



APEX TEXT 1.2

September, 2010

Table of Contents

1.Introduction.....	2
2.Features.....	3
3.What's New in v1.2?.....	4
4.User Guide.....	5
5.Known Bugs.....	17
6.Support.....	18

1. Introduction

ApexText is a product of **Apex Editor** platform. The **Apex Editor** project was initiated in 2006 with a motivation - “Provide efficient and lightweight editors with rich functionalities at free of cost”. In early 2008 decision was taken to open source the editor platform.

ApexText is a general purpose text editor for developers and non-developers. Along with other essential features editor provides multi language support at GUI level. Using configuration file, GUI language can be changed very easily. In version 1.2 of text editor other than English, Hindi (an Indian language) is supported.

Developers can use code template, code completion, document template, programming tools (e.g. source code comment & uncomment, Java class compile & execution etc.), custom tools features of editor to make development process faster.

Switch to **ApexText** today and experience the difference.

2. Features

Following list depicts key features of **ApexText**:

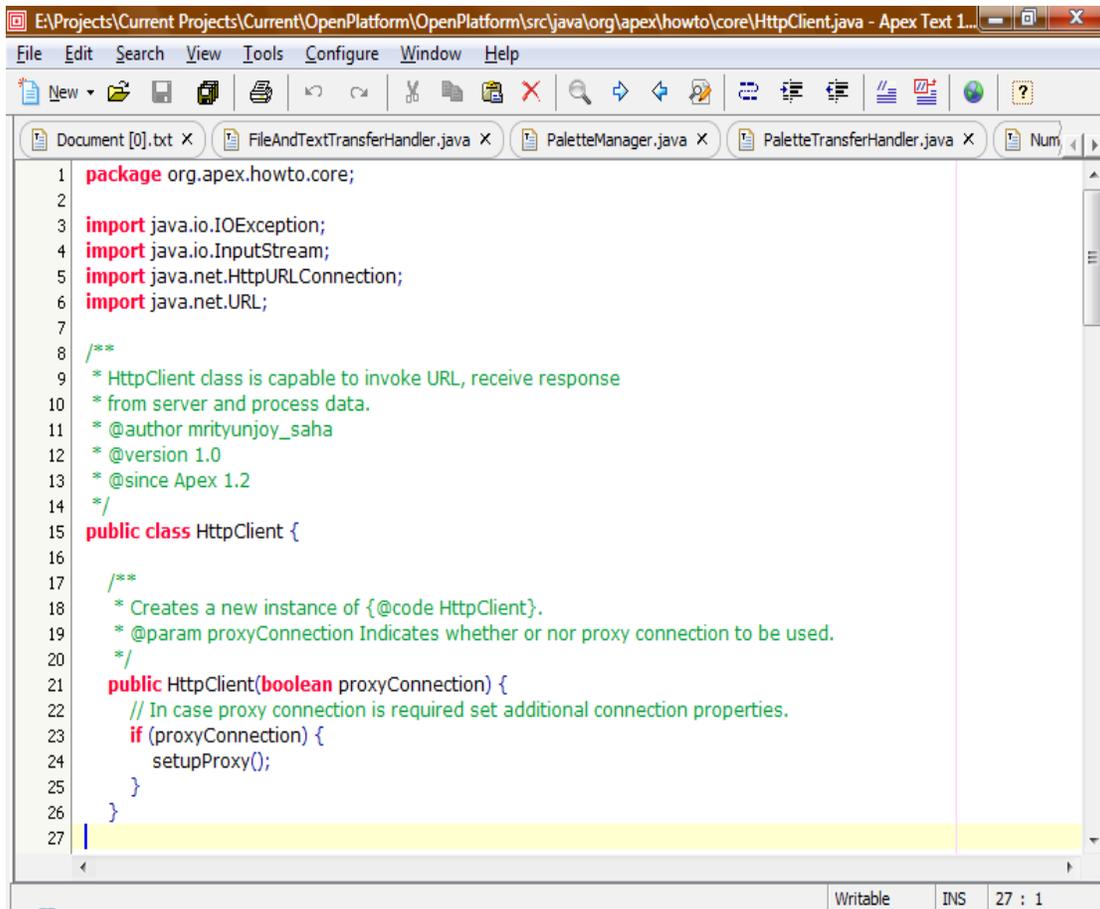
- a) Infinite undo and redo
- b) Syntax highlighting
- c) Code templates
- d) Code completion
- e) Individual tab for each document and close button in each tab
- f) Document templates
- g) Typed document
- h) Document deletion from editor
- i) Renaming document
- j) Move document to a new location
- k) Viewing documents in web browser
- l) Incremental search
- m) Highlighting search results
- n) Efficient way of navigation between documents
- o) Word wrap
- p) Essential tools for Java programming language
- q) Any number of custom tools can be added
- r) Splitting editor window
- s) Document management
- t) Multi language support at GUI level
- u) Sophisticated user preferences setup

The editor can be installed and used in major platforms – Windows, Linux, Mac etc.

