目录

1 Menu: 24

2 Introduction:: Getting started 24

3 Document structure:: A tree works like your brain 24

4 Tables:: Pure magic for quick formatting 24

5 Hyperlinks:: Notes in context 24

6 TODO items:: Every tree branch can be a TODO item 24

7 Tags:: Tagging headlines and matching sets of tags 24

8 Properties and columns:: Storing information about an entry 24

9 Dates and times:: Making items useful for planning 24

10 Capture - Refile - Archive:: The ins and outs for projects 24

11 Agenda views:: Collecting information into views 24

12 Markup:: Prepare text for rich export 24

13 Exporting:: Sharing and publishing notes 24

14 Publishing:: Create a web site of linked Org files 24

15 Working with source code:: Export, evaluate, and tangle code blocks 24

16 Miscellaneous:: All the rest which did not fit elsewhere 24

17 Hacking:: How to hack your way around 24

18 MobileOrg:: Viewing and capture on a mobile device 24

19 History and acknowledgments:: How Org came into being 24

20 GNU Free Documentation License:: The license for this documentation. 24

21 Main Index:: An index of Org's concepts and features 24

22 Key Index:: Key bindings and where they are described 24

23 Command and Function Index:: Command names and some internal functions 24

24 Variable Index:: Variables mentioned in the manual 24

25 Summary:: Brief summary of what Org does 25

26 Installation:: Installing Org 25

27 Activation:: How to activate Org for certain buffers 25

28 Feedback:: Bug reports, ideas, patches etc. 25

29 Conventions:: Typesetting conventions in the manual 25

30 Outlines:: Org is based on Outline mode 27

31 Headlines:: How to typeset Org tree headlines 27

32 Visibility cycling:: Show and hide, much simplified 27

33 Motion:: Jumping to other headlines 27

34 Structure editing:: Changing sequence and level of headlines 27

35 Sparse trees:: Matches embedded in context 27

36 Plain lists:: Additional structure within an entry 27

37 Drawers:: Tucking stuff away 27

38 Blocks:: Folding blocks 27

39 Footnotes:: How footnotes are defined in Org's syntax 27

40 Orgstruct mode:: Structure editing outside Org 27

41 Org syntax:: Formal description of Org's syntax 27

42 Global and local cycling:: Cycling through various visibility states 28

43 Initial visibility:: Setting the initial visibility state 28

44 Catching invisible edits:: Preventing mistakes when editing invisible parts 28

45 Built-in table editor:: Simple tables 28

46 Column width and alignment:: Overrule the automatic settings 28

47 Column groups:: Grouping to trigger vertical lines 28

48 Orgtbl mode:: The table editor as minor mode 28

49 The spreadsheet:: The table editor has spreadsheet capabilities 28

50 Org-Plot:: Plotting from org tables 28

51 References:: How to refer to another field or range 30

52 Formula syntax for Calc:: Using Calc to compute stuff 30

53 Formula syntax for Lisp:: Writing formulas in Emacs Lisp 30

54 Durations and time values:: How to compute durations and time values 30

55 Field and range formulas:: Formula for specific (ranges of) fields 30

56 Column formulas:: Formulas valid for an entire column 30

57 Lookup functions:: Lookup functions for searching tables 30

58 Editing and debugging formulas:: Fixing formulas 30

59 Updating the table:: Recomputing all dependent fields 30

60 Advanced features:: Field and column names, parameters and automatic recalc 30

61 Link format:: How links in Org are formatted 31

62 Internal links:: Links to other places in the current file 31

63 External links:: URL-like links to the world 31

64 Handling links:: Creating, inserting and following 31

65 Using links outside Org:: Linking from my C source code? 31

66 Link abbreviations:: Shortcuts for writing complex links 31

67 Search options:: Linking to a specific location 31

68 Custom searches:: When the default search is not enough 31

69 Radio targets:: Make targets trigger links in plain text 31

70 TODO basics:: Marking and displaying TODO entries 32

71 TODO extensions:: Workflow and assignments 32

72 Progress logging:: Dates and notes for progress 32

73 Priorities:: Some things are more important than others 32

74 Breaking down tasks:: Splitting a task into manageable pieces 32

75 Checkboxes:: Tick-off lists 32

76 Workflow states:: From TODO to DONE in steps 33

77 TODO types:: I do this, Fred does the rest 33

78 Multiple sets in one file:: Mixing it all, and still finding your way 33

79 Fast access to TODO states:: Single letter selection of a state 33

80 Per-file keywords:: Different files, different requirements 33

81 Faces for TODO keywords:: Highlighting states 33

82 TODO dependencies:: When one task needs to wait for others 33

83 Closing items:: When was this entry marked DONE? 34

84 Tracking TODO state changes:: When did the status change? 34

85 Tracking your habits:: How consistent have you been? 34

86 Tag inheritance:: Tags use the tree structure of the outline 34

87 Setting tags:: How to assign tags to a headline 34

88 Tag hierarchy:: Create a hierarchy of tags 34

89 Tag searches:: Searching for combinations of tags 34

90 Property syntax:: How properties are spelled out 35

91 Special properties:: Access to other Org mode features 35

92 Property searches:: Matching property values 35

93 Property inheritance:: Passing values down the tree 35

94 Column view:: Tabular viewing and editing 35

95 Property API:: Properties for Lisp programmers 35

96 Defining columns:: The COLUMNS format property 35

97 Using column view:: How to create and use column view 35

98 Capturing column view:: A dynamic block for column view 35

99 Scope of column definitions:: Where defined, where valid? 36

100 Column attributes:: Appearance and content of a column 36

101 Timestamps:: Assigning a time to a tree entry 36

102 Creating timestamps:: Commands which insert timestamps 36

