

JChecker3.0 Operating Manual



Artman21 Inc.

JChecker1.0 July 1998

JChecker2.0.1 November 1999

JChecker2.0.2 December 1999

JChecker2.1.1 January 2000

JChecker3.0 March 2002

JChecker3.0.3 November 2002

JChecker3.0.4 February 2003

JChecker3.0.5 June 2003

Index

Introduction.....	4
Features of JChecker.....	5
Hardware Requirement.....	6
Distribution, Use and Copyrights.....	6
User Support.....	7
Chapter 1 Getting Started	
1.1 How to Install.....	8
1.2 Starting up JChecker2.1.....	10
1.3 Checking HTML validation.....	11
1.4 Revision Number of JChecker2.1.....	13
Chapter 2 The Basis of HTML validation	
2.1 DTD and the Dictionary.....	14
2.2 HTML Validation Services on the Web.....	15
Chapter 3 Preferences	
3.1 Editor.....	16
3.2 Coloring.....	19
3.3 Beautify.....	20
3.4 Validate.....	22
3.5 Link.....	23
3.6 Messages.....	25
3.7 Browser.....	26
3.8 Auto Correction.....	28
Chapter 4 File Operation	
4.1 Creating New HTML.....	29
4.2 Open local HTML.....	30
4.3 Open remote HTML.....	31
4.4 Accessing Recently Opened Files.....	31
4.5 HTML Templates.....	32
4.6 Closing documents.....	32
4.7 Saving documents.....	32
4.8 Reverting documents.....	33
4.9 Saving error messages.....	33
4.10 Printing HTML documents.....	33
Chapter 5 Working with FTP Clients	
5.1 Settings.....	34
5.2 How to Use.....	37
Chapter 6 Editing	
6.1 Operation of the Editing Window.....	38
6.2 Keyboard Shortcuts.....	41
6.3 Undo and Redo.....	42
6.4 Copy, Cut and Paste.....	42
6.5 Drag & Drop.....	42
6.6 Selecting tags, pair tags, elements.....	43
6.7 Removing tags.....	43
6.8 Tools.....	43
6.9 Paste Date and Time.....	44
Chapter 7 Search	
7.1 Finding Text.....	45
7.2 Regular Expressions.....	48
7.3 Replace.....	50
7.4 Multi-File Search & Replace.....	51
7.5 Consulting dictionaries.....	55
7.6 Mark.....	56
7.7 Mark List.....	56
7.8 Jump by Line Number.....	58
Chapter 8 Check	
8.1 Validating HTML.....	59
8.2 Checking Links.....	60

8.3 Checking Both.....	61
8.4 Checking Links Rcursively.....	61
8.5 Auto Correction.....	62
8.6 Beautifying HTML.....	63
8.7 Minimizing HTML.....	63
8.8 Re-coloring HTML.....	64
Chapter 9 Working with Other Editors	
9.1 Check using Word Services.....	65
9.2 Check using Services Menu.....	67
Chapter 10 Dictionaries	
10.1 Dictionary Window.....	68
10.2 Generating Dictionary (DtdToDic).....	70
10.3 The Structure of the Dictionary File.....	72
10.4 Customizing the Dictionary.....	74
Chapter 11 Previewing with Browsers	
11.1 Registering Browsers.....	76
11.2 Previewing with Browsers.....	77
Chapter 12 Window	
12.1 Macro Window.....	78
12.2 Color Pallet.....	79
12.3 Bring All to Front.....	79
12.4 Switch.....	79
12.5 Tile.....	79
12.6 Zoom.....	80
Chapter 13 Macro	
13.1 Macro Menu.....	81
13.2 Macro Window.....	84
13.3 Contextual Menus.....	85
Appendix 1 Attached Macro.....	86
Appendix 2 Error and Warning Messages.....	91
Appendix 3 AppleScript.....	100

Introduction

The popularity explosion of the Internet has spawned a new generation of individuals and organizations who, either out of personal or business motivation, have set out to challenge this new techno-creative environment; creating their own web-sites and home-pages. Catering to this phenomenal wave of home-page design interest have come various home-page editors, most of which are based on WYSIWYG technology. Editors of this sort are being marketed for use by beginners and professionals alike.

However, all is not so smooth in editor land. Drawback in the actual use of commercial WYSIWYG packages often leaves the novice user quickly frustrated with editors that are slow and complex to use. Often a desired result, ends in undesirable frustration when a newly designed page neither looks nor operates according to plan. Stray tags may suddenly appear seemingly out of nowhere in the text, or even a seemingly rudiment layout may differ drastically depending on the browser used to view it.

With the coming of HTML experience, most users switch to the editing of their HTML source code directly using a text editor, in addition to using their WYSIWYG editor.

Yet, when editing source code directly, the user must become savvy to things like the spelling of elements and attributes, hierarchical relations of tags, existence of end tags, tag order, and an array of other easily overlooked detail.

An answer to continuing frustration? An HTML checker.

As much as a spell checker is essential to word processing, HTML checkers, which check for HTML validation or lost links, are a necessary tool for direct editing.

In July 1998, JChecker1.0 was introduced as an HTML checker to work in tangent with text editors such as our Jedit. JChecker2.0 was later released in November 1999 and JChecker2.1, which supported link checking, was released in January 2000.

Today JChecker3.0 is not only a checker but has been reborn as a powerful HTML source code editor supporting both the Mac OS Classic and OS X environment.

As an HTML validation checker, JChecker3.0 supports 16 HTML/XHTML validation dictionaries, auto-generation for DTD dictionaries, auto HTML error correction, powerful link checking including sub linked pages, and much more.

As a text editor, it inherits powerful editing functions from its brother Jedit, including regular expression search, AppleScriptability, and enhanced macro functions. It also supports many original functions such as a tag editing macro, a tag help macro, and a services menu under OS X.

We hope that JChecker3.0 will become a reliable partner giving you flexibility, freedom, and accuracy in composing your next HTML text.

My personal thanks goes to k16 (Keiichiro Ishino) who has granted us use of his DTDs for Netscape Navigator and Internet Explorer, and also to Liam Quinn and Yasutaka Kato of the Web Design Group who allow JChecker to link to their HTML help web-pages. Their help and friendship has been invaluable in the undertaking of this project.

March 2002
MATSUMOTO Satoshi

Features of JChecker

- Supports both Mac OS Classic and Mac OS X, working as a native Carbon application under Mac OS X.
- Supports 28 character code sets such as Western, Japanese, Chinese, Korean, Cyrillic, UTF-8, UTF-16, among others.
- Includes powerful editing functions like: regular expression search, AppleScripting, a macro menu function, drag & drop, line number display, HTML coloring, multiple Undo, a context menu, inline Japanese input, and other functions, each derived from JChecker's brother "Jedit."
- Supports powerful tag editing and helper macros which cover HTML4.0 specifications. With these macros being written in AppleScript, you can easily modify and extend each.
- Checks over sixty different items such as hierarchical relation of tags, matching end tags, order of tags, types of attribute values and spelling, all based on the HTML dictionary you specify. JChecker supports 17 dictionaries such as HTML2.0, HTML3.2, HTML4.0, XHTML1.0, XHTML1.1, Compact HTML1.0, and XHTML Basic1.0
- Checks existence of linked pages or files, supporting both remote and local link checking. JChecker also supports root and branch checking which checks all pages linked to the root page.
- Supports auto correction of 20 different HTML errors. JChecker corrects simple errors automatically.
- Supports beautifying of HTML text. Beautifying lets you determine the amount of indentation related to tag hierarchy, or the deletion of line feed codes. In the beautifying Preferences, you can set beautifying options for each type of tag.
- Supports minimizing HTML texts. Minimizing can reduce the size of HTML texts by stripping needless spaces, tabs, and line breaks.
- Includes an application "DtdToDic" which automatically generates the JChecker dictionary from the DTD. For any new DTD, an additional dictionary menu may be added.
- Supports Browser preview functions. You can preview your HTML with as many different kinds of browsers as you desire.
- Supports image tag/anchor tag auto-generation, done by dropping an image file/HTML file icon directly onto the editing window,
- Supports the External Editor Protocol, Ftp-clients such as Interarchy, NetFinder, Transmit, or Fetch can each be assigned an "Edit with JChecker" menu command. When you select a file in a remote directory and choose this command, the file is downloaded and opened in the JChecker window. When you save the document in JChecker, the file is automatically uploaded back to the remote machine by the ftp client.
- Compatible with Word Service protocols. JChecker can check HTML documents in other applications which are compatible with a Word Service such as Jedit4.0.

Hardware Requirements

- (1) System: MacOS 8.6 or higher. JChecker3.0 is compatible with MacOS 8.6-9.2 and Mac OS X.
- (2) 4MB of RAM is required for JChecker3.0 (System memory is not included).

Distribution, Use and Copyrights

- (1) The copyright of JChecker3.0 is registered to MATSUMOTO Satoshi (satoshi@artman21.co.jp.)
- (2) You may not alter JChecker3.0 in part or in whole without the expressed permission of the author or copyright holder.
- (3) JChecker3.0 is shareware. If you continue to use JChecker3.0 after a testing period of twenty days, you are required to register. Please read the "Registration" file for information on how to register.
- (4) The author takes no responsibility for damages resulting from the use of JChecker3.0. Note too that the user license of JChecker3.0 may not be leased or transferred to a third party.
- (5) JChecker may be copied and distributed freely through the Internet or a BBS. However, any and all contents and/or composition of JChecker files may not be modified. If you wish to bundle JChecker commercially or distribute it on CD-ROM, then prior, written permission from the author is required.
- (6) The "WASTE Text Engine" used in JChecker3.0 is copyrighted by Marco Piovanelli. (WASTE Text Engine (c) 1993-2002 by Marco Piovanelli, All Rights Reserved.)
- (7) DTD data prepared by Mr. Keiichiro Ishino for Internet Explorer and Netscape Navigator has been included in making the corresponding JChecker dictionaries.

User Support

JChecker3.0 will be updated regularly. We welcome your comments, bug reports, and/or any other suggestions. Please contact us at:

Attention: Artman21 Inc. #User Support Division

[mail to:support@artman21.co.jp](mailto:support@artman21.co.jp)

TEL/FAX 81-466-24-1028

16-11, Minami Fujisawa, Fujisawa, Kanagawa, Japan

2510055

The newest version of JChecker is available at the following web site:

<http://www.artman21.net/>

Chapter 1 Getting Started

1.1 How To Install

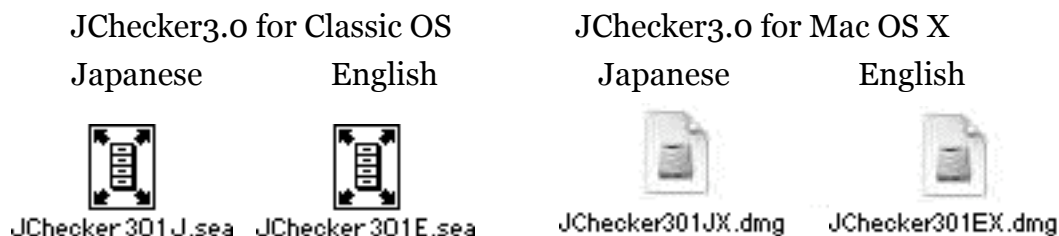
JChecker3.0 is provided as a compressed file through Internet sites, bulletin board services, and CD-ROMs supplied with magazines. You can also download the latest version from our following web site:

<http://www.artman21.net/>

There are two program versions for JChecker3.0 depending on the Operating System.

For Mac OS 8.1-9.2	JChecker3.0 For Classic OS (non-Carbon version)
For Mac OS X	JChecker3.0 For Mac OS X (Carbonized version)

Both English and Japanese versions are available for each version.



For each file, "301" represents the revision number. The file name you see may or may not contain the number "301".

"JChecker3.0 For Classic OS" is supplied as a file compressed in the CompactPro™ Self-Expanding Archive format. You can expand the file by double clicking on its icon or by using expansion software such as CompactPro or StuffIt Expander™. This will create a "Jedit4.0 Folder" icon, which you can then place wherever you wish on your hard drive.

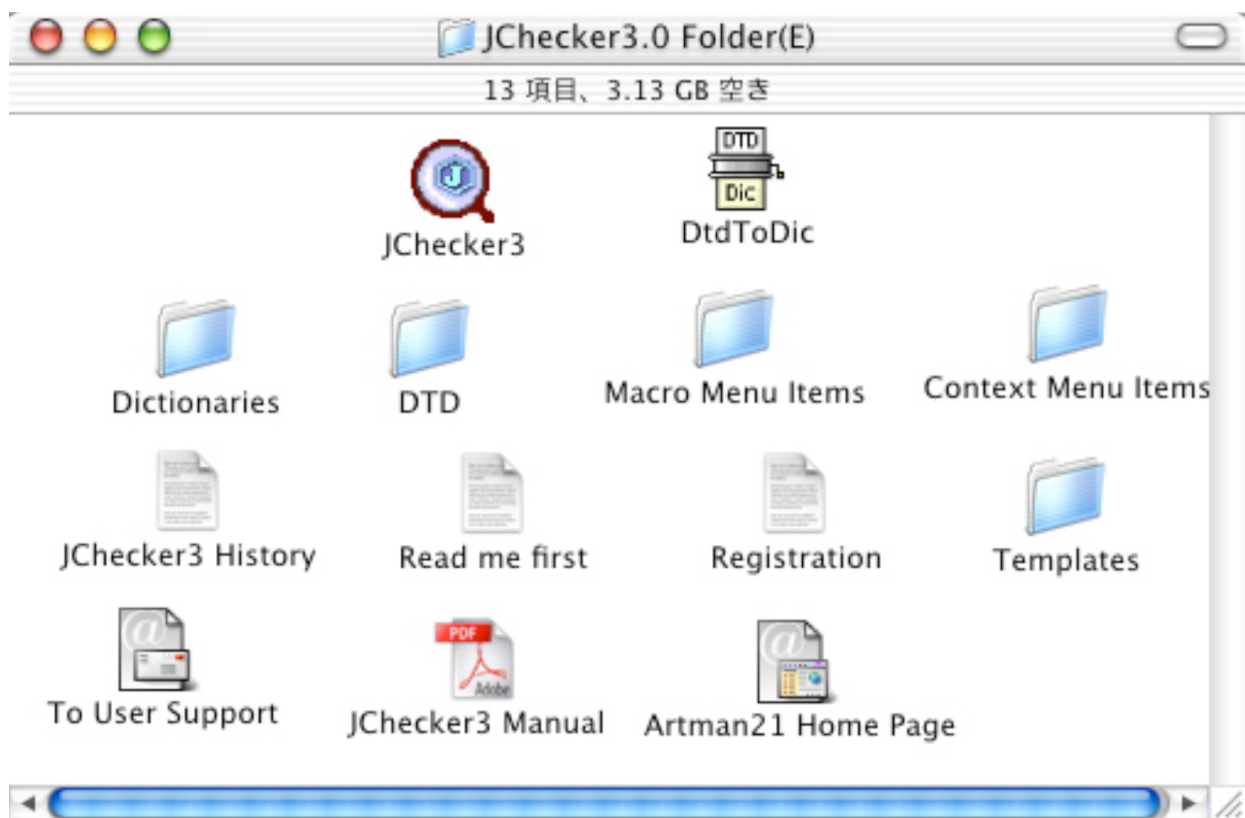
"JChecker3.0 For Mac OS X" is supplied as a dmg file compressed in gzip. It can be expanded with the use of "StuffIt Expander." After expanding the JChecker file, locate and double-click the dmg file to mount and drag-copy the "JChecker3.0 Folder" into the "Applications" folder found under Mac OS X.

If you are using OS X 10.2 (Jaguar) or later, you need to replace the original macros to the new ones specially made for OS X 10.2. You can download the macros for OS X 10.2 from the following URL.

http://www.artman21.net/product/jchecker/index_E.html

More detail for the macro replacement, refer the document "Read me For Macros 10.2" downloaded.

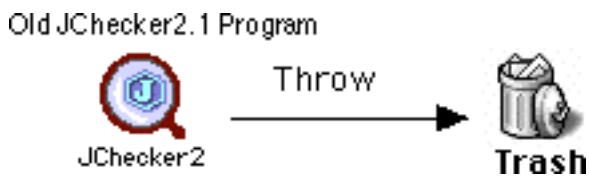
The contents of the "JChecker3.0 Folder" are shown below:



Once this folder and its contents are created, installation is complete. (Note that no files are installed in the system folder.)

Warning: Because of revisions, contents within the JChecker3.0 Folder may differ from those shown above.

If you are replacing JChecker2.1 with JChecker3.0 please be sure to delete the old JChecker2.1 program file from your system.



Failure to remove JChecker2.1 may cause JChecker2.1 to be undesireably invoked when a JChecker3.0 document is double-clicked.

1.2 Starting Up JChecker3.0

To start-up JChecker3.0, simply double click the program icon "JChecker3" found in the JChecker3.0 folder



If you have not yet registered JChecker3.0, a registration dialog as shown below will appear during the startup process.

Please Register

 JChecker3.0

copyright(c)1998-2002
MATSUMOTO Satoshi
All right Reserved.

JChecker is shareware. If you are still using JChecker after 20 days, please register. You can purchase and register on-line via our web site.
<http://www.artman21.net/>

For more details, please read the file "Registration". This dialog will not appear after the registration.

User Name :

Registration code :

Use on trial basis

Register

Registration can be conducted directly online at the Artman21 Inc. web site. Access to our web site can be obtained by clicking the URL address (<http://www.artman21.net/>) found in the register dialog. For more information see the "Registration" file.

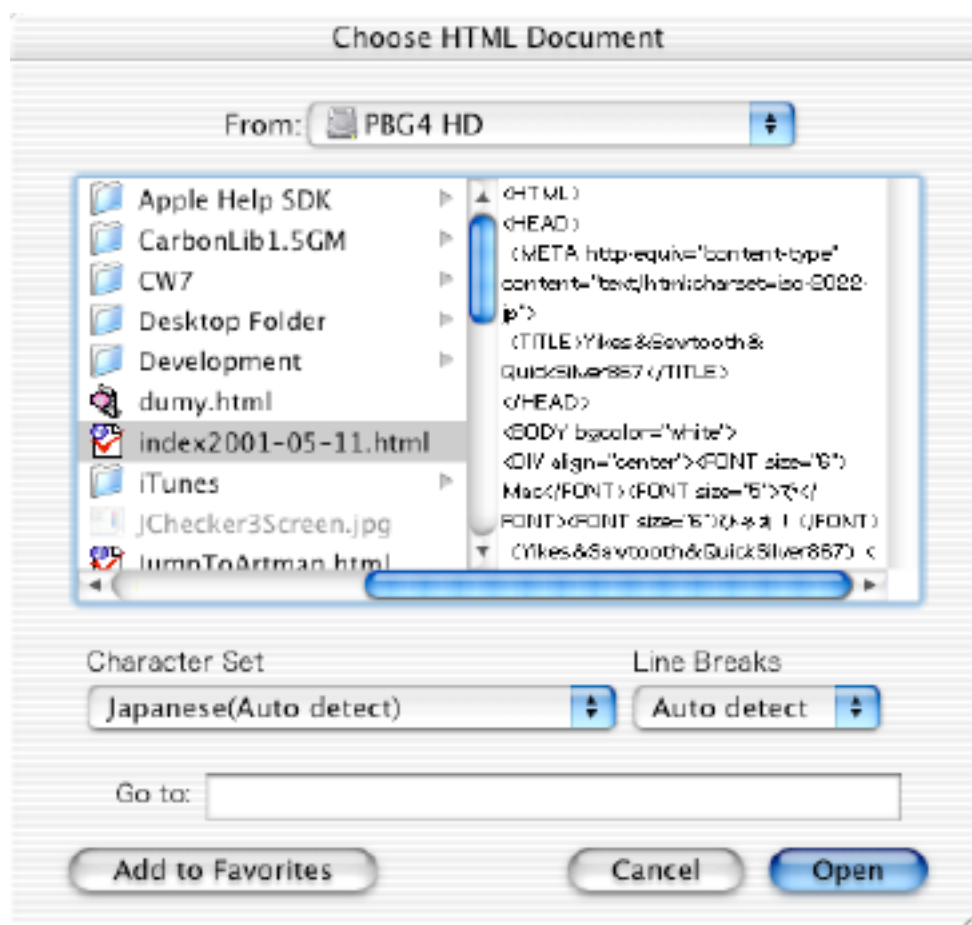
After you register, a user name and registration code will be sent by e-mail to you. Please enter your user name and registration code exactly as written, and click "Register" in the registration dialog. After you register, the registration dialog will no longer appear.

If you simply want to test drive JChecker3.0 on a trial basis without registering, click "Use on trial basis."

After a 20 day trial period, a dialog informing you that the trial period has ended will appear each time you check a document. If you decide against using JChecker or do not wish to register, drag the "JChecker3.0 Folder" to the trash and it will be deleted from your hard drive and system.

1.3 Checking HTML Validation

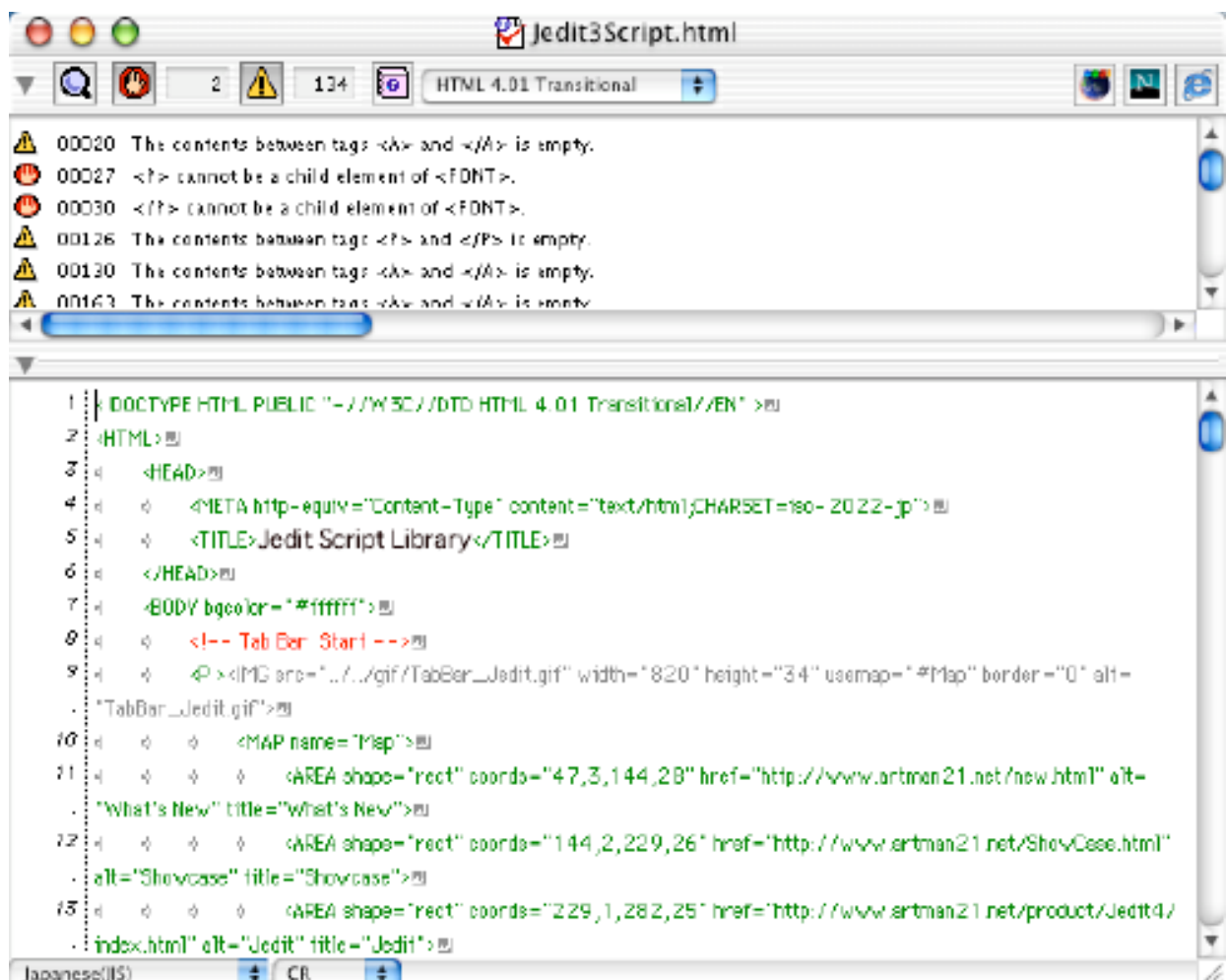
Select "Open..." from the "File" menu to open a HTML file. The following Open File dialog box will appear.



JChecker3.0 supports 28 character code sets currently used on the Internet such as Western, Japanese, Chinese, UTF-16, UTF-8 etc. It is also compatible with three variations of line breaks: CR(Mac), LF(Unix), CR+LF(DOS).

If you wish to open HTML files in Japanese, set the pop up-menu "Character Set" to "Japanese (Auto detect)". Next, set the pop up-menu "Line Breaks" to "Auto detect". JChecker will then automatically detect the appropriate character set and type of line breaks.

After specifying the HTML file type, click the "Open" button. The contents of the HTML file will be displayed in the lower partition, or editor window.



The results of an HTML validation check will be displayed in the upper partition, or message window.

Line numbers followed either by error  or warning  signs are displayed in the message window. Clicking a specific error message in the message window automatically highlights the appropriate line of the source text in the editor window.

By double clicking an error message, a pop up window is displayed containing the explanation of the error.

Validating HTML for the first time, may surprisingly result in a vast number of errors and warnings detected by JChecker. For questions concerning the validity of the multitude of displayed errors and warnings, please consult "Chapter 2: The Basis of HTML validation Check".

Unlike errors in a program's code, errors in HTML will not cause your system to crash. However errors in HTML can cause your pages to be displayed or operate incorrectly and thus requires correction.

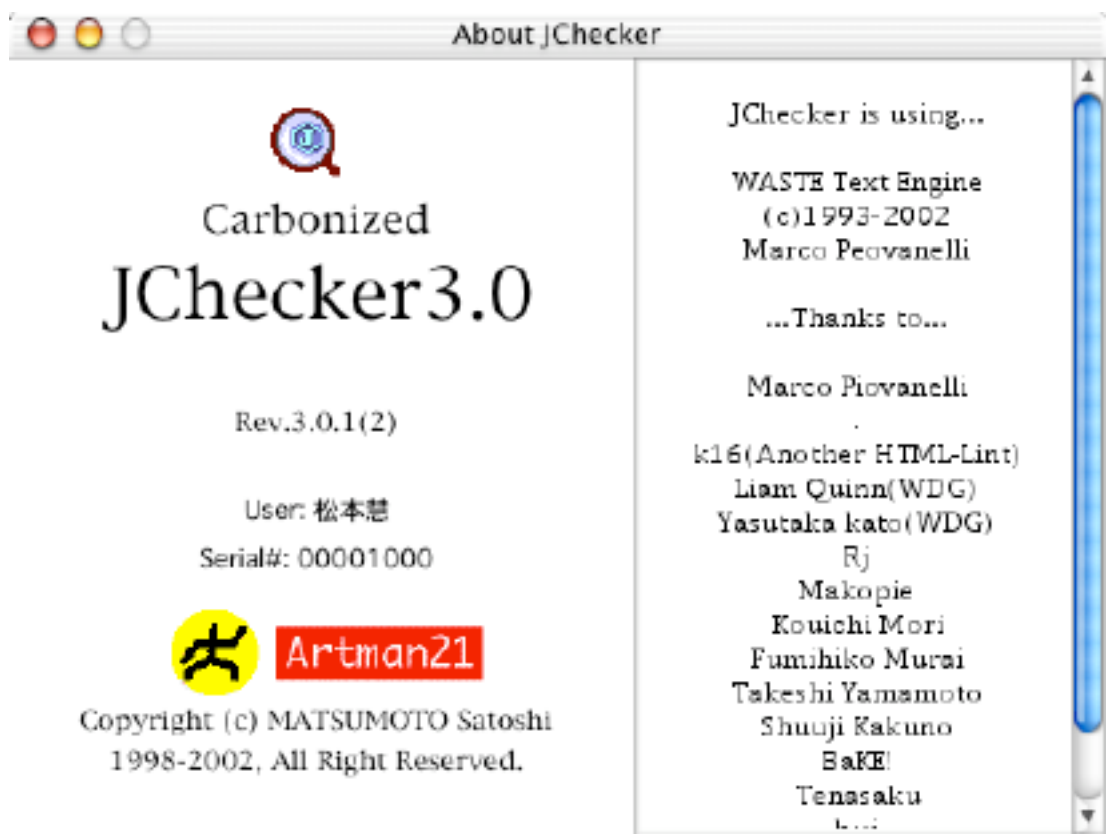
For more on errors and messages, please see "Appendix 2: Error and Warning Messages"

You will need a sufficient understanding of HTML to correct the lines which JChecker indicates as errors or warnings. As this manual does not specifically address information about HTML, it is advisable to have access to a good HTML guide book.

1.4 Revision Number of JChecker3.0

The current version of JChecker can be checked by selecting the "About JChecker " menu under the Apple menu or JChecker menu in Mac OS X. A window, as shown below, will be displayed.

In the following, the version of JChecker is for Mac OS X and its revision number is "3.0.1(2)."



An "***Unregistered***" denotation will be displayed in place of the user name if JChecker remains unregistered. Your name and serial number will be displayed when the registration process is complete.

Note: The serial number displayed is not the same as the registration code that you input during registration.

Chapter 2 The Basis of HTML Validation

Users may be surprised by the number of errors and warnings displayed when an HTML validation check with JChecker3.0 is conducted for the first time. This chapter deals with the basis of these errors and warnings.

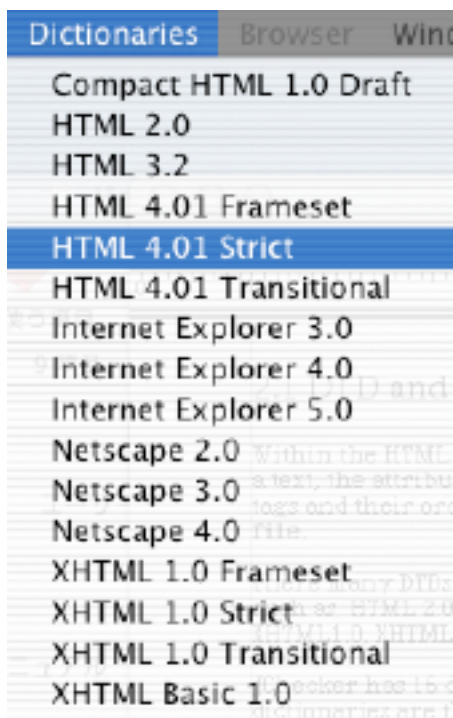
2.1 DTD and the Dictionary

Within the HTML(Hyper Text Markup Language), standards (such as tag types in a text, attributes which each tag supports, and hierarchical relationship between tags and their order of appearance) are set by the **DTD (Document Type Definition) file**.

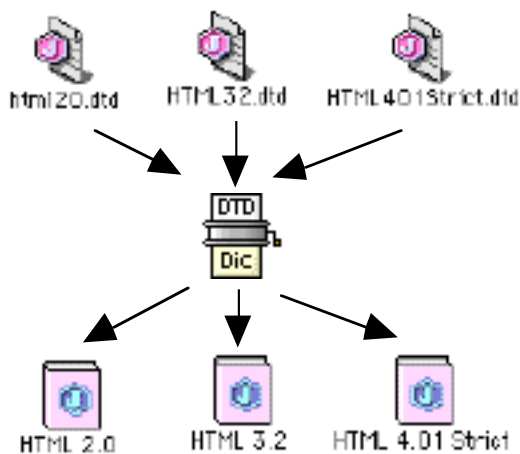
Many DTDs exist which may differ depending on the browser or version of HTML, such as HTML 2.0/3.2/4.0, Netscape Navigator 2.0/3.0/4.0, Internet Explorer 3.0/4.0/5.0, XHTML1.0, XHTML1.1, XHTML Basic1.0 and so on.

JChecker utilizes 16 dictionaries which are generated from these DTD files. These dictionaries are the basis of errors and warning messages which JChecker3.0 outputs.

By selecting the menu "Dictionaries", you can find a list of those dictionaries that JChecker supports.



These dictionaries were generated by the attached program "DtdToDic" in "JChecker3.0 folder".



If adding a new DTD, you are able to create a new dictionary by using "DtdToDic", adding it to the "Dictionaries" menu. If you customize a DTD, you can also make a corresponding dictionary

by using "DtdToDic".

Since no official DTD files for Netscape Navigator and Internet Explorer exist, JChecker dictionaries for these browsers have been assembled from DTD files prepared by Mr. Keiichiro Ishino, who operates a well-known web site "Another HTML-lint", part of an on-line HTML validation service in Japan.

Often, GUI HTML editors such as Adobe Page Mill may turn out a surprising number of errors and warnings that are displayed when you check HTML text. What this signifies usually is that certain parts of the text do not agree with the DTD. Again, you do not need to be concerned that these errors will cause your system to crash. However, what will happen is that these erroneous sections may be interpreted differently or even be ignored by a browser, depending on the type or version of that browser.

2.2 HTML Validation Services on the Web.

Apart from JChecker3.0 (which operates as a self contained application on your Macintosh), there are a few free recommended HTML validation services available on the Web. These, like JChecker3.0, conduct checks based on DTD. If ever there may be some doubt with the analysis done by JChecker3.0, a 'second opinion' check may be worthwhile. (The following two sites are considered the most authoritative.)

<http://validator.w3.org/>

This service is supplied by the W3C (World Wide Web Consortium), publisher of HTML standards. When the URL of a specific web page is entered, it will conduct a validation check of that page.

<http://openlab.ring.gr.jp/k16/htmlint/htmlint.html>

This HTML validation service (offered in Japanese only) is designed and maintained by Mr. Keiichiro Ishino (k16). It provides a very precise validation check. Useful information concerning HTML validation may also be obtained at this site.

Chapter 3 Preferences

Preferences may be set for each of JChecker functions: Editor, Coloring, Beautify, Validate, Link, Messages, Browsers, among others, by going under "Preferences." Preference changes are saved in the "JChecker3.0 Prefs" file located in the "Preferences" folder in the "System Folder."