3. What's New in v1.2?

The below list contains the key enhancements done in v1.2 of **ApexText**:

- a) Performance improved in editor startup, file open and search
- b) Recent files list maintenance by editor
- c) Indent, outdent facilities for text blocks
- d) Comment, uncomment facilities for programming
- e) Right margin in editor to limit line size
- f) Multiprocess execution support with console menus: start, stop and clear
- g) Word wrap
- h) Preferences window is re-organized and faster
- i) Bug reporting and feature request facilities right from the editor



```
1 package org.apex.howto.core;
2
3 import java.io.IOException;
4 import java.io.InputStream;
5 import java.net.HttpURLConnection;
6 import java.net.URL;
7
8 /**
9  * HttpClient class is capable to invoke URL, receive response
10 * from server and process data.
11 * @author mityunjoy_saha
12 * @version 1.0
13 * @since Apex 1.2
14 */
15 public class HttpClient {
16
17     /**
18     * Creates a new instance of {@code HttpClient}.
19     * @param proxyConnection Indicates whether or nor proxy connection to be used.
20     */
21     public HttpClient(boolean proxyConnection) {
22         // In case proxy connection is required set additional connection properties.
23         if (proxyConnection) {
24             setupProxy();
25         }
26     }
27 }
```

Writable INS 27 : 1

4. User Guide

a) File Operations

In editor new temporary documents can be created in two different ways. The default option is click File >> New menu. For the new temporary document, default extension is 'txt'. However, the default document extension can be changed from Configure >> Preferences >> General >> General Tab. Another way to create a specific type of temporary document (e.g. Java, HTML etc.) is File >> New Type menu.

To open a document from file system or external disk, use File >> Open menu. Also, a file can be dragged from file system and dropped into editor to open it.

To close an already opened document in editor, use File >> Close menu. Also, various sub-menus are available under File menu to close selected file, all files except current file and all opened files.

A document can be deleted from file system permanently using editor's File >> Delete menu. In addition to that a document can be renamed and moved to a desired target location.

b) Recent Files

Editor maintains a list of recently used documents. Whenever a document is closed from editor, it goes to recent files queue. Maximum ten files can be present in recent files queue by default. However, number of recent files count can be changed to a desired value using Configure >> Preferences >> General menu. Old files are automatically removed from the queue. To open a recent file go to File >> Recent File menu and then choose the desired document to open it in the editor.

c) Edit

Cut, copy and paste are basic operations in any text editor. The key point here is, editor supports infinite undo and redo operations. Invert case and change case of text facilities are also available.

In addition to standard edit menus, source code comment & uncomment and indent & outdent facilities are available in **ApexText**. To use these functionalities select a block of text in editor and choose the appropriate sub-menu from Edit root menu. Code comment, uncomment, indent outdent etc. menus are also available in toolbar as shortcuts.

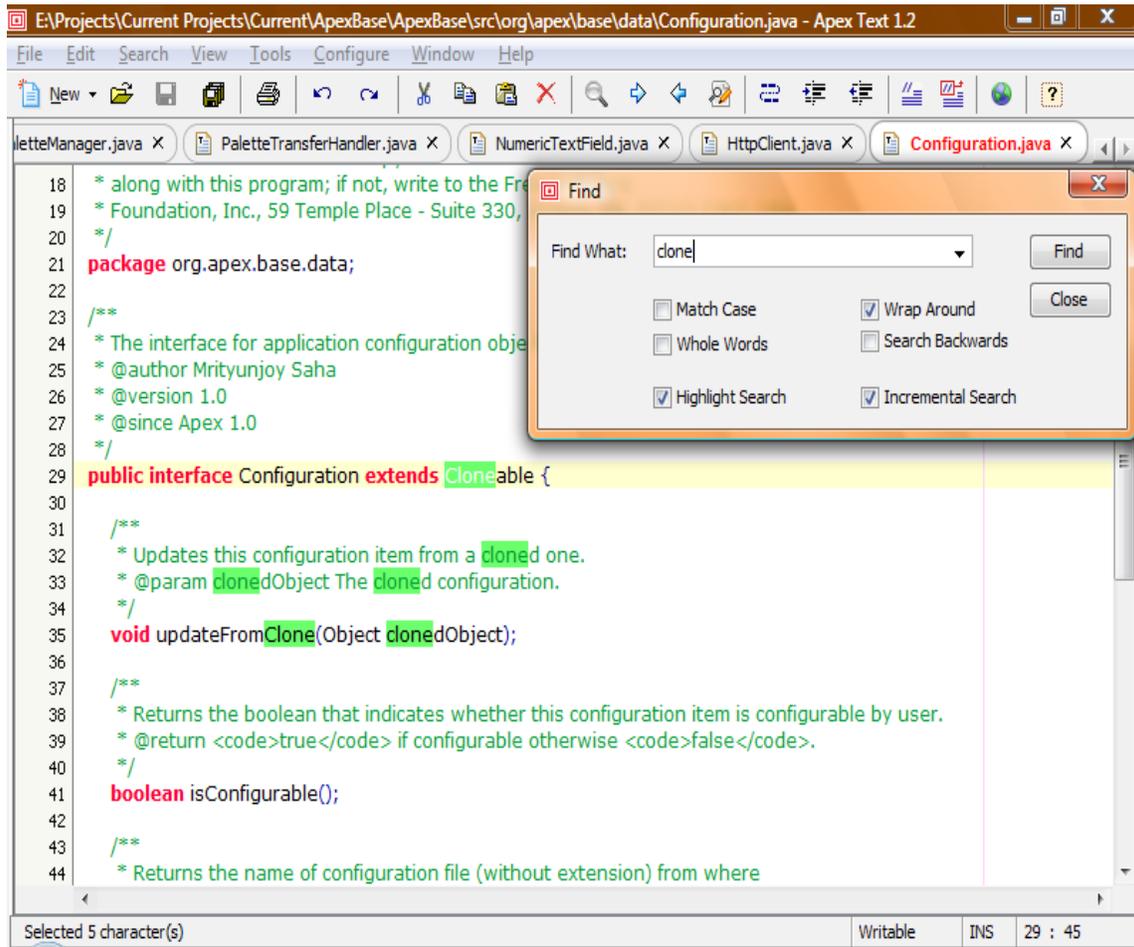
Word wrap feature is now supported. To use word wrap open a file in editor and click Configure >> Word Wrap menu.

d) Find and Replace

Incremental search and highlight search are key features of **ApexText**. Search when each character is typed in search text box, is called incremental search and while searching highlight all search results in the document is referred as highlight search. Find Next, Find Previous and Replace functionalities are also available.