103 Deadlines and scheduling:: Planning your work 36

104 Clocking work time:: Tracking how long you spend on a task 36

105 Effort estimates:: Planning work effort in advance 36

106 Timers:: Notes with a running timer 36

107 The date/time prompt:: How Org mode helps you entering date and time 37

108 Custom time format:: Making dates look different 37

109 Inserting deadline/schedule:: Planning items 37

110 Repeated tasks:: Items that show up again and again 37

111 Clocking commands:: Starting and stopping a clock 37

112 The clock table:: Detailed reports 37

113 Resolving idle time:: Resolving time when you've been idle 37

114 Capture:: Capturing new stuff 38

115 Attachments:: Add files to tasks 38

116 RSS feeds:: Getting input from RSS feeds 38

117 Protocols:: External (e.g., Browser) access to Emacs and Org 38

118 Refile and copy:: Moving/copying a tree from one place to another 38

119 Archiving:: What to do with finished projects 38

120 Setting up capture:: Where notes will be stored 38

121 Using capture:: Commands to invoke and terminate capture 38

122 Capture templates:: Define the outline of different note types 38

123 Template elements:: What is needed for a complete template entry 39

124 Template expansion:: Filling in information about time and context 39

125 Templates in contexts:: Only show a template in a specific context 39

126 Moving subtrees:: Moving a tree to an archive file 39

127 Internal archiving:: Switch off a tree but keep it in the file 39

128 Agenda files:: Files being searched for agenda information 40

129 Agenda dispatcher:: Keyboard access to agenda views 40

130 Built-in agenda views:: What is available out of the box? 40

131 Presentation and sorting:: How agenda items are prepared for display 40

132 Agenda commands:: Remote editing of Org trees 40

133 Custom agenda views:: Defining special searches and views 40

134 Exporting agenda views:: Writing a view to a file 40

135 Agenda column view:: Using column view for collected entries 40

136 Weekly/daily agenda:: The calendar page with current tasks 41

137 Global TODO list:: All unfinished action items 41

138 Matching tags and properties:: Structured information with fine-tuned search 41

139 Timeline:: Time-sorted view for single file 41

140 Search view:: Find entries by searching for text 41

141 Stuck projects:: Find projects you need to review 41

142 Categories:: Not all tasks are equal 42

143 Time-of-day specifications:: How the agenda knows the time 42

144 Sorting agenda items:: The order of things 42

145 Filtering/limiting agenda items:: Dynamically narrow the agenda 42

146 Storing searches:: Type once, use often 42

147 Block agenda:: All the stuff you need in a single buffer 42

148 Setting options:: Changing the rules 42

149 Paragraphs:: The basic unit of text 43

150 Emphasis and monospace:: Bold, italic, etc. 43

151 Horizontal rules:: Make a line 43

152 Images and tables:: Images, tables and caption mechanism 43

153 Literal examples:: Source code examples with special formatting 43

154 Special symbols:: Greek letters and other symbols 43

155 Subscripts and superscripts:: Simple syntax for raising/lowering text 43

156 Embedded LATEX:: LATEX can be freely used inside Org documents 43

157 LATEX fragments:: Complex formulas made easy 44

158 Previewing LATEX fragments:: What will this snippet look like? 44

159 CDLaTeX mode:: Speed up entering of formulas 44

160 The export dispatcher:: The main exporter interface 46

161 Export settings:: Generic export settings 46

162 Table of contents:: The if and where of the table of contents 46

163 Include files:: Include additional files into a document 46

164 Macro replacement:: Use macros to create templates 46

165 Comment lines:: What will not be exported 46

166 ASCII/Latin-1/UTF-8 export:: Exporting to flat files with encoding 46

167 Beamer export:: Exporting as a Beamer presentation 46

168 HTML export:: Exporting to HTML 46

169 LaTeX export:: Exporting to LaTeX, and processing to PDF 46

170 Markdown export:: Exporting to Markdown 46

171 OpenDocument Text export:: Exporting to OpenDocument Text 46

172 Org export:: Exporting to Org 46

173 Texinfo export:: Exporting to Texinfo 46

174 iCalendar export:: Exporting to iCalendar 46

175 Other built-in back-ends:: Exporting to a man page 46

176 Advanced configuration:: Fine-tuning the export output 46

177 Export in foreign buffers:: Author tables and lists in Org syntax 46

178 Beamer export commands:: How to export Beamer documents. 47

179 Beamer specific export settings:: Export settings for Beamer export. 47

180 Sectioning Frames and Blocks in Beamer:: Blocks and sections in Beamer. 47

181 Beamer specific syntax:: Syntax specific to Beamer. 47

182 Editing support:: Helper functions for Org Beamer export. 47

183 A Beamer Example:: An complete Beamer example. 47

184 HTML Export commands:: How to invoke HTML export 49

185 HTML Specific export settings:: Export settings for HTML export 49

186 HTML doctypes:: Org can export to various (X)HTML flavors 49

187 HTML preamble and postamble:: How to insert a preamble and a postamble 49

188 Quoting HTML tags:: Using direct HTML in Org mode 49

189 Links in HTML export:: How links will be interpreted and formatted 49

190 Tables in HTML export:: How to modify the formatting of tables 49

191 Images in HTML export:: How to insert figures into HTML output 49

192 Math formatting in HTML export:: Beautiful math also on the web 49

193 Text areas in HTML export:: An alternative way to show an example 49

194 CSS support:: Changing the appearance of the output 49

195 JavaScript support:: Info and Folding in a web browser 49

196 LaTeX export commands:: How to export to LaTeX and PDF 51

197 LaTeX specific export settings:: Export settings for LaTeX 51

