

# Version 12.0

Released 7th May 2019 • Updated 4th June 2019

FontCreator 12.0 is a major upgrade with powerful new features, and improvements to existing features.

An anchored-based method for creating composites improves the positioning of diacritics, especially for italic typefaces. User-friendly glyph name generation and editing is more powerful. Inserting glyphs anywhere in the font, adding code-points based on glyph names, and glyph sorting are invaluable for large fonts.

CompositeData.xml has been extensively revised, using glyph names instead of glyph mappings to remove dependence on the Private Use Area. Glyphs for OpenType features such as Small Capitals, Alternative Fractions, or Stylistic Alternates are no longer mapped. Transform scripts also use these glyph names.

Many new definitions have been added to aid in the design of Mathematical Operators, and a few in other character sets. Definitions for diacritical marks have also been enhanced.

Free help is available on the [support forum](#), even for the unregistered 30-day trial version. The [Help File](#) is available online, so anyone can see the features that are available before downloading the free trial version. The trial version is the full Professional Edition, but it doesn't save fonts. Existing users may like to see which feature requests have been [implemented already](#).

**Important Note:** FontCreator 12 projects are not backwardly compatible. Edit **copies** of your font projects and keep the originals as backups.

To avoid overwriting a previous version of FontCreator, install FontCreator 12 into a different folder with command: WinKey + R:

---

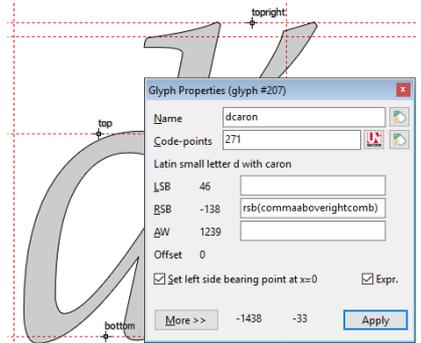
**Tip:** "%UserProfile%\Downloads\FontCreatorSetup-x64.exe" /DIR="C:\Program Files\High-Logic FontCreator 12"

---

The 32-bit version will be installed to C:\Program Files (86)\High-Logic\FontCreator 12

## What is New?

- Improved performance of the Font Overview.
- Improved calculation of font metrics to use Typo Metrics for line-spacing. See the tutorial: [Font Metrics — Vertical Line Spacing](#).
- Support for displaying glyphs by code-point in the Preview Toolbar.
- Enhanced Glyph Properties dialogue with expressions for anchor based composites.
- The Bearings Tab on the Transform dialogue was removed since its function is duplicated on the Glyph Properties dialogue, which now has fields for expressions and advance width.
- New transform script to select glyphs that may need to make use of auto attach.
- Improved design mode sort order.
- **Anchor-based Generation of Composites:** Open the drop-down beside the Complete Composites icon to select the mode to use. Anchor Based will add anchors to the selected glyphs. Complete Composites will use the data in CompositeData.xml and will not add any anchors. Anchor-based positioning is generally more reliable, but Complete Composites works better in some cases. The two methods can be mixed. The last-used method is remembered and shown in the tooltip on hovering the icon. There is a checkbox in the drop-down menu by the last-used method. Auto will use anchor-based for accented letters, but complete composites otherwise. Anchor Based will recompose existing composites, but will not process glyphs such as Æ, while Auto and Complete Composites will compose it with A and E. If you want to use a different method for some glyphs without changing the last-used method, this can be done by using the submenu on the context menu in the glyph overview.
- **Snap to Guidelines Improvements:** If snapping is enabled, anchors will snap to guidelines. Resizing contours horizontally or vertically will also snap to guidelines now, though not if resizing from the corner handles.
- **OpenType Designer Improvements:** The Button at the top left of the OpenType Designer dialogue opens another dialogue to generate and update OpenType features. For example, one can add features for anchors: glyph composition/decomposition, mark positioning, and mark to mark positioning. Fractions (.frac), Alternative Fractions (.afrc) and Localised Forms (locl) features can be generated if the font contains suitable glyphs. Ligatures (.liga, .hlig, .dlig), and Ornaments features (.ornm) can be added, and a Capital Spacing (.csp) feature that uses a percentage of advance width or a percentage of units/em. Lookup tables can be rebuilt to include all suitable glyphs available in the font. Several other buttons have been moved to a drop list — making it harder to use some commands accidentally.



- **Toolbar Improvements:**

- The Standard Toolbar has been split, creating a new Tools Toolbar, and an icon has been added for Custom Formulae.

