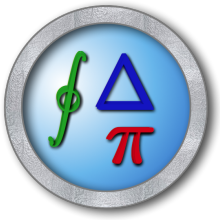

TeXnicle: A User Guide

Issue	1
Revision	1
Number of pages	17
date of issue	July 18, 2012

Contents

1 Introducing TeXnicle

TeXnicle is an editor and project manager for producing documents using LaTeX and similar typesetting languages. TeXnicle has been under development since 2010 and is intended to provide a fast, feature-rich environment for writing LaTeX documents under Mac OS X. Employing all the features of modern Mac OS X applications, TeXnicle fits right at home on a Mac.

One of the main design drivers for TeXnicle was to produce an editing environment similar to Xcode [1] the development environment Apple provides for building applications on Mac OS X.

asd

1.1 In this manual

This manual is split into four chapters. The first two are just to get us started: This introduction; an installation and requirements discussion. The third chapter, The User Guide covers typical usage scenarios and introduces the basic concepts used throughout TeXnicle. The fourth chapter is an in-depth reference guide to TeXnicle's features.

2 Installation, Setup and Requirements

TeXnicle expects you to have an installed LaTeX typesetting system on your machine. By default, TeXnicle is setup to work with installations of MacTeX [2]. If you have an alternative LaTeX installation, you may need to setup some paths. In particular, you may need to copy and edit one or more of the built-in engines that TeXnicle uses to typeset documents. That is described in section 4.4.

In addition to the engines discussed above, TeXnicle uses some commands for typesetting code snippet previews. These are set in the Preferences pane "Palette & Library" and are discussed further in section 4.1.6.

3 User Guide

This sections focuses on using TeXnicle for common tasks.

3.1 Welcome to TeXnicle

Describe the welcome screen

3.1.1 Quickstart

some

3.1.2 Creating a new LaTeX file

two

3.1.3 Creating a new TeXnicle project

3.2 The TeXnicle editor

3.3 The Navigators

3.3.1 Project Manager

3.3.2 Symbol Palette

TeXnicle includes a comprehensive symbol browser so you can easily find the symbol you want. To view the symbol browser, select the symbol browser icon above the project tree. Figure 3.1 shows this.

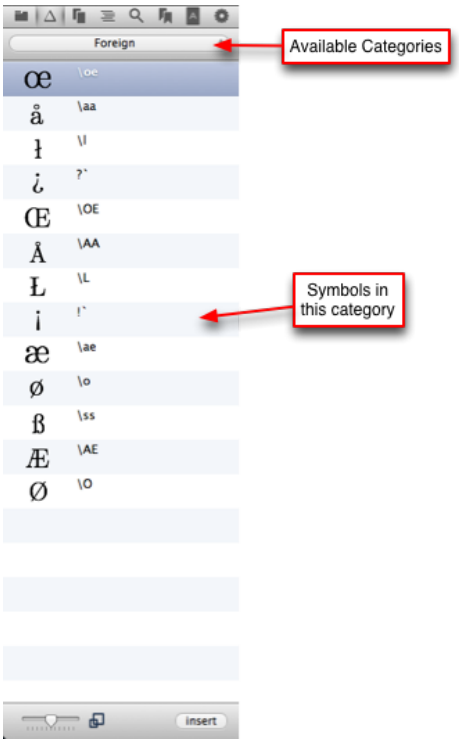

Figure 3.1: The built-in symbol palette of TeXnicle.

You can also use the keyboard shortcut `alt-cmd-2`. You will then see the symbol browser in place of the project tree.

To insert a symbol in to the document currently being edited, select the symbol you want then click the insert button. Alternatively, you can drag the symbol in to the text or just double click the symbol to insert it at the current cursor position.

3.3.3 Code Snippets Library

TeXnicle has a built-in library where you can store code snippets. You can organise the snippets into different categories. Some standard categories and clippings are included to get you started.

To view the clippings library, select the 3rd toolbar icon as shown in Figure 3.2.

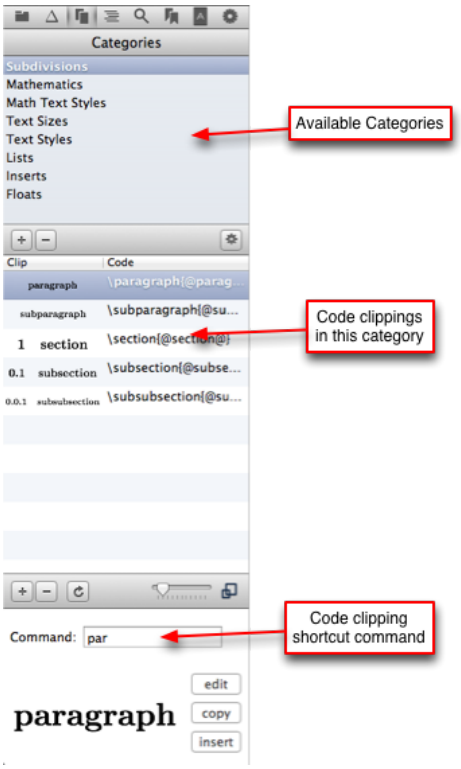

Figure 3.2: The code snippet library of TeXnicle.