The background color of highlighted search results can be changed from Configure >> Preferences >> Style >> Highlighting Tab.

To go to a desired line number in the document, use Search >> Go To Line.

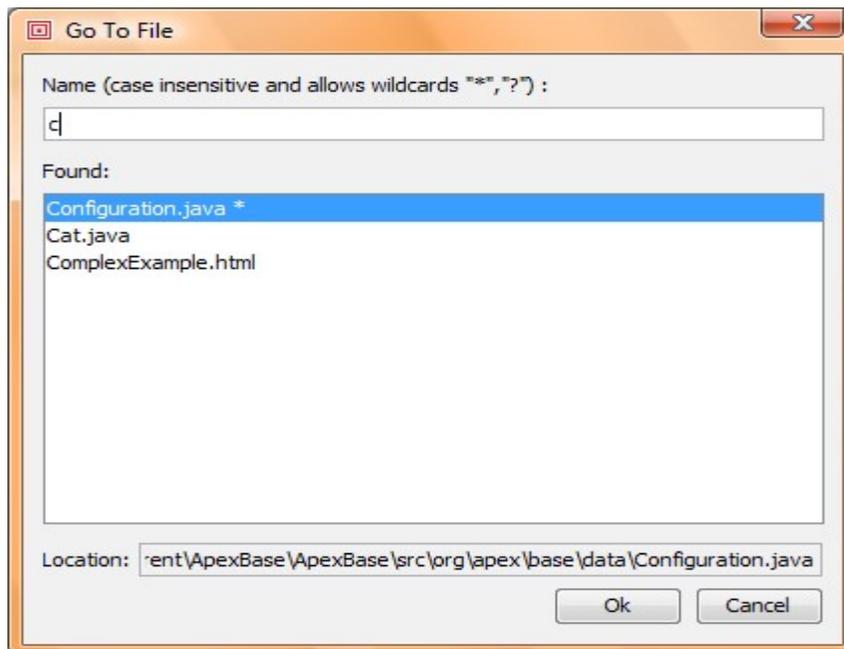
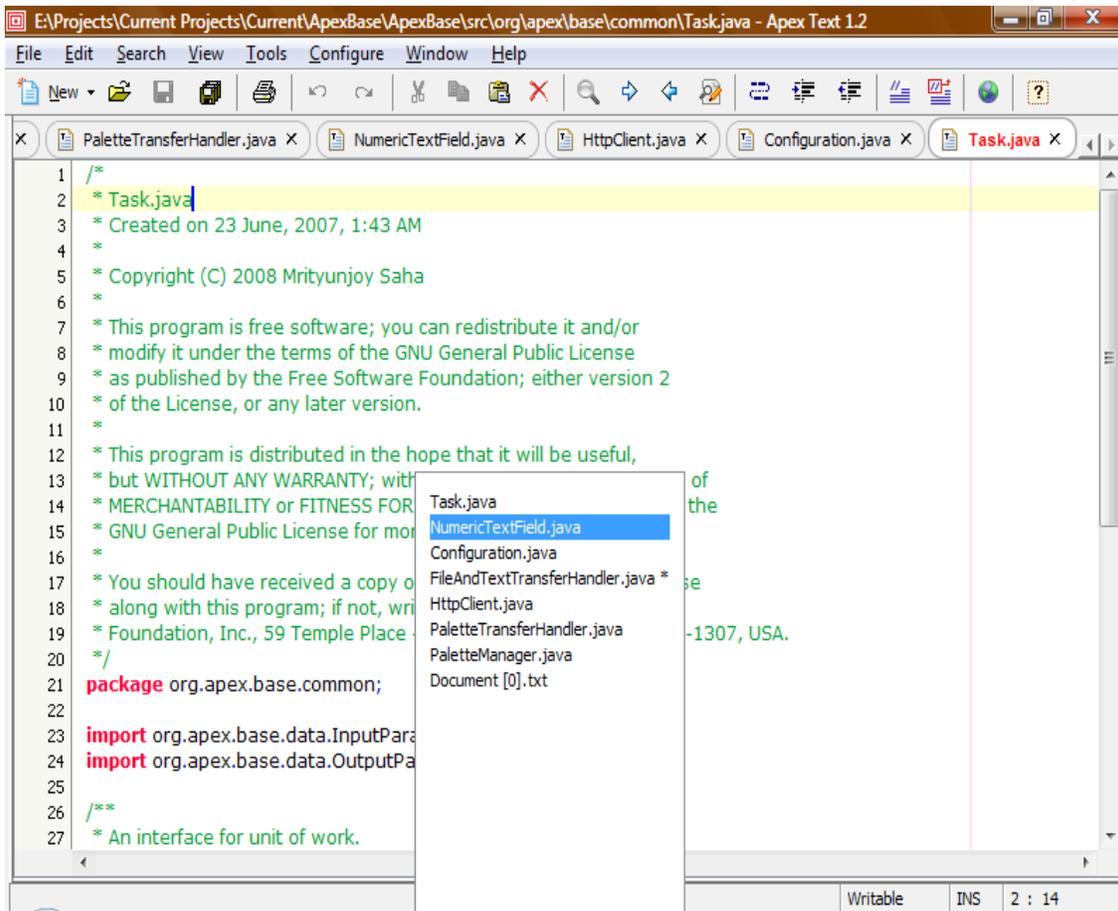


e) Document Navigation

There are two ways to navigate between documents.