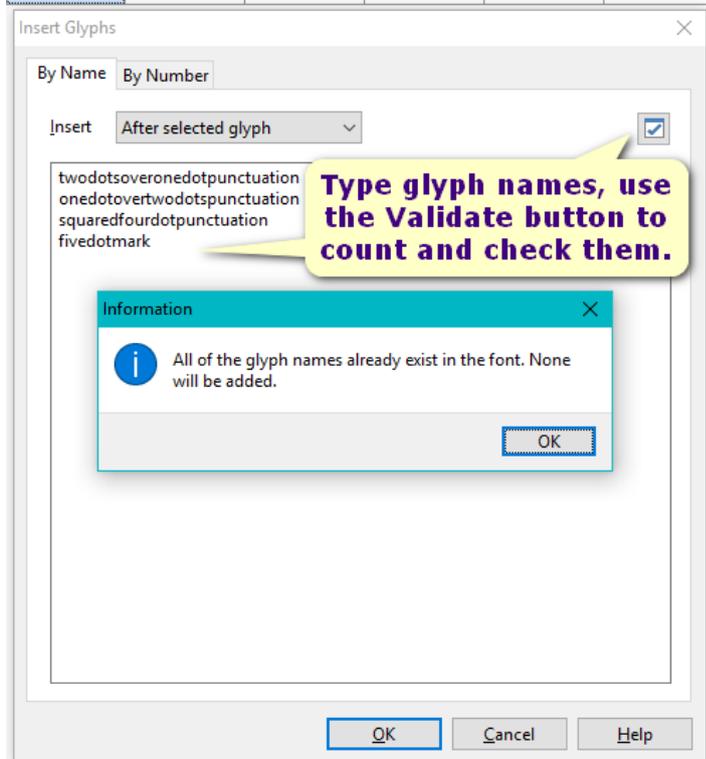


- The Make Simple and Make Empty icons have been moved to the Tools Toolbar.

- **Generation of Friendly Glyph Names:** User-friendly glyph names can be added to the full range of glyphs supported by Unicode 12. For example, Miscellaneous Symbols and Pictograms \$F4A1 will be named "electriclightbulb," Emoticon \$F602 as "facewithtearsofjoy," etc. These user-friendly names are generated automatically from the UnicodeData.txt file supplied by the Unicode Consortium. They cover the full range of Unicode blocks and scripts. The names are stored in the Font Project file. By editing the file "glyphnewnames.dat" users can define their own custom glyph names that will not be overwritten by the automated glyph naming. For example, to use "circled digit one" and so forth for Alternative Annotation Forms (nalt), add new lines: 2460;one.nalt, 2461;two.nalt, 2462;three.nalt, etc. If characters are added via the Insert Characters dialogue or with transform scripts using code-points, user-friendly glyph names will be added too, e.g. if character \$1F6D2 is added it will be named as "shoppingrolley."



- **Insert Glyphs Improvements:** Insert glyphs and add code-points based on glyph names. This improved feature can now add glyphs before or after the selected glyph(s), or at the beginning or end of the font. If user-friendly glyph names are typed in the dialogue, glyphs will be mapped to the correct code-points. If ligatures are typed, composite glyphs will be generated automatically, e.g. type "fi=f+idotless" in the dialogue to create an fi ligature without the dot, or type "fi=f+i" to create an fi ligature with the dot. The validate button counts the number of glyphs, strips duplicates, and checks if the glyphs exist. The "By Number" tab simply inserts unmapped glyphs before or after the selected glyphs, or at the top of bottom of the font. The dialogue remembers its last-used settings.

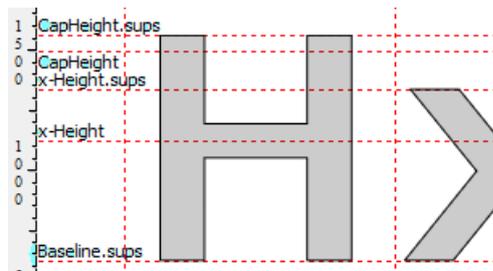


- **Unicode 12.1 Updates:** The Unicode Blocks and Character sets were updated. Comments were added to CompositeData.xml to mark the beginning of each block for users who wish to add their own definitions. New definitions must be added in order of code-points. The comments serve as a guide where to insert definitions.

- **Improved Complete Composites:** The feature is now independent of the Private Use Area. OpenType features can be added to any font even if some PUA code-points are already in use. Users can add sets of glyphs for use by OpenType features: Petite Capitals and Small Capitals for Latin, Greek, or Cyrillic, Titling Capitals, Stylistic Alternates, Alternative Fractions, Superscripts and Subscripts for fractions, Scientific Inferiors, Ordinals, Old Style Figures, Lining, Proportional, or Tabular Figures, Historical Ligatures, and Discretionary Ligatures, while leaving the PUA free for other purposes, e.g. for glyphs that have no Unicode code-point assigned. Many errors were corrected, and positioning was improved, especially for italic font styles. Comments were added for Unicode Blocks to make the file easier to navigate. Complete Composites can be used to generate composite glyphs for fractions, scaled glyphs for superscripts, and for many symbols. Using it for Maths Operators, for example, ensures uniform advance widths and uniform strokes, e.g. for + × = and ±. Currency symbols use the figure width where possible. Trigrams and Digrams can be generated from Monograms; Enclosed alphanumeric can be generated from Large Circle and superscripts (or Petite Capitals), and Alphabetic Presentation Forms can be generated from lowercase letters. There are no definitions yet for Braille Patterns, but a character set like that obviously lends itself to automation. If composites are used, the whole character set can be modified by editing a few base glyphs from which the rest of the character set is composed. Font files will also be smaller because the same