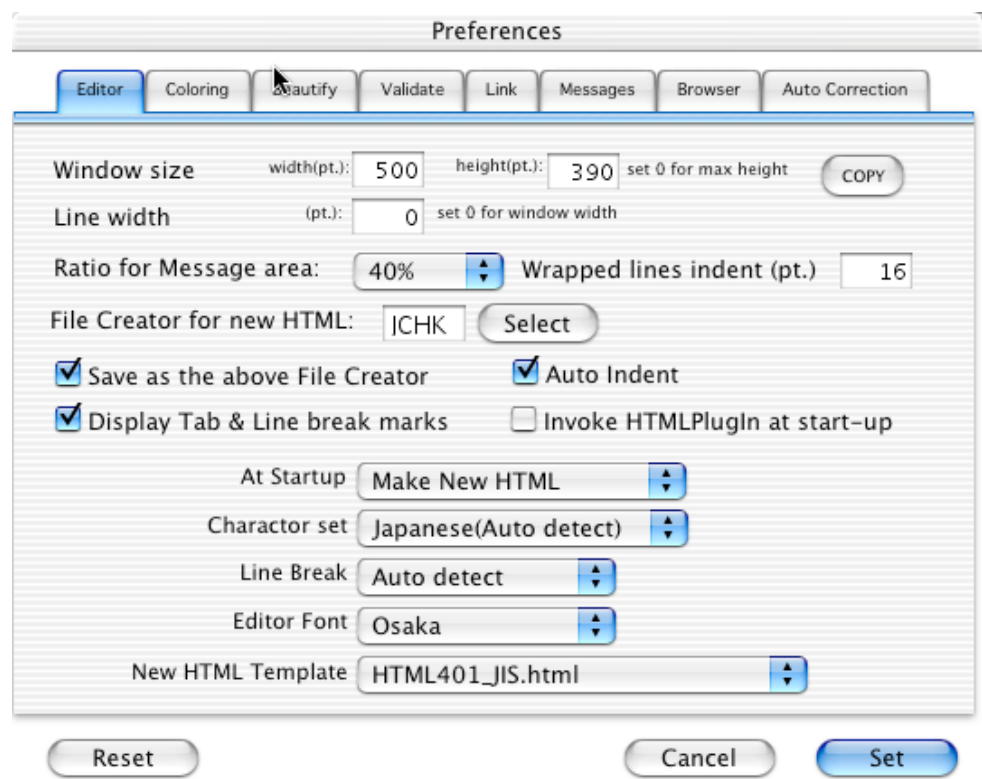


Under Mac OS X, the preferences file is located along the following path.

"~/Library/Preferences/JChecker3.0 Prefs"

3.1 Editor

Select "Preferences..." from the "File" menu under Classic OS or in the "JChecker3" menu under OS X. The dialog box shown below will appear. Click the "Editor" tab if it has not already been chosen.



Window Size

Specifies the height and width of a new editing window. To use the same height and width as the current document, press the "COPY" button. If you enter zero under "height," any new editing window displayed will adjust itself to the maximum height possible as dictated by your screen.

Line Width

Specifies the width of lines in the unit of points, jurisdiction the width of a new editing window. If you enter zero under "Line width", the width of the line will follow the width of your window. If you enter a number other than zero, the line width will remain constant even when adjusting the size of the window.

Ratio of Message Area

Specifies the ratio of the height of the window's message area using a % value. If you select 0%, the window will be opened with the message area closed.

Wrapped Lines Indent

Specifes a point value of wrapped lines indent.

File Creator for New HTML

Icons for new HTML files may be specified by choosing the file creator. For instance, if you set the file creator to "JCHK", the text will be saved with a JChecker document icon. If you are uncertain of the 4 byte code name (for example "JCHK") of your chosen application, click the "Select" button, which will allow you to set the file creator string using the select file dialog:



☒ "Save As Above File Creator"

If you turn this check box on, the file's creator will be changed when the file is saved. If you set the file creator to "JCHK" and check this box, all the HTML files checked by JChecker will be displayed with the following icon:



Checked and cleaned contents of an HTML file may be changed back to its original 'dirty state' when they are reopened by GUI HTML editors (such as Page Mill or Claris Homepage). However, if you save the checked HTML as JChecker documents, you can preserve all the changes made to such corrected file.

☒ "Display TAB and Line Break Marks"

By selecting this check box, tab and line break marks will be displayed. This can be useful in checking the hierarchic structures of the elements.

☒ "Auto Indent"

By selecting this check box, each new line will follow the the same indentation pattern as in a previous paragraph.

☒ "Invoke HTML Plugin at Start-up"

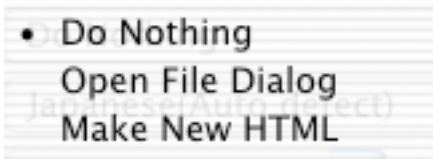
By selecting this check box, the HTMLPlugin will be invoked at the time JChecker starts-up. Convenient, allowing the load time of macros, which utilize the HTML plugin, to be accelerated.



HTMLPlugin

"At Startup" Pop Up Menu

You can specify what action JChecker undertakes when starting up. Three kinds of actions are available: "Do nothing", "Open File Dialog" and "Make New HTML".



"Character Set" Pop Up Menu

You can specify the character code set of an HTML file to be opened. If you select "Japanese (Auto detect)", JChecker will choose the proper kanji character set from "shift-JIS", "JIS", "EUC" and "UTF-8".

"Line Break" Pop Up Menu

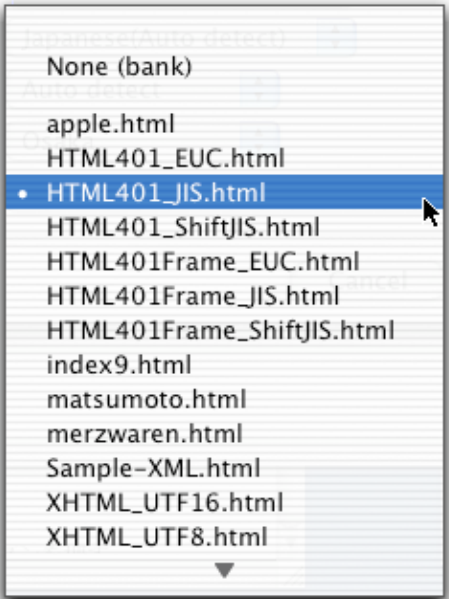
Here, line break specifications of the HTML file to be opened may be made. If you select "Auto detect", JChecker will choose a proper type of line break from the following: "CR(Mac)", "LF(Unix)" and "CR+LF(Dos)".

"Editor font" Pop Up Menu

Here, the editor's font may be selected. The editor font needs to match the character code set. For example, if you specify a western font but the character code set of the HTML document is Japanese Shift-JIS, the kanji characters in the HTML will not be properly displayed.

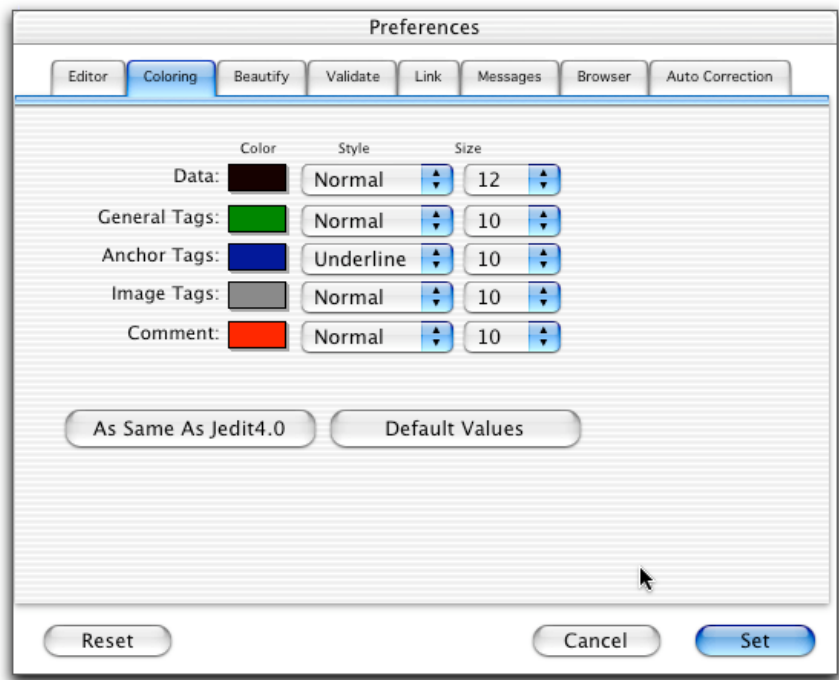
"New HTML Template" Pop Up Menu

Here, you can specify the template HTML file for the menu "New HTML". If you select the item "None (blank)", a blank editing window will be opened when you select the menu "New HTML". The template HTML files are stored in the folder "Templates".



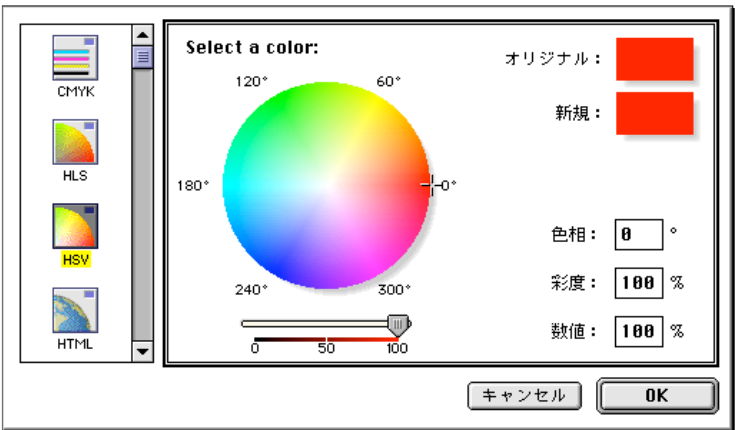
3.2 Coloring

By selecting "Preferences..." under the "File" menu found in Classic OS, or from the "JChecker3" menu in OS X, the following dialog appears. Click the "Coloring" tab if it has not already been chosen.



You can specify different colors, styles and font sizes for each "Data", "General Tag", "Anchor Tag <A>", "Image Tag ", and "Comment Tag <!-- xxx--> ". These settings help make an HTML source program easier to understand.

To specify a character's color, click the color button. A color palette dialog will appear as shown below.

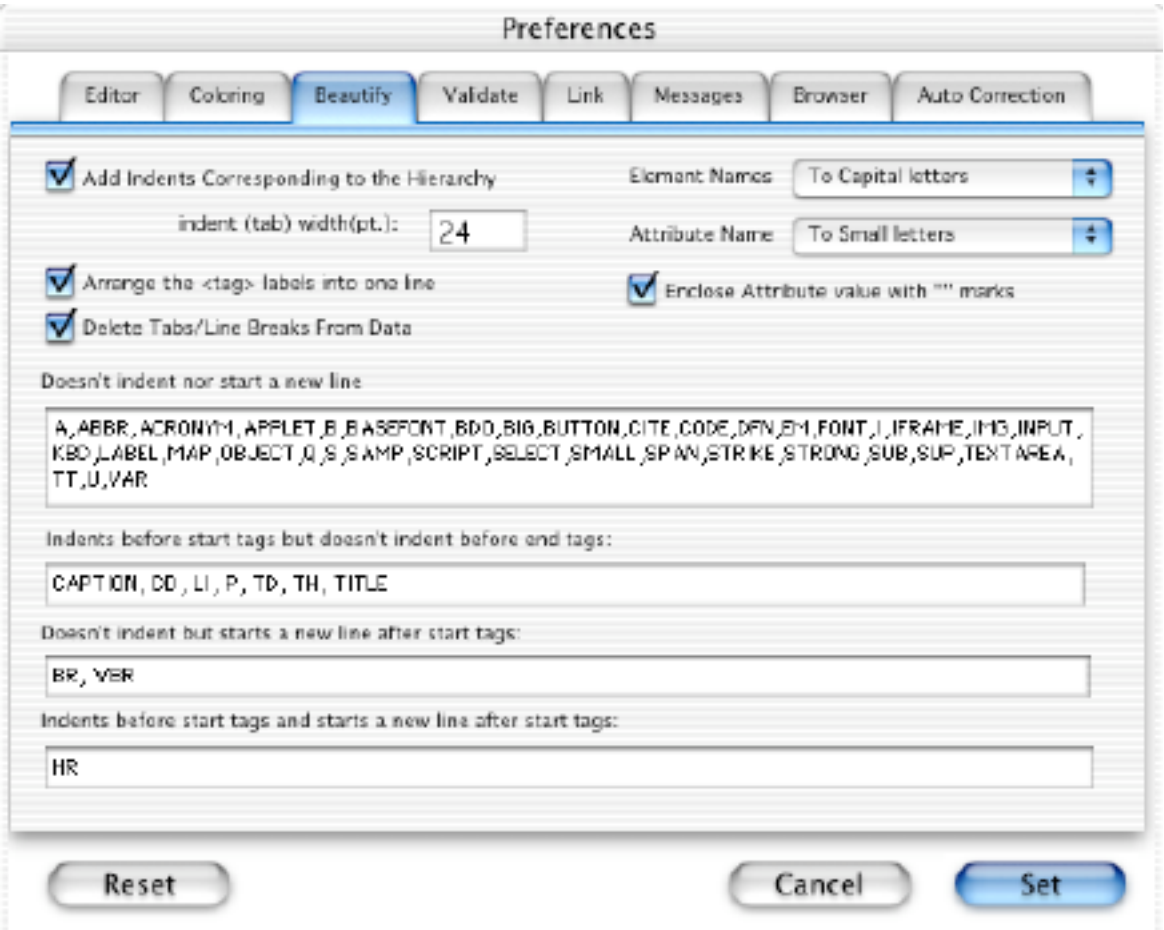


If you use the "Jedit4.0" editor, you can click the button "Same as Jedit4.0" to use same colors, styles, and font sizes specified within the "Jedit4.0 Pref" preference file .

To reset to the default valus of the factory settigs, press the button "Default Values."

3.3 Beautify

By selecting "Preferences..." under the "File" menu found in Classic OS, or from the "JChecker3" menu in OS X, the following dialog appears. Click the "Beautify" tab if it has not already been chosen.



- ☒ "Add Indents Corresponding to the Hierarchy"
Sentences written in HTML are essentially in a hierarchical structure. Under an <HTML> tag are <HEAD> and <BODY> tags. Further, child tags such as <HR>, and <TABLE> tags appear under <BODY> tags. This check box specifies whether such hierarchy will be displayed using or not using indentation. Tabs will then be added before tags according to their location in the hierarchy as shown below.

```
<HTML>
<HEAD>
  <TITLE>title</TITLE>
</HEAD>
<BODY BGCOLOR="#FFFFFF">
  .....
</BODY>
</HTML>
```

Indent (Tab) Width

This specifies the width of an indent (tab). Be sure to specify this value in points and not by the number of space characters.

- ☒ "Arrange <tag> labels within one line"
This check box specifies whether a tag (i.e. areas enclosed by '<' and '>') written over several lines, should be displayed instead within just one line.
- ☒ "Delete Tabs/Line Feeds From Data"
This will invoke JChecker to substitute single spaces for tabs, line feeds and space runs, compositing this data into one line. However, this action does not process data included in elements such as <STYLE>, <LISTING>, <XMP>, <PLAINTEXT>.
- ☒ "Enclose Attribute values With "" marks"

JChecker will enclose all unenclosed attribute values with "" marks.

Element Names Menu

You can specify whether element names are converted to uppercase or lowercase letters when beautifying.

Attribute Names Menu

You can specify whether attribute names are converted to uppercase or lowercase letters when beautifying.

Doesn't Indent Nor Starts a New Line

Lists-up elements which do not indent nor start new lines. These elements will be those that are inline elements, ie the character modifiers. Factory settings register these elements as follows:

A, ABBR, ACRONYM, APPLET, B, BASEFONT, BDO, BIG, BUTTON, CITE, CODE, DFN, EM, FONT, I, IFRAME, IMG, INPUT, KBD, LABEL, MAP, OBJECT, Q, S, SAMP, SCRIPT, SELECT, SMALL, SPAN, STRIKE, STRONG, SUB, SUP, T, EXTAREA, TT, U, VAR

Indents Before Start Tags but Doesn't Indent Before End Tags

Lists-up elements which indent before start tags but do not indent before end tags. Factory settings register these elements as follows:

CAPTION, DD, LI, P, TD, TH, TITLE

Doesn't Indent But Starts a New Line After Start Tags

Lists-up elements which do not indent but do start new lines after start tags. Factory settings register these elements as follows:

BR, WBR

Indents Before Start Tags And Starts A New Line After Start Tags

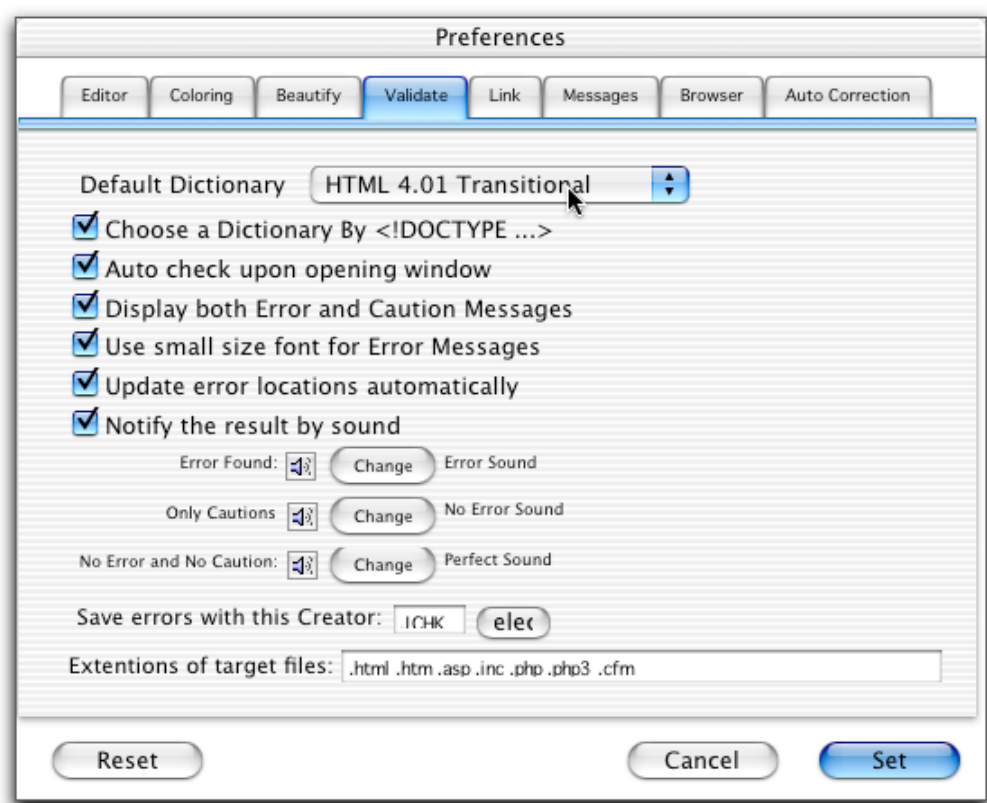
Lists-up elements which indent before start tags and which start new lines after start tags. Factory settings register these elements as follows:

HR

Elements not contained in the above 4 lists will automatically start new lines and indentations, each before start tags and end tags.

3.4 Validate

By selecting "Preferences..." under the "File" menu found in Classic OS, or from the "JChecker3" menu in OS X, the following dialog appears. Click the "Validate" tab if it has not already been chosen.

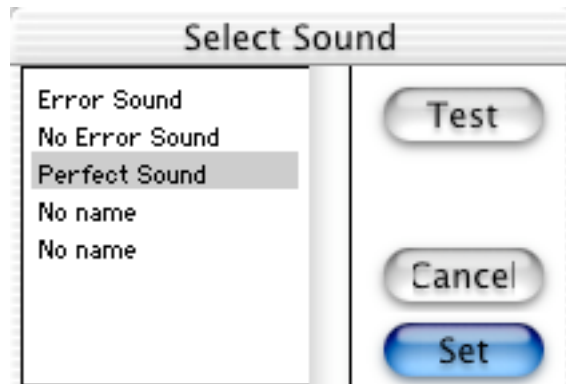


Default Dictionary Pop Up Menu

Here you may specify a default dictionary for HTML validation. JChecker usually consorts with a dictionary which corresponds to the DTD name in the the doctype declaration. If however, the doctype declaration is not found in the HTML, the selected default dictionary will be used for validation checking.

- ☒ "Choose a Dictionary BY <!DOCTYPE...>"
This will invoke JChecker to use a dictionary which corresponds to the DTD name in the the doctype declaration. If not checked, the default dictionary will be resorted to for checking.
- ☒ "Auto check Upon Opening Window"
An HTML validation check will automatically be conducted when a file is opened.
- ☒ "Use Small Size for Error Messages"
The error messages will be displayed in 9 pt. small size font. Otherwise, messages will default and be displayed in 12 pt.
- ☒ "Display both Error and Warning Messages"
Both warning and error messages will be displayed when an HTML validation check is conducted.
- ☒ "Update Error Locations Automatically"
When you make changes in the editor window, line numbers and the location of errors may change. In checking this box, the location of errors will be updated automatically upon changes made in the editor window.
- ☒ "Notify the Result By Sound"
The completion of an HTML validation check are indicated by preselected sounds. A quacking sound indicates the presence of errors, a chime indicates no error and a fanfare indicates there were no errors and no warnings.

You can test each sound by pressing the  button. You can also customize sounds by pressing the "Change" button. The sound selection dialog will appear as follows:



Save Errors With This Creator

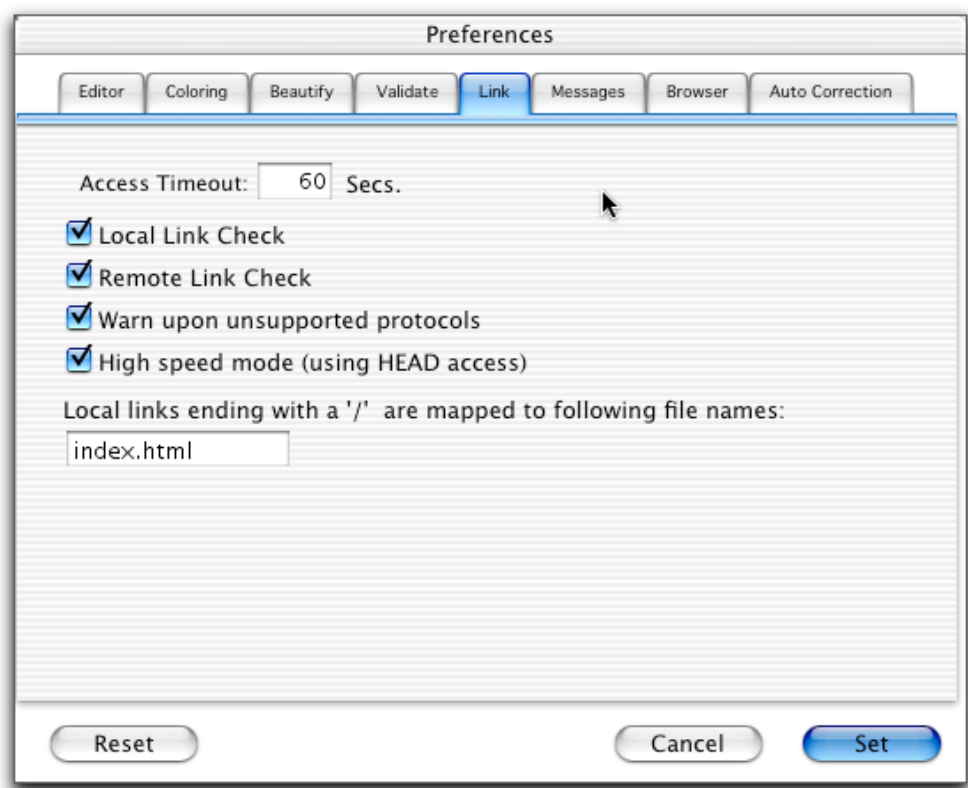
When you select the menu "File">"Save Messages...", the error and warning messages are saved with the file creator shown in this field. The default is 'JCHK' of JChecker. You can change the creator by typing it directly or by using the "Select" button. If you press the "Select" button, a file selection dialog will appear. After you select a file in the dialog, its file creator will be set in the field.

Extensions of target files

Specifies the file extensions of the html files. You can specify multiple extensions delimited with space codes. The auto-coloring, the opening validation check and the recursive link check will be conducted for these files of the extensions.

3.5 Link

By selecting "Preferences..." under the "File" menu found in Classic OS, or from the "JChecker3" menu in OS X, the following dialog appears. Click the "Link" tab if it has not already been chosen.



Access Time-out

You can specify the access time-out for checking remote links. If a check takes too much time (for instance, because the server's IP address cannot be found, or because the server does not respond), then it will be aborted after this selected time interval and the link will then be considered invalid with an error message reported.

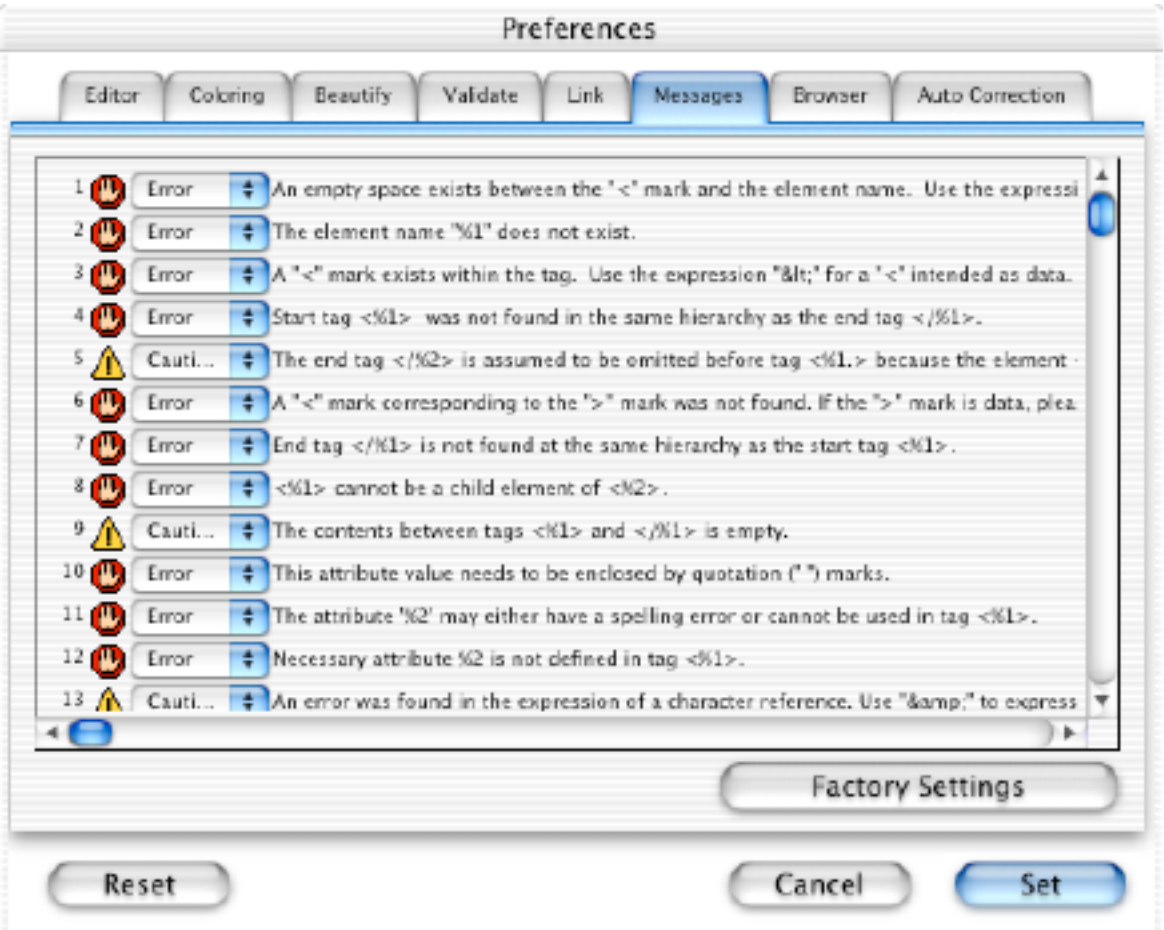
- ☒ "Local Link Check"
By selecting this check box, local links , which point to files or folders on your Macintosh, will be examined. This action does not require Internet access capabilities. Since checking local links does not take much time, you may not need to turn off this check box.
- ☒ "Remote Link Check"
By selecting this check box, remote links, which point to files or folders on the Internet, will be examined. If your Macintosh doesn't have Internet access capabilities or you don't want to go online, leave this box unchecked.
- ☒ "Warn Upon Unsupported Protocols"
JChecker can check "http://", "https://", and "file:///" link protocols. However, JChecker will not support protocols such as "ftp://" and "mailto:". If activate this check box, JChecker will warn you when such unsupported protocols are found. Otherwise these links will be ignored during link checking.
- ☒ "High Speed Mode (using HEAD access)"
By activating this check box, JChecker checks the remote links based on HEAD access, loading only the head information. This is much faster than the GET access, which loads all contents of HTML.

Local links ending with a '/' will be mapped to following file name

When a local link ended with a '/' and its file name was omitted, this default file name will be appended and then the link check will be conducted. Usually the default file name may be "index.html" or "default.html" depending on the web sever setting.

3.6 Messages

By selecting "Preferences..." under the "File" menu found in Classic OS, or from the "JChecker3" menu in OS X, the following dialog appears. Click the "Messages" tab if it has not already been chosen.



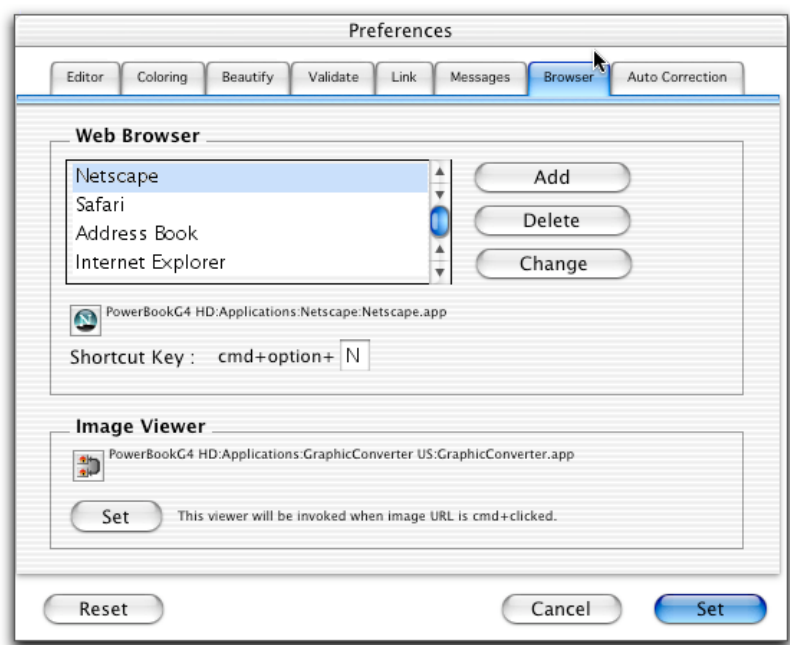
JChecker can display about 70 kinds of error or warning messages as the result of a check, You can determine both the type and whether or not to display each or any of these messages.

By pressing "Factory Settings", all message types are reverted to the factory setting. Under factory setting, an item which breaks HTML rules constitutes an "error" and an item which doesn't break a rule but is not considered preferable, constitutes a "Caution".

Note: The strings "%1" or "%2", appearing in some messages will be replaced with appropriate words upon the message actually being displayed.

3.7 Browser

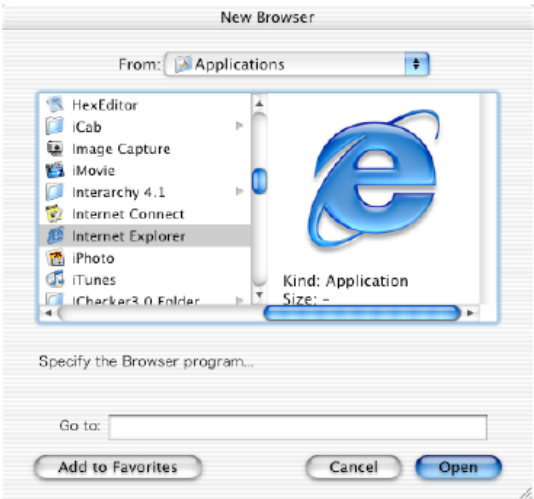
By selecting "Preferences..." under the "File" menu found in Classic OS, or from the "JChecker3" menu in OS X, the following dialog appears. Click the "Browser" tab if it has not already been chosen.



Setting the Web Browser

You can specify the desired web browser for HTML preview, as well as a graphic viewer for the viewing of image files.

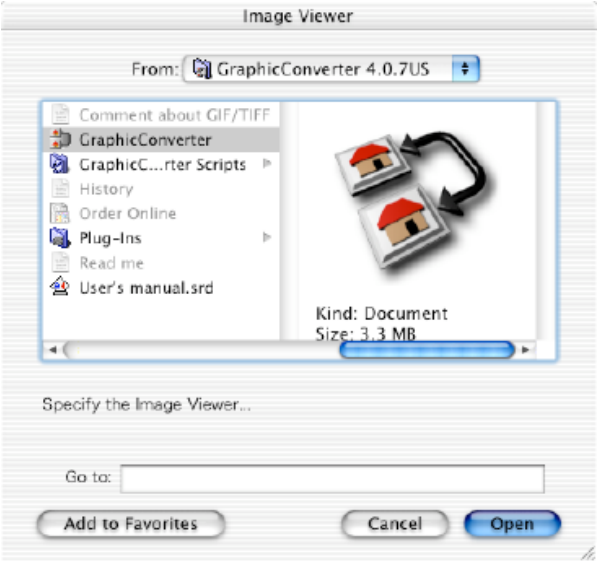
If you desire to add a new browser to be accessible under the menu "Browser", click the button "New". The following file dialog will appear where in which you may specify the desired browser.



Key short cuts may also be specified for each browser's preview menu. If you wish to alter the browser menu, select the desired browser from the list and click "Change". To remove a browser's menu, select the desired browser from the list and click on "Delete".

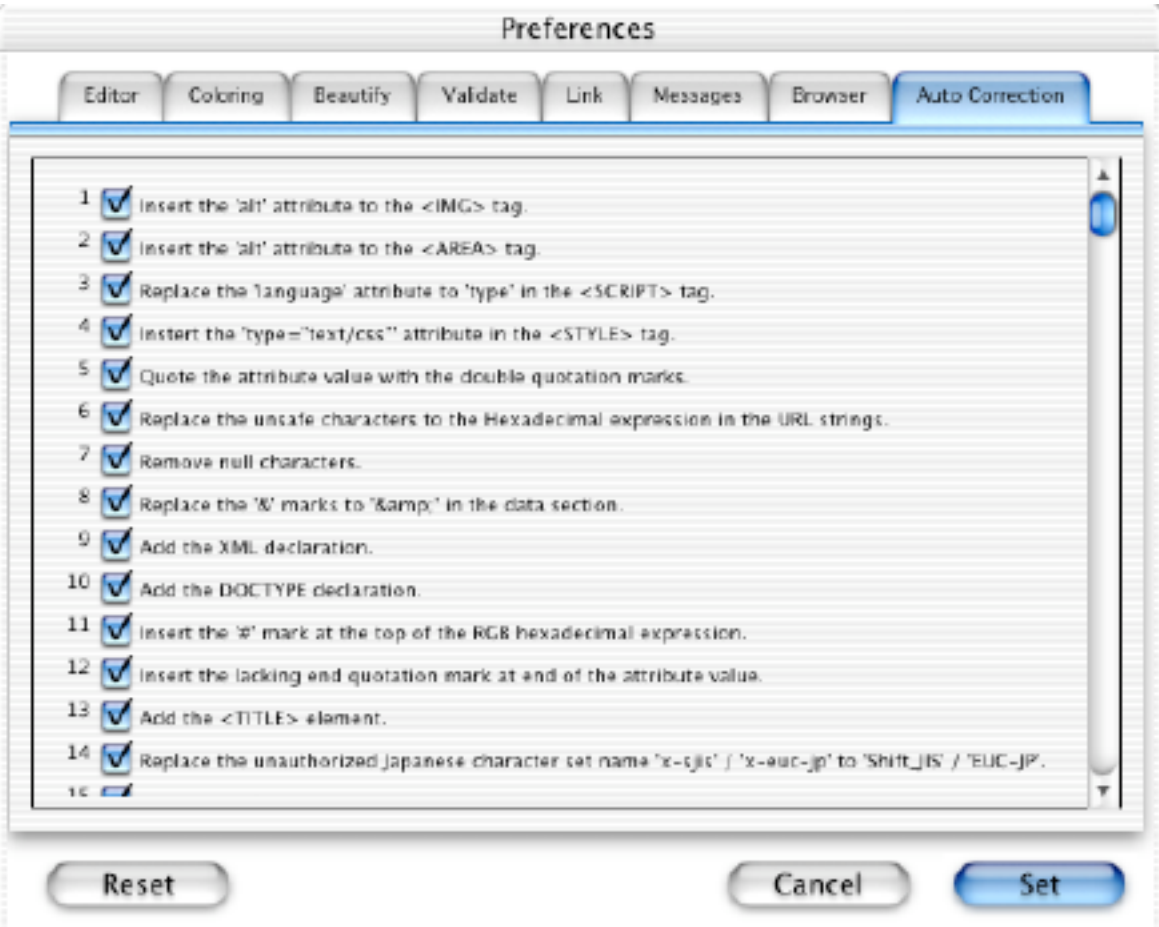
Setting the Graphic Viewer

When you wish to specify a graphic viewer, simply click "Set". The following file dialog will appear which allows you to specify the viewer you want to use.