198 LaTeX header and sectioning:: Setting up the export file structure 51

199 Quoting LaTeX code:: Incorporating literal LaTeX code 51

200 Tables in LaTeX export:: Specific attributes for tables 51

201 Images in LaTeX export:: Specific attributes for images 51

202 Plain lists in LaTeX export:: Specific attributes for plain lists 51

203 Source blocks in LaTeX export:: Specific attributes for source blocks 51

204 Example blocks in LaTeX export:: Specific attributes for example blocks 51

205 Special blocks in LaTeX export:: Specific attributes for special blocks 51

206 Horizontal rules in LaTeX export:: Specific attributes for horizontal rules 51

207 Pre-requisites for ODT export:: What packages ODT exporter relies on 53

208 ODT export commands:: How to invoke ODT export 53

209 ODT specific export settings:: Export settings for ODT 53

210 Extending ODT export:: How to produce ‘doc’, ‘pdf’ files 53

211 Applying custom styles:: How to apply custom styles to the output 53

212 Links in ODT export:: How links will be interpreted and formatted 53

213 Tables in ODT export:: How Tables are exported 53

214 Images in ODT export:: How to insert images 53

215 Math formatting in ODT export:: How LaTeX fragments are formatted 53

216 Labels and captions in ODT export:: How captions are rendered 53

217 Literal examples in ODT export:: How source and example blocks are formatted 53

218 Advanced topics in ODT export:: Read this if you are a power user 53

219 Working with LaTeX math snippets:: How to embed LaTeX math fragments 54

220 Working with MathML or OpenDocument formula files:: How to embed equations in native format 54

221 Configuring a document converter:: How to register a document converter 54

222 Working with OpenDocument style files:: Explore the internals 54

223 Creating one-off styles:: How to produce custom highlighting etc 54

224 Customizing tables in ODT export:: How to define and use Table templates 54

225 Validating OpenDocument XML:: How to debug corrupt OpenDocument files 54

226 Texinfo export commands:: How to invoke Texinfo export 56

227 Texinfo specific export settings:: Export settings for Texinfo 56

228 Texinfo file header:: Generating the begining of a Texinfo file 56

229 Texinfo title and copyright page:: Creating title and copyright pages 56

230 Texinfo ‘Top’ node:: Installing a manual in Info Top node 56

231 Headings and sectioning structure:: Building document structure 56

232 Indices:: Creating indices 56

233 Quoting Texinfo code:: Incorporating literal Texinfo code 56

234 Plain lists in Texinfo export:: Specific attributes for plain lists 56

235 Tables in Texinfo export:: Specific attributes for tables 56

236 Images in Texinfo export:: Specific attributes for images 56

237 Special blocks in Texinfo export:: Specific attributes for special blocks 56

238 A Texinfo example:: Illustrating Org to Texinfo process 56

239 Configuration:: Defining projects 57

240 Uploading files:: How to get files up on the server 57

241 Sample configuration:: Example projects 57

242 Triggering publication:: Publication commands 57

243 Project alist:: The central configuration variable 58

244 Sources and destinations:: From here to there 58

245 Selecting files:: What files are part of the project? 58

246 Publishing action:: Setting the function doing the publishing 58

247 Publishing options:: Tweaking HTML/LaTeX export 58

248 Publishing links:: Which links keep working after publishing? 58

249 Sitemap:: Generating a list of all pages 58

250 Generating an index:: An index that reaches across pages 58

251 Simple example:: One-component publishing 59

252 Complex example:: A multi-component publishing example 59

253 Structure of code blocks:: Code block syntax described 61

254 Editing source code:: Language major-mode editing 61

255 Exporting code blocks:: Export contents and/or results 61

256 Extracting source code:: Create pure source code files 61

257 Evaluating code blocks:: Place results of evaluation in the Org mode buffer 61

258 Library of Babel:: Use and contribute to a library of useful code blocks 61

259 Languages:: List of supported code block languages 61

260 Header arguments:: Configure code block functionality 61

261 Results of evaluation:: How evaluation results are handled 61

262 Noweb reference syntax:: Literate programming in Org mode 61

263 Key bindings and useful functions:: Work quickly with code blocks 61

264 Batch execution:: Call functions from the command line 61

265 Using header arguments:: Different ways to set header arguments 62

266 Specific header arguments:: List of header arguments 62

267 System-wide header arguments:: Set globally, language-specific 62

268 Language-specific header arguments:: Set in the Org file's headers 62

269 Header arguments in Org mode properties:: Set in the Org file 62

270 Language-specific mode properties:: 62

271 Code block specific header arguments:: The most commonly used method 62

272 Arguments in function calls:: The most specific level, takes highest priority 62

273 var:: Pass arguments to 'src' code blocks 64

274 results:: Specify results type; how to collect 64

275 file:: Specify a path for output file 64

276 file-desc:: Specify a description for file results 64

277 file-ext:: Specify an extension for file output 64

278 output-dir:: Specify a directory for output file 64

279 dir:: Specify the default directory for code block execution 64

280 exports:: Specify exporting code, results, both, none 64

281 tangle:: Toggle tangling; or specify file name 64

282 mkdirp:: Toggle for parent directory creation for target files during tangling 64

283 comments:: Toggle insertion of comments in tangled code files 64

284 padline:: Control insertion of padding lines in tangled code files 64

285 no-expand:: Turn off variable assignment and noweb expansion during tangling 64