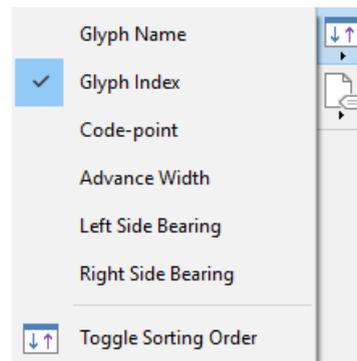
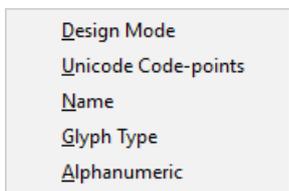
outlines are reused many times. For more detailed information on the changes read Composite Data Updates.

- Metric Lines for Superscripts etc.** When opening glyphs with suffixes such as .sups, .subs, .dnom, .salt, .titl, .pcap, .smcp, etc., in the Glyph Edit Window, relative metric lines are displayed. The relative metrics are calculated from H.sups, x.sups, x.subs, H.salt, x.sinf, H.pcap or h.pcap, etc. If suitable glyphs do not exist, the relative metrics lines will not be displayed. The capital superscripts in the screenshot are aligned with TypoAscender. Baseline.sups and CapHeight.sups make it easier to position superscripts like O.sups or zero.sups with the correct overshoots above and below the relative metric lines.



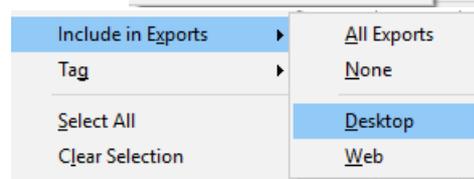
- Enhanced Transform Scripts:** All of the transform scripts that relied on using code-points in the Private Use Area have been updated to use glyph names. The Validate button on the Transform Wizard dialogue will now change the code format from hexadecimal to decimal, or vice versa, depending on the user's selected preference, and convert consecutive glyphs to a range, e.g. "one two three four five" will be converted to "one-five." Terminology on the Transform Wizard dialogue was made more consistent. Some new Transform scripts were added to Align Horizontal Arrows (vertically), set bearings and advance width for combining diacritics, and to select glyphs for tagging based on their names. Other scripts were updated to use user-friendly glyph names in preference to code-points. Note: Generate Glyph Names before using them. The Transform Wizard also supports the new anchored based methods for Complete Composites. For more detailed information on the changes to Transform scripts read Using Glyph Transformations.
- Undo/redo for Glyph Transformations:** Transformations can now be undone, which makes it much easier to test with different values and quickly arrive at the desired result, e.g. when generating Petite Capitals, which should be the same height as the x-Height, and the same weight as lowercase glyphs. The default values were based on the Verajja font, but will nearly always need to be adjusted for other fonts.

- Sort Glyphs Improvements:** Design Mode sorts composite glyphs after their base glyph: A, À, Á, etc. Capital letters come first, followed by glyphs grouped by suffix: .pcap, .salt, .smcp, A.sups, A.subs. These are followed by lowercase a, à, á etc., then lowercase superscripts and subscripts, then other glyphs grouped by type. Sorting glyphs permanently changes the glyph index. To sort glyphs without modifying the font, use the Sort drop-list on the Overview Toolbar. Unicode Code-points sorts by code-points, without sorting un-mapped glyphs. The other options are the same as in earlier versions.



glyphs without modifying the font, use the Sort drop-list on the Overview Toolbar. Unicode Code-points sorts by code-points, without sorting un-mapped glyphs. The other options are the same as in earlier versions.

- Option to Include Glyphs in Exports:** Glyphs can be included in Exports to Desktop (ttf/otf) or Web fonts (woff/woff2) using the context menu. Rarely used glyphs can be included in Open-Type fonts, but not in Web fonts, thus reducing the size of fonts for the web. Glyphs that are only used for design purposes can be excluded from all exports by selecting include in None. The marked glyphs acquire a prefix in the glyph overview caption of (d) for Desktop, (w) for Web, or (x) for None. Unmarked glyphs have no prefix and are included in All exports.