3.8 Auto Correction

By selecting "Preferences..." under the "File" menu found in Classic OS, or from the "JChecker3" menu in OS X, the following dialog appears. Click the "Auto Correction" tab if it has not already been chosen.

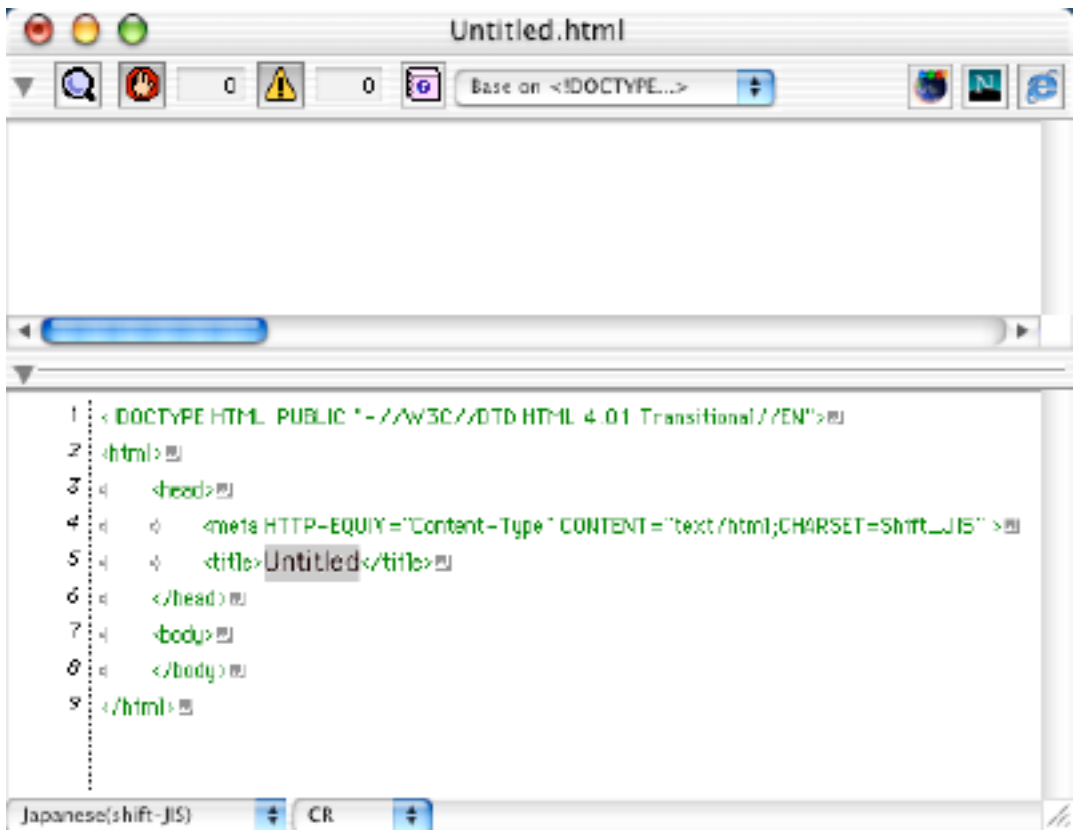


JChecker is able to correct 20 kinds of simple HTML errors automatically. You can specify the ON/OFF of each correction item in this panel.

Chapter 4 File Operation

4.1 Creating New HTML

To create new HTML, select "New HTML" from the "File" menu. A new window containing minimum tags will open on your screen.



The contents of new HTML can be predetermined by accessing the "New HTML Template" popup menu of the Preferences "Editor".

4.2 Opening Local HTML

There are three ways to open local HTML files.

(1) Double Clicking JChecker Documents.

The icon below represents a JChecker document. If you double click on such icons, JChecker automatically starts up and opens the file.



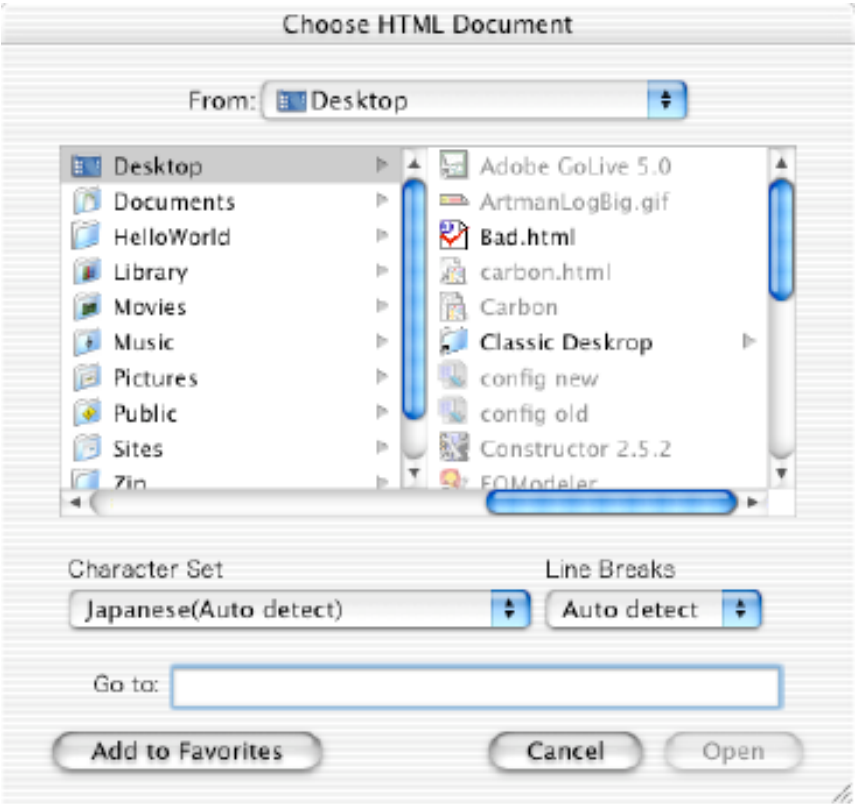
In the event that an older JChecker2.1 program remains on your HD, JChecker2.1 may be mistakenly invoked when a JChecker document is double clicked. Therefore, it is recommended to remove the dated JChecker2.1 program from your HD before beginning to use JChecker3.0.

(2) Drag & Dropping Files onto the JChecker Icon.

Any text file, including non-JChecker documents, can be opened using drag & drop.

(3) Opening an HTML File from Within the File Dialog.

To open a file from within JChecker, select "Open HTML..." under the "File" menu.

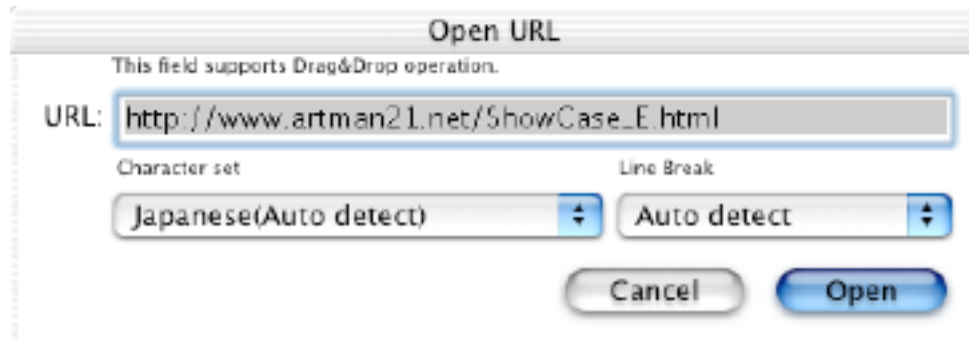


You can specify the character set and line break protocol from the pop up menu with "Character Set" and "Line Breaks" options located at the bottom of the dialog.

Setting the pop up menu "Line Breaks" to "Auto detect", JChecker detects the style of line breaking automatically. When opening Japanese HTML, open the pop up menu's "Character Set" selecting "Japanese (Auto detect)". JChecker will then choose the proper kanji character set from the following: "shift-JIS", "JIS" , "EUC", or "UTF-8".

4.3 Opening Remote HTML

To open an HTML file located on the Internet, select "Open URL..." under the "File" menu. The following dialog, asking for a desired URL address, will appear.



URL's can be dragged & dropped into the input field or entered from the keyboard. If you want to check a web page currently being displayed on your browser, simply drag the URL string from your browser onto this dialog.

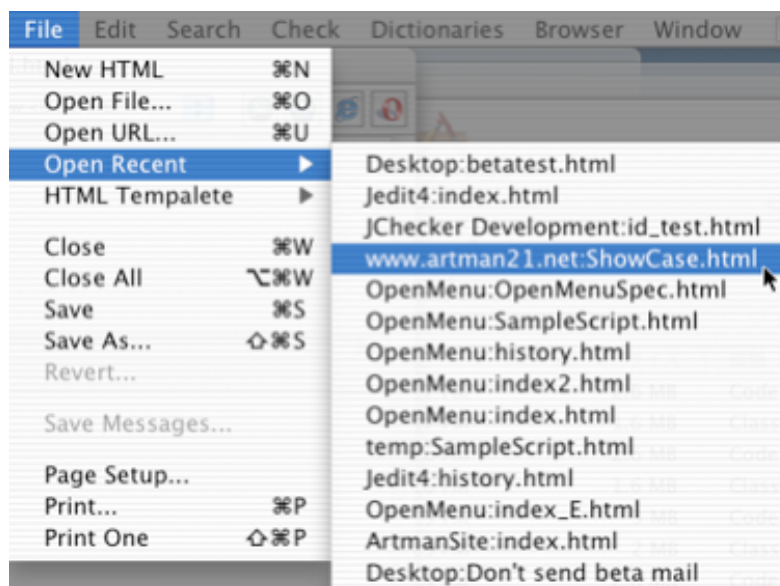
Note: as an "Open URL" is using http: protocol, you are not allowed to write back HTML to the remote file.

You can however load and save these remote files by using "external editor protocol" to be described in the next chapter.

4.4 Accessing Recently Opened Files

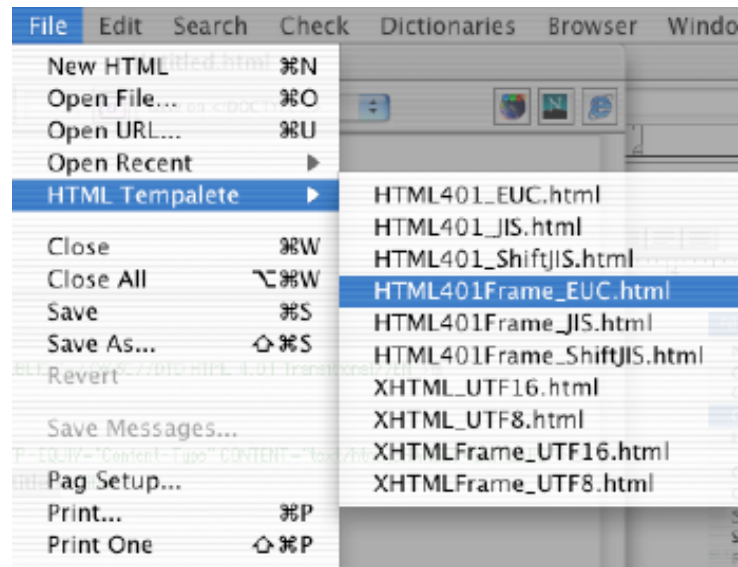
Selecting the menu "Open Recent" under the "file" menu, allows a list of recently opened files to appear, with the most recent file listed at the top. Select the file you want to open. This is a quick, easy way to access and reopen recent files. The maximum number of listings is 15.

To remove a file name from this list, simply select the name of the file while depressing the option key.



4.5 HTML Templates

If you wish to store HTML files within the "Templates" folder of the "JChecker3.0" folder, such HTML files will appear as "HTML Templates" accessible under the "file" menu. This menu is useful for opening either stock or skeleton HTML files.

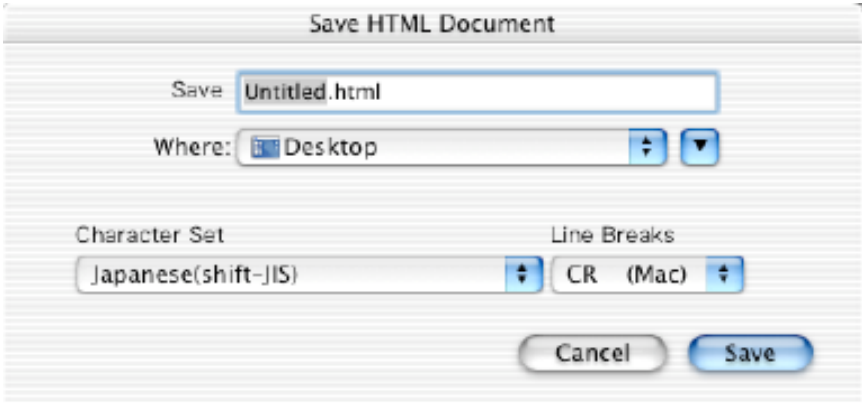


4.6 Closing Documents

- Menu **"File">"Close"**
This command closes the active HTML editing window. If the HTML text has been modified, you will be prompted to whether the text should be saved or not.
- Menu **"File">"Close All"**
This command closes all HTML documents.

4.7 Saving Documents

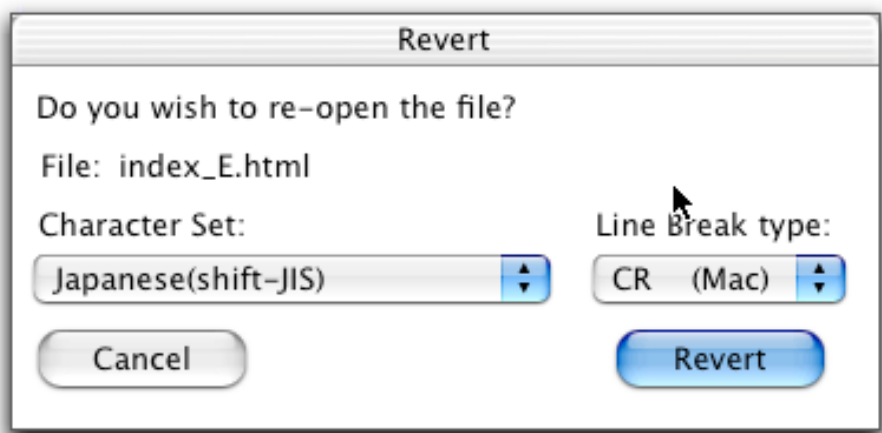
- Menu **"File">"Save"**
This command will save the contents of the active HTML editing window under the current file name. If the HTML text is new, a Save File dialog, requesting where to save the file, and for a file name to be entered, will be displayed.
- Menu **"File">"Save As ..."**
This command will save the contents of an HTML editing window prompting for a preferred name, character set and line break type. Also requested is a location of where to save the file, as in the following dialog:



4.8 Reverting Documents

Menu **"File">"Revert..."**

This command will revert the contents of the HTML editing window back to the state when it was last saved. When you select the menu, the following dialog will appear. You can re-assign the character set and line break type.



4.9 Saving Error Messages

Menu **"File">"Save Messages..."**

This command will save all the contents of the errors and warnings from the message window as a text file. By default, the file is saved as JChecker document. Alternately, you can specify the file's creator from the Menu **"Preferences...">"Validate"**.

4.10 Printing HTML Documents

Menu **"File">"Page Setup..."**

This command opens the page setup dialog. You can specify paper size and other page settings.

Menu **"File">"Print..."**

This command opens the printing dialog allowing for print settings control and document printing.

Menu **"File">"Print one"**

This command simply prints one copy of a document without displaying the print dialog.

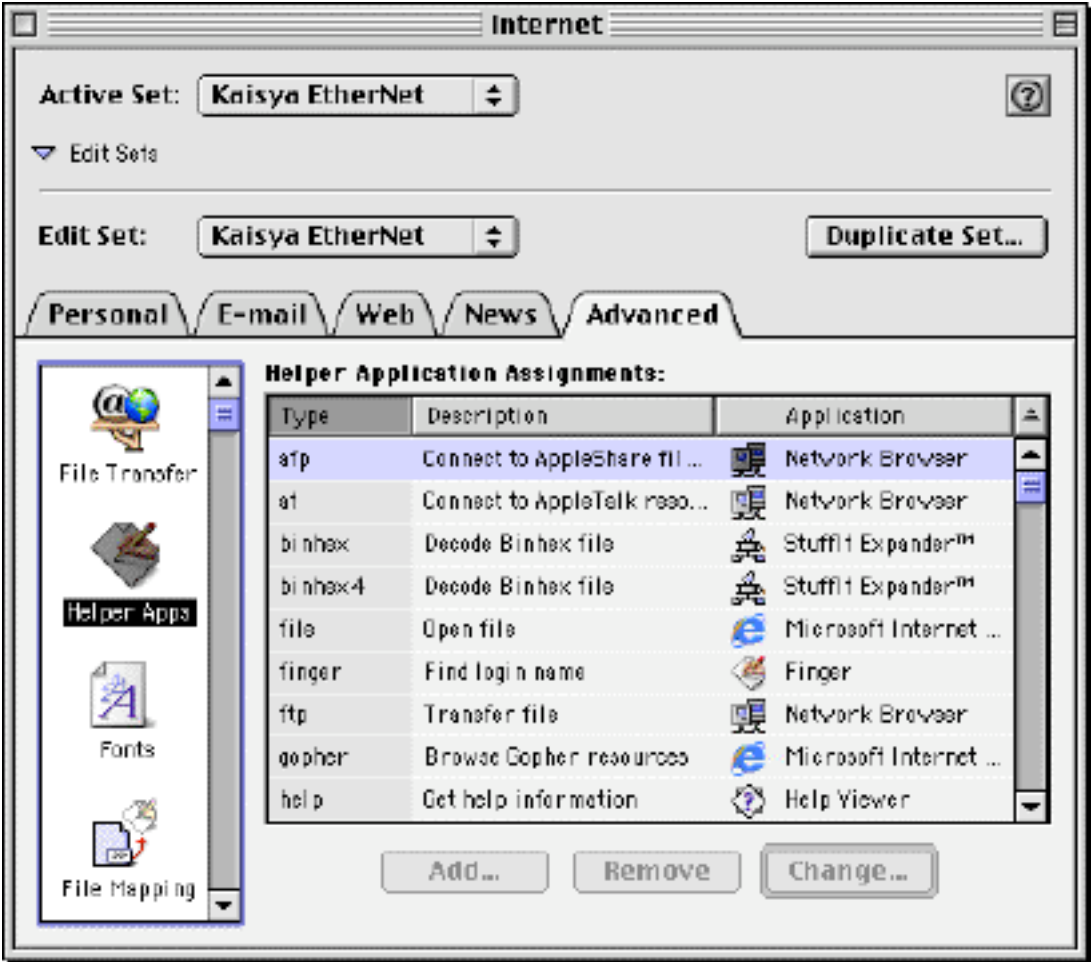
Chapter 5 Working with FTP Clients

As JChecker supports **External Editor Protocol**. Ftp-clients, such as Interarchy, NetFinder, Transmit, and Fetch can incorporate an "Edit with JChecker" menu command. By selecting a file in a remote directory and choosing this command, the file is downloaded and opened in the JChecker window. When you save the document in JChecker, the file is then automatically uploaded back to the remote machine by the ftp client.

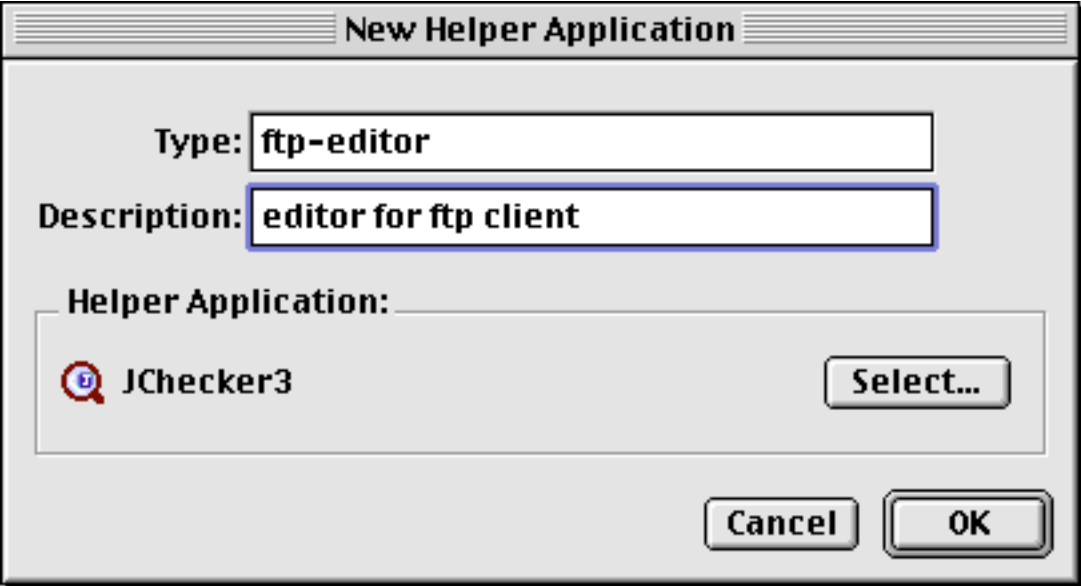
5.1 Setting

In the ftp client application which supports the External Editor Protocol such as Interarchy4.1, you will find the **"Edit with BBEdit"** menu command under the default settings. You can alter this menu to incorporate **"Edit with JChecker"** as follows.

1) Choose **Apple Menu > Control Panels > Internet**. And select the "Advanced" tab from the Internet panel. If you can't find the "Advanced" tab, select within the menu "Edit>User Mode" and set the mode to "Administration". Next, click the icon "Helper Apps" in the list located on the left side. You will see a list of helper applications as follows.



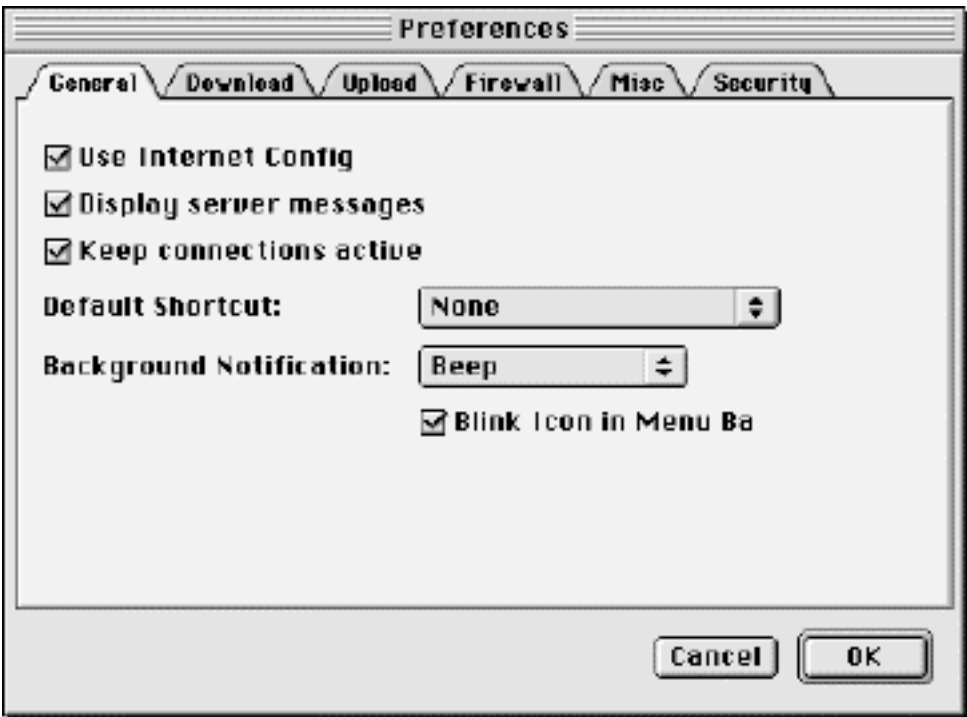
2) Find the "ftp-editor" within this list. Once located, select the **"Change"** button. Another method is to select the **"Add"** button. Add or change the items as shown below and once entered, press "OK".



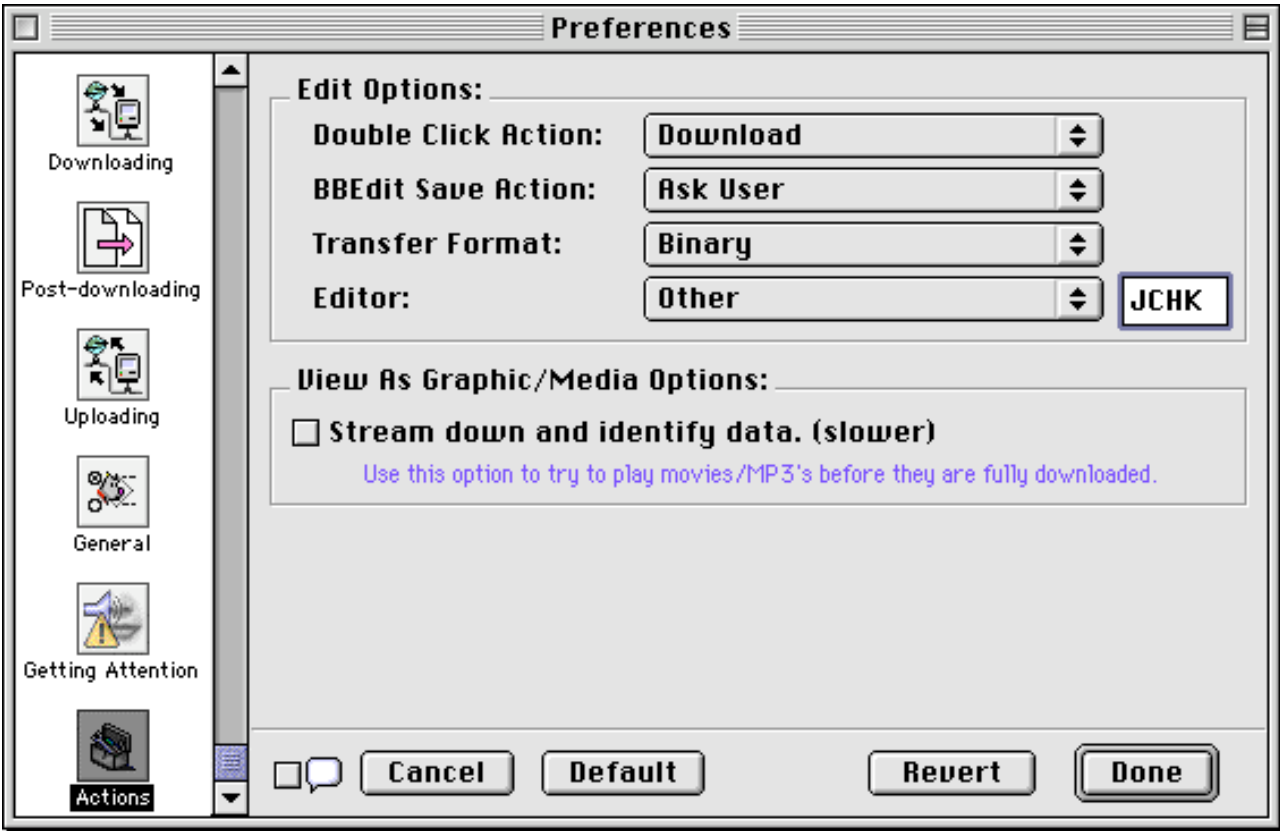
3) Launch your ftp client application (ex. The application 'Interarchy 4.1') and go to the 'Listing' menu. You will now find "Edit with JChecker" listed in the menu as follows.

Listing	
Download	
View as Text	⌘L
Edit with JChecker	⌘J
Upload/	⌘U
Rename/Move/	⇧⌘R
Create Directory/	⌘N
Delete	⌘⌫
Open Parent	⌘↑
Export Listing/	
Set Permissions/	⇧⌘P
Set Upload Permissions/	⇧⌘U
Send Raw FTP Command/	

If your FTP Client is Fetch4.0, the same can be achieved by selecting the menu **Customize > Preferences > General**. Activate the check box **Use Internet Config**.



For "NetFinder", select from the menu **edit > Preferences** and select the **Action** icon within the preferences list. Set the editor pop up menu to **Other** and Key-in "JCHK" in the creator field.



5.2 How To Use

Once JChecker is listed by your ftp client, launch the client and open the remote directory that contains the file you want to edit. Select the file and Choose from the menu "**Listing > Edit with JChecker**". The file will be downloaded and opened in the JChecker window.

When you go to "Save" the document, the document will automatically upload back to the remote machine utilizing the ftp client.

When closing the JChecker window, your file will automatically register under the menu "Recent Files" in the same manner as other files. If you wish to select the remote again by invoking it from the Recent Files menu, the ftp client will also automatically invoke, with the file being downloaded and opened in the JChecker window.

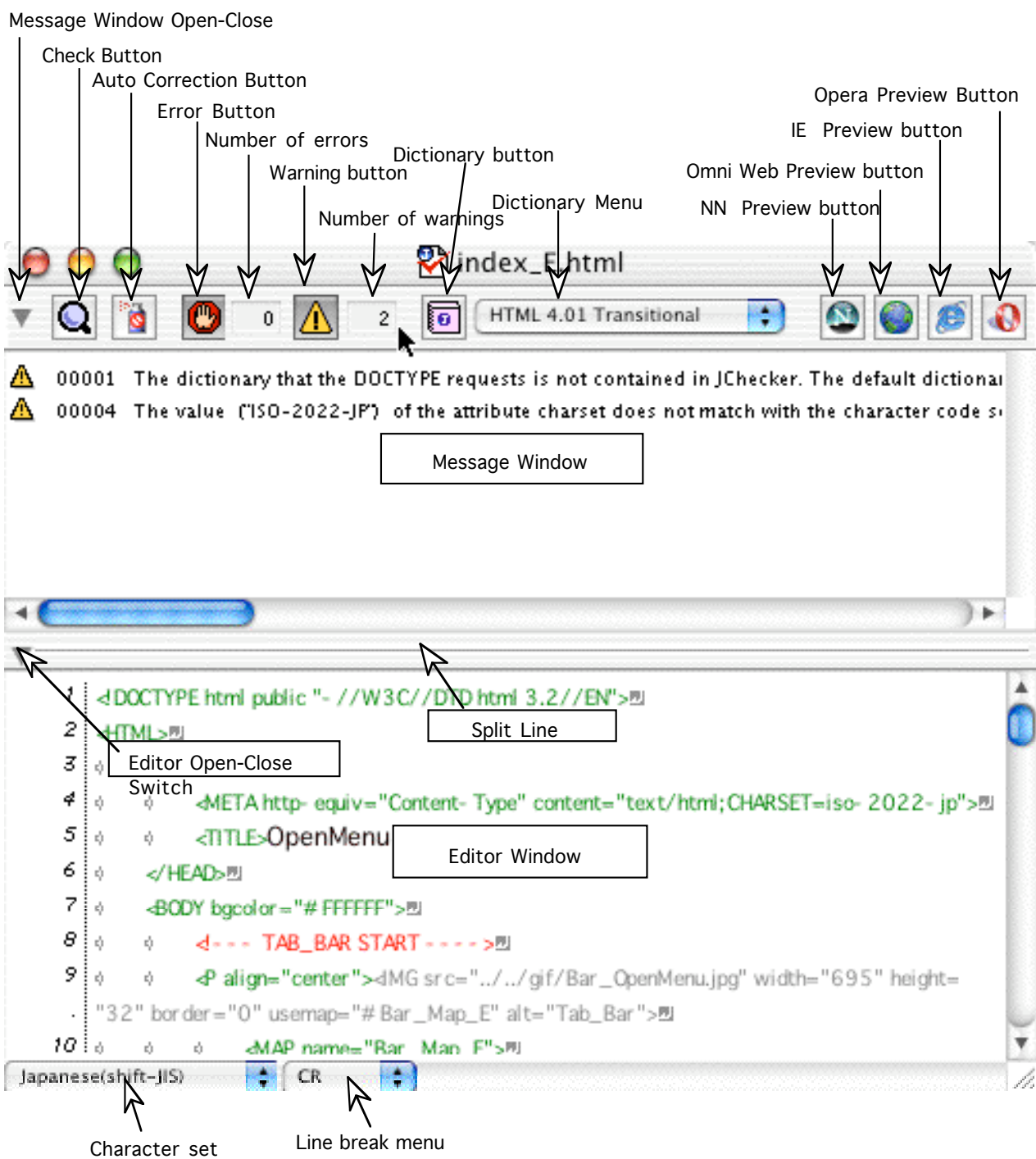
Note

The External Editor Protocol is only valid under the Classic OS environment. You cannot yet use this function with JChecker under Mac OS X.

Chapter 6 Editing

6.1 Operation of the Editing Window.

The editing window is divided into two parts: the message window in the upper section and the editor window in the lower section. The relative space occupied by the message and editor windows can be altered by dragging the "Split Line" located in the center. Also, by clicking the triangular "Editor Open-Close Switch" (located in the middle of the left side), an editor window can be opened and closed.

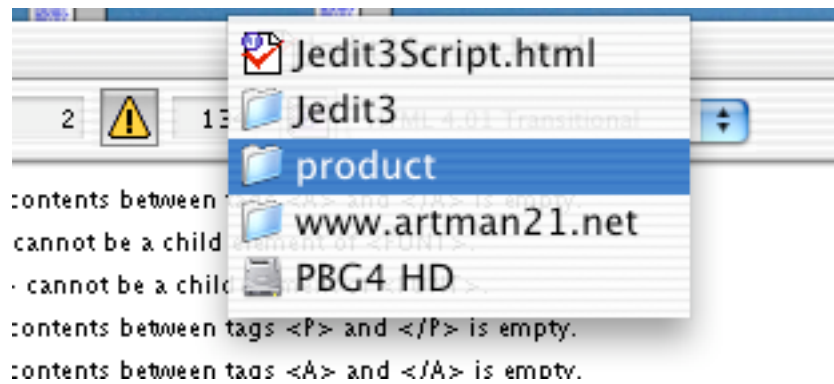


Proxy Icon

The small icon found at the left side of the window tile is called a "Proxy Icon." It represents a document's file. You can move or copy a file by dragging and dropping this icon onto the finder window like with other finder icons. If a document however, is modified and not yet saved, the proxy icon will appear dimmed. You will be unable to drag a dimmed proxy icon onto the finder.

