Alkalami - Font Features

Alkalami is a TrueType font with smart font capabilities added using the OpenType font technology. The Alkalami font includes a number of optional features that provide alternative rendering that might be preferable for use in some contexts. The sections below enumerates the details of these features. Whether these features are available to users will depend on both the application and the rendering technology being used. Features are available in OpenType. Some applications let the user control certain features such as Stylistic Sets to turn on the rendering of variant characters. However, at this point, many applications do not make use of those features so another solution is needed to show the variant characters. TypeTuner creates tuned fonts that use the variant glyph in place of the standard glyph.

See Using Font Features. Although that page is not targeted at Arabic script support, it does provide a comprehensive list of applications that make full use of the OpenType (and Graphite) font technologies.

See also Arabic Fonts — Application Support. It provides a fairly comprehensive list of applications that make full use of the OpenType font technology.

This page uses web fonts (WOFF) to demonstrate font features and should display correctly in all modern browsers. For a more concise example of how to use Alkalami as a web font see *Alkalami-webfont-example.html* in the font package web folder.

If this document is not displaying correctly a PDF version is also provided in the documentation/pdf folder of the release package.

Signs Spanning Numbers

These signs spanning numbers are intended to enclose or hold one or more digits. Specific technical details of how to use them are discussed in the Arabic fonts FAQ -- Subtending marks.

Currently, this font only supports the **Number Sign** (U+0600) and **End of Ayah** (U+06DD). Additionally, Alkalami includes two simplified alternates for U+06DD ARABIC END OF AYAH under the Stylistic Alternates (salt) feature, but at this time we know of no OpenType-based applications that can access these. The two alternates are also available through the Stylistic Sets feature discussed below.

Customizing with TypeTuner

For applications that do not make use of the OpenType Stylistic Sets, you can now download fonts customized with the variant glyphs you choose. Read this document, visit TypeTuner Web, then choose the variants and download your font.

Language

Affects: U+065C, U+0627 (with U+064E, U+064F, U+0650), U+063F, U+069F, U+0751, U+0763, U+08C3, U+08C4

Unfortunately, the UI needed to access the language-specific behavior is not yet present in many applications. Some Harfbuzz-based apps, e.g., XeTeX, can access language-specific behavior.

Language	Imala	Touching	Wagaf	Feature
	е			setting

Language	lmala e	Touching	Wagaf	Feature setting
default	ī	آأ <u>ا</u> باً باً با	ڒؿؿڕۛڟڟڟڋڹڹڹۮػػػڠؙۼؙۛۼٞۿٙ ڡ۠ڡٛ۠ۊ ؞	
Hausa	Ŧ	Hِايِأَيْايا	ڒؿؿڒڟڟڟڎ۪ڹؿڎ۪ػػػػۼۛٛۼؙۛۼٛؖۿ ڰ۬ڡ۠ٚ <u>۫</u> ۊ	lang=ha

Stylistic Sets

There are some character shape differences in different languages which use the Arabic script. These can be accessed by using the OpenType Stylistic Sets, or through the language support mentioned above.

Imala e (ss04)

Affects: U+065C

Feature	Sample	Feature setting
Standard	:	ss04=0
Small	Ŧ	ss04=1

Jeem/Hah (ss07)

Affects: U+062C, U+062D, U+062E, U+0683, U+0684, U+08A2

Feature	Sample	Feature setting
Standard	ݮ ݴݷݷ ݦݲݲݲݲݲݞݠݻݻݦݷݙݷݷݦݴݻݻݻ	ss07=0
Flat style	ج دجم م حصم م فخخ ۾ دي ي ۾ ڊڄې ۾ ٿِڄَيِّ	ss07=1

Alef diacritic placement (ss08)

Affects: U+0627, U+064E, U+064F, U+0650

Feature Sample		Feature setting		
Standard	آلِ يِبَا يِباً يِبا	ss08=0		
Touching	الماية يأيا	ss08=1		

Wagaf (ss09)

Affects: U+063F, U+069F, U+0751, U+0763, U+08C3, U+08C4

Feature value	Sample	Feature setting
Standard	رْيْيْرْطْ طْطْطْبْ بْبْبْ كْخْخْكْ غْرْغْفْمْ وْفُفُو	ss09=0

Feature value	Sample	Feature setting
Small	ڗؿۣؿڟڟڟڎ۪ۺڎ۪ػػػػٵ۫ۼؙۛڡؙ۠ۿ۬ ؋ٷٛڡ ٛٚٛٛۊ	ss09=1

End of ayah

Affects: U+06DD

Firefox allows you to use U+06DD followed by the digits and proper rendering occurs. Some applications require the following:

- precede the entire sequence (subtending mark plus following digits) with 202D LEFT-TO-RIGHT OVERRIDE
- follow the entire sequence with U+202C POP DIRECTIONAL FORMATTING.