286 session:: Preserve the state of code evaluation 64

287 noweb:: Toggle expansion of noweb references 64

288 noweb-ref:: Specify block's noweb reference resolution target 64

289 noweb-sep:: String to separate noweb references 64

290 cache:: Avoid re-evaluating unchanged code blocks 64

291 sep:: Delimiter for writing tabular results outside Org 64

292 hlines:: Handle horizontal lines in tables 64

293 colnames:: Handle column names in tables 64

294 rownames:: Handle row names in tables 64

295 shebang:: Make tangled files executable 64

296 tangle-mode:: Set permission of tangled files 64

297 eval:: Limit evaluation of specific code blocks 64

298 wrap:: Mark source block evaluation results 64

299 post:: Post processing of results of code block evaluation 64

300 prologue:: Text to prepend to body of code block 64

301 epilogue:: Text to append to body of code block 64

302 Completion:: M-TAB guesses completions 66

303 Easy templates:: Quick insertion of structural elements 66

304 Speed keys:: Electric commands at the beginning of a headline 66

305 Code evaluation security:: Org mode files evaluate inline code 66

306 Customization:: Adapting Org to changing tastes 66

307 In-buffer settings:: Overview of the #+KEYWORDS 66

308 The very busy C-c C-c key:: When in doubt, press C-c C-c 66

309 Clean view:: Getting rid of leading stars in the outline 66

310 TTY keys:: Using Org on a tty 66

311 Interaction:: With other Emacs packages 66

312 org-crypt:: Encrypting Org files 66

313 Cooperation:: Packages Org cooperates with 67

314 Conflicts:: Packages that lead to conflicts 67

315 Hooks:: How to reach into Org's internals 69

316 Add-on packages:: Available extensions 69

317 Adding hyperlink types:: New custom link types 69

318 Adding export back-ends:: How to write new export back-ends 69

319 Context-sensitive commands:: How to add functionality to such commands 69

320 Tables in arbitrary syntax:: Orgtbl for LaTeX and other programs 69

321 Dynamic blocks:: Automatically filled blocks 69

322 Special agenda views:: Customized views 69

323 Speeding up your agendas:: Tips on how to speed up your agendas 69

324 Extracting agenda information:: Post-processing of agenda information 69

325 Using the property API:: Writing programs that use entry properties 69

326 Using the mapping API:: Mapping over all or selected entries 69

327 Radio tables:: Sending and receiving radio tables 70

328 A LaTeX example:: Step by step, almost a tutorial 70

329 Translator functions:: Copy and modify 70

330 Radio lists:: Sending and receiving lists 70

331 Setting up the staging area:: For the mobile device 70

332 Pushing to MobileOrg:: Uploading Org files and agendas 70

333 Pulling from MobileOrg:: Integrating captured and flagged items 70

File: org, Node: Top, Next: Introduction, Prev: (dir), Up: (dir)

Org Mode Manual *

This manual is for Org version 9.0.5 (release$_{9.0.5}$).

Copyright (C) 2004-2017 Free Software Foundation, Inc.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, with the Front-Cover Texts being "A GNU Manual," and with the Back-Cover Texts as in (a) below. A copy of the license is included in the section entitled "GNU Free Documentation License."

(a) The FSF's Back-Cover Text is: "You have the freedom to copy and modify this GNU manual."