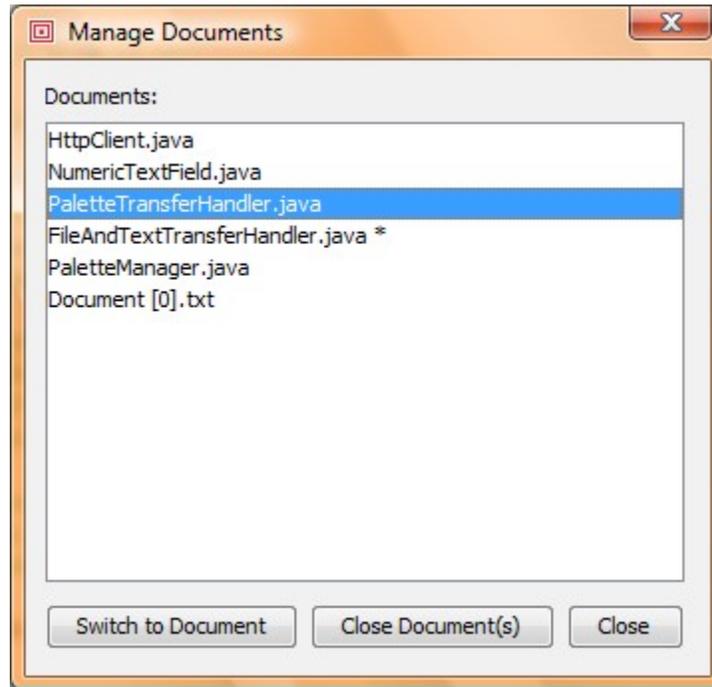
The fastest way is to press 'Ctrl + Tab' to navigate between files. A pop up window containing document names appears. The list is sorted by last accessed time. By holding 'Ctrl' button, 'Tab' button can be pressed multiple times to navigate to desired document. Once 'Ctrl' button is released last selected document in pop up window is displayed in the editor and pop up window disappears.

Another way is to use 'Go To File' sub-menu under 'Search' root menu. On click of this menu item a form appears where user can type few initial characters of target document. While typing each character on search box matching documents are displayed immediately in a list. Right from the search box 'Up' and 'Down' navigation keys can be used to navigate to desired document in the list. Here, incremental search (search as user types) and wild card search facilities are provided.



f) Manage Documents

Documents in the editor can be managed from Window >> Manage Documents. On this screen a list of opened documents are displayed and two options are provided – switch to a particular document and close one or multiple documents.



g) Tools

Few useful tools for Java developers are provided with editor. However, any number of custom tools can be added from Configure >> Preferences >> Tools >> Custom Tab. A custom tool can be a command line tool or GUI based tool.

Visibility of editor provided tools can be controlled from Configure >> Preferences >> Tools >> Provided Tab. This is very useful in case some of the editor provided tools are not related to your work.

For both editor provided and custom tools runtime options and parameters can be set from Configure >> Preferences >> Tools >> Provided or Custom Tab.

Multiple tools can be executed at the same time. Different consoles are opened automatically for different processes in output window. The console window can be maximized or minimized by double clicking the window title bar. In every console, there menus are available:

SL No.	Option	Description
1	Run or Re-Run	To run the last process or tool.

2	Stop	To stop a running process or tool.
3	Clear	To clear the console window.

Alternatively, to run an executable file, open the file in editor and click Tools >> Run menu.

To stop one or more running processes, use Tools >> Stop Run menu. The dialog window shows a list of running processes. Choose one or more processes from the list and click the Stop button.

```

1 package org.apex.howto.swing;
2
3 import javax.swing.JFrame;
4 import javax.swing.JLabel;
5 import javax.swing.JPanel;
6 import javax.swing.JTextField;
7 import javax.swing.SwingUtilities;
8 import javax.swing.text.AbstractDocument;
9 import javax.swing.text.AttributeSet;
10 import javax.swing.text.BadLocationException;
11 import javax.swing.text.DocumentFilter;
12 import javax.swing.text.DocumentFilter.FilterBypass;
13
14 /**
15  * NumericTextField allows only numbers in the text field and not even decimal
16  * points. Also, it restricts the length of input.
17  * @author mityunjoy_saha
18  * @version 1.0
19  * @since Apex 1.2
20  */
21 public class NumericTextField {