Surrounding the sequence with U+202D and U+202C seems to give the most reliable results in different browsers. However, we have not found a solution that works in Internet Explorer/Edge.

In the example below, the following codepoints are used: U+202D U+06DD U+0031 U+0032 U+0033 U+202C U+202D U+06DD U+0611 U+0622 U+0663 U+202C.

Feature	Sample	Feature setting
Standard	123 (Tr	ss02=0 ss03=0
End of ayah A	①23①Y**	ss02=1
End of ayah B	123 174	ss03=1

Proportional Figures

Alkalami supports the OpenType **Proportional Figures (pnum)** for Latin digits. *This feature is not supported in TypeTuner Web.*

Affects: U+0030..U+0039

Feature	Sample	Feature setting
Tabular Figures	9876543210	pnum=0
Proportional Figures	9876543210	pnum=1

Contextual Alternates

Alkalami has a few contextual rules applying subtle differences in the shape and position of certain of the characters depending on context (sometimes shortening, sometimes lengthening or lowering a "tail"). InDesign is able to utilize some of these cross word contextual alternates; however, most applications are not advanced enough to use this behavior in the font. Unfortunately, at this point in the development of the font, combining marks above or below may break these contextual alternates.

Substitutions to avoid collisions in strings with noon-like characters

زترنز

Substitutions to avoid collisions in strings with yeh-like characters پوپلرپای

Substitutions to avoid collisions in strings with seen-like characters سرسه س

Substitutions for shortening dal-like characters (shorter tail on dal)

Substitutions for shortening waw-like characters



Note!

If you used previous versions of this font, we made some decisions which were unfortunately not Unicode compliant. Since the initial release of this font, many of these non Unicode compliant characters have been added to Unicode. You will need to re-encode some characters to be fully Unicode compliant. The characters affected are listed below:

Glyph	Non Unicode Compliant	\rightarrow	Glyph	Convert to
<u>e</u> j	0643 ARABIC LETTER KAF	\rightarrow	<u> </u>	06A9 ARABIC LETTER KEHEH
<u>=</u> 1	06AD ARABIC LETTER NG	\rightarrow	<u> </u>	0763 ARABIC LETTER KEHEH WITH THREE DOTS ABOVE
Ţ	067B ARABIC LETTER BEEH (ss05)	\rightarrow	111 1	0751 ARABIC LETTER BEH WITH DOT BELOW AND THREE DOTS ABOVE (use Stylistic Set ss09 if you wish a small wagaf)
Ė	069F ARABIC LETTER TAH WITH THREE DOTS ABOVE (ss05)	\rightarrow		You may continue to use this codepoint (use Stylistic Set ss09 if you wish a small wagaf)
#	06A0 ARABIC LETTER AIN WITH THREE DOTS ABOVE (ss05)	\rightarrow	<u> </u>	08C3 ARABIC LETTER GHAIN WITH THREE DOTS ABOVE (use Stylistic Set ss09 if you wish a small wagaf)
ٷ	06A8 ARABIC LETTER QAF WITH THREE DOTS ABOVE (ss05)	\rightarrow	وققق	08C4 ARABIC LETTER AFRICAN QAF WITH THREE DOTS ABOVE (use Stylistic Set ssø9 if you wish a small wagaf)

Glyph	Non Unicode Compliant	\rightarrow	Glyph	Convert to
امی	06CC ARABIC LETTER FARSI YEH (ss05)	→ 、	ونيو	063F ARABIC LETTER FARSI YEH WITH THREE DOTS ABOVE (use Stylistic Set ssø9 if you wish a small wagaf)
Ľ	0763 ARABIC LETTER KEHEH WITH THREE DOTS ABOVE (ss05)	\rightarrow		You may continue to use this codepoint (use Stylistic Set ss09 if you wish a small wagaf)
ز	0646 ARABIC LETTER NOON (ss01)	→ _	رنتر	08BD ARABIC LETTER AFRICAN NOON (no Stylistic Set required)
بح	064A ARABIC LETTER YEH (ss01)	→ _	کظیر	06CC ARABIC LETTER FARSI YEH (no Stylistic Set required)
4	06A2 ARABIC LETTER FEH WITH DOT MOVED BELOW (ss01)	\rightarrow	ववंव	08BB ARABIC LETTER AFRICAN FEH (no Stylistic Set required)
اف	06A7 ARABIC LETTER QAF WITH DOT ABOVE (ss01)	→	وفغو	08BC ARABIC LETTER AFRICAN QAF (no Stylistic Set required)

This guide is from the Alkalami project version 2.100 and is copyright © 2015-2023 SIL International.