- Font Overview Side Panel Improvements:**

New categories in the overview side panel make it easy to see which glyphs will be exported. Glyphs that are experimental or only used as components of other glyphs can always be excluded when exporting. Fonts designed for the Web may not need as many symbols as those intended for print publishing. Many glyphs can be excluded from publishing to WOFF formats. I previously renamed my tags to identify glyphs not to include web formats and had to delete them before export. The "Include in Export" options can avoid that step, though I still use it to export Open-Type fonts for Web publishing because in Serif WebPlus using OTF requires no special code and uploads the fonts automatically, unlike the more compact WOFF formats for which one must write a few lines of @fontface code and upload the fonts separately.

Export	2743
Desktop	2743
Web	1635
Tagged	1167
To Do	0
Recently Added	0
Completed	0
Colour	0
Not in WOFF	1167
Type	2743
A Characters	1953
Scripts	
Latin	1149
Greek	143

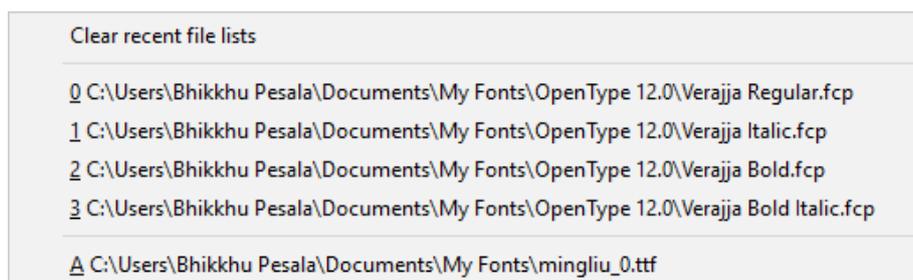
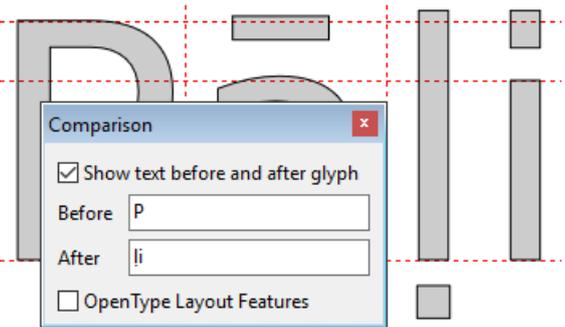
The latest update now also includes an additional side panel category for scripts.

- Improved Window Tab Bar:**

- ❑ Tabs can be dragged to any position.
- ❑ Glyph edit windows open on the tab bar next to the overview tab for that type style.
- ❑ Open in New Window from Composite Glyphs opens a new tab to the right of the current tab.
- ❑ An icon indicates if a project has unsaved changes.
- ❑ The tab bar becomes scrollable when full instead of wrapping to multiple rows.



- **Improved Templates for New Fonts:** These were updated to include combining diacritics and grid-fitting options. Obsolete glyphs .null and nonmarkingreturn were removed. Some glyphs were improved for when "Include Contours," is selected. It is recommended to use combining diacritics with zero advance width for anchor-based composites. New font projects include a basic set of diacritics to support Latin Basic and Latin-1 Supplement character sets. More characters can be added easily using Transform scripts to support more languages, or unmapped glyphs for Small Capitals, etc.
- The ttfautohint plugin was updated to version 1.82.
- Support for delete in multiple substitutions.
- The list of Vendor IDs was updated.
- Unsupported Table Streams can be viewed.
- **Option to decompose composite glyphs with scaled components on export:** Scaled components may display incorrectly in applications. Making the composite glyphs simple on export prevents such issues.
- Scrolling in dialogue boxes supports the use of the mouse-wheel. In the Composite Glyph member properties dialogue, for example, one can now scroll x/y position by single funits and x/y scale by 0.01 instead of 0.1. Note: It is the field under the mouse pointer that changes value, not the active field.
- **Comparison Toolbar Improvements:** Clicking on the comparison glyphs in the Glyph Edit Window jumps to that glyph, while adding the previously viewed glyph to the before or after field on the Comparison Toolbar. The context menu in composite glyphs has a new option to add the glyph members to the comparison toolbar.
- **Reopen Submenu:** The full path and filename is now shown in the reopen submenu. The list can be cleared if it becomes too cluttered. Note: One can reopen a second copy of a font that is already open, so take care to reopen the right file; you could lose edits if you save the older version after saving and closing the newer version.



## Conclusions

FontCreator 12.0 simplifies the process of adding OpenType features, much of which is now automated. The transition from dependence on glyph mappings to glyph names has been eased with the automatic generation of user-friendly glyph names and new transform scripts that add glyphs for OpenType features with just a few clicks. Those new to font editing will find it easier to add OpenType features, while experienced font designers will appreciate the time saved by greater automation. Both types of user will benefit from the improved tab bar, font sorting, and undo of glyph transformations.