```

Output - Java Tools >> Compile - [javac, org/apex/howto/swing/NumericTextField.java]

```

[Java Tools >> Compile] tool executing...
[Java Tools >> Compile] tool execution finished.

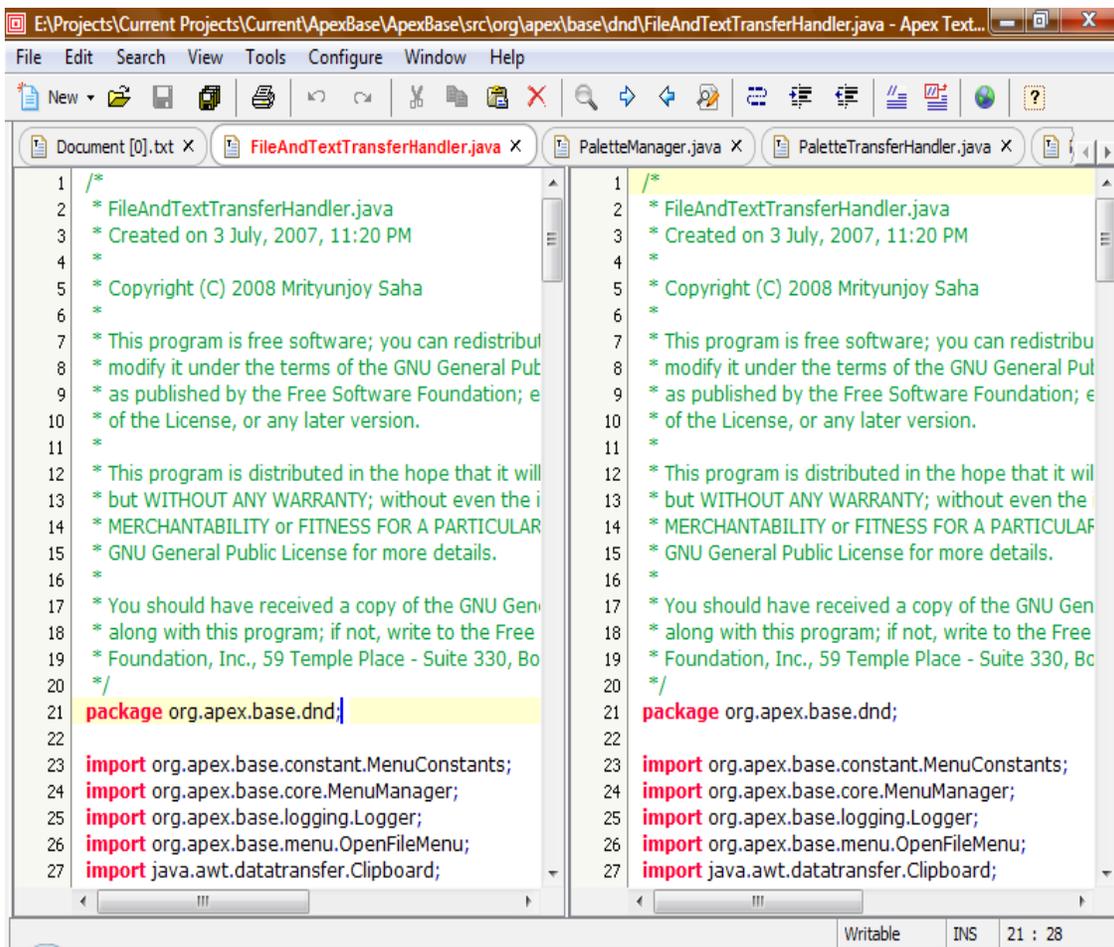
```

Console 1 x

Writable INS 14 : 4

h) Split Window

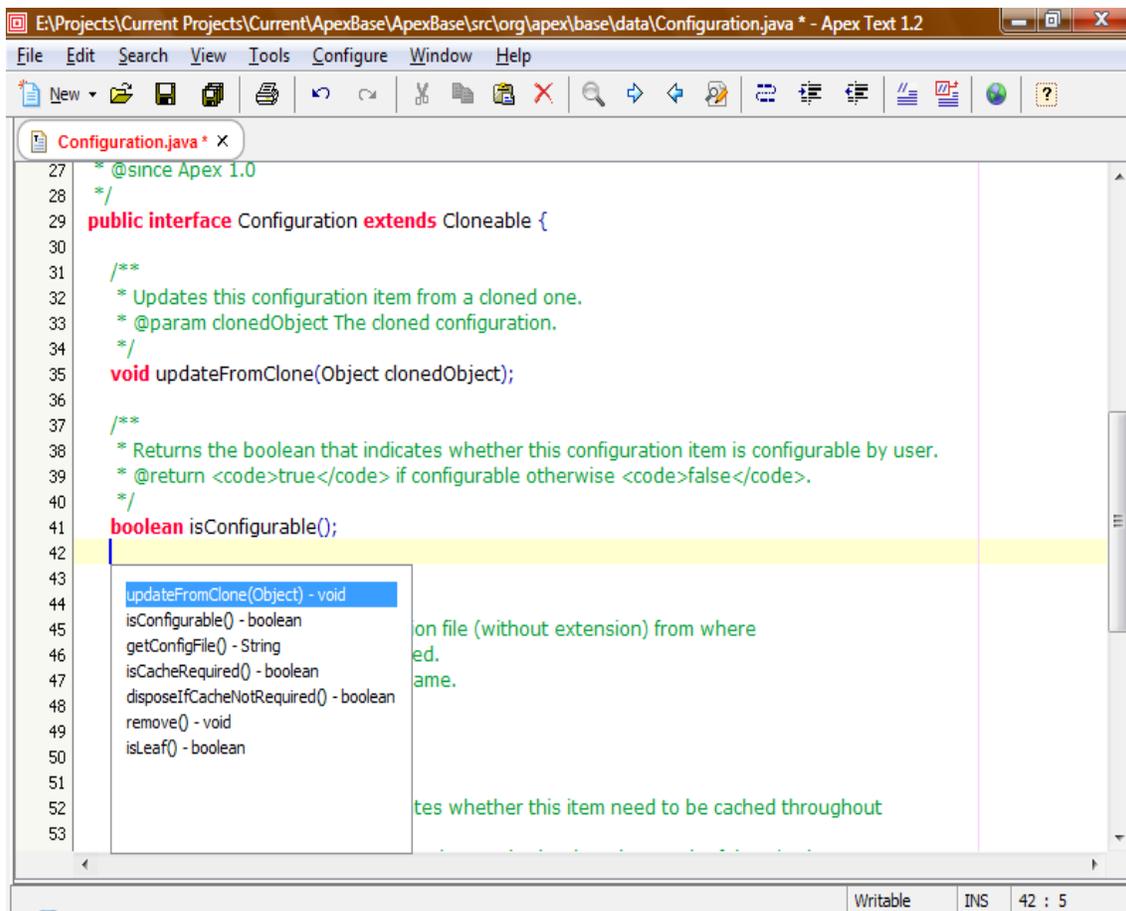
Editor window can be split horizontally or vertically. The split status is maintained at document level and not for the editor. Split window is very useful when a large document is being edited and a section of same document needs to be referred.