1 Menu:

2 Introduction:: Getting started

3 Document structure:: A tree works like your brain

4 Tables:: Pure magic for quick formatting

5 Hyperlinks:: Notes in context

6 TODO items:: Every tree branch can be a TODO item

7 Tags:: Tagging headlines and matching sets of tags

8 Properties and columns:: Storing information about an entry

9 Dates and times:: Making items useful for planning

10 Capture - Refile - Archive:: The ins and outs for projects

11 Agenda views:: Collecting information into views

12 Markup:: Prepare text for rich export

13 Exporting:: Sharing and publishing notes

14 Publishing:: Create a web site of linked Org files

Introduction

25 Summary:: Brief summary of what Org does

26 Installation:: Installing Org

27 Activation:: How to activate Org for certain buffers

28 Feedback:: Bug reports, ideas, patches etc.

29 Conventions:: Typesetting conventions in the manual

Document structure

30 Outlines:: Org is based on Outline mode

31 Headlines:: How to typeset Org tree headlines

32 Visibility cycling:: Show and hide, much simplified

33 Motion:: Jumping to other headlines

34 Structure editing:: Changing sequence and level of headlines

35 Sparse trees:: Matches embedded in context

36 Plain lists:: Additional structure within an entry

37 Drawers:: Tucking stuff away

38 Blocks:: Folding blocks

39 Footnotes:: How footnotes are defined in Org's syntax

40 Orgstruct mode:: Structure editing outside Org

41 Org syntax:: Formal description of Org's syntax

Visibility cycling

42 Global and local cycling:: Cycling through various visibility states

43 Initial visibility:: Setting the initial visibility state

44 Catching invisible edits:: Preventing mistakes when editing invisible parts

Tables

45 Built-in table editor:: Simple tables

46 Column width and alignment:: Overrule the automatic settings

47 Column groups:: Grouping to trigger vertical lines

48 Orgtbl mode:: The table editor as minor mode

49 The spreadsheet:: The table editor has spreadsheet capabilities

50 Org-Plot:: Plotting from org tables

The spreadsheet

51 References:: How to refer to another field or range

52 Formula syntax for Calc:: Using Calc to compute stuff

53 Formula syntax for Lisp:: Writing formulas in Emacs Lisp

54 Durations and time values:: How to compute durations and time values

55 Field and range formulas:: Formula for specific (ranges of) fields

56 Column formulas:: Formulas valid for an entire column

57 Lookup functions:: Lookup functions for searching tables

58 Editing and debugging formulas:: Fixing formulas

59 Updating the table:: Recomputing all dependent fields

60 Advanced features:: Field and column names, parameters and automatic recalc

Hyperlinks

61 Link format:: How links in Org are formatted

62 Internal links:: Links to other places in the current file

63 External links:: URL-like links to the world

64 Handling links:: Creating, inserting and following

65 Using links outside Org:: Linking from my C source code?

66 Link abbreviations:: Shortcuts for writing complex links

67 Search options:: Linking to a specific location

68 Custom searches:: When the default search is not enough

Internal links

69 Radio targets:: Make targets trigger links in plain text

TODO items

70 TODO basics:: Marking and displaying TODO entries

71 TODO extensions:: Workflow and assignments

72 Progress logging:: Dates and notes for progress

73 Priorities:: Some things are more important than others

74 Breaking down tasks:: Splitting a task into manageable pieces

75 Checkboxes:: Tick-off lists

Extended use of TODO keywords

76 Workflow states:: From TODO to DONE in steps

77 **TODO** types:: I do this, Fred does the rest

78 Multiple sets in one file:: Mixing it all, and still finding your way

79 Fast access to TODO states:: Single letter selection of a state

80 Per-file keywords:: Different files, different requirements

81 Faces for TODO keywords:: Highlighting states

82 **TODO** dependencies:: When one task needs to wait for others

Progress logging

83 Closing items:: When was this entry marked DONE?

84 Tracking TODO state changes:: When did the status change?

85 Tracking your habits:: How consistent have you been?

Tags

86 Tag inheritance:: Tags use the tree structure of the outline

87 Setting tags:: How to assign tags to a headline

88 Tag hierarchy:: Create a hierarchy of tags

89 Tag searches:: Searching for combinations of tags

Properties and columns

90 Property syntax:: How properties are spelled out

91 Special properties:: Access to other Org mode features

92 Property searches:: Matching property values