Directory Pop-up Menu

By clicking a document's title while depressing the command key, a pop-up menu will appear as shown below. This example shows that the "JeditScript.html" file is located in the "Jedit3" folder. By selecting a folder or drive menu item in this way, the item will open on the Finder desktop.



You can open-close message areas by clicking the small triangle switch located at the upper left corner of the window.

You can open-close editing areas by clicking the small triangle switch located at the left of the split line.



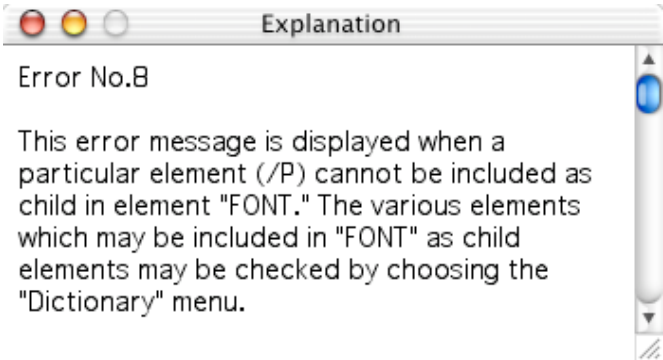
Check Button

Clicking this button starts a validation check of the HTML document being displayed in the editor window.

When clicked while depressing the shift key, a link check may be conducted. While depressing the option key, an auto-correction can be conducted after the check.

When errors or warnings are found, they are displayed in the message window. Clicking on the error or warning message highlights the appropriate section of the HTML text in the editor window.

Upon double-clicking an error message, an error explanation window will appear as follows.



You can also execute checks by selecting either the menu "Check > Validate HTML " or "Check > Check Links".



Auto Correction Button

Clicking this button starts a validation check of the HTML document and then executes an auto correction of the HTML.



Error Button

This is a toggle button. If it is clicked, error messages resulting from a check are displayed in the message window. Deactivating this button prevents all error messages from being displayed.



Warning Button

This is a toggle button. If clicked, warning messages resulting from the check are displayed in the message window. Deactivation prevents all warning messages from being displayed.



Browser Preview Button

Browser Preview Buttons can be located at the upper right corner of the window. The display of the buttons can be controlled under Preferences "Browser". By clicking one of these buttons, the corresponding web browser will be invoked and you will be able to preview the HTML contents. You can register as many kinds of browsers as you want under the Preferences "Browser", however only the first 4 Browser Preview Buttons will be displayed.

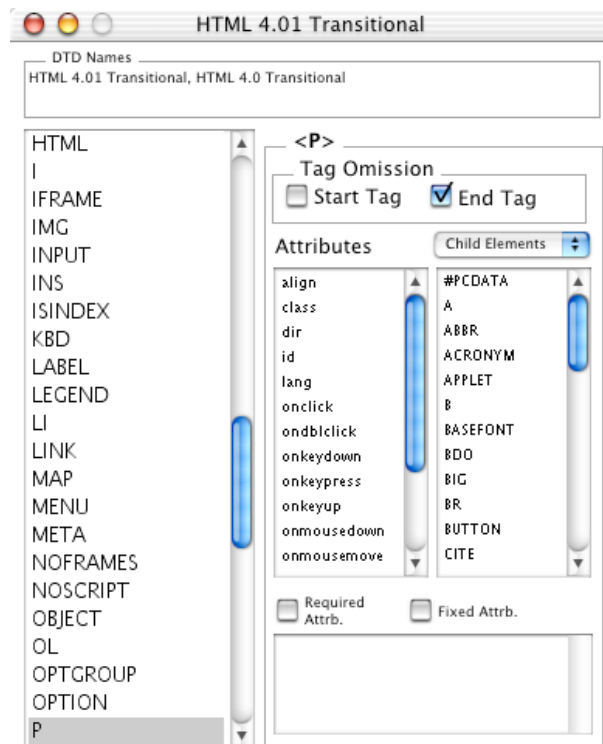
Dictionary Menu

You can select a dictionary for validating HTML. If you select the item "follow <!DOCTYPE...>", the dictionary specified at the doctype declaration will be automatically selected.



Dictionary Button

Clicking this button, allows you to consult tag information located at the current cursor position of the editing window. A dictionary window will be displayed as shown:



Character Set Menu

This menu shows the character set used by an HTML file. Changes to the character set using this menu will result in the HTML being saved with the new character code set.

Line Break Type Menu

This menu shows the type of line break to be used when a file is opened. Changing the line break type using this menu, results in the HTML being saved with the new line break type.

Split Line

Vertically dragging this line by mouse will change the relative heights of the message and editor windows.

6.2 Keyboard Shortcuts

Moving the Cursor

You can use the arrow keys to move the cursor in the following ways:

Left arrow moves the cursor one character to the left.

Right arrow moves the cursor one character to the right.

Up arrow moves the cursor one line up.

Down arrow moves the cursor one line down.

Option+left arrow moves the cursor one word to the left.

Option+right arrow moves the cursor one word to the right.

Command+left arrow moves the cursor to the beginning of the paragraph.

Command+right arrow moves the cursor to the end of the paragraph.

Option+up arrow or page up moves the cursor to the top of the current or previous page.

Option+down arrow or page down moves the cursor to the bottom of the current or next page.

Command+up arrow or Home moves the cursor to the beginning of the document.

Command+down arrow or End moves the cursor to the end of the document.

Other Key Commands

- To open the contextual menu, click the highlighted area with the control key down.
- By clicking a local URL string with the command key down, the corresponding html document will open in a new window. When the URL is an image file (JPEG, GIF, PPG), a graphic viewer will be invoked alternatively.
- Function Keys Action: F1=Undo, F2=Cut, F3=Copy, F4=Paste
- Shift+return insert the tag
 and a line break code. The tag
 will be inserted when editing the xhtml documents.
- Shift+delete deletes a character on the right side of the caret.
- To search backwards, select the "Find Next" menu while pressing the shift key, or type "g" while holding both the command and shift key down.
- To search backward for a selected string, choose "Find Selection" with the shift key down, or type an "h" while depressing both command and shift keys.
- Option+delete deletes characters from the blinking point of the cursor insertion to the end of the paragraph.
- To remove a file name from the "Recent Files" under the "File" menu, select the target file item while depressing the option key.
- To change the linkage of a file in the bookmark list, double click the file's icon in the list with the option key held down.

6.3 Undo and Redo

Select "Undo" from the "Edit" menu to undo a previous action.
Select "Redo" from the "Edit" menu to cancel a previous undo.

Multiple actions can be undone, returning counter-stepwise to the state when the document was initially opened. Similarly, you can redo multiple actions, returning a document to the state where you had begun to undo.

6.4 Copy, Cut and Paste

You can copy, cut and paste text data just as you can in any other Macintosh application.

Shortcuts for the Highlighted Areas

To copy or cut a certain text area, first drag your mouse to highlight it. Otherwise:

- To highlight one word, double click on the word. To highlight an entire paragraph, triple click on that line.
- To highlight between two separated text areas, highlight or click in one area first, and then click another area while pressing the shift key.
- Using shift and cursor control keys (such as shift and arrow keys), you can also highlight areas without the use of mouse action.
- To easily highlight lines, click and drag over the designated line numbers area, if these numbers are actively displayed on the left side of the document window.

6.5 Drag & Drop

By using drag and drop, you can transfer data between document windows or within the application much more easily than using copy and paste.

You can drag text data and drop it onto another document as follows:

Highlight a desired text area in the document window. As the mouse pointer goes over the highlighted area, it will turn into a  shape. By clicking and hold the mouse button, the mouse pointer will then turn into a  shape. While continuing to depress the mouse button, move the pointer onto a different document window. Upon releasing the mouse button, the highlighted text area will be copied into the new document window.

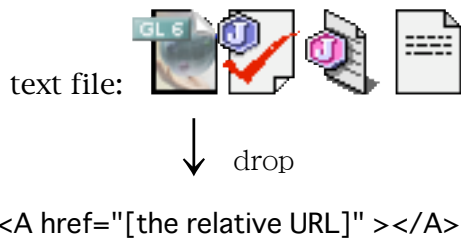
Note: when you drag and drop text from and to the same document window, the text will be "moved," not "copied." If you want to copy the highlighted text to be used in the same document, drag and drop while depressing the option key.

When dropping a selected area onto the Finder desktop, the following example of a clipping file will be created:

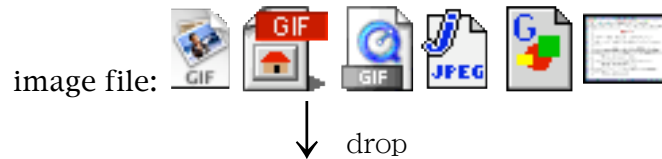


Such clippings may be dropped onto the editing window allowing the contents to be pasted at the point of insertion.

When dropping a text file onto the editing window, an anchor tag will be automatically generated and its relative URL will be inserted.



When dropping desired image files (JPEG, GIF, PPG) onto the editing window, an IMG tag will automatically be generated and the relative URL will be inserted. Conveniently, the height and width of the image will automatically be inserted.



To insert only the relative path of a file, drop the file onto the editing window while depressing the shift key.

6.6 Selecting Tags, Pair Tags and Elements

Set the cursor inside a tag and choose the menu item "Edit > Select Tag". The tag between "<" and ">" will be highlighted.

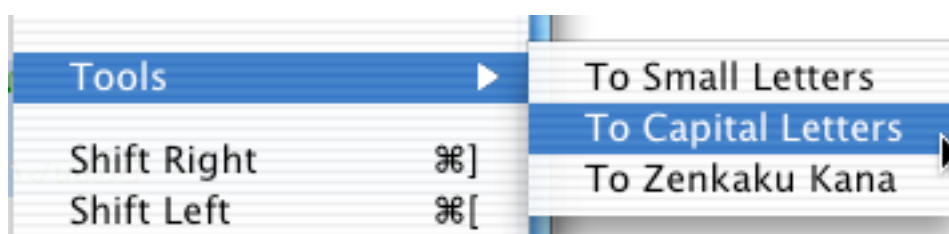
Set the cursor inside a tag and choose the menu item "Edit > Select Pair Tag". If the tag is a start tag, its end tag will be highlighted. If the tag is an end tag, its start tag will be highlighted.

Set the cursor inside a tag and choose the menu item "Edit > Select Element". The contents between the start and end tag, i.e. the area from the start of the tag to the end of the tag (the element) will be highlighted. Even when the end tag is omitted, the area will still be highlighted correctly.

6.7 Removing Tags

Highlight an area that you wish to remove tags and select "Edit > Remove Tags". All tags in the desired area will be deleted leaving only data. This procedure may be useful when converting HTML to ordinary text.

6.7 Tools



The menu "To Small Letters" will convert the capital letters to small letters in the highlighted area.

The menu "To Capital Letters" will convert the small letters to capital letters in the highlighted area.

The menu "To Zenkaku Kana" will convert Japanese hankaku kana letters to zenkaku kana letters in the highlighted area.

6.8 Paste Date and Time

To paste a current date or time string, select "Paste Date" or "Paste Time" under the "Edit" menu.

To paste a longer string, one which includes the day of the week, or a time that includes the seconds, select "Date" or "Time" with the addition of holding shift key down.

See the following examples:

short Date: 99.3.17
short Time: 9:11 PM

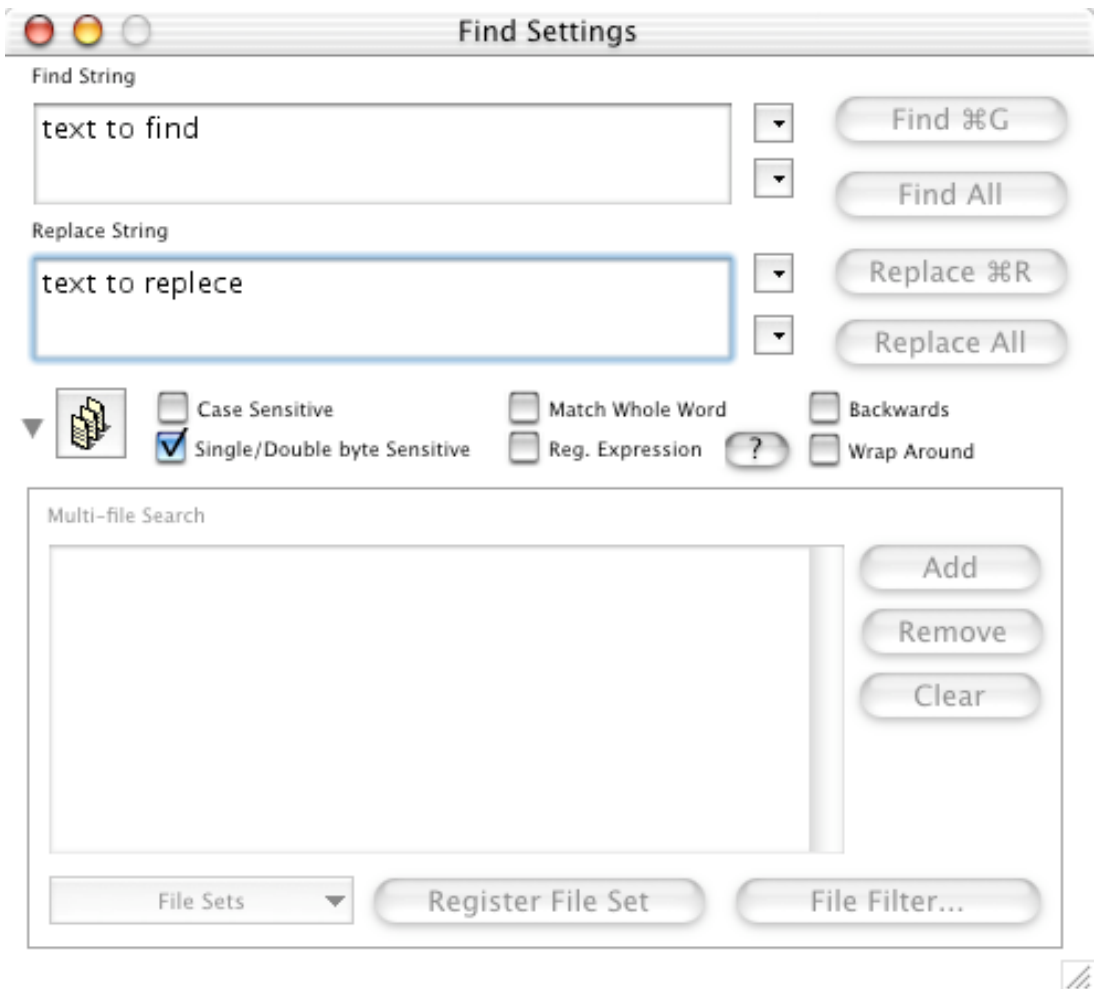
long Date with shift key : Wednesday, June 9, 1999
long Time with shift key : 9:11:41 PM

Chapter 7 Search

JChecker3.0 search functions operate as powerfully as those of Jedit4.0. These functions support regular expression, multi-file search, mark list, among others.

7.1 Finding Text

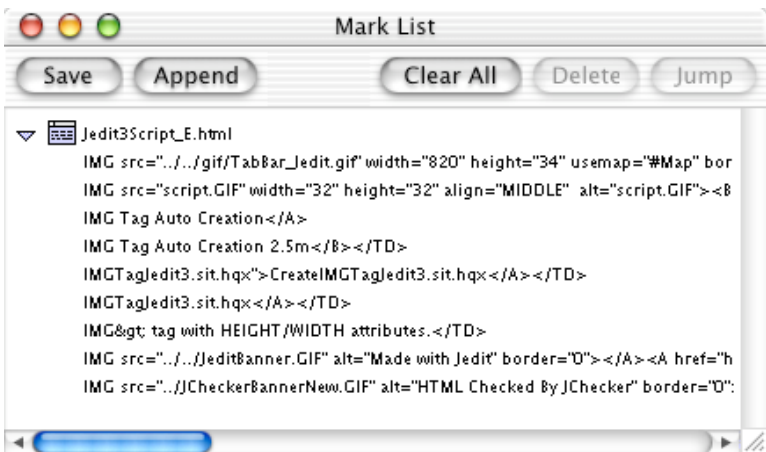
To find a text string, select "Find" listed under the menu "Search." A dialog box named "Find Settings" will appear as shown below.



Type the string that you wish to search within "Find:" and hit "Find." Searching will start from the current cursor position and if found, the string will be highlighted in the document.

To search for a next occurrence of the same string, push the "Find" button again. When there are no more occurrences, a beep sound will inform you that no further possibilities exist and that the search is over. You also can perform this function by selecting "Find Next" under the menu "Edit."

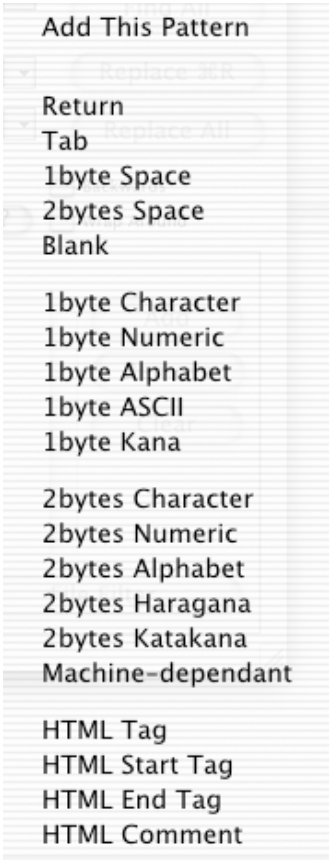
By pushing the "Find All" button, all occurrences of the sought string will be listed in a 'mark list' window as follows:



By double clicking an item in the mark list, you will be taken to the corresponding line in the document window.

By clicking the  button at the upper right side of the find field a pop-up menu, a list containing the last 10 searched for strings will appear. Clicking a selected item will allow it to be copied onto the find field.

Clicking the  button at the lower right side of the find field, a sample pattern pop-up menu will appear as follows:



To add your own search patterns, type the desired pattern into the find field and select the "Add This Pattern" item into the menu. To delete a menu item you have added, select the desired item while depressing the option key.

If you want to search for non-printing characters, enter the following combinations of ("\" and a desired character) into the find field:

`\t=TAB, \r=CR, \n=LF, \s=SPACE, \b=Backspace.`

Other combinations using the "\" key will be explained later in the section "Regular Expressions."

If you want to search elsewhere for specific text within the current document, highlight (select) the text and choose "Enter 'Find' String" under the menu "Search." This text will then be copied into the find field of the find dialog.

By choosing "Find Selection," the highlighted text becomes copied into the find dialog and will be searched for automatically.

By checking the box "Backwards," the search starts from the cursor position and proceeds backwards to the top of the document.

Normally JChecker will stop searching when it reaches the end of a document. If you want to continue, automatically jump-continuing the search from the top of the document, check the box "Wrap Around."

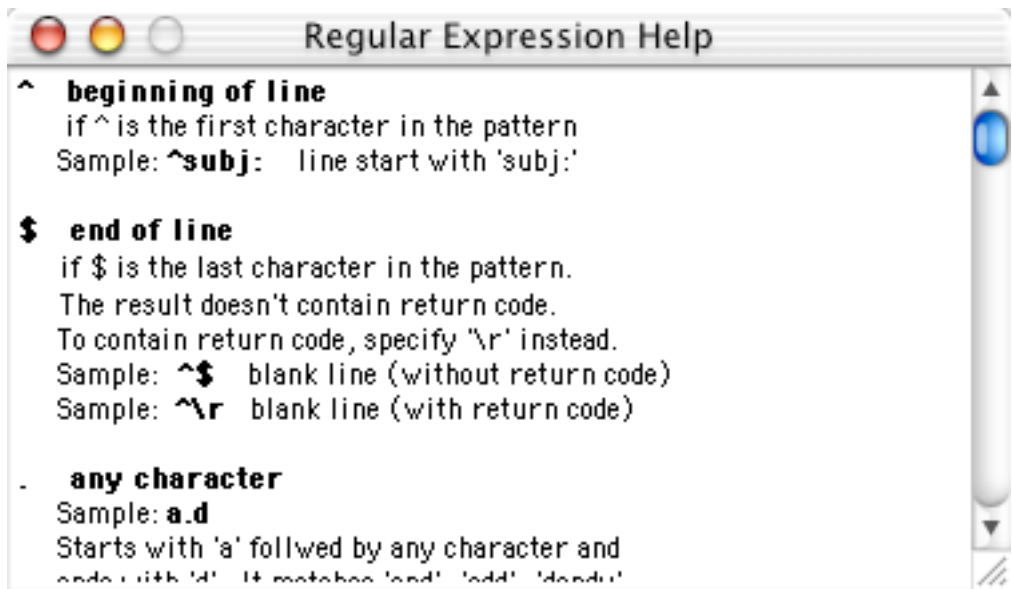
By checking the box "Case Sensitive," JChecker can now discern upper and lowercase characters, narrowing your search. Sans this option, searching will ignore upper and lowercase differences.

If you check the box "Single/Double byte Sensitive," JChecker discerns between single byte and double byte characters when conducting a search. Sans this option, searching will ignore the difference between single and double byte. Note: this option is valid only for Japanese text. Deselect this option when using other languages.

If you check the box "Match Whole Word," JChecker results match a search string, only if such string is isolated in the document text by word-break characters (white-space or punctuation). By not checking this box, JChecker will broadly match the search string if found anywhere within the text.

By checking the box " Reg. Expression" you can use regular expression patterns within the find field. By using such expressions (see 'regular expressions' below), you can specify word 'patterns' without knowing the exact words contained and to be located in the text.

Push the  button calls for the regular expressions function help window to appear:



7.2 Regular Expressions

Using the expressions below, you can specify word patterns without knowing the exact words contained in the text. JChecker can also handle Japanese 2-byte characters within these regular expressions.

- ^** beginning of line (if ^ is the first character in the pattern)
Example: **^subj:**
searches for a line beginning with "subj:"
- \$** end of line (if \$ is the last character in the pattern)
Example: **^\$**
searches for a blank line that only contains a carriage return.
When you use "\$", the searched result will not include the carriage return.
If you use "\r", instead of "\$", the searched result will include the return code.
- .** represents any character
Example: **a.d**
will find words that begin with an "a", followed by any character, then followed by a "d".
Possible matches: "and", "add", "dandy"
- *** designates a character that repeats more than once or which never exists.
Example: **an*d**
will represent a word that starts with an "a", followed by an "n" which repeats 0 or more times, then followed by a "d".
Possible matches: "and", "advance", "add"
- *?** same as "*" but matches the minimum number of times possible.
For example, If you search **"*bbb"** in the text "aaa bbb bbbbbb cccc", the string "aaa bbb bbb bbb" will be matched.
If you search **"*?bbb"**, the string "aaa bbb" will be matched.
- +** means a designated character repeats more than once.
Example: **an+d**
Indicates any word that begins with an "a", followed by an "n" which is repeated more than once, followed by a "d". Matches include "and", "annd" but not "advance" nor "add".
- +?** same as "+" but matching the minimum number of times possible.
- ?** means a designated character exists either once or never.
Example: **an?d**
will represent words beginning with "a", followed by either "n" or nothing, then followed by a "d". Matches include "and", "add" but not "annd".
- ??** same as "?" but matches the minimum number of times possible.
- ** represents an escape mark.
Useful in searching for characters such as '^','\$', '\' which have special meaning in the regular expression.

Example: `\^ \\ \$ \* \+`

You can also use the escape mark to search for none-printable characters such as:

`'\t'` =TAB,
`'\r'` =CR,
`'\n'` =LF,
`'\s'` =SPACE,
`'\b'` =BS,
`'\d'` = [0-9] (digits),
`'\w'` = [A-Za-z_] (alphabeticals and underscores)

To express characters as hexadecimal codes, use 'h' followed by associated hexadecimal codes.

Example: `\h20` 1-byte space
Example: `\h8140` 2-byte kanji space

When you have specified an above 1 byte-expression "\hxx", the byte data contained in 2 byte characters will be ignored. In other words, when you have specified "\h81", the

first byte of the 2 byte Kanji Space "\h8140" will not be hit.

- [] represents character classes.
To express a range of characters, use the [-] expression.
To express the exclusion of characters, use the [^] expression.
This function can also be used for 2-byte characters.

Example: [abc]
a character that matches characters a or b or c.

Example: [A-Z]
a character that matches characters A to Z; any capital alphabet letter.

Example: [^a-z]
any character other than a to z.

Example: [a-xABC]
any character from a to x and/or any character matching A, B, C.

Example: [\t\s\h8140]+\$
tabs, spaces or kanji spacing at the end of a line.

- { } represents a group.
Use { } to divide find strings into desired groups which are to be used in replacement strings.

For example, a person's name will be expressed as follows.

[A-Z][a-z]+\s+[A-Z][a-z]+

Expression {[A-Z][a-z]+} means one capital alphabet [A-Z] is followed by more than one lower case [a-z]+ letter.
First group {[A-Z][a-z]+} is the given name.
Second group {[A-Z][a-z]+} is the family name.
First group and Second group are separated by more than one space \s+ .

You can specify the first group as \1 , and the second group as \2 . These numbers can be used in the Replacement process.

As an example example, if you want to form string searches for personal names in your document by "family name first style," use the following string as a replacement string.

\2, \1

After hitting the "Replace" button, family names will come first, then a comma, and the given names will follow.

- | multiple search
Multiple strings can be searched simultaneously. Use | to specify multiple find strings, as in:

<expression1> | <expression2> | <expression3>...

You may not, however, use the | mark within the character class [] or group { }.

When you use the group expression in a multiple search, each group number will be assigned for each find expressions as follows:

Find date string 99.99.99 or 99/99/99 and replace it with 99-99-99

The find string should therefore be:

$$\frac{\{ [0-9]^+ \} \backslash .}{\backslash 1} \frac{\{ [0-9]^+ \} \backslash .}{\backslash 2} \frac{\{ [0-9]^+ \}}{\backslash 3} | \frac{\{ [0-9]^+ \} /}{\backslash 1} \frac{\{ [0-9]^+ \} /}{\backslash 2} \frac{\{ [0-9]^+ \}}{\backslash 3}$$

The replace string should be:

\1-\2-\3

7.3 Replace

To replace one string with another string, the replacement function is useful.

First, be sure to fill both the find and replacement field within the "Find Settings" dialog. Next, search for the find string by hitting the "Find" button or by selecting "Find Next" under the menu "Search".

When the string is found, commence the replacement process by using either the "Replace" button in the find dialog or by selecting the "Replace" and "Replace & Find Next" items under the menu "Search".

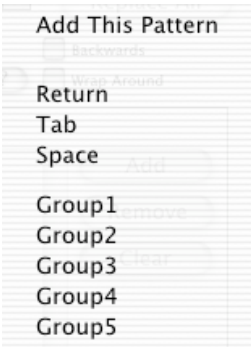
By selecting "Replace & Find Next", the string will be substituted with the replacement string and the search for the next occurrence of the find string will continue.

If you select "Replace All" under "Search" or in the find dialog, all occurrences in the selected area will be substituted with the replacement string.

When a target area is not selected, the "Replace All" will be carried out from the cursor position to the end of the document.

If you push the  button at the upper right side of the replace field, the pop-up menu containing last 10 replacement strings will appear. If you select an item, it will be copied into the replace field.

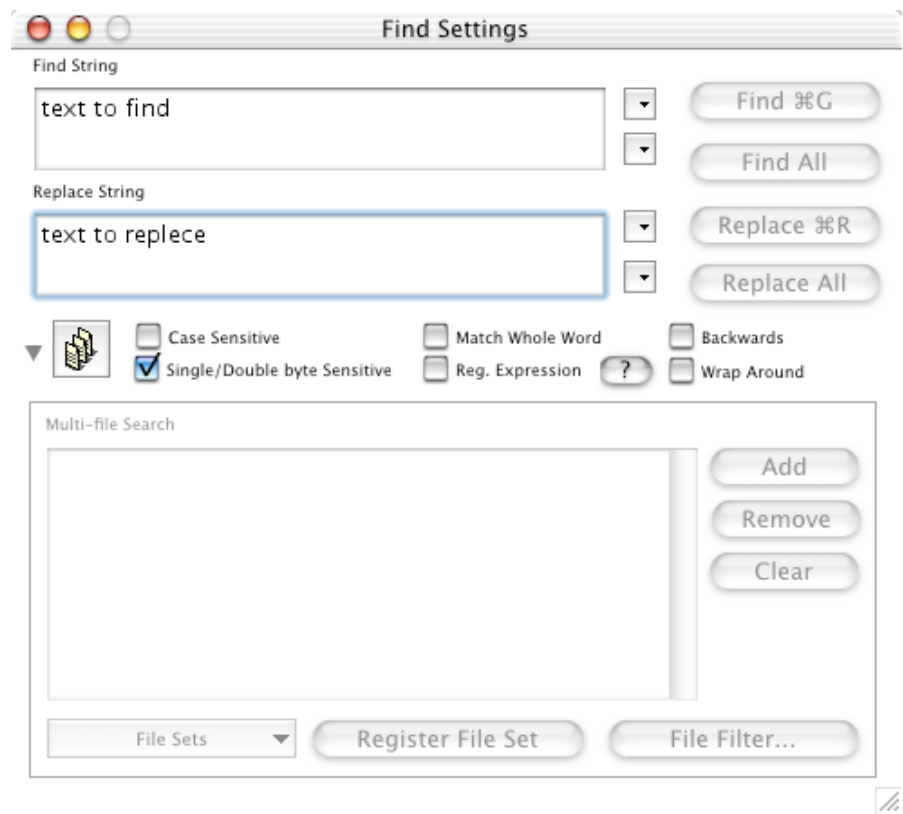
If you push the  button at the lower right side of the replace field, the sample pattern pop up menu as follows will appear.



To add your own replacement patterns, type the pattern into the replacement field and select the "Add This Pattern" menu item. To delete a menu item added, select the item while pressing the option key.

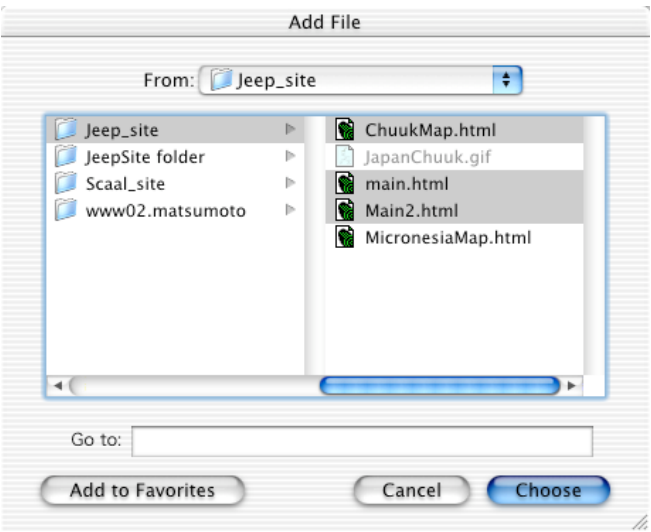
7.4 Multi-File Search & Replace

JChecker can find and replace strings among multiple files.



To switch from normal searching to the multi-file search mode, push the  button . The depressed button will take on the  appearance indicating that multi-file search mode is on. Pushing the small arrow  at left side of the button, shows or hides the target file list for the multi-file search.

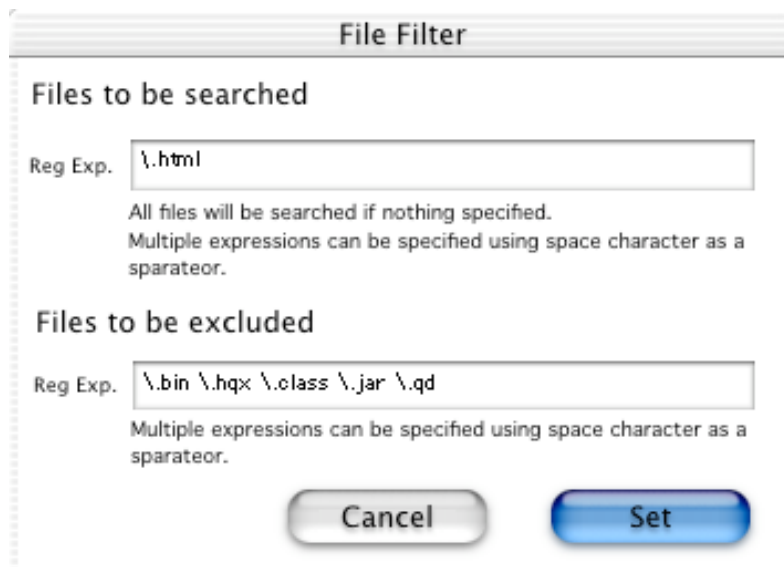
To add files to the target file list, press the "Add" button. The following file dialog will appear displaying a dialog listing files that match "Readable File Types" within "Open" preferences.



You can select more than one file within this dialog and push the "Choose" button. The files will be added to the target file list. Also, by selecting a folder, all files contained within the folder as well as its sub-folders are added to the target file list.

You can also drag and drop files or folders directly onto the target list. You may find this method much easier than using the "Add" button.

In cases such as dropping a folder onto the list, you might want to include or exclude some files. The file name filter enables you to choose files by regular expression. Press the "File Filter" button. A dialog as follows will pop up.



In the above example, files , which have the extensions of ".bin",".hqx", ".class", ".jar", or ".qd", excluded from the searching list.

To a multi-file search, press "Find" or "Find All" button located in the find dialog, or select "Find Next" or "Find All" under the menu "Search."

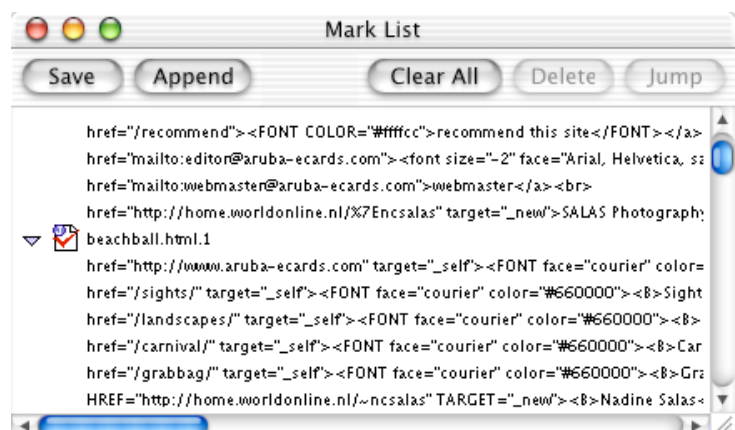


When press the "Find" button, a document window will be opened for each time the string is found. To continue searching, push the "Find" button again or select "Find Next." If you wish to end a search with the current document and commence a new search in the next documents, select the "Find Next File" option under "Search."

To locate the next desired file, select the menu "Find Next File."

Also in multi-file search mode, "Replace & Find Next" method, under the "Search" menu, functions the same as when in the normal search mode.

By pressing the "Find All" button located within the find dialog, or by selecting "Find All" from the "Search" menu, all string occurrences found within denoted target files will be listed within the 'mark list' as exemplified below:



By double clicking an item in the mark list, you will be taken to the corresponding line in the document window. If you double click a file item in the mark list, the corresponding file will be opened.