i) Code Templates and Code Completion

Code templates are used to replace an abbreviation by a template. For example “apx” can be used as abbreviation for text “Welcome to world of Apex Editor platform”. Any number of code templates can be added from Configure >> Preferences >> General >> Code Template Tab. The key to expand code template is 'Ctrl + Enter'.

Code completion is a feature available in modern IDEs to assist developers by providing code hints. In **ApexText**, code completion feature is enabled for Java documents only. After opening a Java document in editor press 'Ctrl + Space' key to see available code hints. Once code completion pop up is opened use 'Up' and 'Down' arrow keys to navigate through the list and press 'Enter' key to choose a hint.



j) Document Templates

For a document type, template can be set from Configure >> Preferences >> Document Template. Whenever a new document is created in editor, for the corresponding document type if a template exists, document is created in editor with template text.

k) Syntax Highlighting

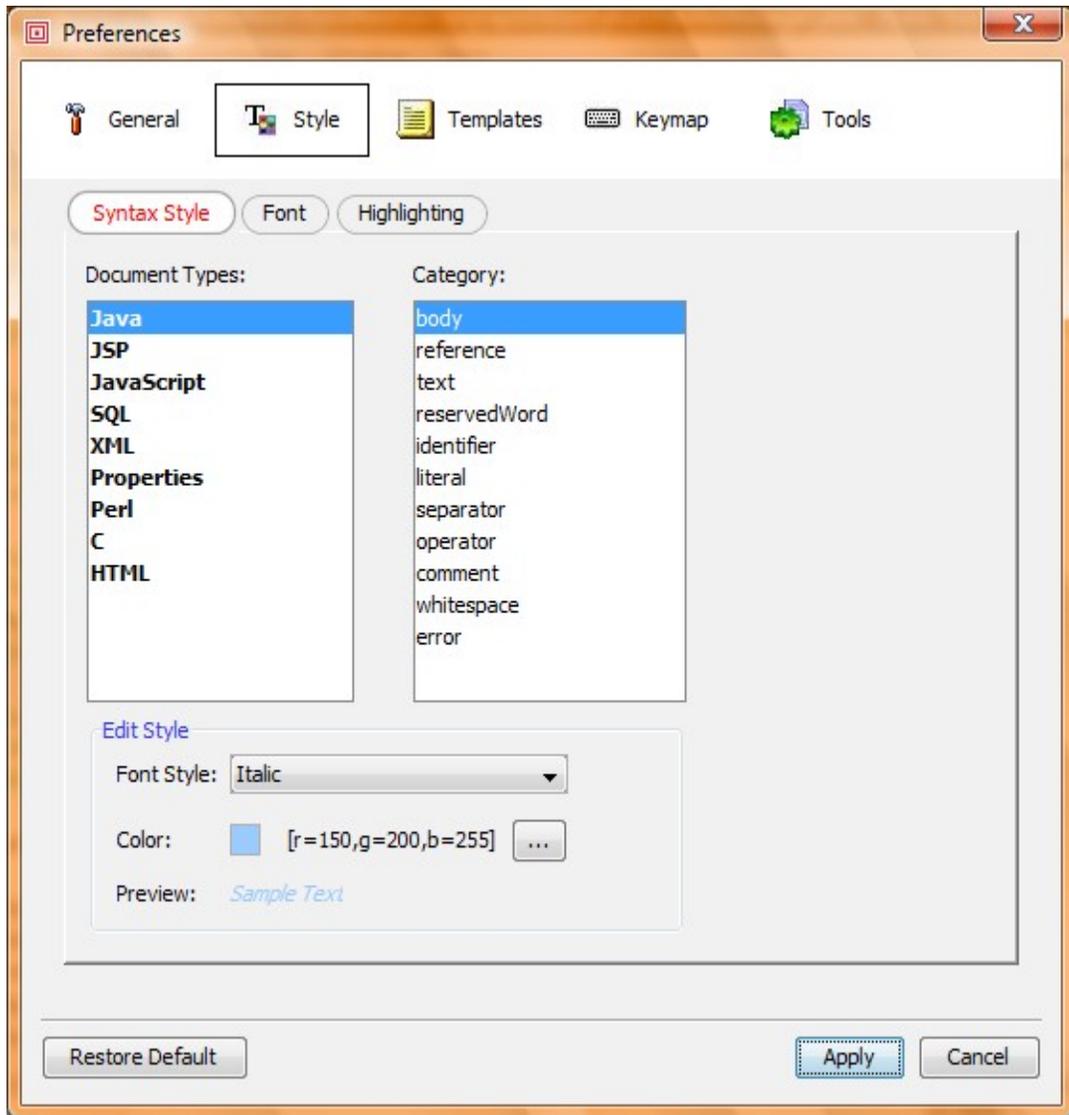
By default syntax highlighting is enabled and default styles are set for different document types. Styles can be changed from Configure >> Preferences >> Style >> Syntax Style Tab.