93 Property inheritance:: Passing values down the tree

94 Column view:: Tabular viewing and editing

95 Property API:: Properties for Lisp programmers

Column view

96 Defining columns:: The COLUMNS format property

97 Using column view:: How to create and use column view

98 Capturing column view:: A dynamic block for column view

Defining columns

99 Scope of column definitions:: Where defined, where valid?

100 Column attributes:: Appearance and content of a column

Dates and times

101 Timestamps:: Assigning a time to a tree entry

102 Creating timestamps:: Commands which insert timestamps

103 Deadlines and scheduling:: Planning your work

104 Clocking work time:: Tracking how long you spend on a task

105 Effort estimates:: Planning work effort in advance

106 Timers:: Notes with a running timer

Creating timestamps

107 The date/time prompt:: How Org mode helps you entering date and time

108 Custom time format:: Making dates look different

Deadlines and scheduling

109 Inserting deadline/schedule:: Planning items

110 Repeated tasks:: Items that show up again and again

Clocking work time

111 Clocking commands:: Starting and stopping a clock

112 The clock table:: Detailed reports

113 Resolving idle time:: Resolving time when you've been idle

Capture - Refile - Archive

114 Capture:: Capturing new stuff

115 Attachments:: Add files to tasks

116 RSS feeds:: Getting input from RSS feeds

117 Protocols:: External (e.g., Browser) access to Emacs and Org

118 Refile and copy:: Moving/copying a tree from one place to another

119 Archiving:: What to do with finished projects

Capture

120 Setting up capture:: Where notes will be stored

121 Using capture:: Commands to invoke and terminate capture

122 Capture templates:: Define the outline of different note types

Capture templates

123 Template elements:: What is needed for a complete template entry

124 Template expansion:: Filling in information about time and context

125 Templates in contexts:: Only show a template in a specific context

Archiving

126 Moving subtrees:: Moving a tree to an archive file

127 Internal archiving:: Switch off a tree but keep it in the file

Agenda views

128 Agenda files:: Files being searched for agenda information

129 Agenda dispatcher:: Keyboard access to agenda views

130 Built-in agenda views:: What is available out of the box?

131 Presentation and sorting:: How agenda items are prepared for display

132 Agenda commands:: Remote editing of Org trees

133 Custom agenda views:: Defining special searches and views

134 Exporting agenda views:: Writing a view to a file

135 Agenda column view:: Using column view for collected entries

The built-in agenda views

136 Weekly/daily agenda:: The calendar page with current tasks

137 Global TODO list:: All unfinished action items

138 Matching tags and properties:: Structured information with fine-tuned search

139 Timeline:: Time-sorted view for single file

140 Search view:: Find entries by searching for text

141 Stuck projects:: Find projects you need to review

Presentation and sorting

142 Categories:: Not all tasks are equal

143 Time-of-day specifications:: How the agenda knows the time

144 Sorting agenda items:: The order of things

145 Filtering/limiting agenda items:: Dynamically narrow the agenda

Custom agenda views

146 Storing searches:: Type once, use often

147 Block agenda:: All the stuff you need in a single buffer

148 Setting options:: Changing the rules

Markup for rich export

149 Paragraphs:: The basic unit of text

150 Emphasis and monospace:: Bold, italic, etc.

151 Horizontal rules:: Make a line

152 Images and tables:: Images, tables and caption mechanism

153 Literal examples:: Source code examples with special formatting

154 Special symbols:: Greek letters and other symbols

155 Subscripts and superscripts:: Simple syntax for raising/lowering text

156 Embedded LaTeX:: LaTeX can be freely used inside Org documents

Embedded LaTeX

157 LaTeX fragments:: Complex formulas made easy

158 Previewing LaTeX fragments:: What will this snippet look like?

159 CDLaTeX mode:: Speed up entering of formulas

Exporting

160 The export dispatcher:: The main exporter interface

161 Export settings:: Generic export settings

162 Table of contents:: The if and where of the table of contents

163 Include files:: Include additional files into a document

164 Macro replacement:: Use macros to create templates

165 Comment lines:: What will not be exported

166 ASCII/Latin-1/UTF-8 export:: Exporting to flat files with encoding

167 Beamer export:: Exporting as a Beamer presentation

168 HTML export:: Exporting to HTML

