard	unsigned [200x]



unstring  
until  
up  
update  
updaters  
upon  
usage  
usage-mode

use  
using  
  
value  
values  
varying  
vfu-channel

wait  
when  
where  
with  
within  
words  
working-storage  
write

writers  
  
zero  
zeroes  
zeros

## Appendix II

### COBOL Intrinsic Functions

acos  
add-duration  
annuity  
asin  
atan  
char  
convert-date-time  
cos  
current-date  
date-of-integer  
day-of-integer  
day-to-yyyyddd  
extract-date-time  
date-to-  
yyyymmdd

display-of  
factorial  
find-duration  
integer  
integer-of-date  
integer-of-day  
integer-part  
length  
locale-date  
locale-time  
log  
log10  
lower-case  
max  
mean

median  
midrange  
min  
mod  
national-of  
numval  
numval-c  
ord  
ord-max  
ord-min  
present-value  
random  
range  
rem  
reverse

sin  
sqrt  
standard-deviation  
subtract-duration  
sum  
tan  
test-date-time  
trim  
triml  
trimr  
upper-case  
variance  
utf8string  
when-compiled  
year-to-yyyy