l) Preferences

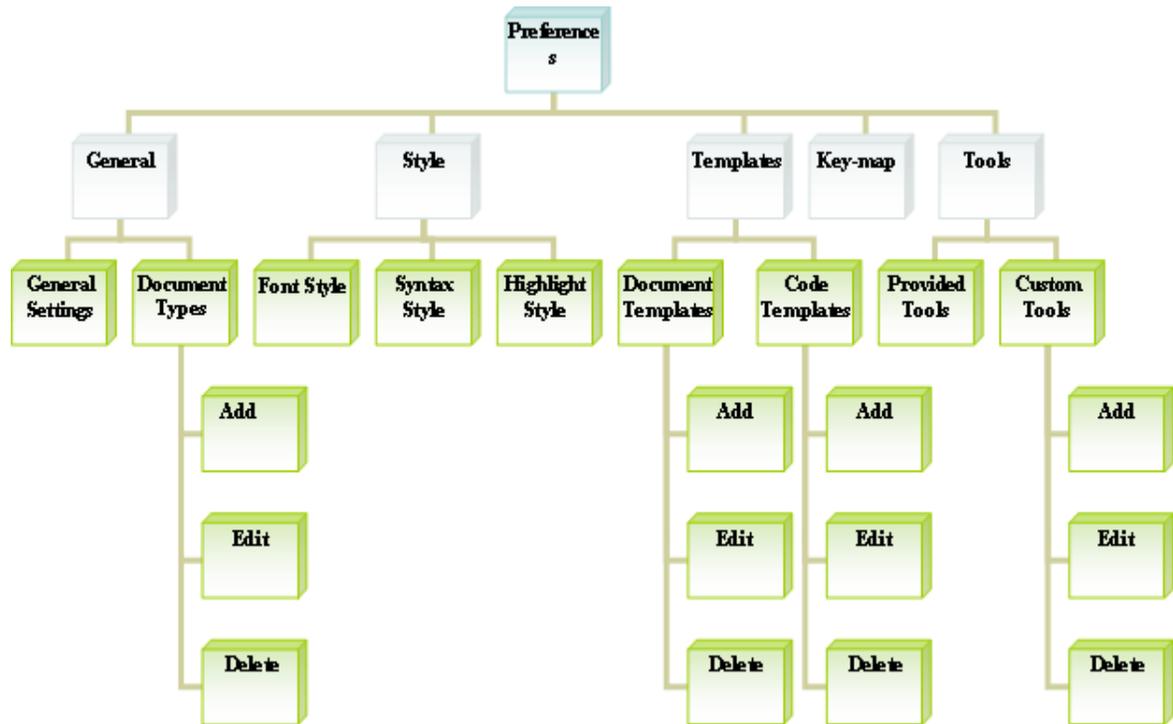
Preferences menu is available under Configure root menu. Using this interface many features of editor can be customized to suit user's choice. Also, editing experience can be improved by using custom tools, key map, code templates and document templates features. Any number of templates can be created and saved.

Options available in user preferences at a glance:

SL No.	Menu	Description
1	General >> General	Global settings, cursor, formatting and right margin configuration
2	General >> Document Types	File extensions management and document types maintenance
3	Style >> Syntax Style	Syntax style maintenance for different tokens in different programming languages
4	Style >> Font	Global font settings for all documents
5	Style >> Highlighting	Highlight style settings for various categories such as search results, line numbers, selected tab etc.
6	Templates >> Document Templates	Document templates can be edited and added using this menu. When a new document is created template of the selected document type is used.
7	Templates >> Code Templates	Code templates can be added and edited using this menu. For example, 'hello' abbreviation can be added for template 'Hello World!!!'. After typing 'hello' in editor, the code can be expanded to display code template 'Hello World' by pressing 'Ctrl + Enter'.
8	Keymap >> Keymap	Shortcut keys can be assigned and deassigned to editor menus.
9	Tools >> Provided	Options for editor default tools can be managed using this menu.
10	Tools >> Custom	Custom tools can be maintained using this menu. Any executable, third party application, command can be added as custom tool.