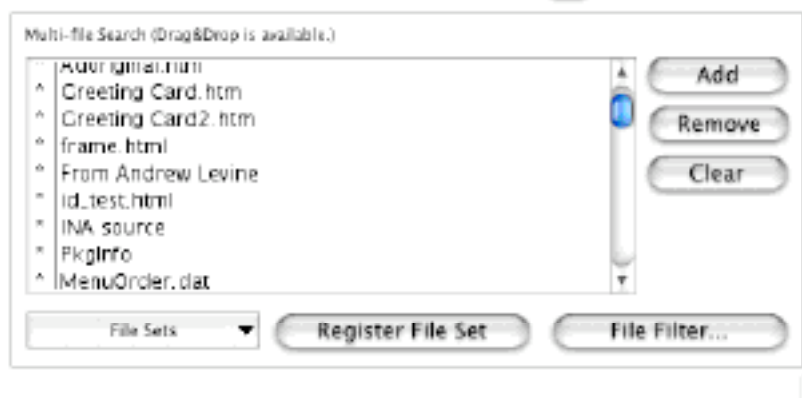
When when "Replace All," all searched string occurrences found in the target files can be replaced with the desired replacement string. upon completion, the following confirmation dialog will appear.



Warning: Do not forget to save documents before closing them, risking loss of any replacements you have made.

How To Use the Target File List from the "Find Setting Dialog"

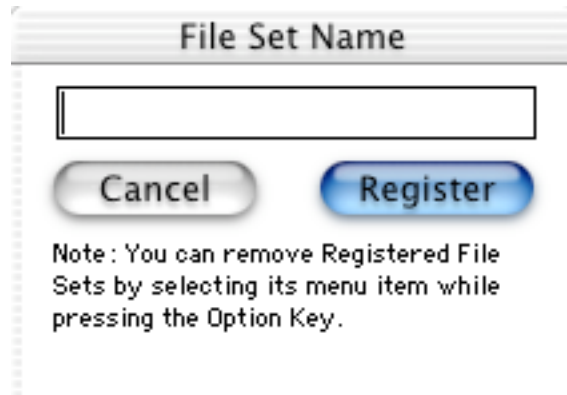
If you want to remove a file from your target file list, select the name of that file and simply push the "Remove" button. If you want to clear all files from the list box, use the "Clear" command instead.



You can easily open a file in the target file list by double clicking on the name of that file.

"*" marks denote that a file has not yet been searched. If you want to skip a file while searching, remove the "*" mark to the left of the file's name by clicking it. Clicking the "*" mark with the shift key down removes all "*" marked files located above the point clicked.

If the same target list is used frequently, the list may be registered under the "File Sets" pop-up menu. Push the "Register File Set" button, and the following dialog will appear.



Enter a name for the set you wish to register and hit "Register." This allows the retrieval of the registered file set(s) any time after by going to the "File Sets" pop-up menu.

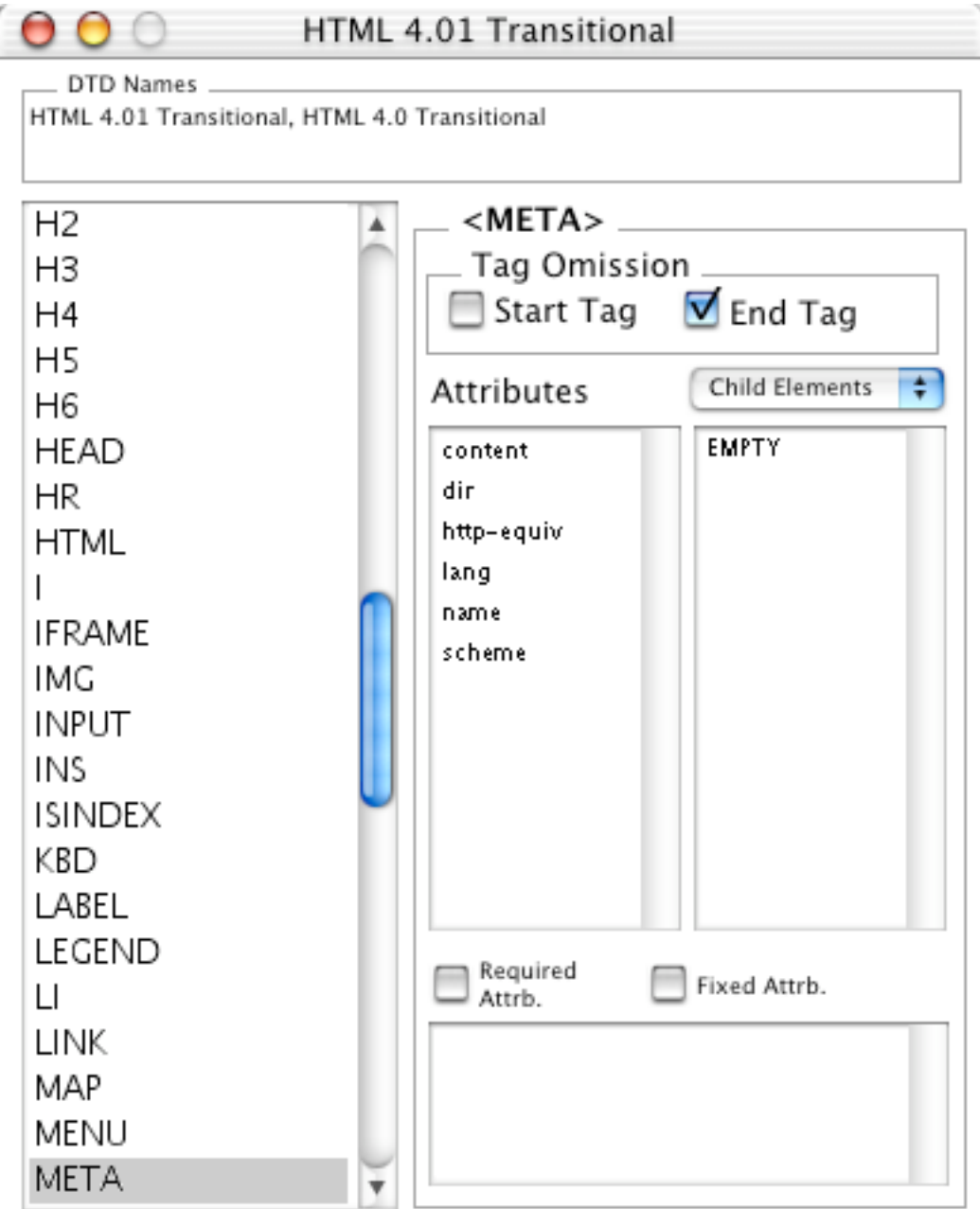
To use registered file set(s), press the pop-up "File Sets" menu button and select the name of sets you have created. All file names under that set will then appear in the target file list.

To remove a file set from the "File Sets" pop-up menu, push the pop-up menu button with the option key down. Select the name of file set you want to remove.

7.5 Consulting Dictionaries

If you wish to look up HTML elements in a dictionary, set the cursor inside the tag in question and select the menu item "Search > Find Definition". The following dictionary dialog will be displayed. (For more about the dictionary dialog, refer the chapter 9.)

The dictionary may also be accessed by clicking the dictionary button  located in the upper portion of the editing window.



7.6 Mark

You can mark locations of specific text areas in a document and list these locations in a mark list. This option offers convenient jump-backs to a targeted text area at any time.

Highlight the text area you wish to target and select "Mark" from the "Search" menu. The mark list will open automatically and the targeted location will be registered onto the list.

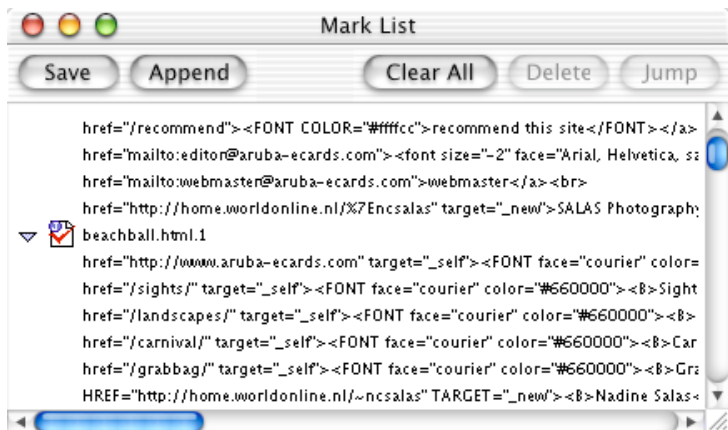


Text areas may be further added by "Drag and Drop" method. Simply drag and drop a selected text area directly onto the mark list.

To jump to target text areas, simply double click the corresponding items in the mark list.

7.7 Mark List

Go to the "Search" menu and select "Mark List." The following mark list will appear:



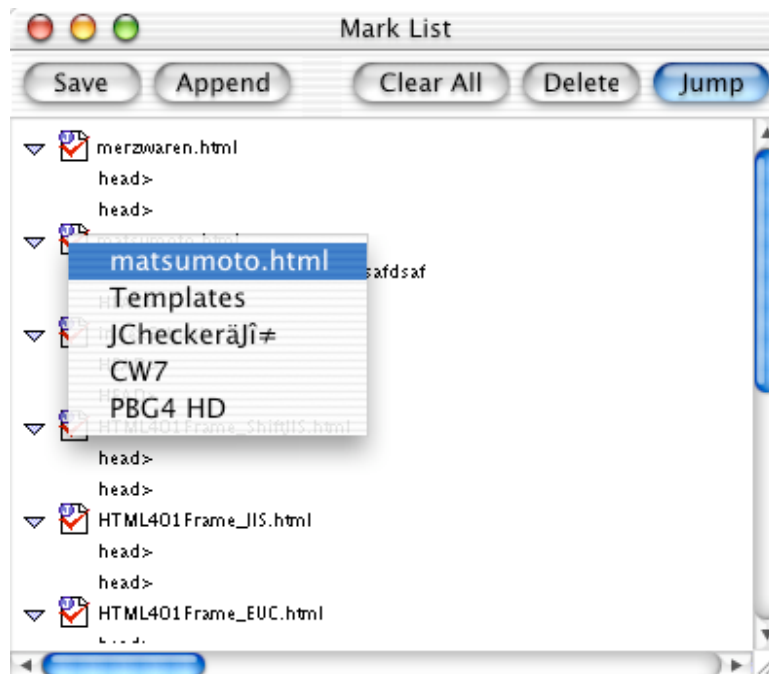
The mark list can contain not only lines that you have targeted but also search results from "Find All" function.

You can show or hide marked sub-items under file by clicking the  denotation to the left side of each file item. If you want to show or hide all marked items on the list, click one of the marks while depressing the option key.

Directory Menu

You gain a file's name from the Mark list, as well as its location (file path). To check out the file path, click the file icon in the Mark list with the command key down. The following directory menu will pop-up. When you select a directory under the menu, the

directory window will be opened in the Finder.



Jump

To open a file in the mark list using its specific creator application, double click on that file item.

To have JChecker open a marked item, double click on the item. JChecker will jump to the corresponding text. You can also open marked items by hitting the "Jump" button located at the top of the mark list.

In the event that a file is already opened as a JChecker document, that file's icon will display using the  shape rather than that of its original icon.

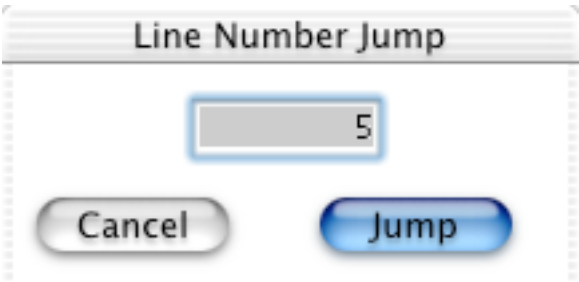
Remove

To remove a marked item from the mark list, select that item and push the "Delete" button. All marks of a given file may be removed by selecting the file item itself and then pushing the "Delete" button. To remove all marks in the mark list, press the "Clear All" button.

JChecker will never erase mark items automatically. New search results will be added to the current mark list, and will remain in the list unless or until they are removed.

7.8 Jump by Line Number

Go to the "Search" menu and select "Go to Line". The following dialog will appear:



Enter the line number of the area to which you wish to jump and press the "Jump" button.

Chapter 8 Check

8.1 Validating HTML

By selecting the menu "Check > Validate HTML" or otherwise clicking the  button, HTML will be checked. When JChecker detects any error or warning, these errors will display in the message area.

Clicking on an error message in the message area automatically highlights the appropriate line corresponding to the text area, in the editing area.

If you double click an error message, a pop up window displays containing an explanation of the error.

Validation is conducted according to the dictionary selected in the dictionary pop up menu in the upper portion of the window. When the item "Follow <DOCTYPE...>" is selected within the dictionary pop up menu, the dictionary matching the declared DOCTYPE will be selected automatically.

Typically, it is recommended to select "Follow <DOCTYPE...>" or "HTML 4.01 Transitional" in the dictionary pop up menu.

When you validate your HTML for the first time, you may be surprised by the number of errors and warnings that JChecker outputs.

By selecting the menu "Check > Auto Correction", simple errors will be automatically corrected. About 20 kinds of errors are supported by auto correction.

Errors that destroy HTML hierarchical structures, such as lacking end tags or incorrect sub-elements, will be a source of and induce many more hierarchical errors. If you were to get an error "Can't find the end tag", yet seeing that the end tag exists, the elements recognition is most likely being hindered due to a wrong sub-element, thus destroying the hierarchical structure. If you correct the principal error, all other errors induced should as well disappear.

8.2 Checking Links

Select the menu "Check > Check Links" or click the  button with the shift key down. The destinations linked from the HTML are checked for valid existence. If errors or warnings are found, they will display in the message area.

In the link preferences, you can specify the type of link to be checked (i.e. local or remote links).

JChecker checks links with the attribute value 'CDATAURI' as defined in the dictionary. Values of URI attributes are:

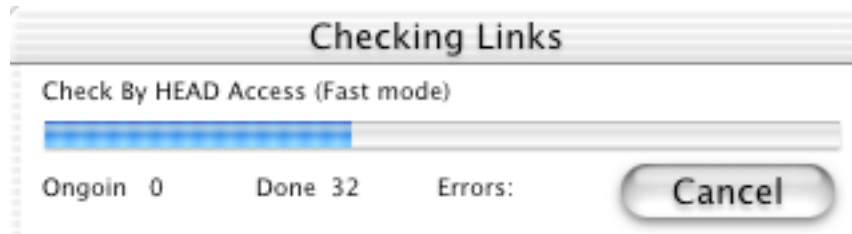
"background", "background-image", "cite", "data", "classid",
"longdesc", "pluginspage", "profile", "action", "src", "usemap", "codebase", "href"

Values for elements are:

"A", "BODY", "LAYER", "BGSOUND", "EMBED", "MARQUEE", "OBJECT", "ILAYER", "FORM",
"DEL", "INS", "INPUT", "IMG", "APPLET", "AREA", "LINK", "BASE"

Note: The links which schemes are "ftp://" or "mailto:" will not be checked.

When you access the link check menu, a progress dialog like the one below appears.



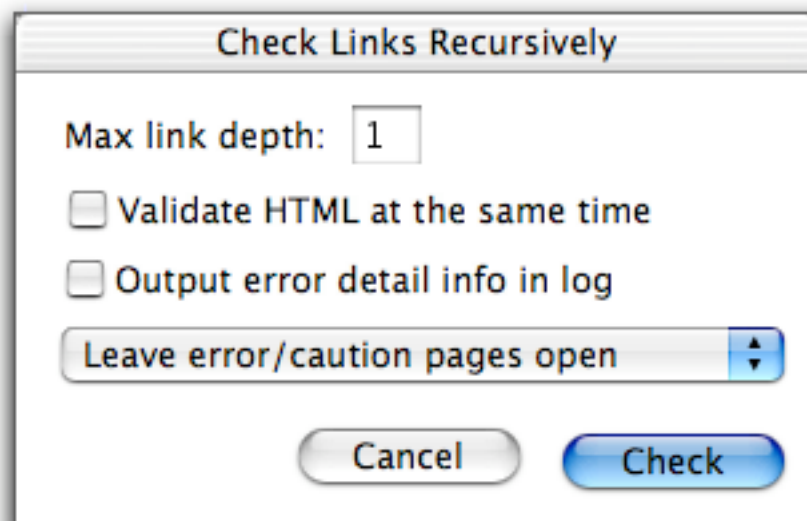
8.3 Checking Both

Select the menu "Check > Check Both." The both validation and link checking will be conducted.

8.4 Checking Links Recursively

By selecting the menu item "Check > Check Links Recursively...", all HTML documents linked from the HTML in the editing window are checked.

When selecting the menu item "Check > Check Links Recursively...", the following dialog will appear.



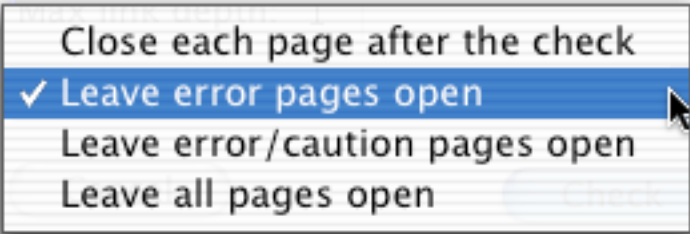
By setting the value to "1" in the "Max link depth" field, the parent page as well any child pages will be checked, If you set the value to "2" in the depth field, parent, child and grandchild pages will be checked.

By setting the checkbox "Validate HTML at the same time " to ON, both HTML validation and link checking will be executed. (Note: As the maximum length of the log is 32KB, old messages that exceed 32KB will be deleted automatically.)

By setting the checkbox "Output error detail info in log " to ON, each error messages will be displayed in additon to error summary.

Choose the depth value carefully as the number of pages potentially checked will increase at an exponential rate proportional to the depth value.

By selecting the popup menu, you can choose a option whether each window should be closed or not after the check.

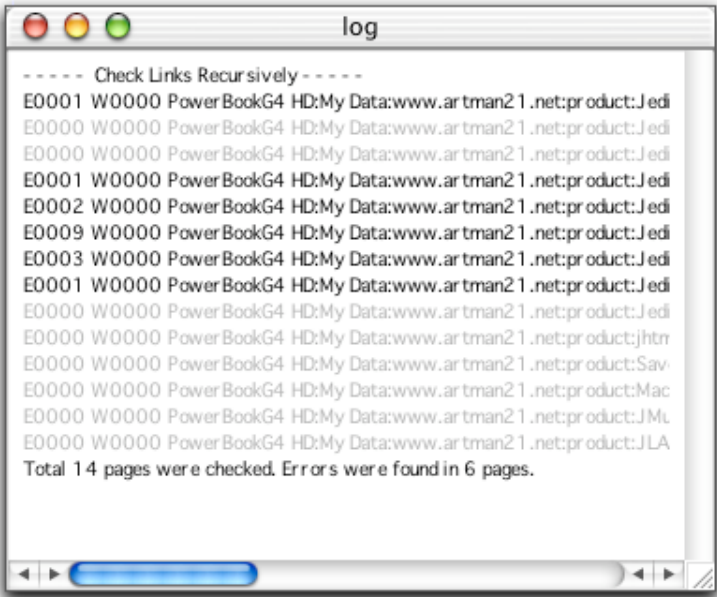


Upon clicking the "Check" button, checking commences and the following progress dialog will appear.



If you want to cancel the checking process, hit "Cancel" in the progress bar.

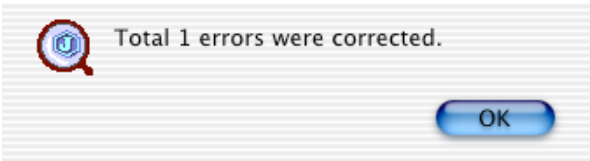
The result summary of checking will be displayed on a log window as follows.



If you double-click a path name in the log window, the file of its path name will be opened in an editing window. You can also save the logs with the menu "File > Save."

8.4 Auto Correction

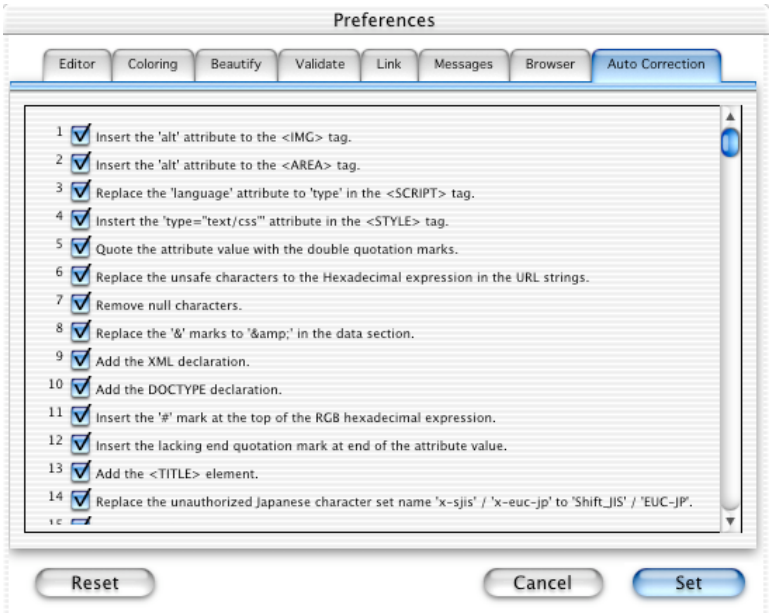
By selecting the menu item "Check > Auto Correction", JChecker automatically corrects about 20 kinds of simple errors. Upon completion, the number or amount of corrections will be notified as follows.



JChecker automatically corrects the following errors:

- Insert the 'alt' attribute to the tag.
- Insert the 'alt' attribute to the <AREA> tag.
- Replace the 'language' attribute to 'type' in the <SCRIPT> tag.
- Insert the 'type="text/css"' attribute in the <STYLE> tag.
- Quote the attribute value with the double quotation marks.
- Replace unsafe characters to Hexadecimal expression in the URL strings.
- Remove null characters.
- Replace the '&' marks to '&' in the data section.
- Add the XML declaration.
- Add the DOCTYPE declaration.
- Insert the '#' mark at the top of the RGB hexadecimal expression.
- Insert lacking end quotation mark at end of attribute value.
- Add the <TITLE> element.
- Replace the unauthorized Japanese character set name 'x-sjis' / 'x-euc-jp' to 'Shift_JIS' / 'EUC-JP'.
- Correct fixed(#FIXED) attribute value.
- Insert lacking attribute values which can't be omitted in XML.
- Correct attributes which name and value should be same.
- Add <HEAD> and <TITLE> elements.
- Remove undefined attributes that were inserted by some HTML editors for their editing convenience.
- Mend the character case in the DOCTYPE declaration.

Selecting the menu "Check > Auto Correction Settings", you can specify the ON/OFF of each correction item.



8.5 Beautifying HTML

The menu "Check">"Beautify HTML" will beautify an HTML document, making it easier to logically view hierarchical structures of tags. This function is helpful when you want to identify hierarchy errors reported.

When you can't find the cause of an error, beautifying will help you to detect a hierarchical error.

Note: the following warning may appear when validation errors are detected when interpreting hierarchical structures.

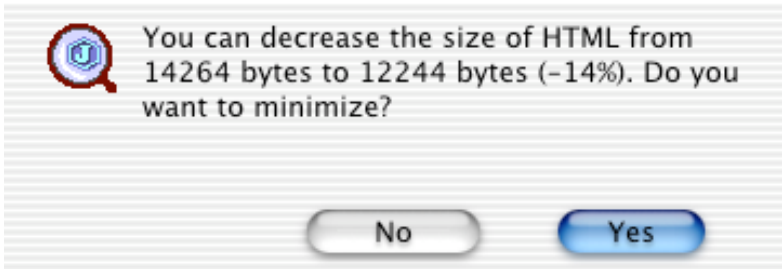


Various options for "Beautifying" may be specified within "Beautify" preferences. For more information, Please see "3.3 Beautify."

8.6 Minimizing HTML

Browsers interpret continuous white spaces (space, tab and line break codes) as single spaces. For the purpose of actually rendering HTML, many unnecessary white spaces may exist in your HTML code. This menu strips such redundant white spacing.

Selecting the menu item "Check > Minimize HTML", the following dialog appears, telling you the results of minimizing your file:



In most cases, minimization can decrease the size of your HTML 10% or more. This reduction decreases transfer and load time and saves server space.

Once minimized, it may be difficult to read, and therefore modify, HTML text. You can easily revert to a more human readable form also by using the "Beautify HTML" menu .

Note: Minimization will not be conducted on the contents of the elements PRE, SCRIPT, XMP, LISTING, PLAINTEXT, STYLE, or SERVER.

8.7 Re-coloring HTML

HTML text is automatically distinguished by using color coding for discerning differences among tags. However, in cases where coloring has become incorrect, simply use the menu item "Check > Re-Coloring" to restore correct coloring.

Chapter 9 Working with Other Editors

As JChecker supports "Word Services Protocol" in Classic OS and "Services Menu" in Mac OS X, you can check other editor's HTML documents with JChecker.

9.1 Check Using Word Services

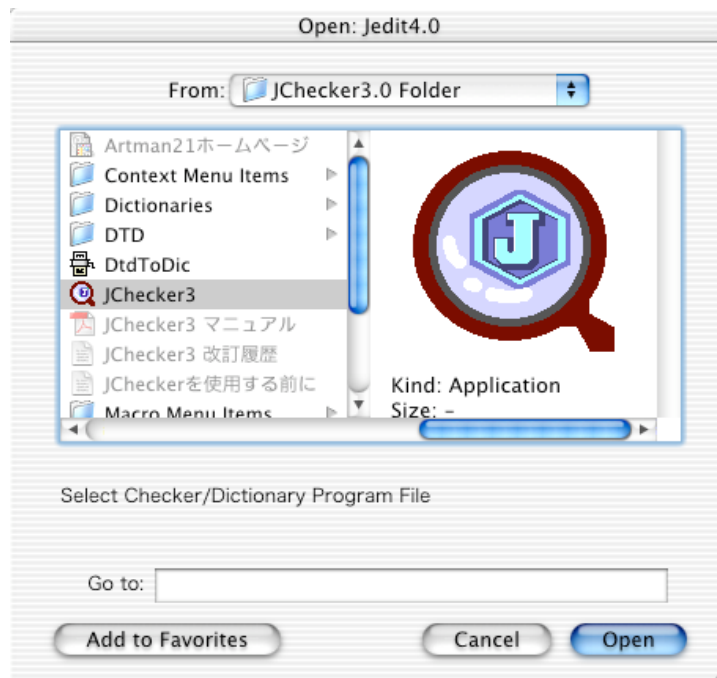
Since JChecker supports Word Service Protocols, it can be used as a validation checking server for other editors that are compatible with Word Service such as BBEdit or Jedit4.0. Jedit also can use JChecker as a Word Services Server both in Classic OS and Mac OS X. Note however: other editors are currently unable to use Word Services protocol under Mac OS X.

Nisus Writer, Word Perfect, Eudora and Claris Works are also compatible with Word Services. Information regarding those applications compatible with Word Service may be obtained at:

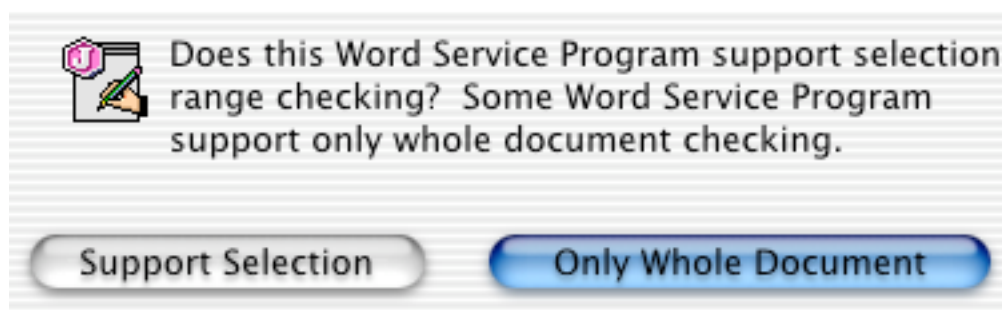
<http://www.wordservices.org/>

Setting JChecker as Jedit's External Checker.

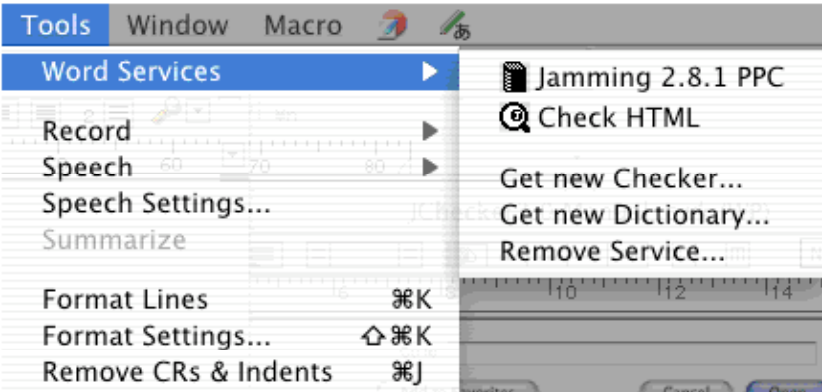
First, start up Jedit4.0 and select menu "Tool">"Word Services > Get New Checker".



The file dialog requesting the desired checker program will appear. Specify the "JChecker3" program and click "OK".



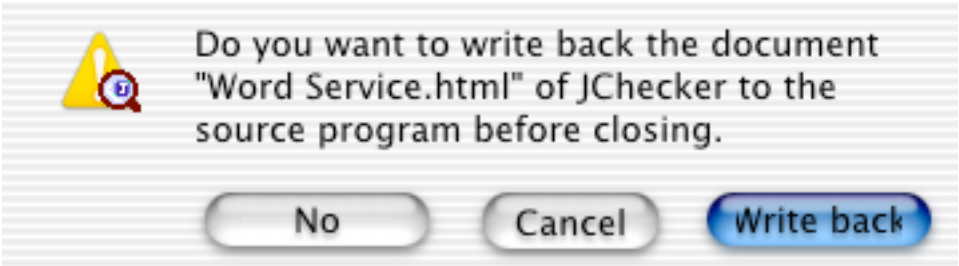
A dialog asking whether JChecker supports the selection check will appear. Click the button "Only Whole Document". The "Check HTML" menu will now be added to the Word Services menu as follows:



Invoking JChecker through Jedit.

After you have set up Jedit as shown above, when the Jedit menu "Tool">"Word Services > Check HTML" is selected, JChecker will start up automatically. You can now check the validity of HTML texts from within Jedit.

The following dialog box will be displayed when the JChecker window is closed. By clicking "Write Back," the content modified in JChecker will be written back to the window of Jedit. The keyboard short-cut for the button "No" is cmd+D.



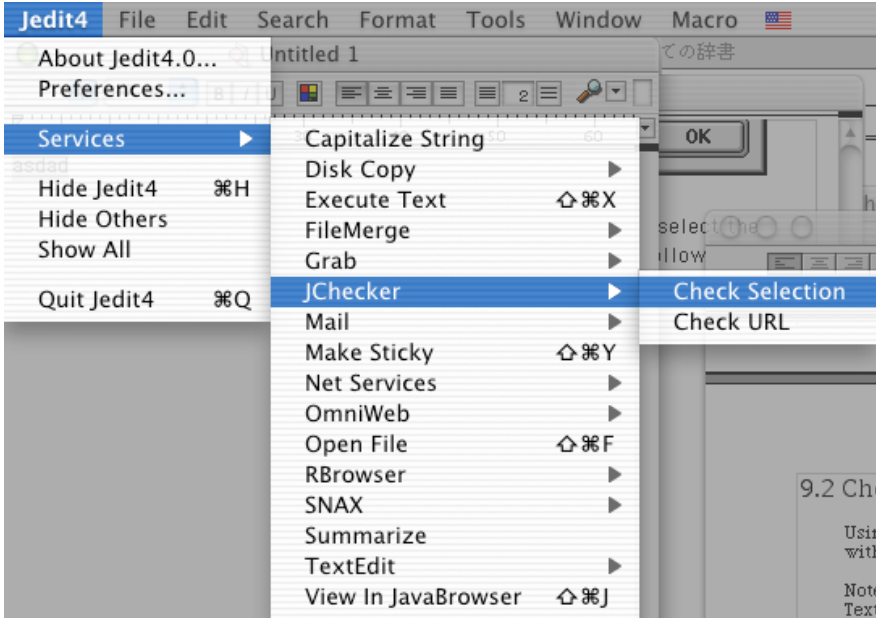
9.2 Check Using Services Menu

By using the "Services Menu" supported under Mac OS X, you can as well check another editor's HTML by using JChecker.

Note: As of March 2002, under Mac OS X 10.1.3, Cocoa applications such as TextEdit cannot call up JChecker through the "Services Menu". Only Carbon applications which supports the "Services Menu" protocol can use JChecker as an external checker.

Invoking JChecker through Jedit.

Select a whole HTML document through the Jedit menu "Edit > Select All". Select the menu "Jedit4 > Services > JChecker > Check Selection".



JChecker will be invoked and the HTML contents will be copied to the JChecker window and checked automatically.

If you select the menu item "Jedit4 > Services > JChecker > Check URL", JChecker will open the URL selected within Jedit and automatically check it.

Note: by using the "Services Menu", you will be unable to write back the contents from JChecker to the editor.

Chapter 10 Dictionaries

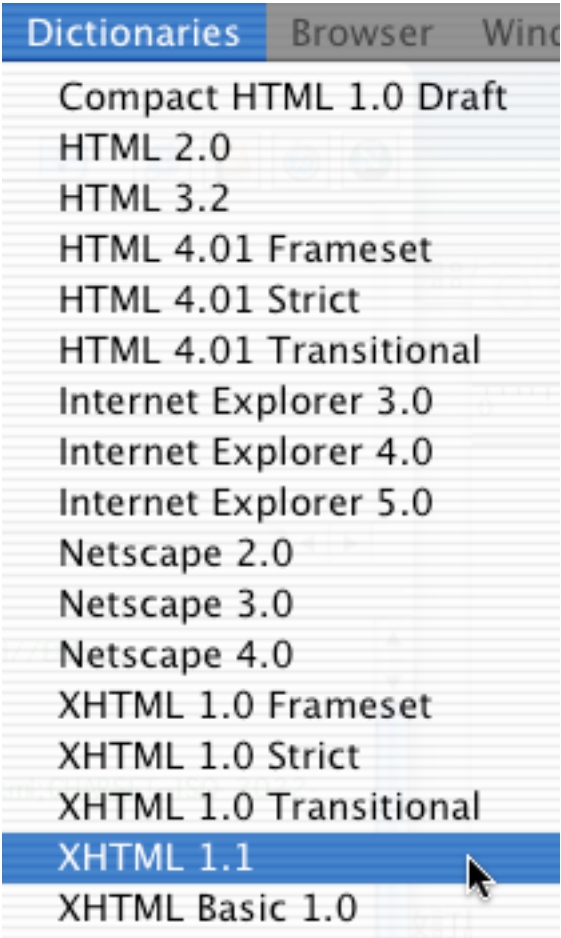
Many versions of HTML exist, such as HTML2.0, HTML3.2, HTML4.01, XHTML1.0, etc. Specification of elements and attributes differ among each HTML version. JChecker supports multiple dictionaries that recognize different rules for each HTML version. Dictionaries may even be created and or customized.