169 LaTeX export:: Exporting to LaTeX, and processing to PDF

170 Markdown export:: Exporting to Markdown

171 OpenDocument Text export:: Exporting to OpenDocument Text

172 Org export:: Exporting to Org

173 Texinfo export:: Exporting to Texinfo

178 Beamer export commands:: How to export Beamer documents.

179 Beamer specific export settings:: Export settings for Beamer export.

180 Sectioning Frames and Blocks in Beamer:: Blocks and sections in Beamer.

181 Beamer specific syntax:: Syntax specific to Beamer.

182 Editing support:: Helper functions for Org Beamer export.

183 A Beamer Example:: An complete Beamer example.

HTML export

184 HTML Export commands:: How to invoke HTML export

185 HTML Specific export settings:: Export settings for HTML export

186 HTML doctypes:: Org can export to various (X)HTML flavors

187 HTML preamble and postamble:: How to insert a preamble and a postamble

188 Quoting HTML tags:: Using direct HTML in Org mode

189 Links in HTML export:: How links will be interpreted and formatted

190 Tables in HTML export:: How to modify the formatting of tables

191 Images in HTML export:: How to insert figures into HTML output

192 Math formatting in HTML export:: Beautiful math also on the web

193 Text areas in HTML export:: An alternative way to show an example

194 CSS support:: Changing the appearance of the output

195 JavaScript support:: Info and Folding in a web browser

196 LaTeX export commands:: How to export to LaTeX and PDF

197 LaTeX specific export settings:: Export settings for LaTeX

198 LaTeX header and sectioning:: Setting up the export file structure

199 Quoting LaTeX code:: Incorporating literal LaTeX code

200 Tables in LaTeX export:: Specific attributes for tables

201 Images in LaTeX export:: Specific attributes for images

202 Plain lists in LaTeX export:: Specific attributes for plain lists

203 Source blocks in LaTeX export:: Specific attributes for source blocks

204 Example blocks in LaTeX export:: Specific attributes for example blocks

205 Special blocks in LaTeX export:: Specific attributes for special blocks

206 Horizontal rules in LaTeX export:: Specific attributes for horizontal rules

OpenDocument Text export

207 Pre-requisites for ODT export:: What packages ODT exporter relies on

208 ODT export commands:: How to invoke ODT export

209 ODT specific export settings:: Export settings for ODT

210 Extending ODT export:: How to produce 'doc', 'pdf' files

211 Applying custom styles:: How to apply custom styles to the output

212 Links in ODT export:: How links will be interpreted and formatted

213 Tables in ODT export:: How Tables are exported

214 Images in ODT export:: How to insert images

215 Math formatting in ODT export:: How LaTeX fragments are formatted

216 Labels and captions in ODT export:: How captions are rendered

217 Literal examples in ODT export:: How source and example blocks are formatted

218 Advanced topics in ODT export:: Read this if you are a power user

219 Working with LaTeX math snippets:: How to embed LaTeX math fragments

220 Working with MathML or OpenDocument formula files:: How to embed equations in native format

Advanced topics in ODT export

221 Configuring a document converter:: How to register a document converter

222 Working with OpenDocument style files:: Explore the internals

223 Creating one-off styles:: How to produce custom highlighting etc

224 Customizing tables in ODT export:: How to define and use Table templates

225 Validating OpenDocument XML:: How to debug corrupt OpenDocument files

Texinfo export

226 Texinfo export commands:: How to invoke Texinfo export

227 Texinfo specific export settings:: Export settings for Texinfo

228 Texinfo file header:: Generating the begining of a Texinfo file

229 Texinfo title and copyright page:: Creating title and copyright pages

230 Texinfo ‘Top’ node:: Installing a manual in Info Top node

231 Headings and sectioning structure:: Building document structure

232 Indices:: Creating indices

233 Quoting Texinfo code:: Incorporating literal Texinfo code

234 Plain lists in Texinfo export:: Specific attributes for plain lists

235 Tables in Texinfo export:: Specific attributes for tables

236 Images in Texinfo export:: Specific attributes for images

237 Special blocks in Texinfo export:: Specific attributes for special blocks

239 Configuration:: Defining projects

240 Uploading files:: How to get files up on the server

241 Sample configuration:: Example projects

242 Triggering publication:: Publication commands

Configuration

243 Project alist:: The central configuration variable