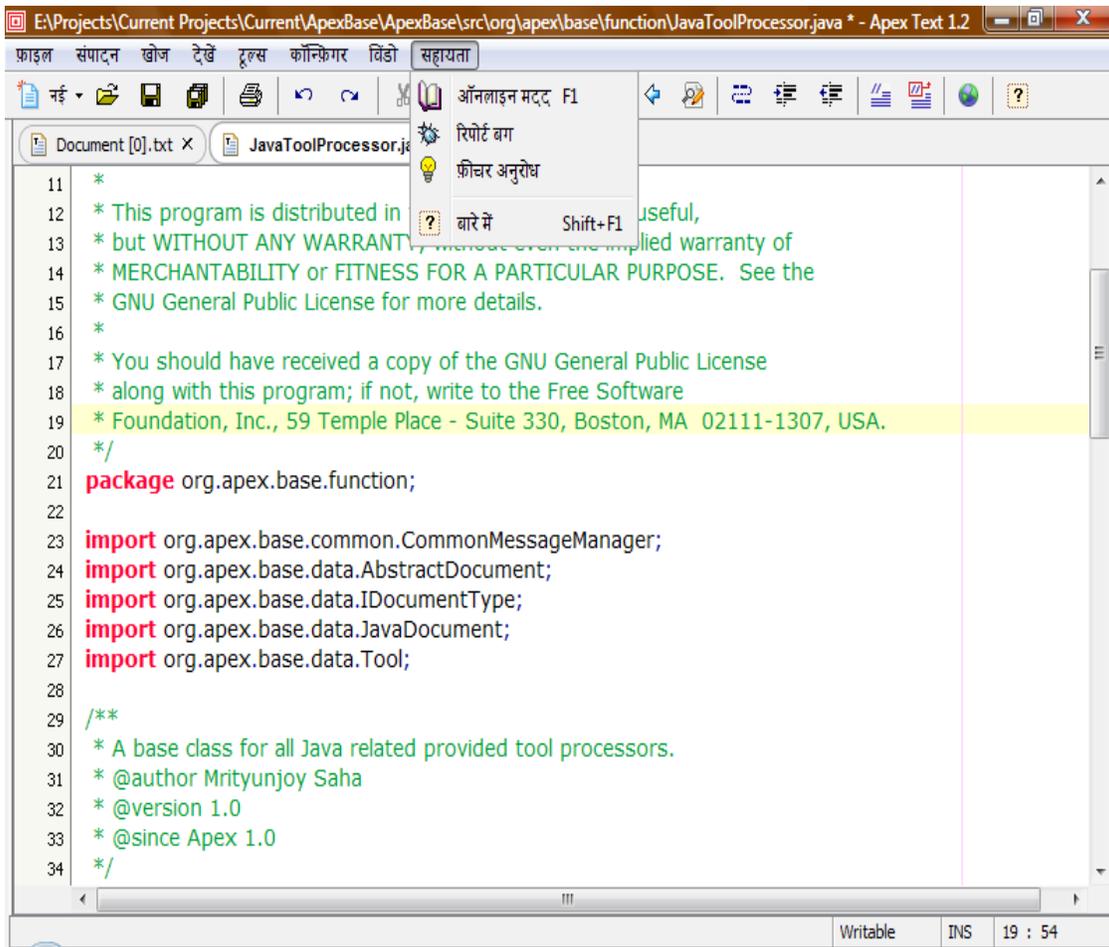
Following diagram shows a quick view of what all things can be controlled from preferences interface:



m) GUI Language Change

GUI language can be changed by modifying following key in {INSTALLATION_DIRECTORY}/config.xml:

SL No.	Key	Value	Description
1	apex.lang	en or english	GUI in English language.
		hi or hindi	GUI in Hindi language.



n) Enable Trace

Trace is controlled at various levels e.g. error, warning, info. By default trace is disabled at every level. However, trace can be enabled by changing following keys in {INSTALLATION_DIRECTORY}/config.xml:

SL No.	Key	Value	Description
1	apex.info	true/false	Enable/Disable information logging.
2	apex.warning	true/false	Enable/Disable warning logging.
3	apex.error	true/false	Enable/Disable error logging.
4	apex.debug	true/false	Enable/Disable debugging. A master key which controls information, warning and error logging.
5	apex.log.dir	A directory in file system.	The directory where log files to be created.

o) Menu-Key bindings

Key bindings are used to execute a menu in editor without navigating the menu hierarchy. Key bindings can be changed from Configure >> Preferences >> Menu-Key Binding. The default menu-key bindings are listed in below table:

SL No.	Menu	Accelerator Key
1	File >> New	Ctrl + N
2	File >> Open	Ctrl + O
3	File >> Save	Ctrl + S
4	File >> Close	Ctrl + D
5	File >> Print	Ctrl + P
6	File >> Exit	Ctrl + X
7	Edit >> Undo	Ctrl + Z
8	Edit >> Redo	Ctrl + Y
9	Edit >> Cut	Ctrl + X
10	Edit >> Copy	Ctrl + C
11	Edit >> Paste	Ctrl + V
12	Edit >> Select All	Ctrl + A
13	Search >> Find	F5
14	Search >> Find Next	Ctrl + F
15	Search >> Find Previous	Ctrl + Shift + F
16	Search >> Replace	F9
17	Search >> Go To Line	Ctrl + L
18	Search >> Go To File	Ctrl + G
19	Configure >> Preferences	Ctrl + Shift + P
20	Tools >> Run	Ctrl + R
21	Tools >> Stop Run	Ctrl + Shift + R
22	Window >> Output	Ctrl + Shift + W
23	Help >> Online Help	F1
24	Help >> About	Shift + F1

p) JRE Requirement

To use the editor, minimum JRE 1.6 is required. It is recommended that, Windows users download the editor setup distributed with JRE. However, the required JRE can be downloaded even after **ApexText** installation from www.java.com.

5. Known Bugs

The text editor is having following top known bugs and these will be fixed in upcoming releases:

- a) The editor becomes slow when a large file is opened.
- b) In printed pages top and bottom content sometimes appears incomplete. And header, footer, margin etc. setting missing.
- c) Indentation lost when an indented text block pasted in editor.

6. Support

Right from the editor bugs can be reported and enhancement requests can be submitted. These sub-menus are available in editor under Help menu.

To report a bug or to request a new feature please visit www.openapex.org/text/Support.php. And to receive immediate support please drop a mail to: apextext@gmail.com.