Dictionary files are stored within the "Dictionaries" folder in the "JChecker3.0" folder.

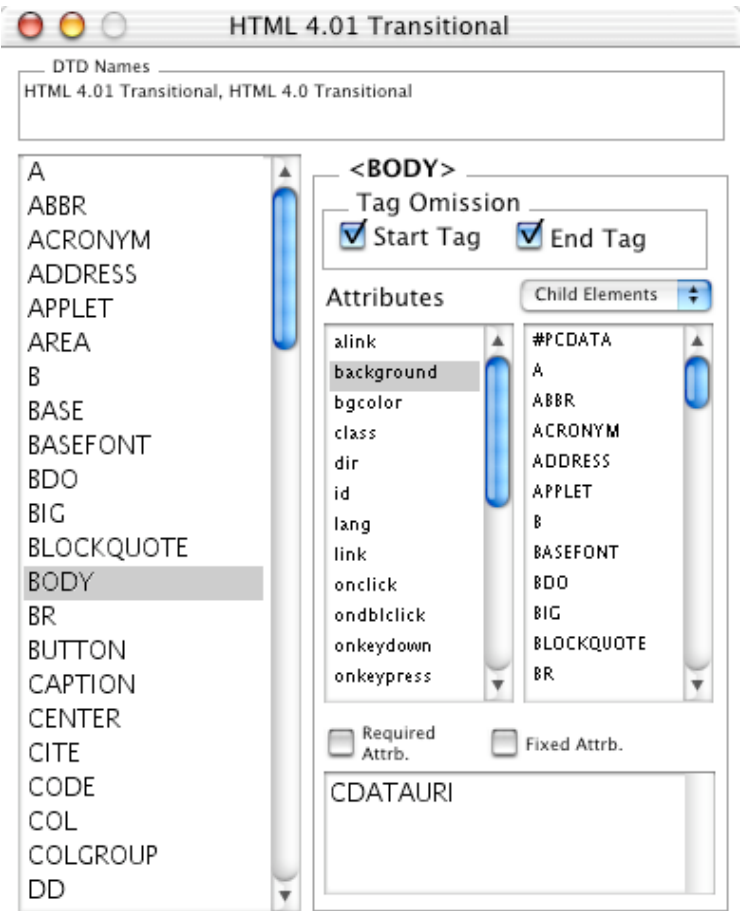


10.1 Dictionary Window

By selecting the menu item "Dictionaries", those dictionaries installed in JChecker will be shown alphabetically:



Select the dictionary you wish to consult. After, a dictionary window follows:



DTD names used in the DOCTYPE declarations are displayed within the "DTD Names" text box at the upper side of the window. When looking at the above example, the dictionary "HTML 4.01 Transitional" has been selected for checking when the DTD name "HTML 4.01 Transitional" or "HTML 4.0 Transitional" is specified within the DOCTYPE declaration.

At the left side of the widow, available elements for this HTML version are listed in alphabetical order. Clicking one of the elements, information about that specific element will be shown to the right.

The "Tag Omission" text box offers information regarding whether you can omit start or end tags. (Note: you cannot ater the check boxes themselves. These check boxes are read-only.)

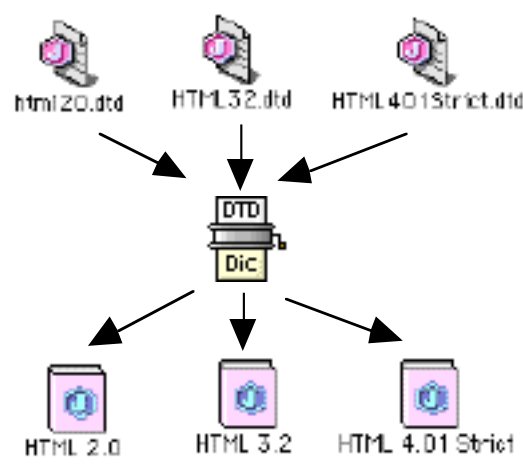
In the "Attributes" list, attributes available for elements are listed alphabetically. Clicking an attribute displays information about that attribute below. Also shown is whether an attribute is required, has a fixed value, and its value type.

When "Child elements" is selected in the pop up menu in the right most list area, child elements will be offered in that area. If you click one of the child elements, dictionary information will be re assigned for that element.

When "Parent elements" is selected in the pop up menu, parent elements that can contain in this element will be displayed.

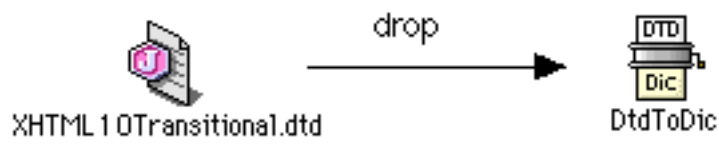
10.2 Generating A Dictionary (DtdToDic)

You can generate JChecker dictionaries from the public DTD files by using the program "DtdToDic" located in the JChecker3.0 folder.

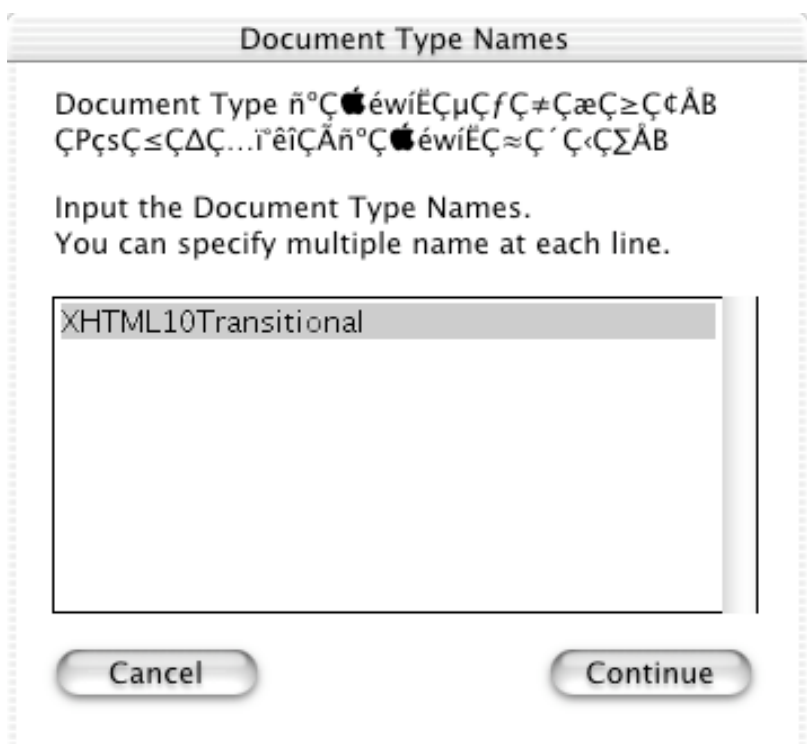


The DTD files, from which JChecker dictionaries were generated, are stored within the folder "DTD". Also the file "public DTD sites" contains the public URL of those DTDs.

To generate a dictionary, drop the DTD file onto the "DtdToDic" program icon.



A dialog asking for DTD names within the DOCTYPE declaration will appear. You can specify a multitude of DTD names.



By clicking "Continue", dictionary generation commences and a Dictionary file, as shown below, will be created and located within the same folder as the DTD file.



Move a desired dictionary files into the folder "Dictionaries" found in the "JChecker3.0 folder" and launch JChecker3.o. You will see the new dictionary in the menu "Dictionaries".

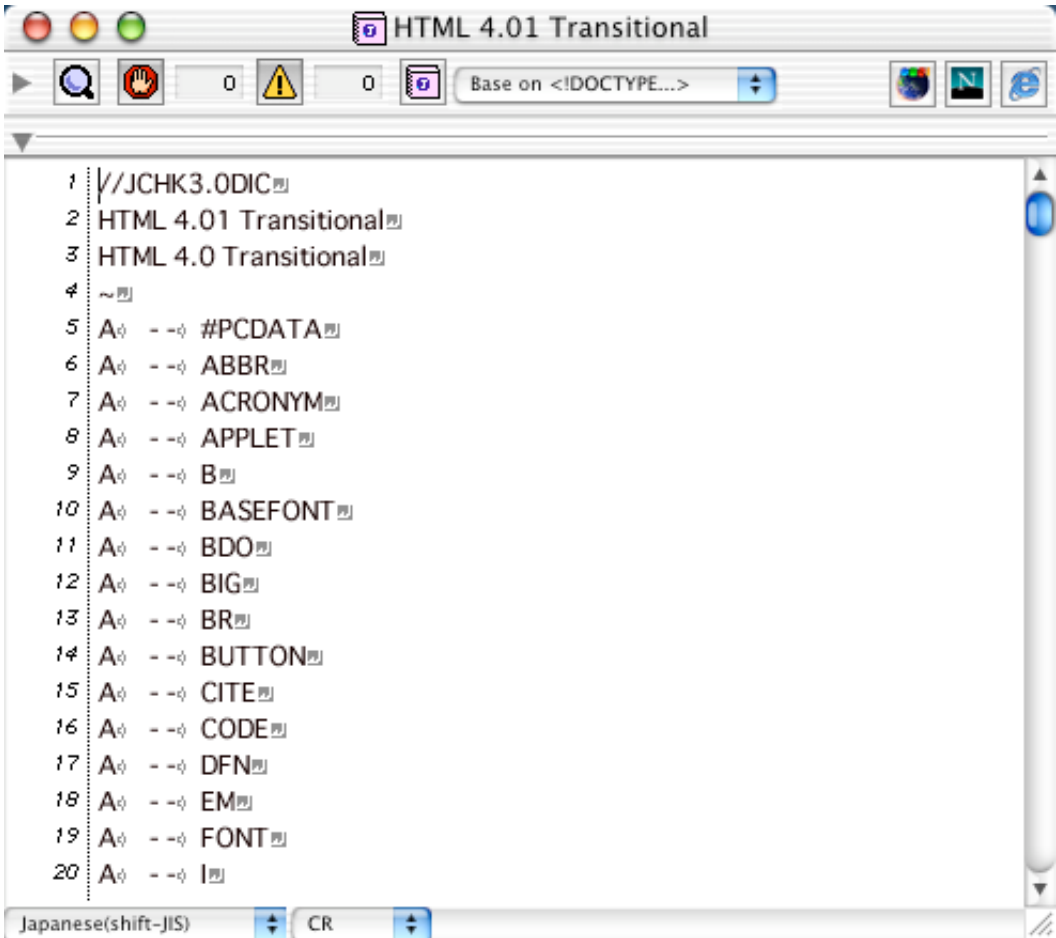
Note Regarding the "DtdToDic" Program

- Dictionary generation will fail if the related DTD file contains errors .
- The "DtdToDic" program is a Carbon application.

10.3 The Structure of the Dictionary File

As the dictionary file is a plain text file, you can see its contents directly with a text editor or by using JChecker itself. You can easily edit its contents as its structure is also very simple.

Double Clicking the dictionary file icon, the contents of the dictionary will be displayed in the JChecker window.



A dictionary consists of three parts: 1) the DTD names declaration, 2) element definition, and 3) attribute definition. These three parts are separated by a line which contains only a '~' character.

In the above example (demonstrating the HTML 4.01 Transitional dictionary) :

- the first line "//JCHK3.0DIC" indicates that this file is a JChecker3.0 dictionary.
- (1) 2nd to 4th line "~": the DTD names declaration
- (2) 5th to 2678th line "~": the element definition
- (3) 2679th to the last 4190th line: the attribute definition

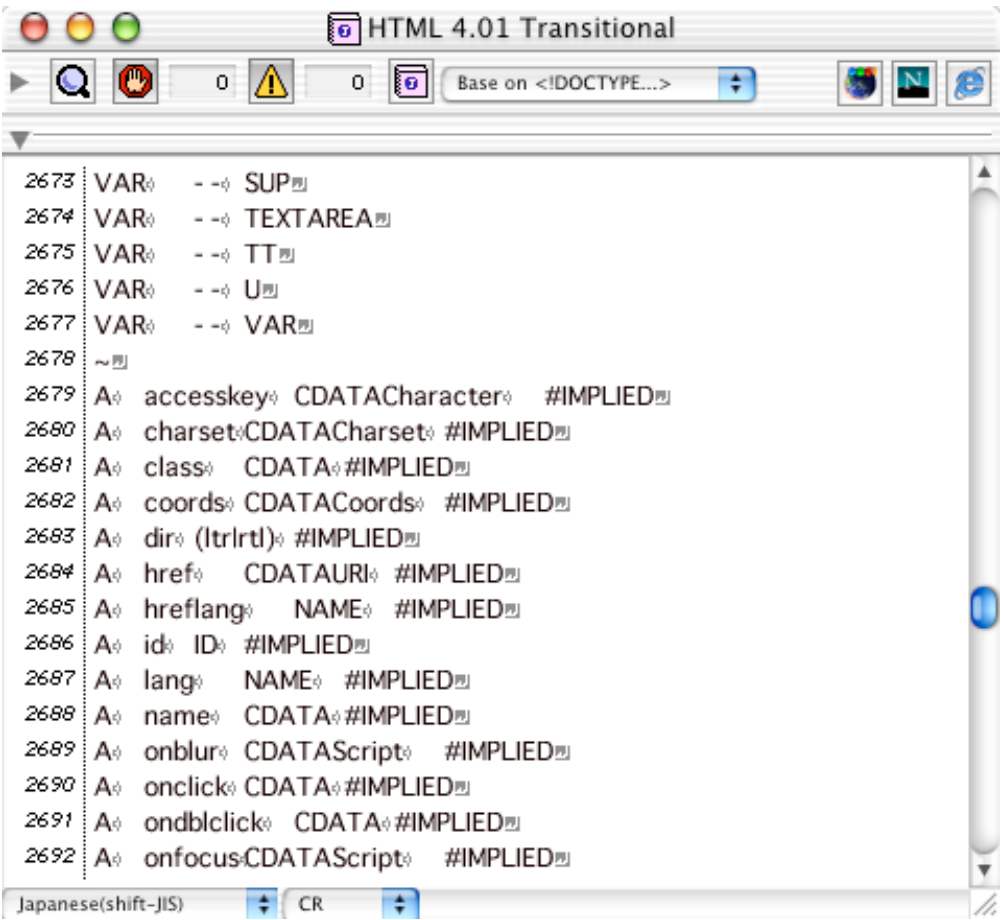
Under the element definition, elements are defined alphabetically beginning with the element "A", continuing through element "VAR". One element consists of multiple lines corresponding to related child elements. The format of such lines are as follows.

[element name] tab [tag omission info] tab [child element name]

"tag omission info" indicates omitted information designated by 3 character sets:

- - start tag and end tag can't be omitted.
- O only end tag can be omitted.
- O O both start tag and end tag can be omitted

The character "O" signifies the capital letter of the word "omit". The "tag omission info" must be identical for each corresponding element.



In the above figure, element definition ends at the line 2677th. Line 2678th is a separator, and attribute definition starts from line 2679th.

Within attribute definition, attributes are defined for each element from the element "A" to element "VAR". One element consists of multiple lines corresponding to its available attributes. The format of such line is as follows.

[element name] tab [attribute name] tab [value type] tab [state]

The "value type" describes the type of the attribute value or the correction of feasible values. The types may be CDATA, NAME, ID, NUMBER among others.

The "state" describes the default value of an attribute or key word using the "#" mark:

- #IMPLIED if the attribute was omitted, the attribute value will be inherited from the parent element or the user agent.
- #REQUIRED the attribute you must specify in the tag
- #FIXED the attribute value has a fixed value.

For example:

CAPTION **align** (top|bottom|left|right) **#IMPLIED**

The value for the attribute "align" corresponding to the element "CAPTION" should be denoted either top, bottom, left, or right. If this attribute is omitted, the attribute value of this element will be inherited from the parent element or the default value of the user agent.

IMG **alt** **CDATA**Text **#REQUIRED**

You must specify the "alt" attribute within the "IMG" tag. The value of the "alt" should match character data (CDATA).

IFRAME **scrolling** (yes|no|auto) **auto**

The value of the attribute "scrolling" in the "IFRAME" tag should adhere either to 'yes', 'no', or to 'auto'. If this attribute is omitted, the default value "auto" will be assigned for this element.

10.4 Customizing the Dictionary

There are two methods for customizing a dictionary.

- 1) edit the contents of the dictionary directly
- 2) edit the contents of the DTD and regenerate the dictionary using "DtdToDic"

This section as an example, explains how to add the attribute "leftmargin" and "topmargin" to the "BODY" element of the "HTML4.01 Transitional" dictionary.

Method 1: Modifying the Dictionary

Double click the "HTML4.01 Transitional" dictionary icon. You will see the contents of the dictionary in the JChecker editing window.

Using the search function of JChecker, find the attribute description area of the "BODY" element. The attribute definition now commences from line 2859th:

```
BODY    alink  CDATA      #IMPLIED
BODY    background CDATAURI #IMPLIED
BODY    bgcolor CDATA      #IMPLIED
.....
.....
BODY    text   CDATA      #IMPLIED
BODY    title  CDATA      #IMPLIED
BODY    vlink  CDATA      #IMPLIED
```

Add two lines for the "leftmargin" and "topmargin" attribute definition:

```
BODY    alink  CDATA      #IMPLIED
BODY    background CDATAURI #IMPLIED
BODY    bgcolor CDATA      #IMPLIED
.....
BODY    lang   NAME #IMPLIED
BODY    leftmargin CDATA      #IMPLIED
BODY    link   CDATA      #IMPLIED
.....
BODY    title  CDATA      #IMPLIED
BODY    topmargin CDATA      #IMPLIED
BODY    vlink  CDATA      #IMPLIED
```

This newly modified dictionary may now be used after saving.

Method 2: Modifying the DTD

Modifying DTDs may require some knowledge about how DTDs work. Note: if you are not familiar with the DTD, it is advisable to reference appropriate DTD information or books.

To modify a DTD, open the DTD file for "HTML4.01 Transitional" (as an example) using your favorite editor. You will find the attribute definition for the "BODY" element beginning at line 313th.

```
<!ATTLIST BODY
  %attrs;                -- %coreattrs, %i18n, %events --
  onload      %Script; #IMPLIED -- the document has been loaded --
  onunload    %Script; #IMPLIED -- the document has been removed --
  background  %URI;    #IMPLIED -- texture tile for document
                                background --
  %bodycolors;            -- bgcolor, text,link,vlink,alink --
>
```

Add two lines for "leftmargin" and "topmargin" attribute definition:

```
<!ATTLIST BODY
  %attrs;                -- %coreattrs, %i18n, %events --
  onload      %Script; #IMPLIED -- the document has been loaded --
  onunload    %Script; #IMPLIED -- the document has been removed --
  background  %URI;    #IMPLIED -- texture tile for document
                                background --
  %bodycolors;            -- bgcolor, text,link,vlink,alink --
  leftmargin  CDATA      #IMPLIED
  topmargin  CDATA      #IMPLIED
>
```

Once completed, save the DTD file and generate a new dictionary from the DTD with the program "DtdToDic". Make sure to replace the old dictionary within the "Dictionaries" folder with new one.

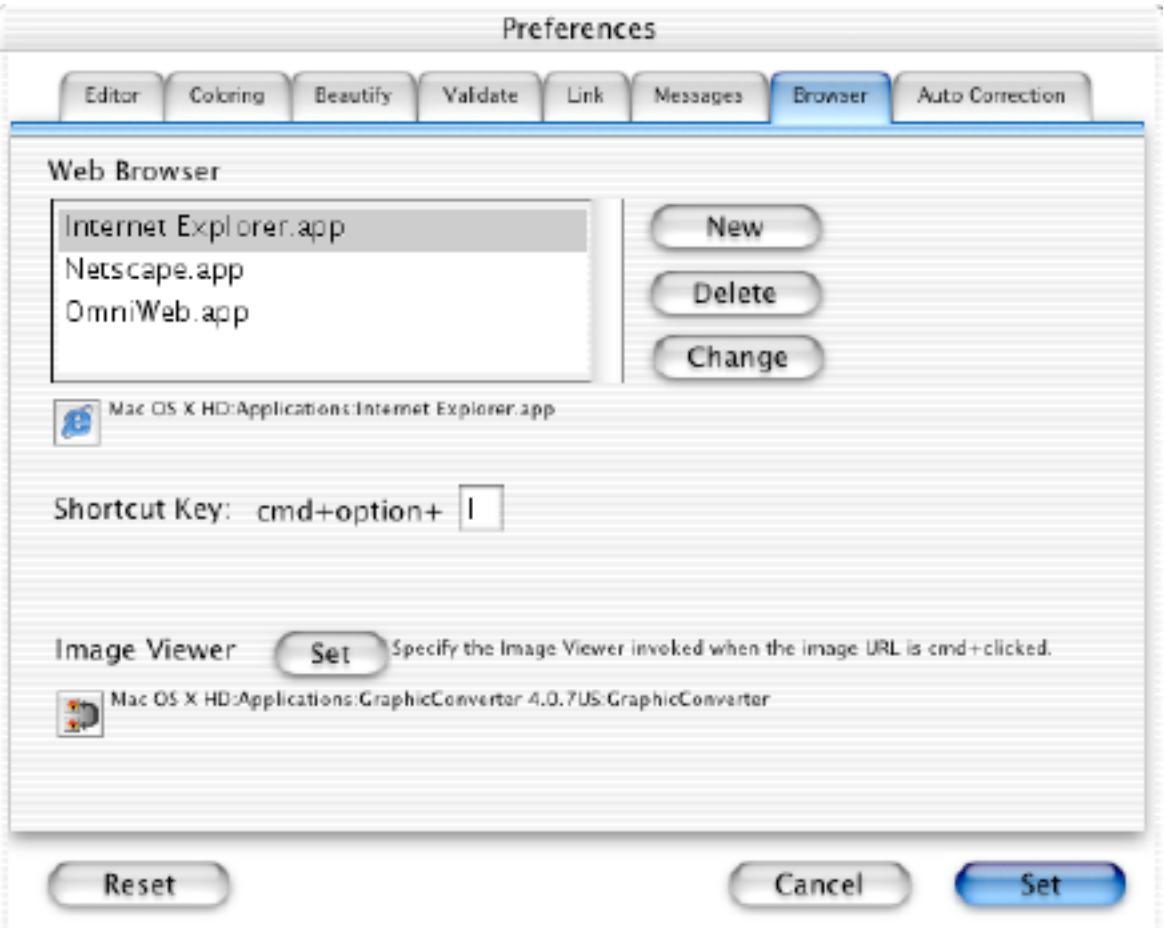
Chapter 11 Previewing with Browsers

With the browser menu , you can preview how current HTML text looks when displayed with a particular browser.

11.1 Registering Browsers

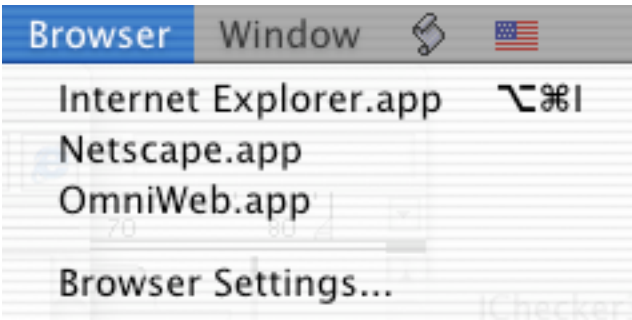
As browsers are not automatically registered in the browser menu at the factory settings, you will first need to register desired browser(s). You can add to the menu as many browsers as you desire.

To do this, select the menu item "Browser > Browser Settings". the Preference window "Browser" will open as seen below. For further on registering methods, refer to Chapter 3, Preferences, 3.7 Browser.



11.2 Previewing with Browsers

By selecting the menu item "Browser", registered browsers items are listed.

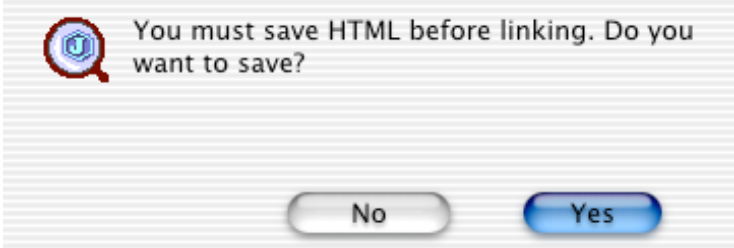


The above shows, three web browsers: Internet Explorer, Netscape and OmniWeb, as being registered and available. By selecting one of the browsers, the browser will be invoked and the HTML will be previewed.

You can also invoke a browser, viewing HTML contents by clicking browser buttons located at the top of the editing window.



When attempting to brows HTML that has not been saved yet, the following dialog will appear.



Click the button "Yes" and save the HTML to the appropriate location.

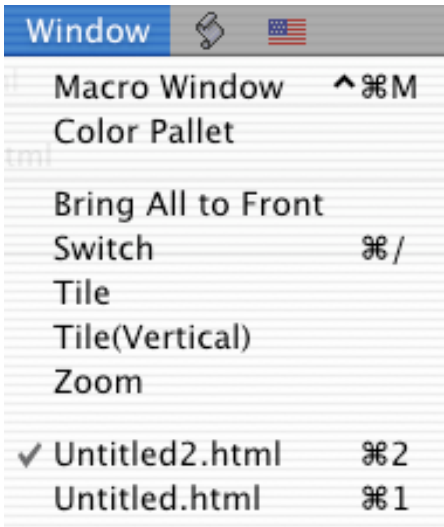
For the previewing, a temporary file whose name starts with '~' mark and is followed by the original file name will be used so that the contents of the original file are not updated by previewing.



The above example demonstrates that a temporary file named "~index.html" is being used as proxy for the original file "index.html". Upon closing the original file's editing window, this temporary file will automatically be deleted.

Chapter 12 Window

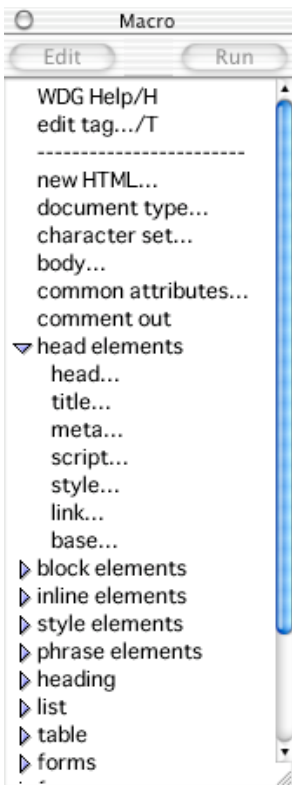
"Window" menus allow you to both arrange and pull up windows quickly.



Beneath the the partition denoting the "Window" menu, all windows open in JChecker will be listed upon clicking this menu. By selecting an item, the corresponding window will become front and active.

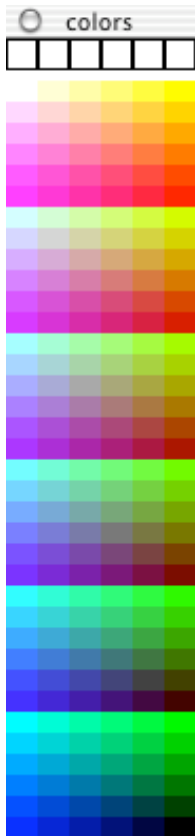
12.1 Macro Window

By selecting the menu item "Window > Macro Window", a macro window as shown below, will appear. For more information on macro windows, refer to Chapter 13 Macro, 13.2 Macro Window.



12.2 Color Pallet

By selecting the menu item "Window > Color Pallet", the color pallet will appear:



'Web-safe' 216 colors are displayed in the pallet. Clicking a color you want to set, the color's hexa-decimal expression will be then be inserted at the cursor position in the editing window. At the top of the pallet, the 6 most recently selected colors are displayed for easy reference.

12.3 Bring All to Front

The window menu "Bring All to Front" brings all JChecker windows to the front.

This menu is valid only under Mac OS X. In Classic OS, all JChecker windows can be brought to front by clicking the JChecker window.

12.4 Switch

When selecting "Switch," the back most window will be switched to the active front most window.

12.5 Tile

When you choose "Tile," all open windows of JChecker will be tiled horizontally.

When you choose "Tile(Vertical)," all open windows will be tiled vertically.

12.6 Zoom

The "Zoom" menu feature acts the same as when clicking a window's zoom box. This menu allows an active window's viewable contents to be enlarged, or can return the viewable contents to their original viewing size, having been previously enlarged by a Zoom command. NOTE: The zoom function does not affect the actual size of text or graphics and has no effect on the printing size.

Chapter 13 Macro

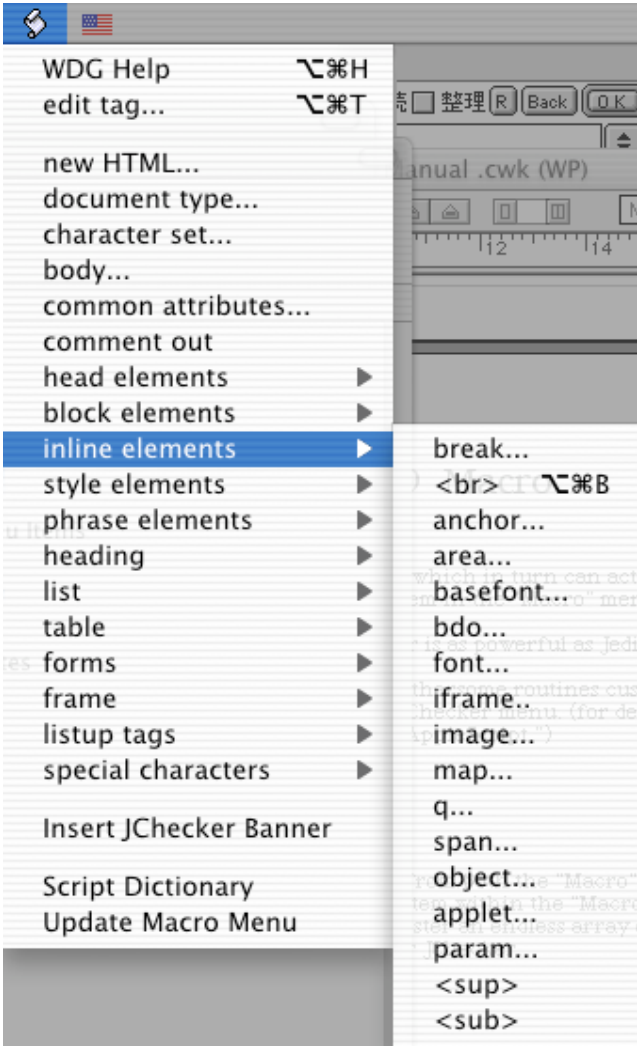
Macros allow you to use your own AppleScript programs which can be utilized as macros within the JChecker environment after registering them in the JChecker "Macro" menu.

The Macro/AppleScript functions under JChecker work as powerful as those of Jedit4.0.

By using a macro, you can take needed but bothersome routines, customizing them into a single key stroke and storing them for easy access and execution under the JChecker menu. (for details of JChecker's AppleScript commands, see the "Appendix 3. AppleScript.")

13.1 Macro Menu

You can execute your AppleScript programs from both the "Macro" menu and the "Macro Window (see below)." As each menu item within the "Macro" menu of JChecker can have sub-menus, you can create and register an endless array of keyboard short-cut macros, storing them in sub-menus under JChecker.



As more than 50 macros are already installed which cover HTML4.01 specification, you may not need to make new macros for the time being. For more information regarding these factory installed macros, see Appendix 1, Attached Macros.

To create your own macro, first write an AppleScript program using the "Script Editor" and save it as a compiled script file.

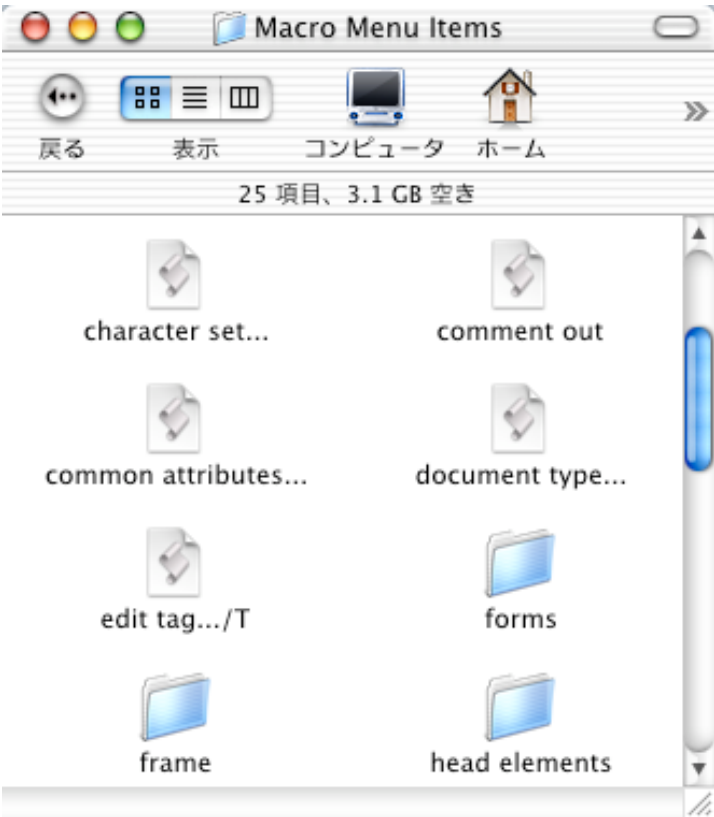


Next, place this file into the "Macro Menu Items" folder inside the "JChecker4.0" folder.

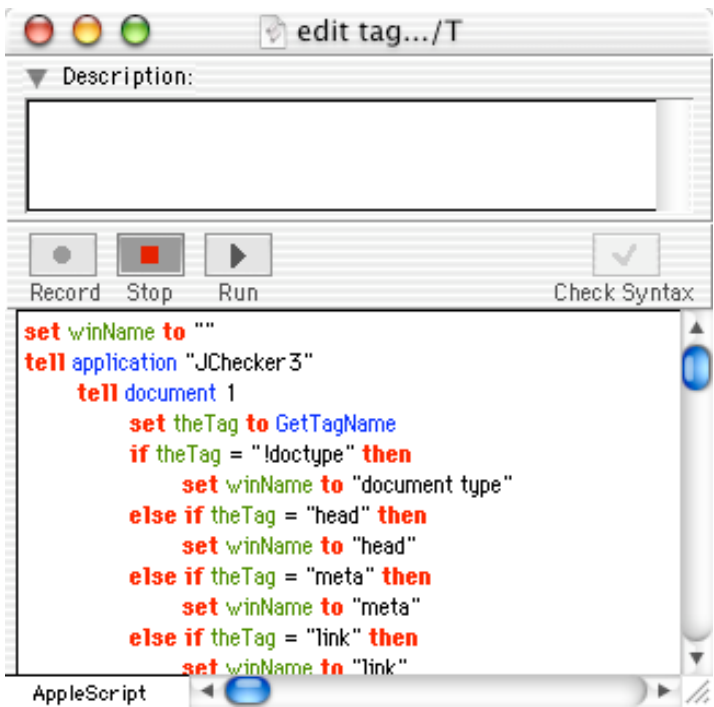


Now start-up JChecker. Your script will appear as a macro in the "Macro" menu and will appear in the "Macro Window."