3.3.4 Document Outline

3.3.5 Project Search

3.3.6 Bookmarks

3.3.7 Spelling

3.3.8 Project Settings

4 Reference

4.1 Preferences

4.1.1 General

4.1.2 Typesetting

4.1.3 Fonts & Colors

4.1.4 Templates

4.1.5 Commands

4.1.6 Palette & Library

4.1.7 File Types

4.2 The stand-alone LaTeX editor

4.3 Project editor

4.4 Engines

TeXnicle has configurable engines. You can simply use the supplied engines, or you can create your own custom engines. TeXnicle engines are simple scripts which reside in `/Library/Application Support/TeXnicle/engines`.

These script files are passed in a number of variables from TeXnicle. If you make a new engine, the template comes preconfigured with the passed-in variables set.

To customise one of the built-in engines, select the "Typesetting" tab in the preferences, from there, select the "Engines" sub-tab. Figure 4.1 shows this tab.

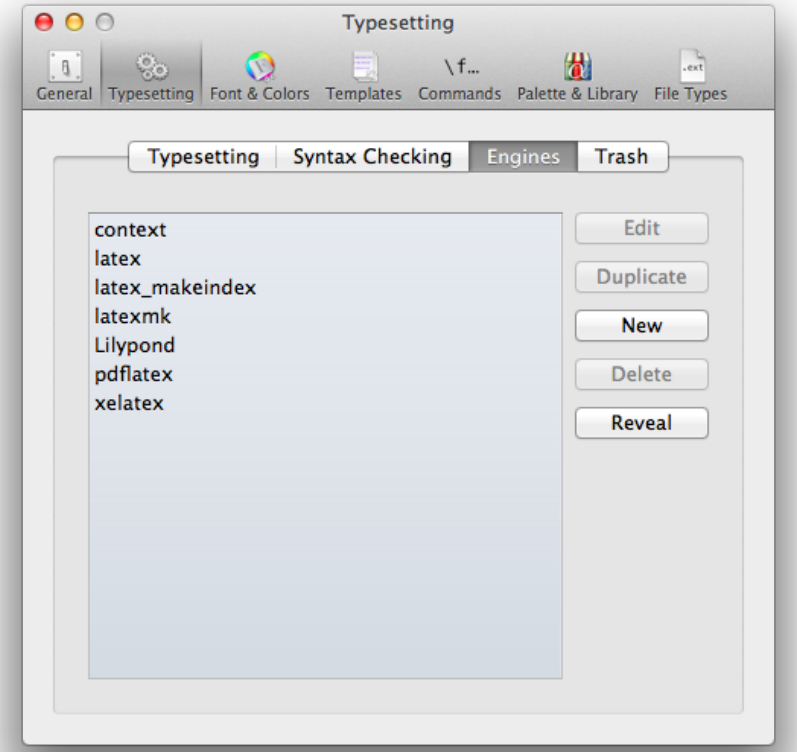

Figure 4.1: The preferences pane where you can edit and create engines.

Note: You can only edit engines which you create yourself. To edit one of the built-in engines, you first need to duplicate it.

Select one of the existing engines and click the âĂŸDuplicateâĂŹ button. You can then go ahead and edit the duplicated engine script file.

On the "Typesetting" tab, you can set the default engine that is used for new projects and documents. You can also give defaults values to some of the configuration variables which are passed to the engines. Not all engines support all configuration variables.

4.5 Project templates

Bibliography

[1] URL developer.apple.com

[2] URL http://www.tug.org/mactex/

References

Bibliography

[1] S Anza et. al. 2005 *Class. Quantum Grav.* **22** S125

[2] M Armano et. al. 2009 *Class. Quantum Grav.* **26** 094001

[3] F J Massey Jr. 1951 *J. Amer. Statistical Assoc.* **46** 253

[4] M Hewitson et. al. 2009 *Class. Quantum Grav.* **26** 094003

[5] LTPDA: a MATLAB toolbox for accountable and reproducible data analysis http://www.lisa.aei-hannover.de/ltpda

[6] D B Percival and A T Walden 1993 *Spectral Analysis for Physical Applications* (Cambridge: Cambridge University Press) p 290

[7] Kolmogorov A N 1933 *Ist. Ital. Attuari.* **4** 83

[8] N Smirnov 1948 *Ann. Math. Statist.* **19** 279

[9] Miller L H 1956 *J. Amer. Statistical Assoc.* **51** 273

[10] N I Fisher 1983 *Int. Stat. rev.* **51** 25

[11] M B Wilk and R Gnanadesikan 1968 *Biometrika* **55** 1

[12] L Ferraioli et. al. 2010 *Phys. Rev. D* **82** 042001

[13] F J Harris 1978 *Proc. IEEE* **66** 51