244 Sources and destinations:: From here to there

245 Selecting files:: What files are part of the project?

246 Publishing action:: Setting the function doing the publishing

247 Publishing options:: Tweaking HTML/LaTeX export

248 Publishing links:: Which links keep working after publishing?

249 Sitemap:: Generating a list of all pages

250 Generating an index:: An index that reaches across pages

Sample configuration

251 Simple example:: One-component publishing

252 Complex example:: A multi-component publishing example

Working with source code

253 Structure of code blocks:: Code block syntax described

254 Editing source code:: Language major-mode editing

255 Exporting code blocks:: Export contents and/or results

256 Extracting source code:: Create pure source code files

257 Evaluating code blocks:: Place results of evaluation in the Org mode buffer

258 Library of Babel:: Use and contribute to a library of useful code blocks

259 Languages:: List of supported code block languages

260 Header arguments:: Configure code block functionality

261 Results of evaluation:: How evaluation results are handled

262 Noweb reference syntax:: Literate programming in Org mode

263 Key bindings and useful functions:: Work quickly with code blocks

264 Batch execution:: Call functions from the command line

265 Using header arguments:: Different ways to set header arguments

266 Specific header arguments:: List of header arguments

Using header arguments

267 System-wide header arguments:: Set globally, language-specific

268 Language-specific header arguments:: Set in the Org file's headers

269 Header arguments in Org mode properties:: Set in the Org file

270 Language-specific mode properties::

271 Code block specific header arguments:: The most commonly used method

272 Arguments in function calls:: The most specific level, takes highest priority

Specific header arguments

273 var:: Pass arguments to ‘src’ code blocks

274 results:: Specify results type; how to collect

275 file:: Specify a path for output file

276 file-desc:: Specify a description for file results

277 file-ext:: Specify an extension for file output

278 output-dir:: Specify a directory for output file

279 dir:: Specify the default directory for code block execution

280 exports:: Specify exporting code, results, both, none

281 tangle:: Toggle tangling; or specify file name

282 mkdirp:: Toggle for parent directory creation for target files during tangling

283 comments:: Toggle insertion of comments in tangled code files

284 padline:: Control insertion of padding lines in tangled code files

285 no-expand:: Turn off variable assignment and noweb expansion during tangling

286 session:: Preserve the state of code evaluation

302 Completion:: M-TAB guesses completions

303 Easy templates:: Quick insertion of structural elements

304 Speed keys:: Electric commands at the beginning of a headline

305 Code evaluation security:: Org mode files evaluate inline code

306 Customization:: Adapting Org to changing tastes

307 In-buffer settings:: Overview of the #+KEYWORDS

308 The very busy C-c C-c key:: When in doubt, press C-c C-c

309 Clean view:: Getting rid of leading stars in the outline

310 TTY keys:: Using Org on a tty

311 Interaction:: With other Emacs packages

312 org-crypt:: Encrypting Org files

Interaction with other packages

313 Cooperation:: Packages Org cooperates with

314 Conflicts:: Packages that lead to conflicts

Hacking

315 Hooks:: How to reach into Org's internals

316 Add-on packages:: Available extensions

317 Adding hyperlink types:: New custom link types

318 Adding export back-ends:: How to write new export back-ends

319 Context-sensitive commands:: How to add functionality to such commands

320 Tables in arbitrary syntax:: Orgtbl for LaTeX and other programs

321 Dynamic blocks:: Automatically filled blocks

322 Special agenda views:: Customized views

323 Speeding up your agendas:: Tips on how to speed up your agendas

324 Extracting agenda information:: Post-processing of agenda information

325 Using the property API:: Writing programs that use entry properties

326 Using the mapping API:: Mapping over all or selected entries

Tables and lists in arbitrary syntax

327 Radio tables:: Sending and receiving radio tables

328 A LaTeX example:: Step by step, almost a tutorial

329 Translator functions:: Copy and modify

330 Radio lists:: Sending and receiving lists

MobileOrg

331 Setting up the staging area:: For the mobile device

332 Pushing to MobileOrg:: Uploading Org files and agendas

333 Pulling from MobileOrg:: Integrating captured and flagged items