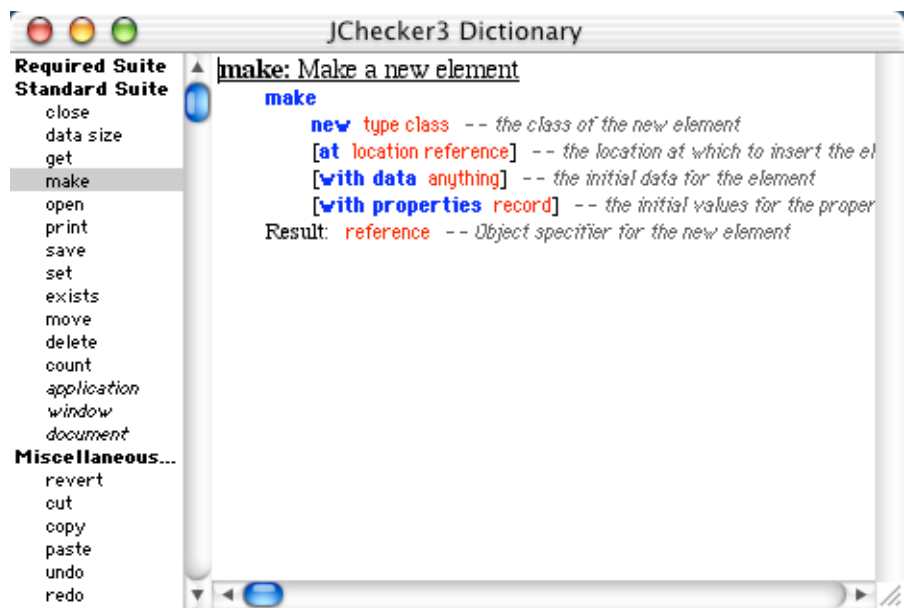
To make sub-menus, create a new 'child folder' within the "Macro Menu Items" folder and place your script files within. (The maximum level of this hierarchy is a depth of two. Thus you cannot make grandchild folders).



If you want to view or edit your macro script file, select it from the "Macro" menu with the control key down. The "Script Editor" will open your macro script file as shown below:



To see details of AppleScript commands supported by JChecker, select the "Script Dictionary" from the menu "Macro." The "JChecker Dictionary" window will appear as shown below. This information may be useful if you wish to make an AppleScript program which controls JChecker.



Rebuilding Macro Menus

Even after you add or delete a script file in the "Macro Menu Items" folder, the contents of the "Macro" menu will not be changed until you have rebuilt the "Macro" menu. To do this, select the "Update Macro Menu" under the menu "Macro." If JChecker is not running when you change contents of the "Macro Menu Items" folder, the "Macro" menu will be updated automatically when you restart JChecker.

Macro Shortcuts

To assign a shortcut key to a certain macro, add a "/" followed by some capital letter that you wish to use as a shortcut key at the end of name of the file's script. For example, you can find a script file named "edit tag.../T" in the "Macro Menu Items" folder. You can run this macro by typing a key "T" while pressing the option and command keys down simultaneously.

Separators on Macro Menus

To add a line separator in the "Macro" menu, create a dummy empty script file with a name that begins with the "-" character. For example, the following file which contains no script serves as a separator file that will allow a line separator to be made in the "Macro" menu.

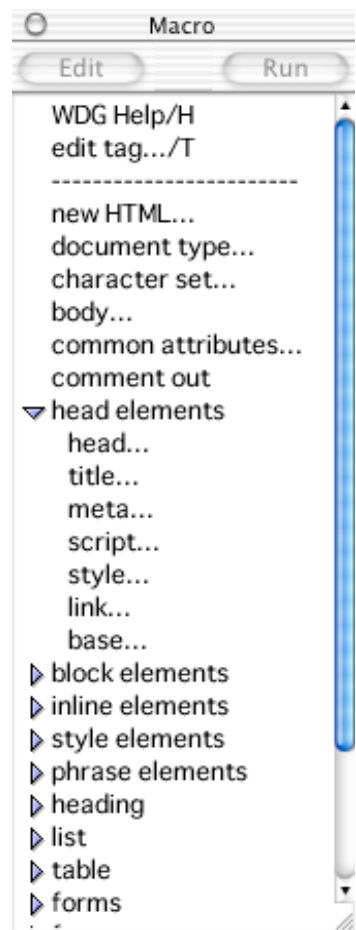


You can change the position of a separator in the menu by the use of the "Macro Window." See further explanation in 13.2.

13.2 Macro Window

From the "Macro Window," not only can you execute macros in the menu "Macro," but can also open macros by using the "Script Editor," thus having the ability to edit them. You can also move macro positioning within the "Macro" menu list by utilizing this window. Removal and Addition of macros can also be done from this window by "Drag and Drop."

To open the "Macro Window," select "Macro Window" from the "Window" menu. A window will appear:



To run a macro, select the macro and double click it. Selected macros can be executed by pressing the "Run" button.

To edit a macro using the "Script Editor," select the name of a macro you desire and double click it while pressing the control key. You can also acces macros via the "Script Editor" accessed by clicking the "Edit" button.

A small triangle to the left side of a particular macro listed denotes that the specific macro has sub 'child' macros. To show or hide these child macros, click the displayed triangle. To show or hide all child macros in the window at simultaneously, click on any one of these triangles with the while depressing the option key.

Moving Macros in the "Macro Window"

To change the order of macros listed in the "Macro Window," simply drag the desired macro item to another position on the list. Thus the order of listed items in the "Macro" menu will reflect such change in the "Macro Window."

Dragging and Dropping Script Files

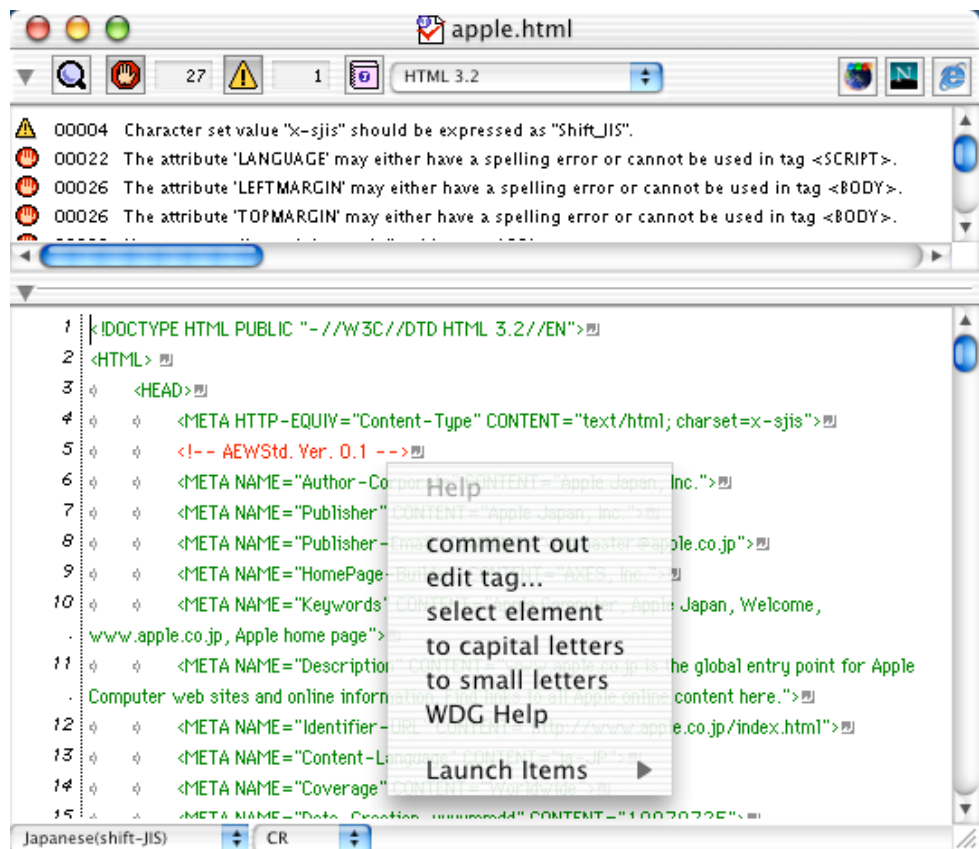
To add a new menu items to the "Macro" menu, drop a created script file onto the "Macro Window." The script file will be automatically moved to the "Macro Menu Items" folder and a new menu item will appear in the "Macro" menu. To delete a script file from the "Macro Menu Items" folder, simply drag the macro item you wish to discard from the "Macro Window" to the "Trash."

File "MenuOrder.dat"

JChecker automatically creates files which contain the name "MenuOrder.dat" in the "Macro Menu Items" folder and in each sub folder. These files contain data keeping track of the order of macros contained in the "Macro" menu.

13.3 Contextual Menus

To open the contextual menu, click a selected area in your JChecker document with the control key down.



You can add or remove macro items from the contextual menu in the same way as you do in the "Macro" menu.

To add a new contextual menu item, first write an AppleScript program using the "Script Editor." Save it as a compiled script file. Next, place this file in the "Context Menu Items" folder within the "JChecker3.0" folder.



Now start up JChecker. A new menu item will then be found in the contextual menu. If JChecker is already running, use the "Update Macro Menu" located within the "Macro" menu.

Note that the contextual menu cannot have added sub-menus or separators, nor can changes in the order of items be changed.

Appendix 1 Attached Macro

More than 50 macro programs are bundled in the folder "Macro Menu Items" located in the "JChecker3.0 folder".



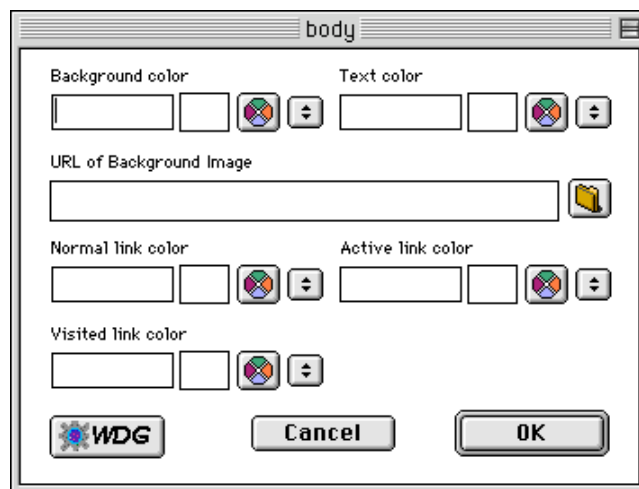
These macro programs cover HTML4.01 specification and support powerful functions like:

- (1) Analysis of tags at the cursor position and their modifications
- (2) Inserting a new tag at the cursor position
- (3) Showing help information for the tag at the cursor position

There are two types of the macro programs. One is a macro type which name ends with "..." characters. The other is a type that doesn't end with "...".

Macros without "..." usually are of a simple macro category that do not possess dialog windows.

When selecting a macro which name ends with "...", a dialog window appears:



These dialog windows are generated by the AppleScript application "HTMLPlugin" which is invoked by each macro program.

What is "HTMLPlugin"?

HTMLPlugin is an AppleScript application which was developed with the AppleScript development tool "FaceSpan3.0" (Digital Technology International Inc). The program file of HTMLPlugin is stored in the folder "Macro Menu Items".



If you use "FaceSpan3.0", you can change the layout of dialogs, read or modify the source code of HTMLPlugins. For more, visit the following web site.

<http://www.faceSpan.com/>

Upon using an HTMLPlugin for the first time, a macro program prompts you for the "HTMLPlugin" location. You will need to specify the HTMLPlugin as being inside the folder "Macro Menu Items".

If you check the check box "Invoke HTMLPlugin" at start-up" in the Preferences "Editor", the HTMLPlugin will be invoked automatically upon the JChecker3.0 start-up. This will allow shorter start-up times notable of each macro.

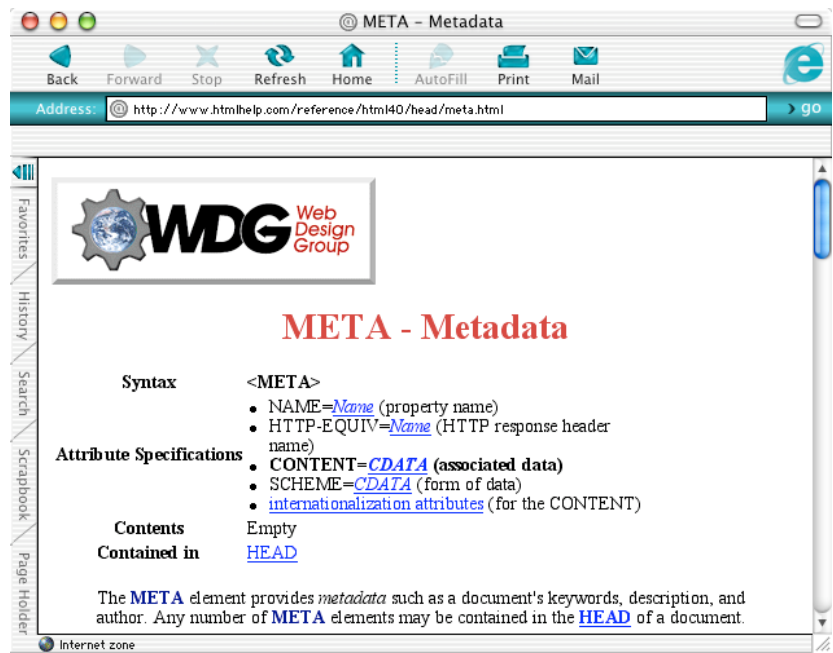
Note: as of March 2002, FaceSpan does not yet support Mac OS X native mode. Upon invoking HTMLPlugins under Mac OS X, they will actually run under the Classic mode. FaceSpan is currently developing FaceSpan4.0, which supports Mac OS X. Artman21 will develop and offer the Mac OS X native HTMLPlugin as soon as the anticipated FaceSpan4.0

is released.

WDG Help

After placing the cursor within a questionable tag, select the macro menu "WDG Help". Associated help information for the troublesome tag will be displayed in the web browser window. These are on-line help pages, and have been issued by the WDG (Web Design Group).

As example, place the cursor within the META tag selecting "WDG Help" from the macro menu. Your browser will automatically take you to a help page for the META element.



As WDG Help is an on-line help service requiring the Internet, you machine thus needs to be connected to Internet.

WDG Help files and help room may also be accessed from the contextual menu:

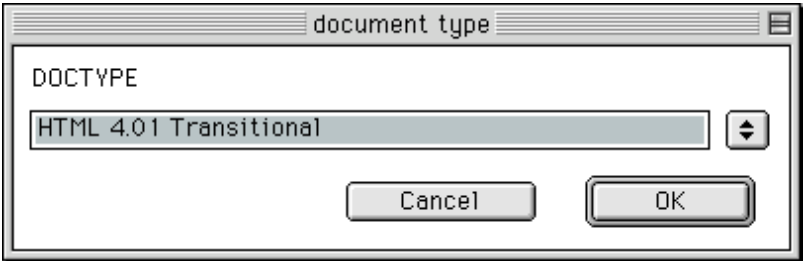


Macro "edit tag..."

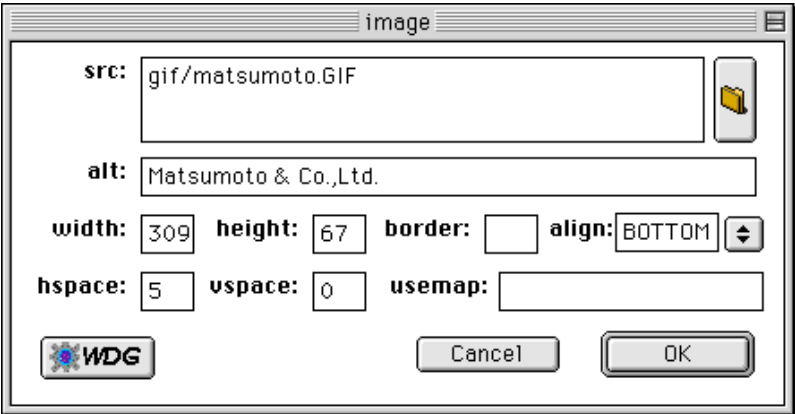
The macro "edit tag..." will most likely become the most useful and powerful macro for the user.

To use the "edit tag", first set the cursor inside the tag you want to edit, then select the macro menu "edit tag...". The macro analyzes the element of the tag and will display an appropriate editing dialog for this tag.

For example, when the cursor is in the DOCTYPE declaration and the macro "edit tag..." is selected, the following editing dialog for DOCTYPE will appear:



When the cursor is within the IMG tag, the following dialog will appear respectively.



The macro "edit tag..." supports the following elements:

doctype head, meta, link, title, base, style, script, body, a, img, br, font, div, p, h1~h6, heading, hr, blockquote, ins, del, table, caption, col, colgroup, thead, tbody, tfoot, tr, td, th, form, legend, button, input, label, select, optgroup, option, textarea, applet, object, param, basefont, bdo, iframe, map, area, q, span, frame, frameset

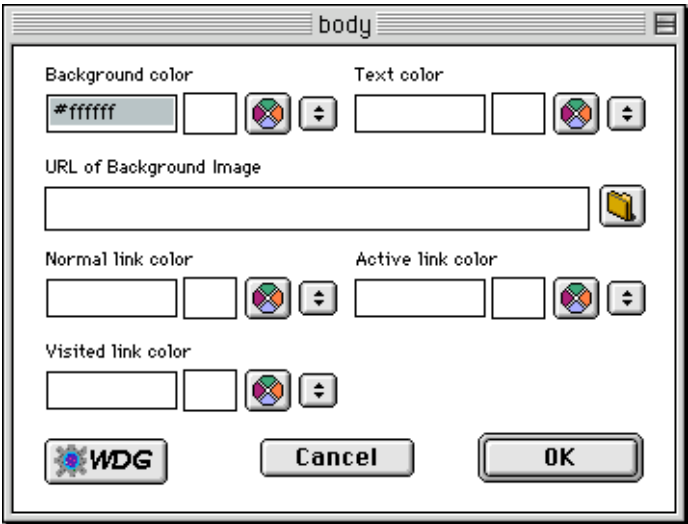
You can also invoke the "edit tag..." macro from the contextual menu:



Usage for the Editing Dialog

There are more than 50 kinds of editing dialogs. Though so many, their usage have much in common. For example consider the usage of the BODY editing dialog.

Setting the cursor in the BODY tag, select the macro "edit tag..." or "body..." The BODY tag editing dialog will appear.

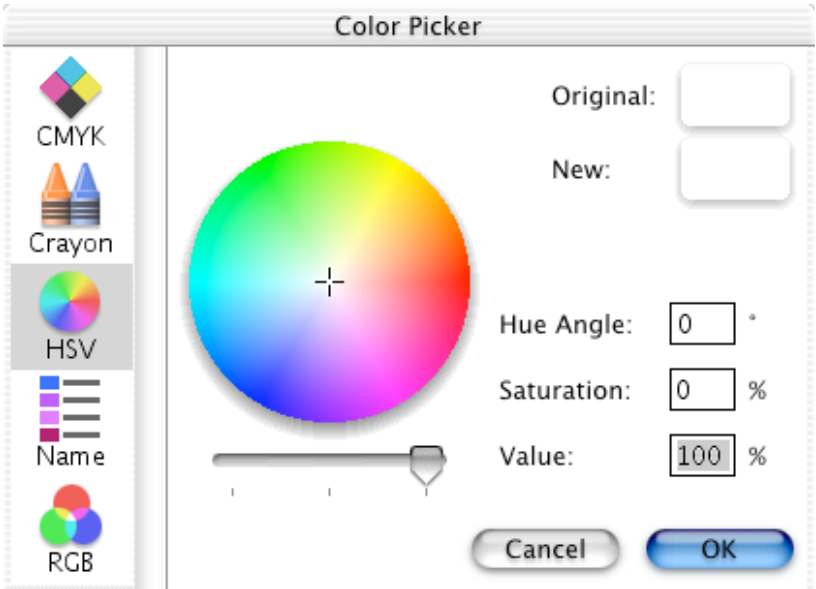


The macro program analyzes attributes of the tag and displays attribute values for each field in the dialog. If the attribute was not specified, its field will be left blank.

Change attribute values as desired and click on "OK". The tag in the editing window will now be replaced by this new updated one. If you specify a blank for a certain attribute field, the corresponding attribute name and value will be removed from the tag.



The button is a color pallet button. Clicking the button, the color pallet as follows will appear.





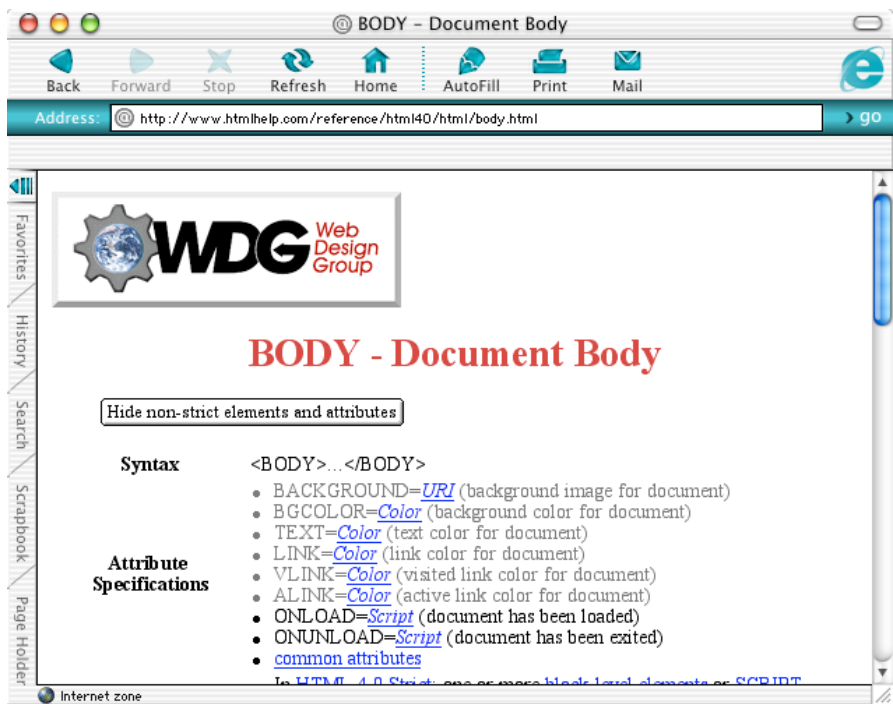
The button is a file browser button used to select the URL of a file or folder. Clicking the button, the file dialog will appear:



Select the desired file and click "Open". The relative URL from the editing file to the selected file will be set in the field. If the editing file has not been specified, the absolute URL of the selected file starting with "file://" will thus be set.



The button is the WDG Help button mentioned previously. Clicking the button, the on-line help page for the dialog will be displayed in the desired web browser. If you click the button while in the BODY tag editing dialog, help concerning the BODY element will be displayed:



In general, such dialog conveniently turns into a tag editing mode when the current cursor is inside the tag and similarly turns into the tag inserting mode when the cursor is outside the tag.

By way of exception, dialogs for the macro "doctype...", "character set..." and "body..." always act in the tag editing mode regardless of the cursor position. The macro programs for these dialogs will search the tag when the cursor is not inside the tag by default.

Appendix 2 Error and Warning Messages

This appendix explains error messages and warning messages provided by the JChecker validation and link check.

Wrong expressions which may clearly disrupt the rendering of the HTML will be notified by  error messages. Likewise, expressions which may have problems depending on the browser or that are not recommended will be notified by  warning messages.



No.1

An empty space exists between the "<" mark and the element name. Use the expression "<" for "<" if this is not a tag.

This error message is displayed when a space, tab or return code is included between the "<" mark(which normally represents the beginning of a tag) and the element name. Internet Explorer interprets this as a tag regardless of these spaces. However, Netscape Navigator will interpret this as simple data instead of the beginning of a tag. If the "<" mark is indeed simply a part of the data, please use the expression "<".



No.2

The element name "XXXXXX" does not exist.

This error message is displayed when a non-existing element name is specified within the tag. Please make sure that the element name is spelled correctly. All available element names are registered in the tag dictionary. If you wish to use an original element, please be sure to register that name in the tag dictionary.



No.3

A "<" mark exists within the tag. Use the expression "<" for a "<" intended as data.

This error message is displayed when a "<" mark is used twice before a ">" mark in a tag. Please use the expression "<" if you wish to express the "<" mark **as part** of the data.



No.4

Start tag <xxxx> was not found in the same hierarchy as the end tag </xxxx>.

This error message is displayed when an element is lacking a start tag or when it is not on the same hierarchical level. (Probable causes include a damaged hierarchy structure.)

This error message may be displayed even though a corresponding start tag seems to be in place. In most cases, some tag disrupting the hierarchical structure may exist between the start tag and the end tag.

For example,

```
<HTML><HEAD>
<B>
<TITLE>my title</TITLE>
</B>
</HEAD>
<BODY BGCOLOR="#FFFFFF">
my body messages
</BODY>
</HTML>
```

If you check the validation of the above HTML, the error message ("The corresponding start tags <HEAD> and <HTML> cannot be found") will be displayed between the fifth and ninth row. Why does this error message appear when both <HTML> and <HEAD> are in place?

The cause of this error is the tag located in the second row. The element cannot be either a child element of <HTML> or <HEAD.> Both <HEAD> and <HTML> are considered to be concluded when is inserted. Thus, the rows will be interpreted as shown below with end tags </HEAD> and </HTML.> The "abbreviated" end tags (in this case, </HEAD> and </HTML>) will be flagged by warning messages..

```
<HTML>
  <HEAD>
    </HEAD>    <----abbreviated end tag
  </HTML>      <----abbreviated end tag
  <B>
    <TITLE>my title</TITLE>
  </B>
</HEAD>
<BODY BGCOLOR="#FFFFFF"> my body messages </BODY>
</HTML>
```

Because these "abbreviated" end tags </HTML> and </HEAD> are located in the third and forth row, the intended end tags </HTML> and </HEAD> located in the eighth and tenth row become errors.

Please check the hierarchical structure using the "Beautify" menu and look for "abbreviated" end tag warnings when this error message is displayed and a beginning tag seems to be in its proper place.



No.5

The end tag </xxxx> is assumed to be omitted before tag <yyyy.> because the element <xxxx> cannot contain tag <yyyy.> as its child element.

Some elements (xxxx) may omit an end tag. If a beginning tag of another element (yyyy), which cannot be included in (xxxx), is found, (xxxx) will be interpreted as lacking an end tag (/xxxx). This warning message is displayed in such a situation. Child elements which can be included in a specific element and elements which can omit their end tag, may be checked by choosing the "Tag Dictionary..." menu.

As seen in the example below, sometimes when this message appears, it will also generate the "Start tag is not found" error" later in the report.

<yyyy> can be child of <xxxx>	<yyyy> can't be child of <xxxx>
<pre><xxxx> <yyyy> </yyyy> </xxx></pre>	<pre><xxxx> </xxxx> <-- omitted end tag <yyyy> </yyyy> </xxx> <-- star tag not found error</pre>
No warning, no error	Warning and error are displayed



No.6

A "<" mark corresponding to the ">" mark was not found. If the ">" mark is data, please use the expression ">".

This error message is displayed when a ">" mark is found in the data . If the ">" mark is a part of a data line, you should use the expression ">".



No.7

End tag </xxxx> is not found at the same hierarchy as the start tag <xxxx>.

This error message is displayed when an end tag is not found and that end tag is necessary for the element. You can determine whether an end tag is necessary for the specified element or not by selecting the "Tag Dictionary ..." menu.



No.8

xxxx cannot be a child element of yyyy.

This error message is displayed when a particular element (xxxx) cannot be included as child in element "yyyy." The various elements which may be included in "yyyy" as child elements may be checked by choosing the "Tag Dictionary ..." menu.



No.9

The contents between tags <xxxx> and </xxxx> is empty.

This warning message is displayed when nothing is found between the start tag and the end tag. In some cases this may be intentional, eg. when there are empty fields in the table. However, if this is not the case, please delete the start and end tags.



No.10

This attribute value needs to be enclosed by quotation (" ") marks.

This error message is displayed when the attribute value is not enclosed by quotation (" ") marks. However, there is no need to enclose with quotation marks if it is less than 72 characters and only contains alphabets, numbers, hyphens (-) and periods (.) .



No.11

Attribute zzzz may either have a spelling error or cannot be used in tag <xxxx>.

This error message is displayed when an ineligible attribute "zzzz" is specified for element "xxxx." The eligible attributes for a particular element can be checked by choosing the "Tag Dictionary ..." menu.



No.12

Necessary attribute "zzzz" is not defined in tag <xxxx>.

This error message is displayed when a necessary attribute "zzzz" is not specified in element <xxxx>. For example, a "src" attribute needs to be included in a tag. Necessary attributes can be checked by choosing the "Tag Dictionary ..." menu.



No.13

An error was found in the expression of a character reference. Use "&" to express the "&" mark.

Whenever the "&" mark is used to specify data or an attribute value, it represents the beginning of a character reference. If you want to use the "&" mark as the character itself, then, just as the "<" mark needs to be replaced by "<" the "&" mark needs to be replaced by the expression "&".



No.14

This attribute value is missing the ending quotation mark (").

This error message is displayed when an attribute value is missing the ending quotation mark (") while the beginning quotation mark is found.



No.15

An <HTML> tag should be inserted at the beginning of the text.

HTML text should begin with a <!DOCTYPE> declaration and a <HTML> tag. This warning tag is displayed when a <HTML> tag is not found at the beginning. However, <!DOCTYPE> declarations and <HTML> tags are not absolutely necessary.



No.16

Element <yyyy> is not defined in proper order in xxxx.

The child elements of <HTML> (<HEAD>, <BODY>, <FRAMESET> and <PLAINSET>) and those of <TABLE> (<CAPTION>, <COL>, <COLGROUP>, <THEAD>, <TFOOT>, <TBODY> and <TR>) need to be specified in proper order. This error message is displayed when this order is not followed.



No.17

Tag <TITLE> should be defined within the HEAD element.

This warning message is displayed when a <TITLE> is not found in the HEAD element. The <TITLE> tag is used for the window title of browsers, etc. Please be sure to insert this tag.



No.18

Data cannot be directly written as an element of xxxx.

This error message is displayed when you attempt to specify data as an element of xxxx but xxxx does not allow it. For example, the <TD> and <TH> may be specified as contents of <TR>, but data itself may not be inserted directly into <TR>. The question of whether a particular element may specify data or not can be checked by choosing the "Tag Dictionary ..." menu.



No.19

Only one byte characters may be specified for the value of attribute ZZZZ in the element XXXXX.

This error message is displayed when an attribute value of two or more bytes is specified for an attribute which can only contain one byte values.



No.20

An attribute value needs to be specified for attribute ZZZZ.

This error message is displayed when an attribute ZZZZ does not have a specified value.



No.21

Specify one of aaalbbblccc for the value of attribute ZZZZZ.

Some attributes (eg. align) may only specify one of a limited number of strings for its value. This error message is displayed when an ineligible string is specified for the attribute value. The eligible strings for such attributes can be checked by choosing the "Dictionary" menu."



No.22

The URL value "VVVV" of the attribute ZZZZ contains a space or backslash. These characters are not allowed in an URL value.

This error message is displayed when a space or a back slash is found within the URL value of an attribute.



No.23

The URL value "VVVV" of the attribute ZZZZ contains a control code or a high-ASCII character.

This error message is displayed when a control code or a high-ASCII character is found within the URL value of an attribute.



No.24

In the URL value of attribute 'ZZZZ', it should be expressed %99.

It is best for some symbols (eg. "~") in an URL to be expressed as a hexadecimal number with a "%" mark.



No.25

The value ("VVVV") of the attribute charset does not match with the character code set of the this file.

This warning message will be displayed when the character set used in the file does not match with the value of the attribute charset.



No.26

Character set value "AAAA" should be expressed as "BBBB".

This warning message is displayed when the attribute charset specifies "x-sjis" or "x-euc-jp" is used. Please use "Shift-JIS" or "EUC-JP". Neither "x-sjis" nor "x-euc-jp" are official names.



No.27

One-byte Katakana may not be displayed properly.

The use of one-byte Katakana is prohibited in Internet communication. Therefore it is best not to use these expressions.



No.28

Tag <HTML> should be inserted after the <!DOCTYPE> declaration.

This warning message is displayed when an <HTML> tag is not found after a <!DOCTYPE> declaration. However, lacking a <HTML> tag will not be flagged as an error.



No.29

Color "VVVV" of the attribute ZZZZ is not included in the basic 16 colors of the HTML specification.

This warning message is displayed when the specified color is not included in the basic 16 colors. ("aqua", "black", "blue", "fuchsia", "gray", "green", "lime", "maroon", "navy", "olive", "purple", "red", "silver", "teal", "white", "yellow") However, this is not flagged as an error because Netscape Navigator and Internet Explorer can both specify 140 color rows.



No.30

These characters may vary their meanings depending on the computer or font. It is best not to use these characters."

In shift-JIS character set, the codes from 0x8450 to 0x889E and from 0xEB40 to 0xEFFC vary their meanings depending on the computer and font. These characters are called: "machine dependent characters." Don't use these characters for Internet communication.



No.31

The Anchor "ZZZZ" can't be found.

The anchor , which is referred to by the tag , can't be found. The anchor tag may not be properly defined.



No.32

The Map "MMMM" can't be found.

The map, which is referred to by the attribute USEMAP="#MMMM", can't be found. The map tag <MAP NAME="MMMM" > may not be properly defined.



No.33

The relative path "UUUU" can't be analyzed because the base URL is not specified. Save the HTML file first, then check it again.

To analyze the relative path, the base URL should be specified in advance. The base URL will be specified automatically when you save the HTML file or it can also be specified explicitly by the <BASE> tag.



No.34

Can't find the local document "DDDD".

The file "DDDD", which is specified in the attribute "ZZZZ", can't be found in your hard disk.



No.35

Can't connect to the host of "HHHH".

The HTTP access to the host computer of "HHHH", which is specified in the attribute "ZZZZ",

failed for an unknown reason. Please check your internet setting.



No.36

Sorry, JChecker can't check this URL "UUUU".

JChecker can check the links of local files and http: protocol. But it does not support checking protocol such as mailto: and ftp:.



No.37

Can't find the remote document "DDDD". (Response Code)"

The remote document "DDDD" does not exist. Response Code 400s represent client side errors and 500s represent server side errors.



No.38

This document has been moved permanently to "BBBB" from "AAAA".

The page has been moved permanently. You should modify this URL from "AAAA" to "BBBB".



No.39

This document has been moved temporarily to "BBBB" from "AAAA".

The page has been moved temporarily.



No.40

Null code was detected. It should be deleted.

Null Code (0x00) was detected. Null code will cause a misjudgment of the auto detection of the character set. It should be deleted.



No.41

Error in DOCTYPE declaration. DTD can't be determined.

JChecker can't determine the dictionary when the DOCTYPE declaration has some format error. For example, the right DOCTYPE declaration of HTML 3.2 is as follows.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">
```



No.42

The dictionary that the DOCTYPE requests is not contained in JChecker. The default dictionary will be used as an alternate.

JChecker dictionary menus doesn't have the dictionary specified in the DOCTYPE declaration. Check the spelling of DTD. Also you can make new dictionary of the DTD by using the DtdToDic convertor.



No.43

Can't find the valid DOCTYPE declaration. "

You should specify the DOCTYPE declaration at the top of your HTML so that HTML client applications can identify the DTD that your document is based on.



No.44

The value of this attribute ZZZZ is the fixed value "VVVV".

You can't assign a value for the attribute ZZZZ other than the fixed value "VVVV".



No.45

Can't omit the attribute value in XML. "

In HTML you can omit the attribute name when it is a boolean attribute. But in XML you must always specify both the name and a value like ZZZZ="ZZZZ".



No.46

This attribute ZZZZ is boolean. The value must match in name: ZZZZ="ZZZZ". Or you can omit the name and only specify the value ZZZZ.

This attribute ZZZZ is boolean. The value must match in name: ZZZZ="ZZZZ". Or you can omit the name and only specify the value ZZZZ.



No.47

Can't assign a value other than a number.

The value of the attribute ZZZZ of the element <XXXX> should be a number string which contains only number (0-9) characters.



No.48

The value of the attribute ZZZZ of <XXXX> should be the name string which starts with an alphabet character.

The name string starts with an alphabet and only contains alphabets, numbers, hyphens, underscores, colons and periods.



No.49

The value of the attribute ZZZZ of <XXXX> should be the NMTOKEN.

NMTOKEN is the string that only contains alphabets, numbers, hyphens, underscores, colons and periods.



No.50

The ID of the attribute ZZZZ of <XXXX> is already defined.

The value of the attribute ZZZZ of <XXXX> should be ID which is unique throughout the document.



No.51

This character set name for the attribute ZZZZ is not registered in IANA."

You should use the character set name which is registered in IANA (Internet Assigned Numbers Authority). The list of the registered character sets can be seen at the following web site.

<http://www.iana.org/assignments/character-sets>



No.52

The value of attribute ZZZZ should be a number (*) or a % value."

The type of this attribute value is MultiLength which consists of number, % value, number*, or *.



No.53

The value of attribute ZZZZ should be a list of numbers(*) or % values separated by a comma.

The value of the attribute ZZZZ should be a list of comma-separated MultiLength values which consists of numbers(*) or % values.



No.54

You can't assign an end tag for the element <XXXX> .

You can't assign an end tag for the element <XXXX> .



No.55

The value of attribute ZZZZ should be an integer value or an integer value with %.

The type of this attribute value is 'Length' which consists of integer value or integer value with %.



No.56

The value of attribute ZZZZ should be a list of numbers or % values separated by a comma.

The value of the attribute ZZZZ should be a list of comma-separated 'length' values which consists of numbers or % values.



No.57

The color value "VVVV" of the attribute is not defined.

To specify an undefined color, use RGB value. For example, The RGB value for the red color is "#FF0000".



No.58

You can't omit the end tag because the element <XXXX> is not an empty element.

In XHTML, you can omit a end tag for the empty element by assigning the start tag <element name/>. But you can't omit a end tag for non empty elements.



No.59

In XHTML, it is recommended to specify the XML declaration <?xml version="1.0" ?> at the top of the document.

When the character encoding is UTF-8 or UTF-16, you can omit the XML declaration <?xml version="1.0" ?>. But it is recommended not to omit the declaration.



No.60

The XML declaration <?xml version="1.0" encoding="your encoding" ?> is needed because the encoding is not Unicode.

As the encoding is not UTF-8/UTF-16, you must specify the XML declaration <?xml version="1.0" encoding="your encoding" ?> at the top of the document.



No.61

Can't find the end tag "]]> " of the CDATA section.

In XHTML, the CDATA section begins with "<![CDATA[" and ends with "]]>".



No.62

Couldn't connect to the remote document "DDDD" because of a time out error.

Failed to connect within the time limit assigned within Link Preferences. Even if the URL is correct, sometimes the connection may fail when the sever or line is busy. Please check it again.



No.63

Error occurred in URL Access Manager.

Failed to check linkage due to an unkown error in the URL Access Manager. This error sometimes occurs when the URL string is very long.



No.64

Element <HEAD> should be defined just after the HTML tag.

This warning message is displayed when a <HEAD> and <TITLE> is not found. The <TITLE> is used for the window title of browsers, etc. Please be sure to insert these elements.



No.65

This attribute ZZZZ is not defined in DTD and may be a special attribute inserted by some HTML editor for editing convenience.

Some HTML editors, such as PageMill or GoLive, sometimes insert original attributes for its editing convenience. These attributes are not defined in DTD.



No.66

This part in the DOCTYPE declaration should be capital.

The DOCTYPE declaration is case-sensitive.



No.67

Invalid scheme in URL "UUUU" of the attribute ZZZZ

The URL scheme should be one of http: https: ftp: gopher: nntp: telnet: wais: mailto: news: or file: .



No.68

The hierarchy structure is broken inside <html> or the end tag </html> not found

You can omit the end tag </html>. But it is not recommended.



No.69

The hierarchy structure is broken inside <body> or the end tag </body> not found.

You can omit the end tag </body>. But it is not recommended.



No.70

Japanese character set '%2' is specified in the attribute '%1' while the document does not contain any Japanese 2 byte characters,

If your document is not Japanese document, specify 'ISO-8859-1' or other non Japanese character set.



No.71

Wrong xml version number in the XML declaration.

Specify "1.0" for the xml version number in both cases of XHTML 1.0 and XHTML 1.1.



No.72

Nothing is specified in URL.

This is not an error but if you don't need to specify URL, delete that meaningless attribute.



No.73

Same attributes zzzz are specified redundantly in a tag xxxx.

Please remove one of the attribute zzzz from the tag xxxx.



No.74

It is recommended for XHTML empty element xxxx to end with '</>' rather than add an end tag </xxxx>.

It may be simpler and more readable to enclose XHTML empty tag with ' <xxxx />' rather than add an end tag </xxxx>.



No.75

The element base should be the first element that refers to an external source in <head> section.

The base element must appear in the head section of an HTML document, before any element that refers to an external source.



No.76

The URL reference of element base should be an absolute URL.

This attribute of element base must specify an absolute URI that acts as the base URI for resolving relative URIs.

Appendix 3 AppleScript

As JChecker3.0 is an AppleScript "scriptable*" application, you can control JChecker3.0 from AppleScript programs that you create using the "Script Editor." You can as well register script programs into a 'customized menu' within the JChecker3.0 "Macro" menu. You can use these created script programs to control other AppleScript scriptable applications.

Using AppleScript commands supported by JChecker, you can launch and quit JChecker, open, close and save JChecker documents, and also edit text within. The AppleScript of JChecker is as powerful as Jedit4.0 and supports many original commands that handles elements and attributes of the HTML documents.

The purpose of this chapter is mainly to explain the specifications of JChecker's AppleScript, not AppleScript in general. If you are not familiar with the fundamentals of AppleScript, please consult an AppleScript guide book or look for useful information within the following web sites.

<http://www.apple.com/applescript/> (in English)

<http://www.asep.apple.co.jp/Techdoc/AppleScript/> (in Japanese)

* You can execute most text edit operations by using JChecker's commands under AppleScript. However, JChecker is not an AppleScript recordable application and there are some operations that will not be supported by AppleScript.

Compatibility with Jedit

As JChecker's AppleScript code was implanted from the editor "Jedit", it is basically compatible with Jedit's AppleScript.

Note: the following are the JChecker original commands that handle HTML documents.

check, linkCheck, correct, checkAll, minimize, beautify, recolor, GetTagName, GetAtrbValue, SelectTag, SelectAtrb, GetColor, GetRelativePath, SetAtrb, HTMLColotToRGB, GetDtdName

Also note: the following objects and commands are not supported in JChecker because these are meaningless in HTML editing.

Unsupported Objects

style run, picture, sound, aliasFile, movie

Unsupported Commands

play, speak, remove break, detab, entab, convert kanji space to ascii, toYomi , kanjiToKana, toKatakana, toHiragana, toZenkaku, toHankaku, convert kana to zenkaku, convert ascii to hankaku, remove tailing space, remove null, jsort, uniq, reverse, toUnicode, fromUnicode, OpenDict, OpenChecker, clear undo,

Objects (Classes)

The so-called "object(class)"s of JChecker3.0 controlled by AppleScript are "application", "window", "document", "byte", "character", "word", "line", "paragraph", and "selection".

"application" object

Commands Available:

run, launch, quit, get, set, exists, preDialog, postDialog, suspend, OpenURL, select, and updateMacroMenu.

Properties:

sound boolean

If true, notification sounds will follow the checking results.

autocheck boolean

If true, HTML checking will initiate upon opening a window.

dictionary international text

Default dictionary name used for HTML Checking.

encoding international text

Default character encoding.

lineBreak auto/mac/mac/unix/dos

Default line break type.

allDTDs a list of international text [r/o]

List of dictionaries registered in the dictionaries menu.

allCharsets a list of international text [r/o]

List of character sets that JChecker supports.

batchMenu international text [r/o]

Batch menu name of word service clients.

appLoc alias [r/o]

JChecker application's location

batchIcon 'SICN' [r/o]

Batch menu icon for word service clients.

"window" object

Commands Available:

open, close, count, delete, exists, get, make, move, select, set

Properties:

bounds bounding rectangle

Rectangular coordinates of the window.

closeable boolean [r/o]

Does the window have a close box?

titled boolean [r/o]

Does the window have a title bar?

index integer

The index number of a window

floating boolean [r/o]

Does the window float?

modal boolean [r/o]

Is the window modal?

resizable boolean [r/o]

Is the window resizable?

zoomable boolean [r/o]

Is the window zoomable?

zoomed boolean

Is the window zoomed?

name international text

The title of the window

visible boolean [r/o]

Is the window visible?

position point [r/o]

The upper left coordinates of the window

"document" object

Commands Available:

make, open, close, print, save, revert, close, data size, get, set,insert, find, replace, mark, exist, cut, copy, paste, select,undo, redo, preDialog, postDialog, suspend, check, linkCheck, correct, checkAll, minimize, beautify, recolor, GetTagName, GetAtrbValue, SelectTag, SelectAtrb, GetColor, GetRelativePath, SetAtrb, HTMLColotToRGB, GetDtdName

Properties:

name international text
the title of the document
modified boolean [r/o]
Has the document been modified since the last save?
errorCount integer [r/o]
error count for an html document
warningCount integer [r/o]
warning count of the html document
errorInfo international text [r/o]
error and warning messages of the checked document
dictionary international text
doctype name
encoding international text
character set name
lineBreak auto/mac/unix/dos
line break type
fileSpec file specification [r/o]
The file specification of the document
specified boolean [r/o]
Has the document file name been specified?
modified boolean [r/o]
Has the document been modified since last saved?

"byte" object

A "byte" object is an object whose length or position is expressed in byte(s). In English text, the byte object and the character object are the same, but in the languages of 2-bytes characters such as Japanese, byte objects and character objects differences in the way they are treated. 2-bytes characters, such as kanji, are counted as one character.

Commands Available:

print, data size, get, set, insert, find, replace, mark, count, cut, copy, select, change case

Properties:

length integer [r/o]
The length of a byte object (in bytes)
byteoffset integer [r/o]
The offset of a byte object from the beginning of the document (the first byte has offset 1)
line number integer [r/o]
The line number of a byte object from the beginning of the document (the first line has number 1)
paragraph number integer [r/o]

The Paragraph number of a byte object from the beginning of the document (the first paragraph has number 1)

"character" object

A "character" object represents one letter. One-byte characters such as ASCII and 2-bytes characters such as kanji are each counted as one character.

Commands Available:

print, data size, get, set, insert, find, replace, mark, count, cut, copy, select, change case

Properties:

length integer [r/o]

The length of a character object (in bytes)

byteoffset integer [r/o]

The offset of a character object from the beginning of the document (the first byte has offset 1)

line number integer [r/o]

The line number of a character object from the beginning of the document (the first line has number 1)

paragraph number integer [r/o]

The paragraph number of a character object from the beginning of the document.

"word" object

A "word" object is a string separated by white spaces (space codes, return codes and tab codes) and punctuation.

Commands Available:

print, data size, get, set, insert, find, replace, mark, count, cut, copy, select, change case

Properties:

length integer [r/o]

The length of a word object (in bytes)

byteoffset integer [r/o]

The offset of a word object from the beginning of the document (the first byte has offset 1)

line number integer [r/o]

The line number of a word object from the beginning of the document (the first line has number 1)

paragraph number integer [r/o]

The paragraph number of a word object from the beginning of the document (the first paragraph has number 1)

"line" object

A "line" object is a physical line displayed in the document window. If you change the document width from "Document Width" under the "Format" menu, the contents of the line objects will also be changed.

Commands Available:

print, data size, get, set, insert, find, replace, mark, count, cut, copy, select, change case

Properties:

length integer [r/o]

The length of a line object (in bytes)

byteoffset integer [r/o]

The offset of a line object from the beginning of the document (the first byte has offset 1)

paragraph number integer [r/o]

The paragraph number of a line object from the beginning of the document (the first paragraph has number 1)

"paragraph" object

A "paragraph" object is a string that is separated by return codes. Even if you change the line width from "Line width" in the Preferences "Editor", the contents of the paragraph objects will remain the same.

Commands Available:

print, data size, get, set, insert, find, replace, mark, count, cut, copy, select, change case

Properties:

- length** integer [r/o]
The length of a paragraph object (in bytes)
- byteoffset** integer [r/o]
The offset of a paragraph object from the beginning of the document (the first byte has offset 1)

"selection" object

A selection object is the highlighted area of the document. When an area is not selected, the cursor position within the 'selection' object will be represented by 0 bytes.

Commands Available:

print, data size, get, set, insert, find, replace, mark, count, cut, copy, select, change case

Properties:

- length** integer [r/o]
The length of a selection object (in bytes)
- byteoffset** integer [r/o]
The offset of a 'selection' object from the beginning of the document (the first byte has offset 1)
- line number** integer [r/o]
The line number of a 'selection' object from the beginning of the document (the first line has number 1)
- paragraph number** integer [r/o]
The paragraph number of a 'selection' object from the beginning of the document (the first paragraph has number 1)

Commands (Events)

"run" command

run *variable or Reference*

Opens an application file in the same way as double clicking it on the Finder Desktop.

```
run application "JChecker3"
```

Each time the "run" command is executed, the OpenApplication event invoke the target application to open a new window.

"launch" command

launch *variable or reference*

Launches an application file in the same way as double clicking it on the Finder Desktop.

```
launch application "JChecker3"
```

Unlike the "run" command, the "launch" command will not open a new window when the application is already running.

"make" command

make

new *type class* -- the class of a new element.

Result: *reference* -- reference to the new element.

Opens a new document or window. Not usable for objects other than window and document objects.

sample:

```
make new document
```

```
make new window
```

"open" command

open *reference* -- the object(s) to open

The "open" command opens JChecker3.0 document files.

The following script will open the file "index.html" in the "mySite" folder under "Macintosh HD":

```
tell application "JChecker3"
    open file "Macintosh HD:mySite:index.html"
end tell
```

"print" command

open *alias* -- list of objects to open

The "print" command will print JChecker3.0 documents in the same way as when you select "Print One" from the "file" menu. The "print" command can also prints objects such as file, selection, paragraph, line and so on.

print the file "index.html"

```
tell application "JChecker3"
    print file "Macintosh HD:mySite:index.html"
end tell
```

prints lines from 3 to 5 of the selection of document 1

```
tell application "JChecker3"
    tell document 1
        print lines 3 thru 5 of selection
    end tell
end tell
```

"save" command

save *reference* -- Objects to save.

[**in** *alias*]

The file in which to save the object(s).

The "save" command will save a specified object in the reference.

Using the parameter "in", you can specify a file where you want to save the target document much the same as when you select a file after you choose the "Save As" command from the "file" menu. Omit the parameter "in" to save the target document like when saving by choosing "Save" from the "file" menu.

The following script will save the entire contents of the current document into the file "HD:mySite:index.html" .

```
tell application "JChecker3"
    save document 1 in file "HD:mySite:index.html"
end tell
```

"revert" command

revert *reference* -- object to revert

The "revert" command reverts an active document to the original state in which you last saved it. You can use this command only for document objects.

```
tell application "JChecker3"
    revert document 1
end tell
```

"close" command

close *reference* -- *the objects to close*
 [**saving** yes/no/ask]
 Specifies whether or not changes should be saved before closing.
 [**in** *alias*]
 The file in which to save the object.

The "close" command will close documents or windows. You can specify how to save a document using the "saving" parameter. If you omit specifying this parameter, the document will be closed like when selecting "Close" under the "file" menu. In this case, if the document you close is not saved, the save dialog box will appear asking you to select a name and location for the file. You can also at this time assign an alias for the file using the "in" parameter.

The following script will save document "MyDoc" in the file "MyFile" and then close it.

```
tell application "JChecker3"
    close document "MyDoc" saving yes in file "MyFile"
end tell
```

"quit" command

quit

The "quit" command will quit JChecker3.0 in the same way as when quitting it by choosing the "Quit" from the "file" menu.

There are no parameters with this command.

"data size" command

data size *reference*

The object whose data size is to be returned

Result: *integer*

The size of the object in bytes

The "data size" command will return the byte size of the specified object. In the case of line or paragraph objects, the returned size will further contain the amount of line breaks existing.

The following example script will count the byte size of paragraph 3 of a current document.

```
tell application "JChecker3"
    data size of paragraph 3 of document 1
end tell
```

The same value will be returned if you replace "data size" in this script with "count bytes".

"get" command

get *reference*

The object whose data is to be returned.

Result: *anything*

The data of the object.

The "get" command will return contents or properties for specified objects.

The following example script retrieves the content of the first word in paragraph 3 for the current document.

```
tell application "JChecker3"
    get word 1 of paragraph 3 of document 1
end tell
```

The following script will copy the contents of the current document.

```
tell application "JChecker3"
    get document 1
end tell
```

"set" command

set *reference*

The object to change.

to *anything*

The new value.

You can use the "set" command broadly to set new values, contents, properties, cursor positions, highlighted areas, etc.

The following script will replace the third word of paragraph 3 with the string "abcd".

```
tell application "JChecker3"  
    set word 3 of paragraph 3 of document 1 to "abcd"  
end tell
```

The following script will position the cursor before the first word of paragraph 4

```
tell document 1 of application "JChecker3"  
    set selection to before word 1 of paragraph 4  
end tell
```

You can use the locators "before", "after" and "end" when you specify an insert point.

The following script will position the cursor at the end of paragraph 4 (before the line break).

```
tell document 1 of application "JChecker3"  
    set selection to end of paragraph 4  
end tell
```

The following script will position the cursor to the point after paragraph 4 (after the line break of that paragraph).

```
tell document 1 of application "JChecker3"  
    set selection to after paragraph 4  
end tell
```

The following script will highlight all text of the 4th paragraph.

```
tell document 1 of application "JChecker3"  
    set selection to paragraph 4  
end tell
```

The following script will specifically select lines from the first line to the third line.

```
tell document 1 of application "JChecker3"  
    set selection to line 1 thru 3  
end tell
```

"insert" command

insert *reference*

The object to be inserted.

at *location reference*

The location of where to insert the object(s).

When you want to control text editing operations of JChecker3.0, the "insert" command will be your most powerful tool. It has a wide range of uses. You can insert or replace various objects at any location you like. The "insert" command will also deal with style information in addition to text data.

You can also use locators such as "before," "beginning of," "after," and "end" when you want to specify an insert position. Without specifying an insert position, the object indicated by the "at" parameter will be replaced by the "reference" object.

The following script will add the contents of the file "new.html" to the end of the current document.

```
tell application "JChecker3"
    insert file "HD:mySite:new.html" at end of document 1
end tell
```

The following script will Insert the string "abcdefg" at the beginning of the current document.

```
tell application "JChecker3"
    insert "abcdedg" at beginning of document 1
end tell
```

The following script will replace the 3rd word with the string "abcdedg".

```
tell document 1 of application "JChecker3"
    insert "abcdedg" at word 3
end tell
```

The following script will add the content of the current document to the end of the 2nd document.

```
tell application "JChecker3"
    insert document 1 at end of document 2
end tell
```

The following script will add the 2nd word of the 3rd line from the current document to the end of the 2nd document.

```
tell application "JChecker3"
    insert word 2 of line 3 of document 1 at end of document 2
end tell
```

The following script will place the 2nd and 3rd paragraph after the last paragraph.

```
tell document 1 of application "JChecker3"
    insert (paragraph 2 Thru 3 ) at after the last paragraph
end tell
```

The following script will add a selected text area of the current document to the end of the 2nd document.

```
tell application "JChecker3"  
    insert the selection of document 1 at after the last paragraph of document 2  
end tell
```

"delete" command

delete *reference*
The element to delete.

The "delete" command will delete an object specified in "reference"

The following script will delete the second line of the current document.

```
tell application "JChecker3"  
    delete line 2 of document 1  
end tell
```

"move" command

move *reference*
The object(s) to move
to *location reference*
The new location for the object(s)

The "move" command will move a specified object in "reference" to a new location. The "move" command is only valid when moving objects within the same document. You can not move an object to other documents with this command.

The following two scripts will each move a selected object to the end of the 5th line.

```
tell document 1 of application "JChecker3"  
    move selection to line 5  
end tell  
or  
tell document 1 of application "JChecker3"  
    move selection to end of line 5  
end tell
```

The following script will move the 5th word of the 3rd paragraph to the top of the 4th paragraph.

```
tell document 1 of application "JChecker3"  
    move word 5 of paragraph 3 to before paragraph 4  
end tell
```

The following two scripts will each move a selected object to end of the current document.

```
tell application "JChecker3"  
    move selection of document 1 to end of document 1  
end tell
```

or

```
tell application "JChecker3"  
    tell document 1  
        move selection to end of it  
    end tell  
end tell
```

"find" command

find *string*

The string being searched for.

in *reference*

The object reference that you use in searching for the string.

[**case sensitive** *boolean*]

If true, will distinguish between upper and lower case.

[**zenkaku sensitive** *boolean*]

If true, will distinguish between hankaku and zenkaku characters.

[**entire word** *boolean*]

If true, will search for a one word string.

[**wrap around** *boolean*]

If true, will continue to search from the top of the document after it reaches the end of the document.

[**start at top** *boolean*]

If true, will start searching at the beginning of the document.

[**reverse search** *boolean*]

If true, will search in a reverse or backward direction (from bottom to the top).

[**grep** *boolean*]

If true, will search utilizing the regular expression mode.

[**batch** *boolean*]

If true, will find and give results in the batch mode (find all occurrences and marks them into the bookmark list)

Result: *boolean*

If the string is found, returns true.

The "find" command searches for a string in a specified object in the parameter "in". If you specify a document object in the parameter "in", JChecker will start searching from the current cursor position of the document. If an object at parameter "in" is not found within the document, JChecker will conduct a range search for that object.

The values of the option parameters under the "find" command are only valid in your script and will not affect the settings of the "Find Settings" dialog. If you omit the option parameters, the values which are defined in the "Find Settings" dialog will be used in your script instead.

If you set the parameter "batch" to true, JChecker will search for all occurrences

and any results will be listed in the mark list.

The following script will search the current document for the string "JChecker" as one word and without distinguishing between upper or lower case letters.

```
tell application "JChecker3"
    find "JChecker" in document 1 with entire word without case sensitive
end tell
or
tell application "JChecker3"
    tell document 1
        find "JChecker" in it with entire word without case sensitive
    end tell
end tell
```

Note that you should use "it" to indicate "document 1" between the "tell document 1" statement and the "end tell" statement.

The following script will search a highlighted area of the current document for the string "JChecker" as one word and without distinguishing between upper or lower case letters.

```
tell application "JChecker3"
    tell document 1
        find "JChecker" in selection with entire word without case sensitive
    end tell
end tell
```

The following script will search the highlighted area of the current document for all occurrences of the string "JChecker".

```
tell application "JChecker3"
    tell document 1
        find "JChecker" in selection with batch
    end tell
end tell
```

"replace" command

replace *string*

The string to search for and to be replaced.

to *string*

The replacement string.

in *reference*

The object reference that you use to search for the string.

[**case sensitive** *boolean*]

If true, will distinguish between upper and lower case letters.

[**zenkaku sensitive** *boolean*]

If true, will distinguish between hankaku and zenkaku characters.

[**entire word** *boolean*]

If true, will search for the string as one word.

[**grep** *boolean*]

If true, will search and find in the regular expression mode.

Result: *small integer*

The number of replacements.

The "replace" command will replace strings under the object specified in the parameter "in". You can specify objects such as document, selection, paragraph or line using the "in" parameter.

The option parameter values under the "replace" command are only valid within your script and will not affect settings contained in the "Find Settings" dialog. If you omit the option parameters, values defined under the "Find Settings" dialog will be used in your script instead.

The following script will replace the string "JChecker" with "JChecker3.0" within the highlighted text area of the current document.

```
tell application "JChecker3"
    tell document 1
        replace "JChecker" to "JChecker3.0" in selection
    end tell
end tell
```

The following script will replace the string "JChecker" to "JChecker3.0" in the current document.

```
tell application "JChecker3"
    tell document 1
        replace "JChecker" to "JChecker3.0" in it
    end tell
end tell
```

"mark" command

mark *reference*
The object to mark.

The "mark" command will list the mark in the bookmark list the location of objects.

The following script will mark the location of the 10th line of the current document.

```
tell application "JChecker3"
    tell document 1
        mark line 10
    end tell
end tell
```

The following script will mark the location of a highlighted area within the current document.

```
tell application "JChecker3"
    mark selection of document 1
end tell
```

"count" command

count *reference*

The object whose elements are to be counted.

each *type class*

The class of the elements to be counted.

Keyword 'each' is optional.

Result: *integer*

The number of elements.

The "count" command will return the size of an object using the specified class type .

The following script will return the number of words contained in the first line of the current document.

```
tell application "JChecker3"
    tell document 1
        count words of line 1
    end tell
end tell
```

or

```
tell application "JChecker3"
    tell document 1
        count line 1 each word
    end tell
end tell
```

The following script will return the number of characters contained within the highlighted area of the current document.

```
tell application "JChecker3"
    count characters of selection of document 1
end tell
```

"exists" command

exists *document object*

The document in question.

Result: *boolean*

True if it exists, false if not.

The command "exists" will determine the existence of the document object. If the document exists, it returns true under the "Result" variable.

The following script will query the existence of document 1.

```
tell application "JChecker3"
    exists document 1
end tell
```

"cut" command

cut *reference*

The object to be cut and retrieved onto the clipboard.

The "cut" command will delete the specified object and copy it onto the clipboard. The style information will also be copied. When the object you specify is a document, only the selected area of the document will be "cut" as opposed to the entire document.

Note: If JChecker is not the foremost and active application, the "cut" command will only delete the object and will not copy it into the clipboard. You should activate JChecker before you use this command.

The following two scripts will cut the selected area of the current document. Both scripts will gain the same result.

```
tell application "JChecker3"
    activate
    cut document 1
end tell
```

or

```
tell application "JChecker3"
    activate
    cut selection of document 1
end tell
```

The following script will cut the 4th line of the current document.

```
tell application "JChecker3"
    activate
    cut line 4 of document 1
end tell
```

"copy" command

copy *reference*

The source object to be copied.

[**to** *reference*]

The target location.

The command "copy" with "to" will copy the specified object into a target location. If you omit the "to" parameter, the object will be copied to the clipboard.

Note: If JChecker is not the foremost and active application, you will be unable to

copy the specified object into the clipboard. You should activate JChecker before you use this command.

The following script will copy the selected area of the current document into the clipboard as you would when you choose "Copy" under the "edit" menu.

```
tell application "JChecker3"  
    activate  
    copy document 1  
end tell
```

If you specify a location object using the parameter "to", the location object will be replaced with the source object.

The following script will replace the 3rd word of the 5th line within the current document with the string "abcdef".

```
tell document 1 of application "JChecker3"  
    copy "abcdef" to word 3 of line 5  
end tell
```

The following script will also bring the same results.

```
tell document 1 of application "JChecker3"  
    set (word 3 of line 5) to "abcdef"  
end tell
```

The following script will replace a highlighted area with the string "abcdef"

```
tell document 1 of application "JChecker3"  
    copy "abcdef" to selection  
end tell
```

"paste" command

paste *reference for document objects*

Pastes the contents residing on the clipboard, replacing the highlighted area of a specified document object. You can use this command only when pasting into document objects.

The following script will paste the contents of the clipboard, replacing the highlighted area of the current document.

```
tell application "JChecker3"
    activate
    paste document 1
end tell
```

"select" command

select *reference*

The object to select.

The command "select" will select (highlight) an area of the specified object.

The following example script will highlight the 1st to 5th line of the current document.

```
tell document 1 of application "JChecker3"
    select line 1 thru 5
end tell
```

If you want to move the cursor, use the "set selection" command instead. The following script will move the cursor to the end of the 5th line.

```
tell document 1 of application "JChecker3"
    set selection to end of line 5
end tell
```

"undo"/"redo" command

undo *reference for documents*

redo *reference for documents*

The command "undo"/"redo" will undo/redo a previous operation the same as when you choose the "Undo"/"Redo" under "edit." This command can only be

used for document objects.

The following script will undo a previous operation performed in the current document.

```
tell application "JChecker3"
    undo document 1
end tell
```

"change case" command

change case reference
to upper/lower/capitalize

Changes the character case in the reference.

```
change case document 1 to upper
change case selection of document 2 to lower
```

"preDialog"/"postDialog" command

preDialog
postDialog

You should use these commands before and after a given statement which opens dialog boxes in your script. Otherwise, JChecker's selection control of windows will possibly be in disorder once your script has completed running.

The following example uses preDialog and postDialog commands before and after the "choose folder" command.

```
preDialog
try
    copy (choose folder with prompt "Where is the folder?") to checkFolder
on error
    postDialog
    error number -128
end try
postDialog
```

Note that in the above example the "postDialog" command is used between "on error" and "error number -128" statements as well as after the "end try" statement.

"suspend" command

suspend

When you open another application from your script, it may occur where, after the script ends, you are unable to use the TSM inline conversion mode with a JChecker document. To avoid such trouble, we recommend that you use the "suspend" command before your script, which will then allow the opening of another application.

In the following script, the "suspend" command is used before statements which will allow the opening of "Internet Explorer."

```
tell application "JChecker3"
    activate
    save document 1
    copy fileSpec of document 1 to FileName
    suspend
    tell application "Internet Explorer 4.5"
        Activate
        open FileName
    end tell
end tell
```

"updateMacroMenu" command

updateMacroMenu

The "updateMacroMenu" command will update the "Macro" menu after you add or remove macro script files in the "Macro Menu Items" folder.

"inhibit redraw"/"allow redraw" command

inhibit redraw -- reference (must be a document)
allow redraw -- reference (must be a document)

The "inhibit redraw" command inhibits redrawing within document windows while the "allow redraw" command resumes redrawing. If you use these commands in some scripts that redraw the window frequently, the script execution speed will increase nearly 2 times.

For example, by inserting the "inhibit redraw" and "allow redraw" commands into the sample macro "Count Occurrence as follows, the execution will be accelerated considerably because each redrawing of find command is then suppressed.

```
.....
inhibit redraw
repeat while (find theString in it without wrap around) = true
    copy xxx + 1 to xxx
end repeat
allow redraw
.....
```

"openURL" command

openURL *reference* -- URL string

Opens a remote HTML page specified with the URL.

```
tell application "JChecker3"
    openURL "http://www.artman21.net"
end tell
```

"check" command

check *reference* -- the document to be checked

chime *boolean* -if true, JChecker notifies with sound

Result: integer--the number of errors

Validates the HTML document specified and returns an error count.

Check document 1 without sound.

```
tell application "JChecker3 "
    check document 1 without chime
end tell
```

"linkCheck" command

check *reference* -- the document to be checked

chime *boolean* -if true, JChecker notifies with sound

local *boolean*-- check / don't check local links

remote *boolean*-- check / don't check remote links

Result: none

Checks the links of the HTML document specified.

You can specify the option parameters of chime, local, and remote.

Check links of the document 1

```
tell application "JChecker3 "  
    linkCheck document 1  
end tell
```

"minimize" command

minimize *reference* -- the document to be minimized

Minimizes the HTML document.

```
tell application "JChecker3 "  
    minimize document 1  
end tell
```

"beautify" command

beautify *reference*
The document to be beautified

Beautify the specified HTML document.

```
tell application "JChecker3 "  
    beautify document 1  
end tell
```

"correct" command

correct *reference*
The document to be mended

Automatically corrects simple errors of a specified HTML document.

```
tell application "JChecker3 "  
    correct document 1  
end tell
```

"checkAll" command

checkAll *reference*
The document that the checking starts from.

[**linkDepth** integer] -- the maximum link depth of documents to be checked
[**validation** boolean] -- if true, a validation check will be conducted as well

Checks the validation and links of the specified documents and its child documents.

```
tell application "JChecker3"  
    checkAll document 1 with validation  
end tell
```

"recolor" command

recolor reference -- the document to be recolored

Executes the HTML coloring again.

```
tell application "JChecker3"  
    recolor document 1  
end tell
```

"GetTagName" command

GetTagName reference --the document to be examined

Result: string -- the element name

Returns the element name of the tag at the current cursor position. If the cursor is not in the tag, it will return an empty string.

"GetAtrbValue" command

GetAtrbName reference --the document to be examined

atrbName string -- the attribute name to check

Result: string -- the attribute value

Returns the attribute value of the specified attribute name in the tag at the current cursor position. If the cursor is not in the tag or It can't find the specified attribute, it will return an empty string.

"SelectElement" command

SelectElement *reference* -- the document object

Selects the whole element at the cursor position, i.e. the area from start tag to end tag.

"SelectTag" command

SelectTag *reference* --the document object

Result: integer -- 0: the cursor not in the tag

1: the cursor in the start tag

2: the cursor in the end tag

Selects the tag at the cursor position. If the cursor was not in a tag, it returns 0. If the cursor was not in a start tag, it returns 1. If the cursor was not in an end tag, it returns 2.

"SelectAtrb" command

SelectAtrb *reference* --the document object

atrbName string -- the attribute name

[valueOnly boolean] -- if true, only the value area will be selected

Result: integer -- 0, 1, or 2

Selects the attribute area at the cursor position. If the "valueOnly" parameter was specified, only the value area will be selected. If the attribute was not found in a tag, it returns 0. If the name and value were both selected, it returns 1. If the attribute value was omitted in the tag, it returns 2.

"GetColor" command

GetColor

[oldColor string] -- the current color string (e.g. red, #ccoocc...)

Result: string -- the HTML RGB string selected in the color pallet

Displays a color pallet and returns the selected color as an HTML hexa-decimal color string such as "#ccoocc." You can pass the current color to the pallet with the "oldColor" parameter.

"GetRelativePath" command

GetRelativePath

target file specification -- the file spec of the file in question

Result: international text -- the relative URL from the current document

Calculates the relative URL of a file specified in the target parameter. If the file of the current document is not specified, the absolute URL for the target file will be returned.

"SetAtrb" command

SetAtrb **reference** -- the document object

atrbName string -- the attribute name

atrbValue string -- the attribute value

Inserts the attribute definition into the tag at the current cursor position. If the attribute already exists , it replaces an attribute value with a new one.

"HTMLColorToRGB" command

HTMLColorToRGB **string** --HTML color string

Result: RGB color -- *the RGB value*

Converts an HTML color string such as "red" or "#COCOCO" to a RGB value that is common in the AppleScript language.

"GetDtdName" command

GetDtdName **reference** -- the document object

Result: string -- the dictionary name

Analyzes the DOCTYPE declaration and returns the dictionary name of the specified document.

"ToZenkakuKana" command

ToZenkakuKana **reference** -- object

Conver Japanese hankaku-kana letters to zenkaku-kana letters in the reference.